Generating New Chinese Fonts Based on Wavelet Transforms

Jiu-chao Feng[†], Chi K. Tse[#] and Yuhui Qiu[‡]

[†]Department of Electronic & Information Engineering, Southwest China Normal University, Chongqing 400715, China
[#]Department of Electronic & Information Engineering, Hong Kong Polytechnic University, Hong Kong, China
[‡]Department of Computer Science, Southwest China Normal University, Chongqing 400715, China

Abstract

Based on the cubic B-spline curve, new Chinese fonts are generated by wavelet transforms in this paper. The outlines of Chinese fonts are first transformed into B-spline curves. Then, using wavelet transforms, the control points of each curve are decomposed into hierarchies containing the detailed features of the Chinese fonts. Using the synthesis procedure of wavelet transforms, new fonts can be generated by modifying details at selected hierarchies.

Keywords — Multiresolution analysis, B-spline wavelets, hierarchical decomposition, Chinese fonts.

1. Introduction

Being a mathematical tool for hierarchical function decompositions, wavelet transformation allows a function to be described in terms of coarse overall shape, plus features that range from broad to narrow. Regardless of whether the function of interest is an image, curve or surface, wavelets offer an elegant technique for representing the levels of details present [1]. Although wavelets have their roots in approximation theory [2] and signal processing [1], they have been recently applied to solve problems in computer graphics, for example, image editing, image compression, etc.[3]. In this paper we use the cubic B-spline wavelet transform to generate new Chinese fonts. The basic idea is to describe curves by hierarchies and details under a multiresolution analysis (MA). Making use of the synthesis procedure of the B-spline wavelet transforms for some modified and/or combined details, new fonts can be generated.

2. B-spline Wavelets and Their Constructions

Consider a function in some approximation space Γ^n. Let us assume that we have the coefficients of the function in terms of some scaling function basis. We can write these coefficients as a column vector $\mathcal{C}^n = [c_0^n \, c_1^n \, \cdots \, c_{m^n-1}^n]^T$, where n is the level of the transform (as is customarily used

in the wavelet literature) and m defines the size (fineness) of the decomposition [4]. Essentially, these coefficients can be viewed as the control points of a curve in $\mathbf{R}^2$, and for level n, there are m^n control points. A low-resolution version $\mathcal{C}^{n-1}$ of $\mathcal{C}^n$ can be created by using linear filtering and down-sampling on the m^n entries of $\mathcal{C}^n$. This process can be expressed as

$$\mathcal{C}^{n-1} = \mathcal{A}^n \mathcal{C}^n, \tag{1}$$

where $\mathcal{A}^n$ is an $m^{n-1} \times m^n$ matrix of constants. Since $\mathcal{C}^{n-1}$ contains fewer entries than $\mathcal{C}^n$, this filtering process clearly loses some details. For many choices of $\mathcal{A}^n$, it is possible to capture the lost details in another column vector $\mathcal{D}^{n-1}$, which can be computed by

$$\mathcal{D}^{n-1} = \mathcal{B}^n \mathcal{C}^n, \tag{2}$$

where $\mathcal{B}^n$ is an $m^{n-1} \times m^n$ matrix of constants related to $\mathcal{A}^n$. The pair of matrices $\mathcal{A}^n$ and $\mathcal{B}^n$ are called *analysis filters*. If $\mathcal{A}^n$ and $\mathcal{B}^n$ are chosen appropriately, then the original $\mathcal{C}^n$ can be recovered from $\mathcal{C}^{n-1}$ and $\mathcal{D}^{n-1}$ by using the matrices $\mathcal{P}^n$ and $\mathcal{Q}^n$, i.e.,

$$\mathcal{C}^n = \mathcal{P}^n \mathcal{C}^{n-1} + \mathcal{Q}^n \mathcal{D}^{n-1}. \tag{3}$$

Recovering $\mathcal{C}^n$ from $\mathcal{C}^{n-1}$ and $\mathcal{D}^{n-1}$ is called *synthesis* or *reconstruction*. In this context, $\mathcal{P}^n$ and $\mathcal{Q}^n$ are called *synthesis filters*. Note that the procedure for splitting $\mathcal{C}^n$ into a "low-resolution part" $\mathcal{C}^{n-1}$ and a "detailed part" $\mathcal{D}^{n-1}$ can be applied recursively to the low-resolution version $\mathcal{C}^{n-1}$. Thus, the original coefficients can be expressed as a hierarchy of lower resolution versions $\mathcal{C}^0$, $\mathcal{C}^1 \cdots$, $\mathcal{C}^{n-1}$ and details $\mathcal{D}^0$, $\mathcal{D}^1 \cdots$, $\mathcal{D}^{n-1}$. The decomposition and synthesis processes are shown in Fig. 1.

Consider a curve which can be represented by

$$F^n(u) = \Omega^n(u)\mathcal{C}^n, \tag{4}$$

where $u \in [0, 1]$, and $\Omega^n(u)$ is a scaling function row matrix of level n, consisting of $(\omega_0^n(u), \, \omega_1^n(u), \, \cdots, \, \omega_{m^n-1}^n(u))$. In this paper, $\omega_j^n(u)$ is the B-spline basis function and $F^n(u)$ is the spline curve. Let Γ^j be the vector space spanned by $(\omega_0^j(u), \, \omega_1^j(u), \, \cdots, \, \omega_{m^j-1}^j(u))$, where j is an integer ($j \in$
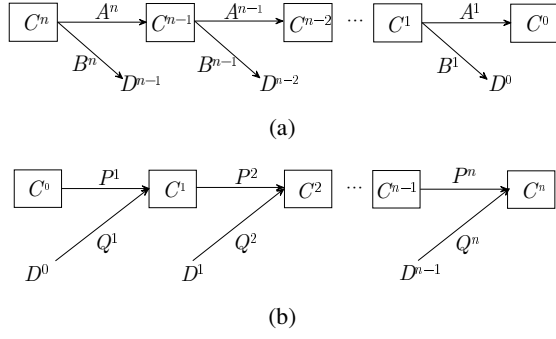
Fig. 1: Wavelet transformation. (a) Decomposition process; (b) synthesis process.

$[0, n]$). It should be noted that $\mathcal{P}^j$ satisfies

$$\Omega^{j-1}(u) = \Omega^j(u)\mathcal{P}^j. \tag{5}$$

Equation (5) indicates that each scaling function at level $j - 1$ must be expressible as a linear combination of the finer scaling functions at level j, and that the subspace Γ^j is nested by $\Gamma^0 \subset \Gamma^1 \cdots \subset \Gamma^n$. Since the wavelet space Σ^{j-1} is the orthogonal complement of Γ^{j-1} in Γ^j, we can write the wavelets $\Psi^{j-1}(u)$ as linear combinations of the scaling functions $\Omega^j(u)$, i.e.,

$$\Psi^{j-1}(u) = \Omega^j(u)\mathcal{Q}^j(u) \tag{6}$$

where $\Psi^{j-1}(u) = (\psi_0^{j-1}(u), \cdots, \psi_{m^{j-1}-1}^{j-1}(u))$. Note that Eqs. (5) and (6) can be expressed, by using block-matrix notation, as

$$[\Omega^{j-1}|\Psi^{j-1}] = \Omega^j[\mathcal{P}^j|\mathcal{Q}^j]. \tag{7}$$

Combining (5), (6) and (7), we have

$$\left[\frac{\mathcal{A}^j}{\mathcal{B}^j}\right] = [\mathcal{P}^j|\mathcal{Q}^j]^{-1}. \tag{8}$$

The MA framework presented above is very general. In practice, one often has the freedom to design an MA which is specifically suited to a particular application. Here, for the purpose of this paper, we summarize the construction of B-spline wavelets as follows:

- Select the endpoint interpolating cubic B-spline basis functions as the scaling functions. For any j ($j \in [0, n]$), construct the $\mathcal{P}^j$ in Γ^j.

- Determine the subspace Ψ^{j-1} in the light of the inner product formula $\langle \omega_k^{j-1}|\psi_l^{j-1}\rangle = 0$ (for all k and l). Determine $\mathcal{Q}^j$ by using

$$[\langle\Omega^{j-1}|\Omega^j\rangle]\mathcal{Q}^j = 0. \tag{9}$$

- Compute the left-hand part of Eq. (8) by using the computed $\mathcal{P}^j$ and $\mathcal{Q}^j$.

3. Synthesis of Chinese Fonts

A single Chinese font consists of many closed outlines, each of which can be represented by a B-spline curve. Therefore, a Chinese font can be described hierarchically by decomposition and synthesis of wavelet transforms for the selected B-spline curves.

3.1. Control Points of the B-spline Curve

A B-spline curve is described by a set of control points. To represent a Chinese font by B-spline curves, we need to decompose the control points hierarchically and in an exponential fashion. Here, the number of sections of a B-spline curve required to realize this operation for the jth level (layer) is 2^j. Hence, the number of the control points is also 2^j, due to the closed curve.

In our implementation, to achieve uniformly spaced B-spline, we choose $k = 2^j + d - 1$ ($d = 4$) to produce 2^j equally spaced interior intervals, thus giving $2^j + d$ B-spline basis functions for degree d and level j. Specifically, in this paper, we re-sample the B-spline curve in order to satisfy the requirements for the sections. The exact procedures for extending control points are summarized as follows:

- Given N points, compute a B-spline curve $F^n(u) = (f_1^n(u), f_2^n(u), \cdots, f_N^n(u))$, where $f_i^n(u)$ is the ith section B-spline curve expressed as

$$f_i^n(u) = \sum_{j=1}^{4} c_{i+j-2}^n \mathcal{N}_{j,4}(u), \tag{10}$$

where $i = 1, 2, \cdots, N$, $u \in [0, 1]$, and $\mathcal{N}_{j,4}(u)$ is the cubic B-spline basis function.

- According to the actual length of the outline, determine the number of levels (layers) for decomposing, which usually requires that $N \leq L$, $L = 2^n + 3$.

- Re-sample $F^n(u)$ with an interval of $\Delta t = \frac{N}{2^n - 1}$, and the number of points is accordingly assigned as $2^n - 1$.

- Determine the new control point c_{i-1}^n by (10).

3.2. Wavelet Transform of Fonts: Examples

Fig. 2 shows the transformed results of four different fonts for the Chinese character "计", namely, 'Song Ti', 'Fang Song Ti', 'Kai Shu' and 'Hei Ti', each of which is decomposed into seven layers. The left most is the original, the second from left is the layer 1 image which has removed $\mathcal{D}^{n-1}$, the third from left is the layer 2 image which has removed $\mathcal{D}^{n-1}$ and $\mathcal{D}^{n-2}$, the fourth from left is the layer 3 image which has removed $\mathcal{D}^{n-1}$, $\mathcal{D}^{n-2}$ and $\mathcal{D}^{n-3}$, etc. As we can see, progressing from left to right in Fig. 2, more details are
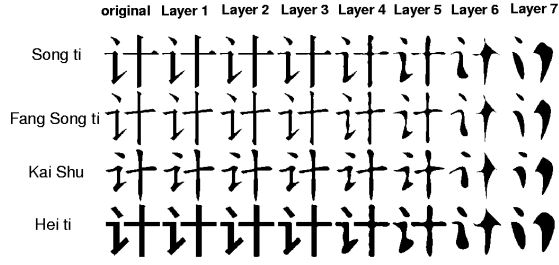
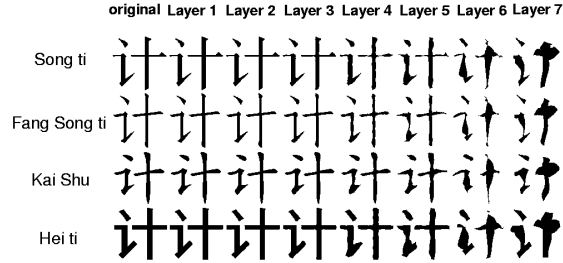Fig. 2: Wavelet transforms when multiple details are removed.

Fig. 3: Wavelet transforms when details of one layer are removed.

Fig. 4: New fonts generated by combining two fonts.

Fig. 5: New fonts generated based on multiple fonts.

lost. Finally, at layer 7, the transforms of the four fonts converge to the same basic image.

Fig. 3 shows the results when only one layer is removed, i.e., omitting $\mathcal{D}^{n-1}$ for layer 1, omitting $\mathcal{D}^{n-2}$ for layer 2, etc. We can see from Fig. 3 that, from layer 1 to layer 7, the appearance of a font is significantly different.

3.3. Algorithm for Generating New Fonts

From the above simulation results, we can see that the B-spline wavelet transform can decompose the outlines of Chinese fonts into different hierarchies and details, which represent some specific features of Chinese characters. In the following, we modify the details on some layers, and then synthesize a new Chinese font.

Two methods can be used to alter the font details. One is to add or reduce details on some selected layers. As shown in Fig. 2, by removing the details of the first four layers, and/or adding new features, we can synthesize a new font. Another method is to combine the details of two or more fonts to produce new fonts. The algorithm for generating new fonts in this paper is summarized as follows:

1. Decompose a Chinese character into different closed areas, from which edges can be extracted and outlines can be tracked.

2. For each outline, repeat steps 3 to 7.

3. Express the outline as B-spline curves with $2^n + 3$ control points.

4. By using B-spline wavelet transform, decompose the control points $\mathcal{C}^n$ into details $\mathcal{D}^{n-1}$, $\mathcal{D}^{n-2}$, $\cdots$, $\mathcal{D}^1$, and the control point $\mathcal{C}^0$.

5. Let $\check{\mathcal{D}}^j$ be $s_1 \mathcal{D}_1^j + s_2 \mathcal{D}_2^j + s_3 \mathcal{D}_3^j + s_4 \mathcal{D}_4^j$, which stands for the combined details of the newly generated outline (which is generated by combining different fonts with the same topological structure), where $0 \leq s_i \leq 1$, $\mathcal{D}_i^j$ $(i = 1, 2, 3, 4)$ stands for the details at layer j, and $\check{\mathcal{D}}^j$ for the new details at layer j.

6. By using the new details $\check{\mathcal{D}}^j$ together with the original details and the lower layer control point, generate new control points after the synthesis procedure.

7. Re-sample the newly generated curve $\check{F}^n$, which consists of $\check{\mathcal{C}}^n$.

8. Fill the connecting area consisting of the new outline, and combine the connecting area in order to obtain new fonts.

4. Results

The above algorithm is used to generate new Chinese fonts. In our simulations, a 200×200 Chinese character array, "计算机", as shown in Figs. 4 and 5, is used to evaluate the generating algorithm.

Fig. 4 shows the synthesis result when two fonts 'Song Ti' and 'Hei Ti' are used as the seed fonts.

- New fonts (A) and (D) are generated by removing the details in layers 1 to 4;

- New fonts (B) and (C) are formed by adding details $\breve{\mathcal{D}}^j$, where the coefficients $s_1 = 0.5$ and $s_2 = 0.5$.

Fig. 5 shows the synthesis results when four fonts 'Song Ti', 'Fang Song Ti', 'Kai Shu' and 'Hei Ti' are used as the seed fonts.

- New font (A) is generated by combining 'Fang Song Ti', 'Kai Shu' and 'Song Ti', with the coefficients $s_1 = 0.2$, $s_2 = 0.4$, $s_3 = 0.4$ and the lowest layer control point $\mathcal{C}^{n-4}$ from 'Fang Song Ti'.

- New font (B) is generated by combining 'Fang Song Ti', 'Kai Shu', 'Song Ti' and 'Hei Ti' with the coefficients $s_1 = 0.1$, $s_2 = 0.3$, $s_3 = 0.3$, $s_4 = 0.3$ and the lowest layer control point $\mathcal{C}^{n-4}$ from 'Fang Song Ti'.

- New font (C) is generated by combining 'Kai Shu', 'Fang Song Ti', 'Song Ti' and 'Hei Ti', with the coefficients $s_1 = 0.1$, $s_2 = 0.3$, $s_3 = 0.3$, $s_4 = 0.3$ and the lowest layer control point $\mathcal{C}^{n-4}$ from 'Kai Shu'.

- New font (D) is generated by combining 'Fang Song Ti', 'Kai Shu' and 'Song Ti', with the coefficients $s_1 = 0.2$, $s_2 = 0.4$, $s_3 = 0.4$ and the lowest layer control point $\mathcal{C}^{n-4}$ from 'Kai Shu'.

From these examples, we can see that there exist localized detailed differences between the new fonts and the original fonts, and the basic structure of the new fonts are mainly determined by the lower layer's control points.

5. Conclusion

A synthesis method of new Chinese fonts by using wavelet transforms has been described in this paper. The basic idea is to decompose a font into layers of differing details. Then, by deleting, incorporating and altering details of selected layers, new fonts can be generated. In the process of generating new fonts, details from several existing fonts can be used to produce hybrid fonts. Essentially, the process involves transforming the outlines of a Chinese fonts into the B-spline curves, and decomposing the control points of each curve into the hierarchies and the detailed features. Once the decomposition is completed, re-ordering from one or more fonts can be performed to generate new fonts. The method can be extended to other image processing applications for creating new images from existing images.

References

[1] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 674–693, 1989.

[2] I. Daubechies, "Orthonormal bases of compactly supported wavelets", *Communication on Pure and Applied Mathematics*, Vol. 41, pp. 909–996, 1988.

[3] S. G. Mallat, *A Wavelet Tour of Signal Processing,* San Diego: Academic Press, 1998.

[4] C. K. Chui, *An Introduction to Wavelets,* San Diego: Academic Press, 1992.