图像编码 块状效应 算法 分形图像编码(15)

57-58,65

计算机科学 1999Vol.26№.4

一种消除分形图像编码中块状效应的算法*)

An Algorithm for Removing Blocking Artifacts in Fractal Image Coding

狄红卫 余英林

(华南理工大学电子与通信工程系 广州 510641)

TN919.8 TP391.41

Abstract Fractal image coding can lead to noticeable blocking artifacts-artificial rectangular discontinuities in the decoded image. In this paper, a fast, easy-to-implement algorithm that removes blockiness by performing a simple nonlinear smoothing of pixels without destroying key image information is developed. This approach achieves subjective and objective gains and improves visual quality.

Keywords Blocking artifacts, Fractal, Image coding

1 引言

自80年代中后期,Barnsley等人提出分形图像压缩的概念[1]以来,分形图像编码作为一种新的具有高压缩比潜力的图像编码方法,越来越受到广泛的关注[2,3]。分形图像编码的数学基础是迭代函数系统(IFS)和拼贴定理[4]。基于迭代函数系统的基本的自动分形图像编码首先将原始图像分割成不相互重叠的小方块,然后对各个小方块构造迭代函数系统,即对每个小方块分别编码[2]。然而,当图像的压缩比提高时,这种基于块的分形图像编码方法会引起图像的信噪比下降,并且一般来说,图像的主观质量也随之降低。主要表现为恢复图像中相邻图像块的边界衔接不连续,即通常所说的存在块状效应。

对于块状效应,有不少文献提出了后处理算法进行改进[5,6]。但是,这些算法一般都是针对基于DCT的图像编码,而它的编码误差来源于量化,因而有先验条件可以利用。本文试图在分析分形图像编码误差的基础上,提出一种消除分形压缩中块状效应的方法,以改善恢复图像的质量。

2 分形图像编码中的块状效应

分形图像编码中,块状效应主要是两方面因素造成的。首先,由于在分形压缩编码中采用的是互不交叠的图像块分割方法,各个图像块独立地进行编码,并在均方误差测度下寻找最佳匹配块,而均方误差准则是一种块平均的匹配误差,图像块边界的匹配误差可能较大而整个块的匹配误差却较小,因而使得相邻块的边界部分产生不连续,造成了视觉上的块状效应。其次,由于各种块分类的方法被应用于分形压缩来降低编码复杂度和提高压缩比,但是块分类得到的阴影块在重建时只是用它的灰度均值来恢复块中的每一像素点的灰度,这样原本细微的灰度变化被完全掩盖,在图像的边界也会产生不连续,形成了边缘。实际上,在分形编码中,块状效应产生还有其它的原因,可以说它是由多种误差因素综合作用的结果,不像基于DCT的图像编码仅由量化误差引起。

在不考虑实现的细节和复杂性的情况下,我们认为去除块状效应的算法受这样两个简单而又基本的原则控制:1)平滑块间的人工不连续能提高图像的质量。2)平滑实际图像的边缘会降低图像的质量。我们的方法尽可能简单、直接地满足这两个原则。

如果不能取得一个好的编码误差的估计,要想在解码端成功地消除块状效应,几乎是不可能的。下面我们来形成一个误差的估计以区分自然边界和人工边界。假定能从编码器得到原始图像的方差 T_1(编码结果增加一个 T_1 对压缩比几乎没有什么影

*)广东省自然科学基金资助课题。狄红卫 讲师,博士生,主要研究方向:图像编码与处理,分形及小波变换。余英林 教授,博士生导师,主要研究方向:图像与图形处理,模式识别,神经网络,进化算法,模糊技术等。

响)。解码后重建图像的方差为 T_2，可以认为平均编码误差 t' 为：

$$T=T_1-T_2 \tag{1}$$

通过多幅图像的实验，我们认为这个估计是合理的。

3 消除块间人工不连续

我们这里提出的消除分形编码中块状效应的方法是，减小块边缘的不连续到视觉系统不能察觉的程度 u，这种不连续我们认为是由人工边界引起的。根据 Weber 定律[7]，u 可以近似为：

$$u=cg \tag{2}$$

其中 g 为图像的平均亮度，$|c|<0.08$。

边界不连续可以简单地通过每一个块边界像素的差分得到。一幅 $N\times N$ 的数字图像 f 可以看成是一个 R^{N^2} 空间的向量 $N^2\times 1$，按列的顺序堆叠可表示为：

$$f=[f_1,f_2,\cdots,f_N]^T$$

其中 f_i 表示图像的第 i 列像素。

定义算子 Q，Qf 给出图像 f 的块边界相邻列像素的差，以向量 d_{col} 表示。例如对于 8×8 的分块

$$d_{col}=Qf=[(f_8-f_9),(f_{16}-f_{17}),\cdots,(f_{N-8}-f_{N-9})]^T \tag{3}$$

同样，可以得到图像 f 的块边界相邻行像素的差向量 d_{row}。

一旦计算出块边界不连续，我们就可以减小这种不连续到可视阈值以下。但是，我们并不能平滑所有的不连续到阈值以下，而只能平滑那些我们认为是由于编码误差引起的不连续，特别是对于不连续值小于 $t=2\sqrt{T}$ 的情况。从(1)式可知，$\sqrt{T}$ 是每个像素平均误差的估计。这样，由于编码误差引起的块状效应的上限 $t=2\sqrt{T}$，发生在当相邻块边界像素的误差朝相反方向变化的时候。

若 x 和 y 表示两个块相邻像素的值，那么 $d=x-y$ 就相应于 d_{col} 或 d_{row} 中的一个元素。当 $|d|\leqslant 3t$ 时，更新边界像素，$x'=x-\lambda d$，$y'=y+\lambda d$，其中 x'，y' 是更新后的像素值，$\lambda=(t-u)/2t$。比例常数 λ 的选取使得当 $|d|\leqslant t$ 时，边界不连续减小到小于或等于阈值 u。$|d|=t$ 时，更新后边界不连续减小到等于阈值 u；$|d|<t$ 时，更新后边界不连续减小到小于阈值 u。模拟的结果表明，这种按比例的平滑在均方误差意义下效果良好。对于 $|d|>3t$，我们用一确定量 λt 来平滑每一像素，即当 $d>3t$ 时，$x'=x-\lambda t$，$y'=y+\lambda t$；当 $d<-3t$ 时，$x'=x+\lambda t$，$y'=y-\lambda t$。这样处理是基于下面两种考虑：一是即使 t 设置得太小(因为 t 是平均误差估计，而不是最大误差估计)，也可以消除大部分的块状效应；二是考虑到实际图像的边缘，不能过分平滑大的不连续值。

上述算法的步骤可以总结如下：

1)设置 $t=2\sqrt{T}$，其中 T 由(1)决定。

2)运用(3)得到水平和垂直方向块边界的差向量 d_{col} 和 d_{row}。

3)设置一比例常数 $\lambda=(t-u)/2t$，其中 u 是与图像的像素平均值相关的可视阈值。

4)对于 d_{col} 和 d_{row} 中的元素 $d=x-y$，通过下式平滑相应像素。

$$x'=\begin{cases} x-\lambda d & |d|\leqslant 3t \\ x-\lambda t & d>3t \\ x+\lambda t & d<-3t \end{cases} \tag{4}$$

$$y'=\begin{cases} y-\lambda d & |d|\leqslant 3t \\ y-\lambda t & d>3t \\ y+\lambda t & d<-3t \end{cases}$$

对于位于每个块角落的像素作了两次更新，为了不过分平滑这些像素，我们采用对两次更新的结果作平均。另外，边界像素的平滑会导致边界点与其相邻的内部点的新的不连续。我们通过二者的平均来降低这种影响，这一步骤可以迭代直到块的中心。

4 实验结果与结论

依照上述方法，我们对多幅图像进行了实验。表 1 给出了 512×512×8 的 Lena 和 Peppers 灰度图像在不同压缩比下用我们的算法处理的结果，这里 c 取 0.06～0.07。从表中可以看出重建图像的 PSNR 提高了 0.3～0.5dB。图 1 是位率 0.106bpp 时的 Lena 重建图像，(a)为分形解码图像，(b)为运用本文方法到(a)上的结果。我们看到，经过后处理后，块状效应得到极大消除，图像的视觉质量得到很大的改善。也就是说，处理后图像的客观和主观质量都有了提高。另外，本文提出的算法计算复杂度小，这对于实现实时的后处理有重要意义。

表 1 不同位率下实验结果比较(PSNR)

重建图像		位率(bpp)		
		0.106	0.390	0.422
Lena	处理前	26.02dB	28.75dB	29.91dB
	处理后	26.33dB	29.24dB	30.33dB
Peppers	处理前	25.80dB	28.73dB	29.88dB
	处理后	26.19dB	29.23dB	30.28dB

(下转第 65 页)

注：表 1 和表 2 中的样本由以下混沌 kent 序列生成：

$$x_{n+1}=\begin{cases} x_n/a & 0\leqslant x_n\leqslant a \\ (1-x_n)/(1-a) & a\leqslant x_n\leqslant 1 \end{cases}$$

其中：$a=0.2$，取初值 $x_1=0.3$，表 1 和表 2 的样本对分别为：

$(x_i,x_{i+1},x_{i+2})\to x_{i+3}$，$(x_i,\cdots,x_{i+299})\to(x_{i+300},x_{i+301})$；$i=1,2,\cdots$

表 1 表 2

样本个数 k		100	150	200	250	300	350	50	100	150	200	250	300
新算法	精度 J	1.9 $\times10^{-23}$	2.8 $\times10^{-23}$	1.6 $\times10^{-22}$	5.3 $\times10^{-21}$	6.5 $\times10^{-20}$	8.3 $\times10^{-20}$	1.48 $\times10^{-28}$	9.75 $\times10^{-28}$	4.21 $\times10^{-26}$	9.26 $\times10^{-24}$	1.09 $\times10^{-24}$	1.18 $\times10^{-24}$
	耗时（秒）	4	8	16	23	35	46	3	8	15	30	48	68
BP 算法	精度 J_{BP}	0.99	1.49	1.98	2.50	3.00	2.48	0.0012	0.007	0.045	3.75	5.54	8.86
	耗时（分）	0.62	2.83	6.68	13.60	26.10	46.13	6.25	8.70	11.11	16.67	20	23.53

参考文献

1 Rumelhart D E, et al. Learning Representation by Back-Propagation Errors. Nature, 1986, 533～536

2 Haykin S. Neural Networks. Macmillan College Publishing Company, 1994. 190～192

3 Baldi P. Gradient Descent Learning Algorithm Overview: Greneral Dynamical Systems Perspective. IEEE Trans Nenral Networks, 1995, 6(1). 182～195

4 张立明．人工神经网络的模型及应用．复旦大学出版社，1992. 44～46

5 焦李成．神经网络系统理论．西安电子科技大学出版社，1990. 36

6 Zhang Daiyuan, Yu Juebang. An Algebraic Algorithm for 3-Layered Perceptron Training with δ Activation Function in Hidden Layer. In: Huang Xinhan, et al. eds. The Intl. Conf. on Artificial Intelligence for Engineering. 1998. 310～312

（上接第 58 页）

(a) (b)

图 1 位率 0.106bpp 时的 Lena 重建图像

参考文献

1 Barnsley M F, Sloan A D. A better way to compress images. BYTE Magazine, 1988, 13(1): 215～223

2 Jacquin A E. Image coding based on a fractal theory of iterated contractive image transformations. IEEE Trans on Image Processing, 1992, IP-1(1): 18～30

3 Fisher Y. Fractal Image Compression. New York. Springer-Verlage, 1995

4 Barnsley M F, Hurd L P. Fractal Image Compression. Wellesley: AK Peters, 1992

5 Yang Yangyi, et al. Projection-based spatially adaptive reconstruction of block-transform compressed Images. IEEE Trans on Image Processing, 1995, 4(7): 896～908

6 Chou J, et al. A simple algorithm for removing blocking artifacts in block-transform coded images. IEEE Signal Processing Letters, 1998, 5(2): 33～35

7 Jain A. Fundamentals of Digital Processing. New York. Van Nostrand Reinhold, 1993