

截断点可行性的率失真优化算法

周映虹^{①②} 马争鸣^①

^①(中山大学电子与通信工程系 广州 510275)

^②(广东工业大学信息工程学院 广州 510006)

摘要: 该文提出了截断点可行性的率失真优化算法, 该算法根据率失真斜率最陡下降准则对精细编码过程相关联的截断点进行重新排序。新算法较 JPEG2000 减少了超过 50% 的非可行性截断点; 获得比 JPEG2000 更精细的嵌入式比特流; 相同的压缩比下重建图像的 PSNR 比 JPEG2000 高 0.1dB~0.2dB; 耗费的比特平面位编码时间少于 JPEG2000。

关键词: JPEG2000; 率失真斜率; 可行性截断点; 堆排序

中图分类号: TN919.81

文献标识码: A

文章编号: 1009-5896(2009)04-0997-04

A Feasibility Strategy of Truncation Points for Rate Distortion Optimality

Zhou Ying-hong^{①②} Ma Zheng-ming^①

^①(Department of Electronic & Communication Engineering, Sun Yat-sen University, Guangzhou 510275, China)

^②(Information Engineering College, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: A feasibility strategy of truncation points for rate distortion optimality is presented. The new algorithm reorders the positions of truncation points associated with refinement passes according to the steepest rate-distortion slope. The proposed scheme reduces the unfeasible truncation points to 50% compared with JPEG2000; gives a finely embedded bitstream, achieves an improvement of PSNR over JPEG2000 0.1dB~0.2dB, and attains a speed-up of the bit-plane encode time compared with JPEG2000.

Key words: JPEG2000 standard; Rate distortion slope; Feasible truncation point; Heap sorting

1 引言

JPEG2000^[1]是最新的静态图像编码国际标准, 其压缩性能非常优异。由于该标准只是明确定义了解码一致的比特流语法结构, 因此其编解码技术存在较大改进空间: 文献[2]用最小码长的准则对 JPEG2000 的上下文量化进行优化; 文献[3]以最小条件熵为准则考察 JPEG2000 的上下文量化方案, 文献[4]使用不同子带以及同一子带内的邻域系数来建立上下文模型; 文献[5]提出上下文信息中去除当前比特平面位的重要性信息来提高编码速度; 文献[6]根据率失真斜率估计值来安排编码次序; 文献[7,8]提出了 JPEG2000 率失真优化的快速算法; 文献[9]根据特定的优先次序对码块进行比特平面编码, 避免对所有编码过程进行率失真斜率计算; 文献[10]在对数域中对率失真斜率进行快速比较, 避免了率失真斜率浮点模式的运算。

JPEG2000 采用 EBCOT(Embedded Block Coding with Optimized Truncation)算法^[11]作为核心编码器来进行码块编码以及码率控制。EBCOT 中, 每个子带被划分成多个码块, 然后每个码块进行比特平面编码。每个比特平面位有重要性编码过程 SP(Significance Pass)、精细编码过程

RP(Refinement Pass)以及清除扫描编码过程 CP(Cleanup Pass)3 个编码过程。每个编码过程的结束点是输出比特流的一个截断点, 因此比特平面编码提供了大量截断点。然后, JPEG2000 使用 PCRD-opt 算法^[11]来实现码率优化分配。简言之, 不考虑量化步骤, JPEG2000 的率失真优化主要由以下两个方面构成: 一是比特平面编码产生的截断点; 二是目标码率下总体失真最小的准则下, 选择或快速选择出每个码块的最终截断点。然而, 许多文献[6-10]仅关注后者。本文的出发点是, 在不改变 JPEG2000 的编码框架前提下, 通过对上述的第 1 方面即 JPEG2000 中产生的截断点的研究, 提出了截断点可行性的率失真优化算法, 从而实现对 JPEG2000 的嵌入式率失真性能的改善。本文结构如下: 第 2 节描述新算法并给出实验结果, 第 3 节是结束语。

2 截断点可行性的率失真优化算法

JPEG2000 中, 每个比特平面的截断点依次出现的顺序是: SP, RP, CP。截断点高于或者正好位于率失真曲线的凸壳上。可行性截断点指的是正好位于凸壳曲线上的截断点, 若不加特别指明, 下文所述的非可行性截断点指的是高于凸壳曲线的截断点。如果所有的截断点都是按照率失真斜率最陡下降排列, 则所有的截断点都是可行性截断点。然而,

从表 1 的第 2, 3, 4 和 5 列可见, JPEG2000 这种固定产生截断点的模式产生了不少非可行性截断点。若调整截断点的位置使得截断点尽可能按照率失真斜率最陡下降排序, 则可提高编码器的率失真嵌入式性能。重排后原来的非可行性截断点变成了可行性截断点, 截断码流的嵌入式性能与调整前相比得以提高。本文称该重排过程为截断点可行性化过程。

那么, 如何调整 JPEG2000 中截断点的位置呢? 我们观察可通过改变 RP 的位置来减少非可行性截断点的个数, 主要原因有二。首先, 从表 1 看来, JPEG2000 中属于 RP 的非可行性截断点的个数在总的非可行性截断点的个数中所

占比最大, 若能减少该过程的非可行性截断点个数则编码器嵌入式性能会有较大提高; 其次, SP 以及 CP 是互相依赖的: 当前比特平面位的 CP 产生的数据流依赖当前比特平面位的 SP 产生的数据流, 而当前 SP 过程产生的数据流又依赖于先前比特平面 SP 以及 CP 产生的数据流。因此调整 SP 以及 CP 的位置会导致解码端无法正确解码。而 RP 在解码端得以正确解码的前提下, 除了原有的位置, 还有 3 种不同的位置可供选择的: 见图 1(a), 1(b), 1(c), 详细的算法流程见 2.1 节。

表 1 在 JPEG2000 以及新算法中出现的非可行性截断点的统计

图像	JPEG2000 ^[12]				本文算法				
	SP	RP	CP	Total	SP	RP	CP	Total	Δ total
Yacht	40/997	116/997	37/997	193/997	35/997	14/997	35/997	84/997	109/997
Barbara	54/1069	152/1069	41/1069	247/1069	50/1069	12/1069	41/1069	103/1069	144/1069
Lena	57/1009	127/1009	40/1009	224/1009	49/1009	17/1009	39/1009	105/1009	119/1009
Peppers	44/1111	142/1111	30/1111	216/1111	40/1111	11/1111	30/1111	81/1111	135/1111
Goldhill	56/1039	159/1039	30/1039	245/1039	49/1039	12/1039	30/1039	91/1039	163/1039
Mandrill	65/1207	164/1207	36/1207	265/1207	67/1207	8/1207	36/1207	111/1207	154/1207
LightHouse	75/1516	233/1516	80/1516	388/1516	71/1516	62/1516	79/1516	212/1516	176/1516
百分比(%)	4.9	13.8	3.7	22.4	4.5	1.7	3.6	9.9	12.6

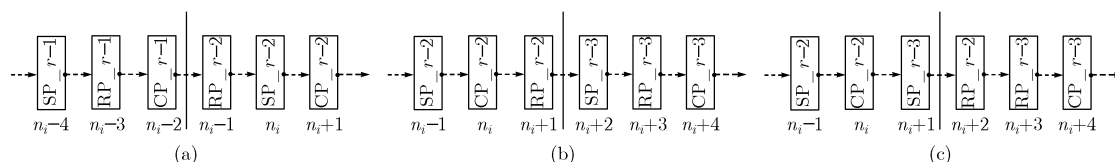


图 1 (a)-(c) 精细编码过程的 3 种位置调整方案

2.1 新算法流程

编码参数设置为: lossy mode, 5 levels of 9/7 wavelet transform, block sizes of 64×64, termall mode。JPEG2000 采用 Generalized Lagrange Multiplier (GLM) 和二分化来寻找可行性截断点的优化集合。文献[7]使用基于堆排序的贪婪算法来选择目标码率下可行性截断点的优化集合。使用堆排序的贪婪算法和使用 GLM 算法得到的最终被包括进输出码流的编码过程几乎是相同的, 但由于堆排序的贪婪算法的复杂度比 GLM 的复杂度小, 因此我们采用堆排序的贪婪算法来取代 GLM 算法。新算法 FSTPrdo 的流程如下:

Algorithm FSTPrdo (int R^{\max} , set $\{B_i\}$)

```

int R; heap h; segment s; blk b; int n; bufferpoint p_b;
for each  $b \in \{B_i\}$  do
    encode ( b, 5 );
    feash_points( b, 5 );
    init_bufferpoint( b, 4, p_b );
    convexhull(b, p_b);

```

```

h = makeheap( B_i );

```

```

R = 0;

```

```

while ( R < Rmax ) do

```

```

    s = the first unmarked feasible truncation point in

```

```

bufferpoint  $p_{\text{root}(h)}$ ;

```

```

    R = R + ratesize(root(h));

```

```

    encode_packet( B_i );

```

```

    after selecting the first unmarked feasible truncation
    point, set  $n$  = the number of the coding passes in

```

```

bufferpoint  $p_{\text{root}(h)}$ ;

```

```

    if  $n < 3$  then

```

```

        encode ( root(h), 3 );

```

```

        feash_points(root(h), 3);

```

```

        update_bufferpoint(root(h),  $p_{\text{root}(h)}$ );

```

```

    convexhull(root(h),  $p_{\text{root}(h)}$ )

```

```

    update(root(h))

```

```

    siftdown(root(h), key(root(h)), h);

```

上述伪代码中, heap h 是满足二叉排序堆特性的树结

构, makeheap 函数负责把 h 中结点进行排序使之满足堆特性; sifttdown 函数负责调整 h 使之满足堆特性; update (root(h))函数把根结点的内容更新为根结点对应的码块的第一个没被选择的可行性截断点; 每个码块 b 中, 经过截断点可行性化调整、明确所在位置且仍没有被堆排序选择的已编码子过程放置在子过程缓冲池 p_b 里, 而 convexhull(b, p_b) 函数则对子过程缓冲池 p_b 里的已编码子过程进行 convex hull 分析, 找出第一个可行性截断点; update_bufferpoint (root(h), $p_{\text{root}(h)}$)更新子过程缓冲池 $p_{\text{root}(h)}$; encode($b, 5$)指的是对码块 b 的下 5 个子过程进行熵编码; encode_packet (B_i)函数指的是对所有码块进行比特流打包。除了遵循 JPEG2000 规定的编码语法以外, 需要额外负载来说明哪些码块的精细编码过程的位置作了哪种调整。只需要对码块中最终被选择进来的精细编码子过程的位置用边信息注释: 如果位置需要调整, 有 3 种位置调整可能。那么, 用 1bit 来标注是否调整位置, 用 2bit 来标注具体的调整位置。feasb_points 是截断点可行性的过程函数。对于码块 b , λ_{n_b} 指的是截断点 n_b 对应的率失真斜率, $\lambda_{(n_b+1, n_b+2)}$ 指的是截断点 n_b+1 以及 n_b+2 对应的率失真斜率。 f_{n_b} 是截断点 n_b 是否作了调整的标识, 初始化为 0。 N_b 是码块 n_b 中的截断点总数, $0 \leq n_b < N_b$ 。若 $(n_b \% 3) == 2$ 则对应的是编码过程 RF。以下是 feasb_points 的伪代码:

```

if  $n_b$  corresponding to RP and has not been
adjusted yet
    if  $\lambda_{n_b}$  not more than 0
        continue;
    if  $\lambda_{n_b} > \lambda_{n_b-1} \ \&\& \ \lambda_{n_b} < \lambda_{n_b-2} \ \&\& \ f_{n_b}, f_{n_b-1} == 0$ 
        exchange positions of  $n_b$  and  $n_b-1$  (as shown
in Fig1.(a)), and then set  $f_{n_b}, f_{n_b-1} = 1$ 
    elseif  $\lambda_{n_b} < \lambda_{n_b+1} \ \&\& \ \lambda_{n_b} > \lambda_{n_b+2} \ \&\& \ f_{n_b}, f_{n_b+1} == 0$ 
        exchange positions of  $n_b$  and  $n_b+1$  (as shown in
Fig.1(b)), and then set  $f_{n_b}, f_{n_b+1} = 1$ 
    elseif  $\lambda_{n_b} < \lambda_{(n_b+1, n_b+2)} \ \&\& \ f_{n_b}, f_{n_b+1}, f_{n_b+2} == 0$ 
         $n_b$  displace the position of  $n_b+2$ ,  $n_b+2$  move
to the original position of  $n_b+1$ , and  $n_b+1$  shift to the
original position of  $n_b$  (as shown in Fig.1(c)) and then set
 $f_{n_b}, f_{n_b+1}, f_{n_b+2} = 1$ 
    else
        set  $f_{n_b} = 1$ 

```

2.2 实验结果及讨论

灰度图像 Yacht(512×480), Lena(512×512), Peppers (512×512), Barbara(512×512), Goldhill(512×512), Mandrill(512×512)以及 Lighthouse(512×768)是本文中使用的 7 个测试图像。

表 1 分别统计了在 JPEG2000 以及在新算法下得到的非

可行性截断点个数与总的截断点个数相比的比率。从表 1 可见, JPEG2000 产生的非可行性截断点中, 属于 RP 产生的占大多数。新算法框架下, SP 以及 CP 产生的非可行性截断点数量与 JPEG2000 算法相比, 略有减少。最为明显的是, 与 JPEG2000 相比, 新算法大大减少了 RP 中产生的非可行性截断点的个数, 从而使得总的非可行性截断点个数下降了超过 50%。JPEG2000 算法是在优化截断的过程前对所有码块的比特平面位编码 (即 Tier1 编码) 完毕后再进行优化截断。由于相邻可行性截断点之间的编码子过程个数 98% 以上都是 3 个子过程以内^[7], 因此新算法每次只进行 5 个或者 3 个编码子过程的编码, 然后对已编码的子过程进行优化截断。从表 2 可见, 低码率的情况下, 新算法减少了那些没有被包括在最终输出码流的编码子过程的编码, 从而节省了大量比特平面位的编码时间。其中, 表 2 里的 T1%指的是 JPEG2000 编码器中 Tier1 编码所耗费时间占整个 JPEG2000 编码器耗费的百分比; P%指的是新算法节省的编码子过程个数占 Tier1 编码子过程个数的百分比; T%指的是新算法在 Tier1 编码中所节省的编码时间。

表 2 与 JPEG2000 相比新算法节省的子过程及比特平面位编码时间的百分比

	bpp	T1%	P%	T%
Yacht	0.0625	50.3	61.8	67.2
	0.125	50.3	57.9	67.1
	0.25	47.8	52.3	67.0
	0.5	47.8	44.4	50.0
Barbara	0.0625	56.0	63.9	70.0
	0.125	56.0	60.8	66.6
	0.25	53.8	55.2	62.1
Lena	0.5	48.1	46.0	47.0
	0.0625	49.9	59.0	63.6
	0.125	50.1	55.4	57.9
Peppers	0.25	50.0	47.7	53.5
	0.5	49.8	39.7	52.5
	0.0625	56.1	62.2	71.8
Goldhill	0.125	56.0	57.9	65.2
	0.25	53.8	50.4	57.2
	0.5	49.8	35.8	43.6
Mandrill	0.0625	56.0	61.1	70.0
	0.125	56.0	57.6	66.6
	0.25	53.8	53.6	63.2
Lighthouse	0.5	51.9	47.5	50.2
	0.0625	61.9	68.2	71.1
	0.125	61.9	66.2	70.5
Lighthouse	0.25	59.9	65.0	69.8
	0.5	57.9	59.0	60.2
	0.0625	54.1	61.5	63.4
Lighthouse	0.125	54.1	60.0	63.3
	0.25	52.7	54.4	54.6
	0.5	51.4	44.1	45.3

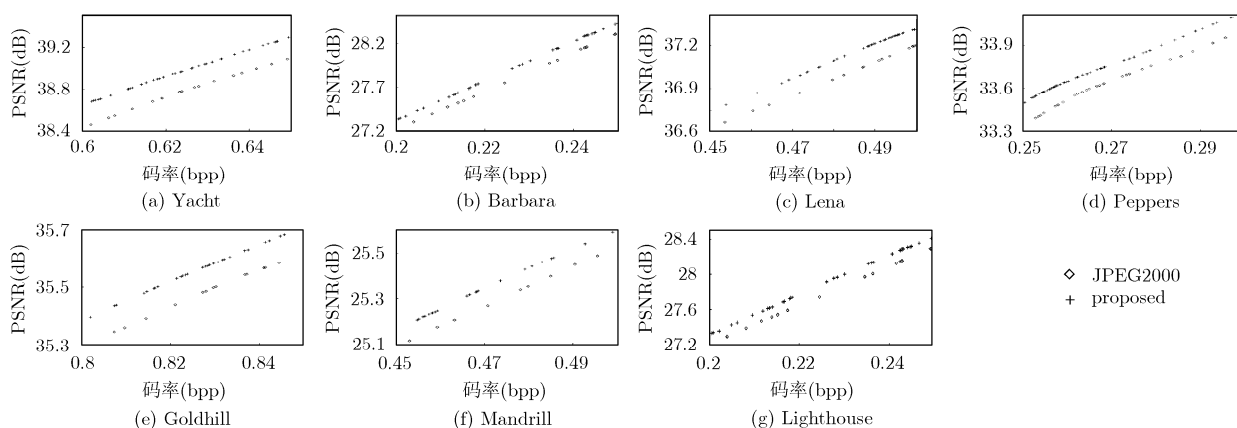


图2 新算法与JPEG2000率失真嵌入式性能比较

以上述7个图像作为测试图像,图2是新算法获得的重建图像的PSNR值与JPEG2000获得的重建图像的PSNR值的比较,图2中的标记点代表的是编码器输出码流的实际码率。从图2可见新算法获得的PSNR值比JPEG2000获得的PSNR值高0.1dB~0.2dB(在0bpp~1bpp下);新算法比JPEG2000提供更多的可行性截断点,其输出码流的目标码率更精细,率失真性能更高。

3 结束语

本文提出了通过截断点的重新排序来实现截断点可行性的率失真优化算法。与JPEG2000相比,新算法的优点在于:减少了超过50%的非可行性截断点,从而使得输出码流的目标码率更精细;耗费的编码时间较少;同时获得了更高质量的重建图像。

参考文献

- [1] ISO/IEC 15444-1: Information Technology—JPEG 2000 Image Coding System—Part1: Core Coding System [S].
- [2] Krivoulets A, Wu Xiaolin, and Forchhammer S. On Optimality of context modeling for bit-plane entropy coding in the JPEG2000 standard [C]. In: 8th International Workshop Very Low Bitrate Video Coding, Madrid, Spain, 2003: 208–216.
- [3] Liu Zhen and Karam Lina J. Mutual information-based analysis of JPEG2000 contexts [J]. *IEEE Trans. on Image Processing*, 2005, 14(4): 411–422.
- [4] 伞兴, 蔡华, 吴秀清. 基于时、频域上下文模型的图像压缩算法[J]. 小型微型计算机系统, 2006, 27(8): 1556–1559.
San Xing, Cai Hua, and Wu Xiu-qing. Image coding based on spatial and frequency context model [J]. *Journal of Chinese Computer Systems*, 2006, 27(8): 1556–1559.
- [5] 田冬庚, 赵亦工, 孙文方. 基于JPEG2000一种上下文生成的改进方法 [J]. 遥测遥控, 2006, 27(2): 37–41.
- [6] Tian Dong-geng, Zhao Yi-gong, and Sun Wen-fang. An improved scheme of forming context based on JPEG2000 [J]. *Journal of Telemetry, Tracking and Command*, 2006, 27(2): 37–41.
- [7] Vikram K N, Vasudevan V, and Srinivasan S. Rate-distortion estimation for fast JPEG2000 compression at low bit-rates [J]. *Electronics Letters*, 2005, 41(1): 16–18.
- [8] Yu Wei, Sun Fangting, and Fritts E. Efficient rate control for JPEG-2000 [J]. *IEEE Trans. on Circuits and Systems for Video Technology*, 2006, 16(5): 577–589.
- [9] Du Weina, Sun Jun, and Ni Qiang. Fast and efficient rate control approach for JPEG2000 [J]. *IEEE Trans. on Consumer Electron.*, 2004, 50(4): 1218–1221.
- [10] 王立, 伍宗富. JPEG2000压缩率控制算法的研究 [J]. 计算机仿真, 2006, 23(6): 147–149.
Wang Li and Wu Zong-fu. Study of rate control for JPEG2000 image coding [J]. *Computer Simulation*, 2006, 23(6): 147–149.
- [11] 杜伟娜, 孙军, 倪强. 基于JPEG2000的高效率控制算法 [J]. 上海交通大学学报, 2006, 40(1): 16–19.
Du Wei-na, Sun Jun, and Ni Qiang. An efficient rate control scheme for JPEG2000 [J]. *Journal of Shanghai Jiaotong University*, 2006, 40(1): 16–19.
- [12] Taubman D. High performance scalable image compression with EBCOT [J]. *IEEE Trans. on Image Processing*, 2000, 9(7): 1158–1170.
- [13] Adams D. Official reference implementation of the JPEG-2000 Part-1 codec. [Online] <http://www.ece.uvic.ca/~mdadams/jasper/>. 2007.Available.

周映虹: 女, 1978年生, 博士, 研究方向为图像与视频压缩。

马争鸣: 男, 1957年生, 教授, 研究方向为通信与信息系统。