

改进分形图像编码的叉迹算法

何传江 申小娜

(重庆大学数理学院 重庆 400030)

摘 要 分形图像编码具有好的视觉质量和高压缩比,然而,固有的编码耗时限制了它的应用范围. 编码时间主要花费于在一个海量码本中搜索每个输入子块的最佳匹配块. 针对这个问题,提出了一个限制搜索空间的算法,改进了原叉迹算法. 具体说,它把子块叉迹的定义修改为其规范子块主次对角元绝对值之和,并利用新叉迹把搜索范围限制在初始匹配块的邻域内;同时,引入误差阈值来控制以初始匹配块为中心的搜索邻域大小. 实验显示,与叉迹算法比较,改进算法在编码时间和图像质量方面获得了更好的效果.

关键词 图像压缩;分形;迭代函数系统;分形图像编码;叉迹

中图法分类号 TP391

Improving Cross Trace-Based Algorithm for Fractal Image Coding

HE Chuan-Jiang SHEN Xiao-Na

(College of Mathematics and Physics, Chongqing University, Chongqing 400030)

Abstract The excellent visual quality and high compression rate of fractal image coding have the limited applications due to exhaustive inherent encoding time. The time is mostly spent on searching for the best-matched block to an input range block in a usually large domain pool; therefore, a fast algorithm is proposed to limit the search space in this paper. It is an improvement of the algorithm based on cross trace proposed by this first author; in detail, the newly-defined cross trace of image block (i. e., the sum of absolute intensities of normalized block along the main and sub-principal diagonal pixels), which is a modification of original one, is utilized to confine efficiently the search space to the vicinity of the initial-matched block (i. e., the domain block having the closest new cross trace to that of the input range block being encoded). Moreover, an error threshold is also used to control the search neighborhood around the initial-matched block. The proposed algorithm has been compared with the fast algorithm based on cross-trace experimentally, showing it performs better in terms of encoding time and image quality.

Keywords image compression; fractal; iterated function system; fractal image coding; cross trace

1 引 言

十余年来,分形图像编码以其新颖的编码思想、高压缩比、解码速度快、分辨率无关性等优点受到学术界广泛关注^[1-5]. 目前,人们对它的研究不仅仅限

于图像压缩应用^[5],也广泛渗透于图像处理的其它应用领域,如图像检索^[6]、图像去噪^[7]和数字水印^[8]等等.

分形图像编码是利用自然图像中广泛存在的局部自相似性来减少图像数据冗余的新型编码技术,它突破了以往图像编码的框架,用一个迭代函数系

统(IFS)来描述一幅图像,巧妙地将分形的自相似性应用到图像编码中,取得了很好的压缩效果.尽管高压缩比、分辨率无关性和解码快等优点使得它有很好的发展前景,但是,编码时间长是它的显著缺点,这严重阻碍了它的广泛应用.因此,在保证图像质量的前提下,如何实现快速的编码是分形图像编码技术的一个重要课题^[1].

构造分形码的时间主要花费于在海量码本中寻找每个输入子块的最佳匹配块,尽管全局搜索法能够得到最优结果,但计算成本高,这限制了基于全局搜索法的分形编码的应用范围.一种自然的想法是,如果能按某种方式在码本中尽可能排除不太可能匹配输入子块的子块,或者设法变全搜索为局部搜索,那么就能在保证图像质量的前提下大大减少编码时间.

基于上述想法,文献[3]提出一个基于叉迹特征的快速分形编码算法(以下简称叉迹算法):一方面,利用子块的方差特征在码本中尽可能排除不太可能匹配输入子块的子块;另一方面,利用子块的叉迹特征把全局搜索变为局部搜索,即把搜索范围限制在叉迹意义下的初始匹配块的邻域内.实验表明,该算法能够大大缩短编码时间,同时实现和基于全局搜索的基本分形算法相同或更好的图像质量^[3].

本文拟对叉迹算法进行两方面的改进:(1)修改叉迹的定义,使新叉迹作为匹配特征比原叉迹更优,同时,使新叉迹克服叉迹不适用于 2×2 子块的缺点;(2)针对原算法的搜索邻域不能自适应于输入子块的缺点(所有输入子块都采用相同的固定邻域),提出一个改进的搜索方案,引进误差阈值来控制子块的搜索邻域的大小.实验结果表明,与叉迹算法比较,改进算法在编码时间和图像质量方面获得了更好的效果.

2 叉迹算法

2.1 基本分形算法

在分形编码中,通常把原始图像划分为互不重叠的子块 $\{\mathbf{R}_i\}$ (称为Range块,简称R块,并设大小为 $n \times n$),它们合起来覆盖整幅图像;同时,在同一图像中确定出大小为 $2n \times 2n$ 的另一类子块(称为Domain块,简称D块),允许重叠且不必覆盖整幅图像,它们可以通过移动一个 $2n \times 2n$ 滑动窗口来获得——滑动窗口始于原始图像的左上角,然后在纵横方向上都以同一步长(通常为 n 或 $2n$)并按行扫描方式滑动.为了与R块大小匹配,D块必须收缩

为 $n \times n$ 子块,这通常是经过4-邻域像素值平均来实现的.类似于矢量量化(VQ)编码,这种收缩子块的集合就构成下一阶段匹配搜索所需的码本(记为 Ω).同时,为了改进图像质量,一般还要考虑8个等距变换(4个90度倍数旋转以及水平方向、垂直方向、主次对角线方向对称反射).

编码阶段,对于每个R块 \mathbf{R} ,确定最佳对比度因子 s 和亮度因子 g ,并在码本 Ω 中搜索子块 $\mathbf{D} \in \Omega$ 以及从8个等距变换中选择最佳变换 S ,使得 \mathbf{R} 与 $\hat{\mathbf{R}} = s\mathbf{S}(\mathbf{D}) + g\mathbf{I}$ 的2-范数误差最小(\mathbf{I} 是全1子块).具体说,对于R块 \mathbf{R}_i ,我们求解极小化问题:

$$\min_{\mathbf{D} \in \Omega, 1 \leq k \leq 8} \min_{s, g \in \mathbb{R}, |s| < 1} \|\mathbf{R}_i - (s\mathbf{S}_k \mathbf{D} + g\mathbf{I})\| \quad (1)$$

其中, \mathbb{R} 是实数集, $\|\cdot\|$ 是向量2-范数(子块视为向量),对比度因子约束 $|s| < 1$ 是为了理论上保证解码迭代序列收敛.问题(1)的内层是约束极小化问题,为了减少计算复杂性,通常的做法是先忽略约束 $|s| < 1$,求解极小化问题:

$$E(\mathbf{R}_i, \mathbf{S}_k \mathbf{D}) = \min_{s, g \in \mathbb{R}} \|\mathbf{R}_i - (s\mathbf{S}_k \mathbf{D} + g\mathbf{I})\|$$

(记解为 s_i, g_i)然后对不满足约束的 s_i 按照某种方式作截断处理.接着用全搜索方法求解问题(1)的外层极小化问题:

$$E(\mathbf{R}_i, \mathbf{S}_{k(i)} \mathbf{D}_{m(i)}) = \min_{\mathbf{D} \in \Omega, 1 \leq k \leq 8} E(\mathbf{R}_i, \mathbf{S}_k \mathbf{D}) \quad (2)$$

其中, $m(i), k(i)$ 分别是输入子块 \mathbf{R}_i 对应的最佳匹配D块和最佳等距变换的序号.

需要说明的是,为了压缩的目的,参数 s, g 都必须进行量化.实验表明,搜索前量化优于搜索后量化.因此式(2)中 $E(\mathbf{R}_i, \mathbf{S}_k \mathbf{D})$ 应先改为 $E(\mathbf{R}_i, \mathbf{S}_k \mathbf{D}) = \|\mathbf{R}_i - (\hat{s}_i \mathbf{S}_k \mathbf{D} + \hat{g}_i \mathbf{I})\|$,再求极小化问题(2).其中, \hat{s}_i, \hat{g}_i 是 s_i, g_i 的量化值.

四元组 $(m(i), k(i), \hat{s}_i, \hat{g}_i)$ 就构成输入子块 \mathbf{R}_i 的分形码.全体 \mathbf{R}_i 的分形码和图像的分割信息(如四叉树分割)一起组成原始图像的分形文件,它描述了一个吸引子近似于原始图像的迭代函数系统(IFS).

解码是相对简单的迭代过程,由分形文件描述的IFS迭代作用于任何初始图像来完成.具体地说,以一幅与原始图像相同大小的任意图像为初始图像,根据分形文件,对初始图像进行分割,对于每个输入子块 \mathbf{R}_i ,用 $\hat{\mathbf{R}}_i = \hat{s}_i \mathbf{S}_{k(i)} \mathbf{D}_{m(i)} + \hat{g}_i \mathbf{I}$ 代之.所有输入子块进行一次这样的替代后被视为一次迭代,一般经10次迭代即可.

2.2 叉迹算法

叉迹是文献[3]定义的一种图像子块特征,即子块去均值、单位化后的主次对角线上像素值之和.具

体说,子块 $\mathbf{X}=(x_{i,j})\in\mathbf{R}^{n\times n}$ 的叉迹定义为

$$C_{\text{tr}}(\mathbf{X})=\sum_{i=1}^n(\hat{x}_{i,i}+\hat{x}_{i,n-i+1}) \quad (3)$$

其中, $\hat{x}_{i,j}=(x_{i,j}-\bar{x})/\sqrt{\sum_{i,j}(x_{i,j}-\bar{x})^2}$, \bar{x} 是子块 \mathbf{X} 的亮度均值.

下面的定理是叉迹算法最主要的理论依据,它给出了叉迹特征与匹配误差的关系.

定理 1^[3]. 设 $\mathbf{R}, \mathbf{D} \in \mathfrak{R}^{n \times n}$ (n 为偶自然数), 则有下面的不等式:

$$E(\mathbf{R}, \mathbf{D}) \geq \frac{\sigma_{\mathbf{R}}}{4} \left| |C_{\text{tr}}(\mathbf{R})|^2 - |C_{\text{tr}}(\mathbf{D})|^2 \right| \quad (4)$$

其中, $\sigma_{\mathbf{R}} = \|\mathbf{R} - \bar{r}\mathbf{I}\|/n$ (标准差).

叉迹算法主要是依据定理 1 提出的, 其理论分析的主要内容如下:

设定阈值 $\tau > 0$ 用于块分类, 对于一个 \mathbf{R} 块 \mathbf{R} , 如果 $\sigma_{\mathbf{R}} < \tau$, 那么 \mathbf{R} 属于平滑类, 否则属于非平滑类. 此外, 设定阈值 $\eta > 0$, 定义容许码本 $\Omega_{\eta} = \{\mathbf{D} \in \Omega \mid \sigma_{\mathbf{D}} \geq \eta\}$. 然后, 按下面的方式给容许码本赋序 ($<$ 表示序关系):

$\mathbf{D} < \mathbf{D}' \Leftrightarrow |C_{\text{tr}}(\mathbf{D})| \leq |C_{\text{tr}}(\mathbf{D}')|, \forall \mathbf{D}, \mathbf{D}' \in \Omega_{\eta}$, 并通过引入下标来体现这个序关系, 即

$$\mathbf{D}_i < \mathbf{D}_j \Leftrightarrow i < j, \forall \mathbf{D}_i, \mathbf{D}_j \in \Omega_{\eta}.$$

编码时, 对于平滑块 \mathbf{R} , 直接用其均值版本 $\bar{r}\mathbf{I}$ 代替, 无需进行匹配搜索. 对于非平滑块 \mathbf{R} , 由式(4)知,

$$E(\mathbf{R}, \mathbf{D}) \geq \frac{\tau}{4} \left| |C_{\text{tr}}(\mathbf{R})|^2 - |C_{\text{tr}}(\mathbf{D})|^2 \right| \quad (5)$$

上式表明, 如果 \mathbf{D} 和 \mathbf{R} 叉迹的绝对值相差较大, 那么匹配误差 $E(\mathbf{R}, \mathbf{D})$ 也较大, 从而 \mathbf{D} 不能匹配 \mathbf{R} ; 等价地说, 如果 \mathbf{D} 匹配 \mathbf{R} (即 $E(\mathbf{R}, \mathbf{D})$ 小), 则由式(5)知 $|C_{\text{tr}}(\mathbf{D})|$ 、 $|C_{\text{tr}}(\mathbf{R})|$ 相差也小. 因此, \mathbf{R} 的最佳匹配块 \mathbf{D} 一定属于码本的子集:

$$\{\mathbf{D} \in \Omega: ||C_{\text{tr}}(\mathbf{D})| - |C_{\text{tr}}(\mathbf{R})|| < \epsilon\}, \epsilon > 0,$$

即在叉迹意义下是 \mathbf{R} 的近邻. 因此, 在赋序容许码本中, 使用二分搜索法找出叉迹意义下与输入子块 \mathbf{R} 最接近的子块 \mathbf{D}_m (初始匹配块), 即

$$\mathbf{D}_m = \arg \min_{\mathbf{D} \in \Omega_{\eta}} ||C_{\text{tr}}(\mathbf{D})| - |C_{\text{tr}}(\mathbf{R})|| \quad (6)$$

接着在初始匹配块 \mathbf{D}_m 的 k -邻域

$$N(\mathbf{D}_m, k) = \{\mathbf{D}_i \in \Omega: |i - m| \leq k\}$$

内再进行匹配搜索, 匹配误差最小者即作为 \mathbf{R} 的最佳匹配块.

3 叉迹算法的改进

3.1 叉迹定义的修改

首先介绍一个结果^[4]: 设 $\mathbf{R}, \mathbf{D} \in \mathfrak{R}^{n \times n}$, 极小化

问题 $E(\mathbf{R}, \mathbf{D}) = \min_{s, g \in \mathfrak{R}} \|\mathbf{R} - s\mathbf{D} - g\mathbf{I}\|$ 的解为

$$s = \frac{\langle \mathbf{R} - \bar{r}\mathbf{I}, \mathbf{D} - \bar{d}\mathbf{I} \rangle}{\|\mathbf{D} - \bar{d}\mathbf{I}\|^2}, \quad g = \bar{r} - s\bar{d} \quad (7)$$

且

$$E(\mathbf{R}, \mathbf{D})^2 = \|\mathbf{R} - \bar{r}\mathbf{I}\|^2 - s^2 \|\mathbf{D} - \bar{d}\mathbf{I}\|^2 \quad (8)$$

其中, $\langle \cdot, \cdot \rangle$ 表示欧氏内积, \bar{r}, \bar{d} 分别表示 \mathbf{R}, \mathbf{D} 的亮度均值.

在分形编码中, 每个 \mathbf{R} 块 \mathbf{R} 由其最佳匹配块的亮度变换来替代 (为叙述方便, 不考虑等距变换), 即 $\mathbf{R} = s\mathbf{D} + g\mathbf{I}$. 根据式(7), 这可变为 $\mathbf{R} = s\mathbf{D} + (\bar{r} - s\bar{d})\mathbf{I}$, 从而得到 $\mathbf{R} - \bar{r}\mathbf{I} = s(\mathbf{D} - \bar{d}\mathbf{I})$. 这说明, 如果 \mathbf{D} 是 \mathbf{R} 的最佳匹配块, 那么它们去均值后相差一个对比度因子. 为了选择不依赖于对比度因子的匹配特征, 需要消除对比度因子的影响. 这可以通过对去均值后的子块进行单位化来实现. 此外, 匹配特征的选择应满足两个原则: (1) 选择的像素点应尽量少, 因为 \mathbf{R} 块、 \mathbf{D} 块都要计算特征, 而这些子块的数目是很大的; (2) 选择的点应尽量具有代表性, 因为子块可能含有不同方向的边缘. 尽管没有明确指出, 但是文献[3]引入的叉迹实际上是依据这两个原则来定义的.

显然, 子块去均值单位化以后, 主次对角线上的像素值有正有负, 直接定义它们之和为叉迹, 就会出现因相互抵消使叉迹数值过小的缺点. 尤其是对于 2×2 子块, 不难知其叉迹均为零. 前已指出, 分形编码不仅限于图像压缩应用. 对于用于非压缩目的的分形算法 (如作为数字水印算法的一部分, 例如文献[8]), 完全可能需要把图像分割为 2×2 子块, 此时叉迹算法已不再适用 (因为 2×2 子块的叉迹均为零), 这就限制了叉迹算法的应用范围.

为了解决上述问题, 我们把叉迹的定义修改为子块去均值、单位化后的主次对角线上像素值的绝对值之和, 即把定义式(3)修改为

$$C'_{\text{tr}}(\mathbf{X}) = \sum_{i=1}^n (|\hat{x}_{i,i}| + |\hat{x}_{i,n-i+1}|) \quad (9)$$

为了区分, 上式定义的叉迹称为新叉迹.

上述修改似乎是平凡的, 但是, 叉迹算法依据的理论基础——定理 1 不再成立. 因此, 必须寻求新叉迹与匹配误差的关系.

下面的定理给出了匹配误差与新叉迹的关系.

定理 2. 设 $\mathbf{R}, \mathbf{D} \in \mathfrak{R}^{n \times n}$ (n 为偶自然数), 则有下面的不等式:

$$E(\mathbf{R}, \mathbf{D}) \geq \frac{\sigma_{\mathbf{R}}}{4} |C'_{\text{tr}}(\mathbf{R}) - C'_{\text{tr}}(\mathbf{D})|^2 \quad (10)$$

其中, C'_u 是指由式(9)定义的新叉迹.

证明. 定义子块 $\mathbf{Q}=(q_{i,j}) \in \mathfrak{R}^{n \times n}$ 如下:

$$q_{i,j} = \begin{cases} 1, & j=i, j=n-i+1 \\ 0, & \text{其它} \end{cases}, 1 \leq i, j \leq n,$$

即 \mathbf{Q} 是一个主次对角线上像素值为 1, 其余为 0 的子块. 对于子块 $\mathbf{X}=(x_{i,j})$, 记 $|\mathbf{X}|=(|x_{i,j}|)$. 根据向量 2-范数和新叉迹定义, 我们有

$$\|\mathbf{Q}\|^2 = 2n,$$

$$C'_{tr}(\mathbf{X}) = \sum_{i,j=1}^n q_{i,j} |\hat{x}_{i,j}| = \langle \mathbf{Q}, |\hat{\mathbf{X}}| \rangle.$$

由 Cauchy-Schwarz 不等式及 $\|\hat{\mathbf{R}}\| = \|\hat{\mathbf{D}}\| = 1$, 得到

$$\begin{aligned} |C'_{tr}(\mathbf{R}) - C'_{tr}(\mathbf{D})|^2 &= |\langle \mathbf{Q}, |\hat{\mathbf{R}}| \rangle - \langle \mathbf{Q}, |\hat{\mathbf{D}}| \rangle|^2 \\ &= |\langle \mathbf{Q}, |\hat{\mathbf{R}}| - |\hat{\mathbf{D}}| \rangle|^2 \\ &\leq \|\mathbf{Q}\|^2 \|\hat{\mathbf{R}} - \hat{\mathbf{D}}\|^2 \\ &= 2n \sum_{i,j=1}^n |\hat{r}_{i,j} - \hat{d}_{i,j}|^2 \\ &\leq 2n \sum_{i,j=1}^n |\hat{r}_{i,j} \pm \hat{d}_{i,j}|^2 \\ &= 2n \|\hat{\mathbf{R}} \pm \hat{\mathbf{D}}\|^2 \\ &= 2n (\|\hat{\mathbf{R}}\|^2 \pm 2\langle \hat{\mathbf{R}}, \hat{\mathbf{D}} \rangle + \|\hat{\mathbf{D}}\|^2) \\ &= 4n (1 \pm \langle \hat{\mathbf{R}}, \hat{\mathbf{D}} \rangle). \end{aligned}$$

根据式(8)和式(7), 并结合上式, 我们有

$$\begin{aligned} E(\mathbf{R}, \mathbf{D})^2 &= \|\mathbf{R} - \bar{\mathbf{r}}\mathbf{I}\|^2 - s^2 \|\mathbf{D} - \bar{\mathbf{d}}\mathbf{I}\|^2 \\ &= \|\mathbf{R} - \bar{\mathbf{r}}\mathbf{I}\|^2 - \left(\frac{\langle \mathbf{R} - \bar{\mathbf{r}}\mathbf{I}, \mathbf{D} - \bar{\mathbf{d}}\mathbf{I} \rangle}{\|\mathbf{D} - \bar{\mathbf{d}}\mathbf{I}\|^2} \right)^2 \|\mathbf{D} - \bar{\mathbf{d}}\mathbf{I}\|^2 \\ &= \|\mathbf{R} - \bar{\mathbf{r}}\mathbf{I}\|^2 \left(1 - \frac{|\langle \mathbf{R} - \bar{\mathbf{r}}\mathbf{I}, \mathbf{D} - \bar{\mathbf{d}}\mathbf{I} \rangle|^2}{\|\mathbf{R} - \bar{\mathbf{r}}\mathbf{I}\|^2 \|\mathbf{D} - \bar{\mathbf{d}}\mathbf{I}\|^2} \right) \\ &= n^2 \sigma_{\mathbf{R}}^2 (1 - |\langle \hat{\mathbf{R}}, \hat{\mathbf{D}} \rangle|^2) \\ &= n^2 \sigma_{\mathbf{R}}^2 (1 + \langle \hat{\mathbf{R}}, \hat{\mathbf{D}} \rangle) (1 - \langle \hat{\mathbf{R}}, \hat{\mathbf{D}} \rangle) \\ &\geq n^2 \sigma_{\mathbf{R}}^2 |C'_{tr}(\mathbf{R}) - C'_{tr}(\mathbf{D})|^4 / (4n)^2 \\ &= \frac{1}{16} \sigma_{\mathbf{R}}^2 |C'_{tr}(\mathbf{R}) - C'_{tr}(\mathbf{D})|^4. \end{aligned}$$

不等式(10)由此推出.

证毕.

基于定理 2, 可以得到类似于式(5)的不等式.

即, 对于非平滑块 \mathbf{R} , 得到其匹配误差与新叉迹特征的关系: $E(\mathbf{R}, \mathbf{D}) \geq \tau |C'_{tr}(\mathbf{R}) - C'_{tr}(\mathbf{D})|^2 / 4$. 因此, 基于这个不等式, 我们可以利用子块的新叉迹特征把全局搜索变为局部搜索, 即把搜索范围限制在新叉迹意义下的初始匹配块的邻域内(理论分析与 2.2 节给出的类似, 这里不再重复). 于是, 基于新叉迹的算法只需把原算法步骤中的“叉迹”换成“新叉迹”即可得到.

基于新叉迹的算法克服了叉迹算法不适用于 2×2 图像分割的缺点. 此外, 实验表明, 作为匹配特征, 新叉迹也是优于叉迹的. 我们知道, 叉迹特征在

叉迹算法中起的作用是把全局搜索变为局部搜索, 即把搜索范围限制在叉迹意义下初始匹配块 \mathbf{D}_m 的预设邻域 $N(\mathbf{D}_m, k) = \{\mathbf{D}_i \in \Omega: |i-m| \leq k\}$ 内. 因此, 对于给定的 k , 全局搜索意义下的最佳匹配块(全局匹配块)落入对应的邻域 $N(\mathbf{D}_m, k)$ 里的 R 块越多, 局部匹配块(在邻域 $N(\mathbf{D}_m, k)$ 中找到的匹配块)代替全局匹配块的 R 块就越少, 因局部匹配块代替全局匹配块而产生的误差就越小, 图像质量自然就越高. 对此, 我们对两种叉迹的优劣进行实验对比, 实验对象为 3 幅不同类型的 $512 \times 512 \times 8\text{bit}$ 标准测试图像 Lena、Boat 和 Peppers. 算法采用基于固定块分割的基于(新)叉迹的算法(参数 $\tau=4, \eta=0, k=|\Omega|$ (码本容量)), R 块大小为 4×4 , D 块大小为 8×8 , 生成 D 块池的滑窗步长为 8 像素.

表 1 给出了实验数据. 表中标识“M-N”表示赋序码本 Ω 中满足 $|\Omega| \times M\% \leq |i-m| < |\Omega| \times N\%$ 的 $\mathbf{D}_i \in \Omega$ 组成的集合, 其中, m 为初始匹配块 \mathbf{D}_m 的序号. 例如, 不妨设 $|\Omega| = 100$, 那么标识“10~20”是指集合 $\{\mathbf{D}_i \in \Omega: 10 \leq |i-m| < 20\}$. “非平滑 R 块数目”表示全局匹配块落入集合“M-N”的非平滑 R 块的数目.

表 1 两种叉迹的对比(测试图像: Lena, Boat 和 Peppers)

全局匹配块所属集合(M-N)/%	非平滑 R 块数目	
	叉迹算法	新叉迹算法
0~10	1544	1883
10~20	1304	1588
20~30	1152	1267
30~40	1041	1045
40~50	867	798
50~60	746	661
60~70	697	547
70~80	567	440
80~90	503	352
90~100	433	271
非平滑 R 块总数目	8853	8853

注: 表中数据均为 3 幅图像的实验数据的平均值.

从表 1 可以看出: (1) 就搜索一半的码块就可找到全局最佳匹配块的非平滑 R 块而言, 对于叉迹特征, 这样的 R 块占了近 70%, 而对于新叉迹特征, 这样的 R 块却占了近 75%, 比前者高出 5 个百分点; (2) 就全局最佳匹配块几乎要进行全局搜索才能找到的非平滑 R 块而言, 对于叉迹特征, 这样的 R 块占了近 5%, 而对于新叉迹特征, 这样的 R 块只占了近 3%, 比前者低 2 个百分点. 这些事实说明, 作为匹配特征, 新叉迹确实优于叉迹.

3.2 搜索方案的改进

从 2.2 节知道, 叉迹算法的搜索方案是以叉迹

作为匹配特征,把对容许码本的全局搜索转化为对初始匹配块的邻域搜索,即仅仅在 $N(\mathbf{D}_m, k)$ 内搜索 (k 预先指定),并把搜索得到的局部匹配块当作全局匹配块.从表 1 的实验结果看到,不同非平滑 R 块的全局匹配块在赋序码本中与其初始匹配块的远近是不同的.因此,对于给定的 k ,某些非平滑 R 块的全局匹配块并不在 $N(\mathbf{D}_m, k)$ 内,这就导致了只能用局部匹配块代替全局匹配块,因而得不到最好的图像质量.特别是 k 较小时,局部匹配块代替全局匹配块的 R 块将会更多,图像质量自然得不到保证.

为了克服上述缺点,我们提出一个改进的搜索方案.基本原理如下:可以想见,对于不同的搜索邻域 $N(\mathbf{D}_m, k)$,局部匹配块代替全局匹配块产生的误差是不同的,而且通常会随着 k 的增大而减小.因此,可以设定一个误差阈值,以保证这种误差不至于过大,从而保证一定的解码图像质量.首先在预设邻域 $N(\mathbf{D}_m, k)$ (初始邻域)内搜索局部匹配块,若最小误差小于误差阈值,则局部匹配块就作为最佳匹配块.否则,按一定的步长扩大搜索邻域的范围,继续进行搜索(不包括上次搜索过的码块),直到搜索邻域扩大为整个码本为止.

我们把改进搜索方案归纳如下:

- 1. 设定误差阈值 E 、初始邻域半径 k 和扩域步长 L .
- 2. 按式(6)找出初始匹配块 \mathbf{D}_m . 设置临时变量 t , 并初始化 $t=k$.
- 3. 若 $\min_{\mathbf{D} \in N(\mathbf{D}_m, t), 1 \leq i \leq 8} E(\mathbf{R}, S_i \mathbf{D}) < E$ 或者 $N(\mathbf{D}_m, t) = \Omega_\eta$,

其中, S_i 是等距变换, 则最佳匹配块为

$$\mathbf{D}_b = \arg \min_{\mathbf{D} \in N(\mathbf{D}_m, t), 1 \leq i \leq 8} E(\mathbf{R}, S_i \mathbf{D}).$$

- 4. 否则, 调整 t 为 $t+L$, 转步 3.

3.3 新叉迹算法的描述

我们把改进算法(新叉迹算法)的具体步骤归纳如下:

1. 图像分割与码本构成. 把图像分割成不重叠的 $n \times n$ 子块(R 块, 记为 \mathbf{R}), 以纵横方向步长均为 δ 像素生成尺寸为 $2n \times 2n$ 的 D 块池. 对于每个 D 块, 采用 4-邻域像素值平均得到 $n \times n$ 块, 这样的子块集合构成码本 Ω .

2. 参数初始化. 设定 R 块的标准差阈值 τ 、码块的标准差阈值 η 、误差阈值 E 、初始邻域半径 k 和扩域步长 L .

3. 容许码本定义与赋序. 定义容许码本 $\Omega_\eta = \{\mathbf{D} \in \Omega | \sigma_D \geq \eta\}$, 并按新叉迹 $C'_{tr}(\mathbf{D})$ 大小赋序.

- 4. 对于子块 \mathbf{R} :

- 4.1. 若 $\sigma_R < \tau$, 则直接用 $\overline{\mathbf{r}}$ 代替;

- 4.2. 若 $\sigma_R \geq \tau$, 则按二分法在赋序容许码本中搜索与子块 \mathbf{R} 的新叉迹相差最小的码块 \mathbf{D}_m (初始匹配块), 即 $\mathbf{D}_m = \arg \min_{\mathbf{D} \in \Omega_\eta} |C'_{tr}(\mathbf{D}) - C'_{tr}(\mathbf{R})|$.

- 5. 最佳匹配块的搜索: 按前面描述的改进搜索方案进行.

- 6. 记录分形码 (b, i, \hat{s}, \hat{g}) . 其中, b 和 i 分别是最佳匹配块和最佳等距变换的序号, \hat{s} 和 \hat{g} 分别是参数 s 和 g 的量化值.

- 7. 对于其余子块 \mathbf{R} , 重复步 4~步 6.

4 实验结果

实验对象为 3 幅不同类型的 $512 \times 512 \times 8\text{bit}$ 标准测试图像, 它们是 Lena, Boat 和 Peppers. 实验平台为运行 Windows XP 的 PC(1.7GHz CPU/256MB 内存), 所有算法采用固定块分割, 程序用 C++ Builder 编写. 测试性能参数是峰值信噪比 (PSNR/dB) 和编码时间 (s). 在实验中, 我们选取 R 块大小为 4×4 , D 块大小为 8×8 , 生成 D 块池的滑窗步长为 8 个像素. 此外, 参数 s 和 g 分别按 5b 和 7b 量化. 对参数 s 的截断方案是, 若 $s > 1$, 取 $s = 31/32$; 若 $s < 0$, 取 $s = 0$. 当然, 这并不是最优截断方案. 所有算法都采用这个方案, 因此不会影响对比的公平性.

4.1 参数的实验研究

又迹算法有搜索邻域半径 k 、R 块的分类阈值 τ 和容许码本阈值 η 三个参数, 新叉迹算法除保留这些参数外, 还引进了另外两个参数, 即误差阈值 E 和扩域步长 L .

关于参数 τ 和 η 的取值, 因为文献[3]是基于 256×256 图像的, 所以, 本文对 512×512 图像重新进行了实验. 结果表明: 综合考虑编码时间和重构图像质量, 对于 512×512 图像, 两个参数的较好取值范围仍然为 $\tau \in [3, 5]$ 和 $\eta \in [20, 30]$. 因此, 为了与文献[3]一样, 本文仍然选取 $\tau = 4$ 和 $\eta = 30$ 作为算法的缺省值.

扩域步长 L 对编码时间和图像质量的影响相对较小, 本文不作过多讨论. 以下设定 $L = 5$.

邻域半径 k 和误差阈值 E 对编码性能(时间和质量)的影响都很大, 它们都可以单独控制算法的编码时间和解码图像质量. 算法性能与邻域半径的依赖关系, 文献[3]有详细讨论, 本文不再重复.

为了对算法性能与误差阈值的依赖关系有一个感性的认识, 我们对所选 3 幅图像进行了实验, 结果由表 2 给出(其中, $k = 10$). 从表 2 可以看出: 随着误差阈值 E 的不断增大, 3 幅图像的编码时间不断减少, 同时 PSNR 也不断降低. 这是不难理解的, 因为误差阈值越大, 扩大邻域搜索的可能性降低, 同时局

部最佳匹配块被误当作全局最佳匹配块的机会也增加. 当 $E \leq 3$ 时, 3 幅图像的 $PSNR$ 基本不再变化, 且都超过或接近基本算法的 $PSNR$, 相应的编码时间缩短了 14~26 倍. 当 $3 < E \leq 10$ 时, 对于 3 幅图像, 编码时间迅速减少, 同时 $PSNR$ 的下降并不太明显, 大约 0.1 dB. 当 $10 < E \leq 30$ 时 (E 按 10 增加), 编码时间的减少并不明显, 但 $PSNR$ 的下降却较为显著, 特别是 $E \geq 20$ 时, 尽管 E 按 10 增加, 但编码时间的减少平均不到 0.2 s, $PSNR$ 却平均有近 0.25 dB 的降低. 此外, 上述分析表明, 误差阈值 E 宜取值 $3 < E \leq 10$, 建议值为 10. 以下设定 $E = 10$.

表 2 关于参数E的实验数据(L: Lena, B: Boat, P: Peppers)

性能		图像 L		图像 B		图像 P	
		时间/s	PSNR	时间/s	PSNR	时间/s	PSNR
误差 阈值 E	1	29.93	34.39	47.77	32.90	44.17	33.99
	2	27.91	34.39	46.69	32.90	43.69	33.99
	3	22.13	34.35	41.38	32.89	37.11	33.96
	4	15.22	34.24	33.39	32.82	24.20	33.86
	5	11.78	34.12	26.28	32.72	15.52	33.76
	6	8.50	33.98	20.00	32.60	10.59	33.65
	7	6.69	33.87	15.50	32.50	8.31	33.54
	8	5.49	33.76	12.17	32.36	6.53	33.44
	9	4.55	33.66	9.74	32.23	5.56	33.35
	10	3.92	33.57	7.95	32.08	4.58	33.26
	15	2.45	33.18	3.55	31.60	3.78	32.90
	20	2.19	32.93	2.50	31.22	2.59	32.61
	30	2.11	32.77	2.20	30.95	2.41	32.29
基本算法		720	34.37	661	32.90	721	33.79

最后指出, 由于引入误差阈值的目的仅仅是控制局部匹配块代替全局匹配产生的误差, 以保证这种误差不至于过大(见 3.2 节). 因此, 与叉迹算法一样, 本文仍然把邻域半径作为控制算法编码时间和解码图像质量的主要参数.

4.2 与叉迹算法的比较

3.3 节描述的新叉迹算法对叉迹算法进行了两个修改——叉迹的定义和搜索方案. 因此, 两个算法的比较可以仅仅从以下两方面进行: (1) 修改叉迹定义的效果(实验 1); (2) 新搜索方案的效果(实验 2).

实验 1. 新叉迹的效果.

算法为叉迹算法和对应的基于新叉迹的算法. 图 1 给出了 3 幅图像的“ $PSNR$ -时间”图, 其中, 两个算法的搜索邻域半径 $k = 10, 20, \dots, 100, 150, 200, 300, 400$, 其余参数设值相同 ($\tau = 4, \eta = 30$). 从图 1 可以看出, 对于 3 幅图像, 在相同的搜索邻域内 (Lena, $k \leq 100$; Peppers 和 Boat, $k \leq 300$), 两个算法的编码时间基本相同, 但算法 2 得到了更好的图像质量 ($PSNR$). 这说明, 新叉迹确实提高了算法的编

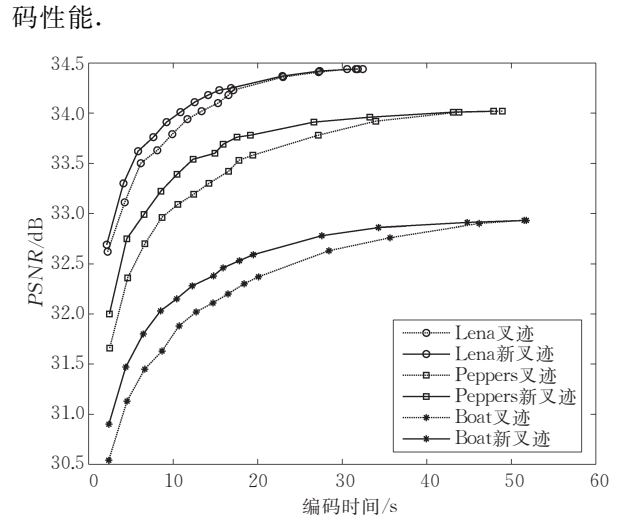


图 1 基于新旧叉迹算法的编码性能对比

实验 2. 新搜索方案的效果.

实验 1 的结果表明, 新叉迹优于叉迹. 因此, 我们在新叉迹的基础上进行新旧搜索方案的比较. 为叙述方便, 我们把叉迹算法中的“叉迹”换成“新叉迹”后的算法称为算法 1, 并把算法 1 的搜索方案换成新搜索方案的算法称为算法 2 ($E = 10, L = 5$).

表 3 给出了 Lena 图像的对比实验结果. 其中, 新搜索方案引进的参数 $E = 10, L = 5$, 两个算法的其余参数取相同值 ($\tau = 4, \eta = 30$).

表 3 新旧搜索方案编码性能对比(测试图像: Lena)

性能		算法 1		算法 2	
		时间/s	PSNR	时间/s	PSNR
邻域 半径 k	10	2.16	32.69	3.88	33.57
	20	4.13	33.30	5.47	33.80
	30	5.86	33.62	6.97	33.95
	40	7.67	33.76	8.44	34.05
	50	9.24	33.91	9.86	34.13
	60	10.88	34.01	11.25	34.18
	70	12.55	34.11	12.64	34.21

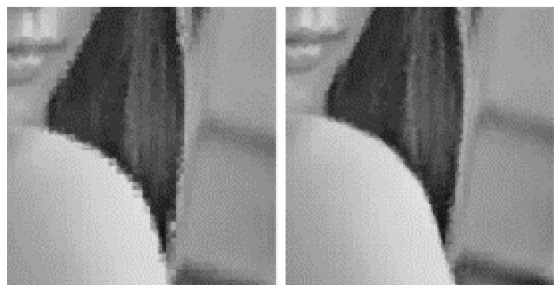
从表 3 看出:

(1) 在相同的搜索邻域内 ($k \leq 70$), 尽管算法 2 的编码时间长一些, 但得到了更好的图像质量. 如果在大致相同的编码时间下比较本文图像质量, 算法 2 显然优于算法 1. 例如, 如果把算法 2 在 k 时的数据与算法 1 在 $k + 10$ 时的数据进行比较, 那么算法 2 在更短的编码时间内得到更好的图像质量. 这说明, 新搜索方案确实能够提高算法的编码性能.

(2) 随着 k 的不断增大, 两个算法的编码性能趋于相同. 不难理解其原因: 正如 3.2 节所指出的, 局部匹配块代替全局匹配块产生的误差通常会随着 k 的增大而减小. 因此, 当 k 足够大时, 这种误差就

会小于误差阈值,误差阈值就失去作用了.这个事实说明,新搜索方案仅仅对初始邻域不太大时才有明显效果.

最后指出,新叉迹算法的编码时间主要都是由邻域半径 k 控制的, k 是可调的,可通过选取较小的 k 值来实现快速的编码,例如 $k=0$.但是,采用原搜索方案的新叉迹算法在 $k=0$ 时的图像质量并不太好.图 2(a)给出了这个算法在 $k=0$ 时的 Lena 解码图像的一部分,尽管编码时间只有 0.20s,但是图像质量并不太好,因为可以明显看出图中边缘部分(如肩膀边缘处等)有“锯齿”效应,说明含有边缘的 R 块直接用初始匹配块代替全局匹配块会产生较大误差,必须限制这种误差不至于过大.显然,采用新搜索方案可以做到这一点.只要设定的误差阈值大小适中,完全可以在较短的编码时间内保证一定的解码图像质量.图 2(b)给出了采用新搜索方案的新叉迹算法(误差阈值 $E=30$)在 $k=0$ 时的 Lena 解码图像的一部分,编码时间只有 0.22s,同时图像质量得到极大的改善,图中边缘部分(如肩膀边缘处)的“锯齿”效应得到较大改善.因此,在实时性要求高、质量次之的应用场合,基于新搜索方案的新叉迹算法显然更优.



(a) 原搜索方案

(b) 新搜索方案

图 2 采用两种搜索方案的新叉迹算法的解码图($k=0$)

5 结 论

本文对文献[3]提出的叉迹算法进行了两个方

面的改进:(1)叉迹的定义.叉迹的定义修改为子块去均值、单位化后的主次对角线上像素值的绝对值之和.新叉迹不仅克服了原叉迹不适用于 2×2 图像分割的缺点,而且是比叉迹更好的匹配特征;(2)搜索方案.引进误差阈值来控制匹配误差的大小,从而在初始邻域较小时能够保证一定的解码图像质量.实验结果显示,与叉迹算法比较,改进算法在编码时间和图像质量方面获得了更好的效果.

参 考 文 献

- [1] Wohlberg B, Jager G. A review of the fractal image coding literature. *IEEE Transactions on Image Processing*, 1999, 8 (12): 1716-1729
- [2] He C, Yang S X, Xu X. Fast fractal image compression based on one-norm of normalised block. *IEE Electronics Letters*, 2004, 40(17): 1052-1053
- [3] He Chuan-Jiang, Huang Xi-Yue. Fast fractal image coding based on local cross trace. *Chinese Journal of Computers*, 2005, 28(10): 1753-1759(in Chinese)
(何传江,黄席樾.基于图像块叉迹的快速分形图像编码算法. *计算机学报*, 2005, 28(10): 1753-1759)
- [4] He C, Xu X, Yang J. Fast fractal image encoding using one-norm of normalized block. *Chaos, Solitons & Fractals*, 2006, 27(5): 1178-1186
- [5] Distasi R, Nappi M, Riccio D. A range/domain approximation error-based approach for fractal image compression. *IEEE Transactions on Image Processing*, 2006, 15(1): 89-97
- [6] Distasi R, Nappi M, Tucci M. FIRE: Fractal indexing with robust extensions for image databases. *IEEE Transactions on Image Processing*, 2003, 12(3): 373-384
- [7] Ghazel M, Freeman George H, Vrsnay E. Fractal image denoising. *IEEE Transactions on Image Processing*, 2003, 12 (12): 1560-1578
- [8] Pi M H, Li C, Li H. A novel fractal image watermarking. *IEEE Transactions on Multimedia*, 2006, 8(3): 488-499



HE Chuan-Jiang, born in 1964, Ph.D., professor, Ph.D. supervisor. His current research interests include fractal, partial differential equations and image processing, etc.

SHEN Xiao-Na, born in 1982, M. S. candidate. His current research interests include fractal, partial differential equations and image processing.

Background

Over the past decade, fractal image coding has not only been attracting extensive attention of the researchers in the field of image compression, but also received a lot of interests from the researchers in other areas of image processing. The technique can provide a high decoded image quality at a high compression ratio; however, the problem with fractal coding is the highly computational complexity in the encoding process. The time to fractally encode an image is mostly spent on finding the best-matched block to an input range block in a large domain pool. Therefore, focal aims of fractal image coding are to accelerate the encoding process, in which many efforts have been attempted to accelerate the encoding process since the early 1990s. However, most of the existing accelerating schemes reduce significantly the required computation only at the expense of more or less degradation of the reconstructed image quality. The proposed algorithm, which is an improvement of the fractal algorithm based on cross trace proposed by this first author [He C, Huang X. Fast fractal image coding based on local cross trace. Chinese Journal of Computers, 2005, No. 10], can considerably shorten

the encoding time, while achieving the same or better decoded image quality as baseline fractal algorithm with full search.

The major research interests of this group are fractal image coding, PDE-based image processing, and so on. The topic in this paper is primarily a part of Ph.D. dissertation, entitled "Study on Algorithms for Fractal Image Coding Technology", by the first author. In this Ph.D. dissertation, the author has explored many aspects of fractal image coding, and then improved some existing fractal encoding/decoding algorithms and proposed several new fractal encoding/decoding algorithms. Most of these algorithms have been published, respectively, in IEE Electronics Letters (2004, No. 2, No. 17), IEE Proc.-Vision Image and Signal Processing (2004, No. 3), Chaos, Solitons & Fractals (2006, No. 5; 2007, No. 4), Acta Electronica Sinica (2004, No. 11), Chinese Journal of Computers (2005, No. 10) and Journal of Image and Graphics (2004, No. 9; 2005, No. 5; 2006, No. 12), etc.