

H. 264 分辨率缩减快速帧内宏块模式选择算法研究^{*}

陈步真^a, 康志伟^b, 张敏^b

(湖南大学 a. 软件学院; b. 计算机与通信学院, 长沙 410082)

摘要: H. 264 视频编码标准对于帧内预测采用了 116×16 、 14×4 和 IPCM 三种模式, 而前两种模式分别有四种和九种预测方向, 这些特性提高了帧内宏块的编码效率, 但同时也增加了帧内模式选择的复杂度。提出了一种适用于 H. 264 视频流空间分辨率缩减转码的帧内宏块模式选择算法, 主要利用输入码流中的宏块模式信息, 推算出下采样后宏块可能的模式及预测方向, 从而减少需要计算的宏块模式数目, 同时对 14×4 宏块采用子块刷新机制, 确保转码质量。实验证明, 在保持编码效率和视频质量的同时, 能有效降低 H. 264 下采样转码过程中帧内宏块模式选择算法的复杂度。

关键词: H. 264; 视频转码; 模式选择; 统计信息

中图分类号: TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2010)06-2356-03

doi: 10.3969/j.issn.1001-3695.2010.06.103

Fast intra mode decision algorithms of downscaling video transcoding for H. 264

CHEN Bu-zhen^a, KANG Zhi-wei^b, ZHANG Min^b

(a. School of Software, b. School of Computer & Communication, Hunan University, Changsha 410082, China)

Abstract: The H. 264 video coding standard adopts some new methods for intra prediction. Though these characteristics enhance the efficiency of coding, it also increases the complexity of intra coding. This paper proposed a fast intra mode decision algorithm of spatial downscaling transcoding for H. 264. Fully used the statistic information of the source video stream to estimate the most possible modes and predict directions. The experimental results show that the proposed algorithms can increase the speed of intra coding significantly with negligible loss of PSNR and increase of bite-rate.

Key words: H. 264; video transcoding; mode decision; statistic information

0 引言

视频转码就是对压缩视频流进行端到端的处理, 即转码器的输入、输出都是压缩数据, 使转码后的压缩码流更能适应传输带宽以及接收端的要求。转码按照变换对象的不同可以分为码率转码、分辨率转码和语法转码^[1]。其中, 空间分辨率的转换是根据客户端显示设备的大小和处理能力来降低码流的分辨率, 并尽量保证视频质量的不显著降低。现阶段, 随着智能手机和掌上电脑等低分辨率显示与处理性能较弱的移动设备的逐渐普及, 分辨率转码在视频应用中得到了广泛的支持。

现阶段空间分辨率转码的结构主要是像素域级联结构。它将输入的压缩码流完全解码到像素域, 然后进行下采样以降低分辨率, 再重新送入编码器中, 从而获得所需要分辨率的视频流。H. 264 标准中引入了帧内预测编码等技术。H. 264 帧内预测编码充分利用图像的空间相关性, 用图像块周围的像素来进行预测, 再利用率失真优化技术进行最优帧内编码模式的选择, 这虽然降低了空域冗余, 提高了转码的压缩效率, 也大大增加了转码器的计算复杂度。空间分辨率下采样后帧内预测

编码模式选择的计算复杂度成为影响 H. 264 编码速率、实现实时性应用的瓶颈之一。为满足实时视频通信的要求, 必须使用快速算法来降低帧内预测编码模式选择的计算复杂度。

本文就此问题作了较深入的研究, 分析了 H. 264 帧内预测编码模式选择基本算法, 介绍了国内外对空域下采样后编码模式选择的快速算法的研究情况, 针对预测模式的选择问题作了较深入分析, 提出了根据原压缩视频流的统计特性来实现快速模式选择的方法, 同时采用了宏块内刷新机制, 确保转码质量, 并进行了仿真实验, 验证了算法的性能。

1 H.264 的帧内模式选择

1.1 帧内模式类型

H. 264 支持 14×4 、 116×16 以及 IPCM 三种帧内预测模式。如表 1 所示, 14×4 模式的 4×4 子块通过周围的像素值来预测本身的像素值。表中, $a \sim p$ 为 4×4 块内的 16 个像素; $A \sim M$ 为其邻接的 13 个像素。 14×4 宏块模式支持九种预测模式, 分别为垂直、水平、DC、水平偏上、水平偏下、垂直偏右、垂直偏左、左下角、右下角模式。

收稿日期: 2009-11-08; **修回日期:** 2009-12-09 **基金项目:** 国家自然科学基金资助项目(60702065); 湘潭市科技计划重点资助项目(ZJ20071008)

作者简介: 陈步真(1985-), 男, 湖南冷水江人, 硕士研究生, 主要研究方向为视频图像处理(solxipin@163.com); 康志伟(1962-), 男, 湖南湘潭人, 教授, 主要研究方向为数字信号处理; 张敏(1983-), 女, 山东日照人, 硕士研究生, 主要研究方向为数字图像处理。

I16 × 16 宏块也是通过其周围的 33 个邻接像素值来预测其本身的 256 个像素值。如图 1 所示,它支持四种模式,分别为垂直、水平、DC 和平面模式。

表 1 I4 × 4 预测的像素分布

M	A	B	C	D	E	F	G	H
I	a	b	C	d				
J	e	f	G	h				
K	i	j	K	L				
L	m	n	o	p				

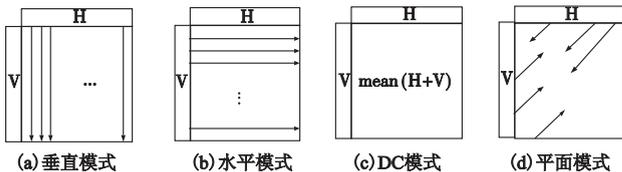


图 1 I16×16 的预测模式

1.2 H.264 的帧内模式选择算法

H.264 采用基于 Lagrangian 优化算法的率失真优化模型来进行模式选择。宏块 S_k 的 Lagrangian 代价函数如下所示:

$$J_{MODE}(S_k, I_k | Q, \lambda_{MODE}) = D_{REC}(S_k, I_k | Q) + \lambda_{MODE} \times R_{REC}(S_k, I_k | Q)$$

其中: S_k 为原始像素; I_k 为相应宏块的编码模式; Q 为量化参数; λ_{MODE} 为 Lagrangian 乘数因子, 取值为 $0.85 \times 2^{Q/3}$; D_{REC} 为失真度, 一般为原始像素与重构像素的方差值之和; R_{REC} 为给定模式和量化参数编码后的比特率。对于帧内宏块, 显然率失真的值是通过帧内预测、整数 DCT 变换、哈达码变换、量化、逆量化、逆哈达码变换、逆整数 DCT 变换、熵编码之后才能够得到, 其计算非常耗费时间。

选择最优编码模式算法是通过遍历所有可能的编码模式, 寻找具有 $\min(J_{MODE}(\cdot))$ 的模式作为最优编码模式。对每一个帧内宏块进行模式选择, 即使不包括色度计算, 也需要进行 $(9 \times 16 + 4) = 148$ 种组合模式的 RDO 计算。用此方法计算所得的帧内预测模式进行编码, 能得到最优的编码压缩效果, 但是过多的候选模式数和每种模式率失真代价计算开销巨大, 使模式选择成为影响空域下采样编码速率的瓶颈之一。

1.3 快速算法的主要研究情况

快速算法研究的主要途径是通过分析输入流的宏块模式统计, 预测最有可能的宏块模式以减少计算复杂度。文献[2]中的帧内预测将预测方向分为 DC 类、水平类和垂直类, 通过根据各个类的特征模式比较, 选择其中一个类的模式继续比较; 文献[3]假设编码代价随宏块的大小而单调变化, 先尝试大块宏块模式, 如果较小宏块模式对应的编码代价大于较大模式的编码代价, 则选择较大宏块对应的编码模式, 提前终止模式选择; 文献[4]利用输入码流中帧内宏块的 DCT 系数选择 H.264 帧内编码模式, 如果四个块的 DCT 系数差异小, 则采用 INTRA16 × 16 模式, 反之采用 INTRA4 × 4 模式; 文献[2]则将各种模式出现概率从大到小排列, 逐个进行选择判断, 当某个模式的编码代价已经足够小时, 终止模式选择, 不再考虑其他模式; 文献[5]根据宏块的边缘特征, 选择出与该特征相应的模式自己进行 RDO 计算, 能大幅度降低帧内预测的模式选择数目;

文献[6]通过周围宏块的模式, 提出了模式选择提前终止算法。

2 基于统计的快速模式选择算法

H.264 的两种帧内模式中, I16 × 16 模式适合于纹理比较平滑的区域, 而 I4 × 4 模式适合于细节比较丰富的区域, 而模式的预测方向则反映了模块的局部边缘方向。显然下采样后的图像与原图像内容基本是不变的, 则下采样后图像与原图像有很大的相关性。

如图 2 所示, 实验是从对 container_cif.yuv 和 container_qcif.yuv 第一帧进行帧内编码的 I4 × 4 与 I16 × 16 模式分布图。其中, 黑块表示为 I4 × 4 模式, 白块表示为 I16 × 16 模式。显然, 不同分辨率下的模式分布非常相似, 且比例大致相同。

表 2 显示了原始图像与下采样后图像中 I4 × 4 模式帧内预测类型的统计数据。从表中可以看出, 原始图像与下采样后图像的预测类型大致相似。

表 2 帧内预测方向分布

预测方向	0	1	2	3	4
container_cif	504	1412	678	141	190
container_qcif	117	386	183	29	43
预测方向	5	6	7	8	总计
container_cif	112	297	80	298	3712
container_qcif	30	72	20	80	960

由此可见, 原始图像中的帧内宏块模式与预测方向的分布可以粗略反映出下采样后的图像帧内宏块模式和预测模式的分布。因此, 本文提出了一种基于统计分布的快速帧内宏块选择算法。具体算法如下:

a) 对原始码流进行熵解码后, 统计如下数据: 一帧中 I16 × 16 宏块的数目与宏块总数目的比例 $\text{prop}_{16 \times 16}$; 一帧中 I16 × 16 宏块各预测模式的数目 $C_{16 \times 16}^{V_i}$, 并排序得到 $V_{16 \times 16} = \{V_n | V_n \in V_{16 \times 16}, C_{16 \times 16}^{V_n} > C_{16 \times 16}^{V_{n+1}}, 0 \leq n < 4\}$; 在下采样后的宏块 B_i 中, 如图 3 所示, 得到 B_i 对应原始帧中 I16 × 16 宏块的比例 $\text{prop}_{16 \times 16}^{B_i}$, 并得到 I16 × 16 宏块中出现的各预测模式的数目 $C_{16 \times 16}^{B_i, V_i}$ 。根据结果及 $V_{16 \times 16}$ 对预测模式排序, 如果在 B_i 对应原始帧的宏块中出现的预测模式次数多则排前, 若出现预测模式的次数相同, 则将同时在 $V_{16 \times 16}$ 中排前的预测模式排前, 即

$$V_{16 \times 16}^{B_i} = \{V_{16 \times 16}^{B_i}(m) | V_{16 \times 16}^{B_i}(m) \in V_{16 \times 16}, C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m)} > C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m+1)} \} \cup \{C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m)} = C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m+1)} \&\& C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m)} > C_{16 \times 16}^{B_i, V_{16 \times 16}^{B_i}(m+1)}, 0 \leq m < 4\}$$

同理得到一帧中 I4 × 4 宏块各预测模式排序后的集合 $V_{4 \times 4}$ 及宏块 B_i 对应原始帧中的 I4 × 4 宏块各预测模式排序后的集合 $V_{4 \times 4}^{B_i}$ 。

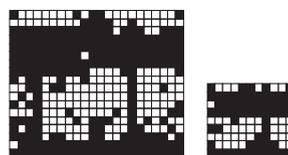


图 2 宏块模式分布图

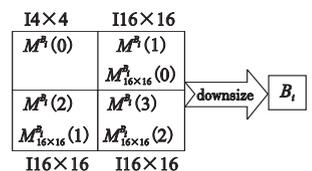


图 3 下采样宏块与原码流宏块对应情形

b) 选择宏块模式。当 $\text{prop}_{16 \times 16} \leq 1/3$ 且 $\text{prop}_{16 \times 16}^{B_i} \geq 3/4$ 时, 选择 I16 × 16 宏块模式; 当 $\text{prop}_{16 \times 16} \geq 2/3$ 且 $\text{prop}_{16 \times 16}^{B_i} \geq 1/2$

时,选择 I16 × 16 宏块模式;当 $1/3 < \text{prop}_{16 \times 16} < 2/3$ 且 $1/4 < \text{prop}_{16 \times 16}^{B_i} < 3/4$ 时,两种宏块模式都将进行下一步计算,以确定最终的宏块模式。其他情况下均采用 I4 × 4 宏块模式。

c) 选择预测模式。

(a) 宏块模式为 I16 × 16。如果 $\text{prop}_{16 \times 16} \leq 1/3$,则在 $V_{16 \times 16}^{B_i}$ 中依次选取两个可用的预测模式进行 RDO 计算;如果 $1/3 < \text{prop}_{16 \times 16} < 2/3$,则在 $V_{16 \times 16}^{B_i}$ 中依次选取三个可用的预测模式进行 RDO 计算;如果 $2/3 \leq \text{prop}_{16 \times 16}$,则对四种预测模式都进行 RDO 计算,从而获得最佳预测模式。

(b) 宏块模式为 I4 × 4。将当前 I4 × 4 子块的上面和左边的预测模式作为候选预测模式。从候选的预测模式及 $V_{4 \times 4}^{B_i}$ 中依次选取三个不同且可用的预测模式进行 RDO 计算,从而获得最佳预测模式。

(c) 如果在选择宏块模式的计算中选择了两种宏块模式,则分别进行(a)(b)的计算,选取可得到最佳结果的宏块模式作为当前宏块的模式,同时确定了预测模式。

d) I4 × 4 宏块内刷新机制。由于每个 4 × 4 子块都用其上面和左边的 4 × 4 子块的预测模式参考,对每个采用 I4 × 4 模式的宏块,对其第 1 个 4 × 4 子块及第 13 个 4 × 4 子块都进行九种预测模式的 RDO 计算,以获得最佳预测模式,防止预测模式误差的扩散。

上面所提出的算法,实际上是在基本算法流程的基础上,利用了转码前后码流间相应帧及相应宏块的高相关性,根据统计出的宏块模式信息及预测模式信息,先决断出宏块模式,并提取出最有可能的几种候选预测模式,然后计算出最优的预测模式。因为对于转码前后的视频来说,它们的内容、纹理、细节等特征都是极其相似的,这样在保证转码质量的同时也就大大减少了算法的计算量。

3 实验结果及分析

本文提出的算法的实验平台是 P4 3.2 GHz CPU,1 GB 内存,Windows XP 操作系统。仿真以 JVT 提供的参考软件 JM 8.6^[7] 作为平台,进行部分修改,然后将实验结果与未作修改的参考软件进行对比。仿真选取了 bus、container、foreman、news 四个 CIF 视频序列作为测试对象。对每个视频序列选取 50 帧,量化参数 Q_p 分别选取 12、18、28、36、42 五个值,编码帧率为 20 fps。

表 3 Bus 和 container 序列

Q_p	算法	bus 序列			container 序列		
		RDO/次	bitrate/kbps	PSNR/dB	RDO/次	bitrate/kbps	PSNR/dB
12	JM8.6	14172	3396	50.12	14172	2460	50.19
	本文	5826.4	3422	49.83	5782.1	2485	49.96
18	JM8.6	14172	2465	44.36	14172	1616	44.67
	本文	5822.5	2490	44.19	5533	1645	44.56
28	JM8.6	14172	1230	35.27	14172	730.0	37.42
	本文	5696.5	1257	35.11	3910.3	751.9	37.33
36	JM8.6	14172	508.9	29.13	14172	334.4	31.70
	本文	4793.6	529.5	29.07	2926.9	358.9	31.65
42	JM8.6	14172	229.5	25.70	14172	168.7	27.42
	本文	3495.3	241.7	25.67	2472.4	179.8	27.47

表 4 Foreman 和 news 序列

Q_p	算法	foreman 序列			news 序列		
		RDO/次	bitrate/kbps	PSNR/dB	RDO/次	bitrate/kbps	PSNR/dB
12	JM8.6	14172	2210	50.11	14172	2079	50.24
	本文	5777.5	2278	49.84	5740	2105	50.07
18	JM8.6	14172	1535	44.63	14172	1433	45.59
	本文	5633.7	1561	44.38	5403	1455	45.40
28	JM8.6	14172	653.2	37.13	14172	735.7	37.77
	本文	5262.3	675.6	37.03	4643	754.6	37.64
36	JM8.6	14172	291.5	31.75	14172	364.8	31.48
	本文	4182	310.7	31.67	3889	379.3	31.40
42	JM8.6	14172	168.2	28.30	14172	191.1	27.40
	本文	2915.6	180.1	28.24	2705	202.8	27.34

由表 3、4 的实际仿真数据可以得出结论:

a) 本文所提出的改进算法能在一定程度上减少帧内预测计算量,而 PSNR 仅仅减小了 0.03 ~ 0.24 dB,甚至还会增加,码率的增加也在 6% 之内,而且 Q_p 越小,码率增加得也越小。

b) 改进算法对帧内预测计算量减小相当大,计算量的减少能达到 60% ~ 83%。计算量的减少与量化参数的取值密切相关,量化参数越大,对计算量的减少就越大,但此时码率会增加。

4 结束语

本文主要提出了一种适用于 H. 264 视频下采样转码的帧内宏块模式选择算法。对于帧内宏块模式,充分利用了原码流中的各种宏块模式信息,选出一个最有可能的模式子集进行搜索,减少了需要搜索的模式数量,并且采用宏块内刷新机制,有效防止了误差的扩散。实验证明该算法能有效地排除可能性较小的模式,降低了模式选择的复杂度,同时对其质量和压缩率仅有轻微的下降。

参考文献:

[1] 尚书林,杜清秀,卢汉青,等. 视频转码技术研究现状与最新进展[J]. 自动化学报,2007,33(12):1233-1241.

[2] LU Xiao-an, TOURAPIS A M, YIN Peng, et al. Fast mode decision and motion estimate for H. 264 with a focus on MPEG-2/H. 264 transcoding[C]//Proc of the 2005 IEEE International Symposium on Circuits and Systems. Washington DC: IEEE Computer Society, 2005:1246-1249.

[3] ZHOU Z, SUN S J, LEI S, et al. Motion information and coding mode reuse for MPEG-2 to H. 264 transcoding[C]//Proc of IEEE International Symposium on Circuits and Systems. Seattle:IEEE, 2005: 1230-1233.

[4] WANG L, WANG Q, LIU Y, et al. A fast intra mode decision algorithm for MPEG-2 to H. 264 video transcoding[C]//Proc of the 10th IEEE International Symposium on Consumer Electronics. 2006:1-5.

[5] 常伟,蔡灿辉. 基于边缘特征的 H. 264/AVC 帧内模式选择算法[J]. 计算机工程与应用,2007,43(29):82-85.

[6] LEE J, BYUNGWOO J. Fast mode decision for H. 264[C]//Proc of the 2004 International Conference on Multimedia and Expo. Washington DC: IEEE Computer Society, 2004:1131-1134.

[7] H. 264/AVC reference software version 8.6 [CP/OL]. (2007-08-07). http://iphome.hhi.de/suehring/tml/download/old_jm/jm86.zip.