Probabilistic Synthesis of Personal-Style Handwriting

Hyunil CHOI†∗, Member and Jin Hyung KIM†, Nonmember

SUMMARY   The goal of personal-style handwriting synthesis is to produce texts in the same style as an individual writer by analyzing the writer’s samples of handwriting. The difficulty of handwriting synthesis is that the output should have the characteristics of the person’s handwriting as well as looking natural, based on a limited number of available examples. We develop a synthesis algorithm which produces handwriting that exhibits naturalness based on the probabilistic character model.

key words: handwriting synthesis, hierarchical character representation, probabilistic context modeling

1. Introduction

Handwriting synthesis or generation is to automatically produce synthesized text by a computer model. It produces an individual person’s handwriting with his or her writing style by analyzing a limited amount of the individual’s handwriting examples. Its output can be used to make a personalized font which would enable friendly messages in digital communication. Handwriting synthesis can also produce common shape of handwritten characters of a particular language studied for the typical writing style. This paper aims to synthesize individual’s handwriting which is visually plausible so that it does not appear forged.

A handwriting synthesis system usually focuses on the production of a glyph such as English word, Chinese character, Korean character, etc. A glyph is usually considered to be a natural unit of handwriting. It consists of one or more of components, called graphemes. A grapheme is an atomic unit of glyph and has a unique shape. Examples are a letter in English, an ideogram or a radical in Chinese character, and a jamo in Korean character (Fig. 1). Synthesizing glyphs is a part of our concerns.

Once trajectories of personal handwriting have been synthesized, a personalized glyph-font can be constructed automatically with some parameters like degree of slant or amount of inking. However, making a personalized font is a time-consuming process. Recently, a company in Korea has developed a font of a celebrity. Since only a small number of examples were available to them, they had to manually synthesize glyphs not present in the examples. They reported that it took 5 months to make the font having approximately 3,000 characters. It shows that applications such as this call for an automatic synthesis based on only a limited number of examples.

1.1 Characteristics of Individuality of Handwriting

One area of studying the characteristics of individuality of handwriting is known as questioned document examination (QDE) [1]. The objective of QDE is to examine whether a document belongs to a specific writer. QDE commonly employs the following characteristics of handwriting to model individuality: arrangement, class of allograph (variant shape of a grapheme), connections, design of allographs and their construction, dimensions (vertical and horizontal), slant or slope (skew), intra-glyph and inter-glyph spacings, abbreviations, baseline alignment, initial and terminal strokes, punctuation (presence, style, and location), embellishments (decorations), legibility or writing quality, line continuity, line quality, pen control (pressure, shading, holding), writing movement (arched, angular, interminable), natural variations or consistency, persistency, lateral expansion, and glyph proportions. Some QDE research, e.g., [2], used more limited characteristics such as the glyph spacing, the spacing between specific graphemes, the size of specific graphemes, the slope of the ascenders and descendents, etc.

1.2 Challenges

The first challenge is to describe randomness of handwriting coming from various writing conditions such as type of pen and paper, strength of muscle, psychological state of writer, etc. However, such randomness occurs within the structure of glyph, that is, the degree of variation lies in a certain range where the structure of glyph is sufficiently maintained.

The second challenge is to consider the differentiation of grapheme shape depending on the preceding or following graphemes. Each grapheme has a unique basic shape,
The shape of a glyph is described by the arrangement of graphemes in writing space. However, in the handwritten glyph, grapheme’s shape varies according to neighboring graphemes. This phenomenon is commonly known as the coarticulation effect. Handwriting, written naturally, shows such coarticulation effect as shown in Figs. 2 (a) to (c).

The third challenge is to represent the individuality of handwriting. Obviously all the features considered by QDE would be useful to express the writer-specific style such as slant, skew, size, etc. as shown in Figs. 2 (d) and 2 (e). However, only a portion of the features in QDE is computable and appropriate for representing online handwriting. For instance, the legibility, line quality, persistence, natural variation, and rhythm are difficult to quantify. In addition, pen control is generally ignored when online handwriting trajectory is modeled.

The fourth challenge is to synthesize handwriting based on a limited number of examples. Obtaining enough examples of individual writer’s glyphs for learning all possible coarticulation effects is not a trivial problem in practice. Assuming that a limited examples is given, the exact handwritten glyph to be synthesized may not be present in the given examples. One of the difficulties of handwriting synthesis is that coarticulation effects should be preserved in the synthesized unseen glyph so that it would appear natural.

1.3 Related Work

Over the past few decades, there have been a large number of studies on handwriting synthesis. One can roughly categorize them into three main approaches: grapheme based, statistical model based, and motor model based.

Grapheme based approaches [3], [4], generally define composition or synthesis rules which use the structure of a specific language. Handwriting synthesis is performed by adapting the graphemes in the training examples by applying a set of pre-defined rules. Although they can synthesize handwriting based on a limited number of examples, generating the rule set is subjective and laborious. Lin [5] proposed a system generating handwritten glyphs by sampling graphemes, adding shape variation, aligning characters with baseline, and connecting graphemes by ligature model. Generated handwriting also has pen-pressure style to be more realistic. They achieved excellent synthesis result showing natural ligatures with limited number of individual’s samples. However, they simplified coarticulation effect of each grapheme as coarticulation effect between ‘a’ and some other graphemes, such as ‘i’, ‘j’, ‘f’, etc. It is questionable that such simplification will represent other types of coarticulation effect.

Motor model based approaches [6]–[8] describe handwriting as the super-imposition of discontinuous strokes that results in a continuous summation of velocity vectors. They can generate various cursive handwriting by applying different parameters of velocity functions to the template of glyph or grapheme. These approaches can represent size, slant, and cursiveness of handwriting which closely resemble human handwriting. But little is known about learning optimal parameters which result a specific writing style. Moreover, these approaches generate a variant of the seen glyph or grapheme. Therefore, they are not applicable to handwriting synthesis based on a limited number of examples.

Statistical model based approaches [9]–[12] model handwriting shapes statistically based on the training examples. Typically they use a set of parameters of statistical model in generating varied shapes. In one notable approach, Wang [13], [14] probabilistically represented connecting patterns of two consecutive graphemes. They accomplished a fluent synthesized handwriting. However, it is difficult to represent fine details of the shape because the model complexity increases for more detailed shape. Besides, generated handwriting was unconditionally connected and only the first order coarticulation effect was modeled, which means only the starting and ending strokes influence one another in consecutive graphemes. Niels [15] developed a system which determines personal handwriting style to one of pre-clustered style groups. When individual’s handwriting is given, it is compared with templates in the groups and membership is assigned. Handwriting is generated according to the membership. They mainly focused categorizing handwriting style, i.e., cursive, mixed, or print, while neglected modeling detail shape of individual’s handwriting. Jawahar [16] modeled Indian language which has relatively complex shape compared with that of English. For each stroke, they keep size-normalized shape, stroke shape distribution, and positional relationships between strokes. Synthesis is performed by generating stroke layout and randomly sampling stroke shape. Although generated handwriting showed individuality and diversity, stroke shape differentiation according to neighboring stroke is not explicitly modeled.

1.4 Overview of the Proposed Method

Our approach for the challenges follows two threads: modeling coarticulation effect as the spatial relationships between graphemes and using the relationships for synthesizing. The spatial relationships between graphemes are represented by grapheme’s distribution of a high order, conditioned on the instances of neighboring graphemes. These are learned by using training data written by various persons.
In order to deal with estimating a high order conditional distribution, we view the spatial relationships as three different perspectives: shape, layout, and global appearance.

In shape and layout, we use a parametric form of conditional distribution and limited important relationships instead of using all relationships. The global appearance of glyph means position, size, slant, skew, and rotation of glyph in writing space. It is incorporated into the shape and layout representation. Using the spatial relationships, unseen glyph is synthesized by defining interchangeability between graphemes in different glyphs which is measured by using the proposed model.

In summary, our approach for the four challenges are 1) modeling probabilistic glyph model 2) incorporating spatial relationships between graphemes into the model 3) reusing glyphs and graphemes in examples, parameterizing global appearance 4) defining interchangeability between graphemes in different glyphs, using commonly written grapheme if no other possible.

1.5 Outline of This Paper

The rest of this paper is organized as follows. In Sect. 2, the probabilistic model of handwritten glyph is introduced. The spatial relationships in shape and layout of glyph are represented by the probabilistic distributions. We then approximate the high order conditional distribution by a low order one and incorporate the global appearance of text block into probabilistic distributions. In Sect. 3, we discuss how unseen glyphs are synthesized with the proposed probabilistic model. In Sect. 4, some examples of synthesized handwriting are analyzed. We then present a human evaluation of the quality of the synthesis of the handwriting, followed by conclusion in Sect. 5.

2. The Probabilistic Model of Handwritten Glyph

This section describes the probabilistic model of glyph for synthesizing handwriting. As an instance of such model, we will demonstrate our approach using Korean characters. Korean character is composed of one consonant jamo, one vowel jamo, and optional one more consonant jamo. These are respectively referred as the first consonant (FC), the vowel (VW), and the last consonant (LC). In contemporary Korean text, 19 jamos are used for the first consonant, 21 jamos for the vowel, and 27 for the last consonant. These jamos consist of basic Korean alphabet set and their legal compound forms. Korean character is constructed by arranging graphemes in 2D space. There are six types of structures for the grapheme arrangement as shown in Fig. 4.

2.1 Shape

A glyph consists of graphemes which in turn are composed of a set of strokes. A stroke is defined as a nearly straight line segment which can be modeled as a sequence of points. The global and local shape of a glyph is respectively characterized by the spatial arrangement of strokes and the sequence of points within strokes. In this manner, coarse-to-fine shape of a glyph can be represented. A point is represented as its \((x, y)\) coordinates in two dimensional space. Through this hierarchy of concepts, glyph-grapheme-stroke-point, the shape of the entire glyph is finally represented by a sequence of points. Figure 5 shows such a hierarchical representation of glyph shape.

The shape of glyph is governed by a joint probability distribution of graphemes. Let \(H\) be a random variable of a glyph consisting of \(K\) grapheme random variables \(\{G^1, \ldots, G^K\}\). The \(i\)th grapheme random variable \(G^i\) consists of stroke random variables which will be formally defined shortly. The joint probability distribution of \(\{G^1, \ldots, G^K\}\) is factorized by the chain rule in the typical writing sequence as

\[
p(H) = p(G^1, \ldots, G^K) = \prod_{i=1}^{K} p(G^i|G^{i-1}, \ldots, G^1) = \prod_{i=1}^{K} p(G^i|C(X)),
\]

where \(C(X)\) is a shorthand for the set of conditioning variables of \(X\).

A glyph is governed by the joint distribution of strokes. Let \(S^j\) be \(j\)th stroke of \(G\) and \(N\) be the number of strokes of grapheme \(G^i\). Then the random variable \(G^i\) is defined as \(G^i = \{S^1, \ldots, S_{N^i}\}\). Applying the chain rule again, the conditional distributions \(p(G^i|C(G^i))\) can be factorized as \(p(S^j|C(S^j))\).

A stroke is governed by the joint distribution of points.
Let $Q_{jk}$ be $k$th point of $j$th stroke at $G^i$ and $L$ be the number of points of stroke $S^j_i$. Then the random variable $S^j_i$ is defined as $S^j_i = \{Q_{j1}, \ldots, Q_{jk}\}$. By chain rule, each conditional distribution in $\prod_j \prod_l^K p(S^j_i | C(S^j_i))$ can be further rewritten as $\prod_j \prod_l^K \prod_j^N \prod_j^m p(Q_{jk} | p(Q_{jk}'))$. Each conditional distribution probabilistically represents the spatial relationships between points. As a result, the set of conditional distributions represent both intra-grapheme-relationships and inter-grapheme-relationships.

However, the cardinality of $C(Q_{jk})$ is likely to be very high, because stroke is a collection of points and a glyph consists of several strokes. In addition, the point $Q_{jk}$ is two dimensional continuous random variable. Thus, estimating such high order distribution demands huge space, computation time, and large number of training data if the conditional probability table is used. To alleviate this problem, a parametric form of conditional distribution is adopted, and a high order conditional distribution is approximated by low order ones.

### 2.2 Parametric Form of Conditional Distribution

The conditional distribution in the form of $p(X|C(X))$ is modeled as a linear regressive Gaussian [17]. In this model, a position of point is estimated by linear combination of the values of conditioning variables with Gaussian noise. It effectively represents dependency between continuous random variables without considering conditional probability table. Its expected value or mean is determined from the summation of the weighted values of conditioning variables. Hereafter, it will be used to visually represent any distributions. The parameters, i.e., the weights and covariance matrix can be estimated from training data using Maximum Likelihood Estimation or Expectation Maximization techniques.

### 2.3 Selecting Important Points

To approximate the high order distribution into a manageable low order one, we limit the number of point relationships and find important points.

Cho and Kim [18] have experimentally shown that positions of intra points in a stroke can be mostly determined by the positions of the given stroke end points. Following their study, a stroke’s intra points are assumed to be independent of the other strokes’ intra points.

Quantitative measuring the strength or importance of relationship between a response variable and a variable in the $n$ conditioning variables enables us to select less number of conditioning variables, $k$. This task has been accomplished by using mutual information (MI) [19]–[21]. We also use MI for finding the important stroke end points. The number of important points is chosen experimentally ($k = 3$), and it was sufficient for maintaining the spatial relationships. Figure 6 shows the coarticulated effect of the vowel (VW) on the first consonant (FC). Such kind of coarticulation effect happens because the selected important stroke end points in Fig. 6(b) mostly determine the positions of the vertical stroke of VW.

Although a subset of points is selected as important points as in Fig. 6, the expected strokes of a grapheme show natural geometry. This is because not only the near points but also some far points are selected as important points and both local and global relationships are reflected.

### 2.4 Layout

The layout of a glyph is represented as points which constitute bounding boxes of graphemes. The bounding box is a rectangle which encloses whole trace of a grapheme and described by two points. It can be directly obtained from glyph shape, but its randomness is different from randomness of glyph shape. From this reason, we separately model layout of glyph from shape. The bounding box is a rectangle which encloses whole trace of a grapheme and described by two points.

A layout of glyph is governed by a joint probability distribution of bonding boxes of graphemes. Let $B_i$ be a bounding box of grapheme $G^i$. The joint probability distribution is factorized by the chain rule in the typical writing sequence as $p_B(H) = \prod_i p_B(G^i | G_1^i, \ldots, G_2^i) = \prod_i p_B(B_i | C(B^i))$, where $C(B^i)$ denotes the set of bonding boxes of previously written graphemes. Since $B^i$ consists of two points $\{P_1^i, P_2^i\}$, $\prod_i p_B(B_i | C(B^i))$ becomes $\prod_i \prod_j p_B(P^i_j | C(P^i_j))$. Each conditional distributions is modeled as linear regressive Gaussian [17] as well. The cardinality of $C(P^i_j)$ is also reduced by selecting important points in $C(P^i_j)$ by the proposed method.

Figure 7 shows the coarticulated positions and sizes of VWs (–+) and LCs (−−) on various FCs (0), and shows configurations of selected important corner points for ‘+’.

### 2.5 Global Appearance

The coarticulation effect globally affects appearance of the
are estimated by minimizing error between the points in the parameters which linearly transforms the glyph model. To incorporate the global appearance into the glyph model, the global appearance is represented as a set of parameters such as slant, skew, or size. Such global appearance is one of the important traits of individuality of handwriting. To incorporate the global appearance into the glyph model, the global appearance is represented as a set of parameters which linearly transforms the glyph model.

The points of a glyph example are viewed as linearly transformed points of the glyph model by unknown parameter values. The transformation parameter values for a glyph are estimated by minimizing error between the points in a glyph example and the points in a glyph model. The former points are obtained by matching the example to the glyph model, the latter points are obtained by finding the expected points from the glyph model. The minimization is accomplished by well known least square method. Once the parameter values are estimated, the distribution of corresponding glyph is transformed by those parameters.

3. Handwriting Synthesis Algorithm

When the glyph to be synthesized already exists in the training examples, its shape and layout are learned by the proposed glyph modeling method (Sect. 2). The global appearance is also learned by assuming that each parameter follows Gaussian distribution. Then the most likely instance in the learned distribution is generated with slight variation.

If a grapheme of an unseen glyph is present in a different glyph in the examples, we reuse such graphemes which fit “naturally” to the unseen glyph. A criterion for selecting a grapheme is how similar coarticulated graphemes are. This is called interchangeability, and the detail of it will be explained shortly. When a grapheme of an unseen glyph is not present, it is predicted from the glyph model with context (previously synthesized grapheme instances). A brief algorithm of synthesizing glyph is shown in Algorithm 1.

Algorithm 1 Proposed algorithm for synthesizing glyph

1: Estimate global appearances of given examples
2: if The target glyph is in examples then
3: Generate the most likely instance with slight variation
4: else
5: Select the most interchangeable first grapheme instance among examples with given target glyph model
6: if The next grapheme instance exists in example then
7: Select the most interchangeable one conditioned on context
8: else
9: Predict it from the target model conditioned on context
10: end if
11: Repeat step 6–10 until all graphemes in the target glyph are synthesized
12: end if

3.1 Measuring Interchangeability

If a grapheme in glyph naturally fits into a different glyph, we say the grapheme is interchangeable. In general, a grapheme in a glyph is interchangeable with that of other glyphs of the same structure type (Fig. 4 in Sect. 2).

The interchangeable grapheme of a glyph, however, could vary depending on their diverse coarticulation effects of different writers. As an example, in Figs. 8 (a) and (b), VWs ( ) of (1) are interchangeable with that of (1). In (c), VW ( ) of (1) is interchangeable with that of (3).

We have observed that human handwriting have small variations on global appearance, so each glyph example has its own parameter values.

We assume that the label or code of a glyph example is known. Thus, the glyph model of the example is known as well.
is most similar to VW (\(\ldots\)) of \(\mathbb{Z}\). The more similar conditional distribution a grapheme has, the more interchangeable the graphemes is.

Let \(V\) be the set of training examples and \(v\) an element of \(V\). The random variable for \(i\)th grapheme of \(v\) is denoted as \(G^{(v)}\) and the instance is denoted as \(g^{(v)}\). The glyph models for \(v\) and an unseen glyph \(u\) are respectively denoted as \(M_v\) and \(M_u\). Following such notations, \(g^{(v)}\) and \(g^{(u)}\) can also be defined for \(u\). Then we say that two grapheme instances \(g^{(v)}\) and \(g^{(u)}\) are maximally interchangeable with each other if the conditional distribution of \(G^{(v)}\) is the most similar to that of \(G^{(u)}\) among different \(M_v\)'s, \(\forall v \in V\).

Formally, we have

\[
v^*(i) = \arg\min_{v \in V} \sum_{j}^{N_v} \sum_{l}^{L} \text{dist}(\text{dist}(p(Q^{(v)}_{jl} | \tilde{C}(Q^{(u)}_{jl}) = \tilde{c}(Q^{(v)}_{jl}); A), p(Q^{(u)}_{jl} | \tilde{C}(Q^{(u)}_{jl}) = \tilde{c}(Q^{(u)}_{jl}); A))
\]

In the above equation, \(\text{dist}\) stands for a distance measure of two distributions. Here we use Kullback Leibler distance in which distance between joint distributions is the sum of distances between factorized distributions by use of the chain rule. We have two conditional distributions in \(\text{dist}\) function; \(\ldots\) of \(\mathbb{Z}\), and \(\ldots\) of \(\mathbb{Z}, \mathbb{X}\) (Fig. 9). The \(\text{dist}\) function computes the similarity of grapheme shape distributions by aggregating the similarity of factorized point distributions of the grapheme \(\ldots\) in each of \(\mathbb{Z}, \mathbb{X}\), and \(\mathbb{X}\). Suppose \(\ldots\) in \(\mathbb{X}\) is most similar to that of \(\mathbb{Z}\) by the equation. Then the shape of \(\ldots\) in \(\mathbb{X}\) is interchangeable with that of \(\mathbb{Z}\). \(\tilde{C}(X)\) and \(\tilde{c}(X)\) are the important points of \(X\) and their values, respectively (Sect. 2.3). Note that the instances \(g^{(v)}\) and \(g^{(u)}\) are not explicitly considered because \(g^{(u)}\) is unseen in this case. Besides, the values of transformation parameter \(A\) is shared between \(v\) and \(u\) while measuring interchangeability.

The most interchangeable bounding box of a grapheme is found by considering distributions of bounding boxes instead of those of shapes. As in the shape case, the similarity of factorized bounding box point distributions of the grapheme \(\ldots\) in each of \(\mathbb{Z}, \mathbb{X}\), and \(\mathbb{X}\) is measured (Fig. 9).

This approach has several advantages in synthesis of unseen glyph. Since it uses probabilistic distribution, underlying variance is absorbed. Besides, coarticulation effect in specific writer’s handwriting is likely to be obtained based on writer’s example, i.e., utilizing individuality appeared in one’s handwritten graphemes under coarticulation effect is major benefit of use of interchangeability.

### 3.2 Predicting Grapheme

When the grapheme of an unseen glyph cannot be found in the training example, its shape needs to be synthesized from the model. To synthesize naturally written shape, a predicted shape should also have coarticulation effect with previously synthesized shapes. While a coarticulated shape can be generated by sampling instance from its conditional distribution, we want the generated shape to have most likely coarticulation effect. This can be obtained by finding an instance whose probability is maximum. Such task is hierarchically performed: end points of strokes are generated first, intra points of strokes are generated next. In this manner, overall shape of the grapheme can be constructed by means of the set of most likely points. The most likely point is the conditional mean of linear regressive Gaussian (Sect. 2.3). The bounding box of grapheme is similarly predicted by use of bounding box distribution. Figure 10 shows examples of predicted grapheme shapes.

### 4. Evaluation and Discussion

#### 4.1 Experimental Data

The data for learning distribution was collected from high school and college students. It consists of approximately 180,000 glyphs written by more than 100 writers. There was no restriction or guidance in the writing styles. As a result, cursive as well as blocky writing style was observed in the data.

For the synthesis of personal handwriting, we additionally collected set of handwriting from 8 writers. The text used was a poetry that appeared in a high school textbook. It has 5 verses and total of 308 characters. Among the 308, there were 144 unique glyphs. Each writer wrote the text 5 times. Personal handwriting is synthesized by each time taking one verse for synthesis, and using the other as the training examples of the writer’s handwriting. It means that the glyphs appeared in 4 verses are assumed to be seen, those in 1 verse are assumed to be unseen. In this way, we can visually compare machine-generated glyphs with human-generated ones.

Since we collected human handwriting samples using tablets and Tablet PCs, those samples might be different from real life handwriting. Hangul, however, consists of mostly linear strokes, so that handwriting using Tablet
doesn’t make big difference with real life handwriting when a little practice has been taken.

4.2 Examples of Synthesized Handwriting

Figure 11 shows true handwriting and synthesized handwriting of three writers.

We now show how the unseen glyphs in Fig. 11 were synthesized in the case of Writer 1. Figure 12 shows the synthesized handwriting and a table containing the glyphs of the training examples which have been used for synthesizing unseen handwriting. The tags of the table whose prefixes are “S-” stand for shape, “L-” for layout. (FC, VW, LC) designation represents grapheme classes of the glyph in example.

The proposed system synthesized 란 (column 3) by selecting the shapes from {리, 반, 란} bounding boxes from {찰, 안, 란} for each grapheme. The selected shapes and bounding boxes have similar structures to 란 for VW (▲) and LC (▼). It shows that, the training examples that are structurally similar to the target handwriting are chosen. Likewise, 려 (6) shows that the shape of LC (▼) was selected in 려 whose VW (▲) belongs to the same grapheme class “▼”. Even when the training examples don’t have structurally similar examples such as 란 (11), the synthesized result is visually plausible. This demonstrates that unseen glyph is synthesized by means of learned interchange-abilities between graphemes.

4.3 Writer Dependency of Synthesized Handwriting

One of important aspects of personal handwriting synthesis is within-writer consistency (WWC) and between-writer distinctiveness (BWD). To show how well WWC and BWD are represented, a text block is synthesized for each of writers to compare writer dependency by means of visual inspection. Figures 13 (a) through (f) show examples of synthesized and reproduced handwriting of six writers. Each of synthesized unseen handwriting is consistent with seen handwriting within the writer, while distinctive to those of other writers on the whole. The global appearance, especially slant, is clearly shown in Fig. 13 (a) in contrast to the others. The position and size of synthesized characters is also consistent within writer, distinctive between writers, notably in Figs. 13 (c) and (e). As 닫 (column 1) in Figs. 13 (a) through (f) shows, typical writing styles such as class of allograph, position, and size of grapheme are well expressed. The allograph class of the LC (▼) at 닫 (1) in Fig. 13 (a) shows z-like pattern which is consistent with FC (▲) of 닫 (12), 닫 (27), and 닫 (32), distinctive from that of the others. The position, size, and spacing of graphemes show WWC and BWD as well.

4.4 Human Perception Evaluation

The goal of the evaluation is to quantitatively measure whether machine-generated handwriting can preserve personal writing style. On the assumption that the average opinion of human perception can be used for the evaluation, we designed a variant of a Turing test: a subject identifies handwritten images (test images) in terms of whether they are generated by a machine or human by referring to another set of images written by a human. When the test images do not preserve the personal writing style, they are regarded as being machine-generated.

A set of test images consist of 100 machine-generated
Table 1  Total results of 26 subjects’ evaluations. Each subject assessed 1,600 test images.

| Subjects’ judgment | Test images | M (positive) | H (negative) |
|--------------------|-------------|--------------|--------------|
| M (positive)       | 2161        | 1456         |
| H (negative)       | 18639       | 19344        |

handwriting (M) and 100 human-generated handwriting (H). The subject judges each test image to be M or H by referring to another set of 100 images (reference images). Eight sets of test images are used for the evaluation. Thus, a subject should judge 1,600 test images in total. The exact numbers of M and H in the test images are not known to subjects so that they will evaluate the test images without preconceptions. The number of training examples for M is approximately 75.

The subject looks at two computer monitors, one showing test images and the other showing reference images, and evaluates test images one by one. The subject can freely re-examine previously evaluated test images at any time. To allow for a intensive evaluation, the subject is provided with unlimited time to accomplish the task.

Table 1 shows the aggregated judgment results of 26 subjects (9 lab members).

Type I error (1456) represents the portion of human-generated handwriting that was judged to being machine-generated. The false positive rate is 0.07 (1456/20800), which shows that only 7% of human-generated handwriting was not appeared to have individual writing style. Type II error (18639) represents the portion of machine-generated handwriting that was judged to being human-generated. The false negative rate is 0.896 (18639/20800), which shows that 89.6% of the machine-generated handwriting was falsely identified as being human-generated. This implies that 1) when a human wrote 100 examples, the subjects judged 7 examples to being machine-generated, 2) when machine wrote 100 handwriting, the subjects judged 89.6 handwriting to being human-generated.

4.5 Impact of Training Examples

As the number of training examples for machine-generated handwriting is increased, the machine-generated handwriting will become more similar to the original handwriting. Figure 14 shows the numerical differences between the original handwriting and the machine-generated handwriting for varying numbers of training examples. The numerical differences are measured by the root mean squared error (RMSE). We see that the RMSE is decreased as the number of training examples is increased.

4.6 Discussions on Extension to Other Languages

The application of proposed method to other language, such as Chinese or English, needs a little care. Firstly, grapheme-based probabilistic representation of handwriting is necessary. Secondly, the representation is supposed to have hierarchy. For instance, character-radical-stroke-point is one of good hierarchies of Chinese character and word-letter-stroke-point could be one of choices of English. Additionally, for English, more refined stroke modeling is needed to represent more degree of freedom because writing style of English is more cursive than that of Korean or Chinese characters. Extending the current synthesis procedure also include the inter-glyph spatial relationship.

5. Conclusion

We proposed a new method to synthesize personal-style handwriting based on a limited number of examples. We introduced four challenges in handwriting synthesis that are describing randomness, modeling coarticulation effect, representing individuality, and using limited number of examples. Our approach for the four challenges was 1) modeling probabilistic glyph model, 2) incorporating spatial relationships between graphemes into the model, 3) reusing glyphs and graphemes in examples, parameterizing global appearance, 4) defining interchangeability between graphemes in different glyphs, using commonly written grapheme if no other possible. The experimental results showed that synthesized handwriting was visually plausible. The evaluation result showed that, 1) when a human wrote 100 examples, the subjects judged 7 examples to be machine-generated on average, 2) when machine wrote 100 handwriting, the subjects judged 89.6 handwriting to be human-generated on average.

Acknowledgement

The authors would like to thank Dr. Se June Hong for extensive English editing of the manuscript. This research is supported by the Ubiquitous Computing and Network (UCN) project, the Ministry of Information and Communication (MIC) 21st century frontier R&D program in Korea.

References

[1] S.N. Srihari, S.H. Cha, H. Arora, and S. Lee, “Individuality of handwriting,” Journal of Forensic Sciences, vol.47, no.4, pp.1–17, 2002.
[2] S. Ling, “A preliminary investigation into handwriting examination by multiple measurements of letters and spacing,” Forensic Science International, vol.126, no.2, pp.145–149, 2002.

[3] I. Guyon, “Handwriting synthesis from handwritten glyphs,” International Workshop on Frontiers of Handwriting Recognition, 1996.

[4] K.L. Pak, D.Y. Yeung, and M.C. Pong, “Chinese glyph generation by heuristic search,” Technical Report HKUST-CS96-17, Dept. of Computer Science, Hong Kong University of Science and Technology, 1996.

[5] Z. Lin and L. Wan, “Style-preserving English handwriting synthesis,” Pattern Recognit., vol.40, no.7, pp.2097–2109, 2007.

[6] R. Plamondon and W. Guerfali, “The generation of handwriting with delta-lognormal synergies,” Bilingual Cybernetics, vol.78, pp.119–132, 1998.

[7] R. Plamondon, W. Guerfali, and X. Li, “The generation of oriental characters new perspectives for automatic handwriting processing,” International Journal of Pattern Recognition and Artificial Intelligence, vol.12, no.1, pp.31–44, 1998.

[8] D. Lee and H. Cho, “A new synthesizing method for handwriting Korean scripts,” The Visual Computer Journal, vol.17, no.3, pp.147–157, 2001.

[9] M. Mori, A. Suzuki, A. Shio, and S. Ohtsuka, “Generating new samples from handwritten numerals based on point correspondence,” International Workshop on Frontiers in Handwriting Recognition, pp.281–290, 2000.

[10] A. Herzmann, N. Oliver, S. Seitz, and B. Curless, “Curve analogies,” Eurographics Workshop on Rendering, pp.233–246, 2002.

[11] T. Varga and H. Bunke, “Generation of synthetic training data for an HMM-based handwriting recognition system,” International Conference on Document Analysis and Recognition, pp.618–622, 2003.

[12] Y. Zheng and D. Doermann, “Handwriting matching and its application to handwriting synthesis,” International Conference on Document Analysis and Recognition, pp.861–865, 2005.

[13] J. Wang, C. Wu, Y.Q. Xu, H.Y. Shum, and L. Ji, “Learning-based cursive handwriting synthesis,” International Workshop on Frontiers in Handwriting Recognition, pp.157–162, 2002.

[14] J. Wang, C. Wu, Y.Q. Xu, and H.Y. Shum, “Combining shape and physical models for online cursive handwriting synthesis,” International Journal on Document Analysis and Recognition, vol.Online First, 2004.

[15] R. Niels and L. Vuurpijl, “Generating copybooks from consistent handwriting styles,” International Conference on Document Analysis and Recognition, pp.1009–1013, 2007.

[16] C.V. Jawahar and A. Balasubramanian, “Synthesis of online handwriting in Indian languages,” International Workshop in Handwriting Recognition, 2006.

[17] K. Murphy, “Inference and learning in hybrid Bayesian networks,” Technical Report 990, U.C. Berkeley, Dept. Comp. Sci., 1998.

[18] S.J. Cho and J.H. Kim, “Bayesian network modeling of strokes and their relationships for on-line handwriting recognition,” Pattern Recognit., vol.37, no.2, pp.253–264, 2004.

[19] T.M. Cover and J.A. Thomas, Elements of Information Theory, John Wiley and Sons, 1993.

[20] C.K. Chow and C.N. Liu, “Approximating discrete probability distributions with dependency trees,” IEEE Trans. Information Technology, vol.14, no.3, pp.462–467, 1968.

[21] I.J. Kim and J.H. Kim, “Statistical character structure modeling and its application to handwritten Chinese character recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol.25, no.11, pp.1422–1436, 2003.