

快速 AVS2 帧内预测选择算法

赵超*, 赵海武, 王国中, 李国平, 滕国伟

(上海大学 通信与信息工程学院, 上海 201900)

(*通信作者电子邮箱 zhaochaosens@163.com)

摘要:针对目前数字音视频编解码技术标准(AVS2)中帧内预测模式判断过程计算较为复杂,而如今超高清视频的普及给编解码系统带来很大压力的问题,提出了一种快速帧内预测选择算法。该算法先对最底层最小编码单元(SCU)进行预测模式删选,减少了底层 SCU 的计算量;再通过下层编码单元(CU)的预测模式得到上层 CU 的预测模式,从而减少了上层 CU 的计算量。实验表明,该算法对压缩效率的影响很小,并且编码时间平均下降超过 15%,并可有效地降低帧内编码的复杂度。

关键词:数字音视频编解码技术标准; 四叉树; 帧内预测模式; 编码单元

中图分类号: TN911.73 **文献标志码:** A

Fast selection algorithm for intra prediction in AVS2

ZHAO Chao*, ZHAO Haiwu, WANG Guozhong, LI Guoping, TENG Guowei

(School of Communication and Information Engineering, Shanghai University, Shanghai 201900, China)

Abstract: For Audio Video coding Standard II (AVS2) intra-prediction mode determination process is complicated to calculate, and the popularity of ultra-high definition video put encoding and decoding system under great pressure, a kind of fast intra prediction algorithm was presented in this paper. The algorithm selected the part of the Smallest Coding Unit (SCU) prediction mode, reducing the amount of computation of the underlying SCU, and then the upper layer Coding Unit (CU) obtained the prediction mode by the lower CU prediction mode, thereby reducing the amount of computation of the upper CU. The experimental results show that the impact on the compression efficiency of the algorithm is very small, the encoding time on average decreases more than 15 percent, and can effectively reduce the complexity of intra-coding.

Key words: Audio Video coding Standard II (AVS2); quadtree; intra prediction mode; Coding Unit (CU)

0 引言

AVS 工作组于 2012 年开始着手制定新一代视频编码标准(Audio Video coding Standard II, AVS2),在 2014 年 6 月已基本完成新标准的制定。经测试,AVS2 的编码压缩效率比 AVS1 提高一倍以上,与 HEVC(High Efficiency Video Coding)相当^[1]。

AVS2 采用的主要技术有:大块四叉树结构(图 1);更多的预测块划分模式;更多的帧内预测模式;更多的帧间预测模式,包括双前向预测、背景图像(G 和 GB 图像)预测技术;B 图像可作为参考图像;参考图像灵活配置技术;样本偏移补偿技术等^[2-3]。新技术在提高压缩效率的同时,也显著增加了编码复杂度。例如:在编码时采用大块四叉树结构编码,理论上需要尝试所有四叉树划分方式,选择其中最优的一个,这样增加了相当大的计算量。为了提高编码效率,AVS2 的帧内编码采用了 30 种角度预测模式和 3 种非角度模式,为了获得最优模式^[4],必须计算 33 种帧内预测模式的率失真代价,计算量很大。在一些运动比较剧烈的视频中,I 帧编码的比例比

较大,其编码过程也就有更高的要求,因此,研究如何降低 AVS2 帧内编码复杂度,是非常有必要的。

针对视频编码帧内预测的高计算复杂度,已有众多学者提出了快速算法;文献[5]提出一种基于边缘方向强度的快速帧内预测模式决策算法;文献[6]提出一种基于残差绝对值总和(Sum of Absolute Transformed Difference, SATD)准则和空间相关性的快速帧内预测算法;文献[7]提出了一种基于纹理方向和空域相关性的帧内快速编码算法;文献[8]提出基于在编码树块的新检测顺序的快速帧内模式决策算法。以上算法基本上是提前决定了 CU 的大小划分以及减少帧内粗选模式列表的数量等方法。本文使用一种新的方法,参考了文献[5]中利用边缘方向强度来减少帧内预测需计算的帧内预测模式,以及文献[6]中利用空间相关性的特点来实现快速帧内预测模式决策过程,对帧内各 CU 层次预测模式进行优化,从而降低帧内编码时间,提高编码效率。

1 AVS2 帧内预测相关技术介绍

AVS2 采用图 1 所示的大块四叉树编码结构,一幅图像被

收稿日期:2015-05-16;修回日期:2015-07-16。

基金项目:国家自然科学基金资助项目(61271212);上海市自然基金资助项目(14ZR1415200)。

作者简介:赵超(1992-),男,湖南益阳人,硕士研究生,主要研究方向:视频编解码、图像处理; 赵海武(1973-),男,上海人,副教授,博士,主要研究方向:数字音视频编解码技术、编解码技术标准; 王国中(1962-),男,上海人,教授,博士,主要研究方向:视频编码与多媒体通信、图像处理; 李国平(1974-),男,上海人,副教授,博士,主要研究方向:数字音视频编解码技术、复用技术; 滕国伟(1975-),男,上海人,副教授,博士,主要研究方向:数字音视频编解码技术、智能视频分析。

划分为若干最大编码单元 (Largest Coding Unit, LCU), LCU 的尺寸最大可以达到 64×64 (亮度样本及其对应的色差样本)。LCU 可以进一步划分为各种尺寸的编码单元 (Coding Unit, CU), CU 的尺寸最小可以是 8×8 。LCU 被进一步划分的次数称为划分深度 (depth), 不同深度对应不同的 CU 尺寸^[3]。为了叙述方便, 给划分得到的 CU 进行如图 1 所示的编号。

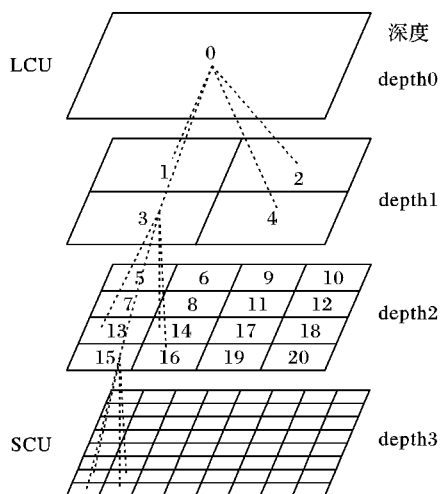


图 1 CU 四叉树层次划分

AVS2 中 CU 被划分为若干预测单元 (Prediction Unit, PU), 在帧内模式编码中, CU 划分为 PU 的方式有如图 2 所示的 4 种划分模式^[1], 图中字母表示 PU 的编号。

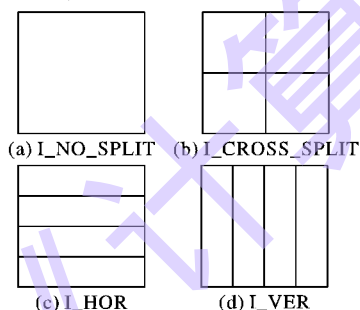


图 2 帧内编码单元划分为预测单元

对每一个亮度 PU, AVS2 定义了 33 种帧内预测模式, 如图 3 所示, 编号为 0~32。其中 3~32 属于角度预测模式, 0~2 属于非角度预测模式。对色差 PU, AVS2 定义了 5 种帧内色度预测模式, 包括直流、水平、垂直、双线性和一个 DM 模式。其中 DM 模式根据对应编号的亮度 PU 的预测模式灵活确定色度 PU 的预测模式, 故帧内色度块的模式也包含在 33 种预测模式中^[1]。

在 AVS2 软件测试模型中, 首先获得所有帧内预测模式的预测块, 再通过对预测残差哈达曼变换计算出各个模式下的残差绝对值总和 (SATD), 选出 SATD 最小的前 9 种预测模式为粗选预测列表。最后对各粗选预测模式进行率失真优化计算, 得到帧内最优预测模式^[5]。

要确定帧内一个 LCU 下所有的划分帧内预测模式, 不但要遍历所有深度的 CU 尺寸, 还要对每个 CU 的所有预测划分 PU 以及所有 33 种角度预测模式进行遍历。可见 AVS2 在进行模式选择时, 编码复杂度非常高, 因此降低 AVS2 的帧内编码复杂度的研究非常必要。

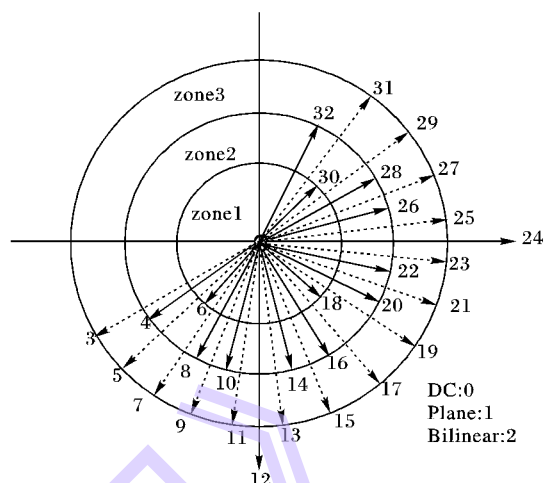


图 3 亮度预测块帧内预测模式

2 快速 AVS2 帧内模式选择算法

2.1 基于上下层 CU 深度之间的帧内预测模式选择算法

在 HEVC 编码框架中 LCU 采用了由上到下的编码结构, 先进行大块 CU 的预测, 再将大块 CU 划分成 4 个小块, 对 4 个下一层小块 CU 进行预测, 再划分, 直到最底层最小编码单元 (Smallest Coding Unit, SCU)。采用由上到下的预测方式的好处在于可以提前决定划分深度, 在一些深度划分决策的算法上存在优化的空间, 例如: 在一些比较平坦的序列 Beach、Crew 等中, 采用大块作的预测效果足够好, 可以提前中止划分, 直接采用大块模式作为最佳模式。而在 AVS2 参考软件代码中对一个 LCU 进行编码时, 是以类似树的后序遍历方法, 从 SCU 到 LCU, 由下而上进行编码处理。这样的好处在于预测完底层块后, 我们可以得到底层块的信息, 可以通过底层子块的信息对上层块进行一些判断。例如: 以深度为 3 的一个编码树为例 (如图 1), 对该树结构的编码单元进行编码。首先尝试编码标号为 5、6、7 和 8 的最小编码单元, 然后尝试编码它们所属的上层编码单元 (标号为 1 的编码单元), 最后比较这两种编码方式的率失真代价, 选择率失真代价较小的方式作为标号为 1 的 CU 的编码方式; 用同样的方法继续依次编码标号为 2、3、4 的 CU, 并得到它们的编码方式; 标号为 1、2、3 和 4 的 CU 编码完成后, 相当于完成了标号为 0 的 LCU 划分成 4 个子 CU 并分别编码的尝试, 将每个 CU 的率失真代价相加就得到这种方式的率失真代价; 尝试将标号为 0 的 CU 不进行子块划分方式进行编码, 并得到率失真代价; 最后比较这两种方式的率失真代价, 选择率失真代价较小的编码方式作为该 LCU 的最终编码方式。

可知, 在对当前层的 CU 进行尝试编码时, 其下层的 4 个子 CU 已完成了编码。由于 AVS2 是以四叉树的形式对编码单元进行划分, 4 个子 CU 组成一个父 CU, 对不同深度上下层之间的 PU 的帧内粗选预测列表进行实验观察, 发现源于同一 LCU 下的不同深度下 PU 之间帧内模式粗选列表 (Rough Mode Decision, RMD) 中大块 CU 的粗选列表中的方向基本上都包含在 4 个子 CU 的列表中, 尤其是列表中前几种代价较小的模式, 那么父 CU 的预测信息与其 4 个子 CU 的预测信息应该有较强的相关性。

根据在对当前深度下的 CU 进行尝试编码时, 其下层的 4 个子 CU 已完成了编码, 而父 CU 的预测信息与其 4 个子 CU 的预测信息有较强的相关性, 我们可对各个深度 CU 下 PU 的

帧内预测模式选择进行改进,进行如下的尝试:首先在 SCU 下的各个 PU 遍历所有的帧内预测模式,每个 PU 都会获得一个粗选模式列表,将源自同一个上层 CU 的 4 个子 SCU 下的所有 PU 的粗选模式列表组合成一个集合,为了更加准确将相邻块的预测模式(Most Probable Mode, MPM)添加进这个集合中;然后上层 CU 下的各个 PU 使用该集合作为粗选模式列表,进行预测模式遍历,获得各个 PU 的最佳预测模式,并将该层源自同一上层 CU 的 4 个子 CU 的所有 PU 粗选模式列表重新组成一个集合,用于其上层 CU 下的 PU 粗选模式列表,如此递归,完成一个 LCU 的编码。

算法是在当最大编码单元 LCU 划分成最小编码单元 SCU(8×8 尺寸)后开始执行的,具体步骤如下:

步骤 1 对分割后的 4 个 SCU 进行 PU 划分,对 CU 内的各个 PU 进行帧内粗选模式选择处理,统计保存当前深度下的粗选模式列表。

步骤 2 若当前编码块为 SCU,则重复步骤 1;若不为 SCU,则将当前 CU 下的 4 个子 CU 的所有 PU 粗选模式列表重新组成一个并集,选取前九种代价最小的模式作为当前 PU 下的粗选模式列表。

步骤 3 若当前 CU 不是 LCU 所属的深度,重复步骤 2 处理;若当前 CU 是 LCU,则进行步骤 2 处理,结束后进入下一个 LCU 编码。

2.2 底层角度判决算法

由于这样的优化在最底层的 8×8 块 CU 中没有得到体现,而 8×8 块的数量确实最多的,因此计算量也非常大,有很大的优化空间,所以有必要在 8×8 块中进行优化,减少编码计算复杂度。

图像的角度模式预测是与其纹理有相关性的,在与其纹理相关的几个预测方向获得的预测块是比较接近的。通过实验观察发现在帧内角度预测时,两个相邻角度获得的预测块其实是非常接近的。这是因为在两个相邻位置的预测模式其每个像素的预测像素是非常接近的。例如:在 AVS2 帧内预测中,图 2 中角度预测模式 3 和模式 4 都是先通过周边像素预测出首行的,首行的像素点通过块上边行的 4 个像素预测得到,通过实验观察模式 3 和模式 4 其预测的 4 个像素完全一样,因此得到的预测值也完全一样,之后的差别只是在角度扫描的过程中有些少许的差别,所以其得到的预测块非常接近。正是如此通过实验可以发现在粗选模式列表中,几种粗选模式都是比较接近的模式^[9]。在有些 HEVC 帧内优化文献中也提到选择一部分方向的预测模式而不是所有的预测模式去计算粗选代价,在文献[5]中提出一种基于边缘方向强度检测的快速帧内预测模式决策算法,将 35 种帧内预测模式根据图像的纹理特征划分为 5 个预测候选模式集合,每个集合中有 11 种预测模式,这些预测模式基本都是相邻近角度模式。文献[6]中利用 SATD 准则确定候选模式,以大幅度降低帧内预测模式选择的数量;然后,利用相邻宏块预测模式相关性,以进一步减少亮度块候选模式数量,提高模式选择速度。

与文献[5]中有 5 种预测候选模式集合不同,本文从 AVS2 帧内 30 种角度预测方向中均匀选取取出 {3, 6, 9, 12, 15, 18, 21, 24, 27, 30} 这 10 种预测模式方向组成一个预测模式集合,计算出其粗选 RdCost 代价值。类似于 H. 264/AVC 中的九种角度预测^[10],这 10 种方向均匀分布于帧内 33 种角度范围内,其中包含了水平、垂直、正负 45° 角以及 135° 角等各种角度,其基本上能够对应当前块的梯度纹理方向的各种情

况^[11],通过这 10 种方向的预测情况得到粗选列表。

首先,我们获得这些预测角度中代价最小的两个方向 $D1$ 和 $D2$,基本上 $D1$ 和 $D2$ 比较接近。接着,将 $D1$ 与 $D2$ 之间的方向,以及部分 $D1$ 与 $D2$ 相邻的方向组成粗选列表 $\{D1-2, D1-1, D1, \dots, D2, D2+1, D2+2\}$,如图 5 所示。考虑到相邻快帧内块之间的相关性,左边块及上边块的模式与当前块的最佳模式之间有着相当大的相关性,故将这两个的模式加入到粗选列表中。最后,考虑剩余 3 种非角度模式 DC、Plane、Bi,若其未存在粗选列表中则计算其代价大小,若小于 $D2$,则将它们放入粗选列表;反之则不放入粗选列表。这样我们最终所需计算的模式比原算法减少了将近 $2/3$ 。

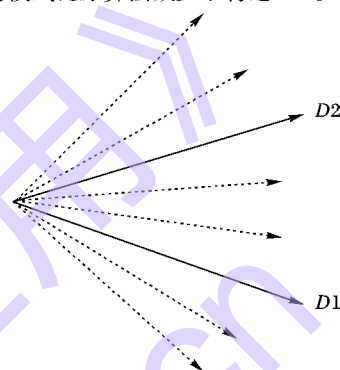


图4 $D1$ 与 $D2$ 角度模式

为验证本文底层角度判决算法特性,将文献[5]中的算法与本文算法作比较,测试了不同序列在不同量化参数(QP 为 27、32、38、45)下的平均编码时间、峰值信噪比(Peak Signal-to-Noise Ratio, PSNR)以及码率的变化情况,共测试了 5 个序列。其中 $\Delta Time$ 表示本文算法相对于文献算法的平均编码时间比,即(本文算法时间 - 文献算法时间)/文献算法时间; $BD_PSNR(Y)$ 表示了在给定的同等码率下,两种方法亮度峰值信噪比的差异; BD_Rate 表示在同样的客观质量下,两种方法的码率节省情况(负值表示编码时间减少,亮度 PSNR 降低和编码码率下降)^[12]。从表 1 可以看出本文算法在亮度 PSNR 上有所降低,但编码时间有所减少,编码码率也有所下降,因此本文算法具备一定的相对优势。

表1 本文算法与文献[5]算法比较情况

序列	$\Delta Time/\%$	$BD_PSNR(Y)$	$BD_Rate\%$
PartyScene_832 \times 480	-5.70	-0.20	3.63
BasketballDrill_832 \times 480	-1.80	-0.13	3.11
BasketballPass_416 \times 240	-2.22	-0.07	2.31
BlowingBubbles_416 \times 240	-0.66	-0.17	2.77
BQSquare_416 \times 240	-3.94	-0.12	3.87

根据以上两点所述,将两点结合成一种快速帧内预测判决方法,首先在底层 SCU 8×8 块中采用底层角度判决算法,然后从 SCU 到 LCU 所有 CU 层上实现基于上下层 CU 深度之间的帧内预测模式选择算法。

3 实验结果分析

为了测试本文提出的快速帧内模式选择算法的性能,将算法加入到 AVS2 视频编码标准的参考软件模型 RD8.0.1 中,与 AVS2 的传统的遍历所有预测模式的算法进行比较。根据表 1 给定的测试条件。本文通过式(1)进行数据分析,测试典型分辨率序列在不同量化参数(QP 为 27、32、38、45)

下的编码时间。对算法进行性能分析时,采用了编码比特率差值 (Bjontegaard Delta Bit Rate, BDBR) 和 PSNR 差值 (Bjontegaard Delta PSNR, BDPSNR)^[10] 来评估算法的整体率失真特性,它们是目前各视频编码标准最常用的客观性能检测指标,BDBR 在数值上主要体现为所提出的技术与参考的技术在编码比特率上差异的百分比,其值若为正数,表示所提出的技术较参考的技术占用更多的比特率;反之,则表示占用更少的比特率。同样,BDPSNR 表示所提出的技术与参考的技术在 PSNR 上的差异。

$$\Delta Time = \frac{Time_{proposed} - Time_{reference}}{Time_{reference}} \quad (1)$$

其中: $Time_{reference}$ 是原始算法在 RD8.0.1 上编码的总时间, $Time_{proposed}$ 是使用本文的快速帧内模式选择算法在参考软件模型 RD8.0.1 上编码的总时间,统计 4 个 QP 下的时间比之后取平均值。负值 BDPSNR 或正值 BDBR 表示编码质量的损失或者码率的升高,负值 Time 表示编码时间减少^[10]。

表 2 实验测试条件

编码帧结构	IPPP
测试序列	BasketballDrive、Kimonol、City、Crew、PartyScene、BQMall、BlowingBubbles、RaceHorses
量化参数 QP	27、32、38、45
最大编码单元(LCU)	64

为了验证算法的通用性,本文选择具有代表性的不同分辨率的视频在全 I 帧的条件下进行测试;其中 BasketballDrive 与 Kimonol 序列分辨率是 1920×1080 , City 与 Crew 序列分辨率是 1280×720 , PartyScene 与 BQMall 序列分辨率是 832×480 , BlowingBubbles 与 RaceHorses 序列分辨率是 416×240 。所使用的软件平台是 VS2013,表 3 为本文算法与遍历所有预测模式的方法编码时间及 BDBR 和 BDPSNR 的实验比较结果。

表 3 实验测试结果

分辨率	序列	本节算法与传统方法的实验比较		
		$\Delta Time/\%$	BDR/%	BDP/dB
1920 × 1080	BasketballDrive	19.01	1.55	-0.03
	Kimonol	18.12	1.13	-0.04
	Average	18.57	1.38	-0.03
1280 × 720	City	17.43	1.04	-0.03
	Crew	18.28	1.33	-0.03
	Average	17.85	1.18	-0.03
832 × 480	BQMall	15.83	0.57	-0.03
	PartyScene	15.12	0.40	-0.04
	Average	15.47	0.48	-0.03
416 × 240	BlowingBubbles	13.77	0.45	-0.02
	RaceHorses	14.76	0.63	-0.02
	Average	14.24	0.54	-0.02

从表 3 中可以看出,本文提出的快速帧内模式选择算法,在保证编码性能基本不变的前提下,对减小编码时间有明显效果,平均编码时间下降 16.5% 以上,而 BDBR 平均上升 0.89%,BDPSNR 下降平均 0.03 dB。在 QP 较大或分辨率较大时,编码时间下降较多,编码时间能下降到 18% 以上。由此可以看出,本文提出的模式判决方法有效地降低了编码复杂度。

4 结语

本文重点研究了在 AVS2 的帧内预测过程中,使用多角度模式带来的编码复杂度增加问题。针对这一问题,本文对帧内模式选择进行研究,实验分析不同序列、不同 QP 及不同深度下各预测模式的情况,提出一种快速帧内模式选择算法,从而减小遍历所有模式带来的计算量,同时对编码性能影响极小。实验结果表明,本文算法应用于 AVS2 参考软件中,可使编码时间平均下降超过 15%,而编码性能基本保持不变。因此,使用本文的预测模式选择算法,可以有效减小 AVS2 的帧内预测的编码时间,降低 AVS2 编码复杂度,从而有助于促进 AVS2 视频编码的进一步发展。

参考文献:

- [1] MA S. History and recent developments of AVS video coding standards[J]. Journal of Computer Research and Development, 2015, 52(1): 27-37. (马思伟. AVS 视频编码标准技术回顾及最新进展[J]. 计算机研究与发展, 2015, 52(1): 27-37.)
- [2] YU L, CHEN S, WANG J. Overview of AVS-video coding standards[J]. Signal Processing: Image Communication, 2009, 24(4): 247-262.
- [3] HE Z, YU L, ZHENG X, et al. Framework of AVS2-video coding [C]// Proceedings of the 2013 20th IEEE International Conference on Image Processing. Piscataway: IEEE, 2013: 1515-1519.
- [4] PIAO Y, CHEN J, LEE S, et al. Intra coding of AVS2 video coding standard[C]// Proceedings of the 2014 IEEE International Conference on Multimedia and Expo Workshops. Piscataway: IEEE, 2014: 1-5.
- [5] LEI H, WEI X, YANG Z, et al. A fast mode decision algorithm for intra prediction in HEVC[J]. Computer Engineering, 2014, 40(5): 270-273. (雷海军, 危雄, 杨张, 等. 一种快速 HEVC 帧内预测模式决策算法[J]. 计算机工程, 2014, 40(5): 270-273.)
- [6] CHEN Y, SU W, WANG C, et al. Optimization of intra mode decision for AVS[J]. Computer Science, 2013, 40(12): 86-89. (陈云善, 苏宛新, 王春霞, 等. AVS 帧内模式选择优化[J]. 计算机科学, 2013, 40(12): 86-89.)
- [7] LI C, PENG Z, CHEN F. Fast intra coding algorithm for HEVC based on texture direction and spatial correlation[J]. Journal of Optoelectronics · Laser, 2014, 25(9): 1777-1783. (李持航, 彭宗举, 陈芬. 基于纹理方向和空域相关的 HEVC 帧内快速编码算法[J]. 光电子·激光, 2014, 25(9): 1777-1783.)
- [8] PALOMINO D, CAVICHIOLI E, SUSIN A, et al. Fast HEVC intra mode decision algorithm based on new evaluation order in the coding tree block[C]// Proceedings of the 2013 30th Picture Coding Symposium. Piscataway: IEEE, 2013: 209-212.
- [9] SHI F, LIU Y. Fast intra mode decision algorithm for HEVC[J]. Video Engineering, 2013, 37(11): 8-11. (石飞宇, 刘昱. 一种 HEVC 快速帧内模式判断算法[J]. 电视技术, 2013, 37(11): 8-11.)
- [10] CHENG Y, TENG G, SHI X, et al. A fast intra prediction algorithm for HEVC[M]// Proceedings of the 9th International Forum on Digital TV and Wireless Multimedia Communication. Heidelberg: Springer, 2012: 292-298.
- [11] ZHAO W, SHEN L, CAO Z, et al. Texture and correlation based fast intra prediction algorithm for HEVC[C]// Proceedings of the 9th International Forum on Digital TV and Wireless Multimedia Communication. Heidelberg: Springer, 2012: 284-291.
- [12] BJONTEGAARDA G. Calculation of average PSNR differences between RD-curves, VCEG-M33[R/OL]. [2015-06-20]. http://ftp3.itu.ch/av-arch/video-site/0104_Aus/.