

## 差分与快速图像分形映射压缩算法\*

赵德平<sup>1</sup>, 朱伟勇<sup>1</sup>, 苏畅<sup>2</sup>, 陈维<sup>2</sup>

<sup>1</sup>(东北大学 信息工程学院, 辽宁 沈阳 110006);

<sup>2</sup>(辽宁省 人民医院 B 超室, 辽宁 沈阳 110015)

E-mail: zhao\_dp@263.net

http://www.syacei.edu.cn

**摘要:** 针对分形编码算法编码时间太长等缺点, 提出了对编码图像进行差分和分级逼近的新的分形编码算法. 在理论上对这一思想的可行性进行了有益的探索, 给出了该算法成立的理论基础. 根据此思想给出了一个新的具体实现分形编码的算法. 实验表明, 在提高压缩比和图像恢复质量的同时, 运算时间也大大缩短.

**关键词:** 分形; 差分; 算法; 分级逼近; 图像编码

**中图法分类号:** TP317 **文献标识码:** A

分形图像编码是 20 世纪 80 年代中后期提出的一种新型图像压缩编码方法<sup>[1~5]</sup>, 由于其具有高压缩比的潜力, 特别是 Jacquin 第一个实现完全由计算机编码以后, 近年来得到了迅速的发展<sup>[6~9]</sup>. 然而, 在用 Jacquin 方法实现图像编码算法的过程中发现, 比较平滑的子块易于用分形编码方法对其进行编码压缩; 而图像中纹理和频谱特征不同的区域就较难用该方法进行编码压缩. 在多数情况下, 如果对这部分图像不采用特殊的方法加以处理就使用分形方法对其进行编码, 会使得压缩后的重建图像和原始图像相差较远(传统方法对这部分图像的编码采用的是不断细分的方法, 并对细分后的每一块小图像再采用上述方法进行编码, 直至达到原始图像与重建图像的误差小于事先给定的标准为止); 同时, 也正是由于上述原因的存在, 增加了许多寻找最佳压缩变换的搜索范围和时间, 甚至(在小块图像不细分的情况下)搜索范围划定为整个图像, 也很难找到最佳压缩变换.

Jacquin 创造的完全由计算机实现的分形编码方法的一个最大缺点是对寻找最佳压缩变换的搜索范围的划定上, 由于该方法把整个图像作为搜索范围, 增加了大量的不必要的计算量. 该缺点的一个直接结果是导致了分形压缩编码时间太长, 以至于该方法很难达到普及性应用. 虽然在 Jacquin 以后其他研究人员所提出的方法中, 例如, 文献[7, 9]等提到的方法, 对 Jacquin 使用的方法有所改进, 但是, 在搜索范围问题上, 仍然存在许多问题, 没有在理论上给出论证或证明.

针对上述分形编码压缩的缺点, 本文提出了对图像进行分级预处理的方法, 即不同级的图像不作匹配处理, 这就使得被编码的小块图像仅在和自己同一级的小块图像中作搜索匹配块, 从而给寻找最佳压缩变换确定了合理的搜索范围, 节省了大量的搜索时间. 在实验中和理论上, 人们都发现有些小块图像即使搜索整个图像也很难找到较好的匹配, 当然, 在指定的同一级小块图像范围内进行搜索也不会得到较好的匹配. 为了解决这个问题, 本文对这样的小块采用“叠加”和“传统细分”相

\* 收稿日期: 1999-05-04; 修改日期: 1999-09-28

基金项目: 国家教育部博士点基金资助项目(97014510); 辽宁省自然科学基金资助项目(962177)

作者简介: 赵德平(1959-), 男, 辽宁辽中人, 副教授, 主要研究领域为计算机图形学, 计算机图像处理与编码; 朱伟勇(1938-), 男, 江苏无锡人, 教授, 博士生导师, 主要研究领域为最优设计, 混沌与分形, 计算机证明与构造; 苏畅(1964-), 女, 辽宁营口人, 副主任医师, 主要研究领域为超声专业; 陈维(1935-), 女, 辽宁沈阳人, 主任医师, 主要研究领域为超声诊断.

结合的处理方法,即把一小块图像分为两小块或3小块图像,使得新的两小块或3小块图像的叠加之和非常接近原小块图像,再对新的两小块或3小块图像进行分别编码,作为原小块图像的编码.由于光滑图像易于实现分形编码,因此,本文采用差分方法对图像进行压缩前的预处理,能使部分图像变得比原来图像更加光滑(见第1.8节示例).由于采用了对原始图像进行差分、分级预处理和叠加等新技术,使得图像压缩速度大大提高,同时,压缩比也有一定的提高.

## 1 分形压缩的理论基础

### 1.1 迭代函数系统

设  $D$  是  $m$  维欧氏空间  $R^m$  上的闭子集,称映射  $W: D \rightarrow D$  是  $D$  上的压缩映射,如果对  $D$  上的所有  $x, y$ , 存在一个常数  $s$  满足  $0 < s < 1$ , 使得

$$|W(x) - W(y)| \leq s|x - y|. \quad (1)$$

若式(1)取等式,则称  $W$  为相似映射.若  $w_i (i=1, 2, \dots, n)$  是压缩映射,则称  $\{w_i | i=1, 2, \dots, n, w_i: D \rightarrow D\}$  是压缩映射集,或称为迭代函数系统(iterated function system).

设  $W = \{w_i | i=1, 2, \dots, n\}$  是压缩映射集,称  $D$  的子集  $E$  对压缩变换集  $W$  是不变的,如果

$$E = W(E) \quad \text{或者} \quad \sum_{i=1}^n w_i(E) = E. \quad (2)$$

### 1.2 压缩映射定理

设  $W = \{w_i | i=1, 2, \dots, n\}$  是  $D$  上的压缩映射集,则存在唯一非空紧致集  $E$ , 它对  $W$  是不变的,并且可以从  $D$  中的任意一幅图像  $P$  开始,通过  $W$  反复迭代得到“不动点”图像  $E$ , 即

$$E = \lim_{k \rightarrow \infty} W^k(P).$$

### 1.3 上、下导数的定义

为了叙述方便,不妨仅对二维平面  $R^2$  上长度为  $2^n$  的闭正方形(包括边界)图像  $D$  加以定义.否则,只要用边长为  $2^n$  的正方形把图像包围,并把图像外、正方形内的图像灰度定义为 0 即可.

设  $D$  中任意一点  $(x, y)$  的灰度  $z$  定义为

$$z = f(x, y). \quad (3)$$

为了定义图像  $D$  的零阶下导数,引进如下记号:

$$L_0 = [l_{ij}], \quad (4)$$

其中  $l_{ij} = \inf f(x, y)$  当  $i, j=1, 2, \dots, 2^n$  和  $(x, y) \in D$  时,

$$L_1 = [l_{ij}], \quad (5)$$

其中(如图1(a)所示,把图像  $D$  均匀地分为4个相等的小正方形图像区域,分别用  $D_{11}, D_{12}, D_{13}, D_{14}$  表示)

$$l_{ij} = \begin{cases} \inf f(x, y) & \text{当 } i, j=1, 2, \dots, 2^{n-1} \text{ 和 } (x, y) \in D_{11} \text{ 时,} \\ \inf f(x, y) & \text{当 } i=2^{n-1}+1, 2^{n-1}+2, \dots, 2^n; j=1, 2, \dots, 2^{n-1} \text{ 和 } (x, y) \in D_{12} \text{ 时,} \\ \inf f(x, y) & \text{当 } i=1, 2, \dots, 2^{n-1}; j=2^{n-1}+1, 2^{n-1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{13} \text{ 时,} \\ \inf f(x, y) & \text{当 } i, j=2^{n-1}+1, 2^{n-1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{14} \text{ 时.} \end{cases}$$

$$L_m = [l_{ij}]. \quad (6)$$

其中(如图1(b)所示,把图像  $D$  均匀地分为相等的小正方形区域  $D_{m1}, \dots, D_{mv}$ ;  $v=2^m$ ,

$$p=r-1-2^m, q=p+1, r=2^{2m}.$$

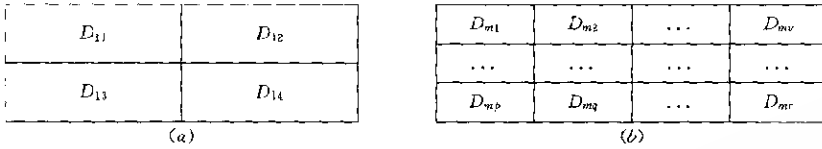


Fig. 1 Dividing image  $D$  equally  
图 1 均匀剖分图像  $D$

$$L_{ij} = \begin{cases} \inf f(x, y) & \text{当 } i, j=1, 2, \dots, 2^m \text{ 和 } (x, y) \in D_{m1} \text{ 时,} \\ \inf f(x, y) & \text{当 } i=2^{m-1}+1, 2^m+2, \dots, 2^{m-1}+1; j=1, 2, \dots, 2^{m-1} \text{ 和 } (x, y) \in D_{m2} \text{ 时,} \\ \dots \\ \inf f(x, y) & \text{当 } i=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n; j=1, 2, \dots, 2^{n-m} \text{ 和 } (x, y) \in D_{mn} \text{ 时,} \\ \dots \\ \inf f(x, y) & \text{当 } i=1, 2, \dots, 2^{n-m}; j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mp} \text{ 时,} \\ \inf f(x, y) & \text{当 } i=2^{n-m}+1, 2^{n-m}+2, \dots, 2^{n-m+1}; j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mq} \text{ 时,} \\ \dots \\ \inf f(x, y) & \text{当 } i, j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mr} \text{ 时.} \end{cases}$$

在定义了上述  $L_0, L_1, \dots, L_m$  符号之后, 定义图像  $D$  的零阶下导数为

$$\underline{D}^{(0)} = L_0. \tag{7}$$

一般在定义了  $D$  图像的  $(m-1)$  阶下导数之后, 定义图像  $D$  的  $m$  阶下导数为

$$\underline{D}^{(m)} = (m!) * (L_m - L_{m-1}). \tag{8}$$

为了定义图像  $D$  的上导数, 引进下面的记号:

$$H_0 = [h_{ij}], \tag{9}$$

其中  $h_{ij} = \sup f(x, y)$ , 当  $i, j=1, 2, \dots, 2^n$  和  $(x, y) \in D$  时,

$$H_1 = [h_{ij}], \tag{10}$$

其中(如图 1(a)所示)

$$h_{ij} = \begin{cases} \sup f(x, y) & \text{当 } i, j=1, 2, \dots, 2^{n-1} \text{ 和 } (x, y) \in D_{01} \text{ 时,} \\ \sup f(x, y) & \text{当 } i=2^{n-1}-1, 2^{n-1}+2, \dots, 2^n; j=1, 2, \dots, 2^{n-1} \text{ 和 } (x, y) \in D_{12} \text{ 时,} \\ \sup f(x, y) & \text{当 } i=1, 2, \dots, 2^{n-1}; j=2^{n-1}-1, 2^{n-1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{13} \text{ 时,} \\ \sup f(x, y) & \text{当 } i, j=2^{n-1}+1, 2^{n-1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{14} \text{ 时,} \\ \dots \end{cases}$$

$$H_m = [h_{ij}], \tag{11}$$

其中(如图 1(b)所示)

$$h_{ij} = \begin{cases} \sup f(x, y) & \text{当 } i, j=1, 2, \dots, 2^{n-m} \text{ 和 } (x, y) \in D_{m1} \text{ 时,} \\ \sup f(x, y) & \text{当 } i=2^{n-m}+1, 2^{n-m}+2, \dots, 2^{n-m+1}; j=1, 2, \dots, 2^{n-m} \text{ 和 } (x, y) \in D_{m2} \text{ 时,} \\ \dots \\ \sup f(x, y) & \text{当 } i=2^n-2^{m+1}, 2^n-2^{m+1}+2, \dots, 2^n; j=1, 2, \dots, 2^{n-m} \text{ 和 } (x, y) \in D_{mn} \text{ 时,} \\ \dots \\ \sup f(x, y) & \text{当 } i=1, 2, \dots, 2^{n-m}; j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mp} \text{ 时,} \\ \sup f(x, y) & \text{当 } i=2^{n-m}+1, 2^{n-m}+2, \dots, 2^{n-m+1}; j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mq} \text{ 时,} \\ \dots \\ \sup f(x, y) & \text{当 } i, j=2^n-2^{m+1}+1, 2^n-2^{m+1}+2, \dots, 2^n \text{ 和 } (x, y) \in D_{mr} \text{ 时.} \end{cases}$$

在定义了上述  $H_0, H_1, \dots, H_m$  符号之后,定义图像  $D$  的零阶上导数为

$$\overline{D}^{(0)} = H_0. \quad (12)$$

一般在定义了  $D$  图像的  $(m-1)$  阶上导数之后,定义  $D$  图像的  $m$  阶上导数为

$$\overline{D}^{(m)} = (m!) * (H_m - H_{m-1}). \quad (13)$$

#### 1.4 图像之间距离的定义

任意包含  $M$  个点的灰度数字图像  $\varphi$  可以看做是  $R^M$  (其中  $R$  表示实数集) 中的一个元素,用  $\Omega = \{1, 2, \dots, M\}$  表示图像中的坐标点集,在  $R^M$  中考虑距离测度.

$$\forall \varphi_1, \varphi_2 \in R^M, \quad d(\varphi_1, \varphi_2) = \left( \sum_{j=1}^M |\varphi_1(j) - \varphi_2(j)|^2 \right)^{1/2}. \quad (14)$$

#### 1.5 图像 $D$ 的展开级数及其收敛性

$$\begin{aligned} \max D(k, t) &= \overline{D}^{(0)} + \overline{D}^{(1)} * \frac{t}{1!} + \overline{D}^{(2)} * \frac{t^2}{2!} + \dots + \overline{D}^{(k)} * \frac{t^k}{k!}, \\ \min D(k, t) &= \underline{D}^{(0)} + \underline{D}^{(1)} * \frac{t}{1!} + \underline{D}^{(2)} * \frac{t^2}{2!} + \dots + \underline{D}^{(k)} * \frac{t^k}{k!}, \end{aligned}$$

其中  $t$  是  $(0, 1]$  单位区间上的变量.

在下面的定理 1 和定理 2 中,文章在一定的条件下按式(14)所定义的距离严格地证明  $\max D(k, t)$  和  $\min D(k, t)$  都是关于参数  $t$  一致收敛的矩阵级数,并且在  $t=1$  点,收敛到图像  $D$ .

**定理 1.** 假设图像  $D$  的最大灰度值是有界的,则矩阵级数  $\max D(k, t)$  ( $\min D(k, t)$ ) 在  $(0, 1)$  单位区间上是一致收敛的,并且  $\max D(k, t)$  ( $\min D(k, t)$ ) 收敛的矩阵级数  $\overline{D}$  ( $\underline{D}$ ) 还是连续的.

证明:对于固定的  $\delta \in (0, 1)$  和  $t \in (0, 1-\delta)$ ,由于(关于级数的细节,可参考文献[10])

$$\sum_{i=1}^p t^i \leq t^q / (1-t) < (1-\delta)q / \delta,$$

因此,只要  $q$  充分大,上式就可以任意小.另一方面,根据图像  $D$  的最大灰度值是有界的这一假设可知,  $\overline{D}^{(i)} / i!$  ( $\underline{D}^{(i)} / i!$ ) 都是有界矩阵.由此就证明了  $\max D(k, t)$  和  $\min D(k, t)$  在  $(0, 1-\delta)$  区间上都是一致收敛的矩阵级数,并且其收敛的矩阵级数  $\overline{D}$  ( $\underline{D}$ ) 还是连续的,再由  $\delta$  的任意性可知,定理 1 的结论成立.  $\square$

**定理 2.** 假设图像  $D$  的最大灰度值是有界的,并且当区域足够小时,其上的灰度值是常数,则图像  $D$  可以在单位 1 左侧附近展开成为收敛的矩阵级数.

证明:实际上,只要证明  $\max D(k, t)$  和  $\min D(k, t)$  在  $t=1$  点收敛到  $D$  即可.而根据上、下导数定义,并且当区域足够小时,其上的灰度值是常数的条件,易知

$$\lim_{k \rightarrow \infty} \max D(k, 1) = \lim_{k \rightarrow \infty} \min D(k, 1) = D.$$

另外,当  $k$  充分大时,  $D$  和  $\min D(k, t)$  (同理可对  $\max D(k, t)$  证明) 之间的距离满足:

$$|D - \min D(k, t)| \leq (1-t) * \left( \sum_{i=1}^k |\underline{D}^{(i)} / i!| * \sum_{j=0}^i t^j \right),$$

而根据当区域被剖分足够小时其上的灰度值是常数的条件,可以得出

$$\left( \sum_{i=1}^k |\underline{D}^{(i)} / i!| * \sum_{j=0}^i t^j \right)$$

是有上界的.因此,当  $t$  趋向 1 时,有下式成立:

$$\lim_{t \rightarrow 1} \lim_{k \rightarrow \infty} \min D(k, t) = D.$$

定理 1 和定理 2 的意义说明可以对图像进行“叠加”处理.并且,如果选择的  $t$  较好,就可以比

较容易地使用分形编码方法对图像进行“叠加”编码压缩。

### 1.6 $\max D(k,t)$ 和 $\min D(k,t)$ 级数表示成压缩变换的“不动点”集

仅对  $\max D(k,t)$  级数进行证明(同理可对  $\min D(k,t)$  级数进行证明). 同时, 为了叙述方便和表示清楚起见, 仅对  $D$  在均匀分成四等分(分别为  $D_{11}, D_{12}, D_{13}, D_{14}$ )和边长为 1 的情况下进行叙述.

设  $D_{1i}(i=1,2,3,4)$  的灰度值为  $g_i$ , 其中  $g_i$  是  $t$  的函数, 则  $D_{1i}$  与整个  $D$  是相似的, 压缩变换可以构造如下, 其中  $D_{1i}$  对应的压缩变换为下面的  $w_i(i=1,2,3,4)$ :

$$w_1 = \begin{cases} (0.5x, 0.5y, 0.5z + 0.5g_1) & (x, y) \in D_{11}, \\ (0.5x, 0.5y, 0.5z + g_1 - 0.5g_2) & (x, y) \in D_{12}, \\ (0.5x, 0.5y, 0.5z + g_1 - 0.5g_3) & (x, y) \in D_{13}, \\ (0.5x, 0.5y, 0.5z + g_1 - 0.5g_4) & (x, y) \in D_{14}, \end{cases}$$

$$w_2 = \begin{cases} (0.5x + 0.5, 0.5y, 0.5z + g_2 - 0.5g_1) & (x, y) \in D_{11}, \\ (0.5x + 0.5, 0.5y, 0.5z + 0.5g_2) & (x, y) \in D_{12}, \\ (0.5x + 0.5, 0.5y, 0.5z + g_2 - 0.5g_3) & (x, y) \in D_{13}, \\ (0.5x + 0.5, 0.5y, 0.5z + g_2 + 0.5g_4) & (x, y) \in D_{14}, \end{cases}$$

$$w_3 = \begin{cases} (0.5x, 0.5y + 0.5, 0.5z + g_3 - 0.5g_1) & (x, y) \in D_{11}, \\ (0.5x, 0.5y + 0.5, 0.5z + g_3 - 0.5g_2) & (x, y) \in D_{12}, \\ (0.5x, 0.5y + 0.5, 0.5z + 0.5g_3) & (x, y) \in D_{13}, \\ (0.5x, 0.5y + 0.5, 0.5z + g_3 - 0.5g_4) & (x, y) \in D_{14}, \end{cases}$$

$$w_4 = \begin{cases} (0.5x + 0.5, 0.5y, 0.5z + g_4 - 0.5g_1) & (x, y) \in D_{11}, \\ (0.5x + 0.5, 0.5y, 0.5z + g_4 - 0.5g_2) & (x, y) \in D_{12}, \\ (0.5x + 0.5, 0.5y, 0.5z + g_4 - 0.5g_3) & (x, y) \in D_{13}, \\ (0.5x + 0.5, 0.5y, 0.5z + 0.5g_4) & (x, y) \in D_{14}. \end{cases}$$

以此类推(此时  $k=2$ ), 可以用数学归纳法对  $k$  证明定理, 并且从任意图像出发, 该组变换经反复迭代都能收敛到级数  $\max D(k,t)$  ( $\min D(k,t)$ ). 由此, 可以得出下述结论:

**定理 3.** 设  $D$  是  $R^M$  中的非空紧子集, 任给  $\epsilon > 0$ , 则存在以  $E$  为不变集的相似压缩映射集  $\{w_i | i=1, 2, \dots, m\}$ , 使  $d(D, E) < \epsilon$ , 如图 2 所示.

(0,0)	(0.5,0)	(1,0)
$D_{11}$		$D_{12}$
(0,0.5)	(0.5,0.5)	(1,0.5)
$D_{13}$		$D_{14}$
(0,1)	(0.5,1)	(1,1)

Fig. 2 Dividing image  $D$  equally

图 2 均匀剖分图像  $D$

当然, 这里的  $E$  是随  $\epsilon$  的变化而变化的. 该定理为寻找一组压缩变换(并使该组压缩变换的迭代不变集就是要编码的图像的近似)的存在性提供了理论基础.

### 1.7 分级预处理

在分形编码实现过程中, 实现数据块快速搜索到其匹配块是提高压缩速度的一个关键环节. 而

解决这一问题的一个有效途径就是把被编码的数据块进行分级预处理,使被编码图像仅在其同级的数据块中寻找最佳匹配块.为此,我们将在下面较详细地叙述和讨论分级预处理的理论.

每一小块图像分级的基本原则是:

- (1) 每一小块图像灰度最大值与最小值的差.
- (2) 每一小块图像的最大值与该块平均值的差.
- (3) 每一小块图像的平均值与该块最小值的差.

根据上述原则,可以得出如下结论:

**定理 4.** 对于给定的  $\delta > 0$ , 两个级数的差大于  $\delta$  的图像块  $A$  和  $B$  (或称矩阵), 即  $(\text{Max } A - \text{Min } A) - (\text{Max } B - \text{Min } B) > \delta$ , 则  $A$  和  $B$  不能在误差失真  $\delta$  水平下相匹配.

证明: 记  $A$  和  $B$  分别是如下两个矩阵块 (其中  $A$  是某矩阵乘以压缩因子  $s$ , 然后再经过平移的结果):

$$A = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \dots & & \dots \\ a_{n1} & \dots & a_{nn} \end{bmatrix} \quad \text{和} \quad B = \begin{bmatrix} b_{11} & \dots & b_{1n} \\ \dots & & \dots \\ b_{n1} & \dots & b_{nn} \end{bmatrix},$$

并记

$$\text{Aver } A = \left( \sum_{i,j=1}^n a_{ij} \right) / (n * n), \quad \text{Aver } B = \left( \sum_{i,j=1}^n b_{ij} \right) / (n * n),$$

$$\text{Max } A = \max_{i,j=1,\dots,n} \{a_{ij}\}, \quad \text{Min } A = \min_{i,j=1,\dots,n} \{a_{ij}\}, \quad \text{Max } B = \max_{i,j=1,\dots,n} \{b_{ij}\}, \quad \text{Min } B = \min_{i,j=1,\dots,n} \{b_{ij}\}.$$

这里采用反证法来证明. 假设  $A$  和  $B$  分别是相差较远的两个级中的小块图像 (或称矩阵), 并且在误差小于或等于  $\delta$  水平下相匹配, 即

$$\sum_{i,j=1}^n (a_{ij} - b_{ij} - (\text{Aver } A - \text{Aver } B))^2 < \delta. \tag{15}$$

另一方面, 由式 (15) 和定理假设得出:

$$\begin{aligned} \delta < & |(\text{Max } A - \text{Min } A) - (\text{Max } B - \text{Min } B)| \leq |(\text{Max } A - \text{Min } A) - (b_{ki} - b_{pq})| = \\ & |(\text{Max } A - b_{ki} - (\text{Aver } A - \text{Aver } B)) - (\text{Min } A - b_{pq} - (\text{Aver } A - \text{Aver } B))| \leq \\ & |(\text{Max } A - b_{ki} - (\text{Aver } A - \text{Aver } B))| + |(\text{Min } A - b_{pq} - (\text{Aver } A - \text{Aver } B))| \leq \\ & \sum_{i,j=1}^n (a_{ij} - b_{ij} - (\text{Aver } A - \text{Aver } B))^2 < \delta, \end{aligned}$$

与假设矛盾. 这里需要详细解释的是  $b_{ki}$  和  $b_{pq}$ ,  $b_{ki}$  对应的  $a_{ki}$  为  $\text{Max } A$ , 而  $b_{pq}$  对应的  $a_{pq}$  为  $\text{Min } A$ .

定理的意义是十分重要的, 它是分级预处理的理论基础. 虽然文章把最大值与最小值的差作为分级的主要标准, 但是, 由于实际情况比较复杂, 因此, 当矩阵  $A$  与其对应的级不能较好地匹配时, 则下一个搜索目标就是“最大值与该块平均值的差”或者是“平均值与该块最小值的差”所对应的级.

### 1.8 差分理论

既然光滑图像易于编码, 那么如何使图像变得更加光滑呢? 采用差分方法就可以使部分图像变得比原来更加光滑. 对于图像  $D$ , 如果写成如下的 (离散) 矩阵形式:

$$D = \begin{bmatrix} d_{11} & d_{12} & \dots & d_{1n} \\ d_{21} & d_{22} & \dots & d_{2n} \\ \dots & \dots & \dots & \dots \\ d_{n1} & d_{n2} & \dots & d_{nn} \end{bmatrix},$$

则其差分  $\Delta D$  记为

$$\Delta D = \begin{bmatrix} \Delta d_{11}, & \Delta d_{12}, & \dots, & \Delta d_{1n} \\ \Delta d_{21}, & \Delta d_{22}, & \dots, & \Delta d_{2n} \\ \dots, & \dots, & \dots, & \dots \\ \Delta d_{n1}, & \Delta d_{n2}, & \dots, & \Delta d_{nn} \end{bmatrix},$$

其中

$$\Delta d_{ij} = \begin{cases} d_{i2} - d_{i1}, & \text{当 } j=1, i=1, 2, \dots, n \text{ 时} \\ d_{ij} - d_{i,j-1}, & \text{当 } j=2, 3, \dots, n, i=1, 2, \dots, n \text{ 时} \end{cases}$$

如果在保留第 1 列数据的同时,再保留对  $\Delta D$  的分形编码,则恢复  $D$  图像就能做到了.

下面举简例说明差分的优点.例如, $D$  为如下形式:

$$D = \begin{bmatrix} 14 & 13 & 13 & 12 & 13 & 14 & 14 & 13 \\ 13 & 14 & 13 & 13 & 12 & 12 & 13 & 14 \\ 13 & 14 & 12 & 14 & 12 & 14 & 13 & 14 \\ 13 & 14 & 14 & 14 & 13 & 13 & 14 & 14 \\ 16 & 15 & 14 & 15 & 15 & 16 & 15 & 15 \\ 4 & 5 & 6 & 4 & 5 & 5 & 5 & 5 \\ 4 & 6 & 5 & 6 & 4 & 5 & 4 & 6 \\ 6 & 6 & 5 & 5 & 5 & 4 & 3 & 4 \end{bmatrix},$$

$$\Delta D = \begin{bmatrix} -1 & -1 & 0 & -1 & 1 & 1 & 0 & -1 \\ 1 & 1 & -1 & 0 & -1 & 0 & 1 & 1 \\ 1 & 1 & -2 & 2 & -2 & 2 & -1 & 1 \\ 1 & 1 & 0 & 0 & -1 & 0 & 1 & 0 \\ -1 & -1 & -1 & 1 & 0 & 1 & -1 & 0 \\ 1 & 1 & 1 & -2 & 1 & 0 & 0 & 0 \\ 2 & 2 & -1 & 1 & -2 & 1 & -1 & 2 \\ 0 & 0 & -1 & 0 & 0 & -1 & 1 & 1 \end{bmatrix}.$$

由此可见, $\Delta D$  比  $D$  要光滑很多,而且  $\Delta D$  也比  $D$  易于进行分形编码.

## 2 图像压缩算法的具体实现

(1) 首先保留图像  $D$  的第 1 列信息,然后按照第 1.8 节中提出的方法对  $D$  图像进行差分预处理,这样就得到了比  $D$  更光滑的图像  $\Delta D$ . 本节谈到的小块图像都是指  $\Delta D$  中的小块图像.

(2) 把整个  $\Delta D$  图像划分为  $N$  个互不相交的子块  $R_i (i=1, 2, \dots, N)$ , 然后对每一子块  $R_i$  进行分级预处理,以决定把该块划分到哪一级中.

(3) 对每一小块图像进行分形编码,搜索匹配范围按照预处理指定的级中进行.

(4) 对匹配不能满足要求的块可以进行“叠加”或细分处理,然后对“叠加”或细分处理后的块进行重新匹配,直至得到满足要求的匹配为止.

(5) 如果每一小块图像的分形编码都完成,则结束. 否则,返回到(3).

## 3 实验结果

文章的实验对象是 Miss Amber, 这是一幅灰度图像,分辨率是  $320 \times 200 \times 8\text{bit}$ . 所有结果由图 3 给出,其中图 3(a) 是原始图像,图 3(b) 和图 3(c) 分别是用经典方法和本文新方法把图像分成  $8 \times 8, 4 \times 4$  和  $2 \times 2$  压缩后恢复的图像. 此外,在表 1 中给出压缩时间、压缩比和峰值信噪比 PSNR

(实验使用的是主频率 33 的 PC486DX2 计算机,用 C 语言编程实现),其中 PSNR 按下式计算:

$$\text{PSNR} = 10 * \log_{10}(255 \times 255 \times X \times Y / \text{DIS}). \quad (16)$$

这里,DIS 代表按式(14)定义的原始图像与恢复图像之间的距离, $X$  和  $Y$  则分别代表图像长度和高度.

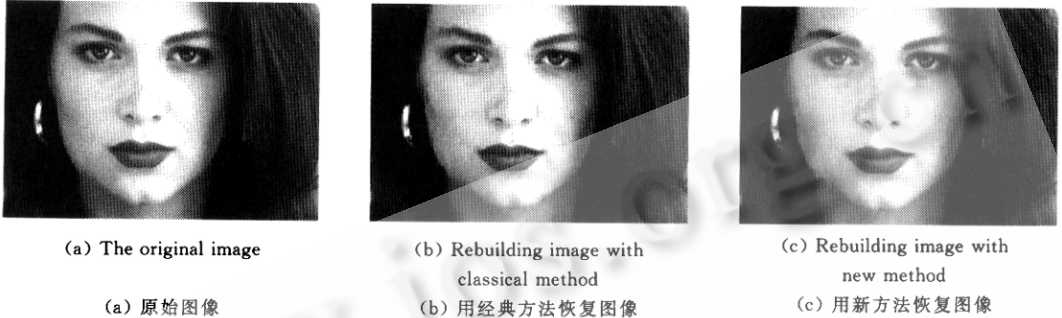


Fig. 3

图 3

Table 1 The coding results of two algorithms

表 1 两种算法的编码结果

	Size of range blocks <sup>①</sup>	Compression ratio <sup>②</sup>	PSNR	Search of domain blocks <sup>③</sup>	Using method <sup>④</sup>	Coding time (min) <sup>⑤</sup>
Image <sup>⑥</sup> 5	8×8,4×4,2×2	13.12	26.80	All domain blocks in the image <sup>⑦</sup>	Jacquin's classical method <sup>⑧</sup>	1 124
Image 6	8×8,4×4,2×2	20.00	26.50	Domain blocks of various grey levels <sup>⑨</sup>	New method in this paper <sup>⑩</sup>	18.0

①块的大小,②压缩比,③搜索范围,④采用方法,⑤编码时间(分钟),⑥图,⑦整个图像,⑧经典方法,⑨分级确定范围,⑩本文新方法.

## 4 结 论

本文针对分形编码压缩时间太长这一严重缺点,提出了对编码图像首先进行差分预处理,以使被编码的部分图像变得更加光滑;然后,再对差分预处理后的图像进行分级预处理以解决分形编码搜索范围太大的问题,从而使得分形编码的压缩时间极大地缩短.此外,由于匹配误差的存在,本文提出了用“叠加”图像这一新技术进行解决的方法.实验表明,进行差分预处理可以有效地解决部分图像的光滑性;而进行分级预处理则实现了分形编码搜索范围是可行的和有效的;“叠加”技术则较好地解决了匹配误差的问题.然而本文的工作还是初步的,尤其是图像分级的原则仍然需要进一步研究.

## References:

- [1] Barnsley, M. F., Sloan, A. D. A better way to compress images. BYTE Magazine, 1988,13(1):215~223.
- [2] Mandelbrot, B. B. The Fractal Geometry of Nature. New York: W. H. Freeman and Company, 1983.
- [3] Jacquin, A. E. A fractal theory of iterated Markov operators with applications to digital image coding [Ph. D. Thesis]. Department of Mathematics, Georgia Institute of Technology, 1989.
- [4] Jacquin, A. E. Image coding based on a fractal theory of iterated contractive image transformations. IEEE Transactions on Image Processing, 1992,1(1):18~30.
- [5] Falconer, K. J. Fractal Geometry—Mathematical Foundations and Applications. New York, NY: John Wiley and Sons,



Inc., 1989.

- [6] Jacquin, A. E. Fractal image coding: a review. *Proceedings of the IEEE*, 1993, 81(10):1451~1465.
- [7] Fang Yu-dong, Yu Ying-lin. A quick fractal image compression coding method. *Acta Electronica Sinica*, 1996, 24(1):28~33 (in Chinese).
- [8] Wang Zhou, Yu Yung-lin. An image compression algorithm based on partial fractal mapping. *Computer Research and Development*, 1998, 35(3):199~204 (in Chinese).
- [9] Lao Zhi-qiang, Pan Yun-he. A fast image compression algorithm based on its block coding. *Computer Research and Development*, 1998, 35(3):270~273 (in Chinese).
- [10] Beijing University's Higher Mathematical Teaching and Research Section Translated. *Differential and Integral Course*. Beijing: People's Educational Publisher, 1979, 2(2):378~392 (in Chinese).

#### 附中文参考文献:

- [7] 房育栋, 余英林. 快速分形图像压缩编码. *电子学报*, 1996, 24(1):28~33.
- [8] 王舟, 余英林. 图像的部分分形映射压缩算法. *计算机研究与发展*, 1998, 35(3):199~204.
- [9] 劳志强, 潘云鹤. 基于IFS块的快速图像编码方法. *计算机研究与发展*, 1998, 35(3):270~273.
- [10] 北京大学高等数学教研室译. *微积分学教程*. 北京: 人民教育出版社, 1979, 2(2):378~392.

## A Difference and Quick Image Compression Algorithm Based on Fractal Mapping

ZHAO De-ping<sup>1</sup>, ZHU Wei-yong<sup>1</sup>, SU Chang<sup>2</sup>, CHEN Fei<sup>2</sup>

<sup>1</sup>(School of Information Science and Engineering, Northeastern University, Shenyang 110006, China);

<sup>2</sup>(Ultrasonic Diagnosis B Clinic, Liaoning People's Hospital, Shenyang 110015, China)

E-mail: zhao\_dp@263.net

http://www.syacei.edu.cn

Received May 4, 1999; accepted September 28, 1999

**Abstract:** In this paper, a new fractal image coding algorithm is presented to process image coding by difference and approximation with various grey levels because the coding time of classical fractal image coding algorithm is very long. In addition, some researches with respect to its feasibility in compression theory are also discussed and the theoretical foundation of the new algorithm is given. Based on the idea, a new concrete algorithm for realizing fractal image coding is designed in the paper. The experimental results show that the new algorithm can improve compression ratio and reconstruct high quality images while computational time of image coding is reduced greatly.

**Key words:** fractal; difference; algorithm; approximation of different levels; image coding