Handwritten Hangul Recognition by Random Hypergraphs: Random Graphs with High-Order Dependency Modeling

Kyung-Won Kang and Jin-Hyung Kim
CS Div., EECS Dept., KAIST, 373-1, Kusong-Dong, Yusong-Gu, Taejon #305-701 KOREA
E-mail: kwkang@ai.kaist.ac.kr, jkim@ai.kaist.ac.kr

Abstract

A random graph based modeling method is useful for the recognition of oriental languages such as Hangul and Chinese in that it can represent a structural feature in a statistical manner. Unfortunately, a random graph based recognition system suffers from high time complexity and weak power in dependency modeling. To overcome these problems, a new notion of a random hypergraph is introduced. The pairwise dependency modeling in a random graph is generalized to the high-order dependency modeling in a random hypergraph. In addition, a random hypergraph based handwritten Hangul recognition system is proposed and its performance is compared to that of a random graph based system.

Figure 1. Difficulty of grapheme segmentation in Hangul recognition

1 Introduction

Offline character recognition is a representative pattern recognition problem which exhibits human cognitive capability. It is composed of optical character recognition (OCR) and handwritten character recognition according to writing styles. Through a few decades of researches, OCR is now being practically applied to commercial domains. Several techniques including artificial intelligience, pattern matching, neural networks, hidden Markov models, and linguistic postprocessing make OCR to be effectively used. On the other hand, handwritten character recognition is not so mature as OCR is. It is because of the shape variation which is caused by different writers, different writing instuments, and so on. It is generally believed that most state-of-the-art systems work only in a laboratory environment and their application to real world problems is restricted.

Approaches to handwritten character recognition are classified into structural methods, statistical methods, and their hybrid ones. Structural methods are based on strokes which have conceptual meaning to human being and can recognize character patterns by finding stroke correspondence and a relationship between a class pattern (model pattern) and an input pattern. Generally, their performance is affected by image preprocessing algorithms such as stroke extraction and thinning, which are also regarded as difficult problems. On the other hand, statistical methods are based on statistical pattern analysis theories such as a Bayes theory, neural networks, hidden Markov models, k-nearest neighbor classification and so on. The main deficiency of this approach is that the performance of the system is dependent heavily on features used. Unfortunately, it is well known that it is a difficult job to select features suitable for each problem domain. To compromise these two approaches, recent systems adopt the hybrid methods such as random graph matching and concatenation of statistical methods and structural methods. Among these hybrid systems, random graph based modeling seems to be relevant to Hangul and Chinese character recognition in that it can represent a structural feature in a statistical manner.

Handwritten Hangul character recognition contains a few specific problems compared to the recognition of other languages. First, the information of strokes is not redundant. That is, there exist similar categories of characters. The class of a pattern can be changed by adding or removing small strokes. For example, removing a horizontal stroke

from a Hangul grapheme ' ㅏ' results in a totally different grapheme ' ㅣ'. It also means that discrimination of meaningful strokes from noisy strokes is not easy. Second, stroke features have various writing variations and much distortion. This is a common problem in handwritten character recognition, but this problem gets severer in handwritten Hangul character recognition. It is because a Hangul character consists of several graphemes which are combined in a two dimensional space and the shape of a grapheme can be distorted by the nearby graphemes. Third, Hangul character recognition involves a segmentation problem. In other words, many interpretations of a character are possible according to the different grapheme segmentation. As shown in figure 1, a little difference in grapheme segmentation results in a totally different character class. The left-most image shows an input image and its true label, and 5 images on the right side show its possible grapheme segmentation and the resulting class labels. Due to the difficulties mentioned above, the handwritten Hangul recognition problem has been considered a hard problem.

Recently, H. Y. Kim and J. H. Kim proposed a Hangul recognition system based on random graphs [5]. They used the hierarchical characteristics of Hangul and adopted a bottom-up approach to find the best grapheme segmentation result. In this system, grapheme hypotheses are generated by combining primitive strokes and then character hypotheses are generated by combining the grapheme hypotheses in a similar manner. However, this bottom-up approach suffers from a problem of time complexity which is due to the numerous grapheme hypotheses generated.

In this paper, we propose a Hangul recognition system based on a random hypergraph which enhances the relationship modeling of a random graph. Instead of using the first-order relationships in a random graph, higher-order relationships are considered in a random hypergraph. As a consequence, invalid grapheme hypotheses can be eliminated in advance and moreover the recognition performance can improve.

The rest of this paper is as follows. In section 2, random graph based modeling is reviewed and a Hangul recognition system based on the random graph is analyzed in section 3. In section 4, a new definition of a random hypergraph is given and the proposed Hangul recognition system based on the definition is proposed. Preliminary experimental results and their analyses conclude the paper.

2 Random Graph Based Modeling

A random graph is a graph whose vertices and edges are random variables. It is instantiated by mapping its vertices and edges to some states. That is, its instantiation is a labeled or attributed graph. A formal definition of random graphs was introduced firstly by Wong and Ghahraman [1]. They formulated structural-contextual dichotomy of random graphs using the definition. The structure of an object is represented by the structure of a graph and its context is stochastically expressed by the randomness of random variables. In most cases, the random variables of a random graph has finite states. In an usual application to pattern recognition, the vertices are mapped to features and the edges are mapped to the relationships between those features.

Chen and Lieh presented a new definition of an attributed graph and a random graph [2]. Using the notion, they proposed a random graph based recognition of handwritten Chinese characters. Compared to Wong and Ghahraman's definition, this definition makes the applicability of random graphs clearer. Therefore, we adopt Chen and Lieh's definition.

Definition 1 (Attributed graph) *An attributed graph over $V_N \cup V_E$ is a 4-tuple $G = (N, E, \mu, \delta)$ where*

1. *N is a finite, nonempty set of vertices;*
2. *$E \subset N \times N$ is a set of ordered pairs of distinct elements in N, called edges;*
3. *V_N is a finite, nonempty set of vertex labels (primitive descriptions);*
4. *V_E is a set of edge labels (relation descriptions);*
5. *$\mu : N \to V_N$ is a function, called a vertex interpreter;*
6. *$\delta : E \to V_E$ is a function, called an edge interpreter.*

Definition 2 (Random graph) *A random graph over $R_N \cup R_E$ is a 4-tuple $R = (N, E, \mu, \delta)$ such that*

1. *R_N, representing a random vertex set, is an n-tuple $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ where each α_i, called a random vertex, is a random variable;*
2. *R_E, called a random edge family, is an m-tuple $(\beta_1, \beta_2, \ldots, \beta_m)$ where each β_j, called a random edge, is also a random variable;*
3. *N is a finite and nonempty set of vertices;*
4. *$E \subset N \times N$ is a set of ordered pairs of distinct elements in N;*
5. *$\mu : N \to R_N$ and $\delta : E \to R_E$ are functions.*

An instantiation of a random graph is an attributed graph. Each α_i is mapped to a primitive feature and each β_j is mapped to a relationship. Namely, there exists an isomorphism from an attributed graph to a random graph. However, this definition lacks in matching graphs which are not

of the same order. When the numbers of edges and vertices of an attributed graph and a random graph are not same, there exists no matching between them. In a real pattern recognition problem, some features can be missing or noise can be added. To alleviate this problem, Chen and Lieh introduced the notion of a null graph and k-extension of a graph and defined a matching between an attributed graph and a random graph as a matching between extensions of the graphs [2]. The matching score or matching probability with which a random graph is instantiated to an attributed graph is defined relevant to each problem domain [1, 2, 3, 4, 5]. In general, all random variables of a random graph are assumed to be independent (independence assumption) and the matching score has the form of the following functions (equation 1 for a matching probability and equation 2 for a heuristic matching score).

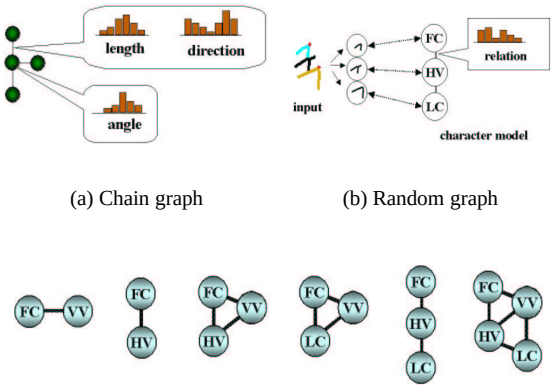
$$P_R(G,M) = \prod_{\alpha_i \in R_N} P_N(a = \mu^{-1}(\alpha_i)) \times \prod_{\beta_i \in R_E} P_E(b = \delta^{-1}(\beta_i)|\mu) \quad (1)$$

$$S_R(G,M) = \sum_{\alpha_i \in R_N} S_N(a = \mu^{-1}(\alpha_i)) + \sum_{\beta_i \in R_E} S_E(b = \delta^{-1}(\beta_i)|\mu) \quad (2)$$

where, $a \in N$, $b \in E$, μ^{-1} and δ^{-1} are the inverses of μ and δ, respectively, and $M = (\mu, \delta)$ is a monomorphism. These equations are divided into vertex matching terms ($P_N(\cdot)$ or $S_N(\cdot)$) and edge matching terms ($P_E(\cdot)$ or $S_E(\cdot)$).

3 Analysis of random graph based Hangul recognition system

As of a Chinese character, a Hangul character is constructed by combining components. That is, a Hangul character is hierarchically represented by its sub-components, graphemes. Furthermore, a grapheme can be divided into simpler sub-components, primitive strokes. Utilizing this hierarchy, a Hangul character can be efficiently represented by graphs. H. Y. Kim and J. H. Kim proposed a hierarchical random graph representation of Hangul characters [5]. They used chain graphs for modeling primitive strokes which adds a multiple-to-one feature matching capability to a random graph and used random graphs for modeling components of higher levels, graphemes and characters (figure 2). Particularly, they formulated the recognition as the problem that finds a model which maximizes a posteriori probability given an input data on a probabilistic framework. Most former random graph based recognition systems used heuristic score functions for calculating matching scores [2, 3, 4]. Using this hierarchical representation, character hypotheses



(a) Chain graph (b) Random graph

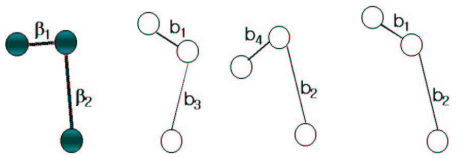
(c) Random graph models for characters

Figure 2. Hierarchical graph representation of a Hangul character (FC: first consonant, HV: horizontal vowel, VV: vertical vowel and LC: last consonant) [5]

are generated by combining primitive stroke hypotheses and grapheme hypotheses successively in a bottom-up manner. Among the character hypotheses, the hypothesis with the best a posteriori probability is selected as a recognition result.

As explained above, the advantage of a random graph is that it can model a shape of an object on a probabilistic framework. As illustrated in figure 2, each random variable is represented with a probability mass function (*pmf*). This characteristic makes it possible that most objects can be matched to the random graph, while some objects are matched with low probabilities and some objects are matched with high probabilities. However, this advantage also works for a disadvantage. Because most objects can be matched to a random graph, numerous hypotheses are generated and the time complexity for the matching increases. This problem gets more serious in component-based recognition systems like a grapheme-based Hangul character recognition system, for as the number of grapheme hypotheses increases, the number of character hypotheses increases by exponentional progression.

Another disadvantage of a random graph modeling comes from the independence assumption between random variables. Due to the independence assumption, an attributed graph parts of which are matched to random variables of a random graph with high probabilities is matched to the random graph with a high probability. Figure 3 shows an example of this case. When the representative patterns of a grapheme ' ㄱ ' are like figure 3(b), a pattern of figure 3(c) is matched to a chain graph (3(a)) with a high probability.



(a) (b) (c)

Figure 3. Independence assumption in a random graph can generate a rare instance with a high probability. (a) A chain graph model for a grapheme '¬' (b) Representative patterns for '¬' (c) A rare instance of '¬'

It is because the *pmf*s for the random variables, β_1 and β_2 have high peaks at the stroke directions of the pattern, b_1 and b_2, respectively. Certainly, this problem can be resolved by adding a new random variable which models a relationship between β_1 and β_2. However, the random graph modeling has an intrinsic deficiency in relationship modeling, or dependency modeling.

In general, every random variable of a random graph is dependent on the other random variables in a some degree. And some random variables have high dependencies on other random variables. In a chain graph and a random graph, such dependencies are represented by random edges which model pairwise relationships between two random variables. However, in a certain case, a random variable can be dependent on more than one random variables, that is, there can be high-order dependencies between random variables. In this case, there is a possibility that the problem of figure 3 will happen, for the random graph models only the first-order relationships between two variables.

4 Handwritten Hangul recognition based on random hypergraphs

As explained in the previous section, random graph based Hangul recognition has high time complexity and comparatively weak dependency modeling power. To alleviate these problems, we propose a random hypergraph based modeling which generalizes the first-order dependency modeling of a random graph to high-order dependency modeling. Considering high-order dependencies, global shape as well as local shape of objects can be modeled. By modeling local and global shape of an object simultaneously, discrimination of a class instance from instances of other classes becomes easier.

4.1 Random hypergraph

A hypergraph is a graph in which generalized edges (called *hyperedges*) may connect more than two vertices. Intuitively, a random hypergraph makes a difference with a random graph in the dimension of edges. A formal definition of a random hypergraph is given as follows.

Definition 3 (Random hypergraph) *A random hypergraph over $R_N \cup R_E$ is a 4-tuple $R = (N, E, \mu, \delta)$ such that*

1. *R_N, representing a random vertex set, is an n-tuple $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ where each α_i, called a random vertex, is a random variable;*
2. *R_E, called a random hyperedge family, is an m-tuple $(\beta_1, \beta_2, \ldots, \beta_m)$ where each β_j, called a random hyperedge, is also a random variable;*
3. *N is a finite and nonempty set of vertices;*
4. *E is a subset of a set of subsets of N, that is, a set of ordered sequences of distinct elements in N;*
5. *$\mu : N \to R_N$ and $\delta : E \to R_E$ are functions.*

Let a random variable X be dependent on k random variables, $\{Y_i\}_{i=1}^{k}$ $(0 < k < n)$. Then this dependency can be represented with a probability, $P(X|Y_1, Y_2, \ldots, Y_k)$. Considering that vertices of a random hypergraph are themselves random variables, the edge matching probability for a hyperedge, $e = (x, y_1, y_2, \ldots, y_k)$, can be represented with the following function.

$$P_E(X, Y_1, Y_2, \ldots, Y_k) = P(X|Y_1, Y_2, \ldots, Y_k) \quad (3)$$

In the case of a random graph, the value of k is fixed to 1.

4.2 Dependency modeling

When we consider the high-order dependencies in modeling, an important issue is how to represent the high-order probabilities. Roughly, there are two kinds of representation for a probability, a discrete probability and a continuous probability. In the case of a discrete probability, it is represented with a probability mass function (*pmf*), that is, a table. Therefore, as the order of dependencies increases, the number of training data for estimating sufficient statistics increases exponentially. Practically, it is very difficult to collect such large training data. Some researchers solved this problem by approximating the high-order probability with a product of several low-order probabilities. Chow and Liu approximated an n-th order discrete probability with a product of n-1 first-order component distributions using a dependence tree technique [6]. They also proved

that the proposed method guarantees to find an optimum set of n-1 first-order dependence relationships among the n variables. However, this method is based on the assumption that there are only pairwise dependencies between variables, and there is a possibility that the problem revealed in a figure 3 will happen.

The other method is to represent the high-order probability with a continuous probability distribution function (*pdf*). Usually, the dependency between a variable and its parental variables can be represented by a function of the parental variables. A linear regression model is one of these kinds of dependency modeling methods [7, 8]. It has advantages that the dependencies among variables are expressed in an explicit and simple way and that model parameters are estimated by a well-known maximum likelihood estimation (MLE) technique.

For these advantages, we use the linear regression based dependency modeling to represent the high-order probabilities. The linear regression based dependency model can be summarized as follows. Let X be a random variable and it be dependent on k random variables, $\{Y_i\}_{i=1}^k$. Then the conditional distribution of a random variable X given its parents $\{Y_i\}_{i=1}^k$ can be specified by a Gaussian distribution.

$$X = \mu_X + \sum_{i=1}^{k} b_i(Y_i - \mu_{Y_i}) + \sigma_X W \quad (4)$$

$$f(x|y_1,\ldots,y_k) = (2\pi\sigma_X^2)^{-\frac{1}{2}} exp\left[-\frac{1}{2\sigma_X^2}(x-u)^2\right] \quad (5)$$

where $W \sim N(0,1)$ is a white noise random variable, $u = \mu_X + \sum_{i=1}^{k} b_i(y_i - \mu_{Y_i})$, and the b_i's are the regression coefficients. In most cases, we need to calculate a probability that $(x, y_1, \ldots, y_k)$ will occur. However, the function values of a continuous probability distribution function have no probabilistic meaning. Therefore, we need to convert continuous *pdf* values to discrete *pmf* values using a definite integral or other ways. We used the following equation to deduce a probabilistic meaning from the Gaussian *pdf*.

$$P(x|y_1,\ldots,y_k) = \frac{1 - \int_{-|x|}^{|x|} f(x|y_1,\ldots,y_k)}{\sum_{\forall x'} 1 - \int_{-|x'|}^{|x'|} f(x'|y_1,\ldots,y_k)} \quad (6)$$

4.3 Hangul recognition system

The proposed Hangul recognition system is almost the same as that of [5]. The main difference is that random hypergraphs are used for modeling in the proposed system. The overall process consists of preprocessing, primitive stroke extraction, grapheme hypotheses generation, and character hypotheses generation. An attributed graph is constructed from an input character image through skeletonization, stroke extraction, and linear approximation of strokes. In an attributed graph, vertices are mapped to feature points like end points, cross points and bending points of high curvature, and edges are mapped to line segments between those feature points. After the construction of an attributed graph, primitive strokes are extracted from the attributed graph by a subgraph isomorphism and chain graph matching. And then grapheme hypotheses are generated by combining the primitive strokes. Finally, non-overlapped grapheme hypotheses are combined to form character hypotheses. Grapheme and character hypotheses are scored and verified by matching them with grapheme and character models, respectively. The character hypothesis with the maximum a posteriori probability is selected as the recognition result among those character hypotheses.

Let M_i be a character model and X be an input attributed graph. Then the recognition can be formulated as the problem of finding a model $\hat{M}$ which maximizes the a posteriori probability given the attributed graph, which is written as

$$\hat{M} = \arg max_{M_i} P(M_i|X) \quad (7)$$

Using a Bayes' rule, this equation can be rewritten as

$$\hat{M} = \arg max_{M_i} P(X|M_i)P(M_i) \quad (8)$$

where $P(X|M_i)$ can be calculated by the equation 1 and the mapping from an attributed graph to a character model is determined hierarchically by the combination of primitive strokes and grapheme hypotheses which form a character hypothesis.

In a chain graph which models primitive strokes, angles between line segments are observed at each random vertex and the direction and the length of line segments are observed at each random edge. In our system, the length of line segments is quantized to 12 levels which is normalized by a character height, and angles between line segments and directions of line segments are quantized to 16 levels where each level covers 22.5 degrees. In a random graph which models graphemes and characters, primitive strokes and grapheme instances are matched to random vertices and their positional relationships are defined by random hyperedges. In our system, the hyperedges of a grapheme model is manually designed to model the positional relationships between feature points of primitive strokes. In other words, the (x, y)-locations of the feature points are used for input of the linear regression based dependency modeling (see equation 5).

5 Experiments

To evaluate the performance of the proposed random hypergraph based recognition system, preliminary experiments were conducted with 520 Hangul characters, a set of 520 classes of the KU-1 database [9]. The performance

System	*Speed(s)*	*Recognition(%)*
random graph based system	277	88.2
the proposed system	203	93.3

Table 1. The performance comparison between a random graph based system and the proposed system

is compared to that of a random graph based system in [5]. The only difference between two systems is in the use of hyperedges in grapheme models. Grapheme candidates with low matching probabilities were pruned out in both systems. Table 1 shows the comparison result. The speed field of the table is total time in seconds which it takes to recognize a whole set of characters. According to this experiment, the proposed system outperforms a random graph based system in both speed and a recognition rate. Numerically, 26.7% of speed improvement and 43.2% of error reduction in a recognition rate were obtained.

In spite of this promising result, the proposed random hypergraph based modeling has a few problems. For good dependency modeling, the variance σ_X of the equation 5 should be sufficiently small, for the smaller the variance is, with less uncertainty the position of X can be estimated from its parents, $\{Y_i\}_{i=1}^{k}$. However, the analysis of our experiments revealed that some grapheme models had large variances. And as the result, some grapheme instances of a true label were pruned out for low matchinig probabilities and some grapheme instances of a wrong label were matched with high probabilities. This problem seems to be indispensable because of the cursive nature of handwritings.

6 Conclusion

In this paper, a new notion of a random hypergraph was introduced to overcome the deficiencies of a random graph. Usually, a random graph based recognition system has high time complexity for the recognition and it has a comparatively weak power of modeling dependencies among random variables due to the independence assumption between variables. In a random hypergraph, the relationship modeling of a random graph is enhanced to represent a high-order dependency which is realized by a random hyperedge. The high-order dependency is represented with a high-order conditional probability using a linear regression based dependency modeling method [7, 8].

Based on the random hypergraph, a handwritten Hangul recognition system was developed. In a preliminary recognition experiment conducted with a set of 520 Hangul characters, the proposed system outperformed a random graph based recognition system in both speed and a classification rate. In spite of this success, the analysis of the proposed system showed a problem in the dependency modeling. The regression based dependency modeling showed a limitation in dealing with the writing variation of handwritings. Future works will aim at modeling the high-order dependencies to absorb the variation in handwritings.

References

[1] A. K. C. Wong and D. E. Ghahraman, Random Graphs: Structural-Contextual Dichotomy, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 2, No. 4, 1980, pp. 341-348

[2] L. -H. Chen and J. -R. Lieh, Handwritten Character Recognition Using a 2-Layer Random Graph Model By Relaxation Matching, Pattern Recognition, Vol. 23, No. 11, 1990, pp. 1189-1205

[3] S. W. Lu, Y. Ren, and C. Y. Suen, Hierarchical Attributed Graph Representation and Recognition of Handwritten Chinese Characters, Pattern Recognition, Vol. 24, No. 7, 1991, pp. 617-632

[4] J. Liu, W. K. Cham, and M. M. Chang, Online Chinese Character Recognition Using Attributed Relational Graph Matching, IEE. Proc. Vis. Image Signal Process, Vol. 143, No. 2, 1996, pp. 125-131

[5] H. -Y. Kim and J. H. Kim, Hierarchical Random Graph Representation of Handwritten Characters and its Application to Hangul Recognition, Pattern Recognition, Vol. 34, No. 2, 2001, pp. 187-201

[6] C. K. Chow and C. N. Liu, Approximating Discrete Probability Distributions with Dependence Trees, IEEE Trans. Information Theory, Vol. 14, No. 3, 1968, pp. 462-467

[7] Kevin Murphy, Inference and learning in hybrid Bayesian networks, U.C. Berkeley Technical Report CSD-98-990, 1998

[8] S. -J. Cho and J. H. Kim, Bayesian Network Modeling of Strokes and their Relationships for On-line Handwriting Recognition, 6th Int'l Conf. on Document Analysis and Recognition, Seattle, WA, Sep. 10-13, 2001, pp. 86-90

[9] D. -I. Kim and S. -W. Lee, An automatic evaluation of handwriting qualities for off-line handwritten Hangul character database KU-1, Proceedings of the 25th Korea Information Science Society Conference, Vol. 25 (I), 1998, pp. 707-709