Generation of Handwritten Characters with Bayesian network based On-line Handwriting Recognizers

Hyunil Choi, Sung-Jung Cho and Jin H. Kim
CS Div., EECS Dept., KAIST,
373-1 Kusong-dong, Yousong-ku, Daejon,
305-701, KOREA
E-mail: {hichoi, sjcho, jkim}@ai.kaist.ac.kr

Abstract

In this paper, we propose a new character generation method from on-line handwriting recognizers based on Bayesian networks. On-line handwriting recognizers are trained with handwriting samples from many writers. Then, character shapes are generated from given texts by searching the most probable input point sequences. Since Bayesian network based classifiers have large number of parameters for modeling components and their relationships, they generate more natural character shapes than various kinds of hidden Markov models.

1 Introduction

The recent emergence of pen computers makes handwriting more convenient and natural input method in human-computer interaction (HCI). Computer users are getting familiar to input messages with handwriting. If an user has a font which shows his/her own writing style, messages can be inputted faster than handwriting, looked friendly. From this requirement, we made a system which generates handwriting when a person types text strings. It is analogous to the text-to-speech system in the speech recognition literature. However, it is different in that handwritten characters, instead of speech, are generated, and they are trainable by writers.

Possible applications of character generations are as follows. We could make handwriting fonts automatically. With some handwriting control parameters like cursiveness to the baseline system, various fonts can be made. If the system is trained with specific writer's handwriting examples, the system will enable the writer's messages look more friendly by showing the writer's handwriting. Suppose that an user has a computer font which shows his own writing style. He can use the font in writing E-mail or greetings. We can say the counterparts might feel friendly towards him. Another good application could be use of personalized font in chatting service. Someone who wants to chat using his own font gives some handwritings to the service. The service creates a font for him and every message that he types will show his writing style.

Handwriting generation task had been studied for significant time. The motor model [7] views handwriting process as a result of motor processes that control writing method. Although it established unified framework for various kinds of handwriting generation models, strong assumptions on both linearity of subsystems and simplification of complex muscle movement are impractical. Based on the motor models, *delta-lognormal theory* [6] enables parametric representation of handwriting generation. Motor model based approaches can be used to simulate handwriting in terms of motor aspect. But it is commonly known that the dynamic-inverse problem is hard to solve [9].

For the pratical use of handwriting generation, learning-based approach is another promising methodology. By collecting handwriting examples from a writer, a system learns the writer's writing style. In [3], common glyphs are predefined and the system keeps handwritten glyphs of the writer. When texts are given, the system concates corresponding glyphs to make handwriting. In spite of it's simplicity, it is hard to extend the system if a language has many classes. Wang [10] extracts strokes of letter and ligature using tri-unit handwriting model. By minimizing deformable energy, synthesized handwriting is generated.

In this paper, we propose a system which generates characters by utilizing on-line handwriting recognizer. The system is trained with handwriting examples from a lot of writers. For a given character code, we pick up the corresponding model in the model set and let the model generate most probable points on 2-D plane. The advantage of our approach is that character shapes can have systematic variations because components and their relationships of charac-

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03)
0-7695-1960-1/03 $17.00 © 2003 IEEE

ters are modeled in detail. Another feasible characteristic of out system is that it can generate characters with large number of classes. In case of Hangul, at least 2,350 characters are required. Instead of training all the classes, a character can be generated by composing several grapheme models.

The rest of the paper is organized as follows. Section 2 introduces the Bayesian network based on-line handwriting recognizers. Section 3 describes character generation algorithm from the recognizers. Section 4 presents the character shapes generated from the proposed model and compare them with shapes from HMMs. Section 5 concludes this paper.

2 Bayesian network based on-line handwriting recognizer[2]

In this section, we describe on-line handwriting recognizer for generating character. Although we have a generative system designed for both Hangul and digit, we only show modeling of Hangul characters. Digits models correspond to grapheme models in this paper.

Hangul characters have four level components: syllable characters, graphemes, strokes and points. A syllable character is structurally composed of graphemes: one first consonant (19 classes, denoted as *C*), one vowel (21 classes, *J*) and one optional last consonant (27 classes, *Z*). Therefore, 11,772 syllable characters can be constructed theoretically. Among them, only 2,350 ones are practically used. A grapheme, which corresponds to an alphabet in English words and a digit in digit strings, is composed of several strokes. A stroke is a straight or nearly straight trace that has distinct directions from connected traces in writing order, and is composed of points. A point, the primitive component, is represented by its (x, y) coordinate.

We proposed a Bayesian network framework for explicitly modeling components and their relationships of Hangul characters [2]. A Hangul character is modeled with hierarchical components: a syllable model, grapheme models, stroke models and point models. Each model is constructed with subcomponents and their relationships. A point model, the primitive one, is represented by a 2-D Gaussian for point positions on X-Y plane. Relationships are modeled with position dependencies between components.

2.1 Introduction to Bayesian networks

A Bayesian network [4] is a probabilistic graph for representing random variables and their dependencies. It efficiently encodes the joint probability distribution of a large set of variables. Its nodes represent random variables and its arcs represent dependencies between random variables with conditional probabilities at nodes.

In order to model the conditional probability between continuous random variables with high order dependencies, we adopt conditional Gaussian distributions [5]. The mean of a random variable is assumed to be determined from the linear weight sum of dependent variable values. The difference between the mean and the random variable value is assumed to be Gaussian. When a multivariate random variable X depends on $X_1, \ldots, X_n$, the conditional probability distribution is given as follows:

$$\begin{aligned} & P(X = \mathbf{x} | X_1 = \mathbf{x_1}, \ldots, X_n = \mathbf{x_n}) \qquad (1) \\ = & \quad (2\pi)^{-\frac{d}{2}} |\mathbf{\Sigma}|^{-\frac{1}{2}} \exp[-\frac{1}{2}(\mathbf{x} - \mu)^T \mathbf{\Sigma}^{-1} (\mathbf{x} - \mu)] \end{aligned}$$

The mean μ is determined from the dependant variable values $\mathbf{Z} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_n^T, 1]$ as follows:

$$\mu = \mathbf{W}\mathbf{Z}^T \qquad (2)$$

where **W** is a $d \times k$ linear regression matrix, d is the dimension of X, and k is the dimension of $\mathbf{Z}^T$.

2.2 Hangul character model

A point instance has the attribute of (x,y) position on the 2-D plane. So, a point model has 2-D Gaussian distribution for modeling 2-D point positions. It is represented by one node in Bayesian networks.

When a point $P = (x, y)$ depends on other points $P_1 = (x_1, y_1), \ldots, P_n = (x_n, y_n)$, its matching probability $P(x, y | x_1, \ldots, y_n)$ is given from the conditional Gaussian distribution (Eq. (1), (2)) by setting $X = (x, y)$ and $\mathbf{Z} = [x_1, y_1, \ldots, x_n, y_n, 1]$.

A stroke instance is composed of points. Therefore, a stroke model is composed of point models with their relationships, called within-stroke relationships (WSRs). A stroke model is constructed by recursively adding mid point models and specifying their dependencies from stroke end points. At the mid point, the lengths of the left and the right partial strokes are equal. The recursion stops when the covariances of newly added point models become smaller than some threshold.

The matching probability of a stroke S with a point sequence O_1^t is given as follows:

$$\begin{aligned} & P(S = O_1^t) \qquad (3) \\ = & \quad P(EP_0 = O_1) P(EP_1 = O_t) \prod_{i=1}^{2^d - 1} P(IP_i = ip_i | pa(IP_i)) \end{aligned}$$

A grapheme instance consists of strokes which have relationships. Therefore, a grapheme model consists of stroke models with inter-stroke relationships (ISRs). ISRs are abstractly represented with dependencies between stroke end points.

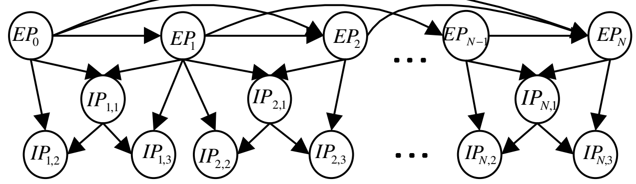


Figure 1. Bayesian network representation of a grapheme model.

Figure 1 shows a Bayesian network based grapheme model with N strokes and the stroke recursion depth $d = 2$. EP_i's are the stroke end point models and $IP_{i,j}$'s are the point models within the i-th stroke. The right end point of the previous stroke is shared with the left one of the following stroke. ISRs are represented by the arcs between EP_i's, and WSRs are represented by the incoming arcs to $IP_{i,j}$'s.

When a grapheme model G with N strokes and a grapheme instance O_1^T whose one stroke segmentation instance is $\gamma = (t_0, t_1, \ldots, t_N)$, $t_0=1<t_1<\ldots<t_N=T$, and whose whole segmentation set is Γ, the matching probability is given as follows:

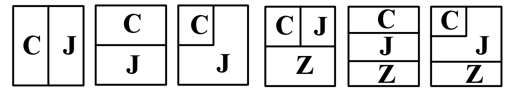
$$P(O_1^T|G)=\sum_{\gamma\in\Gamma}\prod_{i=1}^{N}P(S_i = O_{t_{i-1}}^{t_i}|O_{t_0},\ldots,O_{t_{i-1}}) \quad (4)$$

Ligatures are the traces connecting graphemes, which are usually pen-up movements. Their shapes are simple and nearly straight. Therefore, a ligature model is made of one stroke model.

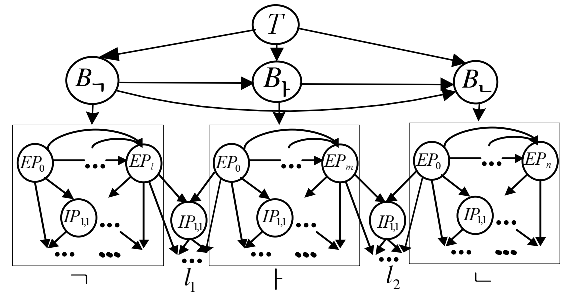
Hangul character models are constructed by combining their grapheme and ligature models and specifying inter-grapheme relationships. IGRs are encapsulated by relationships between grapheme bounding boxes (BC, BJ, BZ). Six different IGR types (Fig. 2 (a)) are denoted as T. Fig. 2 (b) shows the Bayesian network model corresponding to graphemes 'ㄱ', l_1, 'ㅏ', l_2, 'ㄴ' (Hangul character'간'). Grapheme bounding boxes have dependencies from T(=4). Their dependencies are modeled as the conditional Gaussian distribution (Eq. (1), (2)). Because graphemes have scaling and translation variations in Hangul syllable bounding boxes, the dependencies from grapheme bounding boxes to grapheme models are modeled by translation- and scaling-invariant conditional Gaussian distributions; compared to Eq. (2), the mean is changed as follows:

$$\mu = \mathbf{S}\mathbf{W}(\mathbf{Z}')^T + \mathbf{T} \quad (5)$$

where $\mathbf{S}$ and $\mathbf{T}$ are scaling and translation matrix for the input space, and $\mathbf{Z}'$ is the set of normalized coordinates of grapheme points.



(a) Six types of IGRs



(b) Hangul character model for 간 (ㄱ,l_1, ㅏ,l_2,ㄴ).

Figure 2. Hangul character model.

When a Hangul character model $H = C \cdot L_1 \cdot J$ whose type is T matches a point sequence O_1^M, the matching probability is calculated by finding the most probable grapheme segmentation. Let $t_0 = 1 \le t_1 \le t_2 \le t_3 = M$ denote a segment instance of O_1^M such that C is matched to $O_1^{t_1}$, L_1 to $O_{t_1}^{t_2}$, J to $O_{t_2}^{t_3}$. Then the matching probability is as follows:

$$\begin{aligned} &P(O_1^M|C \cdot L_1 \cdot J) \quad (6)\\ &= P_C(B(O_1^{t_1})|T)P_{J|C}(B(O_{t_2}^{t_3})|B(O_1^{t_1}),T)\\ &\quad P_C(O_1^{t_1}|B(O_1^{t_1}))P_{L_1}(O_{t_1}^{t_2})P_J(O_{t_2}^{t_3}|B(O_{t_2}^{t_3})) \end{aligned}$$

In the case of a Hangul character with the ligature L_2 and the final consonant Z, O_1^M is divided into five segments and $P_{L_1}(O_{t_3}^{t_4})$, $P_Z(O_{t_4}^M|B(O_{t_4}^M))$ and $P_{Z|C,J}(B(O_{t_4}^M)|B(O_1^{t_1}), B(O_{t_2}^{t_3}))$ are multiplied to Eq. (6).

3 Algorithm for character generation [1]

Since the Bayesian network is generative model, we can choose the most probable characters from the Bayesian network based character models. The most probable character can be interpreted as the most representative character pattern that each model has. If a model successfully learns the concept of characters from training data, then it can generate natural shapes and vice versa.

A character is generated from the proposed character model by generating points according to their dependency order. The points without dependencies are generated at first. Then, points whose all dependent points are sequentially generated.

Step 1. (Hangul) From the given Hangul syllable label, determine its syllable type, grapheme and ligature labels.

Step 2. **(Hangul)** Construct a Hangul syllable model by concatenating grapheme models, ligature models, the type variable and the bounding box variables.

Step 3. **(Hangul)** Generate the most probable grapheme bounding boxes B_j's given the type by finding the mean of conditional Gaussian distributions (Eq. (2), (5)). The one whose dependent variables are already generated is generated first.

Step 4. Sort all the point models in a grapheme (also an alphabet or a digit) model according to the dependency topology order. Let's denote them as P_i's ($i = 1, \ldots, N$).

Step 5. For $i = 1, \ldots, N$, generate the most probable point instance $O(i)$ from the point model P_i given the grapheme bounding box $B_j = (l, t, r, b)$ as follows:

$$O(i) = (x, y) = \begin{bmatrix} r-l & 0 \\ 0 & b-t \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} + \begin{bmatrix} l \\ t \end{bmatrix}$$

where

$$(x', y') = \mathbf{W}\,[x'_1, y'_1, \ldots, x'_n, y'_n, 1]^T$$

(from Eq. (2), (5)). x_i's and y_i's are the normalized coordinates of points with the bounding box.

Step 6. Sort $O(i)$'s according to their writing order.

The character generation algorithms for various kinds of HMMs appeared in [8].

4 Experimental results [1]

We conducted experiments which show the ability of handwriting generation of proposed method. We applied our system to Hangul syllable and digit generation.

4.1 Training data

The Hangul data for the experiment were collected from high school and college students. There was no restriction or guidance in the writing styles. As a result, cursive writing style as well as run-on style were found in the data. The system is trained with 49,049 characters written by 48 writers. Figure 3(a) shows examples of traning data.

For the numeral character generation, we used 4,046 samples from KAIST DB. It has well written characters by high school students without any writing restriction in Korea. Figure 3(b) shows some sample characters from the DBs. Since each writer has different writing style and degree of variation, we can expect that Bayesian network learns average writing style.

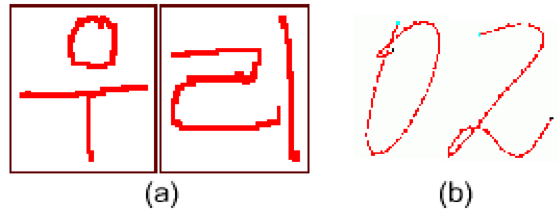


Figure 3. Samples from training data (a) Hangul syllable characters (b) digits

4.2 Comparison of generated character shapes

Figure 4 and 5 show that the proposed character models seem to fit well our concepts of characters. Hangul syllables and digits generated by them have natural shapes. Grapheme relationships are natural and not overlapped one another. Strokes have not only straight but also cursive shapes. Their relationships are also natural. On the other hand, digits generated by several kinds of HMMs are not natural as shown in (b), (c) and (d) of Figure 4, 5 (reprinted from [8]). In (b), only one stroke of each grapheme or digit is enlarged unrealistically because stroke lengths are not realistically modeled in normal HMMs. In (c) and (d), strokes have more realistic lengths but the overall character shapes are not natural because stroke relationships are not explicitly modeled. Also, grapheme relationships are not well modeled. This result shows the superior modeling capacity of the proposed model.

5 Conclusions

In this paper, we generate character shapes from online handwriting recognizers based on Bayesian networks. Characters are generated by searching the most probable point sequences.

Experimental results with Hangul and digits show that the proposed system generated more natural character shapes than HMMs. The overall shapes are more natural and the relationships between graphemes, strokes and points are also more natural.

Our future work is to make the system adaptable to specific writer. The current system generates the average character shapes from many writers. The other is to model the cursiveness and inter-character relationships for Hangul word. Also, the system needs to be extended to English word.

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03)
0-7695-1960-1/03 $17.00 © 2003 IEEE

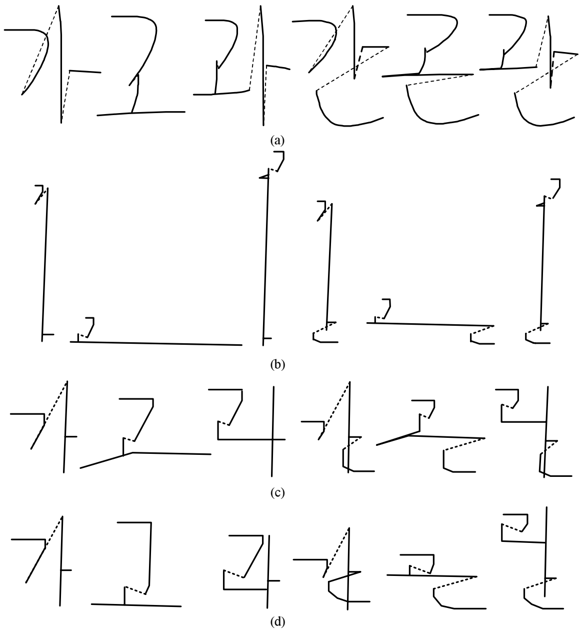


Figure 4. Hangul syllables generated from (a) proposed Bayesian network based models. (b) standard HMMs. (c) duration modeling HMMs. (d) non-stationary HMMs. Figures of (b),(c),(d) are reprinted from [8] with permission of the authors and the publisher.

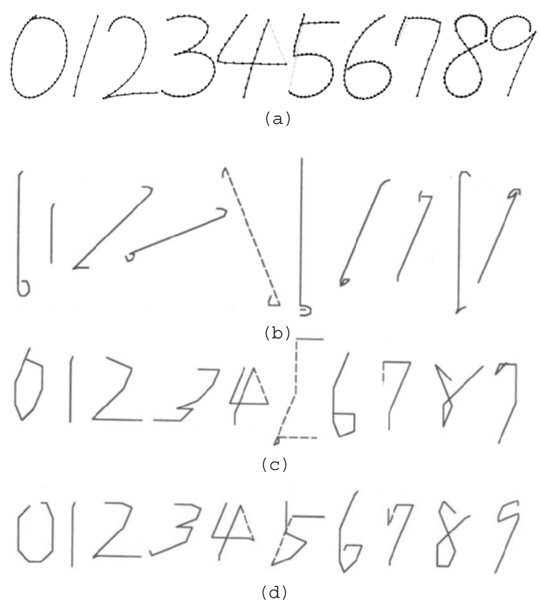


Figure 5. Digits generated from (a) proposed Bayesian network based models. (b) standard HMMs. (c) duration modeling HMMs. (d) non-stationary HMMs. Figures of (b),(c),(d) are reprinted from [8] with permission of the authors and the publisher.

Acknowledgement

This work has been partly supported by the Korean Ministry of Science and Technology for National Research Laboratory Program M1-1014-00-0165.

References

[1] S. Cho. Baysian network modeling of character components and their relationships for on-line handwriting recognition. *Ph. D thesis, EECS Dept., KAIST*, 2003.

[2] S. Cho and J. Kim. Bayesian network modeling of hangul characters for on-line handwriting recognition. *submitted to ICDAR*, 2003.

[3] I. Guyon. Handwriting synthesis from handwritten glyphs. *Porc. the Fifth International Workshop on Frontiers of Handwriting Recognition, Colchester, England*, 1996.

[4] F. Jensen. *An Introduction to Bayesian Networks*. Springer Verlag, New York, 1996.

[5] K. Murphy. Inference and learning in hybrid bayesian networks. *Technical Report 990, U.C.Berkeley, Dept. Comp. Sci*, 1998.

[6] R. Plamondon and W. Guerfali. The generation of handwriting with delta-lognormal synergies. *Bilogical Cybernetics*, 78:119–132, 1998.

[7] R. Plamondon and F. J. Maarse. An evaluation of motor models of handwriting. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1060–1072, 1989.

[8] B.-K. Sin and J. Kim. Network-based approach to korean handwriting analysis. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(2):233–249, Jun. 1998.

[9] Y. Singer and N. Tishby. Dynamical encoding of cursive handwriting. *Biological Cybernetics*, 71(3):227–237, 1994.

[10] J. Wang, C. Wu, Y.-Q. Xu, H.-Y. Shum, and L. Ji. Learning-based cursive handwriting synthesis. *IWFHR*, pages 157–162, 2002.