

一种改进的 H.264 Intra 4 × 4 帧内预测算法

韦 虎¹, 林 涛², 林争辉³

(上海交通大学电子工程系, 上海 200240)

摘要: H.264 帧内编码通过空间预测减小变换编码输入值的大小, 提高编码压缩性能。由于块编码总按照从左往右和从上往下的顺序, 帧内预测可选的参考像素限制于块的左边和上边, 因此预测结果不理想。该文提出一种基于宏块内部灵活的子块编码顺序的 Intra 4 × 4 预测编码算法, 自适应地选择最佳预测参考像素集, 同时对空间预测模式进行了改进。通过有限的子块分组和子块顺序选择, 可以实现宏块内部编码最优化。仿真试验表明, 和 H.264 相比, 该算法可以取得平均 0.4 dB 的 PSNR 编码性能增益。

关键词: 帧内预测; 灵活的子块编码顺序; 参考像素集

Improved Intra 4 × 4 Prediction Algorithm for H.264

WEI Hu¹, LIN Tao², LIN Zheng-hui³

(Dept. of Electronic Eng., Shanghai Jiaotong University, Shanghai 200240)

【Abstract】 In H.264, intra prediction is employed to reduce energy of prediction error. The order of Intra block encoding is from left to right horizontally and from top to bottom vertically, so the prediction reference pels can only appear on the left side or upside of the current block, without ideal predicted result sometimes. In this paper, a new Intra 4 × 4 prediction algorithm based on flexible order of subblocks encoding is introduced, with improved prediction modes and a new set of reference pels. It realizes RD optimization of intra Macroblock(MB) encoding by mode decision of subblock grouping and subblock ordering. Experimental results show that compared with the standard intra prediction mode of H.264, this method can gain 0.4 dB of PSNR averagely.

【Key words】 intra prediction; flexible subblock encoding order; set of reference pels

1 概述

H.264 帧内编码采用新的空间域预测编码技术, 利用图像的空间相关性, 选择几种预定方向中的一个来预测 16×16, 8×8 或 4×4 待编码块, 以降低需传输的预测残差的大小, 极大地提高了编码性能。而随着 H.264 在高清晰度视频领域(如 HDTV)的应用, 对帧内编码性能的进一步提高仍然有迫切的需求。因此, 近年来在研究快速算法来降低帧内编码复杂度的同时, 对提高 H.264 的帧内预测率失真性能也进行了许多研究^[1-3]。

文献[1]通过增加预测方向和进行插值获得更多参考像素来提高帧内预测性能; 文献[2]利用预测的残差仍然具有方向性的特点, 在变换前进行多种模式前期处理来提高编码性能。文献[3]中打破了宏块的限制, 改变子块编码顺序, 优先编码相邻几个宏块内的上方子块, 减少了子块右上侧参考像素不可用的几率。

而以上方法所进行的空间预测只利用了预测块上方和左方的相邻像素, 当预测块的纹理方向在块上方和左方边缘处发生了变化, 或者由于量化误差较大使块左方和上方相邻像素与预测块之间的相关性减弱时, 预测的结果经常不理想。此外, 空间预测的引入使预测块的率失真性能还受到已编码块率失真性能的影响, 对此不进行考虑也会影响整个帧的率失真性能。本文提出一种基于宏块内部灵活的子块编码顺序的 Intra 4×4 帧内预测算法, 使用 II 型参考像素集, 使利用块右方或下方相邻像素作为参考像素成为可能, 并且通过选择子块编码顺序, 将率失真最优化扩展到子块组范围。此外本文对预测模式也进行了优化, 使空间预测能取得更好的精度。

2 基于 II 型参考像素集的 Intra 4 × 4 空间预测模式

2.1 H.264 的 Intra 4×4 空间预测模式

H.264 的 Intra 4×4 模式帧内预测采用子块左边和上边相邻像素构成的 L 型参考像素集进行空间预测, 如图 1(a)所示。其中, a~p 是子块的 16 个组成像素, A~M 是已编码的参考像素。每个 4×4 子块共有 9 个预测模式, 包括一个 DC 模式和 8 个方向预测模式。8 个方向预测模式适合预测具有某个角度的纹理和边界图形。图 1(b)列举了其中 3 个预测模式。

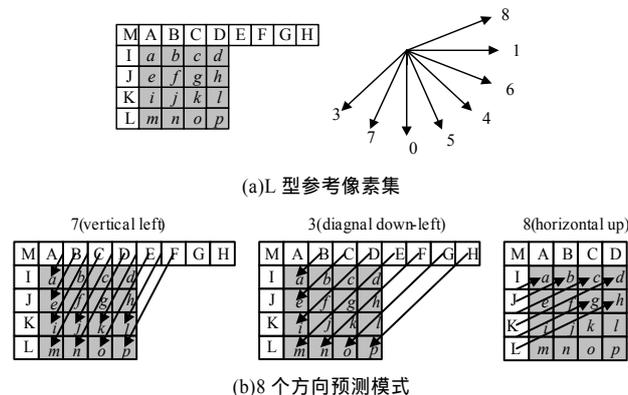


图 1 Intra 4×4 帧内预测模式

基金项目: 美国国家科学基金资助项目(NSS'USA5978-EAST Asia Pacific Program); 国家“863”计划基金资助项目(2002AA1Z1190)

作者简介: 韦 虎(1974 -), 男, 博士研究生, 主研方向: H.264 算法优化, SoC 实现; 林 涛、林争辉, 教授、博士生导师

收稿日期: 2007-09-30 **E-mail:** microtiger@sjtu.edu.cn

在图 1(b)中, 模式 8 右下方的 6 个像素($k\sim p$)全部由参考像素 L 预测, 预测精度显然较差。模式 3 和模式 7 使用参考像素 FGH 来预测右下方的几个像素, 而一般帧内像素间的相关性随距离变远而快速下降, 采用距离较远的 FGH 作为参考像素, 会使预测的精度较差。在这 3 种预测方向, 如果子块右方相邻像素已编码, 代替 FGH 作为预测参考像素, 可得到更好的预测效果。即使是水平方向预测时, 使用子块右方相邻像素作为参考像素, 也可能使预测效果更好。

2.2 基于 II 型参考像素集的空间预测模式

与 H.264 的 L 型参考像素集不同, 本文将块右方相邻像素也作为帧内预测的参考像素, 则形成的参考像素集如图 2 所示, 称为 II 型参考像素集。其中, $a\sim p$ 是子块的 16 个组成像素; $A\sim T$ 是已编码的参考像素。

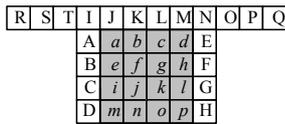
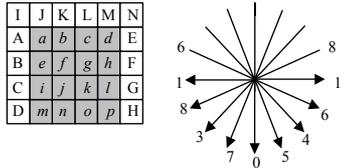
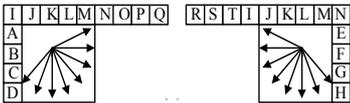


图 2 II 型参考像素集

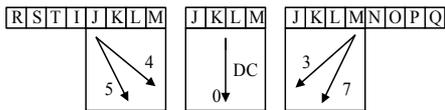
使用 II 型参考像素集预测时, 如果相邻块未编码, 可能参考像素 $A\sim T$ 不全部都可用, 如图 3 所示。当参考像素 $A\sim N$ 都可用时, 使用 U 型参考像素集 $A\sim N$ 预测子块, 如图 3(a) 所示; 当参考像素 ABCD 或 EFGH 不可用时, 则当前子块使用 L 型参考像素集来预测, 见图 3(b); 在图 3(c)中, 当左右 2 边相邻块都未编码时, 使用块上方的参考像素 $I\sim T$ 预测。



(a)U 型参考像素集



(b)L 型参考像素集



(c)参考像素 $I\sim T$

图 3 预测模式和参考像素

对于自然视频, 子块像素值沿着纹理方向常略有变化, 如渐增或渐减, H.264 的有方向预测往往使用预测方向上左端或上端的参考像素, 没有考虑对这种情况进行优化。而采用了 II 型参考像素集进行帧内预测时, 预测方向两端通常都存在已编码的参考像素。这时候用两端的参考像素值的加权平均作为预测参考值, 可以进一步减小残差的 DC 值, 提高预测编码性能。采用更多的参考像素也可以减少参考像素的量化误差对预测精度的影响。例如图 3(a), 在左下方向(模式 3)预测时, 本文采用如下的预测公式:

$$\begin{aligned} a &= (K + B + 1) \gg 1 \\ e, b &= (2L + 2C + B + D + K + M + 4) \gg 3 \\ i, f, c &= (2M + 3D + C + L + N + 4) \gg 3 \\ m, j, g, d &= (5N + M + E + D + 4) \gg 3 \\ n, k, h &= (2E + N + F + 2) \gg 2 \end{aligned}$$

$$o, l = (E + 2F + G + 2) \gg 2$$

$$p = (G + 2F + H + 2) \gg 2$$

3 基于灵活的子块顺序的帧内预测编码算法

在 H.264 中, 亮度宏块 4×4 块模式帧内编码时, 每个 4×4 像素块总按照从左往右和从上往下的顺序编码, 帧内预测仅可选择块左边和上边的参考像素进行, 因此, 预测结果常不理想。如果改变 16 个 4×4 子块的编码顺序, 允许先右后左和先下后上的顺序, 则子块右边和下边的像素也可被用作帧内预测的参考像素, 本文称之为灵活子块顺序(FSBO)预测编码。采用 FSBO 预测编码后, 宏块编码码流需要增加关于子块编码顺序的信息 $blkorder$, 因此, 宏块总的率失真代价函数如式(1)所示:

$$J_{macroblock} = \sum_{i=0}^{15} J_i(\lambda, blkorder) + \lambda \cdot R_{blkorder} \quad (1)$$

因为各子块可用的预测模式受编码顺序制约, 所以式(1)率失真代价 J_i 依赖于编码顺序。当改变子块编码顺序带来的率失真性能的提高大于传输子块编码顺序带来的比特率 $R_{blkorder}$ 增加的影响时, FSBO 预测编码就是有效的。对 16 个 4×4 子块排序, 所需的位数相当多, 影响了算法的适用性。本文采用将 16 个 4×4 子块按每行或列分为一组的方法, 减少 $R_{blkorder}$ 所需位数。对每个宏块, 仅对同一行或列内的 4 个子块采用 FSBO, 行(列)之间仍采用从上向下(从左向右)的顺序, 如图 4 所示。在 H.264 中, 对第 4 个和第 12 个 4×4 子块无法使用模式 3 或模式 7 进行预测, 而按本文行(列)分组进行从左往右、从上往下顺序编码时, 同时克服了这一缺点。

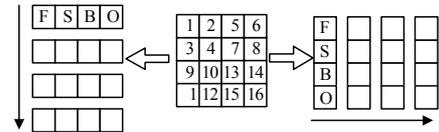
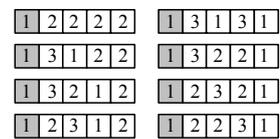


图 4 4×4 子块行分组和列分组

由于在子块组内各子块编码顺序可变, 可用的参考像素分布也相应改变, 以行分组为例, 可选的预测参考像素可能位于子块上边、左边或右边, 因此本文应用 II 型参考像素集进行预测。当对一个子块组按照某个顺序进行预测编码时(以行分组为例), 具有以下规律:

一行子块中只使用上边参考像素预测的子块, 它可先于左右的子块进行编码, 在本文中称之为起始子块(SB), 它有 6 种可选预测模式, 如图 3(c)所示; 当相邻两边只有一个子块已编码时, 当前子块存在 2 条可作参考像素的邻边(本文称之为双参考边子块 TRSB), 它有 9 种预测模式, 如图 3(b)所示; 几个双预测边子块和对应的起始子块组成一个独立块组, 2 个独立块组之间由一个如图 3(a)的三参考边子块(THRSB)连接。可以发现, 改变独立块组之间的编码顺序, 不会影响率失真性能。因此, 进行比较时, 只要考虑独立块组的各种分布模式, 而不必穷尽所有可能的子块排序。虽然每行 4 个子块有 24 种编码顺序, 但只有 8 种独立块组分布模式。在图 5 中, 当前宏块外部相邻的已编码子块也看成起始子块。



1:SB 2:TRSB 3:THRSB

图 5 8 种独立块组分布情况

本文对每种独立子块分布模式选取一个编码顺序，得到 8 种有效的编码顺序：1234, 2134, 3214, 1324, 2143, 4321, 1432, 1243。所以在未考虑熵编码的情况下，采用 FSBO 帧内预测，平均每个子块只需增加个 3/4 bit 用于块顺序传输。此外，宏块头需增加 1 bit 对行分组或列分组进行选择。

FSBO 预测编码算法具体流程如下：

(1)将宏块按行和列分别分组。

(2)对每组 4 个子块根据某种编码顺序进行编码，采用 RDO 方法确定各子块的预测模式，并按式(2)计算子块组的联合率失真代价，其中 λ 为拉各朗日乘子； $R_{blkorder}$ 为传递子块组内各子块编码顺序所需的位。

$$J_{blockgroup} = \sum_{i=0}^4 J_i(\lambda_i, blkorder) + R_{blkorder} \cdot \frac{\sum \lambda_i}{4} \quad (2)$$

循环 B 过程，完成 8 种编码顺序编码，选取联合率失真代价最小的编码顺序作为子块组的编码顺序。然后分别累加分组和列分组的联合率失真代价进行比较，将较小的联合率失真代价和作为宏块 Intra4×4 模式的率失真代价，并确定分组方式。

4 实验结果与分析

本文提出的FSBO预测编码在H.264 JM10^[4]参考软件上进行了实现。在实验中，对每个序列，进行连续1帧(仅采用Intra4×4模式)编码，固定量化因子，率失真最优化和CABAC选项打开。测试序列包括qcif序列(Foreman, Silent, Carphone, Suzie)和cif序列(Mobile, Tempete, Stefan)，每个序列编码前300帧。

图 6 给出了 cif 格式 Mobile 序列率失真随 Qp 变化的结果。实验结果显示，当不考虑子块编码顺序信息时，FSBO 的率失真性能始终优于 JM8.6，和参考软件相比，本文提出的方法可以取得平均 0.4 dB 的 PSNR 增益，且在低比特率情况下优势更明显，这主要是由于 FSBO 提供了更好的参考像素选择，减小了量化误差对预测的影响。

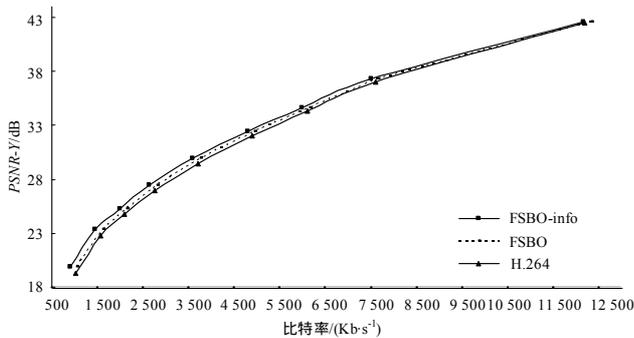


图 6 Mobile(CIF)RD 曲线

表 1 列出了 2 种分组模式和 8 种编码顺序之间的比例，可以看到，从上往下从左往右顺序编码(1234)只在 41.9%的情况下是最优的。当加入子块编码顺序信息后，编码性能有所下降，在高比特率情况下 PSNR 增益变为负，这时候 FSBO 方法不再适用，图 6 中临界比特率大约在 11 000 Kb/s 处。由于实验中对子块顺序信息没有采用自适应熵编码，因此在改

进后 FSBO 方法适用范围和 PSNR 增益可得到进一步提高。对其余视频序列进行仿真也取得了相似的结果，如表 2 所示。

表 1 Mobile 序列块分组模式和子块编码顺序的比例

子块顺序	行分组/(%)	列分组/(%)
1234	27.4	14.5
2134	2.5	2.6
3214	6.2	4.1
1324	4.1	1.9
2143	3.0	0.6
4321	12.3	4.9
1432	8.9	2.9
1243	2.9	1.2
sum	67.3	32.7

表 2 其他序列实验结果

	FSBO-info	FSBO	
CIF	$d=700$ Kb/s	$d=700$ Kb/s	$\Delta PSNR_Y > 0$ dB
	$\Delta PSNR_Y = (dB)$	$\Delta PSNR_Y = (dB)$	$d < (Kb/s)$
Tempete	+0.53	+0.26	8 100
Stefan	+0.32	+0.11	6 900
QCIF	$d=250$ Kb/s	$d=250$ Kb/s	$\Delta PSNR_Y > 0$ dB
	$\Delta PSNR_Y = (dB)$	$\Delta PSNR_Y = (dB)$	$d < (Kb/s)$
Foreman	+0.29	0.09	1 500
Silent	+0.41	0.11	1 700
Carphone	+0.32	0.02	1 000
Suzie	+0.35	0.13	1 500

5 结束语

考虑到 Intra4×4 编码时宏块的编码性能受子块编码顺序影响，且用预测块右方和下方的相邻像素作为参考像素可能取得更佳的预测效果，本文采用了灵活子块分组和编码顺序，并改进了预测模式和预测参考像素集，使得通过对各子块预测模式和有限的子块编码顺序的选择，实现宏块内的编码最优化。本文的 Intra4×4 模式帧内预测算法和 H.264 相比，在中低比特率条件下取得了良好的编码性能提升。这一思想同样可扩展应用到 FRExt Intra8×8 模式及 FMO 帧内编码中。

本文的方法需要增加对子块编码顺序信息的传输，制约了在高比特率下的应用范围，因此，未来的工作将研究对子块顺序信息进行自适应的熵编码以减少码率，进一步提高本文算法的性能。

参考文献

- [1] Zhang Peng, Zhao Debin, Ma Siwei, et al. Multiple Modes Intra-prediction in Intra Coding[C]//Proc. of IEEE International Conference on Multimedia and Expo. [S. l.]: IEEE Press, 2004.
- [2] Robert A, Amonou I, Pesquet-Popescu B. Improving Intra Mode Coding in H.264/AVC Through Block Oriented Transforms[C]// Proc. of IEEE MMSP'06. Victoria, Canada: [s. n.], 2006-10.
- [3] Palfner T, Wedi T. Advanced Scanning for Intra Frame Coding[C]// Proc. of 2006 IEEE International Conference on Image Processing. [S. l.]: IEEE Press, 2006: 3121-3124.
- [4] JVT-ISO/IEC-ITU-T. Joint Model 10[Z]. (2001-01-01). <http://iphome.hhi.de/suehring/ttml/index.htm>.