

一种基于交互式的图像生成插值算法

王兴元 孟庆业 常沛军

(大连理工大学电子与信息工程学院,大连 116024)

摘 要 在对隔行扫描、四叉树和 Baptista 算法分析的基础上,提出了一种基于交互式的图像生成插值算法。该算法首先对图像进行均匀取样,然后基于待插像素点的邻域像素值分布的几何特征,对像素点进行评估以决定像素点的值。经过上述 4 种算法的对比实验,结果表明,采用插值算法,仅由较少的数据量,即可生成较清晰的图像。

关键词 隔行扫描算法 四叉树算法 Baptista 算法 插值算法 图像邻域

中图分类号: TP391.5 **文献标识码**: A **文章编号**: 1006-8961(2005)08-0949-08

An Interpolation Algorithm for Image Generation Based on Interactive Design

WANG Xing-yuan, MENG Qing-ye, CHANG Pei-jun

(School of Electronic & Information Engineering, Dalian University of Technology, Dalian 116024)

Abstract In this paper, based on the analysis of interlaced image, quadtree and Baptista algorithm, the authors presented an interpolation algorithm for interactive image generation: Firstly, sample evenly on image, then evaluate the pixel's value according to the distribution geometry characteristics of pixel's neighborhood. After the comparison among the four algorithms, the experimental results show that interpolation algorithm can generate high quality images with less data.

Keywords interlaced image algorithm, quadtree algorithm, Baptista algorithm, interpolation algorithm, Image neighborhood

1 引言

传统增强图像生成算法,如隔行扫描^[1]、四叉树^[2~8]算法都是按照从左到右、从上到下的扫描方式扫描整幅图像,计算每个像素点的颜色值和特征,然后把像素逐个输出到显示设备上形成最终图像。传统算法的缺点是:图像生成过程中不能显示出图像的整体特征,特别是在交互式环境中,无法让用户较早地了解图像的整体框架;由于目前网络带宽的限制,使用传统算法在网络上浏览较大的图像文件需较长的时间,特别是在处理庞大复杂的专业图像时,传统算法更是力不从心^[9]。1999年,Baptista提出了由部

分像素点再现整体图像的算法^[10]。本文对 Baptista 算法进行了改进,基于已知像素点邻域像素点分布的几何特征,对待插像素点进行了分类,从而预测像素的值。研究表明,由于采用了局部平滑策略,重构图像保留了较多的边缘细节信息;在图像的生成过程中,可以逐渐地了解图像的特征。

2 几种增强图像生成算法

2.1 隔行扫描算法

隔行扫描算法^[1]是一种传统的增强图像生成的方法,目前广泛地应用在图像的浏览方面。其原理是:在浏览或快速查看一幅图片时,只计算部分行的

基金项目:国家自然科学基金项目(69974008);辽宁省教育厅高等学校科学技术研究项目(20040081)

收稿日期:2004-05-09;改回日期:2005-03-04

第一作者简介:王兴元(1964~),男,教授,博士生导师。1999年毕业于东北大学计算机软件与理论专业并获博士学位。主要研究方向为分形理论及分形图像压缩。曾出版专著3部,在国内外核心期刊上发表论文70余篇(其中被SCI、EI收录共40篇)。E-mail: wangxy@dlut.edu.cn

像素,未被计算的像素值通过填充的方法显示。该算法的优点是减少了 CPU 运算的次数,节约了存储空间,加快了图像显示的速度。但由于被访问和计算的点分布得相对集中,故图像的清晰度降低很多。

2.2 二叉树算法

二叉树^[2-8]作为图像处理的常用分割方式,也可以用于图像的生成。根据分而治之和逐步求精的思想,采用二叉树增长的模型(如图 1 所示)来描述图像的生成过程。在这里可以把每个叶子结点想象为 $2^k \times 2^k$ 大小的子图像甚至可以是一个像素点,用叶子结点来区分图像的子块单元,随着二叉树的向下延伸使图像逐渐由模糊变为清晰,最终通过不断向下延伸生长的子块来构造整幅图像。



图 1 经典二叉树增长模型

Fig. 1 Classical quadtree increasing model

图 2 给出了二叉树广度优先算法^[11]生成 128×128 Lena 图像的过程,其中图 2(a) ~ 图 2(d)采用了填充策略,而图 2(e) ~ 图 2(h)未采用填充策略。该算法仍然采用从左到右、从上到下的次序访问叶子节点。

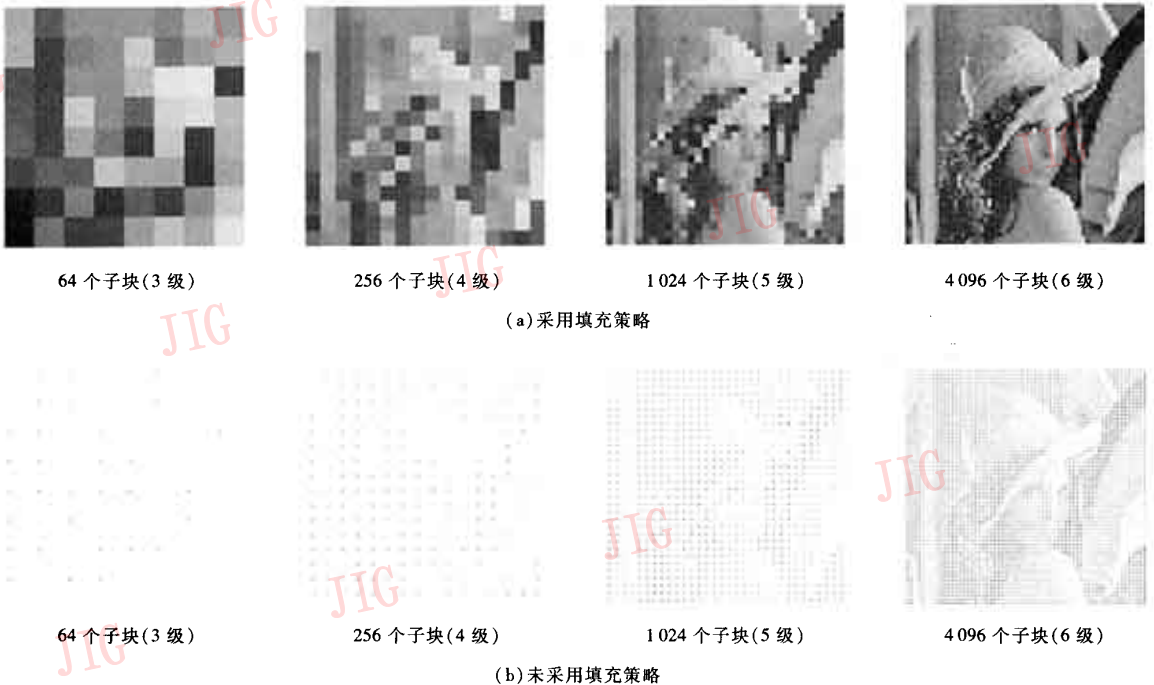


图 2 二叉树广度优先算法生成图像的过程

Fig. 2 Processing for generating images with quadtree extent priority algorithm

在二叉树广度优先算法中,对整个系统资源占用较多,特别是对空间的需求非常大。尽管生成图像的质量令人满意,但由于它无法根据用户的需求选取特征点去重构图像,因此该算法不能应用到交互式环境中。二叉树深度优先算法可减少对存储器的需求,但采用此算法通常生成了一个由不规则分布的点组成的图像,像素点的不均匀性使生成图像的质量不佳,因此更无法把握图像的整体特征。二叉树广度优先和深度优先算法一样,其像素分布同

样是不均匀的,并有低端开销。

2.3 Baptista 算法

Baptista 提出解决问题的方法^[10]是以抖动方法为基础的,不需要计算所有的像素点,而只是提取其中部分像素点进行计算,用已知点的百分率平均地覆盖这个区域,就可以保证在图像生成过程中提供图像的特征信息。在加快图像浏览和查看速度的同时,也节约了数据的传输量。这种增强 2 维图像生成的算法,可较好地应用在交互式的图像处理中。

本文的均匀取样模型借鉴了 Baptista 算法,但是取点序列不尽相同,也没有采用位运算。

3 图像插值算法

通常一幅图像中包含了丰富的目标和背景信息。若图像的某局部区域像素点的像素值存在着明显的跳跃和突变,则该局部区域是边缘与纹理十分丰富的区域,即存在着目标与背景信息;若某局部区域中所有的像素点均有同一属性,则在该区域中每个像素点的值都比较相近,不存在很大的跳跃点,这种局部区域是图像中的平坦区域,其占了图像所含的大部分信息。因此,在图像插值过程中,应该对图像进行分类处理,根据局部区域的不同属性采取相应的插值方法。

总之,不管某局部区域是图像中的边缘纹理区域,还是平坦区域,区域中的所有像素点都与其他像素点存在着密切的联系和相关性。因此在图像插值过程中,应该通过分析待插值点的像素值的不同属性进行图像插值,即分析待插值点是否为目标,或是背景,或属于过渡带。对不同的情况,应将待插值点分别划到局部区域中相应的类别中去,通过使类内离差最小、类间离差最大来估计这个待插值点的像素值。本文插值算法中局部区域的划分原则就是由待插值点所处邻域的几何特征决定的。

3.1 图像的均匀取样方法

寻找一个可列举出给定图像中所有满足以下规则的点的目标函数: $f(c) \rightarrow (x, y) (c \in \{0, 1, \dots, 2^k - 1\}; x, y \in \{0, 1, \dots, 2^k - 1\}; k$ 为二进制计数器的位数)。 f 可产生一个指示序列用来代替传统意义上 $M \times N$ 的矩阵 (M 和 N 分别表示图像行和列像素点的个数)。这里使用的图像,大小为 $2^k \times 2^k$,该方法可以扩展到任意大小的图像。

将图像看作一个 2 维矩阵,把一个 $2K$ 比特的计数器拆分为两个大小为 K 比特的坐标轴索引,分别用偶和奇数位的比特生成水平和垂直方向的索引。首先把 $2K$ 比特的计数器分成两个索引 i 和 j ; 然后颠倒这两个二进制索引,得到新的索引 i_1 和 j_1 , 指向通用的像素。但这却产生了一个问题: $\text{image}(i_1, j_1)$ 中的点不是均匀分布的。特别当计算到 50% 的像素点时,像素点排列成两条线,而不是均匀的模式。但是用这个方法填充图像所有的像素,每个像素仅访问一次。如果把图像当作环面来

考虑,这种取样无疑是最佳的。

为解决不均衡分布问题,可以把关于 $y = x$ 直线对称的行列相加。按 $(i, j) \rightarrow ((i + j) \bmod 2^k, i)$ 进行换算,就可以得到 $\text{image}(i_2, j_2)$ 。以 4×4 的 2 维图像为例描述图像的生成策略如表 1 所示。其中 $\text{image}(i_1, j_1)$ 、 $\text{image}(i'_1, j'_1)$ 和 $\text{image}(i_2, j_2)$ 分别表示圆环、Baptista 及插值算法的取样策略,若对插值算法的每一级给 $2^{\lfloor 2K - 1 - \lfloor \log_2(C + 0.5) \rfloor \rfloor}$ 个像素点着色 (C 是实际计数器的值, K 是坐标比特数),可得到增强图像清晰度的插值取样策略 $\text{image}(i'_2, j'_2)$ 。由表 1 可见插值算法可在最大间距下预测像素值。

下面具体说明上述算法的取样策略。若令计数器 $\text{Counter} = 0111$, 则 Counter 的偶数位为 $i = 11$, 奇数位为 $j = 01$; 颠倒 i, j 的二进制位得到 $i'_1 = 11$ 和 $j'_1 = 10$, 故可得到圆环法下一个取样点 $(3, 2)$; 按 $(i_1, j_1) \rightarrow (i_1, (i_1 + j_1) \bmod 2^k)$ 换算可得到 Baptista 算法的下一个取样点 $(3, 1)$; 按 $(i_1, j_1) \rightarrow ((i_1 + j_1) \bmod 2^k, i_1)$ 换算即可得到插值算法的下一个取样点 $(1, 3)$ 。 $\text{Counter} = 0111$ 时, 由表 1 可知此时取样点已达到全部图像的 50%, 而圆环法的取样点却集中在偶数行, 奇数行却没有一个取样点, 显然不均匀分布的取样点将对重构图像的质量造成很大影响; 此时 Baptista 算法和插值算法的取样点却分布得很均匀, 没有偏重哪个局部区域取样, 取样的均匀化为图像的重构奠定良好的基础。

图 3 是使用 $\text{image}(i_2, j_2)$ 取样方法, 没有使用填充策略生成的 128×128 Lena 图像的过程。需要说明的是: 按照表 1 中 $\text{image}(i_2, j_2)$ 的取点方式, 图像是按照每多计算一类像素点, 图像的取样率 η 增加 6.25% 逐渐完成的。例如当要取得 10% 的像素点时, 采用一部分点按照 $C = 0000$ 的图形取像素点, 一部分按照 $C = 0001$ 的图形取像素点, 并且尽量让这两种取法按比例分布到整个图像当中去。

3.2 邻域的定义及分类

(1) 图像邻域的定义 对于坐标为 (x, y) 的像素 p , 它可以有 4 个水平和垂直的近邻像素, 其坐标为 $(x+1, y), (x-1, y), (x, y+1), (x, y-1)$ 。这些像素(均用 r 表示)组成 p 的 4-邻域, 记为 $N_4(p)$, 如图 4(a) 所示。 p 的 4 个对角近邻像素(用 s 表示)的坐标是 $(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$, 记为 $N_D(p)$, 如图 4(b) 所示; p 的 4 个近邻像素加上 4 个对角邻域像素构成 p 的 8-邻域, 记为 $N_8(p)$, 如图 4(c) 所示。

表 1 圆环、Baptista、插值及增强清晰度的插值取样方法对比

Tab. 1 The Comparison of sampling methods for cirque, Baptista, interpolation and increasing definition interpolation

Counter	i_1	j_1	i_2	j_2	image(i_1, j_1)	image(i_1', j_1')	image(i_2, j_2)	image(i_2', j_2')
0000	00	00	00	00				
0001	10	00	10	10				
0010	00	10	10	00				
0011	10	10	00	10				
0100	01	00	01	01				
0101	11	00	11	11				
0110	01	10	11	01				
0111	11	10	01	11				
1000	00	01	01	00				
1001	10	01	11	10				
1010	00	11	11	00				
1011	10	11	01	10				
1100	01	01	10	01				
1101	11	01	00	11				
1110	01	11	00	01				
1111	11	11	10	11				

(2) 邻域的分类 对于一个待插值点的邻域 D_j , 它的每个像素点的像素值为 g_{ji} , 其中 $i = 1, 2, 3, 4$ 。根据的像素点分布的几何特征, 把邻域分为光滑邻域 D^* 、边缘邻域 D^e 和中间纹理邻域 D^m 3 类。分类的依据是邻域的像素值的均值和方差。一个邻域 D_j 的像素均值为

$$\text{aver } g_j = \frac{1}{m} \sum_{i=1}^m g_{ji}$$

方差为

$$\text{var } g_j = \frac{1}{m} \sum_{i=1}^m (g_{ji} - \text{aver } g_j)^2$$

其中, m 为像素 p 的近邻像素点数。

给定阈值 T_1, T_2 和 T_3 , 满足 $T_1 \ll T_3, T_2 < T_3$ 。邻域 D_j 的分类为

$$\textcircled{1} \text{var } g_j < T_1 \text{ 或 } \sum_{\substack{i_1 \neq i_2 \\ i_1, i_2 \in D_j}} |g_{ji_1} - g_{ji_2}| < T_2 \text{ 时, 为光}$$

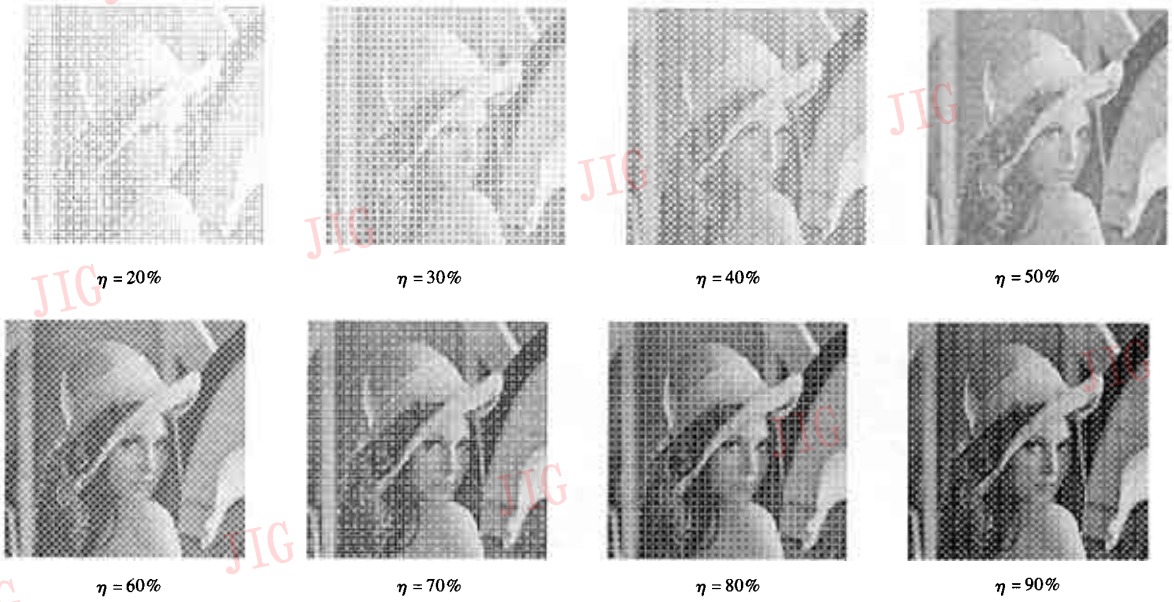


图 3 无填充的插值算法生成 Lena 图像的过程

Fig. 3 Processing for generating Lena image with non-filling interpolation algorithm

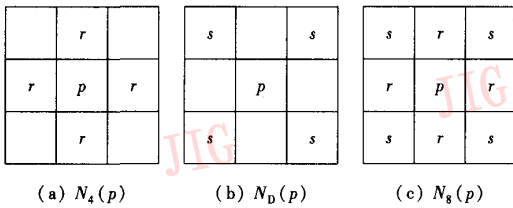


图 4 图像邻域示意图

Fig. 4 Sketch map of adjacent regions of image

滑类 D^* 。在此类邻域中,每个像素之间的像素值差很小,待插值点的像素值 $I(p) = \text{aver } g_j$ 。

② $\text{var } g_j > T_3$ 且 $\sum_{\substack{i_1 \neq i_2 \\ i_1, i_2 \in D_j}} |g_{j_{i_1}} - g_{j_{i_2}}| > T_2$ 时,为边缘类 D^* ,在此类邻域中,包含了物体的边缘,有明显的黑白反差,待插值点的像素值 $I(p) = \max[I(r)]$ ($r \in D_j$)。

③ $T_1 < \text{var } g_j < T_3$ 且 $\sum_{\substack{i_1 \neq i_2 \\ i_1, i_2 \in D_j}} |g_{j_{i_1}} - g_{j_{i_2}}| > T_2$ 时,为纹理类 D^m ,其方差不像边缘类那么大,但像素点之间的像素差仍比较大,待插值点的值可以取该邻域的中间像素值,即 $\min[I(r)] < I(p) < \max[I(r)]$ ($r \in D_j$)。

为保证重建图像的质量,需要由局部统计实验来确定阈值 T_1 、 T_2 和 T_3 ,一般取: $T_1 \ll T_3$;

$$\frac{1}{n} \sum_{j=1}^n \left[\frac{1}{m} \sum_{i=1}^m (g_{ji} - \text{aver } g_j)^2 \right] \leq T_3 < \max \left[\frac{1}{m} \times \sum_{i=1}^m (g_{ji} - \text{aver } g_j)^2 \right] \quad (j = 1, 2, \dots, n)$$

$$T_2 \geq \frac{1}{n} \sum_{j=1}^n \sum_{\substack{i_1 \neq i_2 \\ i_1, i_2 \in D_j}} |g_{j_{i_1}} - g_{j_{i_2}}|$$

。本文取 $T_1 = 200$ 、 $T_3 = 5\,000$ 、 $T_2 = 853$ 。

4 实验与结果

要发挥图像生成算法的优势,需建立起重构图像完整的体系结构,即将此图像生成方案内置在浏览器中,以达到快速浏览的目的。在交互式应用中,二叉树算法很难适用于用户自主选择图像质量及修改生成参数^[2-8]。为此仅对插值算法与隔行扫描算法进行了比较。

把 128×128 的 Lena 图像划分成若干个 4×4 的单元,在每一个单元中,按照表 1 的插值算法 $\text{image}(i_2, i_2)$ 的取点顺序进行图像插值。这样就保证了取得的像素点尽可能是均匀分布的,而不会出现类似隔行扫描和二叉树深度优先算法中像素点相对集中的情况。图 5 为取样率 $\eta = 20\% \sim 90\%$ 时,采用填充策略的插值算法生成 Lena 图像的过程。由于采用了均匀取样算法,使图像的效果较好。由图 5 可见,采用填充策略的图像取样率 η 在 $0\% \sim 20\%$ 之间时,图像的质量有明显的跳变; η 为 $20\% \sim 30\%$ 时,已经非常接近原始图像了,尽管在图像的边

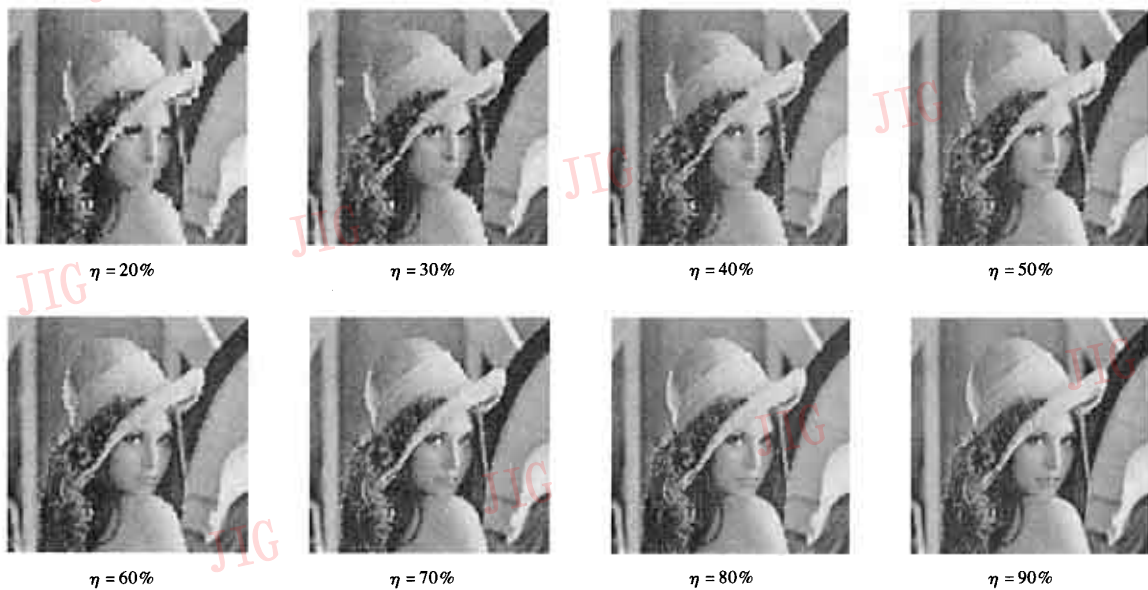


图 5 采用填充策略的插值算法生成 Lena 图像的过程

Fig. 5 Processing for generating Lena image with filling interpolation algorithm

缘有许多毛刺,但是基本上可以把握图像的整体特征; η 为 40% 左右就可以更清晰地把握图像的整体特征; η 大于 40% 时生成图像之间没有太大的差异,仅仅细节部分的光滑度有些不同。

Baptista 采用了填充方式的均匀取样算法生成图像,尽管生成图像的质量高于隔行扫描算法,但由于填充过程中降低了图像的边缘信息,故图像有明显的块效应^[10]。对 512×512 的 Lena 图像进行测试,图像插值算法与 Baptista 算法的峰值信噪比 (PSNR) 如图 6 所示。

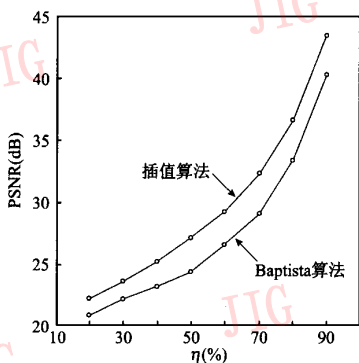


图 6 Baptista 算法与插值算法的取样率与峰值信噪比的关系曲线

Fig. 6 The relation curve of between sampling rate and peak value signal-to-noise rate

由图 6 可见,插值算法的 PSNR 要高于 Baptista 算法的。

图 7 为图像插值算法与 Baptista 算法重建图像的对比图,可以看出插值算法重建图像的效果要略好于 Baptista 算法的。

图 8 为图像插值算法与隔行扫描算法生成 512×512 Lena 图像的结果。

$\eta = 6.25\%$ 时,隔行扫描算法生成图像的质量无法与插值算法的相比。尽管只有 6.25% 的数据量,使用插值算法后,就可以清楚地观察到图像的特征。这表明,仅取原图像 $1/16$ 的特征点,节约 $15/16$ 的数据传输,仍可得到较高清晰度的图像。同时还可以观察到此时的问题是看不清图像细节,特别是亮度较暗区域。从重建图像的客观标准上,也可以看到两个算法的差异, $\eta = 6.25\%$ 时插值算法生成图像的 PSNR 为 22.6407 (dB),远远高于隔行扫描算法的 17.6369 (dB) (图 6)。

$\eta = 25\%$ 时,使用插值算法生成图像的质量仍然明显优于隔行扫描算法,插值算法生成图像基本上每一处细节都清晰可见,图像的光滑度也优于隔行扫描算法。隔行扫描算法生成的图像在细节处较粗糙,在帽沿、发梢上有明显的齿痕。此时插值算法的 PSNR 为 27.0165 (dB),而隔行扫描算法的 PSNR 为 24.0704 (dB)。 $\eta = 25\%$ 时,这两种算法生成图像已经相当清晰,原始图像与重建图像之间的差异较



图 7 插值算法与 Baptista 方法重建图像的对比

Fig. 7 The comparison of reconstruct images for Baptista algorithm and interpolation algorithm

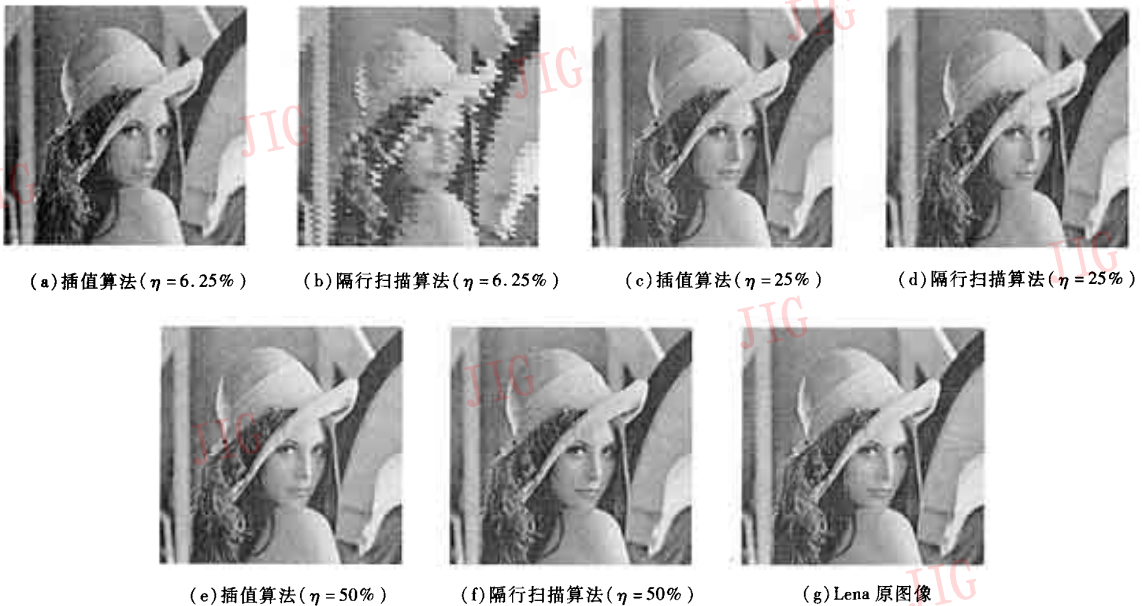


图 8 插值算法与隔行扫描算法生成 Lena 图像的对比

Fig. 8 Compare of generating Lena image for interpolation algorithm and interlaced algorithm

小,即保真度较高,可以满足图像搜索、图像浏览的需要。

$\eta = 50\%$ 时,无论是隔行扫描还是插值算法生成的图像与原始图像比较,还是两种算法生成的图像之间的比较,在图像质量上,在主观上已经很难分清,此时插值算法的 PSNR 为 29.318 5 (dB),而隔行扫描算法的 PSNR 为 28.853 6 (dB),可见,插值算法生成图像的质量在主观上与隔行扫描算法接近,但从客观标准上仍略优于隔行扫描算法。

5 结 论

通过对 Baptista 算法的改进,提出了一种基于交互式的图像插值算法。研究结果表明:插值算法

在生成图像的质量上要优于隔行扫描与 Baptista 算法,且生成的图像保真度较高,与原始图像的偏离程度较低。与二叉树和隔行扫描算法相比,插值算法更适合应用于交互式环境中,它可以根据用户的需要生成较高清晰度的数字图像,这样就节约数据的传输量,从而减少了数据的传输时间,加快了图像的生成速度。如果把插值算法应用在网络上进行图像搜索、图像浏览和图像快速查看,或者应用在空间庞大的图像上,可以节省很多网络和系统资源。该算法也可以应用到 3 维建模表面的 2 维贴图 and 实时应用程序中,可以用到在连续图像显示时保证帧速,也可以用在科学成像的交互实验中。

对于图像插值算法,能够改进的地方还有很多。从硬件角度来讲,在专门的图像处理系统中,可以用

硬件来实现部分功能和算法。从软件角度来讲,可以采用多进程同时对图像进行处理。在编写程序的时候还应该在每种平台上分别对算法进行优化,尽量减少位运算,因为不是所有的编译器都能生成高效的位运算,也不是所有的程序都能处理好这些运算。利用插值算法生成图像时,需对图像进行低端评估和预测,特别是在确定阈值 T_1 、 T_2 和 T_3 时采用的局部统计法,需要消耗一定的系统资源。如果用这种方法生成 100% 的图像,反而会增加一些系统的负担。另外如何寻找到最优的阈值还需要进一步研究。

参考文献 (References)

- 1 Hebert D J. Cyclic interlaced quadtree algorithms for quincunx multiresolution[J]. *Journal of Algorithms*, 1998, **27**(1): 97 ~ 128.
- 2 Bijan G M. A generalized solution to the quadtree expected complexity problem[J]. *Pattern Recognition Letters*, 1995, **16**(5): 443 ~ 456.
- 3 Chen Z, Chen I P. A simple recursive method for converting a chain code into a quadtree with a lookup table[J]. *Image and Vision Computing*, 2001, **19**(7): 413 ~ 426.
- 4 Chung K L, Tseng S Y. New progressive image transmission based on quadtree and shading approach with resolution control[J]. *Pattern Recognition Letters*, 2001, **22**(14): 1545 ~ 1555.
- 5 Souza F J, Vellasco M M R, Pacheco M A C. Hierarchical neuro-fuzzy quadtree models[J]. *Fuzzy Sets and Systems*, 2002, **130**(2): 189 ~ 205.
- 6 Manouvrier M, Rukoz M, Jomier G. Quadtree representations for storage and manipulation of clusters of images[J]. *Image and Vision Computing*, 2002, **20**(7): 513 ~ 527.
- 7 Chen P M. Variant code transformations for linear quadtrees[J]. *Pattern Recognition Letters*, 2002, **23**(11): 1253 ~ 1262.
- 8 Golchin F, Paliwal K K. Quadtree-based classification in subband image coding [J]. *Digital Signal Processing*, 2003, **13**(4): 656 ~ 668.
- 9 Wang Xing-yuan. Fractal Mechanism of the Generalized M-J Set [M]. Dalian: Dalian University of Technology Press, 2002:1 ~ 58. [王兴元. 广义 M-J 集的分形机理 [M]. 大连: 大连理工大学出版社, 2002:1 ~ 58.]
- 10 Baptista H R. A method for incremental image generation [J]. *Computers & Graphics*, 1999, **23**(3): 449 ~ 454.
- 11 Chang H K C, Tso C K. A breadth-first quadtree coding scheme for multicolored spatial data [J]. *Computers & Geosciences*, 1996, **22**(2): 139 ~ 148.