

基于单队列递归扫描的嵌入式零树图象编码方法

张海翔 陈 纯 庄越挺

(浙江大学计算机科学系, 杭州 310027)

摘 要 为了提高基于互联网的图象编码的压缩比和多带宽适应性,提出了一种新的嵌入式零树图象编码方法.该方法首先在连续累进量化方法中使用单队列算法,从而不仅无需使用动态队列,而且率失真特性也得到改善;其次设计了一种基于空间树递归扫描的零树编码快速算法,因而完全解决了重复扫描问题;最后基于编码符号的频带内邻域相关性,为自适应算术编码定义一种新的 Markov 模型.实验结果表明,该算法在保持高粒度多码率特性的同时,压缩效率较 EZW 算法有明显提高.

关键词 图象压缩 小波变换 零树编码 连续累进量化

中图分类号: TN919.81 **文献标识码:** A **文章编号:** 1006-8961(2002)07-0709-07

Embedded Zerotree Wavelet Image Coding Algorithm Based on Single List and Recursive Scan

ZHANG Hai-xiang, CHEN Chun, ZHUANG Yue-ting

(Department of Computer Science, Zhejiang University, Hangzhou 310027)

Abstract An improved embedded zerotree wavelet image coding algorithm is proposed in this paper. It is an improved version of Shaprio's EZW algorithm. First, a single list scheme is used in successive-approximation Quantization(SAQ), dispensing with double dynamic list of EZW algorithm. The single list make the scan process more simple and faster, and lead to better rate-distortion characteristic. Second, a fast algorithm based on spatial orientation tree recursive scan for the zerotree coding is designed. The zerotree symbols of one quantization layer can be calculated with only one access per coefficient, solving the rescanning problem completely. At last, a new Markov model based on quantization symbol field relativity for adaptive arithmetic coding algorithm is used, for higher compression efficient. Experimental results show that it performs better than the EZW algorithm in coding efficiency, maintains the fine-granularity scalability.

Keywords Image compression, Wavelet transform, Zerotree coding, Successive approximation quantization

0 前 言

随着互联网的日益发展和普及,尤其是宽带技术的成熟和发展,使可视电话、视频会议等新一代多维通讯的广泛应用成为可能.目前,基于互联网的图象和视频传输技术研究已经成为宽带互联网应用中的关键课题.由于基于 IP 协议的互联网是多应用协议共享信道,即一个通讯用户在互联网上的可用带宽与网络的运行情况相关,是一个变化量,因此,基

于互联网的图象和视频传输对于图象编码方法不仅要求高压缩比,还要求有多带宽适应性.

嵌入式零树图象编码方法(EZW)^[1]输出的图象编码既有高压缩比,又有高粒度多码率特性,非常适合于解决基于互联网的图象和视频传输问题,但 EZW 算法在各个环节都存在一些问题,如:

(1) 基于系数空间树的零树编码模型定义比较简单,有待改进.

(2) 连续累进量化方法是采用双队列算法,由于需要用到动态队列技术,即需频繁进行动态内存

基金项目:国家计委产业化前期关键技术与成套设备研制开发项目(计高技[1998]2077 批文)

收稿日期:2001-04-27; 改回日期:2001-12-27

管理,这样队列节点的维护开销大,且算法结构比较复杂,因此不利于跨平台实现和移植。

(3) 在进行零树量化编码时,因需要对系数空间树进行重复扫描,故算法效率不高。

(4) 自适应算术编码的 Markov 模型比较简单,因其只考虑量化符号在系数空间树结构中的相关性,故对于提高压缩比作用不大。

在 EZW 算法基础上发展起来多带宽适应图象编码方法,如 SPIHT^[2], MPEG-4 wavelet still-image coding tool^[5,4]等,其主要工作均集中在改进编码方法的模型定义上,而对于前述第(2,3,4)等问题都没有充分研究和解决。

本文在对连续累进量化编码算法、基于零树编码的扫描算法和用于自适应算术编码的 Markov 模型等 3 个方面进行研究的基础上,提出了一种连续累进量化编码的单队列算法,该算法不仅无需使用动态队列,而且率失真特性也得到改善;同时设计了一种基于空间树递归扫描的零树编码快速算法,从而完全解决了重复扫描问题;还基于编码符号的频带内邻域相关性,为自适应算术编码定义了一种新的 Markov 模型。实验结果表明,本文算法在保持高精度多码率特性的同时,压缩效率较 EZW 算法有明显提高。

1 基本概念

本文算法的结构和 EZW 算法结构类似,即以二维小波变换为基础,使用系数空间树的零树编码,和连续累进量化方法,并利用自适应算术编码方法来提高压缩性能。下面给出系数空间树和连续累进量化方法的概念定义。

1.1 系数空间树

图象经 n 层小波分解得到的小波变换系数矩阵为 $W_{M,N}$,简记为 W 。它可以表示为频带集合:

$$W = \{LL_n, HL_n, LH_n, HH_n, \dots, HL_1, LH_1, HH_1\} \quad (1)$$

这些频带可以构成树型结构,其中 HL_i, LH_i, HH_i , ($i=1, \dots, n$) 为同级频带, LL_n 为顶级频带。位于坐标 (x, y) 处的系数表示为 $c(x, y)$ 。

系数空间树是由频带中相同空间位置的系数组成的系数结构,如图 1 所示。它可以用来考察图象中某一空间位置不同分析尺度系数的相关性。

定义 1 对于系数 $c(x, y) \in W$, 有子节点集合 $S_{child}(c(x, y))$ 为

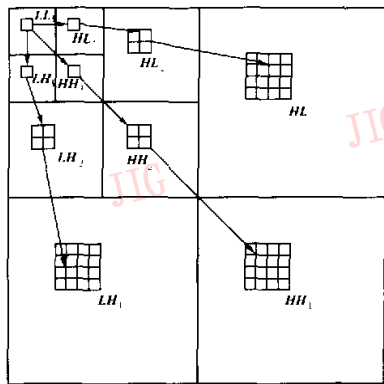


图 1 系数空间树

$$S_{child}(c(x, y)) = \begin{cases} \text{当 } c(x, y) \in LL_n \text{ 时} \\ \{c(x+l, y), c(x, y+l), c(x+l, y+l)\} \\ \text{当 } c(x, y) \in \{HL_n, LH_n, HH_n, \dots, \\ HL_2, LH_2, HH_2\} \text{ 时,} \\ \{c(2x, 2y), c(2x-1, 2y), c(2x, 2y+1), \\ c(2x+1, 2y+1)\} \end{cases} \quad (2)$$

其中, l 为频带 LL_n 的尺寸。系数空间树的二元关系为 $P < c(x, y), c, >$, 其中 $c, \in S_{child}(c(x, y))$ 。 $D(x, y)$ 表示以系数 $c(x, y)$ 为根的系数空间树的系数集合, $O(x, y)$ 表示系数集合 $\{D(x, y) - \{c(x, y)\}\}$, 即 $O(x, y)$ 表示 $D(x, y)$ 中, 除根系数 $c(x, y)$ 外的其他系数集合。对于系数 $c(x, y) \in \{HL_1, LH_1, HH_1\}$, 有 $D(x, y) = \{c(x, y)\}$ 。

1.2 连续累进量化方法

连续累进量化方法是一种分层量化方法, 整个量化过程可以分为一系列量化层。这里, 量化层的量化阈值 T 构成一个阈值序列

$$T = T_0 \left\{ \frac{1}{2} \right\}^i \quad (i = 0, 1, 2, 3, \dots) \quad (3)$$

其中, T_0 为初始阈值, i 为量化层序号。这里用符号 $Q(T)$ 表示与阈值 T 对应的量化层。一般量化层数越多, 量化阈值越小, 输出量化结果的精度也越高, 而相应的压缩比则越低。这种量化方法输出的量化编码流有嵌入式特点, 即使编码/解码过程在任意位置停止, 都可以得到有一定精度的编码/解码结果。这一特点在图象编码传输应用领域就表现为高精度多码率特性。

图象信息一般用系数矩阵表示, 由于其量化结果中必须包含系数的量化值和它在系数矩阵中的位置等两部分信息, 且一般系数量化值由量化符号表

示,位置信息用编码顺序隐含表达,因此,量化过程也应包含两部分:(1)量化编码模型,它把一个系数值 $c(x, y)$ 根据一定的转换规则和量化阈值 T 转换为一个量化符号,一般可以表示为函数;(2)扫描顺序,量化过程需按照这个顺序对系数进行量化编码.连续累进量化方法的量化层是一个复合量化过程,它包括两个子量化过程,即主量化过程(dominate pass)和次量化过程(subordinate pass)^[1~3],分别量化主要系数和量化剩余系数.由于这两个子过程所处理的系数集合在不同的量化层中有变化,因此在算法中还要反映系数集合变化.现将连续累进量化方法的量化层 $Q(T)$ 结构组成表示如下:

定义 2

$$\begin{cases} Q(T) = \langle Q_z(T), Q_r(T) \rangle \\ Q_z(T) = \langle C_z(c(x, y), T), L_z, Z \rangle \\ Q_r(T) = \langle C_r(c(x, y), T), L_r, R \rangle \end{cases} \quad (4)$$

其中,下角 z 代表 zero(零树), r 代表 refine(修正), $Q_z(T)$ 表示主(零树)量化过程(dominate pass), $C_z(c(x, y), T)$ 表示零树量化编码模型,其输出符号表为 {POS, NEG, IZ, ZTR}^[1], L_z 表示 $Q_z(T)$ 过程扫描顺序, Z 为主要系数集合, $Q_r(T)$ 表示次(修正)量化过程(subordinate pass), $C_r(c(x, y), T)$ 表示修正量化编码模型^[1],其输出符号表为 {1, 0}, L_r 为 $Q_r(T)$ 过程扫描顺序, R 表示剩余系数集合.

2 连续累进量化方法的单队列算法

定义 2 表示的连续累进量化方法是一种系数重排序方法,其采用的是双队列结构,在本文中称为双队列算法.在其中的量化层中,由两个量化过程($Q_z(T)$ 和 $Q_r(T)$) 分别处理两个队列 ($\langle L_z, Z \rangle$ 和 $\langle L_r, R \rangle$). 在编码过程中,这两个队列按照以下的规则动态变化:

如果 $C_z(c(x, y), T) = \text{POS} | \text{NEG} | \text{IZ}$, 则,

- (1) 将 $c(x, y)$ 从队列 $\langle L_z, Z \rangle$ 中删除;
- (2) 将 $c(x, y)$ 追加到队列 $\langle L_r, R \rangle$;
- (3) 将 $c(x, y)$ 的子节点 $S_{\text{child}}(c(x, y))$ 追加到队列 $\langle L_z, Z \rangle$.

由于小波系数矩阵中每一级频带的系数个数是按 4^l 的关系增长,且在编码过程中随着量化阈值 T 的变化,在集合 Z 中,需要被单独编码的系数数量急剧增长,因此,在用双队列算法进行编码过程中,需要大量的队列节点管理操作.这些操作不仅包括

队列节点的插入、删除,还包括节点内存的动态分配,这不仅计算开销大,而且算法结构复杂,故不利于跨平台实现和移植.

笔者认为,这种基于系数重排序的双队列算法不是必须的,因为通过合并两个量化过程的系数量化编码模型、系数集合和扫描顺序,也可以构造一种算法结构更加简单的单队列方法.

2.1 量化编码模型和扫描顺序的合并

由于量化编码模型 $C_z(c(x, y), T)$ 和 $C_r(c(x, y), T)$ 所定义的系数集合 Z 和 R 是不相交的,即: $\Phi = Z \cap R$, 又有 $C = Z \cup R$, 如果把系数 $c(x, y)$ 的集合属性作为一个编码判断条件,则可以把两者合并为一个等价的统一量化编码.

定义 3 在系数矩阵 W 上,统一量化编码模型 $C(c(x, y), T)$ 定义为

$$C(c(x, y), T) = \begin{cases} C_z(c(x, y), T) & \text{if } c(x, y) \in Z \\ C_r(c(x, y), T) & \text{if } c(x, y) \in R \end{cases} \quad (5)$$

其输出量化符号表为 {POS, NEG, IZ, ZTR}, {0, 1}. 这里,统一量化编码模型的直观含义就是:系数矩阵在进行量化时,先进行主要系数,后进行剩余系数的量化次序不是必须的.

在零树量化编码模型 $C_z(c(x, y), T)$ 中,当系数 $c(x, y)$ 被编码为 ZTR 时,表示 $c(x, y)$ 及其空间树中的后代节点系数都被量化为“0”.由于这种编码模型有空间树后向预测性,它对符号编码顺序的要求就是保持各系数在空间树中的次序;而修正量化编码模型 $C_r(c(x, y), T)$ 只涉及量化剩余系数 $c(x, y)$ 本身,且集合 R 中的系数之间不存在空间树关系,因此,可以用任何一种顺序进行修正量化编码.

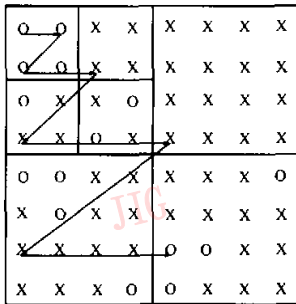
定义 4 扫描顺序 L 中任意两个系数 c_j, c_i , 它们在扫描顺序中的位置分别为 $N_L(c_j)$ 和 $N_L(c_i)$, 若它们存在空间树二元关系 $P\langle c_j, c_i \rangle$, 则有 $N_L(c_j) < N_L(c_i)$, 称扫描顺序 L 保存系数空间树的次序.

定理 1 扫描顺序 L 保持系数空间树的次序. 有一个系数 c_i , 对于 L 中任一系数 c_j , 并不存在空间树二元关系 $P\langle c_j, c_i \rangle$. 当将系数 c_i 插入 L 中的任一位置, 如得到一个扫描顺序 L' , 则扫描顺序 L' 仍保持系数空间树的次序.

证明 很明显, 在新扫描顺序 L' 中, 由于原扫描顺序 L 中的各系数之间的次序关系仍保持不变, 同时又由于新增系数 c_i 与其他系数不存在空间树二元关系, 即对于新扫描顺序 L' 中任一存在空间树二元关系的系数, 它们的位置关系不变, 所以扫描

顺序 L' 也保持系数空间树的次序。

树型“Zig”扫描顺序,简记为 L_{zg} ,是双队列算法中主量化过程扫描顺序 L_s 的初始状态。它可以看作是将扫描顺序 L_r 中的系数插入扫描顺序 L_s 而构成的一种顺序,图2为扫描顺序 L_{zg} 示意图。根据定理1可知, L_{zg} 顺序也保持系数空间树的次序,符合零树量化编码模型 $C_z(c(x,y),T)$ 的要求。由于统一量化编码模型 $C(c(x,y),T)$ 是 $C_z(c(x,y),T)$ 和 $C_r(c(x,y),T)$ 构成的复合模型,且 $C_z(c(x,y),T)$ 对扫描顺序没有限制,因此 L_{zg} 顺序符合 $C(c(x,y),T)$ 的要求。



X:重要系数,O:剩余系数
图2 扫描顺序 L_{zg} 示意图

2.2 单队列算法及其率失真特性

定义5 连续累进量化方法单队列算法的量化层定义为

$$Q'(T); \langle C(c(x,y),T), L_{zg}, W \rangle \quad (6)$$

其中, $C(c(x,y),T)$ 为统一量化编码模型, L_{zg} 为树型“Zig”顺序, W 为图象系数矩阵。

简单地说,单队列算法就是取消了系数重排序,始终按照 L_{zg} 顺序进行扫描,在进行量化编码时,则根据系数的量化情况来选择量化编码模型。和双队列算法相比,这种算法不仅结构简单、跨平台移植和实现更加容易,而且图象编码流的率失真特性有很大改善。由于篇幅所限,这里只给出定性分析和结论:

(1) 由于两种算法的量化编码模型等价,它们对应的量化层输出的量化符号集也相同,即在两种算法中进行相同层数的量化编码时,其得到的量化符号也相同,因此在率失真曲线^[5]上的量化层分界点上两种算法的率失真特性相等。

(2) 由于在顺序 L_{zg} 上,小波系数值存在递减的统计规律,即在顺序 L_{zg} 上,位置越靠后的系数,其值小的概率大,因此这些小系数在量化过程中,往往被量化为 IZ 和 ZTR,其对应的量化值为 0。同时由于在

编码/解码的过程中,系数的量化值默认为 0,即当系数没有得到相应的量化符号时,本层的量化值默认为 0,因此这些小系数重要性相对较低。另外,由于按 L_{zg} 顺序进行编码输出的量化符号流符合先重要后次要的优先次序原则,因此本文的单队列算法在量化层内的率失真特性要优于双队列算法。

3 零树量化编码模型的递归扫描算法

零树量化编码模型描述了系数值在空间树结构上的相关性,是 EZW 算法压缩效率的主要来源,但由于在进行零树编码计算时,需要对系数空间树反复扫描,仍存在着重复扫描问题,故算法效率不高。本文给出了零树量化编码模型的递归形式定义,同时给出了一种递归扫描算法,从而完全解决了重复扫描问题。

3.1 重复扫描问题

零树量化编码模型 $C_z(c(x,y),T)$ 定义在空间树结构上,其编码的计算公式^[1]表示如下:

定义6

$$C_z(c(x,y),T) = \begin{cases} POS; c(x,y) \geq T; \\ NEG; c(x,y) \leq -T; \\ IZ; \max(|O(x,y)|) \geq T \text{ and } |c(x,y)| < T; \\ ZTR; \max(|D(x,y)|) < T \end{cases} \quad (7)$$

零树编码的重复扫描问题主要由不符合系数值“频谱衰减”规律的特殊点引起。如以图3(a)为例,有3个系数 $c(x_1, y_1)$, $c(x_2, y_2)$ 和 $c(x_3, y_3)$ 符合下列条件:

$$\begin{cases} |c(x_1, y_1)| < T, \text{ 且 } |c(x_2, y_2)| < T, \text{ 且 } |c(x_3, y_3)| \geq T \\ c(x_3, y_3) \in D(x_2, y_2), \text{ 且 } c(x_2, y_2) \in D(x_1, y_1) \end{cases} \quad (8)$$

根据定义6,编码 $c(x_1, y_1)$ 和 $c(x_2, y_2)$ 时,需要分别扫描空间树 $D(x_1, y_1)$ 和 $D(x_2, y_2)$, 而由于 $D(x_2, y_2) \subset D(x_1, y_1)$, 因此计算时,会对 $D(x_2, y_2)$

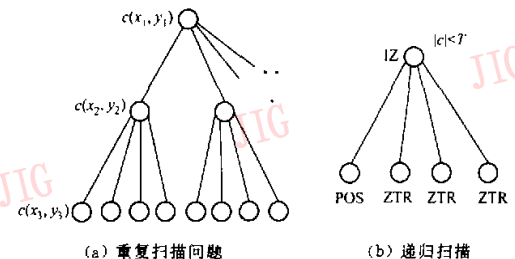


图3 零树编码的递归扫描方法

进行重复扫描.这种计算时的重复扫描在某些类型的图象区域(如高频纹理区、边缘区)是非常严重的.

3.2 递归扫描算法

由定义 1 可知,空间树是一个自相似结构,可以表示成若干子空间树和根节点的集合.下面将给出表示空间树 $D(x,y)$ 自相似性质的递归定义.

定义 7 系数空间树 $D(x,y)$ 的递归形式定义为

$$D(x,y) = \begin{cases} \text{当 } c(x,y) \in LL_n \text{ 时,} \\ \{c(x,y), D(x+l,y), D(x,y+l), \\ D(x+l,y+l)\} \\ \text{当 } c(x,y) \in \{HL_n, LH_n, HH_n, \dots, \\ HL_2, LH_2, HH_2\} \text{ 时,} \\ \{c(x,y), D(2x,2y), D(2x+1,2y), \\ D(2x,2y+1), D(2x+1,2y+1)\} \end{cases} \quad (9)$$

为讨论方便,这里假定

$$D(x,y) = \{c(x,y), D_1, D_2, D_3, D_4\}$$

其中, $D_i (i=1,2,3,4)$ 表示子空间树,它们的根节点为 $c_i (i=1,2,3,4)$, 是 $c(x,y)$ 的子节点.

根据定义 7, 即可给出零树量化编码模型 $C_r(c(x,y), T)$ 的递归形式定义:

定义 8

$$C_r(c(x,y), T) = \begin{cases} \text{POS; } c(x,y) \geq T; \\ \text{NEG; } c(x,y) < -T; \\ \text{IZ; } c(x,y) < T \text{ and } (C_r(c_i, T))! = \\ \text{ZTR or } C_r(c_2, T)! = \text{ZTR or } C_r(c_3, T)! = \\ \text{ZTR or } C_r(c_4, T)! = \text{ZTR} \\ \text{ZTR; } c(x,y) < T \text{ and } C_r(c_1, T) = \\ \text{ZTR and } C_r(c_2, T) = \text{ZTR and } C_r(c_3, T) = \\ \text{ZTR and } C_r(c_4, T) = \text{ZTR} \end{cases} \quad (10)$$

式中的判断计算只需涉及根节点系数 $c(x,y)$ 的系数值和直接子节点系数 $c_i (i=1,2,3,4)$ 的量化符号,如图 3(b)所示.

定义 8 给出了零树量化编码模型 $C_r(c(x,y), T)$ 的递归扫描算法,其主要目的是为了解决重复扫描问题.根据定义 8, 可把某个系数 $c(x,y)$ 的零树编码过程分解为其后代系数的零树编码过程,即可以在对根系数的零树编码中得到其所有后代子节点在本层量化中的零树编码符号.本文的做法是将这些符号作为中间结果保留下来,即在本量化层时,对后代子节点进行编码时,就不需要进行扫描编码了,只要直接输出编码符号即可.这样就避免了对系数的重复扫描.

如果单纯就一个系数的零树编码过程来看,虽然定义 8 和定义 6 所给出方法的系数扫描次数是相同的,但区别就在于,定义 8 的方法,由于在一次扫描编码中能够得到所有后代节点的编码符号,因而可以用空间换取时间的方法来消除重复扫描,以提高算法计算效率,而定义 6 给出的方法则不能.

4 基于量化符号邻域相关性 Markov 模型的自适应算术编码

自适应算术编码的主要作用是利用量化符号之间存在的相关性来提高压缩比. EZW 算法的自适应算术编码由于主要利用了量化符号在空间树中的相关性^[1] 而忽略了邻域相关性,故压缩效率提高不明显^[1].

由于修正量化模型 $C_r(c(x,y), T)$ 输出的符号 $\{1,0\}$ 接近平均分布,故本文主要利用零树编码符号的邻域相关性.考虑零树量化编码符号的如下几个邻域统计规律:(1)零量化值邻域:与图象中连续光滑点对应的量化值往往为 1Z 或 ZTR,而连续光滑点之间则有比较强的相关性;(2)正量化值邻域:量化为 POS 的系数的值比较大,其对应于图象边缘点的系数之间,也有较强的相关性;(3)负量化值邻域:NEG 符号对应负系数,这种系数是由于小波变换的“振铃效应”而引起的,伴随着大系数值出现,其多位于图象的边缘地带,且相互之间的相关性较强.图 4 给出了系数矩阵示意图,图中,1,0 表示修正量化编码模型 $C_{refine}(c(x,y), T)$ 的输出符号, i, p, n, z 分别为主要量化编码模型 $C_{zero}(c(x,y), T)$ 输出符号 IZ, POS, NEG, ITR 的缩写.

1	0	i	p	Z	Z	n	p
1	1	n	Z	Z	Z	Z	Z
Z	Z	i	p	Z	Z	-	-
Z	Z	p	i	p	Z	-	-
-	-	-	-	p	Z	Z	Z
-	-	-	-	Z	Z	Z	Z
-	-	-	-	Z	Z	Z	Z
-	-	-	-	Z	Z	n	Z

图 4 量化符号矩阵

定义 9 对于系数 $c(x,y)$, 有零树量化编码符号 $s(x,y)$ 和邻域 $F(x,y)$.

定义邻域 $F(x,y)$ 上的统计函数 $n(F(x,y), b)$ 为 $n(F(x,y), b) = \text{count}(\{c(i,j) | c(i,j) \in F(x,y), \text{且}\}$

$$(s(i, j) = b \text{ 或 } c(i, j) \in D(m, n), \text{ 且 } s(m, n) = b):) \quad (11)$$

在无歧义情况下, $n(F(x, y), b)$ 可简记为 $n(b)$, 其中, $b \in \{\text{POS}, \text{NEG}, \text{IZ}, \text{ZTR}, 1, 0\}$.

定义 10 对于零树量化编码符号 $s(x, y)$, 这里定义 Markov 模型的条件概率 P 为

$$P(s(x, y) = b) = P(s(x, y) | \theta(F(x, y)) = k) \quad (12)$$

其中, $b = \text{POS}, \text{NEG}, \text{IZ}, \text{ZTR}$, $\theta(F(x, y))$ 是邻域 $F(x, y)$ 的条件函数, 具体为

$$\theta(F(x, y)) = \begin{cases} 1, & n(\text{IZ}) + n(\text{ZTR}) > \sigma_0 \\ 2, & n(\text{POS}) > \sigma_1 \\ 3, & n(\text{NEG}) > \sigma_{-1} \\ 4, & \text{otherwise} \end{cases} \quad (13)$$

其中, $\sigma_{-1}, \sigma_0, \sigma_1$ 分别为负量化值邻域、零量化值邻域、正量化值邻域的判断特征值, 由实验给出.

5 算法实现

由定义 5 可知, 虽然基于单队列的连续累进量化方法在每个量化层中的系数扫描顺序 L_{mrg} 是固定不变的, 但需要注意的是, 在零树量化编码模型中, 当系数 $c(x, y)$ 被编码为 ZTR 符号时, 空间树 $D(x, y)$ 中, 其他系数在本量化层的编码顺序将被忽略. 另外, 在量化层内还存在着扫描顺序的动态变化. 这种顺序变化在算法中, 可用一个静态的顺序列表来处理. 当系数 $c(x, y)$ 被量化为 ZTR 符号时, $D(x, y)$ 中其他系数就被从队列中删除.

嵌入式图象编码的一个主要优点是其编码/解码过程可以根据需要在任何时候中止, 并且在最少重复扫描的零树编码算法中, 可以在一次扫描中, 计算出空间树中的所有编码符号. 由于当系数被编码为 POS 和 NEG 符号时, 是不需要对其空间树系数进行全部扫描的, 因此考虑到嵌入式的特点, 本文在递归扫描中, 不采用简单的扫描空间树全部系数的方法, 而是只对需要的系数进行扫描.

本文的自适应算术编码参考了文献[6]的方法, 具体实现不再复述. 下面给出基于单队列递归扫描的嵌入式图象零树编码方法的算法流程:

- (1) 对图象进行小波变化, 计算小波系数矩阵.
- (2) 初始化算术编码频率表, 计算初始量化阈值 $T = T_0$.
- (3) 初始化扫描顺序 L_{mrg} 队列和量化符号缓存区, 令当前系数 $c(x, y)$ 等于 L_{mrg} 队列第 1 个系数.
- (4) 如果 $c(x, y)$ 对应的量化符号缓存不为空,

则转步骤(5); 否则, 进行量化编码 $C(c(x, y), T)$ 计算, 以得到编码符号. 如果 $c(x, y)$ 是主要系数, 则进入零树量化编码模型 $C_r(c(x, y), T)$, 进而计算零树编码符号, 并将编码符号存入量化符号缓存区; 否则, 进入修正量化编码模型 $C_i(c(x, y), T)$, 并计算修正编码符号.

(5) 将量化符号进行算术编码, 取下一个系数, 并进行如下检查:

- ① 如果编码停止条件满足, 转步骤(6);
- ② 如果到队列尾, 则转步骤(3);
- ③ 否则, 转步骤(4).
- (6) 结束量化过程.

6 实验结果

为了验证上述算法的正确性, 本文进行了如下 3 方面的实验. 在实验中的小波变换采用 5 级 18/10 的 Daubchies 小波变换.

(1) 连续累进量化算法的单队列方法和双队列方法的率失真特性比较实验. 以 Lena 图象为例, 不考虑算术编码方法, 其计算的两种连续累进量化方法的率失真差曲线如图 5 所示. 图中圆点处对应量化层分界点. 实验结果表明, 率失真差曲线除在压缩率 0.05bpp 之前有较大振幅外, 在以后的压缩率中都为正, 其中, 最高达 0.43dB, 由此可见, 率失真特性改善明显.

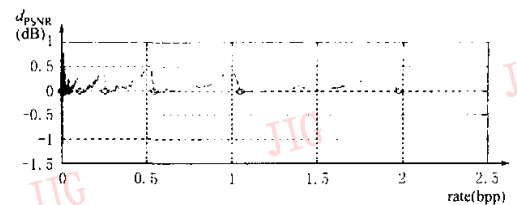


图 5 单队列方法和双队列方法的率失真差 d_{PSNR} 曲线

(2) 零树量化编码模型 $C_r(c(x, y), T)$ 的扫描重复率比较实验. 扫描重复率的计算方法为:

$$d = (n_{scan} - n_r) \times 100 / n_r \quad (14)$$

其中, d 为扫描重复率, n_{scan} 为扫描次数, n_r 为队列 $\langle L_r, Z \rangle$ 中的系数个数. 表 1 给出了在对图 Lena 和 barbara 进行零树编码时, 两种算法在各量化层中扫描重复率. 由于零树量化编码模型的递归扫描算法没有重复扫描, 因此比直接扫描算法提高效率 3%~20%.

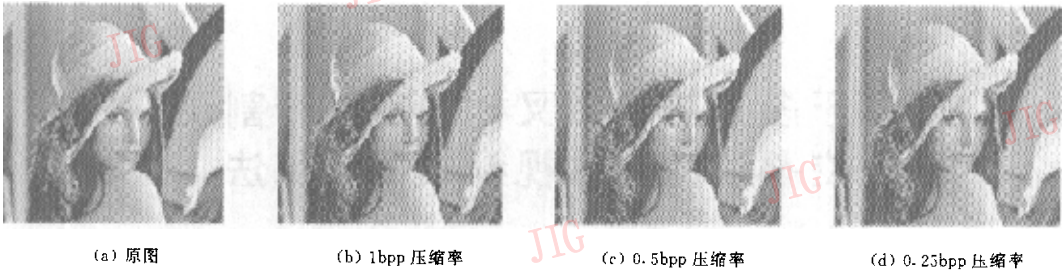


图 6 Lena 原图和本文算法不同压缩率的恢复图

表 1 零树量化编码模型的扫描重复率

量化层	输出符号数	扫描重复率(%)	
		直接扫描算法	递归扫描算法
4	261 821	3.10	0
5	261 460	5.18	0
6	260 303	6.13	0
7	257 643	7.44	0
8	252 277	8.38	0
9	241 419	16.48	0
10	216 504	18.99	0
11	160 865	13.36	0

(3)整体图象编码方法的压缩效率比较实验.表 2 给出了 EZW 算法、SPIHT 以及本文算法的压缩性能数据.由表 2 可见,本文算法的压缩效率较 EZW 算法有明显提高.

表 2 本文算法和 EZW、SPIHT 算法的压缩性能

图象	压缩率 (bpp)	EZW(dB)	本文算法 (dB)	SPIHT(dB)
Lena	0.25	33.17	33.85	34.12
	0.50	36.28	37.00	37.22
	1.00	39.55	39.91	40.41
Barbara	0.25	26.77	27.24	
	0.50	30.53	30.74	
	1.00	35.14	35.89	

7 结 语

实验结果表明,本文提出的嵌入式图象编码算法,在保持高粒度多码率特性的同时,不仅简化了连续累进量化方法的算法结构,而且无需使用动态队列,同时由于有效地解决了零树量化编码模型的重复扫描问题,因而,算法压缩效率较 EZW 算法有明显提高.另外,本文算法中体现出的算法思想对 SPIHT 等算法也是有效的.目前本文采用的量化编码模型是由零树量化编码模型和修正量化编码模型构成的复合模型,由于也存在零树量化编码模型定义比较简单的问题,因而压缩性能比 SPIHT 算法略

差.为此,改进量化编码模型,进一步提高压缩效率是下一步需进行的工作.

参 考 文 献

- Shapiro J M. Embedded image coding using zerotrees of wavelet coefficients[J]. IEEE Trans. Signal Processing, 1993,41(12): 3445~3462.
- Said A, Pearlman W A. A new fast, and efficient image code based on set partitioning in hierarchical trees[J]. IEEE Trans on Circuits System Video Technology, 1996,6(3):243~249.
- ISO IEC 14496-2. Information technology coding of audio-visual objects: visual. Committee Draft [S.]. ISO/IEC JTC1/SC29/WG11, N2202, March 1998.
- Liang J, Yu J, Wang Y *et al.* Fine granularity scalable video coding using combination of MPEG4 video objects and still texture objects[A]. In contribution to 45th MPEG meeting[C], Atlantic City, USA, October 1998.
- Antonio Ortega and Kannau Ramchandran. Rate distortion methods for image and video compression [J]. IEEE Signal Processing Magazine, 1998,15(11):23~50.
- Witten I H, Neal R, Cleary J G. Arithmetic coding for data compression[J]. Comm. ACM. 1987,30(6):520~540.



张海翔 1973 年生,博士研究生,当前研究方向为图象压缩、多媒体技术、计算机网络、软件工程.



陈 纯 1955 年生,博士生导师,计算机系主任.当前研究方向为 CAD/CAM、多媒体计算技术、计算机网络.



庄越挺 1965 年生,博士生导师,当前研究方向为网络多媒体、智能 CAD、数据库与信息检索、智能动画.