

文章编号: 1002-2082(2007)04-0388-04

基于SPIHT的改进图像压缩算法

赵米暘, 陈卫东, 卢晓燕

(西安应用光学研究所, 陕西 西安 710065)

摘要: 针对渐进式图像传输算法都受限于庞大的内存空间和计算复杂度, 提出一种基于SPIHT的改进型算法——静止图像编码方法, 即对变换后的小波系数高频区块进行细分, 并对不同频率图像块分别设置阈值, 采取新的阈值判别策略, 减少了链表的结点数。使用MATLAB 6.5开发环境对上述改进编码方法进行仿真。仿真结果表明: 通过对原算法构架进行改进, 减少了内存空间占用, 降低了计算复杂度, 取得了较好的压缩效果。

关键词: SPIHT; 图像压缩; 小波变换; 阈值

中图分类号: TN911.73-32

文献标志码: A

Improvement of image compressing algorithm based on SPIHT

ZHAO Mi-yang, CHEN Wei-dong, LU Xiao-yan

(Xi'an Institute of Applied Optics, Xi'an 710065, China)

Abstract: To overcome the limitation of the internal memory space and calculation complexity for the asymptotic image transmission algorithms, an improved algorithm, the static image coding method, is proposed. The algorithm carries out the high-frequency section subdivision of the wavelet coefficients after the wavelet transform, sets the thresholds for the image blocks of different frequency respectively, adopts a new discriminating criterion, and improves the framework of the algorithm. This concept is adopted in JPEG2000. The improved coding method is simulated with MATLAB 6.5. The result shows that the occupation of the internal memory space is reduced and the computational complexity is reduced, and the compression effect meets the expectation.

Key words: SPIHT; image compression; wavelet transform; threshold

引言

在基于小波变换的图像压缩算法中, 零树小波编码(EZW)是一种非常有效而且实现简单的方法。其输出的码流具有嵌入式特征, 在解码端解码时可以在任意位置停止码流的接收, 因而得到广泛的关注。近年来, 在嵌入式零树编码(EZW)算法的基础上出现了许多新的改进算法, SPIHT(多级树集合分裂算法)就是其中最为典型的一个算法。本文针对SPIHT算法占用内存空间较大且重复扫描探测次

数过多的缺点, 对其构架进行改进, 即通过对高频区块细分并分别设置阈值, 采用相应的阈值判断策略, 降低扫描次数和内存占用, 提高压缩效率。

1 SPIHT的基本原理

SPIHT(set partitioning in hierarchical tree)主要是利用渐进式传输的理论进行编码。渐进式传输理论是将数值的绝对值由大到小排列, 然后将最重要的数值先传输, 还原时图像的恢复质量将渐渐

收稿日期: 2006-11-19; 修回日期: 2007-02-04

作者简介: 赵米暘(1982—), 女, 陕西西安人, 硕士研究生, 主要从事图像处理等研究。E-mail: zmy221@hotmail.com

变好。图像在做小波变换后,其系数特性如下:位于图像左上角的系数最少但是最为重要,图像的大部分能量都集中在最低精度的子图像里,并且各子图像的小波系数间存在着空间自相似性,这一点比幅值顺序在图像编码中更为重要。例如,若较低精度的子图像(如 LH_3)的某一系数相对某阈值 T 为不重要系数,则系数 $LH_2(2x, 2y)$, $LH_2(2x+1, 2y)$, $LH_2(2x, 2y+1)$ 和 $LH_2(2x+1, 2y+1)$ 相对某阈值 T 为不重要系数,其概率为98.2%。基于这个特点,有人提出了一种新的数据结构:零树结构,也称为空间向量树结构,如图1所示。

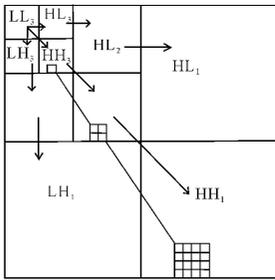


图1 零树结构示意图

Fig. 1 Schematic diagram zero tree structure

由图1可知,这种数据结构是一种金字塔结构,精度高的子图像对应于金字塔的底层,精度低的子图像对应于金字塔的高层。位于高层的每个像素点在低层对应方向均拥有4个后代系数,这4个后代在其低层的相应方向也会有4个孩子存在,这样就构成了一种家族关系,如图1中箭头所示。位于最低层的像素点和位于最高层的根节点的低频部分没有后代。SPIHT编码算法^[1]如下。

SPIHT将小波系数用零树的形式组织起来,其特点是,如果父节点的系数值小于量化门限值 T ,那么该树的所有节点系数都小于 T 的概率很大(由小波系数在不同分解级的分布特性决定);如果该事实成立,只需用1 bit(0)表示这些节点在当前编码平面的值,就可达到压缩效果。SPIHT算法定义了以下3个链表:

- 1) LIS(list of insignificant sets 次要像素集合链表)
 - 2) LSP(list of significant pixels 重要像素链表)
 - 3) LIP(list of insignificant pixels 次要像素链表)
- R 表示小波分解后的所有顶层节点的坐标

集合;

• $D(i, j)$ 表示节点 (i, j) 的所有后代节点的坐标集合;

• $O(i, j)$ 表示节点 (i, j) 的所有儿子节点的坐标集合;例如,若节点 (i, j) 不在空间定位树中的最高和最低层,则 $O(i, j) = \{(2i, 2j), (2i+1, 2j), (2i, 2j+1), (2i+1, 2j+1)\}$

• $L(i, j) = D(i, j) - O(i, j)$ 表示节点 (i, j) 所有儿子节点的后代节点的坐标集合,即 $L(i, j) = \bigcup_{(k,l) \in O(i,j)} D(i, j)$

其简要编码步骤如下。

1) 初始化过程。确定初始量化门限 T , $n = \lceil \log_2 c_{\max} \rceil$ (其中 c_{\max} 为最大小波系数),门限值 $T_0 = 2^n$ 。初始化LSP, LIP和LIS的过程是将LSP置空,把 R 中所有节点 (i, j) 放入LIP,并且将其后代集合放入LIS。

2) 对所有LSP中的节点 (i, j) ,输出 $|c_{i,j}|$ 的第 n 位比特值。

3) 对所有LIP中的节点 (i, j) ,输出 $S_n(i, j)$,若 $S_n(i, j) = 1$,则输出 $c_{i,j}$ 的符号,并将其移到LSP中。

对所有LIS中的集合 (i, j) ,分2种情况:

(1) 如果是A型集合,即 $D(i, j)$,则输出 $S_n(D(i, j))$;若 $S_n(D(i, j)) = 1$,则

①对所有 $O(i, j)$ 中的节点 (k, l) ,输出 $S_n(k, l)$;如果 $S_n(k, l) = 1$,则输出 $c_{k,l}$ 的符号,并将其移到LSP中;否则,将其置入LIP。

②删除这个集合。如果其对应的 $L(i, j)$ 非空,则将 $L(i, j)$ 置入LIS,并标记为B型集合。

(2) 如果是B型集合,即 $L(i, j)$,输出 $S_n(L(i, j))$;若 $S_n(L(i, j)) = 1$,则删除这个集合,并将所有 $D(k, l), (k, l) \in O(i, j)$ 置入LIS,并标记为A型集合。

4) 如果 $n = 0$,则终止;否则 $n = n - 1$,并转到2)继续执行。

由于该算法中搜索次数多,花费时间长,同时,必须建立LIS, LIP和LSP 3个链表存储系数坐标,占用内存较大。比如,假设图像大小为 $N \times N$,则LIP和LSP各为 $N \times N \times \lg_2(N \times N)$ bits, LIS所需内存大小为 $(N \times N / 4) \times \lg_2(N \times N / 4) \times 2$ bits。

2 设计原理

针对SPIHT算法的不足之处,笔者提出的改进方法更适于硬件设计,同时可减少内存占用。具

体设计如下。

在变换前对图像进行直流平移^[4]。图像的直流平移用于图像采样点是在无符号数的情况下,将图像采样点的无符号描述转换为以零为中心的互补描述。直流平移在小波变换之前进行,并不影响图像采样点的数值分布。其平移公式为

$$I(x,y) - 2^{p-1} \rightarrow I'(x,y)$$

式中: $I(x,y)$ 是采样点的对应灰度值; $I'(x,y)$ 是直流平移后的采样点对应灰度值; p 为图像像素位精度。直流平移后图像灰度数据以零或接近零的数值为中心进行分布,整体数据动态范围变小,经小波变换后得到的小波系数均方差也变小,因此可以用较少的码字进行编码。具体编码流程如下。

1) 编码顺序按照Z字型扫描顺序,如图2所示。

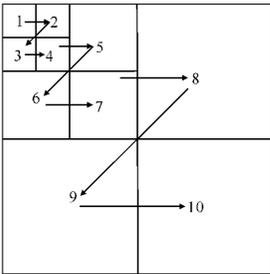


图2 Z字型扫描顺序

Fig. 2 Zigzag scan order

2) 将变换后的高频子带划分为多个区块,编码时首先对这些小区块进行扫描。若扫描到该区块内系数大于当前门限值,则对此区块进行编码。此方法主要是为了减少编码时过多“0”的出现,如图3所示。

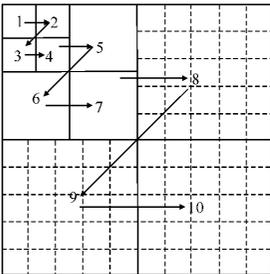


图3 高频小区块划分图

Fig. 3 High frequency block partition

3) 在图3中标号为1~10的不同大小的区块代表了小波系数的高低频,区块1代表低频系数,幅值最大;区块10代表高频系数,幅值最小。其他区块依此类推。根据系数的大小分布不同,我们将

每个区块分别设定各自的最大门限值 T_m 。对不同频率区块分别计算,其中绝对值最大的像素幅值 $f_{m \max}$,令 $B_m = \lceil \lg_2(f_{m \max}) \rceil$,取 $T_m = 2^{B_m}$, $m = 1, 2, \dots, 10$ 。

4) 第1次扫描时就设定当前的门限值优先,当区块内设定的门限值 T_m 小于目前所扫描的门限值,则不将此区块进行编码;当区块内所设定的门限值大于当前扫描的门限值,则进行编码,直到当前门限值扫描全部图像后,再将门限值递减。

5) 若扫描到的区块有切割的功能,则首先对内部的小区块进行扫描,若大于当前门限值,则对此小区块进行编码;若小于当前门限值,则只需对这个小区块编出1个“0”。

6) 当系数达到编码条件时,必须记录此系数已编码。

7) 编码规则是,首先从内存中搜索此系数是否已编码,若未编码,则对此系数做符号编码,然后再做系数编码。对系数编码时,若幅值大于门限值,则输出“1”,若小于则输出“0”。

为了比较2种编码方式的优点,我们在相同的比特率下比较编码后的误差值范围(绝对误差值),如图4、图5和图6所示。我们选定图中左上1/4范围进行比较发现,图5的SPIHT编码方式误差值是193,图6的编码方式误差值是107,可以明显看

63	-34	49	10	7	13	-12	7
-31	23	14	-23	3	4	6	-1
15	14	3	-12	5	-7	3	9
-9	-7	-14	3	4	-2	3	2
-5	9	-1	47	4	6	-2	2
3	0	-3	2	3	-2	0	4
2	-3	6	-4	6	6	3	6
5	11	5	6	0	3	-4	4

图4 原始小波系数

Fig. 4 Original wavelet coefficient

50	-40	56	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

图5 SPIHT小波系数

Fig. 5 SPIHT wavelet coefficient

出在相同比特率下改进编码方式可以降低误差值,提高图像品质。

50	-32	12	8	0	0	0	0
-24	16	8	-8	0	0	0	0
8	0	8	-8	0	0	0	0
-8	0	-8	8	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

图6 M-SPIHT小波系数

Fig. 6 M-SPIHT wavelet coefficient

3 算法仿真

我们使用MATLAB6.5开发环境对上述改进编码方法进行仿真。将使用本算法压缩图像的PSNR值,在相同码率下与传统的SPIHT算法及EBCOT等方法进行了比较。EBCOT(最优截断嵌入式块编码)是JPEG2000的核心算法,其基本思想是将小波变换后的子带分成编码块,并对每个编码块独立地执行位平面编码,形成压缩后率失真优化(PCRD-opt)算法。该算法具有较高的压缩效率,产生的码流具有分辨率可伸缩,随机访问和处理等优点,但也造成了算法复杂度增加,计算速度降低,难以实现等问题。采用改进的SPIHT算法结合算术编码可以达到与之近似的性能,是解决此问题的一种有效方法。

仿真模拟流程图如图7所示。

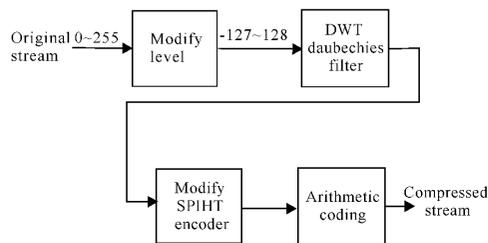


图7 仿真系统模拟流程图

Fig. 7 Flow chart of simulation system

图7采用Daubechies(9-7)小波基,并将算数编码作为可选项。

表1是几种算法对一幅 256×256 的灰度Lena图像进行压缩的比较结果。其中原始SPIHT及EBCOT算法的PSNR数据见参考文献[3]和[5]。

表1 各种编码方式的PSNR比较

Table 1 PSNR comparison of various coding methods

	比特率					
	1.716	0.037	0.1	0.246	0.543	1.021
SPIHT	20.247	22.772	25.851	29.639	33.961	38.697
改进方法	21.401	23.998	26.639	30.422	34.909	39.680
改进+算术	21.961	24.866	28.002	32.003	36.370	41.141
EBCOT	22.899	27.881	31.742	36.0165	40.808	46.627

由上表可以看出,改进算法的PSNR值高于原始算法1db,若在编码的最后一级加入算术编码,可使PSNR值进一步提高;原始的SPIHT与EBCOT构架相比,其PSNR远低于后者,而采用本文的改进算法加算术编码方式与EBCOT相比PSNR约少4db左右。如果采用自适应或更高阶的无失真算术编码器,则可使差距进一步减少^[6]。

4 结论

笔者提出一种基于SPIHT算法的静止图像编码方法,即在原有算法基础上通过前置直流滤波,缩小系数动态范围,在小波变换后对小波变换系数按高低频进行小区块划分,并分别设置阈值。采用新的阈值比较策略,可减少搜索次数,减少了链表中结点数,缩减了计算所需的内存空间,提高了编码效率。通过对处理后图像的PSNR值的对比可以看出,改进算法比原始算法的效能有所提高。仿真结果表明:在码率相同的条件下,改进的方法比原始算法在降低复杂度的同时具有更高视觉效果和编码性能。

参考文献:

- [1] SAID A, PEARLMAN W A. A New, fast, and efficient image codec based on set partitioning in hierarchical trees [J]. IEEE Trans. on Circuit and System for Video Technology, 1996, 6(3): 243-250.
- [2] 陈军, 吴成柯. 基于整数提升小波变换的彩色图像零树编码算法[J]. 电路与系统学报, 2002, 7(2): 6-9.
CHEN Jun, WU Cheng-ke. Zerotree coding algorithm for color image based on integer lifting wavelet transform [J]. Journal of Circuitry and System, 2002, 7(2): 6-9. (in Chinese)
- [3] TAUBMAN D S, MARCELLIN M W. JPEG2000 图像压缩基础、标准和实践[M]. 魏江力, 柏正尧, 译. 北京: 电子工业出版社, 2004.
TAUBMAN D S, MARCELLIN M W. JPEG2000 image compression foundation standard and practice [M]. Translated by WEI Jiang-li, BO Zheng-yao. Beijing: Publishing Housing Electronics Industry, 2004. (in Chinese) (下转第396页)