

基于内存优化的小波零块嵌入图像编码算法

王 娜, 李 霞

(深圳大学信息工程学院, 广东深圳 518060)

摘 要: 小波嵌入零块编码算法(Set Partitioned Embedded block, SPECK)是一种高效的具有渐进传输特性的图像编码算法,但其在编解码过程中的巨大内存占用致使编解码速度慢且不利于硬件实现.本文提出了一种基于内存优化的小波零块嵌入图像编码算法,采用标志状态图和块深度优先搜索策略完成嵌入编码中的排序和细化过程.两张标志状态图分别标识编解码过程中的重要系数和不重要集合,同时结合块深度优先搜索策略检索块结构中的不重要集合,代替排序和细化过程中的重要系数链表和不重要集合链表,较大地节省了内存占用.实验结果表明,与 SPECK 算法相比,在保持相当信噪比的情况下,内存占用仅为原来的 1/12,与另一种低内存零树编码算法 LZC(Listless Zerotree Coding)相比,在内存略有增加的情况下,信噪比至少提高 1.1dB,为硬件实现小波零块编码算法探讨了一条有效途径.

关键词: 图像编码; 小波变换; 嵌入式块编码; 内存优化

中图分类号: TN919.81 **文献标识码:** A **文章编号:** 0372-2112(2006)11-2068-04

A New Low Memory Set Partitioned Embedded Block Coder

WANG Na, LI Xia

(College of Information Engineering, Shenzhen University, Shenzhen, Guangdong 518060, China)

Abstract: Said and Pearlman's Set Partitioned Embedded block (SPECK) algorithm is computationally simple and efficient for image compression and progressive transmission, however there is a main drawback of high memory requirement for hardware implementation. We propose a new low memory embedded block coder that keeps those desirable features, and the novelties are twofold. A compact form of the flag maps is introduced to store the significant coefficients and sorting information instead of using lists. A new block depth finding strategy is developed for searching insignificant sets at sorting stage. Experimental results show that the obtained PSNR values for the decoded images are very close to those of SPECK algorithm and the memory consumption is reduced by 12 times. Compared with another low memory coder (Listless Zerotree Coder, LZC), the PSNR of our algorithm improves at least 1.1dB with a minor increase of memory requirement. This provides an efficient way for hardware implementation of wavelet embedded block coding.

Key words: image coding; wavelet transform; embedded block coding; memory optimization

1 引言

近年来,随着多媒体应用的快速增长和网络的不断发展,具有嵌入性和渐进传输性的小波嵌入式编码方法受到越来越多的重视.典型的小波嵌入编码方法有两类,一类是以 Shapiro 提出的嵌入式小波零树编码方法^[1](Embedded Zerotree Wavelet coder, EZW)和 Said 等提出的集合分裂树算法^[2](Set Partitioned in Hierarchical Tree, SPIHT)为代表的小波嵌入零树编码方法,另一类是在 SPIHT、AGP^[3]、SWEET^[4]三种编码方法基础上提出的小波嵌入零块图像编码^[5](Set Partitioned Embedded block, SPECK)算法.小波嵌入零树编码是基于小波变换后小波子带形成的等级树结构,采用逐次逼近量化方法开发子带间的系数相关性,以获得较高的压缩比和较好的图像压缩质

量,并形成嵌入式码流.相反,小波零块图像编码方法利用小波变换子带内的能量集中特性而非子带间的系数相关性来压缩图像,在获得嵌入码流的同时,得到高于 SPIHT 算法的压缩比和更好的图像质量.上述两类小波嵌入式编码方法在编解码过程中都需要维护排序信息和细化信息,而用于存储排序信息和细化信息的链表需要大量内存.以 SPECK 算法为例,由于在编码过程中需要维护 2 个链表,对于一幅 $512 \times 512 \times 8\text{bit}$ 大小的图像,需要内存 1.125M 字节,巨大的数据存储量使编码时的内存占用很大,既影响编码速度又不利于硬件实现.因此,研究如何节省编码内存已成为小波嵌入式编码方法是否实用的一个关键因素.针对小波嵌入零树编码中的大量内存消耗,作者曾提出过一种基于内存优化和启发式深度优先搜索策略的小波零树编码方法^[6],改进了 SPIHT 算法和

EZW 算法中的排序和细化过程,即采用标志位图代替细化过程中的重要系数链表,同时采用启发式的深度优先搜索策略省去存储排序全局信息的链表和位图,较大地节省了编解码内存。

本文就是在此基础上,将位图标志的思想进一步引入到嵌入式小波零块编码(SPECK)中,根据嵌入式小波零块的编码结构提出了一种新的基于内存优化的零块嵌入图像编码算法,即用两种类型的标志状态图——系数重要性位图和小波块分裂状态图,代替 SPECK 算法中的重要系数链表和排序信息链表,结合块深度搜索策略完成小波嵌入零块编码中的排序和细化过程,在保持高信噪比和原嵌入码流结构的情况下,较大地节省了所需编解码内存。

2 基于内存优化的零块嵌入图像编码

2.1 小波嵌入零块图像编码和小波嵌入零树图像编码对比分析

小波嵌入零块图像编码(SPECK)遵循小波零树编码的思路,将对小波变换系数的编码分解为对标志系数重要性的排序信息编码和对重要系数的幅度编码两部分,所以也是通过排序和细化两个过程来实现。在排序过程中,用一个 LIS(List of Insignificant Set) 链表存储不重要集合信息,维护排序过程。在细化过程中,用一个 LSP(List of Significant Pixel) 链表存储重要系数信息。在小波零块图像编码中,虽然排序和细化过程中的链表含义和功能类似于小波零树编码 EZW 算法中的主列表、副列表和 SPIHT 算法中的单向链表(LSP、LIS),但在两类嵌入编码中链表的形成却有本质的不同,这主要是由于小波嵌入零块图像编码与小波嵌入零树编码的编码结构不同所造成,下面给出详细分析。

小波嵌入零树编码利用小波分解图像子带之间(不同尺度子带间)系数的相关性,即粗尺度子带能较好的预测细尺度子带系数,在高阈值下不同尺度间的小波系数构成了零树结构,如图 1(a)所示,粗尺度下的父系数和在相同空间位置的所有系数(level3 中的系数)和下一个尺度更细的在相同方向上的所有子系数(level2 和 level1 中箭头所指的系数)构成一棵零树。因此,在编码过程中维护排序信息的主链表或 LIS 链表存储的是这些零树结构中的不重要集合信息,重要系数信息就存储于副链表或 LSP 链表中,这两个链表是严格按照对每棵零树的深度优先搜索策略形成的。相反地,小波嵌入零块编码方法利用的是子带内(同一尺度子带)系数的相关性,即利用小波块内的频域和时域能量集中特性,高阈值时同一尺度子带下的小波系数呈现整体不重要性,构成零块结构,如图 1(b)所示, S 集合中按照四叉树的分裂方法和 I 集合中按照锥形分裂的方法^[5]形成了不重要集合和重要元素集合,分别存储于 LIS 链表和 LSP 链表。

两类编码中的链表形成过程不同决定了在节省编码内存时要利用不同的状态标志图结构和搜索策略。在小波嵌入零树编码中,由于不重要集合信息的标注是严格按照从树干到树叶的深度方向,即节点间有严格的阶梯父子关系,每个节点要么是一棵树的子节点,要么是另一棵子树的父节点,父节点

标注的集合不重要性信息明确预示了其子集的不重要性,因此可以采用一张位图标志结合严格的深度优先搜索策略代替零树编码中的不重要集合链表,详见文[6]算法。但在小波嵌入零块编码中,不重要集合的标注是按照从大块到小块的深度方向,节点之间没有严格的父子关系,即同一个节点可能有多个以该节点为父节点的子集合,如图 1(b)所示,节点 1 既是整个块集合 S_1 的父节点又是具有不同集合长度的子块 1、2、3 集合的父节点(集合长度定义为该集合中的象素个数),因此,需要开发一种新的基于状态标志图的内存优化结构和块深度搜索策略。

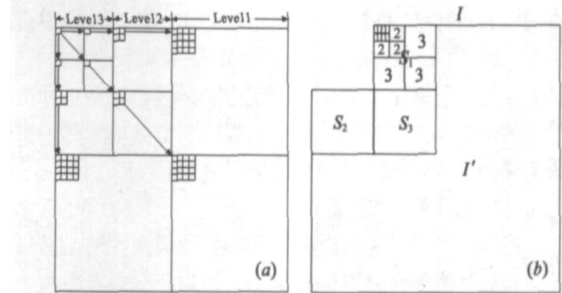


图 1 (a) 零树编码中的节点关系;(b) 零块编码中的节点关系

2.2 状态标志图的内存优化结构和块深度搜索策略

2.2.1 基于状态标志图的内存优化结构

小波嵌入零块图像编码可归结为基于块分裂的重要系数渐进搜索。在搜索过程中提出采用两张状态图 F_s 和 F_c 的内存结构,状态图 F_c 用以标记块中象素是否为重要系数,代替小波嵌入零块编码中的存储重要系数的链表 LSP。状态图 F_s 用以标记块集合的分裂状态,提供搜索过程中的状态信息以维护排序过程,代替存储排序信息的链表 LIS。状态图 F_c 的定义和功能如前^[6]所述,即如果 $F_c(i, j) = 1$,表明该系数为重要系数。 F_c 图和原图大小相同,对于一幅 512×512 的灰度图像, F_c 图占用 32K 字节的内存。为了解决小波零块图像编码在块分裂中节点间父子关系不明确的问题,本文定义了一种新的采用 2bit 位图结构的状态图 F_s 。采用 2bit 位图结构的原因有两个:(1) 可以表示出集合的分裂状态,即 2bit 中的 '00' 状态代表该块还未分裂,表明该块为不重要集合,其他状态就代表该块已经分裂,代表该块已含有重要系数,已分裂出不重要集合,其父子关系由分裂次数决定;(2) 可以表示出集合的分裂次数。集合的分裂次数可以解决块分裂中的父子关系不明确问题,如图 1(b)所示,若父节点 1 的分裂次数为 1,代表节点 1 是块 1 和 2 的父亲节点,若分裂次数为 2,代表节点 1 是块 1 的父亲节点,因此分裂次数明确了块中节点间的父子关系。之所以选择 2 比特位图,主要是综合考虑了内存占用和图像编码的要求。选择越多位的位图结构,就需要占用更多的内存。相反,最少位的位图表示(1bit 位图)只能表示 4 次分裂,不能满足图像编码的需求。在小波零块的块分裂中,最小块的集合长度为 4,四个 2bit 信息就可以表示 256 次分裂,能满足块大小 $2^{25} \times 2^{26}$ 的分裂,对应于原始图像大小为 $2^{257} \times 2^{257}$ 的编码,因此选择 2bit 位图既能满足图像编码的需求,所占用的内存又较少。 F_s 图和原图大小相同,对于一幅 512×512 的灰度图

像, F_s 图占用 64K 字节的内存. 因此, 本文采用的两种标志态图共占用 96Kbytes, 与 SPECK 算法占用 1.125Mbytes 的内存相比, 内存占有是原来的 1/12, 较大地减少了编码内存, 优化了编码内存结构.

2.2.2 基于内存优化的块深度搜索策略

为了配合采用状态位图 F_s 和 F_c 的编码结构, 本文提出了一种块深度优先搜索策略, 首先对集合及象素重要性检测函数 $S_n(T(i, j))$ 定义如下

$$S_n(T(i, j)) = \begin{cases} 1, & \text{if } \max_{c(k, l) \in T(i, j)} |c(k, l)| \geq 2^n \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

其中, $T(i, j)$ 的含义由 $T(i, j)$ 集合大小决定, 当 $T(i, j)$ 集合大小为 1 时, 代表位置在 (i, j) 的单个象素; 否则代表以 (i, j) 为父节点的一个集合. $c(k, l)$ 和 n 分别表示小波系数值和当前搜索的位平面值. 如果式 (1) 为 1, 表明父节点的子孙中有重要象素. 整个块深度优先搜索步骤归结如下:

①初始化分裂次数变量;

②把集合 S 四叉树划分四个子集合, 分裂次数加 1, 并根据分裂参数赋值集合分裂标志 $F_s(i, j)$, 即若分裂参数为真时, $F_s(i, j)$ 加 1;

③把四个子集合的位置, 集合大小和集合序号压入临时栈中;

④对每个子集合根据集合序号和集合分裂标志 $F_s(i, j)$ 确定该集合是否已分裂.

若已分裂, 判断集合是否可扩展, 若是, 令分裂参数为假, 转②; 否则转④;

否则, 判断该象素代表的是否为单个重要象素, 若不是则输出重要性检测信息 $S_n(T(i, j))$

若 $S_n(T(i, j)) = 1$, 代表集合中有重要象素

若集合可扩展, 即不是单个象素, 则设分裂参数为真, 转②;

否则, 设该象素为重要象素, 输出其符号位, 并且令 $F_s(i, j) = '00'$; 转④;

需要说明的是, 根据集合序号和集合分裂标志 $F_s(i, j)$ 确定该集合是否已分裂的方法如下: 当集合序号为 1 时, 集合分裂标志应为 $F_s(i, j)$ 减去分裂次数, 否则集合分裂标志就为 $F_s(i, j)$. 从块深度优先搜索策略可以看出, 一经访问到某节点就访问到了它的所有子孙, 不用预先分配内存来存储重要象素信息和不重要集合信息, 状态位图 F_c 标识了重要象素, 省去了传统小波零块编码中存储重要系数的链表 LSP. 状态图 F_s 决定了排序过程中的搜索方向, 根据状态图 F_s 可以寻找出不重要集合, 即若状态位图 F_s 不等于 '00', 表明该块必然包含重要象素, 根据它的分裂次数进行相应分裂, 可定位出在上一阈值下的不重要集合和本次阈值下的不重要集合, 省去传统小波零块编码算法中的排序信息链表 LIS, 较好地满足了节省内存占用编码的需要.

2.3 基于内存优化的小波嵌入零块图像编码算法

对一幅小波分解之后的图像进行

内存优化的小波嵌入零块编码步骤如下:

(1) 选定初始阈值为 $\theta = 2^{\lfloor \log_2 \max_{i,j} c(i, j) \rfloor}$, $\lfloor \cdot \rfloor$ 表示取整运算;

(2) 初始化重要系数位图 $F_c(i, j)$ 和集合分裂标志位图 $F_s(i, j)$;

(3) 小波分解后的所有子带带集合 X 表示, 其中低频子带系数构成集合 S , 其余小波子带构成集合 I , 即 $I = X - S$;

(4) 编码集合 S , 即首先判断分裂集合标志位 $F_s(i, j)$

若 $F_s(i, j) = '00'$, 则输出集合重要性检测信息 $S_n(T(i, j))$

若 $S_n(T(i, j)) = 1$, 代表集合中有重要象素

若集合可扩展, 即不是单个象素, 则设分裂参数为真, 调用块深度搜索策略检出重要系数;

否则, 设该象素为重要象素, 若 $F_c(i, j) = 0$, 则输出其符号位, 并且令 $F_c(i, j) = 1$, $F_s(i, j) = '01'$;

否则, 判断集合是否可扩展, 若是令分裂参数为假, 调用块深度搜索策略检出重要系数;

(5) 编码集合 I , 即首先判断分裂集合 I 标志位 $F_I(i, j)$

若 $F_I(i, j)$, 输出 $S_n(T(i, j))$; 若 $S_n(T(i, j)) = 1$, 则 $F_I(i, j) = 1$;

若 $F_I(i, j) = 1$ 或 $S_n(T(i, j)) = 1$

锥形分裂^[5]集合 I , 将集合 I 分成三个子集合 S 和一个新的集合 I' ;

编码每个子集合 S ; 编码集合 I' ;

(6) 令 $\theta = \theta/2$, 如果阈值 $\theta < 1$, 则退出; 否则转(3);

在上述算法中位图标志 $F_I(i, j)$ 所占用内存的大小由小波分解层数决定, 小波分解层数与 $F_I(i, j)$ 所占用内存位数一致, 即小波分解层数为 8 层时, 位图标志 F_I 才占用 1 字节, 因此在计算内存占用时可以忽略不计. 由此可见, 本文采用两张状态标志图 F_c 和 F_s , 结合块深度优先搜索策略, 省去了 SPECK 算法中的用于存储排序信息和重要系数的链表, 在保持高信噪比和嵌入码流的情况下, 较大地节省了编码时的内存占用.

3 实验结果与分析

本文算法在 PIII 700MHz 的微机上用 Visual C++ 6.0 进行验证, 选用 $512 \times 512 \times 8\text{bit}$ 的标准测试灰度图像 Camera、Goldhill、Lena 和 Barbar. 小波滤波器采用 Antonini⁹/7^[7] 双正交滤波器, 在图像边界处采用对折式扩展. 为了便于和文章[6]进行比较, 首先对几种小波嵌入式编码码流采用了文章[6]中的自适应算术熵编码方案, 图 2 显示了 EZW 算法、SPIHT 算法、LZC^[8] 算法、SPECK 算法、文章[6]算法和本文算法对几幅图像的峰值信噪比. 表 1 给出了在采用 SPECK 算法中的基于

表 1 几种算法在基于上下文的熵编码方案下的峰值信噪比

编码算法	编码 Lena 图像			编码 Babara 图像			编码 Goldhill 图像		
	0.25bpp	0.50bpp	1.00bpp	0.25bpp	0.50bpp	1.00bpp	0.25bpp	0.50bpp	1.00bpp
LZC	31.10	33.70	37.01	26.24	30.22	35.51	29.35	31.89	35.32
SPIHT	34.11	37.21	40.44	27.58	31.40	36.41	30.56	33.13	36.55
SPECK	34.03	37.10	40.25	27.76	31.54	36.49	30.50	33.03	36.36
本文算法	34.12	37.13	40.25	27.84	31.58	36.50	30.64	33.05	36.37

上下文的熵编码方案时, 本文算法和其他几种算法在不同码率下的信噪比比较. 表 2 给出了这六种算法的内存占用表.

表 2 几种小波嵌入编码算法的内存占用表(单位: 字节)

编码算法	SPIHT 算法	EZW 算法	LZC 算法	文章[6]算法	SPECK 算法	本文算法
占用内存	$27/4 \times N \times N$	$9/2 \times N \times N$	$5/32 \times N \times N$	$1/8 \times N \times N$	$9/2 \times N \times N$	$3/8 \times N \times N$

从图 2 可以看出, 本文算法和 SPECK 算法的信噪比曲线几乎重叠, 两种算法具有相当的信噪比, 但本文算法所占内存仅为 SPECK 算法的 1/12. 分析表 1 和表 2 数据, 与 SPIHT 算法相比, 在占用内存进一步减少的情况下, 对于不同图像信噪比稍有波动, 但都明显高于 EZW 算法; 与另

一种低内存占用的 LZC 算法相比, 尽管内存占用增加, 但信噪比却明显改善, 至少提高了 1.1dB. 此外, 在采用基于上下文的熵编码方案时, 本文算法和其它几种算法一样, 信噪比都获得了明显提高, 但增加了编解码的复杂度.

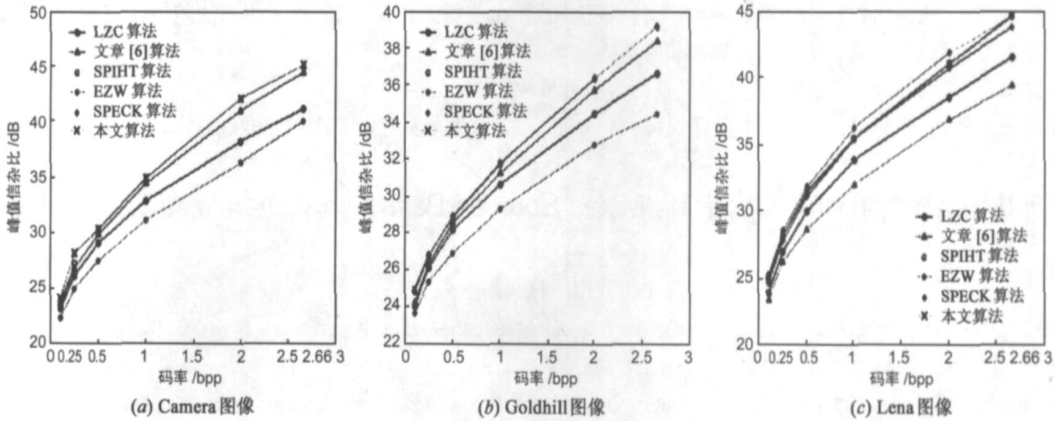


图 2 不同图象的峰值信噪比图

4 结论

适合网络传输的小波嵌入式编码算法由于要采用零树结构或零块结构组织小波系数, 所以需要开辟大量内存来存储排序和细化过程中的全局信息. 本文针对小波嵌入零块编码提出了一种新的标志状态图的内存优化结构, 并开发了一种适合这种内存优化结构的块深度优先搜索策略, 在保持高信噪比的情况下, 大大减少了编解码中的内存占用, 为硬件实现嵌入零块图像编码提供了有效途径.

参考文献:

[1] J M Shapiro. Embedded image coding using zerotrees of wavelets coefficients [J]. IEEE Trans. Signal Processing, 1993, 12(41): 3445- 3462.

[2] A Said, W A Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees [J]. IEEE Trans. on Circuits Syst. Video Technology, 1996, 6(3): 243- 250.

[3] A Said, W Pearlman. Low-complexity waveform coding via alphabet and sample set partitioning [A]. Proc. Visual Communications and Image Processing [C]. San Jose, California: SPIE, 1997. 25- 37.

[4] J Andrew. A simple and efficient hierarchical image coder [A]. IEEE Int. Conf. on Image Proc [C]. Washington, DC: IEEE Computer Society, 1997. 658- 661.

[5] A Islam, W A Pearlman. An embedded and efficient low complexity hierarchical image coder [A]. Proc. of Visual Communication and Image Processing [C]. San Jose, California: SPIE, 1999. 294- 305.

[6] 王娜, 纪震, 贾传荧, 李霞. 基于内存优化和启发式深度优

先搜索的小波零树图像编码算法 [J]. 电子学报, 2005, 33 (3): 440- 443.

WANG Na, et al. A wavelet zerotree image coding method based on memory optimization and heuristic depth finding strategy [J]. Acta Electronica Sinica, 2005, 33(3): 440- 443. (In Chinese)

[7] M Anonini, M Barland, P Mathieu, I Daubechies. Image coding using wavelet transform [J]. IEEE Trans. Image Processing, 1992, 2(1): 205- 220.

[8] W K Lin, N Burgess. Low memory color image zerotree coding [A]. Proc. Information, Decision, and Control [C]. Adelaide, Australia: IEEE, DSTO, CSSIP, 1999. 91- 95.

作者简介:



王 娜 女, 1977 年生于河北保定, 工学博士, 深圳大学信息工程学院副教授, 主要从事图像编码、小波理论、模式识别等方面的研究工作. E-mail: wangna@szu.edu.cn



李 霞 女, 1968 年生于四川乐山, 工学博士, 深圳大学信息工程学院教授, 博导, 主要从事图像编码、智能计算、最优化等方面的研究工作.