

# 一种用于 AVS-M 视频编码的快速帧内 预测模式判断算法

刘东华<sup>1),2)</sup> 杨立志<sup>1)</sup> 陈益强<sup>1)</sup>

<sup>1)</sup>(中国科学院计算技术研究所, 北京 100080) <sup>2)</sup>(国防科技大学, 长沙 410073)

**摘要** AVS-M 作为中国具有自主知识产权的移动视频压缩标准, 是采用基于块的混合编码方法, 其帧内预测针对  $4 \times 4$  块进行。由于 AVS-M 对亮度块考虑 9 种预测模式, 从而导致最优预测模式判断的计算相当复杂。为了提高编码速度, 提出通过计算块内下采样像素点代价, 然后确定候选预测模式集的方法来实现帧内预测模式的快速判决。仿真结果表明, 该快速算法在保证编码性能几乎不变的前提下可使得全帧内编码的总时间减少约 20% ~ 30%。

**关键词** AVS-M 帧内预测 模式判断 下采样

**中图分类号**: TN919.81 **文献标识码**: A **文章编号**: 1006-8961(2006)11-1566-04

## A Rapid Intra Prediction Mode Decision Algorithm for AVS-M Video Coding

LIU Dong-hua<sup>1),2)</sup>, YANG Li-zhi<sup>1)</sup>, CHEN Yi-qiang<sup>1)</sup>

<sup>1)</sup>(Institute of Computing Technology, CAS, Beijing 100080) <sup>2)</sup>(National University of Defense Technology, Changsha 410073)

**Abstract** As a mobile video standard owned by China, AVS-M adopts hybrid coding based on block whose intra prediction is just for  $4 \times 4$  block. For luma intra prediction, there are 9 modes and the choice of optimized mode is very complex. In this paper, a rapid intra prediction mode decision algorithm is presented. Its basic idea is to down sample in  $4 \times 4$  luma block for cost computation and determine a candidate mode set in advance. Simulation results show the rapid algorithm can save about 20% ~ 30% encoding time for all intra prediction coding with a little performance degradation.

**Keywords** AVS-M, intra prediction, mode decision, down-sampling

## 1 引言

AVS-M<sup>[1]</sup>作为《AVS 信息技术——先进音视频编码》的重要组成部分, 是我国拥有自主知识产权、面向移动应用而制定的视频标准。该标准采用与目前国际最新视频标准 H. 264/AVC<sup>[2]</sup>类似的技术框架, 但是在算法上则采用了许多我国拥有自主知识产权的核心技术。由于 AVS-M 采用基于块的混合编码方法, 帧内预测针对  $4 \times 4$  块进行, 对亮度块和

色度块分别考察 9 种和 3 种预测模式, 多模式选择和判断使得运算复杂性大大增加。

近几年, 针对 H. 264/AVC 的帧内预测模式判断的快速算法虽出现很多<sup>[3-5]</sup>, 但这些算法都是针对 H. 264/AVC 视频标准设计的, 仅考虑了  $4 \times 4$  和  $16 \times 16$  两种块类型。AVS-M 标准提出较晚, 如今还没有针对帧内预测的快速算法。本文通过对 AVS-M 标准中亮度块帧内预测过程的分析, 提出一种新的帧内预测快速算法, 该算法经过两个步骤实现, 不仅不需要增加任何额外计算和操作, 而且在不影响

基金项目: 上海市博士后科研资助计划项目(05R214207)

收稿日期: 2006-06-25; 改回日期: 2006-09-08

第一作者简介: 刘东华(1976 ~ ), 男, 讲师, 2002 年获国防科技大学博士学位, 现为中国科学院计算技术研究所博士后。主要研究方向为移动多媒体信息处理和信道编码。E-mail: dhliu@shiet.cn

编码性能的前提下能有效提高编码速度。

## 2 AVS-M 标准的帧内预测

AVS-M 支持  $4 \times 4$  块帧内预测,其主要方法是利用当前  $4 \times 4$  块上面和左侧相邻行/列像素按不同方式填充进行预测,AVS-M 支持 9 种亮度预测模式和 3 种色度预测模式,由于色度预测模式类别较少,因此这里仅考虑亮度帧内预测模式(如图 1 所示,图中未包括 DC 模式)。

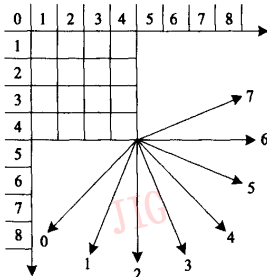


图 1  $4 \times 4$  亮度块帧内预测模式

Fig.1 Intra prediction modes for  $4 \times 4$  luma block

在 AVS-M 标准中,有以下两种最优帧内预测模式判决准则:一是基于预测模式的,一是基于率失真优化(rate distortion optimization, RDO)代价的。对于基于预测模式的方式,首先计算原始块与每一种模式下预测块之差,然后根据得到的残差块元素绝对值和(sum of absolute difference, SAD)按照式(1)计算当前模式下的代价,即

$$C_p = SAD_{16} + 3 \cdot P \cdot \lambda(QP) \quad (1)$$

其中

$$SAD_{16} = \sum_{i=0}^3 \sum_{j=0}^3 |f_o(x+i, y+j) - f_p(x+i, y+j)| \quad (2)$$

$f_o(x, y)$  和  $f_p(x, y)$  分别表示原始像素值和预测像素值( $f_o, f_p$  下角 O 代表 original, 下角 P 代表 predictive, 下同),  $x$  和  $y$  表示当前  $4 \times 4$  块左上角元素坐标。

编码器在对每个  $4 \times 4$  块进行帧内预测模式选择之前,先根据其已编码的左边块和上边块的帧内预测模式确定当前块的最可能的预测模式<sup>[1]</sup>,在式(1)中,如果当前的预测模式与最可能的预测模式相同,则  $p=0$ , 否则  $p=1$ 。

式(1)的最后一项为量化参数(quantization

parameter, QP)的指数函数:

$$\lambda(QP) = \text{round}\left(2^{\frac{QP}{8}}\right) \quad (3)$$

预测过程中,首先对 9 种模式分别计算相应的代价函数  $C_p$  的值,然后选择  $C_p$  值最小的一个作为最优预测模式。

对于基于 RDO 的预测方式(简称 RDO 算法),则对每一种预测模式可按照式(4)计算代价:

$$C_{RDO} = SSD + \lambda \cdot R \quad (4)$$

其中  $R$  表示残差块经过熵编码后得到的码长,  $\lambda$  为与量化参数相关的率失真参数。 $SSD$  表示原始块与重构块之间的误差均方和,即

$$SSD = \sum_{i=0}^3 \sum_{j=0}^3 (f_o(x+i, y+j) - f_{RDO}(x+i, y+j))^2 \quad (5)$$

其中重构块元素值  $f_{RDO}(x, y)$  需要对残差块经过变换→量化→反量化→反变换过程后再与预测块相加得到。

预测时,首先对 9 种模式分别计算相应的 RDO 代价  $C_{RDO}$  的值,然后选择值最小的作为最优预测模式。

通常,选择 RDO 算法可以获得比基于预测模式的方法更好的编码性能,但由于需要进行变换、量化、反量化、反变换以及熵编码过程,因此计算相对复杂。

## 3 快速帧内预测模式判断算法

快速算法相对于 RDO 算法,可分以下两步实现:

(1)以基于预测模式的方式计算代价函数  $C_p$ ,但是这里采用简化的计算方法,即对  $4 \times 4$  块内像素进行下采样,采样方式如图 2 所示。

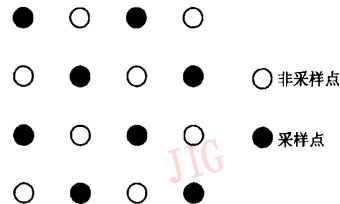


图 2  $4 \times 4$  块内像素下采样

Fig.2 Down-sampling for  $4 \times 4$  luma block

然后仅对采样点计算 SAD,记做  $SAD_8$ ,即

$$SAD_8 = |f_o(x, y) - f_p(x, y)| + |f_o(x + 2, y) - f_p(x + 2, y)| + |f_o(x + 1, y + 1) - f_p(x + 1, y + 1)| + |f_o(x + 3, y + 1) - f_p(x + 3, y + 1)| + |f_o(x, y + 2) - f_p(x, y + 2)| + |f_o(x + 2, y + 2) - f_p(x + 2, y + 2)| + |f_o(x + 1, y + 3) - f_p(x + 1, y + 3)| + |f_o(x + 3, y + 3) - f_p(x + 3, y + 3)| \quad (6)$$

仅对采样点计算的代价函数如式(7)所示,即

$$\hat{C}_p = SAD_8 + 1.5 \cdot P \cdot \lambda(QP) \quad (7)$$

先对 9 种模式分别计算代价函数  $\hat{C}_p$ , 然后选择代价最小的 3 种模式构成候选模式集。

(2)对步骤(1)得到的候选模式集中每个模式,按照式(4),通过计算基于率失真的代价来实现基于 RDO 的模式选择,也即选择  $C_{RDO}$  值最小的模式作为最终预测模式。

## 4 实验结果与分析

### 4.1 计算复杂性

在该快速算法中,以  $\hat{C}_p$  代替  $C_p$ , 预测块的生成和 SAD 计算分别减少了 50% 的计算量,同时在  $C_{RDO}$  计算部分节省了 2/3 的计算量。为达到较好的编码性能,通常选择 RDO 算法,虽然在 RDO 算法中不需要计算  $C_p$  或  $\hat{C}_p$  的值,但本文快速算法与 RDO 算法相比,计算量更少。对于每个  $4 \times 4$  块的模式选择,两种算法中主要运算单元的计算量对比如表 1 所示。

表 1 RDO 算法与快速算法的计算量比较  
Tab.1 Computation comparison between RDO algorithm and improved algorithm

操作	快速算法		RDO 算法
	$\hat{C}_p$	$C_{RDO}$	$C_{RDO}$
赋值	$8 \times 9$	$8 \times 3$	$16 \times 9$
加法	$9 \times 9$	$(272 + 6n + 5m) \times 3$	$(272 + 6n + 5m) \times 9 + 16$
乘法	$2 \times 9$	$322 \times 3$	$322 \times 9$
移位	$\lfloor QP/8 \rfloor \times 9$	$(80 + 2n + 2m) \times 3$	$(80 + 2n + 2m) \times 9$
查找	—	$(64 + 2n) \times 3$	$(64 + 2n) \times 9$
比较	8	$(32 + 9n + 4m) \times 3$	$(32 + 9n + 4m) \times 9$

表中快速算法的计算包括  $\hat{C}_p$  和  $C_{RDO}$  两部分,但其  $C_{RDO}$  的计算仅针对候选模式集进行,而 RDO 算法的  $C_{RDO}$  计算则要对 9 种模式进行。表中  $n$  表示 9

种预测模式下得到的残差块中非零元素的平均个数,  $m$  表示相应熵编码的平均长度。从表 1 可以看出,对于各种操作,快速算法的计算量都小于 RDO 算法。

下面考察采用该快速算法与采用 RDO 算法进行编码的时间对比。仿真实验时,采用细节变化和运动程度为低、中、高的视频序列 *silent*、*foreman* 和 *football*。AVS-M 编码参考软件版本为 WM3.3a, 视频格式为 QCIF, 所有帧均采用帧内预测编码,总编码帧数为 250 帧,其他编码参数选择默认设置<sup>[1]</sup>。

图 3~图 5 给出了不同 QP 条件下 RDO 算法和快速算法编码时间的比较曲线。仿真实验采用 CPU 为 Pentium IV 1.7GHz、内存为 256MB 的计算机进行。从仿真结果可以看出,对所仿真的视频序列,采用快速算法后,在性能基本不变的条件下,帧内预测编码时间相对于 RDO 算法减少了 20%~30%。

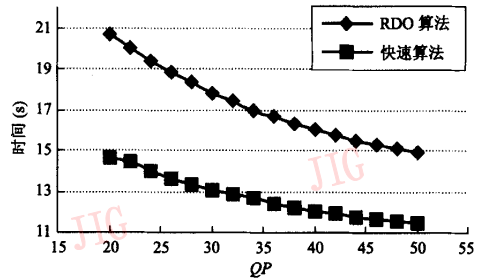


图 3 *silent* 序列的编码时间比较曲线  
Fig.3 Encoding time curve of *silent* sequence

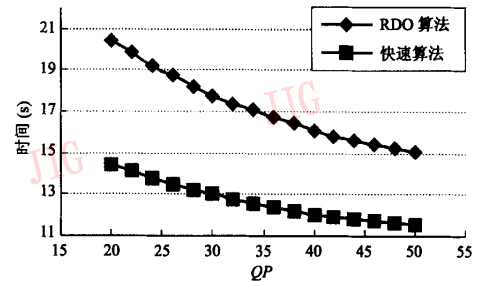


图 4 *foreman* 序列的编码时间比较曲线  
Fig.4 Encoding time curve of *foreman* sequence

### 4.2 编码性能

图 6~图 8 给出了编码时,采用 RDO 算法和本文的快速算法的性能比较曲线。由该组图可见,两种算法的编码性能基本相同(两条曲线基本吻合),这也说明了该快速算法的有效性。

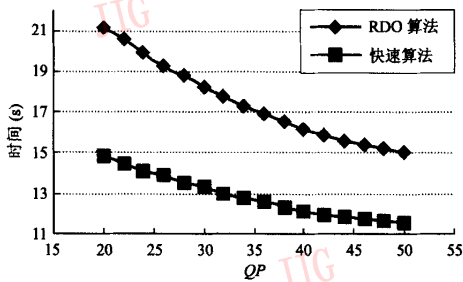


图 5 football 序列的编码时间比较曲线  
Fig. 5 Encoding time curve of football sequence

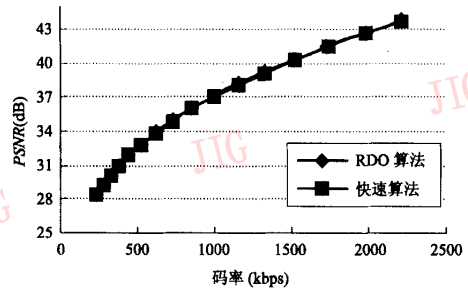


图 8 football 序列的率失真性能曲线  
Fig. 8 R-D curve of football sequence

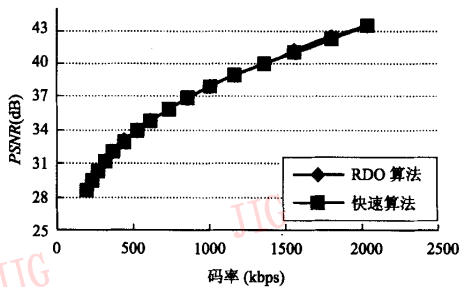


图 6 silent 序列的率失真性能曲线  
Fig. 6 R-D curve of silent sequence

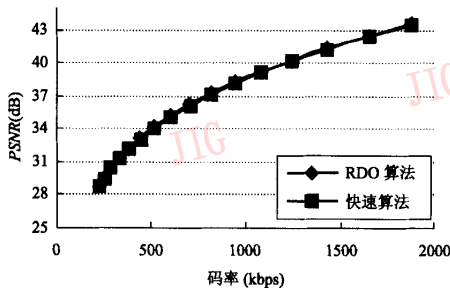


图 7 foreman 序列的率失真性能曲线  
Fig. 7 R-D curve of foreman sequence

#### 4 结 论

针对 AVS-M 中帧内预测模式判断计算量大的特点,本文提出了快速判断算法,该算法能在保证编码性能的前提下提高编码速度。由于在 AVS-M 编

码过程中,帧内预测模式选择不仅出现在帧内编码帧(I 帧),而且也出现在帧间编码帧(P 帧),用于 P 帧编码时需要将帧内预测与帧间预测进行比较,所以该快速算法也适合 P 帧编码时编码模式的快速选择。此外,还可以通过调整块内的下采样像素点数及候选模式集的元素个数来满足不同编码性能和计算复杂性要求。

#### 参 考 文 献 (References)

- 1 AVS Group. Information technology—Advanced coding of audio and video-Part 7: Mobile video[S]. April 2006. [AVS 工作组. 信息技术——先进音视频编码 第七部分: 移动视频(报批稿) [S], 2006 年 4 月].
- 2 Joint Video Team of ITU-T and ISO/IEC JTC1. Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H. 264 [S] | ISO/IEC 14496-10 AVC). Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050r1. March 2003.
- 3 Zhang Y D, Dai F, Lin S X. Fast  $4 \times 4$  intra-prediction mode selection for H. 264 [A]. In: Proceedings of IEEE International Conference on Multimedia and Expo (ICME2004) [C], Taipei, Taiwan, China, 2004; 1151 ~ 1154.
- 4 Meng B, Au O C, Wong C W, et al. Efficient intra-prediction mode selection for  $4 \times 4$  blocks in H. 264 [A]. In: Proceedings of the 2003 International Conference on Multimedia & Expo (ICME2003) [C], Baltimore, Maryland, USA, 2003; 521 ~ 524.
- 5 Fu F W, Lin X G, Xu L D. Fast intra prediction algorithm in H. 264/AVC [A]. In: Proceedings of International Conference on Signal Processing (ICSP2004) [C], Beijing, China. August 2004; 1191 ~ 1194.