

# 快速覆盖式分形压缩算法

颜飞翔 蔡宣平 孙茂印

(国防科技大学电子工程学院,长沙 410073)

**摘要** 提出了一种覆盖式分形压缩方法,并给出了相应快速搜索算法。该方法采用四连通扩充的方式,在给定误差范围内,尽可能地合并规则 Range 分区,从而减少所需迭代变换的数目。为减少存储 Range 分区信息所需的字节,在存储压缩映射时,根据 Range 分区多边形的边数调整存储顺序,对于那些复杂的 Range 分区,只记录其外接矩形框信息。在解码时,采用一种覆盖式绘制方法,可以完全恢复原 Range 分区情形。通过对 Lenna 图象的实验表明,在恢复图象的信噪比为 29.63dB 时,可以达到 41 倍的压缩倍数。

**关键词** 分形,图象压缩,IFS

## 1 引言

分形(Fractal)的概念最初是由 B. B. Mandelbrot 于 1975 年提出的<sup>[1]</sup>。在迭代函数系统理论<sup>[2]</sup>的基础上,Barnsley 和 Sloan 首次采用分形方法实现了对图象的压缩,声称可以将彩色图象压缩比提高至 10 000 比 1<sup>[3,4]</sup>。1989 年,Jacquin 以迭代函数系统理论和循环迭代函数系统理论<sup>[5]</sup>为基础,首次提出了一种分形图象压缩的自动实现算法<sup>[6]</sup>。从某种程度上说,Jacquin 的方法以及随后的几篇相关论文<sup>[7]</sup>打破了分形图象压缩的僵局,为更进一步的研究提供了一个新起点。

在 Jacquin 方法的基础上,Thomas 等人<sup>[8]</sup>提出了一种基于域的分形图象压缩方法。该方法在 Jacquin 方法的基础上,先对原图象进行规则划分,而后按扫描线的顺序,选定一个未处理过的规则 Range 分区作为种子,采用一种四连通种子扩充的方法,通过将最小方差距离测度与给定误差门限相比较,以决定在某一方向上的扩充是否可以接受,可以接受则将该相邻规则分区与当前种子合并。然而在应用该方法时存在一个缺陷,就是事先很难确定图象规则划分的合适尺寸,如果规则划分尺寸偏大,则可能对某个规则 Range 分区在整个 Domain 范围

内,都找不到一个合乎误差要求的最佳匹配 Domain 区。而如果规则划分尺寸偏小,则将导致计算量的加大和压缩所需时间的增加。

针对 Thomas 方法存在的缺陷,我们提出了一种覆盖式分形压缩方法,并给出了相应快速搜索算法。对那些其最佳匹配 Domain 区不满足误差要求的规则 Range 分区以一种递归的方式进行四叉树划分,将划分得到的 4 个子图象代替原规则分区。另外在存储 Range 分区信息时,采用一种覆盖式的算法,对边数较多的 Range 分区只需记录其外接矩形框,这样就从两个方面改善了分形图象压缩性能。

## 2 分形图象压缩 Jacquin 算法

我们先对 Jacquin 提出的方法作一个简单的描述。不妨假设有一幅点阵为  $64 \times 64$  二维图象  $f$ (如图 1 所示),其灰度级为 256,对它作  $8 \times 8$  划分(如图 2 所示),得到拼接分区。

在拼接分区中任意一个分区  $R_i$  均为  $8 \times 8$  大小的图象块,各拼接分区是不重叠的。对拼接分区中每个  $R_i$ ,根据拼贴定理及相应推论<sup>[9]</sup>,在源图象中搜索一个扩充分区  $D_i$ (为方便计算,取  $D_i$  面积为  $R_i$  的 4 倍),使得按某种测度计算得到的误差  $\delta(w_i(D_i), R_i)$  最小,也就是说,在源图象中找到一个与  $R_i$  最相

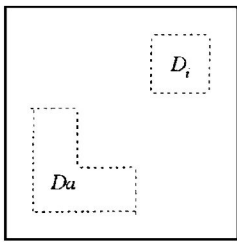


图 1

Fig. 1

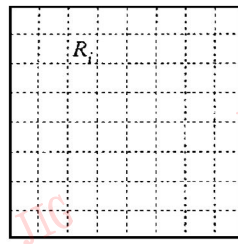


图 2

Fig. 2

似的  $D_i$ ，因为  $D_i$  图象块比  $R_i$  图象块大，这样就保证了变换  $w_i$  的压缩性。这里，变换  $w_i$  具有如下形式：

$$\forall p = \begin{pmatrix} x \\ y \end{pmatrix} \in D_i, w_i(p) = \begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix} p + s$$

其中  $s$  为  $D_i$  的左上角点的坐标。存储得到的迭代函数系统  $W() = \bigcup_{j=1}^{64} w_j()$ ，就完成了对源图象  $f$  的分形压缩。根据压缩映射吸引子定理<sup>[9]</sup>，对任意一初始图象  $f'$ ，不断施加压缩变换  $W$ ，将得到与被编码的源图象近似的不变集  $g = \lim_{n \rightarrow \infty} W^n(f')$ ，从而完成解码。

### 3 覆盖式分形图象压缩

在 Jacquin 的方法中，对图象采用的是规则划分，图象分块的数目决定了所需迭代变换的数目，亦即决定了图象的压缩比。现在考虑这样一种情形，如果对图 2 中某 2 个(或更多)规则拼接分区合并后(如图 3 中的  $R_a$ )，仍能找到一个满足误差要求的最佳匹配扩充分区(如图 1 中的  $D_a$ )，那么可以将该合并得到的拼接分区当作一个分区，从这种意义上来说，就减少了所需迭代变换的数目，从而也提高了分形压缩比。

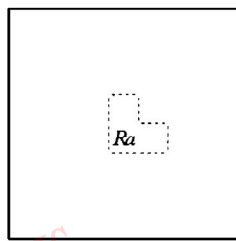


图 3  
Fig. 3

基于上述考虑，我们提出一种覆盖式分形压缩方法，其主要思想是：先以一给定尺寸对图象进行规则划分，而后按扫描线的顺序，对每个规则分区进行四连通扩充操作，在一定的误差门限内，尽可能地合并规则拼接分区。在四连通扩充过程中，如果对某个规则拼接分区找不到合乎误差要求的扩充分区，则将该分区作四叉树划分，将划分得到的 4 个子图象代替该规则分区重新进行四连通扩充操作。在算法中，取拼接分区和相应扩充分区面积比 1:4，误差计算采用最小方差测度。

#### 3.1 压缩算法描述

压缩算法用伪代码描述如下：

给定误差门限  $e_c$ ；

给定初始规则划分尺寸  $c_x, c_y$ ；

按划分尺寸  $c_x, c_y$  对源图象进行规则划分，得到规则拼接分区  $R_{ij}, 0 < i < M, 0 < j < N$ ；

建立规则分区表  $R\_Table$ ，初始包含所有规则拼接分区；

将分区表  $R\_Table$  中每个拼接分区标记为未处理；

while(分区表  $R\_Table$  存在未处理的拼接分区){

    取出未处理拼接分区中排在最前的一个  $R_i$ ；

    以  $R_i$  为输入参数调用拼接分区扩充操作子过程；

    整理处理标志(对分区表  $R\_Table$  中已输出的拼接分区置已处理标志，其它的还原为未处理标志)；

    }

其中，“拼接分区扩充操作子过程”伪代码描述如下：

将输入参数——规则拼接分区  $R_m$ ——压入待处理堆栈；

将  $R_m$  标记为已处理

建立当前合并拼接分区  $R\_Cur$ ，初始为空；

while (待处理堆栈不为空){

    取出栈顶的规则分区  $R$ ；

    将  $R$  加入当前合并拼接分区  $R\_Cur$ ；

    对当前合并拼接分区  $R\_Cur$ ，从所有的可能  $Domain$  分区  $D$  中，按最小方差测度，找到  $R\_Cur$  最佳匹配的分区  $D_i$  以及相应的最佳匹配映射  $W_i$ ，计算匹配误差  $Error$ ；

    if ( $Error < e_c$ ) { /\* 存在合并规则拼接分区的可能 \*/

        将该规则分区  $R$  的四连通相邻规则分区中没有被处理过的规则分区压入待处理堆栈，并标记为已处理；

        while (存在未处理的与  $R$  四连通的相邻规则拼接分区  $R\_Not$ ) {

            将  $R\_Not$  压入处理待堆栈；

            标记  $R\_Not$  为已处理；

        }

    }

else { /\* 当前合并不成立或当前规则分区需进一步细分 \*/

    if (当前合并拼接区  $R\_Cur$  中包含二个或以上规则拼接分区)

        从当前合并拼接区  $R\_Cur$  中删去规则拼接分区  $R$ ；

    else { /\* 当前合并拼接区  $R\_Cur$  中只包含一个规则拼接分区 \*/

对当前合并拼接区  $R\_Cur$  中包含的唯一一个拼接分区  $R\_Sin$  进行二叉树划分, 得到 4 个子拼接分区;  
 从分区表  $R\_Table$  中删去  $R\_Sin$ ;  
 将得到的 4 个子拼接分区加入到分区表  $R\_Table$  中, 标记为未处理;  
 return;

输出当前合并拼接区  $R\_Cur$  压缩码以及分区信息;  
 return;

### 3.2 快速搜索算法

从压缩过程可以看出, 压缩时间主要消耗在寻找最佳匹配扩充分区上。在 *Jacquin* 的算法中, 搜索压缩映射码时采用的是穷尽搜索方法, 压缩所需时间很长。为解决分形压缩中时间代价太大的缺陷, 房育栋等<sup>[10]</sup>提出了一种快速映射码搜索算法。基于一种区域相关的假定, 即若拼接分区  $R_i$  在  $R_j$  的邻域, 则其对应的最佳匹配扩充分区  $D_i$  在  $D_j$  的邻域的概率较大, 该算法在搜索最佳映射时, 将最初的搜索范围定为在以前一次得到的最佳匹配扩充分区为中心, 事先设定窗宽和窗高值的窗口。如果搜索到的最佳映射不满足要求, 扩大搜索窗口, 直到找到合乎要求的最佳映射或者搜索窗口已扩展至整个扩充分区范围。房育栋的快速搜索算法主要是针对拼接分区为规则分区的情形, 然而在我们的算法中得到的合并拼接分区并不规则, 而且随着合并拼接分区的加大, 其快速搜索算法要求的区域相关假定不成立, 因而必须对房育栋的快速搜索算法加以改进。具体做法是: 在搜索最佳匹配扩充分区过程中, 如果当前合并拼接区只包含一个规则分区, 此时采用房育栋的快速搜索算法。如果当前合并拼接区包含多个规则分区, 则按以下步骤进行搜索:

- (1) 将初始搜索范围定为: 当前合并分区的前一次合并结果对应的最佳匹配扩充分区的外接矩形框加上增加步长得到的窗口;
- (2) 在增加的搜索范围内, 搜索最佳匹配扩充分区, 及相应的最佳匹配映射;
- (3) 计算出匹配误差;
- (4) 如果匹配误差小于误差门限, 当前合并成立, 记录最佳匹配结果, 转至(8);
- (5) 如果匹配误差大于误差门限, 将搜索范围以步长  $Step$  加大;

- (6) 如果搜索范围大于整个扩充分区范围, 当前合并不成立, 转至(8);
- (7) 转至(2);
- (8) 搜索结束。

## 4 拼接分区信息的存储

对扩充后的拼接分区可以采用 *Freemann* 提出的经典的方向编码方法<sup>[11]</sup>记录, 这种方法采用方向链码描述将区域外形转换为字符串, 而对方向链码采用跑长编码的方式进行记录。例如若采用图 4 所示的标准 4-方向编码, 则对于图 5 所示的区域边界, 以  $A$  为起点其顺时针跑长编码为

$$Q = \text{ESENESWNWN}, M = 1132161144$$

其中  $Q$  称为方向码字符串,  $M$  称为跑长码字符串。

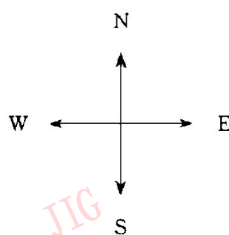


图4  
Fig. 4

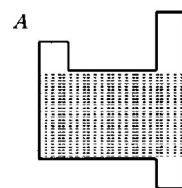
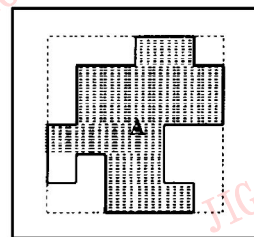


图5  
Fig. 5

事实上, 扩充后的拼接分区是各种各样的多边形, 这些多边形的边是由垂直边和水平边组成。如果对每个拼接分区都要记录其边界链码, 势必要影响到压缩比的提高, 就得考虑记录分区信息所需的存储位与压缩变换数目之间的权衡。这里, 我们提出一种覆盖式的存储方案对扩充后的拼接分区信息进行存储。假设我们得到图象的一个拼接分区, 如图 6 所示的多边形  $A$ , 我们记录其外接矩形框(如图中虚线所示), 显然记录外接矩形框要比记录多边形  $A$  节省许多。将对应多边形  $A$  的变换  $W_A$  范围扩展至其外接矩形框。在解码时为正确恢复该拼接区, 通过适当安排解码顺序, 将对应拼接分区  $A$  的外接矩形框的变换放在与  $A$  相邻的拼接分区之前进行。这样, 虽然经  $W_A$  得到的图象块为  $A$  的外接矩形框, 然而, 随后经变换得到的与  $A$  相邻的拼接分区将会覆盖



A 的外接矩形框的相应部分,最终得到 A 的实际形状。

## 5 结果与结论

在 486-75M-PC 机上,我们用上述方法对  $512 \times 512$  的 Lenna 图象(图 7)进行了压缩试验。图象初始规则划分为  $8 \times 8$  的子图象块,经拼接分区合并后,得到划分结果如图 7 所示。采用穷尽搜索方法,整个压缩过程需要 47 小时,而采用快速搜索算法,所需时间为 16 小时。从压缩代码得到的恢复图象结果如图 8。该压缩方法与基本  $8 \times 8$  的 Jacquin 方法比较的结果见列表 1,从表中可以看出,由于减少了拼接分区数,而且采用一种覆盖的方式降低了记录拼接分区所需的位数,因而压缩比得到明显的提高。

表 1  
Table 1

	Jacquin 方法	本文方法
变换数目	4 096	1 173
分区信息 (bit)	0	4 248
压缩比	19.8:1	41:1
PSNR(dB)	28.57	29.63
所需时间(h)	27	16



图 7 拼接分区

Fig. 7 Range blocks



图 8 恢复图象

Fig. 8 Reconstructed image

## 参考文献

- 1 Mandelbrot B B. The Fractal Geometry of Nature. San Francisco, Freeman, 1982.
- 2 John E. Hutchinson, Fractals and Self Similarity. Indiana University Mathematics Journal, 1981, 35(5)
- 3 Barnsley M F, Sloan A D. Chaotic compression, Computer Graphics World, 1987. 11.
- 4 Barnsley M F, Sloan A D, A better way to compress images, BYTE Magazine, 1988. 6.
- 5 Barnsley M F, Jacquin A E, Application of recurrent iterated function system to images. SPIE Visual Comm. Image Process, 1988, 1001: 122~131.
- 6 Jacquin A E, A Fractal Theory of Iterated Markov Operators with Applications to Digital Image Coding. Doctoral Thesis, Georgia Institute of Technology, 1989.
- 7 Jacquin A E, Fractal image coding: A review, Proceeding of the IEEE 1993, 8(10): 1451~1465.
- 8 Lester Thomas, Farzin Deravi. Region-based fractal image compression using heuristic search. IEEE Transaction On Image Processing, 1995, 4(6): 832~838.
- 9 Yuval Fisher. Fractal image compression, SIGTAPH'92 Course Notes.
- 10 房育栋, 余英林. 快速分形图象压缩编码. 电子学报, 1996, 24(1): 29~33.
- 11 Freemann H. On the encoding of arbitrary geometric configuration, IEEE Trans. EC, EC-10(1961), 260~265.



颜飞翔,博士研究生。主要研究方向:计算机图形学、图形图象编码等。

## A Quick Fractal Image Compression Method in A Region-covered Way

Yan Feixiang, Cai Xuanping, Sun Maoyin

(Dept. 4, room 405, NUDT, Changsha 410073)

**Abstract** The paper presents a new fractal image compression scheme, which merges the range blocks as much as possible in a 4-connected extending way to decrease the amount of the transformations under a given error tolerance, with a quick searching algorithm. Through rearranging the storing order of the transformations according to their edges, we only need to record their surrounding rectangles for those complicated ranges blocks, whose shape can be recovered using a paint-like rendering method in decoding. Our experiment on Lenna is pleasing, the compression ratio reach 41 at a PNSR of 29.63dB.

**Keywords** Fractal, Image compression, IFS

(上接 588 页)

## Image Vector Quantizer based on a Classification in the WT Domain

Sheng Chunming, Fu Ping

(Jilin University of Technology, Changchun 130025)

**Abstracts** A classified VQ (CVQ) technique, based on the edge-oriented classifier, can reduce the edge degradation as well as the encoding complexity. In this paper, according to the WT coefficients feature of a still image transformed, we propose a classification algorithm in the wavelet transform (WT) domain for the CVQ. This approach can be easily extended to the wavelet transform coding technique. Coded images based on this classification are shown to enjoy considerable bpp reduction over three existing methods in the same or a little higher PSNR conditions.

**Keywords** Classification, Vector quantization, Wavelet transform