MATEC Web of Conferences 25, 03014 (2015)
DOI: 10.1051/matecconf/20152503014

Application and Realization of the Computer Animation Design Based on Improved Cubic B-spline Curves

Na Ni
Department of Animation, College of Arts Science Branch, Henan University, Kaifeng, Henan, China

ABSTRACT: Based on the application of the cubic B-spline curves in the computer animation design, taking into account the security and confidentiality of the information, this paper improves the animation design techniques by the use of the improved cubic B-spline curves. Finally, this paper provides the relevant C language programs of the animation design.

Keywords: cubic B-spline curve; computer; animation design; C language

1 INTRODUCTION

With the in-depth research of the computer technology and the theory of computer graphics, the computer animation design is characterized with high intelligence, high level and strong artistic quality, which has entered into people's daily life. China's computer animation design also has reached a certain level and height with its gradual increase, but there is still a room for improvement in terms of the theoretical research, design method, animation effect and so on. It presents a higher and harder challenge for the personnel who are engaged in it. Nowadays, the computer animation is widely used for game production, assisting education, commercial advertising, special technology making and other categories. The computer animation design has been promoted to the category of design art. As an art, it is determined by people's subjective opinion, which not just adopts one kind of creation method. However, for China, the animation is an industry which requires a large batch of production. Therefore, it requires advanced and superb technology as a basis, and the computer animation design is an important one. At present, the researches of the relevant theory, technology, algorithm and model are very important research hotspots [1].

In the computer animation design, the shape design is an important factor. At the beginning of the design, people always use spline curves for interpolation. In this way, the curves shall pass through all known points. However, the common method of curve design first uses the broken line to draw up a rough outline, and then uses the curves obtained by the fitting method to approximately replace the outline of this broken line. There is no need for the fitted curves to pass through all known points, so it is easy to accomplish in reality without distortion. In the early 1970s, R. F. Riesenfeld, W. J. Gorden and other experts from the United States proposed the B-spline curve, which has a geometrical invariability, continuity, intuition, local modification, variation reduction, convexity- preserving property and other good properties. Therefore, in the computer animation design, the B-spline curve is widely used, especially the twice B-spline curves and cubic B-spline curves [2]. On this basis, taking into account the security and confidentiality of the information, this paper carries out the computer animation design by the use of the improved cubic B-spline curves, and also gives some relevant C language programs and provides practical methods.

2 CUBIC B-SPLINE CURVE

The position vector of (n+1) vertices in the space constructs (n−2) segment of (k = 0, 1, 2, 3; fourth-order n = 3) B-spline curve segments, and every adjacent four points can define a curve segment $P_i(u)\ (i=1,2,\cdots,n-2; 0\le u\le 1)$. Its defined expression is as follows [3]:

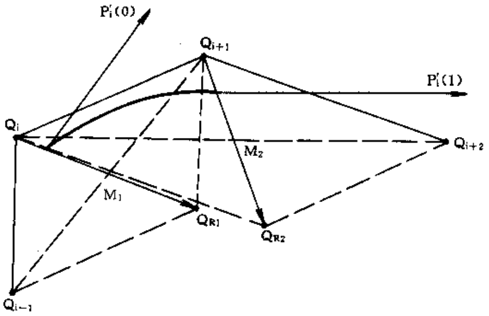
$$P_i(u)=\frac{1}{6}\begin{bmatrix}u^3 & u^2 & u & 1\end{bmatrix}\begin{bmatrix}-1 & 3 & -3 & 1\\ 3 & -6 & 3 & 0\\ -3 & 0 & 3 & 0\\ 1 & 4 & 1 & 0\end{bmatrix}\begin{bmatrix}\mathbf{P}_{i-1}\\ \mathbf{P}_i\\ \mathbf{P}_{i+1}\\ \mathbf{P}_{i+2}\end{bmatrix}$$

$$=\frac{1}{6}(1-u)^3P_{i-1}+\frac{1}{6}(3u^3-6u^2+4)P_i+\frac{1}{6}(-3u^3+3u^2+3u+1)P_{i+1}+\frac{1}{6}u^3P_{i+2}$$

$$=N_{0,3}(u)P_{i-1}+N_{1,3}(u)P_i+N_{2,3}(u)P_{i+1}+N_{3,3}(u)P_{i+2}$$

The position vector of the endpoint: $P_i(0)=\frac{1}{6}(\mathrm{P}_{i-1}+4\mathrm{P}_i+\mathrm{P}_{i+1})$ and $P_i(1)=\frac{1}{6}(\mathrm{P}_i+4\mathrm{P}_{i+1}+\mathrm{P}_{i+2})$. That is, the starting point is in one third of the central line P_iM_1 of the triangle $\mathrm{P}_{i-1}\mathrm{P}_i\mathrm{P}_{i+1}$, while the end point is in one third of the central line $\mathrm{P}_{i+1}M_2$ of the triangle $\mathrm{P}_i\mathrm{P}_{i+1}P_{i+2}$. Thus the endpoint of the

B-spline curve does not pass through the control points.



First-order derivative vector of the endpoint: $P_i'(0)=\frac{P_{i+1}-P_{i-1}}{2}$ and $P_i'(1)=\frac{P_{i+2}-P_i}{2}=P_{i+1}'(0)$. That is, the tangent vector of the starting point of the curve is parallel to the base $P_{i-1}P_{i+1}$ of $P_{i-1}P_iP_{i+1}$, and its die length is one half of the base length $P_{i-1}P_{i+1}$. Similarly, the tangent vector of the end point of the curve is parallel to the base P_iP_{i+2} of $P_iP_{i+1}P_{i+2}$, and its die length is one half of the base length P_iP_{i+2}. Moreover, two adjacent curve segments have the first-order derivative continuity ($P_i'(1)=P_{i+1}'(0)$).

Second derivative vector: $P_i''(0)=P_{i-1}-2P_i+P_{i+1}$ and $P_i''(1)=P_i-2P_{i+1}+P_{i+2}=P_{i+1}''(0)$. That is, the second derivative vector of the endpoint at the curve segment is equal to diagonal of the parallelogram formed by two adjacent straight lines, and the nodes at two curve segments have the second derivative continuity ($P_i''(1)=P_i''(0)$).

The cubic B-spline curve has good properties, so it is widely used in the computer animation design.

Locality: According to the defined equation, n-th B-spline curve at the k segment is only related to (n+1) vertices $P_i\ (i=0,1,\cdots,n)$, so the modification of one control endpoint only has an impact on (n+1) adjacent segments rather than the entire curve (when $m>n$). It brings great convenience to the modification of local shape in the design of the curve.

Geometrical invariability: B-spline curve expressed by the defined equation is a parametric form, so it is the same with Bezier curve. The shape and position of the B-spline curve is independent in the selection of coordinate system.

Continuity: When the given (m+n+1) control vertices $P_i\ (i=0,1,\cdots,m+n)$ which are of no multiplicity, then the entirely controlled B-spline curve has n-1-order geometric continuity (G^{n-1}). When the given maximum number of vertices which is adjacent to the control vertices is h (that is, h control vertices coincide together), then the entire B-spline curve has n-h-1-order geometric continuity (G^{n-h-1}).

Symmetry: The symmetry of primary function can be deduced according to the B-spline curve.

$$P_{k,n}(1-t)=\sum_{i=0}^{n}P_{i+k}G_{i,n}(1-t)$$
$$=\sum_{i=0}^{n}P_{i+k}G_{n-i,n}(t) \qquad (t\in[0,1])$$

It shows that, the geometrical properties of the starting point and end point of the B-spline curve segments are identical.

Recurrence: expression form of the recurrence curve at n-th B-spline curve segment:

$$P_{k,l}^{i}(t)=\begin{cases}P_{i+k} & l=0\\ \lambda_{i,l}(t)P_{k,l-1}^{i}(t)+\mu_{i,l}(t)P_{k,l-1}^{i+1}(t), & l=1,2,...,n,\end{cases}$$
$$k=0,1,...,m$$

Where:

$$\lambda_{i,l}(t)=\frac{1}{n-l+1}(i+1-t);$$

$$\mu_{i,l}(t)=\frac{1}{n-l+1}(n-l-i+t); \qquad t\in[0,1];$$

$i=0,...,n-l;\ l=1,2,...,n$.

Convexity-preserving property: B-spline curve is the same with Bezier curve which also has the convexity-preserving property. That is, when all the control vertices form a closed planar convex polygon, $P_{k,n}(t)$ is a planar convex curve.

Convex hull property: When $t\in(0,1)$, there is $0\le G_{i,n}(t)\le 1\ (i=0,1,...,n)$ and $\sum_{i=0}^{n}G_{i,n}(t)\equiv 1$. Therefore, according to the convex hull definition, $t\in(0,1)$ and $P_{k,n}(t)$ must be in the convex hull constructed by the control vertices.

3 IMPROVED B-SPLINE CURVE

If F and G are coefficient matrixes of two primary functions of the free curve, α is an arbitrary real number of $[0,1]$, then $S=\alpha F+(1-\alpha)G$ is a mixing matrix of F and G.

$\alpha\to 1$, S approaches F; $\alpha\to 0$, S approaches G. This method can mix two coefficient matrixes, thus realizing the curve encryption. Meanwhile, the coefficient matrix of the primary function of the curve after encryption is

$$G = \frac{S - \alpha F}{1 - \alpha}$$

Next, this paper uses this method to mix the coefficient matrixes of the primary function of the cubic B-spline curve and the primary function of the cubic Bézier curve, thus realizing the improvement in the cubic B-spline curves.

The primary function of the Bézier curve is also the primary function of Bernstein. For example, n -th primary function is as follows:

$$B_{i,n}(u) = C_n^i (1-u)^{n-i} u^i \text{ , } u \in [0,1] \text{ , } i = 0,1,\cdots,n$$

The primary function of the cubic Bézier curve is as follows:

$$B_{0,3}(u) = C_3^0 (1-u)^3 u^0 = (1-u)^3$$

$$B_{1,3}(u) = C_3^1 (1-u)^2 u^1$$

$$B_{2,3}(u) = C_3^2 (1-u)^1 u^2$$

$$B_{3,3}(u) = C_3^3 (1-u)^0 u^3 = u^3$$

Its coefficient matrix F is as follows:

$$F = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$T_{n,k} = \{u_i\}_{i=0}^{n+k}$ is the partition of the parameter u coordinate axis, so the primary function of the k-th B-spline of $T_{n,k}$ is $B_{i,k}(u)$.

$$\begin{cases} B_{i,k}(u) = \begin{cases} 1, t_i \le u \le t_{i+1} \\ 0, other \end{cases} \\ B_{i,k}(u) = \dfrac{u - u_i}{u_{i+k} - u_i} B_{i,k-1}(u) + \dfrac{u_{i+k+1} - u}{u_{i+k+1} - u_{i+1}} B_{i+1,k-1}(u) \end{cases}$$

The primary function of the cubic B-spline curve is as follows:

$$B_{0,3}(u) = \frac{1}{6} C_3^0 (1-u)^3 u^0 = \frac{1}{6}(1-u)^3$$

$$B_{1,3}(u) = \frac{1}{6} C_3^1 (1-u)^2 u^1$$

$$B_{2,3}(u) = \frac{1}{6} C_3^2 (1-u)^1 u^2$$

$$B_{3,3}(u) = \frac{1}{6} C_3^3 (1-u)^0 u^3 = \frac{1}{6} u^3$$

Its coefficient matrix G is as follows:

$$G = \frac{1}{6} F = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

The mixing matrix S is as follows:

$$S = \alpha F + (1-\alpha) G = \alpha F + \frac{1}{6}(1-\alpha) F$$

$$= \frac{1}{6} \begin{bmatrix} -5\alpha - 1 & 15\alpha + 3 & -15\alpha - 3 & 15\alpha + 1 \\ 15\alpha + 3 & -30\alpha - 6 & 15\alpha + 3 & 0 \\ -15\alpha - 3 & 18\alpha & -3\alpha + 3 & 0 \\ 5\alpha + 1 & -4\alpha + 4 & -\alpha + 1 & 0 \end{bmatrix}$$

Then, when S is the coefficient matrix, the primary function of the spline is as follows:

$$B_{0,3}(u) = \frac{1}{6}\left[(-5\alpha - 1)u^3 + (15\alpha + 3)u^2 + (-15\alpha - 3)u + (5\alpha + 1)^3\right]$$

$$B_{1,3}(u) = \frac{1}{6}\left[(15\alpha + 3)u^3 + (-30\alpha - 6)u^2 + 18\alpha u + (4\alpha - 4)\right]$$

$$B_{2,3}(u) = \frac{1}{6}\left[(-15\alpha - 3)u^3 + (15\alpha + 3)u^2 + (-3\alpha + 3)u + (-\alpha + 1)\right]$$

$$B_{3,3}(u) = \frac{1}{6}\left[(15\alpha + 1)\right]u^3$$

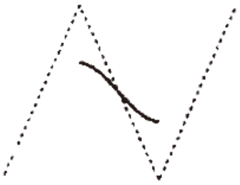


Figure 1. Cubic B-spline curve

Next, this paper carries out example verification for this method. For example, for $Q_0(1,2)$, $Q_1(4,3)$, $Q_3(1,4)$ and $Q_4(4,5)$, we can obtain a B-spline curve [4, 5]. Then, we can use the improved cubic B-spline curve to compare with two results, as shown in Figures 1-3.

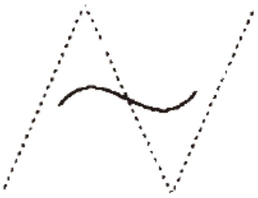


Figure 2. Improved cubic B-spline curve when $\alpha = 0.4$

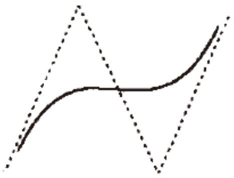
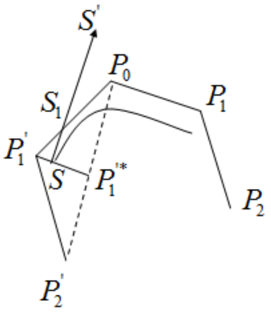


Figure 3. Improved cubic B-spline curve when $\alpha = 0.9$

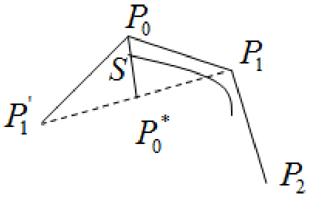
4 APPLICATION OF THE IMPROVED CUBIC B-SPLINE CURVE IN THE COMPUTER ANIMATION DESIGN

Typically, the curve segment will not pass through all the points, but it will have some problems in reality. This paper uses the improved cubic B-spline curve to deal with these problems.

The following figure shows that S is at the starting point of the curve, and $SS^{'}$ is its tangent vector:



Then add points $P_1^{'}$ and $P_2^{'}$, and meet $SS_1 = \frac{SS^{'}}{3}$, $P_1^{'}S_1 = \frac{P_0S_1}{2}$, $SP_1^{'*} = 2P_1^{'}S$ and $P_0P_1^{'*} = P_1^{'}P_2^{'}$. If we do not emphasize the tangent vector, we can make the changes as follows, and meet $SP_0^{*} = 2P_0S$ and $P_1P_0^{*} = P_1^{'}P_0^{*}$.



Similarly, if S is at the end point of the curve, the mode of processing is the same.

5 REALIZATION OF C LANGUAGE

There are many functions of processing graphics in C language, which build a good platform for the computer animation design. In this platform, this paper researches single-page and multi-page scenario in the computer animation design.

For single-page scenario, the process is to quickly copy all object images in the animation that are stored in the storage area to the screen from the memory for display, and then restore some background images that are covered by the foregrounds according to the background pixels of the images and XOR operation between pixels, and ultimately realize the effect of animation. This process can be finished through the putimage library function in C language.

For multi-page scenario, the display page is the first page in multiple pages, while the drawing page is another page in multiple pages, and then they are a rotation process. The basic principle is as follows: If the first opening page is the display page, the other page will be the drawing page; when a new page is opened, two pages are changed and the display page is turned into the drawing page. Under normal conditions, the first page is used to display the singular picture of the animation process, then the even picture will be displayed on the second page, and then taking turns. This process can be finished through the function of libraries setvisualpage() and setactivepage()in C language.

6 CONCLUSION

In this computer animation design, the cubic B-spline curves are widely used. After adding the factors of the security and confidentiality of the information, this paper applies the improved cubic B-spline curves to the computer animation design, and provides the comparative results and application programs. Finally, this paper improves some the animation design programs via the powerful C language platform.

REFERENCES

[1] Zhao Yunpeng. 2014. China's computer animation design and development. *Management & Technology of Small and Medium-sized Enterprise*, 4: 157-158.

[2] Wan Jing. 1993. Cubic B-spline curve and its application in computer-aided animation design. *Micro Computer Information*, 2: 22-25.

[3] Liu Rui, & Zhong Lanfen. 2013. Generation of flowers by the cubic B-spline curve modeling technique. *Yinshan Academic Journal*, 27 (1): 13-54.

[4] Yuan Jing, Du Na, Dong Wenzhong, & Wu Lijuan. 2015. Design and implementation of B-spline surface reconstructing algorithm of the scattered data points. *Journal of Shenyang Normal University (Natural Science)*, 33 (2): 253-256.

[5] Gu Yuping, & Tang Yuehong. 2015. B-spline surface reconstruction of arbitrary scattered points. *Computer Application Research*, 32 (2): 635-637.