

一种分频的分形图象编码方法^{*}

谭兆信 吴怡敏

(中山大学计算机科学系, 广州 510275)

摘 要 提出 1 种分频的分形图象编码方法将图象分为低频、高频、超高频 3 部分, 并针对各部分的不同特性采用相应的编码技术, 以实现有效的压缩. 实验数据表明, 与传统分形块变换方法相比, 该方法不仅使图象的压缩比有所提高, 而且压缩速度和还原速度均有明显的改进.

关键词 分形, 图象编码, 余弦变换

分类号 TP 391. 41

分形几何用简单的迭代式^[1]描述了传统的欧氏几何所不能描述的不规则的的自然景观. 知道 1 个迭代函数系统 (IFS), 可描绘出 1 幅分形图来, 反过来, 知道 1 幅图象, 可以求出它的 IFS 码^[2]. 基于这一思想, Barnsley 等人^[3]提出了分形图象编码技术, 取得了惊人的压缩比. 随后 Jacquin^[4]提出了分形块变换编码方法, 利用图象局部与局部之间的相似性, 实现机器的自动编码.

为了提高分形图象压缩的速度, 本文提出了 1 种采用余弦变换将图象分频, 综合余弦变换和分形块变换的编码方法 (简称 CFT). 实验表明, 该方法是切实可行的, 与传统的分形块变换 (FBT) 方法相比, CFT 的压缩速度和还原速度均有明显的提高. CFT 带来的另 1 个好处是, 便于实现图象的分层传送.

1 分形块变换编码

分形块变换编码的原理大致如下: 把原图象划分为互不重叠的区域, 称为值域块 (Range block); 同时把原图象划分为可以重叠的区域, 称为定义域块 (Domain block), 定义域块的大小应该比值域块大, 一般而言, 定义域块与值域块的相似比取 2: 1. 压缩的过程就是为每一个值域块 R 寻找与之匹配定义域块 D 的过程. 对 D 先后作如下 3 种变换^[5, 6]:

几何变换 在 xy 平面对 D 作变换, 得到 D' . D' 与 R 有同样的大小.

同构变换 同构变换包括恒等、反射、旋转等 8 种, 对 D' 作其中 1 种, 得到 D'' .

亮度变换 对 D'' 作亮度变换, 产生 D''' , 使得 $d'''_{i,j} = s^{\circ} \cdot d''_{i,j} + o$. 为了使能很好地逼近 R , 假定 R 的大小是 m^2 , 令

$$MSE = \frac{1}{m^2} \sum_{0 \leq i, j \leq m-1} (s^{\circ} \cdot d''_{i,j} + o - r_{i,j})^2 \quad (1)$$

^{*} 广东省自然科学基金 (930231) 资助项目

收稿日期: 1997-06-11 谭兆信, 男, 56 岁, 副教授

对 MSE求极小值，可以求出 s, o 的值.

如果 MSE小于某阈值，则 D 与 R 是匹配的，否则，继续在定义域库中搜索. 对每个值域块储存其 IFS码，包括：与之匹配的 D 在图象中的位置 (x, y) ，对 D 采用了哪一种同构变换 m ，亮度变换中的 s 和 o 值.

解码时，从压缩文件中读入各值域块的 IFS码，然后从任意初始图象出发，按照从 D 到 R 的变换，得到一图象，对该图象作同样的变换，如此迭代，最终可得到重建图象.

2 CFT图象编码方法

CFT算法通过余弦变换把图象分为低频、高频和超高频 3部分，由低频数据组合成 3幅子图象，对其逐个施以分形块变换编码，对高频部分采用量化和游程编码，对超高频则记录非零分量.

这里用来测试的图象是 3幅 256×256 的 256级灰度图象，对于重建图象的质量用峰值信噪比 (r_{SN}/dB)来衡量. 假设 256级灰度的原始图象由 n 个象素组成，其灰度值分别为 $u_1, \dots, u_n$ ，还原图象的相应象素的灰度值分别为 $u'_1, \dots, u'_n$ ，则 r_{SN} 为

$$r_{SN} = 10^{\circ} \lg [255^2 / (1/n^{\circ} \sum_{i=1}^n (u'_i - u_i)^2)] \tag{2}$$

显然， r_{SN} 越大，还原图象的质量就越高.

2.1 压缩图象

压缩图象分为图象分解与编码 2个步骤.

2.1.1 图象分解 把图象分成 4×4 的一系列区域，并将其灰度值减去 128,使其取值范围由 $[0, 255]$ 变为 $[-128, 127]$,然后每个区域 X 实行二维余弦变换 (DCT)(以下 $N=4$)

$$Y = AXA^T \tag{3}$$

其中， $a_j = \frac{2}{N} \left[c(i)^{\circ} \cos \frac{(2j+1)\pi}{2N} \right]$ ， $c(i) = \begin{cases} 1/\sqrt{2}, i=0 \\ 1, 1 \leq i \leq N-1 \end{cases}$

余弦变换是可逆的，且

$$X = A^T Y A \tag{4}$$

余弦变换把图象从空间域变换到频域，在频域矩阵 Y 中，随着行列逐渐远离原点 (左上角)，矩阵的系数表示越来越高的频率分量.

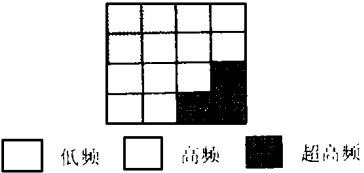


图 1 在频域分解图象

Fig. 1 Decomposition of the image in frequency domain

把频域矩阵 Y 分为 3部分：左上角 3个系数为低频，右下角 3个系数为超高频，剩下的高频部分，如图 1所示.

2.1.2 编码过程 低频部分集中了频域中的大部分能量，高频的能量较小，而超高频系数大部分为 0或接近于 0. 根据各部分的不同特性采用不同的编码方法.

(1) 低频. 提取每一频域矩阵中 $(0, 0)$ ， $(0, 1)$ 和 $(1, 0)$ 位置上的系数，形成 1个 6×4 的频域子图象，分别记为 F_A, F_B 和 F_C . 对 3幅频域子图象逐个实施分形块变换编码.

把值域块的大小定为 4×4 ,相应地，定义域块的大小定为 8×8 .对每个值域块 R 求 IFS码，即求与其匹配定义域块 D 的位置 x 与 y ，同构变换方式 m ，亮度变换中的 s 和 o . 在这里， x 与 y 各占 6 bits， m 占 3 bits， s 占 5 bits， o 占 8 bits. 值得注意的是，频域子图象中

的每点反映了原图象中 4×4 区域的信息, 因此如果单纯用分形编码, 在某些局部区域可能会导致较大的误差. 为了保证重建图象的质量, 对 MSE 值过大的值域块采用动态 Huffman 编码. 对 F_A , F_B 和 F_C 的编码由算法 1 实现, 以 F_A 为例.

算法 1

- 1) 设定阈值 Threshold, Row $\rightarrow$ 0
- 2) 循环, 当 Row < 64 时反复执行下列语句
 - ① Col $\rightarrow$ 0
 - ② 循环, 当 Col < 64 时反复执行下列语句
 - a) MSE $\rightarrow +\infty$
 - b) 如果 Col < 61

则对 F_A 中以 (Row, Col) 为起点位置的 1 个 4×4 区域作为值域 R , 对其寻求匹配的定域块 D 的位置 x 和 y , 同构变换方式 m , 亮度缩小 s 和亮度平移 o , 计算 R 和变换后的 D 之间的均方差 MSE
 - c) 如果 MSE $< \text{Threshold}$

则对 s, o 作离散化处理, 使其分别成为 5 bits 和 8 bits 的编码, 输出标志位 1, 输出 x, y, m, s, o 到编码文件; Col $\rightarrow$ Col + 4

否则输出标志位 0 到编码文件, 对位置 (Row, Col) ~ (Row + 3, Col) 上的 4 个系数实行动态 Huffman 编码; Col $\rightarrow$ Col + 1
 - ③ Row $\rightarrow$ Row + 4
- 3) 算法结束

(2) 高频. 对频域矩阵中的高频部分上的系数进行量化, 即对位置 (m, n) ($2 \leq m+n \leq 4, 0 \leq m \leq 3, 0 \leq n \leq 3$) 上的点除以量化系数 $1 + (1 + m + n) \times \text{quality}$, 其中, quality 为量化因子, 对量化后的方块进行曲徊排序, 如图 2 所示.

从左上角第 1 个频域矩阵开始, 沿着排序路线, 从左至右, 从上到下扫描所有的频域矩阵, 直到右下角最后 1 个频域矩阵. 对扫描途中经过的高频系数进行游程编码: 碰到非 0 系数, 则把它编码到压缩文件中去; 否则继续扫描, 直到达到一定的游程长度或者遇到非 0 系数为止, 输出 1 个特殊符号, 然后输出一定位数的信息, 表明 0 系数的游程长度.

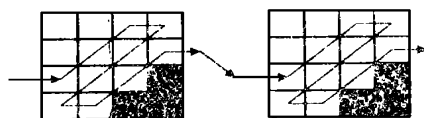


图 2 方块排列顺序

Fig. 2 The sequence of the blocks

(3) 超高频. 最后处理超高频部分的数据, 对超高频的系数进行量化, 记录其非 0 系数, 把这些非 0 系数所在的位置和具体的数值输送到压缩文件中去.

2.2 还原图象

解码过程则相反, 首先读入低频部分 3 幅子图象的编码, 分别通过迭代, 还原出 3 幅频域子图象, 并填充到各频域矩阵中的相应位置; 然后读入高频和超高频部分的编码, 解码后填入频域矩阵中剩下的空位, 对各频域矩阵逐个施以逆余弦变换, 即可重建图象.

3 实验结果

把传统的分形块变换编码 FBT 与本文提出的 CFT 作比较, 用 3 个标准的图象 Lena, Cronkite, Girl (图 3) 进行测试. 表 1 为实验数据. 其中, C 表示压缩比, tc 表示压缩时

间比, t_D 表示还原时间比.

与 FBT相比, CFT具有明显的速度优势. 因为 CFT仅对 3 个 64×64 的频域子图象实施分形编码, 而 FBT则对整个 256×256 的图象实行分形编码. 假定图象尺寸是 $n \times n$, 值域块大小是 $T \times T$, 定义域块大小是 $U \times U$, 则搜索次数 T 由下式给出

$$T= (n/T)^2 (n-U-1)^2 /w \quad (5)$$

在 FBT中, $n= 256, T= 8, U= 16$; 而在 CFT中, 对每幅子图象均有 $n= 64, T= 4, U= 8$. 因此在理想情况下, FBT与 CFT 的搜索次数之比为 $R= (256/8)^2 \cdot (256-16-1)^2 / [(3 \cdot 64/4)^2 \cdot (64-8-1)^2] \approx 23.835$

但为了实现更高的压缩比和保证重建图象的质量, CFT采用了一些额外措施, 比如, 若某值域块的 MSE值超出阈值, 则对第 1列实行熵编码, 其余 3列与后面紧接着的 1列再构成 1个值域块, 这会增加搜索时间. 实验结果表明, CFT的压缩速度大约是 FBT的 4倍. 类似的道理, CFT的还原速度也比 FBT快.



图 3 测试图象

Fig. 3 Test images

原始图象 (a)~ (c); FBT编码图象 (d)~ (f);
CFT编码图象 (g)~ (i)

表 1 FBT与 CFT的效果对比

Tab. 1 Comparisons between FBT and CFT

图样	C_r		r_{SN}/dB		t_c/s		t_D/s	
	FBT	CFT	FBT	CFT	FBT	CFT	FBT	CFT
Lena	8.16: 1	9.64: 1	30.482	30.486	3512	900	3	1
Cronkite	12.63: 1	14.70: 1	32.700	33.853	1266	447	3	1
Girl	8.44: 1	12.28: 1	30.976	30.304	3175	597	3	1

CFT在分层传送技术中也能方便应用. 分层传输 (Hierarchical transmission) 即对图象作分层处理, 在传输过程中, 最初传送图象的粗略部分, 然后分阶段地传输图象的细节, 使得在接收端, 得到一幅逐渐精确的图象. 这一技术在网络传递中具有一定实用价值.

由于 CFT把原始图象分为低频、高频、超高频来处理, 在低频部分又分 3个频域子图象逐个压缩, 因此可十分方便地实现分层传输. 在发送端可依次发送低频第 1、第 2、第 3子图象, 高频和超高频这 4部分的压缩数据, 而在接收端, 首先把频域方阵中所有元素置为 0, 每接受一部分数据, 即作解码, 把解码所得数据置于方阵中相应位置, 作逆余弦变换, 得出相应的还原图象.

本文在传统的分形块压缩方法的基础上, 结合余弦变换, 提出了 1种新的分频的分形

图象编码方法 CFT. 实验表明, 较之一般的分形块变换编码, CFT 具有更高的压缩比, 且压缩速度和还原速度均约为原来的 4 倍. 由于 CFT 对图象的低频、高频和超高频顺序压缩, 因此压缩后的图象便于实现分层传送, 在接收端由粗至精地还原图象.

对分形块变换编码的任何改进, 同样适用于 CFT. 由于分形编码的均匀性 (即对各值域块均采用同样的规则求其 IFS 码), 因此可以很方便地设计出相应的并行算法, 可以预计压缩速度会随 CPU 数目的增加而呈线性提高, 并行的 CFT 将达到更快的压缩速度.

参 考 文 献

- 1 Falconer K J. 分形几何——数学基础及其应用. 曾文曲译. 沈阳: 东北大学出版社, 1991. 14~127
- 2 高勇, 徐宗本. 分形图像压缩技术的数学基础. 数学进展, 1996, 25 (5): 400~412
- 3 Barnsley M F, Sloan A. A better way to compress images. Byte, 1988, 13 215~223
- 4 Jacquin A E. Image coding based on a fractal theory of iterated contractive image transformations. IEEE trans. Image Process, 1992, 1 (1): 18~30
- 5 Jacobs E W, Fisher Y, Boss R D. A study of iterated transform method. Signal processing, 1992, 29 (3): 251~263
- 6 Jackson D J, M shoud W. Parallel pipelined fractal image compression using quadtree recomposition. The Computer Journal, 1996, 39 (1): 1~13

A Frequency-Division Fractal Image Coding Approach

Tan Zhaoxin^{} Wu Yimin*

Abstract Presents a frequency-division fractal image coding approach, which decomposes an image into low frequency, high frequency and ultra-high frequency components. In order to compress images efficiently, different encoding methods are applied in different components. The result shows that the fractal image coding approach may achieve a higher compression ratio, compress and decompress faster than the standard fractal block transformation method does. Furthermore, it is capable of transmitting an image hierarchically.

Keywords fractal, image coding, cosine transformation

^{*} Department of Computer Sciences, Zhongshan University, Guangzhou 510275.