

一种基于 Bézier 插值曲面的图像放大方法*

孙庆杰 张晓鹏 吴恩华

(中国科学院软件研究所计算机科学开放研究实验室 北京 100080)

摘要 文章提出了一种利用 Bézier 插值曲面进行图像放大的方法,该方法是为数字图像的每一个色彩分量构造一个分块双三次 Bézier 插值 C^1 曲面,图像放大等价于以不同的采样速率对该曲面进行采样的过程.实验结果表明,该方法可以大大改善放大图像的效果.

关键词 图像放大, Bézier 曲面, 插值, 加速, 像素子分.

中图分类号 TP391

在对图像进行处理时,常常要对图像进行放大.一般的方法是使用简单的比例变换,但这种方法时常会引起比较严重的图像走样,使图像产生大量锯齿.文献[1]提出了一种利用 B-样条插值的图像放大方法,使放大后的图像质量得到极大的改善,但该方法的缺点是速度慢.如一幅大小为 640×512 的 24bit 的彩色图像,若使用文献[1]中的方法将该图像的某一部分放大为 640×512 大小,所需时间为 30s(二阶 B-样条, APOLLO DN-590)和 60s(三阶 B-样条, APOLLO DN-590).

近几年来,基于图像的虚拟现实^[2~4]的研究与系统开发已经引起了人们越来越多的注意.我们开发了一个基于图像的虚拟环境漫游系统 VEWALK.在该系统中,当用户漫游时,可以对场景渐进观察.为了实现这种渐进观察,我们需要一种既快速又能保证质量的图像放大方法.本文利用分块双三次 Bézier 插值 C^1 曲面的连续性和局部性,提出了一种图像放大方法,很好地满足了 VEWALK 的要求.

本文第 1 节介绍由数字图像构造 Bézier 插值曲面的方法.第 2 节介绍利用 Bézier 插值曲面放大图像的过程.第 3 节提出一种加速方法.最后给出实验结果与结论.

1 Bézier 曲面及其插值方法

1.1 Bézier 曲面

Bézier 曲面是由一组控制顶点生成的曲面.它可以用于在一组数据点之间进行插值,以便获得平滑过渡.本文将 Bézier 曲面应用于颜色分量的插值.

Bézier 曲面的定义如下:

在空间给定 $(n+1) \times (m+1)$ 个点 $P_{i,j}$ ($i=0,1,\dots,n; j=0,1,\dots,m$), 称下列张量积形式的参数曲面为 $n \times m$ 次的 Bézier 曲面:^[5]

$$P(u,v) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} B_i^n(u) B_j^m(v) \quad (u,v \in [0,1]), \quad (1)$$
$$B_i^n(t) = C_m^i t^i (1-t)^{m-i},$$

其中 $P_{i,j}$ 是 $P(u,v)$ 的控制顶点.由两组多边形 $P_{i,0}P_{i,1}\dots P_{i,m}$ ($i=0,1,\dots,n$) 和 $P_{0,j}P_{1,j}\dots P_{n,j}$ ($j=0,1,\dots,m$) 组成的网称为 $P(u,v)$ 的控制网格,记为 $\{P_{i,j}\}$.可以认为,控制网格 $\{P_{i,j}\}$ 是 $P(u,v)$ 的大致形状的勾画, $P(u,v)$ 是

* 本文研究得到国家自然科学基金和国家 863 高科技项目基金资助.作者孙庆杰,1972 年生,硕士,主要研究领域为虚拟现实.张晓鹏,1963 年生,讲师,主要研究领域为计算机图形学,计算几何.吴恩华,1947 年生,博士,研究员,博士生导师,主要研究领域为计算机图形学.

本文通讯联系人:孙庆杰,北京 100080,中国科学院软件研究所计算机科学开放研究实验室

本文 1998-04-27 收到原稿,1998-07-14 收到修改稿

对 $\{P_{i,j}\}$ 的逼近.

1.2 Bézier 曲面的拼接

定义. 当两块曲面相连接时,若经过其公共边界上的任一点处在任一方向上的两张曲面的导数相等,我们称这两块曲面达到 C^1 连续.

假设 $P(u,v)$ 是由公式(1)定义的一个 $n \times m$ 次的 Bézier 曲面, $Q(u,v)$ 是一个 $k \times m$ 次的 Bézier 曲面,

$$Q(u,v) = \sum_{i=0}^k \sum_{j=0}^m Q_{i,j} B_i^k(u) B_j^m(v) \quad (u,v \in [0,1]). \tag{2}$$

$P(u,v)$ 与 $Q(u,v)$ 达到 C^1 连续的充分必要条件是^[5]:

$$\begin{aligned} P_{n,j} &= Q_{0,j} & j &= 0, 1, 2, \dots, m, \\ P_{n-1,j} P_{n,j} &= Q_{0,j} Q_{1,j} & j &= 0, 1, 2, \dots, m. \end{aligned}$$

在第 1.3 节中,我们将根据这个条件构造一个 Bézier 曲面阵列,它们在整体上达到 C^1 连续.

1.3 由数字图像构造 Bézier 插值曲面

假设 $I(x,y)$ 是一个 $m \times n$ 的数字图像,它提供了图像色彩的 3 个离散的信息阵列,每个色彩分量对应于一个阵列.插值方法的价值在于,将有限的离散信息扩展为一个连续的信息系统,或者说,为离散信息补充了各种中间信息.

设 $R_{i,j} (1 \leq i \leq m, 1 \leq j \leq n)$ 是 $I(x,y)$ 中第 i 行第 j 列像素的红色分量,它与像素平面上的二维点 (i,j) 相对应.我们的任务是构造一个二元函数 $F(u,v) (0 \leq u \leq m, 0 \leq v \leq n)$,使 $F(i,j) = R_{i,j}$,并且 $F(u,v)$ 在任一点处达到 C^1 连续.

我们的方法是:首先,将矩阵 $R_1 = \{R_{i,j} | 1 \leq i \leq m, 1 \leq j \leq n\}$ 扩充为 $R_2 = \{R_{i,j} | 1 \leq i \leq m+1, 1 \leq j \leq n+1\}$,也就是说,续上 $R_{i,j} (i=m+1 \text{ 或 } j=n+1)$ 的值,我们用向外插值法,取 $R_{m+1,j} = 2R_{m,j} - R_{m-1,j} (1 \leq j \leq n), R_{i,n+1} = 2R_{i,n} - R_{i,n-1} (1 \leq i \leq m), R_{m+1,n+1} = R_{m+1,n} + R_{m,n+1} - R_{m,n}$,然后,为了下一步更加容易地构造 Bézier 曲面,我们再将 R_2 扩充为 $R_3 = \{R_{i,j} | 0 \leq i \leq m+2, 0 \leq j \leq n+2\}$,其扩充方法与 R_2 的方法相同.最后构造 $m \times n$ 个 3×3 的 Bézier 曲面 $S_{i,j}$,

$$S_{i,j}(x,y) = \sum_{\alpha=0}^3 \sum_{\beta=0}^3 Q_{\alpha,\beta}^{i,j} B_{\alpha}^3(x) B_{\beta}^3(y) \quad (x,y \in [0,1]), \tag{3}$$

每一个 $S_{i,j}$ 的 4 个角控制顶点取为对应的 $R_{i,j}$ 的值,各 $S_{i,j}$ 之间达到 C^1 连续,或者说,

$$F(u,v) = S_{[u][v]}(u - [u], v - [v]),$$

其中 $[u]$ 表示不超过 u 的最大整数.

$$Q_{3\alpha,3\beta}^{i,j} = R_{i+\alpha,j+\beta} \quad (\alpha, \beta = 0, 1) \quad (1 \leq i \leq m, 1 \leq j \leq n), \tag{4}$$

$$\begin{cases} Q_{1,3\beta}^{i,j} = R_{i,j+\beta} + (R_{i+1,j+\beta} - R_{i-1,j+\beta})/6 \\ Q_{2,3\beta}^{i,j} = R_{i+1,j+\beta} + (R_{i,j+\beta} - R_{i+2,j+\beta})/6 \\ Q_{3\alpha,1}^{i,j} = R_{i+\alpha,j} + (R_{i+\alpha,j-1} - R_{i+\alpha,j+1})/6 \\ Q_{3\alpha,2}^{i,j} = R_{i+\alpha,j+1} + (R_{i+\alpha,j} - R_{i+\alpha,j+2})/6 \end{cases} \quad (\alpha, \beta = 0, 1), \tag{5}$$

$$\begin{cases} Q_{1,\beta}^{i,j} = Q_{0,\beta}^{i,j} + (Q_{3,\beta}^{i,j} - Q_{0,\beta}^{i,j-1})/6 \\ Q_{2,\beta}^{i,j} = Q_{3,\beta}^{i,j} + (Q_{0,\beta}^{i,j} - Q_{3,\beta}^{i,j+1})/6 \end{cases} \quad (\beta = 1, 2). \tag{6}$$

这样, $Q_{\alpha,\beta}^{i,j}$ 便是 16 个点 $\{R_{i+x,j+y}, -1 \leq x, y \leq 2\}$ 的线性组合.

用同样的方法可以为 $I(m,n)$ 的绿色分量 $G(m,n)$ 和蓝色分量 $B(m,n)$ 构造分块双三次 Bézier 插值 C^1 曲面.

可以验证,这种构造方法符合第 1.2 节中的条件.因此, $F(u,v)$ 在整体上是 C^1 连续的.所以, $F(u,v)$ 是 $I(m,n)$ 信息的一种连续表示.我们注意到,每一 $S_{i,j}$ 只依赖于像素 (i,j) 附近的 4×4 个像素的值,因而 $S_{i,j}$ 具有局部性质,这就是第 3 节中加速方法的理论基础.另一方面,插值曲面 $F(u,v)$ 具有一次严格性,当所有 $R_{i,j}$ 共面时, $F(u,v)$ 便是一个平面.如果 $I(m,n)$ 表示一张色彩均匀变化的图像,任意放大,它仍然是均匀的.

2 利用 Bézier 插值曲面放大图像

我们将图像放大描述为如下问题:

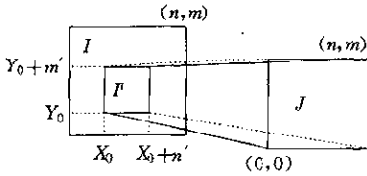


图1 图像放大

设 $I(x, y)$ 是一个 $(n+1) \times (m+1)$ 的数字图像, $I'(x', y')$ 是 $I(x, y)$ 的一部分, 其大小为 $(n'+1) \times (m'+1)$. 现在要求将 $I'(x', y')$ 放大为 $(n+1) \times (m+1)$ 大小的图像 $J(x, y)$, 如图 1 所示.

可以看出, $J(m, n)$ 相对于 $I'(m', n')$ 而言, 在 i 和 j 方向上的压缩比为

$$scale_x = (n'+1)/(n+1),$$

$$scale_y = (m'+1)/(m+1).$$

令 $R(i, j)$ 是 $J(m, n)$ 的第 i 行第 j 列像素 $P(i, j)$ 的红色分量. 由映射关系可以知道, $P(i, j)$ 对应于 $I'(m', n')$ 中的位置 (x', y') 为

$$x' = scale_x \times i,$$

$$y' = scale_y \times j.$$

令

$$x = [x'], \quad y = [y'],$$

$$coord_x = x' - x, \quad coord_y = y' - y.$$

则 $I'(m', n')$ 中坐标为 (x', y') 的点即为 $I'(m', n')$ 的第 (x, y) 像素内 $(coord_x, coord_y)$ 坐标处的点. 所以, $R(i, j)$ 的值就等于 $I'(m', n')$ 的红色分量的 Bézier 插值曲面的第 (x, y) 块 $S_{x,y}$ 在 $(coord_x, coord_y)$ 坐标处的值.

利用同样的方法, 可以求出绿色分量 $G(i, j)$ 和蓝色分量 $B(i, j)$ 的值.

3 加速方法

使用第 2 节中的方法, 可以产生质量很好的图像, 但速度比较慢. 为此, 我们提出如下的加速方法.

在第 1.3 节中我们已经看到, $S_{i,j}$ 具有局部性质, 其值只依赖于像素 (i, j) 及其附近的 4×4 个像素的值. 使用矩阵的形式可以更加清楚地表达这种依赖关系.

公式(3)写成矩阵形式为

$$S_{i,j}(x, y) = A \times B. \tag{7}$$

其中 $A = [a_0, a_1, \dots, a_{15}]$; $B = [b_0, b_1, \dots, b_{15}]^T$; $b_k (k=0, 1, \dots, 15)$ 为 Bézier 曲面的控制点, $b_0 = Q_{0,0}^{i,j}$, $b_1 = Q_{0,1}^{i,j}, \dots, b_{15} = Q_{15,15}^{i,j}$; $a_l (l=0, 1, \dots, 15)$ 为公式(3)中控制点 b_l 的系数, 它是 x 和 y 的函数.

公式(4~6)写成矩阵形式为

$$B = C \times P, \tag{8}$$

$$B = [b_0, b_1, \dots, b_{15}]^T,$$

$$C = \begin{bmatrix} C_{0,0} & C_{0,1} & \dots & C_{0,15} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ C_{15,0} & C_{15,1} & \dots & C_{15,15} \end{bmatrix}.$$

其中 $P = [p_0, p_1, \dots, p_{15}]^T$; $b_l (l=0, 1, \dots, 15)$ 为 Bézier 曲面的控制点, 如公式(5)所示; $p_k (k=0, 1, \dots, 15)$ 为 (i, j) 及其相邻像素的颜色分量, $p_0 = P_{i-1, j-1}$, $p_1 = P_{i-1, j}$, \dots , $p_{15} = P_{i+2, j+2}$; $C_{l,k}$ 为 b_l 关于 p_k 的系数.

由公式(7)和公式(8)可知:

$$S_{i,j}(x, y) = A \times B = A \times C \times P = M \times P, \tag{9}$$

其中 $M = [m_0(x, y), m_1(x, y), \dots, m_{15}(x, y)]$.

公式(9)表明, (i, j) 像素内 (x, y) 坐标处的点在 Bézier 插值曲面上的值是 (i, j) 像素及其相邻像素值的一个线性组合, 其系数分别为 $m_0(x, y), m_1(x, y), \dots, m_{15}(x, y)$.

使用第 2 节中的方法时, 大量时间均用于计算系数 $m_i(x, y) (i=0, 1, \dots, 15)$. 为解决这个问题, 我们将 (i, j) 像素沿 i 方向和 j 方向均匀地划分成 8 个部分, 这样便把 (i, j) 像素均匀地划分成了 64 个小块, 分别用

(0,0),(0,1),..., (7,7)标记这些小块,如图 2 所示.

可以看出,块(x,y)的左下角顶点的坐标为(x/8,y/8),可以预先计算出这些顶点的系数矩阵 $M_{x,y}$. 当图像放大时,若需计算 $S_{x,y}$ 在(u,v)(0≤u,v<1)坐标处的值,则首先判断出(u,v)所属的块为([8×u],[8×v]),然后用该块的左下角顶点对应的系数矩阵 $M_{[8×u],[8×v]}$ 近似(u,v)坐标所对应的系数矩阵,用该矩阵中的元素作为系数,将(i,j)及其相邻像素线性组合,其结果即为(u,v)处的色彩分量值.

实验表明,该方法可以大大提高计算速度,而由系数矩阵的近似所导致的误差非常小,对图像质量的影响不大.

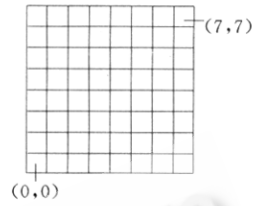


图2 像素子分

4 实验结果

我们在 Pentium 100 上用 VC++ 实现了该算法.

图 3 是一幅 400×400 的 24bit 的图像. 图 4 是采用本文的方法将图 3 中的一部分图像(左下角顶点为(50, 105), 右上角顶点为(149, 204))放大为 400×400 大小(放大比率为 4.0)所得到的图像. 图 5 是采用本文的方法将图 3 中的一部分图像(左下角顶点为(48, 48), 右上角顶点为(197, 197))放大为 400×400 大小(放大比率为 2.67)而得到的图像.

将一幅图像放大为 400×400 大小时,所需时间大约为 1.2s.

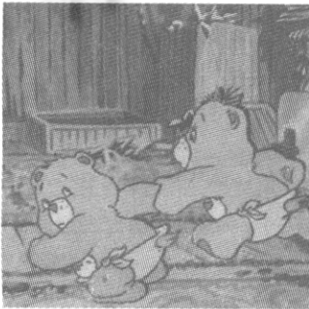


图 3

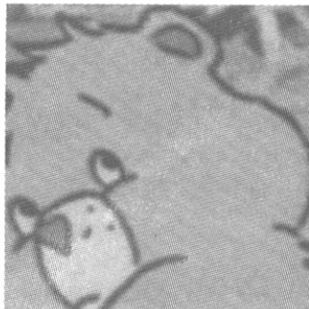


图 4

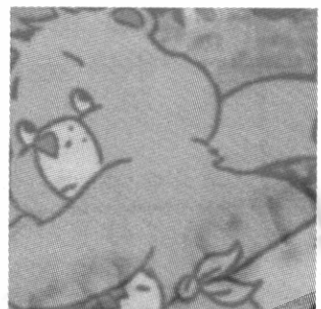


图 5

5 结论

本文提出了一种利用 Bézier 插值曲面进行图像放大的方法. 应用该方法可以在较短时间内生成放大图像,同时可以保证放大图像的质量.

参考文献

- 1 Durand C X, Faguy D. Rational zoom of bit maps using B-spline interpolation in computerized 2-D animation. Computer Graphics Forum, 1990,9(1):27~37
- 2 Chen S E. QuickTime VR——an image-based approach to virtual environment navigation. In: Cook R ed. Proceedings of the Siggraph'95 (Computer Graphics). Los Angeles, CA: Addison-Wesley Publishing Company, 1995. 29~38
- 3 McMillan L, Bishop G. Plenoptic modeling: an image-based rendering system. In: Cook R ed. Proceedings of the Siggraph'95 (Computer Graphics). Los Angeles, CA: Addison-Wesley Publishing Company, 1995. 39~46
- 4 蔡勇,刘学慧,吴恩华. 基于图像绘制的虚拟现实系统环境. 软件学报,1997,8(10):721~728
(Cai Yong, Liu Xue-hui, Wu En-hua. Image-based rendering: a technology for virtual reality system. Journal of Software, 1997,8(10):721~728)
- 5 唐荣锡,汪嘉业,彭群生. 计算机图形学教程. 北京:科学出版社,1994

(Tang Rong-xi, Wang Jia-ye, Peng Qun-sheng. Computer Graphics. Beijing: Science Press, 1994)

6 Farin G. Curves and Surfaces for Computer Aided Geometric Design (2nd ed). New York: Academic Press, 1990

A Method of Image Zooming-in Based on Bézier Surface Interpolation

SUN Qing-jie ZHANG Xiao-peng WU En-hua

(Laboratory of Computer Science Institute of Software The Chinese Academy of Sciences Beijing 100080)

Abstract A method of image zooming-in is presented in this paper. It is based on Bézier interpolation surface. First, it constructs a piecewise bicubic Bézier interpolation C^1 surface for each color component of a digital image, and then, samples the surface with different sampling rate to zoom in. The experimental results show that the method can improve the quality of zoomed image greatly.

Key words Zoom in, Bézier surface, interpolation, acceleration, pixel decompose.