

文章编号:1001-9081(2006)10-2386-03

一种基于 H. 264 与近邻预测的无损视频压缩 H. 264-LS

叶迎宪,都思丹

(南京大学 电子科学与工程系,江苏 南京 210093)

(dsd@nju.edu.cn)

摘要:提出了一种无损视频压缩方案 H. 264-LS。该方案为适应无损压缩的需要,根据运动补偿后残差系数的特点,使用了一种二维近邻预测技术以取代 H. 264 原有的变换。实验测试结果表明,该算法的整体表现要优于现有的一些无损视频压缩方案,特别是对于运动量大的视频序列优越性很突出。

关键词:H. 264; 无损压缩; 近邻预测

中图分类号:TP391.41 **文献标识码:**A

Lossless video compression H. 264-LS based on H. 264 and neighbor prediction

YE Ying-xian, DU Si-dan

(Department of Electronic Science & Engineering, Nanjing University, Nanjing Jiangsu 210093, China)

Abstract: A lossless video compression method H. 264-LS was proposed in this paper. To meet the needs of lossless compression and according to the features of residual coefficients after motion compensation, a two dimensional neighbor prediction technology was used instead of the H. 264's integer transform. The experimental results demonstrate that the proposed approach is, on the whole, better than other existing lossless video compression algorithms, especially in case of heavy movement.

Key words: H. 264; lossless compression; neighbor prediction

0 引言

近年来,数字图像与视频压缩有了很大的发展,先后出现了 MPEG-4、JPEG2000、H. 264/MPEG-4 AVC 等先进的压缩标准。这些标准大多数着眼于有损压缩上。然而在特定的应用场合中,无损视频压缩显得更为重要。例如对于航拍视频,再细小的损失都可能造成重要地理信息的遗漏。对数字视频无损压缩的研究逐步得到了人们的重视。

数字视频的压缩从三个方面去除原始视频的冗余性:色度、空间和时间。色度冗余度一般通过 RGB 到 YCbCr 的转换来去除。空间冗余度的去除可以通过 DCT、DWT 等变换方法,也可以通过上下文预测的方法。而时间冗余度一般采用运动估计+运动补偿的方法去除。

现有的无损图像压缩方法主要包括了基于 MED (Median Edge Detector) 预测器的 JPEG-LS^[1]、基于整数小波变换的 JPEG2000^[2]以及基于上下文的自适应预测编码 CALIC^[3]。由于这些编码方案只考虑了图像的空间冗余性,没有考虑视频序列前后图像之间的空间冗余性,因此直接把它们作用在视频序列的压缩,压缩的效率不会很高。文献[4]中给出了一些结合空间预测的无损视频压缩方案,但是由于使用了固定的 16×16 运动补偿方式,对于运动情况复杂的视频序列,去除空间冗余性还不够有效。要最终实现一个无损视频压缩方案,需要设计出一个高效的熵编码器,但这并不是一件简单的工作。文献[5]设计出一种基于运动补偿的压缩方案,使用了整数到整数的无损离散余弦变换 (I2I-DCT) 来取代 MPEG-1 中原用的 DCT 变换。由于 MPEG-1 的视频压缩方法

还不够成熟,因此压缩率仍有很大的提升空间。文献[9]提出了一种基于 MPEG-2、JPEG-LS 以及算术编码的方案。

本文提出了一种无损视频压缩方案 H. 264-LS,它充分利用了现有 H. 264 压缩标准的先进编码方法,同时使用了近邻预测技术减少预测残差间的相关性。

1 可逆变换分析

H. 264 编码过程中会造成像素精度损失的主要有三个步骤:变换、量化以及去块滤波。量化的主要作用是允许带有误差的压缩系数值的动态范围,为了实现无损压缩必须跳过这一步。去块滤波的主要作用是改善图像的主观质量,减少分块压缩所带来的块效应,在无损压缩中就不再需要这一步骤。

1.1 现有的一些可逆变换

在 H. 264 中使用了 4×4 的整数变换来对宏块的残差进行变换。整数变换的作用相当于取代原来的 DCT 变换。虽然整数变换实现了从整数到整数的变换,中间运算过程中避免了舍入误差,但是在经过逆变换后并不能精确重组原来的系数矩阵,其原因在于它的变换核并不是一个真正的正交矩阵。一个完整的正交变换过程应包括矩阵系数缩放调整这一步,只不过在 H. 264 中把这一步合并在了量化的过程中。

为了实现无损的视频压缩,可以有两种选择:直接跳过变换;使用精确可逆的变换取代 H. 264 原来的整数变换。目前已有的精确可逆图像变换包括:

1) 在 JPEG2000 无损压缩中使用的 W5/3 小波变换^[2],其一维变换公式为:

收稿日期:2006-04-11;修订日期:2006-07-11

作者简介:叶迎宪(1982-),男,广东广州人,硕士研究生,主要研究方向:图像编码、视频应用; 都思丹(1963-),女,江苏南京人,教授,博士生导师,博士,主要研究方向:图像工程、模式识别。

$$\begin{cases} d_{j+1,i} = s_{j,2i+1} - \left\lfloor \frac{s_{j,2i} + s_{j,2i+2}}{2} \right\rfloor \\ s_{j+1,i} = s_{j,2i} + \left\lfloor \frac{d_{j+1,i-1} + d_{j+1,i} + 2}{4} \right\rfloor \end{cases}$$

2) 文献[6]中的可逆 DCT,其四点一维变换可以表示为信号流。

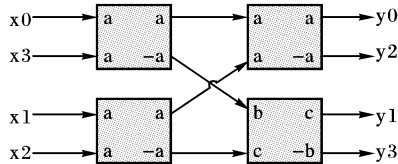


图 1 一维可逆 DCT 信号流

其中 $a = \frac{1}{\sqrt{2}}, b = \cos \frac{\pi}{8}, c = \cos \frac{3\pi}{8}$ 。信号流图上每个方格中的变换都可以利用三步提升实现精确可逆变换。以变换

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} b & c \\ c & -b \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$$

为例,其提升实现为:

$$\begin{cases} t = x_0 + \left\lfloor \frac{c-1}{b} x_1 + 0.5 \right\rfloor \\ y_0 = x_1 + \left\lfloor b \cdot t + 0.5 \right\rfloor \\ y_1 = y_0 + \left\lfloor \frac{c-1}{b} y_0 + 0.5 \right\rfloor \end{cases}$$

符号 $\lfloor \cdot \rfloor$ 表示截断取整。

3) 文献[7]中的二维可逆 DCT。其实该方法是文献[6]可逆 DCT 的一个改进。图 1 中第一阶段变换是一维的两点沃尔什-哈达马变换。由于二维 DCT 中需要在行方向进行一次该变换,然后在列方向进行一次该变换,因此整个 2D-DCT 可以等价为先进行四点的二维沃尔什-哈达马变换,然后分别在行以及列方向上进行图 1 第二阶段的一维变换。四点的二维沃尔什-哈达马变换提升实现步骤为:

$$\begin{cases} t = x_{00} - x_{01} + x_{10} + x_{11} \\ y_{01} = x_{00} - x_{01} + x_{10} - (t \gg 1) \\ y_{00} = x_{01} + (t \gg 1) \\ y_{10} = x_{00} - (t \gg 1) \\ y_{11} = t - y_{01} - y_{00} + y_{10} \end{cases}$$

测试表明,使用上述的无损变换均不能增加编码效率。文献[5]中也有类似的结论,发现无损 DCT 用于帧间编码时不能降低残差系数的熵值。这是因为上述变换的设计主要针对相关性较强的原图像,直接把它们用在经过运动补偿后的相关性较弱的残差系数,效果不太理想。有必要根据残差系数的特点,设计一种新的变换手段。

1.2 二维近邻预测

一个典型的 4×4 残差块有以下特征:1)在大多数情况下,一个系数与相邻系数之间有一定的相关性,例如一个大的误差系数周围很可能有其他非零的误差系数,而且相邻的误差系数之间往往

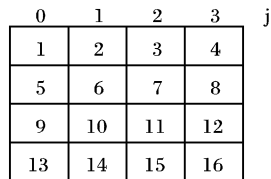


图 2 4×4 块示意图

往往是符号相同的;2)误差系数随着坐标变化得很快,坐标距离超过 2 的像素之间相关性很小。根据这些特征我们设计出了一种二维近邻预测方法以减小系数间的相关性,设预测前输入系数为 $f(i,j)$,预测后输出误差为 $d(i,j)$,其中 $i = 0 \dots 3, j = 0 \dots 3$ 。

1) 对于最左上角的系数 $f(0,0)$,不经过预测直接输出 $d(0,0) = f(0,0)$;

2) 对于最上面一行系数 $f(0,j), j \neq 0$,使用左边的系数进行预测,输出误差 $d(0,j) = f(0,j) - a \cdot f(0,j-1)$;

3) 对于最左面一行系数 $f(i,0), i \neq 0$,使用上边的系数进行预测,输出误差 $d(i,0) = f(i,0) - a \cdot f(i-1,0)$;

4) 对于其他的系数 $f(i,j), i \neq 0, j \neq 0$,同时使用上面以及左边的系数进行预测,输出误差 $d(i,j) = f(i,j) - bf(i-1, j) - cf(i, j-1)$;

由于残差系数之间的相关性不是太强,参数 a, b 应取得比较小。通过试验发现 $a = 0.4, b = 0.35$ 是一组比较合适的选择。这组参数用在了后面的实验测试中。

在实际的操作中,先对最右下角的 $f(3,3)$ 进行预测,然后按照从右到左、从下往上的顺序对其他系数进行预测输出误差。这样的操作顺序可以保证一个系数经过预测后不会再用于其他系数的预测。而解码端使用相反的从左到右、从上往下的顺序通过预测误差重建系数,实现精确的重构。

2 H.264-LS

H.264-LS 编码器结构如图 3 所示。与基本的 H.264 编码器相比,帧内预测、帧间编码、熵编码等技术都得到了保留,可以最大限度的使用 H.264 的高效工具去除视频在空间与时间上的冗余度。

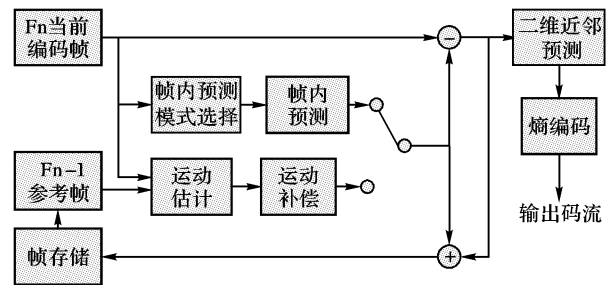


图 3 H.264-LS 视频编码器结构

3 实验测试结果

3.1 可逆变换的测试

我们选取了一些 QCIF 的标准测试序列,编码这些序列的前 100 帧,只有第一帧编码为 I 帧,其余编码为 P 帧,测试下列可逆变换用于 H.264-LS 中的压缩率:1)不对残差系数进行变换;2)对每个 4×4 块使用一层 W5/3 小波变换;3)对每个 4×4 块使用两层 W5/3 小波变换;4)使用文献[6]中的一维无损 DCT;5)使用文献[7]的二维无损 DCT;6)使用二维近邻预测。测试结果如表格 1 所示。

表 1 各种无损变换对压缩率的影响(单位:%)

测试序列	无	一层 W5/3 小波	两层 W5/3 小波	1D-LDCT	2D-LDCT	二维近邻预测
Foreman	35.47	37.84	38.51	37.14	37.10	35.32
Akiyo	11.34	13.19	13.72	11.77	12.57	11.15
Miss AM	33.49	34.35	34.86	34.94	34.22	32.51
Coastguard	38.22	39.83	40.68	38.40	38.93	37.63
Claire	21.10	24.40	24.99	22.13	23.13	20.76
Average	27.93	29.93	30.55	28.87	29.19	27.48

3.2 H.264-LS 与其他无损压缩方案的比较

我们使用了 QCIF 的 Foreman、Akiyo、Miss American、

Coastguard 标准序列的前 100 帧对 H. 264-LS 进行了测试。测试只对第一帧编码成 I 帧,其余帧均编码为 P 帧。运动搜索使用 H. 264 当中的全部 7 种尺寸块,使用的参考帧为 3 个,用 CABAC 进行熵编码。测试的结果(压缩率)与 JPEG-LS、CALIC、I2I-DCT 的无损 MPEG-1 视频压缩(使用两点 I2I-DCT)^[5]、AT SVC-LS 压缩方案^[4]进行了比较。文献[4]中原给出的数据是 bits/per pixel,将这个结果除以 8 得到压缩率。此外,还用了 Flower Garden 以及 Table Tennis 两个分辨率稍高的序列与基于 MPEG-2 无损压缩^[9]进行了压缩倍数的比较。结果如表 2、表 3 所示。

表 2 QCIF 分辨率下压缩率测试结果(单位:%)

测试序列	JPEG-LS	CALIC	I2I-DCT	AT SVC-LS	H. 264-LS
Akiyo	40.00	40.30	16.65	16.93	11.16
Miss American	33.62	33.99	44.61	26.73	32.51
Foreman	55.68	53.97	53.37	42.04	35.32
Coastguard	62.87	62.69	60.37	48.98	37.63
Average	48.04	47.74	43.751	33.67	29.15

表 3 H. 264-LS 与无损 MPEG-2 的比较

序列	分辨率	帧数	无损 MPEG-2	H. 264-LS
Flower Garden	352 × 240	24	1.78	1.92
Table Tennis	352 × 240	24	2.29	2.22

在 QCIF 分辨率测试中可以看出对于 Akiyo、Foreman 以及 Coastguard 序列,H. 264-LS 的压缩率要明显优于其他压缩算法,特别是对于运动比较剧烈的 Foreman 及 Coastguard 测试序列,压缩后的大小比 AT SVC-LS 分别下降了 6.72% 和 11.35%。比较例外的是 Miss American 序列,H. 264-LS 的压缩率只跟 JPEG-LS、CALIC 等静态图像压缩算法相当。造成该现象的主要原因是该测试序列的背景噪声比较大,在运动搜索的过程中比较难找到很好的匹配块,因此压缩效率提高得不多。而时空预测自适应预测相当于对图像做空域以及时域的低通滤波,能滤除一定的噪声,所以效果更佳。

在与无损 MPEG-2 的比较中,H. 264-LS 的表现没有低分辨率下这么突出。对于运动量较大的 Flower Garden 序列压缩率优于无损 MPEG-2,而运动量较小的 Table Tennis 序列则不及无损 MPEG-2。

3.3 算法复杂度测试

将 H. 264-LS 的编码复杂度与 JPEG-LS、使用 I2I-DCT 的 MPEG-1 无损压缩进行对比测试。测试选用了 QCIF 的 Foreman 序列前 100 帧,而测试平台为 P4 1.5GHz。结果如表 4 所示。

表 4 算法编码复杂度测试

压缩算法	压缩时间(ms)
JPEG-LS	1 121
I2I-DCT	2 944
H. 264-LS	7 556

与已有的无损压缩算法相比,H. 264-LLS 编码复杂度有较大幅度的提升,主要与 H. 264 复杂的多种编码方式有关。然而 H. 264-LLS 当中许多运算都可以用并行方式进行,使用 MMX、SSE2 等 SIMD 指令集可以有效的进行优化。

3.4 H. 264 中的部分技术对于压缩率的影响

这一步测试 H. 264 当中的部分编码选项时对压缩率的影响。测试分为四组:1)仅允许使用 16 × 16、16 × 8、8 × 16、

8 × 8 的块进行帧间预测;2)仅允许使用 16 × 16 的块进行帧间预测;3)使用单个参考帧代替三个参考帧;4)使用 CAVLC 代替 CABAC。对 Foreman、Akiyo、Miss American、Coastguard、Claire 五个标准序列的前 100 帧进行测试。测试的结果与原始的编码结果进行比较。

从表 5 可以看到,前三组比特率增加得不是很多,这说明了 H. 264 帧间编码中的新技术在无损压缩中起到的作用不大,如果对压缩时间有比较高的要求,完全可以去除这些选项。第四组测试的结果表明 CAVLC 压缩率明显劣于 CABAC,对不同的测试序列比特率会有 10% ~ 20% 的增加。这主要是因为 CAVLC 原来设计的码表并不针对无损压缩,需要重新设计出专门的码表才可望使压缩率有较好的改进。

表 5 编码选项对压缩率的影响(单位:%)

测试序列	原始	第一组	第二组	第三组	第四组
Foreman	35.32	35.43	35.66	35.54	40.39
Akiyo	11.16	11.17	11.19	11.17	13.10
Miss American	32.51	32.52	32.60	32.58	38.53
Coastguard	37.63	37.82	38.22	37.78	42.40
Claire	20.76	20.78	20.83	20.78	25.31
Average	27.48	27.54	27.70	27.57	31.94

4 结语

本文提出了一种基于 H. 264 的无损视频压缩方案 H. 264-LS,采用了新的二维近邻预测取代原有的整数变换。实验测试表明,近邻预测能使预测残差的熵值有所降低,而 H. 264-LS 的压缩率在总体上要优于现有的一些无损视频压缩方案。H. 264-LS 实现起来并不太复杂,可以充分利用现有的一些开源的 H. 264 工程,而且可以根据应用划分不同的等级,使得算法在复杂度和压缩率之间取得平衡。

参考文献:

- [1] MARCELO JW, GADIEL S, GUILLERMO S. The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS[J]. IEEE Transaction on Image Processing, 2000, 9(8): 1309 - 1322.
- [2] MICHAEL WM, MICHEAL JG, ALI B, et al. An overview of JPEG-2000[A]. Proceedings of IEEE Data Compression Conference [C]. 2000. 523 - 541.
- [3] XIAOLIN W, NASIR M. Context-based adaptive lossless image coding[J]. IEEE Transaction on Communications, 1997, 45(4): 437 - 443.
- [4] 张明锋,胡佳,张立明. 基于时空自适应预测的无损视频压缩[J]. 计算机工程与科学, 2004, 26(10): 49 - 52.
- [5] CHARITH G, ABHAYARATNE K. N-point discrete cosine transforms that map integers to integers for lossless image/video coding [A]. Proceedings of Picture Coding Symposium (PCS 2003) [C]. France (St. Malo) 2003. 417 - 422.
- [6] KUNIROSHI K, KAORU S. Reversible discrete cosine transform [A]. Proceedings of IEEE ICASSP98 [C]. 1998. 1769 - 1772.
- [7] KUNIROSHI K, KAORU S. 2D lossless discrete cosine transform [A]. Proceeding of IEEE ICIP2001 [C]. 2001. 466 - 469.
- [8] ITU, ITU-T Recommendation H. 264: Advanced video coding for generic audiovisual services[Z]. March 2005.
- [9] XIA J, JIANG J, YANG SY, et al. An empirical study to turn MPEG-2 into lossless video compression [A]. Visual Information Engineering 2003 (VIE 2003) [C]. 2003. 234 - 237.