Calligraphic Beautification of Handwritten Chinese Characters: A Patternized Approach to Handwriting Transfiguration[1]

Xinghua Zhu *Lianwen Jin**
School of Electronic and Information Engineering
South China University of Technology
vcpudding@gmail.com ***lianwen.jin@gmail.com**

Abstract

In this paper, a novel method for online handwritten Chinese character beautification is proposed. User-input trajectories captured from a mouse or a preasure-senselesss touchscreen are first simulated into paint-brush style using a speed-based calligraphic simulation algorithm. Then a template-based beautification method is applied to transfigure the overall architecture of the user-written character. On a preprocessing stage, the user-written character is matched to the template using stroke matching technique. Each stroke of the character is also modeled by a tri-unit stroke model. The processed user-written character is finally fed to a simulated analogous reasoning module to fuse with a regular template character. Our proposed system is proved to be able to preserve users' individualities as well as to transfigure handwritten Chinese characters.

Keywords: Handwriting beautification, tri-unit stroke model, calligraphic simulation, digital Asian calligraphy, feature correspondence.

1. Introduction

Calligraphy has been one of the most important forms of art over thousands of years. Handwritten calligraphy outperforms printed type characters in aspects of expressional power such as flexibility and originality. It's even commonly believed that handwritings can reflect one's personality, emotions and educational level. These unique features define handwritings as an inseparable part of life and an irreplaceable form of art and culture.

Personalized calligraphic rendering tools are also drawing attention among public users as well as scientists. For example, Xu Jinglei(a pop star in China) style font [1] and Mao style font are published in 2007. However, these font styles are essentially traditional rendering methods that treat all characters as copies of template characters.

The problem of handwriting beautification has not been substantially addressed, but still a few studies remotely relating to this topic can be found in the literature, including calligraphy generation, handwriting synthesis, and trajectory-based calligraphy simulation. Songhua Xu et al proposed a system to generate novel calligraphic style through blending existing calligraphic works [2]. It resembles the ideal of handwriting beautification that the characteristics of one style can be partly preserved in the blending results. Nevertheless, the system Xu proposed is not compatible with the application in user-computer interactive environment. The topic of handwriting synthesis is more often addressed [3]. Both movement simulation and shape simulation techniques are intensively discussed in the literature. Although the learning-based handwriting synthesis method proposed in [3] achieved satisfactory experimental results with cursive English handwritings, this algorithm can not be directly applied to Chinese characters due to their complexity and huge vocabulary. A trajectory-based calligraphy simulation method is proposed in [4], which deforms user-input trajectories and results into Caoshu style calligraphy work. Unlike [2] and [3], the deformation is performed purely according to the features of a stroke. Yet the method proposed in [4] also has a number of drawbacks, namely the monotony of resulting style, loose control on the overall architecture of a character and lack of systematic mathematic model.

In this paper, we are going to introduce a novel technique that is able to beautify user-written hand-writings and yet preserve the personal characteristics of the user's handwriting. The personalized handwriting beautification technique will provide common users, other than specially trained calligraphers, with the chance to create better-looking handwritten materials.

[1] This paper is partially sponsored by the following research funding: NSFC (no. U0735004, 60772216), GDNSF (no. 07118074).

2. System Architecture

The main system flow chart of our proposed system is given in Figure 1. The approach we propose is driven by common input devices such as mousse and pressure-senseless touchscreens. As soon as users' input trajectory is captured, automatic character recognition is applied to user-written characters in order to read the corresponding template from the library. Also, a real-time stroke matching is performed on the user-input characters, which automatically match user input to the template character. Based on the fact that the shape of each stroke is not only affected by its own condition, but also affected by the ligature stroke between its former and latter neighbors, a tri-unit stroke model for Chinese characters is proposed. After applying a simulated analogous reasoning process to the extracted stroke models, a character with both better stroke shapes and overall architecture can be achieved.

The rest of this paper is arranged as follows. The algorithm of speed-based calligraphy simulation will be described in Section 3. Then the pre-processing stage, including stroke matching, tri-unit stroke model and recognition-driven character model, is discussed in Section 4. The application of simulated analogous reasoning is described in Section 5. Section 6 analyzes experimental results and performance of our proposed system. Finally, conclusions are given in Section 7.

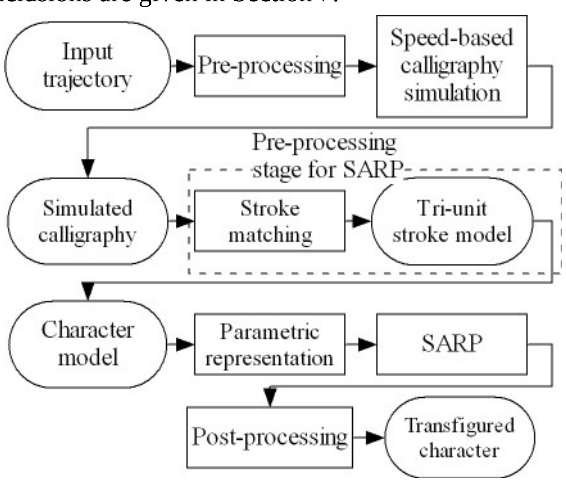

Figure 1. system flow chart

3. Speed-Based Calligraphy Simulation and Anti-Aliasing Rendering

3.1. Speed-Based Calligraphy Simulation

A number of studies were published on paint-brush calligraphy simulation using a tablet and an electronic pen as input devices [6]. However, as we observe, tablets and electronic pens are not common accessories among PC users, especially among those with no experience in computer graphics or calligraphy. On this basis, we assume the majority of users of the handwriting beautification system to use a mouse in a PC or touchscreens without pressure sensitivity （such as those in PDA's） as the input medium. Therefore, a speed-based calligraphy simulation (SBCS) method is adopted to turn a discrete trajectory into a paint-brush style stroke.

From a subjective aspect, the width of a stroke decreases if the brush moves faster, and increases when the brush moves slower. Correspondently, denoting the writing velocity as v, the width of the current segment should be defined as a function of v:

$$w = f(v) \quad (1)$$

When simulating the physical paint-brush behavior, an abrupt change of stroke width must be prevented. Thus the definition of segment width is modified as

$$w = f(v, last\_w), \quad (2)$$

where *last_w* is denoted as the width of the last segment.

A detailed definition of ***f*** is given by equation 3. Constants A, B and C can be adjusted to accommodate with users' preference of writing speed and pen width.

$$w = f(v, last\_w) = \begin{cases} A \cdot \exp\{-B \cdot v + C\}, |w - last\_w| / last\_w \le thres \\ last\_w \cdot (1 + thres \cdot sign(w - last\_w)), others \end{cases} \quad (3)$$

3.2. Anti-Aliasing Rendering

While we simulate the paint-brush style calligraphy using SBCS, anti-aliasing technique is also applied to smooth the contours of the strokes.

The contours of certain patterns such as a line, an ellipse or an arc are often jig sawed resulted from the lost of precision during quantization. This phenomenon is called aliasing in computer graphics. A series of anti-aliasing techniques have been proposed in the literature. We employ the Wu method [5] to draw anti-aliased ellipses and thus construct strokes of dynamic widths using ellipses of corresponding radiuses whose centers are lined following the original trajectory. A compare of anti-aliased stroke with non-anti-aliased one is shown in Figure 2.

4. Stroke and Character Models

4.1. Stroke Matching

Stroke matching is one key part for our proposed handwriting beautification method. Unacceptable output may occur if mismatched strokes are delivered into the simulated analogous reasoning process. It's necessary for the system to accommodate to variations in common handwritings, giving prominence to the mismatch of stroke numbers between user written characters and template characters.

To obtain identical number of strokes with the template character, a stroke is split into two if the number of user strokes is smaller than that of template ones; or two strokes are merged into one otherwise. This process is iterated until a user character achieves the same number of strokes with its template. The ideal is straight forward but not accurate. Noticing that the ultimate target of stroke splitting or merging is to match user strokes with its templates, we propose an advanced method to solve the problem of mismatched stroke numbers.

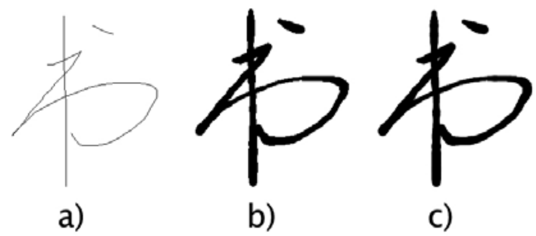

Figure 2. a) Original trajectory; b) SBCS stroke; c) anti-aliased SBCS stroke

Without loss of generality, we assume that the original user character has fewer strokes than its template, which means we must split one or more of the user strokes at appropriate positions. The splitting method we propose is based on the ideal of combining splitting with verification.

First of all, critical points on all strokes of the user character are detected, which are the points with the local maximum curvatures. The stroke on which the maximum curvature point is located is pre-split into two strokes. With this pre-split result, the system would verify its authenticity. The result will be accepted if the verification succeeds. Or else the result will be rejected and the pre-split is canceled. In this case, the next maximum curvature critical point will be tested and verified until all critical points are traced or one eligible split is found.

A vector distance calculator serves as the verification. Dynamic Time Warping (DTW) [7] is a measure of the distance between two vectors of different dimensions. Long et al used DTW algorithm to calculate the distance between the feature vectors of two Chinese character strokes, and thus reflect the degree of resemblance between the two strokes [7]. We employed this algorithm as the verification of stroke matching. Each time the user character is pre-split into the same number of strokes with its template, a distance is calculated between each user stroke and its template. If the distance is within the range of a predefined threshold, the user stroke is verified to be properly matched with its template. Otherwise a mismatch between user stroke and its template is indicated, which informs the system to reject the pre-split state.

In experiments, the threshold of feature vector distance can be adjusted through graphic user interface. Relaxing the threshold may give more liberty to users' writing style, while restricting it ensures more accurate matching result.

On the other hand, due to the difference in users' writing habits, one stroke may be inputted as two successive strokes. These separated strokes can be handled similarly with the concatenated ones discussed above.

An illustration of the proposed stroke matching method is shown in Figure 3.

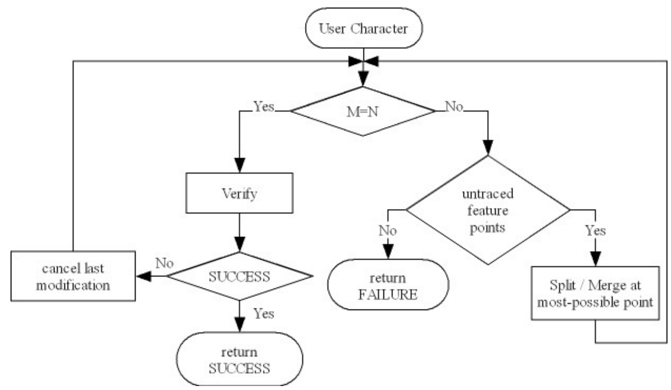

Figure 3. Flowchart of stroke matching

In experiments, our proposed algorithm is able to match user-input characters to its template correctly (see Figure 4). However, errors may occur if the inputted character is written too scratching for the system to handle.

Figure 4. Examples of stroke matching (the start point of each stroke is marked with a black dot): a) user characters; b) template characters.

4.2. Tri-unit stroke models

Concatenation of successive strokes is a common phenomenon in Chinese calligraphy creation. Concerning writing styles that are mostly regular, although connection parts between successive strokes are not necessary to construct a stroke or a character, they have unneglectable influence on handwriting styles. Therefore, a stroke model that takes the connection part expressly is required to ensure the preservation of users' handwriting styles.

In cursive English handwriting, each letter is called "context related", which means that each letter is connected to and affected by its two neighbors in specific ways. Wang et al proposed a tri-unit handwriting model for cursive English letters [3]. As components of handwritings, a stroke in a Chinese character plays a similar role with an English letter, especially in the way they interact with their former and latter neighbors. Inspired by this idea, we propose a tri-unit model for Chinese character strokes. Similarly, the trajectory of each stroke, obtained after the stroke matching procedure, is divided into three parts in sequence: the head, the body and the tail units. The head unit and tail unit are the connection parts with the previous and next strokes respectively.

The algorithm we used to decide the position of head, body and tail units is similar with the stroke matching algorithm proposed in 4.1. Whenever a user stroke is split into two in the stroke matching procedure, the resulting strokes are defined as concatenated strokes. To these concatenated strokes, the connection-part-determination algorithm is applied.

Initially, we set the head unit of each stroke to be the first point of the stroke; the tail unit the last point of the stroke; and the body unit the complete trajectory of the stroke. The body units are first compared with template strokes, which are written in a most regular manner. The overall distance between user strokes and template strokes is denoted as D. Then, starting from the connection point, which is also the joint point of two concatenated strokes, we search for the first critical point previous and next to the connection position. If such critical points exist, they are assumed to the boundaries of the connection part by default. The connection part is thus removed from the body units of the correspondent strokes. Afterwards, the body units are again compared with the template strokes, with the overall distance denoted as $D^{'}$. The connection boundaries are proved if $D' < D$, and denied otherwise.

Figure 5 illustrates the procedure of connection-part-determination algorithm. Figure 6 gives some examples of concatenated strokes and determined connection part.

4.3. Recognition-driven Character Model Creation

In our proposed system, user-input characters are created in a dynamic manner, as described in previous sections. In contrast, to create template character models, a predefined static template library is employed for stroke extraction methods are not able to achieve 100% validity. In experiments, the library we used is in Kai font style, a most frequently used regular font style in both printed and handwritten materials. The trajectory of every stroke in a character is distinctly recorded in the static template library, in the sequence of commonly accepted orders of strokes. Once users finish writing in the input window, the written character is automatically recognized by an MQDF classifier for handwritten Chinese characters [8]. Afterwards the system loads the template of the recognized character from the library.

5. Generating Beautified Handwriting

Songhua Xu et al proposed an algorithm to blend static calligraphy and produce calligraphic work of novel styles, named the simulated analogous reasoning (SARP) algorithm [2]. We employed and made a few improvements to the SARP algorithm to make the overall architecture of user-input handwritings better.

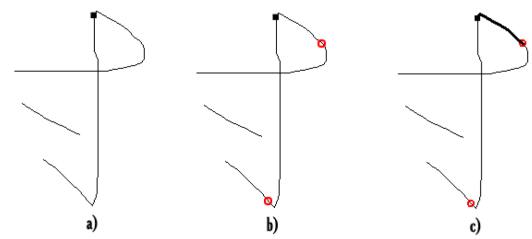

Figure 5. Illustration of connection part determination: a) one split point of trajectory is located (marked with a black rectangle); b) the nearest turning points back and forth the split point is found (marked with circles); c) the connection part is determined (bold line).

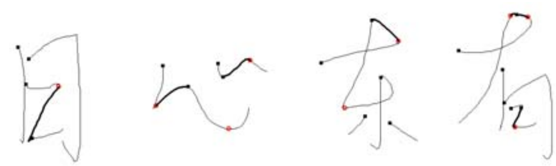

Figure 6. Examples of connection part determination

5.1. Overall Architecture Transfiguration

As mentioned above, we employed a simulated analogous reasoning module in our system to transfigure the general architecture of user characters. In this procedure, character models created in previous sections are fused in SARP. In our system, SARP is performed on stroke level, the naturally inherited individual components in real handwritings.

Denoting parametric representation of the user stroke as P_{usr} and its template P_{std}, the mathematic principle of SARP can be stated as:

$$R = \omega_{usr} * P_{usr} + \omega_{std} * P_{std} \tag{4}$$

ω_{usr} and ω_{std} are defined as analogous intensities of fusing sources. Typically, they obey the constraints that $\omega_{usr} + \omega_{std} = 1, \omega_{usr} \geq 0, \omega_{std} \geq 0$, implying that the resulting style is a mimic of the template style made by individual writers, varying in the extent to which it assembles the template style.

To expand the ideal into fusing styles of various sources, the equation can evolve into

$$R = \sum_{i=1}^{n} \omega_i * P_i \quad (5)$$

where i = 1, 2, …, n is the indices of source styles, with i=1 being the user input style.

Here we are assuming a one-to-one correspondence among different versions of P_i. In reality, a stroke derived from one example can correspond to one stroke in another training example in more than one way. The process therefore requires a feature correspondence process, discussed in the next section, before it can blend together features extracted from the different examples.

SARP can be viewed as a process that synthesizes new knowledge—shapes, in this case—by fusing or blending with existing, independent knowledge sources—that is, the training examples [2]. In concern with Chinese character beautification, it particularly simulates a real process of learning calligraphy. Generally, we start practicing calligraphy by imitating work pieces of masters, and gradually engender styles of our own. It's coherent with SARP in the ideal of fusing existing knowledge with the style of individual writers.

5.2. Feature Correspondence between User Stroke and Its Template

Numbers of Chinese stroke types contain one or more turns, such as 乛 ㇆ ㇛, etc. In real paint-brush handwriting, featured shapes are made at the turnings by calligraphers. Therefore, the shape of strokes at a turning point is one of the crucial reflections of writers' styles. Since, when a correspondence between user stroke and its template is established, we try our best to ensure that the turning points on both sides correspond to each other approximately. A sectional linear algorithm is used in [2] to set up feature correspondence between different samples. Yet we found its performance unsatisfactory. Therefore, we propose another means to obtain an optimal correspondence between user stroke and its template.

Again, DTW algorithm is adopted in our feature correspondence method. In DTW algorithm, the feature correspondence path must not go backwards. Also, every step the path takes must result in a minimum local distance. Based on these premises, starting from the top-right corner of a distance table, we may trace the correspondence path backwards following a route of minimum local distance. Examples of feature correspondence are shown in Figure 7. As it shows, feature correspondence using DTW achieves more appropriate correspondence regarding turning point matching.

6. Experimental Results

We designed a series of experiments to evaluate the performance of our proposed handwritten Chinese character beautification method. First of all, characters written by a user are transfigured by our system to

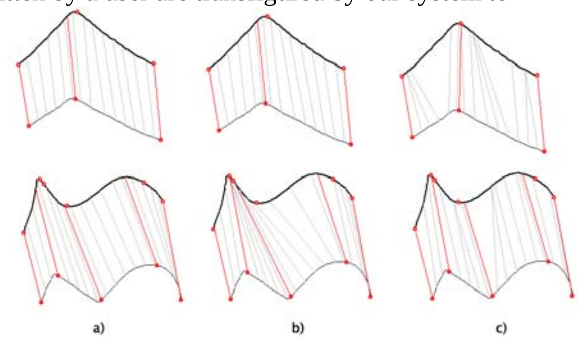

Figure 7. Examples of feature correspondence between a stroke and its template: a) linear feature correspondence; b) sectional-linear feature correspondence; c) DTW feature correspondence.

prove its ability in accommodating to various characters as well as in producing better-looking Chinese characters according to user inputs (see Figure 8). Secondly, a comparison of fusing results produced with different SARP intensities is presented in Figure 9. As it shows, users have an effective control over the performance of our proposed system. By adjusting fusing intensities according to their

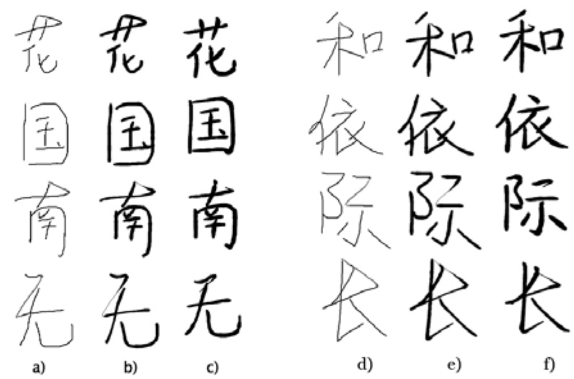

Figure 8. Characters written by the same user: a) d): original trajectory; b) e): speed-based simulated calligraphy; c) f): results of analogous reasoning.

needs and favor, it's easy for users to obtain writing styles they want. Moreover, samples of a character "光" (meaning "the light" in Chinese), written by several users respectively, are given in Figure 10. It gives evidence on the user-independency of our proposed system. Also, our system proves to be capable of preserving users' individual writing styles as shown in Figure 10.

A comparative result is also given in Figure 9. In the last row, a character "章" written by the same user with that in the 5^{th} row, is transformed with ougishi system [4]. As it shows, ougishi system failed to differently transfigure the unshapely written character according to different cursive levels, while our system shows an effective control on transfiguration level.

Figure 9. Five characters transfigured using a series of intensities: in row 1~5, the intensities of template-resemblance used from left to right is: a) 0 b) 0.2, c) 0.4, d) 0.6, e) 0.8, f) 1.0; row 6 shows the results produced by the ougishi system [4], where the cursive level are: a) 0, b) 2, c) 4, d) 8, e) 10, f) 12.

Figure 10. A character written by five different writers: a) original inputs, b) results of transfiguration.

7. Conclusions and Future Work

In this paper we introduced a novel ideal to transfigure the appearance of user-written Chinese characters. Based on a regular template library, trajectories captured from a mouse or a pressure-senseless touchscreen is simulated into calligraphic style and beautified in overall architecture. The system is able to preserve users' own individualities while obtaining better-looking hand-writings. Experimental results are given to illustrate the effectiveness of our system on transfiguring handwritings and preserving users' originality.

There're several potential applications for our handwritten Chinese character beautification system. First, it may serve as a component in the computer graphic toolbox that specifically helps improve the appearance of handwritten Chinese characters. Also, it's useful in computer-aided language learning (CALL) by giving advices on how to improve handwriting quality on the basis of characters written by the students, which makes the CALL even more individualized and intelligent. Moreover, our system can be implemented into PDA's, generally configured with pressure-senseless touchscreens. Under circumstances where signatures, drawings on a greeting card, or other handwritings are needed, ink marks can be greatly beautified using our proposed system. Furthermore, fed with only one set of handwriting samples, our proposed system could produce multiple times of samples in different styles. This is an excessively effective way to obtain massive samples required in training handwritten Chinese character classifiers [10].

A number of issues remain to be addressed in future work. The success of transfiguration heavily depends on the results of stroke matching. More studies have to be done in this area to improve the accuracy and flexibility. Secondly, in our prototype system, the user-inputted character is fused only with one template style. We're going to carry out experiments that allow multiple template styles in the fusing process in order to produce diverse styles in the results. Last but not the least, the current system is built on single character processing. However, in reality a calligraphic work often composes of a number of characters and is made as an integral entity. Therefore, the system needs to be expanded to the level of integral calligraphy rendering in the future.

References

[1] "Computer font libraries entered the characterized era", *http://www.founder.com.cn/cn/news_info/2007-12/31/content_5165.htm*

[2] Songhua Xu, et al. "Automatic Generation of Artistic Chinese Calligraphy", *IEEE Intelligent Systems*, vol. 20(3): 32-39, May/June 2005.

[3] Jue Wang, Chenyu Wu, et al. "Learning-based Cursive Handwriting Synthesis", *Frontiers in Handwriting Recognition*, 2002, pp. 157-162.

[4] Mikihiro OHBA. Ougishi, *http://www.ne.jp/asahi/o/o/ougishi/*

[5] Xiaolin Wu, et al, "Double-step incremental generation of lines and circles", *Computer Vision, Graphics, and Image Processing*, Academic Press Professional, Inc. San Diego, CA, USA, 1987.

[6] Junji Mano, et al, "A Method to Generate Writing-Brush-Style Japanese Hiragana Character Calligraphy," *icmcs*, p. 9787, 1999 IEEE International Conference on Multimedia Computing and Systems (ICMCS'99) – vol. 1, 1999, pp. 9787-

[7] Teng LONG, Lianwen Jin, "One stroke cursive character recognition using combination of directional and positional features ", *Proceedings of IEEE ICASSP* 2005.

[8] Teng Long, Lianwen Jin. "Building Compact MQDF Classifier for Off-line Handwritten Chinese Characters by Subspace Distribution Sharing", *ICDAR*, Sep. 2007.

[9] Chenglin Liu, "Vector-to-Image Transformation of Character Patterns for On-line and Off-line Recognition", *International Journal of Computer Processing of Oriental Languages*, vol. 15, No. 2 (2002), pp. 187 – 209.