Synthesis of Chinese Character Using Affine Transformation

LianWen JIN XiaoNa ZU
School of Electronics & Information,
South China University of Technology, Guangzhou, 510640, China
lianwen.jin@gmail.com

Abstract

We present a novel Chinese synthesis method based on affine transform. A set of basic Chinese character element (BCCE) are designed, which can be used to generate any Chinese character in standard GB2312-80 level 1. Structure similarity measurement is used to evaluate the synthesis quality. Experiments showed that the synthesized characters look smooth and natural. Storage of synthesized characters can be greatly reduced. The proposed Chinese character synthesis method has many potential applications, such as building small-size Chinese font, building compact classifier for Chinese OCR, and etc.

1. Introduction

Character is an important human-human and human-machine communication tool. The Chinese character has the largest users around the world. However, the number of Chinese characters is very huge. There are 6763 characters in the National Standard GB2312-80 set, 27484 characters in standard GB18030-2000 set, and more than 47000 characters can be found in Kang-Xi dictionary. About 3755 characters in GB2312-80 level 1 are most frequently used in daily life (Covering more than 99.5% usage frequency). To encode and store so many Chinese characters on computer requires vast storage. Although there exists efficient parametric graphics encoding approach for designing and storing Chinese fonts[10], the size of most Chinese characters in True Type Font (TTF, a popular standard curve parameter approach for most Windows font representation) is still very large. For example, the size of Song TTF file on Windows XP is 10.2M）. This leads to some problems when trying to display and process of Chinese fonts on hand-held devices with limited storage resource.

Chinese character is an ideograph and is composed of many relatively independent elements. The element may be a stroke, a radical, or a sub-character element [1]. It is a fact that many Chinese characters share same or similar elements. For example, the characters “林”, “树”, “枝”, “果” share the similar radical “木”, “投”, “股”, “段”, “殷” share similar element “殳”. The character composition can be considered in a hierarchical manner. For example, the character “啊”can be constructed from “口”, “阝” and “可” , the character “题” can be constructed from “是” and “页”. Therefore, we can design a set of basic character elements to represent (or synthesize) any character in some way. The storage can be certainly reduced efficiently if we could find a small basic character element set.

Several attempts have been carried out to synthesize Chinese characters, especially for characters that are unrepresented in the existing Chinese fonts. Dong and Li [3] proposed a Chinese character design system which took a parametric approach to create characters in different styles. Lim and Kim [4] developed a system for designing oriental character fonts by composing stroke elements. Yiu and Wong [5] defined a Chinese character description language named HanGlyph to capture the topological relation of strokes in a character. A character is generated by rendering the HanGlyph expressions of basic strokes. Most of these methods try to create a character according to its composition of strokes and radicals. But it does not seem to be very successful due to the complex structure of Chinese strokes and radicals in different characters. P.K Lai et.al.[2] proposed a method to use character composition expressions (CCE) to encode unrepresented characters by specifying the structural relationships between character components [2]. Based on the use of CCE, Chinese glyphs can be generated in a heuristic way. However, the transform operators in [2] only contains linear shift, and size scaling (shrink or enlarge), which cannot model the radical variations accurately.

Generating Chinese character from a basic character element set (strokes, radicals or components) is an intuitive and feasible way. However, due to the

Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)
0-7695-2822-8/07 $25.00 © 2007 IEEE

different deformation of elements in characters, synthesis using simple transform such as linear shift and size scaling (shrink or enlarge) can not always get a satisfactory looking result in which the proper ration of element could not be preserved, as shown in figure 1.

Fig.1. The radical "木" in two characters "揸" and "架" cannot be proper generated (in red color) using shift and scaling transform.

In this paper, we present a novel Chinese synthesis method based on affine transformation. 490 basic Chinese character elements are designed, which can be used to generate any Kai-font Chinese character in GB2312 Standard set 1. Experiments showed that the synthesized characters look smooth, natural and perceivable satisfactory. Comparing with conventional method that encodes each character independently, the storage of synthesized characters can be greatly reduced by our method.

2. A framework for Chinese character synthesis

In general, a Chinese character (CC) is composed of several character elements:

$$\mathrm{CC} = \{\mathrm{R}_1, \mathrm{R}_2, \cdots\cdots \mathrm{R}_n\} \quad (1)$$

where R_i is a character element image, which can be denoted by n_i black pixels as following:

$$\mathrm{R}_i = \{\mathrm{t}_1, \mathrm{t}_2, \cdots\cdots \mathrm{t}_{n_i}\} \quad (2)$$

where $\mathrm{t}_j = (\mathrm{x}_j^t, \mathrm{y}_j^t)^T$ is the j^{th} pixel of R_i.

In contrary to the tremendous number of Chinese characters, the number of character elements is relatively small (usual less than 500) and can be shared between different characters. We defined a Basic Chinese Character Element (**BCCE**) set as：

$$\mathrm{BCCE} = \{\mathrm{E}_1, \mathrm{E}_2, \cdots\cdots \mathrm{E}_m\} \quad (3)$$

where the i^{th} basic element image E_i is denoted by m_i black pixles:

$$\mathrm{E}_i = \{\mathrm{s}_1, \mathrm{s}_2, \cdots\cdots \mathrm{s}_{m_i}\} \quad (4)$$

with $\mathrm{s}_j = (\mathrm{x}_j^s, \mathrm{y}_j^s)^T$. The basic element E_i may be a stroke, a radical, or a character component.

It is expected that an arbitrary Chinese character could be generated (synthesized) from the BCCE set in some way. However, it should be noted that the same BCCE may appear in different position with different deformation when it is used to synthesize different Chinese characters. Therefore, a proper transformation which that can model such deformation should be built for the synthesis task when it is employing to map the BCCEs to target Chinese character.

A Chinese Synthesis problem can be defined as follows: Given an arbitrary character CC，we want to seek for a transformation function $\phi : E_i \rightarrow R_i$, so that all character elements of CC can be generated from the BCCE set:

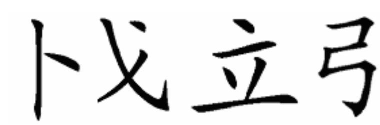
$$\mathrm{R}_i \approx \phi(E_i) = \{\phi(\mathrm{s}_1), \phi(\mathrm{s}_2), \cdots\cdots \phi(\mathrm{s}_{m_i})\} \quad (5)$$

for all i.

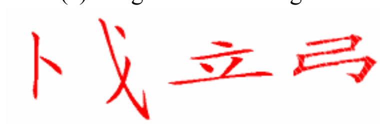
By this way, we can use the BCCE set and the transform ϕ to synthesize any character. In this paper, the transformation ϕ we used is Affine Transformation (**AT**), defined by:

$$\phi(\mathrm{s}_i) = \mathbf{A}\mathrm{s}_i + \mathbf{b} \quad (6)$$

where $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$ are six parameters of AT. Affine transformation can be used to represent rotation, scaling, shearing and even some rigid deformation of image/shape [8]. Some examples of affine transform of several BCCEs are shown in figure 2.



(a). Original BCCE images.



(b). Corresponding AT images of (a).

Fig.2. Example of affine transformation of four BCCEs.

The Chinese character synthesis approach we proposed in this paper contains two phases. In the training phase, the BCCE set is first designed, then the Chinese character are decomposed into several character elements (eg. the character "啊"can be decomposed into "口", "阝" and "可") and finally, several BCCEs are selected accordingly to build a set of affine transformations between them and the decomposed character elements. For each AF, six parameters are stored. In the synthesis phase, with the corresponding stored AT parameters and the BCCEs, any character can be synthesized using affine transformation. The diagram of our Chinese synthesis is shown in figure 3.

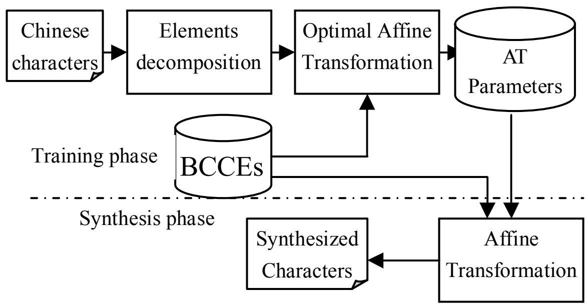


Fig.3. The diagram of Chinese synthesis based on affine transformation.

From figure 3, it can be seen that there are three key modules for the synthesis process, say, (a) decomposition of Chinese character; (b) selection of corresponding character elements from BCCE set during training phrase; and (c) determination of the six parameters for each AT. Currently, we decompose the Chinese character into character elements manually, and the selection of element from BCCE set can be solved by using some state-of-the-art OCR technology [eg. 9]. Determination of AT parameters is described in section 3.

3. Determination of the optimal global affine transformation parameters

Using an appropriate affine transform plays an important role for the correct synthesis of a character element (see figure 4). Suppose $\mathrm{E_i}$ from BCCE set is used to synthesize the ith component $\mathrm{R_i}$ of a Chinese character. The synthesized component is given by:

$$\begin{aligned}\widetilde{\mathrm{R}}_{\mathrm{i}} &= \{\widetilde{r}_1, \widetilde{r}_2, \ldots, \widetilde{r}_m\} = \phi(E_i) \\ &= \{\phi(\mathrm{s}_1), \phi(\mathrm{s}_2), \cdots\cdots \phi(\mathrm{s}_{\mathrm{m_i}})\}\end{aligned} \tag{7}$$

In order to determine the optimal parameter of the affine transformation ϕ, a solution is to minimized the mean of nearest-neighbor inter-point distances between $\widetilde{\mathrm{R}}_{\mathrm{i}}$ and $\mathrm{R_i}$ for **A** and ***b*** [8],

$$\{\mathrm{A}, \mathrm{b}\} = \arg\min\{\frac{1}{2}[\frac{1}{m}\sum_i \min_j \left\|\widetilde{r}_i - r_j\right\|^2 + \frac{1}{n}\sum_j \min_i \left\|\widetilde{r}_i - r_j\right\|^2]\} \tag{8}$$

where $\|\bullet\|$ denotes Euclidian norm.

(a) (b) (c)

Fig.4. Different AT produced different synthesis effects. (a). The BCCEs. (b). Results using incorrectly AT. (c). Correct AT results.

Wakahara and Odaka showed [8] that the above optimization problem is equivalent to iteratively minimizing the following object function Ψ:

$$\Psi = \frac{1}{2}\sum_i \sum_j \rho_{ij}(D)\left\|\widetilde{r}_i - r_j\right\|^2 \tag{9}$$

where:

$$\rho_{ij}(D) = \frac{\mu_{ij}(D)}{m} + \frac{v_{ij}(D)}{m}$$

$$\mu_{ij}(D) = \exp\left[-\frac{\left\|\widetilde{r}_i - r_j\right\|^2 - \min_k \left\|\widetilde{r}_i - r_k\right\|^2}{D}\right]$$

$$v_{ij}(D) = \exp\left[-\frac{\left\|\widetilde{r}_i - r_j\right\|^2 - \min_k \left\|\widetilde{r}_k - r_j\right\|^2}{D}\right]$$

$$D = \frac{1}{2}[\frac{1}{m}\sum_i \min_j \left\|\widetilde{r}_i - r_j\right\|^2 + \frac{1}{n}\sum_j \min_i \left\|\widetilde{r}_i - r_j\right\|^2$$

Thus, the minimization of Ψ results in a set of simultaneous linear equations by differentiating Ψ with respect to A and ***b*** as:

$$\frac{\partial \Psi}{\partial A} = 0, \frac{\partial \Psi}{\partial b} = 0 \tag{10}$$

The linear equations of (10) for A and ***b*** can be easily solved by conventional techniques such as Gaussian elimination.

4. Encoding the BCCEs using Cubic-Bezier Curve

The BCCE is indeed a kind of shape, which can be encoded by a variable number of Bezier curve segment [6,10]. Bezier curve is an ideal shape description method which has been widely used in font design and CAD[10]. To encode a BCCE, the contour of BCCE is first extracted, then the contour is divided into several segments at the feature points (cross points or corner points), and finally each segment is encoded with a cubic Bezier curve using LS (Least-Square) method. If the curve fitness cost is large, we split the segment into two small segments and apply the Bezier curve fitting recursively until the fitness cost is small enough. As there are only four parameters need to be stored for a cubic Bezier curve segment, the storage of encoded BCCEs is greatly reduced comparing with storing the original BCCE image or shape. Moreover, by changing the control points of Bezier curve, scaling or rotating of the BCCEs can be performed easily and smoothly.

5. Evaluation of synthesis quality using structure similarity

One simple way to measure the synthesis quality is using the mean square error (MSE). However, as pointed by Z.Wang and A.C. Bovik [7], the MSE is not a suitable measurement for image quality index. In this paper, we used the image quality index defined in [7] as the measurement of the structural similarity between the synthesized character image $\widetilde{SC} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\}$ and the original character image $C = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_N\}$ (here $\mathbf{x}_i$ and $\mathbf{y}_i$ are the ith pixel of the two images respectively, and the size of the synthesized character and the original character has been normalized to the same size of $N \times N$. The structure similarity (SS) between $\widetilde{SC}$ and C is defined as following:

$$SS = \frac{4\sigma_{xy}\overline{\mathbf{x}}\overline{\mathbf{y}}}{(\sigma_x^2 + \sigma_y^2)(\|\overline{\mathbf{x}}\| + \|\overline{\mathbf{y}}\|)} \quad (11)$$

where:

$$\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i, \quad \sigma_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})^2$$

$$\overline{\mathbf{y}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{y}_i, \quad \sigma_y^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{y}_i - \overline{\mathbf{y}})^2$$

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{y}_i - \overline{\mathbf{y}})$$

The dynamic range of SS is [-1,1]. The best value 1 is achieved if and only if $\mathbf{x}_i = \mathbf{y}_i$ for all i=1,2,….N. It can be seen from figure 5 that higher SS values, which mean better synthesis results, are in accord with human perceived judgment.

Fig.5. Corresponding structure similarity measurement of two pairs of synthesized characters.

6. Experimental results

The Chinese Kai font is used in our experiments. By analysis of 3755 Chinese characters in standard GB2312-80 level 1, 490 BCCEs are designed. Some examples of the BCCEs are illustrated in figure 6.

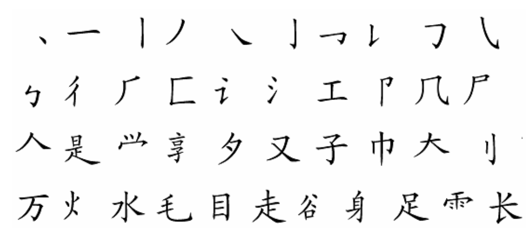
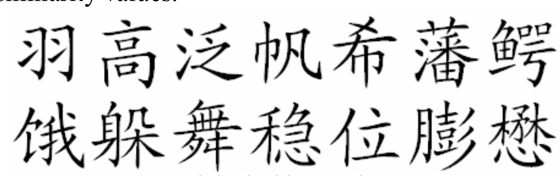


Fig.6. Examples of some BCCEs

Experiment shows that any character in GB2312-80 set1 can be generated using the designed BCCEs and pre-stored AT parameters. Figure 7 gives some of the synthesis characters and their corresponding structure similarity values.



(a). Original Chinese Characters

(b). Synthesized Chinese characters.

Fig. 7. Examples of some synthesized characters and their corresponding synthesis quality values.

From figure 7, it can be seen that the synthesized character looks very similar with the real one. The stroke width and calligraphy style are preserved. The average structure similarity is 0.89.

The storage required to store the 490 Bezier encoded BCCEs (cubic-Bezier parameters) is 124K. With a few additional AT parameters, any Chinese character could be represented. The comparison of storage requirement for storing synthesized characters and directly Bezier encoded characters is shown in figure 8.

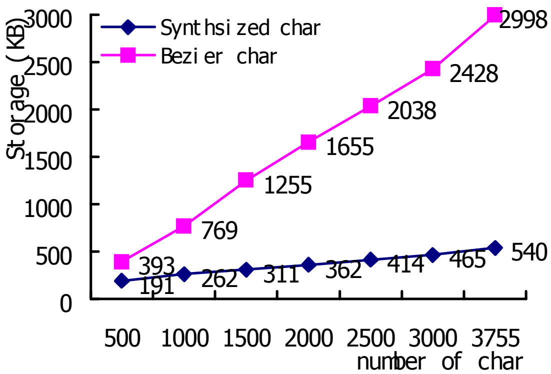


Fig. 8. Storage comparison between synthesized characters and conventional cubic Bezier characters.

It can be seen from figure 8 that by using the proposed synthesis method, the storage for 3755 Chinese character can be reduced from 2998K (directly encoding of 3755 characters using cubic Bezier curves) to 540K(synthesized character from cubic Bezier curve encoded BCCEs with AT). The average storage size of a synthesized Chinese character is only 0.14K. Comparing with that of a conventional encoded Chinese character, which is 0.80K, the storage is reduced by more than 82.5%. It is expected that with the increase of character category number, the average storage for each character can be reduced furthermore, suggesting the proposed Chinese synthesis method has great advantage when processing Chinese characters in embedded device with limited memory.

7. Conclusion

We present a novel Chinese synthesis method based on affine transform of basic Chinese character elements. 490 BCCEs are designed, which can be used to generate any Kai-font Chinese character in GB2312 Set1. The structure similarity measurement is used to evaluate the synthesis quality. Experiments showed that the synthesized characters look smooth and natural. The storage of synthesized characters can be reduced greatly. The proposed Chinese synthesis method has many potential applications, such as building small-size Chinese font, building compact classifier for Chinese OCR, etc.

Currently, the BCCEs are designed manually in a heuristic manner. How to design optimal BCCEs automatically is an interesting research topic for future study. Moreover, to extend our method to other Chinese fonts (such as Song, Fang Song, etc), Chinese calligraphy, Japanese Kanji font, or even handwritten Chinese character is another future research topic.

Acknowledgments

This paper is partially sponsored by the following research foundations: New Century Excellent Talent Program of MOE of China (No.NCET-05-0736) and The University Fund of Microsoft Research Asia (No.FY07-RES-THEME-58)

Reference

[1] Institute of Linguistics of Chinese Academy of Social Sciences, Modern Chinese Dictionary（fifth edition, The Commercial Press，Beijing, 2005.

[2] P.K. Lai, D.Y. Yueng, M.C. Pong, A Heuristic Search Approach to Chinese Glyph Generation Using Hierarchical Character Composition. Computer Processing of Oriental Languages, 10(3): 1996.

[3] Y. Dong and K. Li, "A parametric graphics approach to Chinese font design," in R. Morris and J. Andre (eds.), *Raster Imaging and Digital Typography*, Cambridge University Press, Cambridge, 1991, pp. 156–165.

[4] S. Lim and M. Kim, "Oriental Character Font Design by a Structured Composition of Stroke Elements," Comput.-Aided Design, 27: 193-207, 1995.

[5] Candy L.K. Yiu, Wai Wong, Chinese Character Synthesis using METAPOST, in In proceedings of TUG 2003, TUGboat 24(1), pp. 85-93, 2003.

[6] L. Cinque, S. Levialdi and A. Malizia, "Shape description using cubic polynomial Bezier curves," Pattern Recognition Letters, 19(9): 821-828, 1998.

[7] Z. Wang, A.C. Bovik, A Universal Image Quality Index, IEEE Signal Processing Letters, 9(3):81-83, 2002.

[8] T. Wakahara, K.Odaka, Adaptive normalization of handwritten characters using global/local affine transformation, IEEE Trans. PAMI, 20(12): 1332-1341, 1998.

[9] Q. Huo, Y. Ge, Z.D. Feng, High Performance Chinese OCR Based on Gabor Features, Discriminative Feature Extraction and Model Training, Proc. ICASSP-2001, USA, 2001.

[10] G. Farin, Curves and surfaces for CAGD, Published by Morgan Kaufmann, USA, 2002.