

基于图像特征的 HEVC 快速帧内预测算法

甘勇, 赵晓荣, 李天豹, 薛峰

(郑州轻工业学院 计算机与通信工程学院, 河南 郑州 450001)

摘要:针对新一代视频编码标准 HEVC 有极高的计算复杂度影响编码效率的问题,提出了一种快速帧内预测算法,利用 SAMD 值和相邻已编码块深度对块类型的选择进行预判断.实验结果表明,该算法与校验模型 HM5.0 相比,在保持峰值信噪比基本不变和码率略增的情况下,可节省 40.95% 的帧内编码时间.

关键词:HEVC;帧内预测;视频编码

中图分类号:TP301.6 **文献标志码:**A **DOI:**10.3969/j.issn.2095-476X.2014.01.019

Fast intra prediction algorithm based on picture feature for HEVC

GAN Yong, ZHAO Xiao-rong, LI Tian-bao, XUE Feng

(College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450001, China)

Abstract: Aiming at the problem that high efficiency video coding (HEVC) as a new generation of video coding standards increased the computation complexity and influenced the coding efficiency, a fast intra prediction algorithm was proposed. It used SAMD value and the depth of the adjacent encoded block to screen the block type selection. The experimental results showed that the algorithm provided averagely 40.95% reduction of intra coding time compared with the one accepted by HM5.0 with unimportant loss of PSNR and a slight increase of bite-rate.

Key words: high efficiency video coding (HEVC); intra prediction; video coding

0 引言

HEVC (high efficiency video coding) 作为新一代视频编码标准,于 2010 年 4 月在德国德累斯顿召开的 JCT-VC 会议上被提出,同时建立了测试模型 HM2.0^[1]. 经过近几年的发展与完善,2013 年 1 月 26 日,HEVC 被正式定为国际标准. HEVC 主要针对高清视频的高效编解码,旨在从本质上提高编码压缩率. HEVC 在秉承 H.264/AVC 的技术基础上做了大范围改进,其中包括帧内预测技术^[2],采用高度灵活的块结构^[3]等.但是,HEVC 在获得高性能的同

时也引入了极高的计算复杂度,不利于 HEVC 的发展和应用.因此,在保持 HEVC 编码效率的前提下,有效地降低算法的复杂度成为重要的研究课题.鉴于此,本文拟提出一种快速帧内预测算法,以减小编码时间,提高编码效率.

1 HEVC 帧内预测

1.1 HEVC 编码结构

新一代视频编码标准 HEVC 仍采用基于块的混合编码方案^[4],其框图如图 1 所示.与以往的混合编码方案相比,它提供了高度灵活的块划分结

收稿日期:2013-11-15

基金项目:国家自然科学基金项目(61302118);郑州轻工业学院博士基金项目(2010BSJJ006);郑州市科技创新团队计划项目(112PCXTD344)

作者简介:甘勇(1965—),男,湖南省株洲市人,郑州轻工业学院教授,博士,主要研究方向为分布式计算机系统与网络.

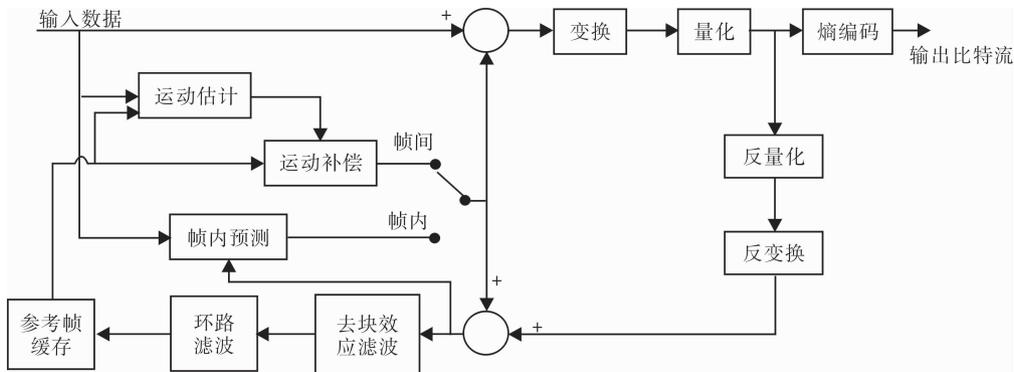


图 1 HEVC 编码框图

构,块结构包括 3 部分:编码单元 CU(coding unit), 预测单元 PU(prediction unit)和变换单元 TU(transform unit). 这种分离的块结构对每个单元的优化是非常有利的. CU 是进行帧内或帧间编码的基本单元块,它类似 H.264 中宏块的概念. CU 块始终是正方形,其尺寸从 8 像素 \times 8 像素到 64 像素 \times 64 像素,其中尺寸为 64 像素 \times 64 像素的 CU 为最大编码单元 LCU(largest CU). CU 块从 LCU 开始递归分裂成 4 个同样大小的块,一直可划分到 8 像素 \times 8 像素,这个过程将形成一个由 CU 块组成的基于内容自适应的编码树状结构. 用深度 $depth$ 来表示递归时 CU 划分的尺寸,定义 LCU 深度 $depth = 0$. 当 CU 进一步划分为 4 个子 CU,它们的尺寸是父 CU 的 $1/2$,此时 $depth = depth + 1$,依次递归划分下去,当子 CU 的尺寸为 8 像素 \times 8 像素时 $depth = 3$,该 CU 将不再继续划分,递归结构如图 2 所示^[5]. 预测单元 PU 仅用在二叉树结构的叶节点处的 CU,对于帧内预测,PU 的划分支持 2 种尺寸,即 $2N \times 2N$ 和 $N \times N$

(其中 N 的取值为 4, 8, 16 或 32). 除了 CU 和 PU 外,还有另一种与变换和量化相关的单元即 TU,其大小不能超过 CU 的尺寸.

1.2 帧内预测过程

1) 首先,获取图像块深度 $depth = 0$ 所对应的最大编码单元 LCU,并对其进行帧内编码,计算相应的率失真代价记为 $Rdcost_0$;

2) 然后,对 LCU 进行递归划分,划分为 4 个子 CU ($depth = 1$),分别对 4 个子 CU 进行帧内编码,计算 4 个子 CU 的率失真代价,将其率失真代价和记为 $Rdcost_1$,比较 $Rdcost_0$ 和 $Rdcost_1$,将较小值所对应的帧内预测模式作为 LCU 的最佳预测模式;

3) 同样地,对 $depth = 1$ 的子 CU 再次按照步骤 2) 的过程计算率失真代价,并找出对应的最佳预测模式,直至子 CU 的深度 $depth = 3$ 为止.

基于这种递归结构,编码器需要对所有 CU, TU 和 PU 进行线性组合,找到最佳方案,这是一个非常耗时的过程. 更重要的是,HEVC 帧内预测方向可支持 35 种,进一步提高了编码器的计算复杂度. 事实上,纹理平坦的区域适合大尺寸 CU 进行编码,而纹理复杂的区域则适合小尺寸 CU 进行编码,可直接跳过对大尺寸 CU 的编码,节省编码时间.

2 帧内预测模式选择的优化算法

2.1 基于 SAMD 的块类型选择

当选择 HEVC 帧内编码时,假如一个编码单元的纹理总体上看是平坦的,那么它适合采用大尺寸 CU 进行编码,可以提前中止模式选择过程. 文献 [5] 分别定义了 DC、水平及垂直方向上的纹理特性判断公式,本文将采用一种更简单的判断方法,通过计算图像块的像素值与平均像素值之间的差值 (SAMD),确定图像的纹理特性.

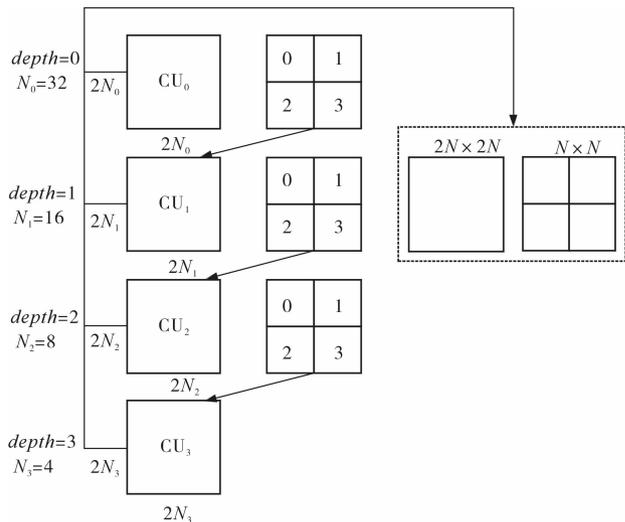


图 2 CU 划分结构示意图

在定义 *SAMD* 之前,先计算整个图像块的平均像素值. 如果 *SAMD* 值较小,说明图像块的像素值与平均像素值之前的差值较小,表示该图像块的纹理比较平坦,可以采用大尺寸 CU 进行编码;反之,则说明图像块的纹理比较复杂,细节较丰富,应该采用小尺寸 CU 进行编码.

$$\begin{cases} SAMD = \sum_{i,j} |pel(i,j) - Mean| \\ Mean = \frac{1}{Num} \sum_{i,j} pel(i,j) \end{cases}$$

其中, $pel(i,j)$ 代表图像块第 i 行第 j 列上的像素值, $Mean$ 表示图像块的平均像素值, Num 表示图像块像素点个数.

在量化参数 QP 相同的情况下,本算法对 64 像素 \times 64 像素到 8 像素 \times 8 像素的 CU 块都设置了上下 2 个阈值,它们的选取是按照参考文献 [6] 进行定义的. 根据这 2 个阈值来判断最佳块类型.

对于 64 像素 \times 64 像素的 CU 块,其上下阈值为

$$\begin{cases} \text{下阈值 } SAMD_{Tx} = 170QP + 1\ 560 \\ \text{上阈值 } SAMD_{Ts} = 1\ 040QP + 18\ 720 \end{cases}$$

对于 32 像素 \times 32 像素的 CU 块,其上下阈值为

$$\begin{cases} \text{下阈值 } SAMD_{Tx} = 90QP + 870 \\ \text{上阈值 } SAMD_{Ts} = 760QP + 13\ 680 \end{cases}$$

对于 16 像素 \times 16 像素的 CU 块,其上下阈值为

$$\begin{cases} \text{下阈值 } SAMD_{Tx} = 30QP + 290 \\ \text{上阈值 } SAMD_{Ts} = 260QP + 4\ 680 \end{cases}$$

对于 8 像素 \times 8 像素的 CU 块,其上下阈值为

$$\begin{cases} \text{下阈值 } SAMD_{Tx} = 12QP + 116 \\ \text{上阈值 } SAMD_{Ts} = 100QP + 1\ 800 \end{cases}$$

按照上述定义,分别计算 CU 块的 *SAMD* 值,如果小于下阈值 $SAMD_{Tx}$,则说明图像块之间的像素值相差不大,纹理比较平坦,可以直接采用当前块进行预测编码;反之,则说明图像块的细节较丰富,纹理比较复杂,需要进一步进行块的划分. 否则,需要根据相邻编码块和对应编码块的深度进行进一步的判断.

2.2 基于相邻编码块的深度分析

自然图像具有很强的空间相关性,而相邻编码块之间则具有更强的相关性. 笔者定义在前一帧中和现在欲编码块位置相同的 CU 称为对应 CU^[5]. 考虑到当前编码块 CU 与相邻已编码块 CU 和对应 CU 之前的关联性,在对编码单元进行帧内模式选择前,首先通过对当前 CU 与周围各个 CU 深度之间的对比,来判断当前 CU 细节是否丰富,从而判定是否跳过当前 CU 的模式选择过程. 各 CU 对应位置如图

3 所示.

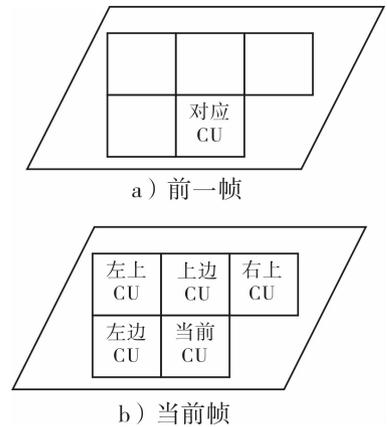


图 3 相邻 CU 及对应 CU 示意图

具体的评判过程如下:首先检测当前 CU 相邻的编码块,如左边 CU、上边 CU 及前一帧对应 CU,若它们的深度都大于当前 CU 深度,则说明当前 CU 纹理细节较为丰富,然后再判断左上方 CU 和右上方 CU 深度,并与当前 CU 深度作比较,如果前者仍大于或等于后者,则可以进一步判定当前 CU 细节非常细腻,适合较小尺寸 CU 进行编码,可跳过对当前 CU 的帧内预测模式搜索,直接对其进行分块;反之,则按正常程序进行,对当前块进行模式选择.

2.3 综合算法流程及算法分析

通过以上分析,本文算法流程如下:

- 1) 计算当前 CU 块的 *SAMD* 值,如果大于上阈值 $SAMD_{Ts}$,则跳过对当前 CU 块编码,直接跳到 4),如果小于下阈值 $SAMD_{Tx}$,则跳到 3);否则,进行下一步;
 - 2) 获取当前 CU 块的深度,与相邻 CU 及对应 CU 的深度作比较,如果相邻 CU 及对应 CU 的深度大于当前 CU,则跳过对当前块的编码,跳到 4),否则进行下一步;
 - 3) 采用当前块类型,并对其进行预测编码,然后跳到 5);
 - 4) 对当前 CU 进一步分块,然后对其 4 个小块进行预测编码;
 - 5) 结束当前块的编码.
- 流程图见图 4.

3 实验结果与分析

采用 HM5.0 参考软件作为实验平台,对 4 个测试序列各 30 帧进行测试,量化参数 QP 分别为 24, 32. 从峰值信噪比、比特率、编码时间 3 个方面比较快速算法与参考模型 HM5.0 的性能^[7].

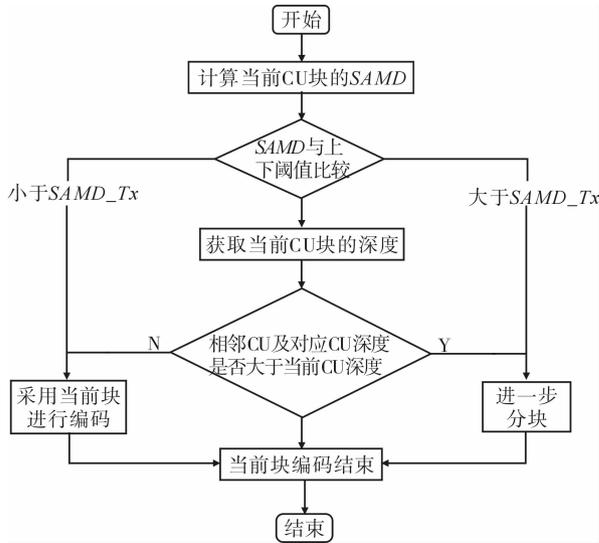


图4 算法流程图

表1 本文算法与 HM5.0 的比较

序列名	量化参数	峰值信噪比差值/dB	比特率差值/%	编码时间差值/%
Vidyo1	QP = 24	-0.031	0.72	-51.05
	QP = 32	-0.046	0.45	-52.89
Traffic	QP = 24	-0.025	0.11	-37.01
	QP = 32	-0.015	0.12	-38.22
BasketballDrive	QP = 24	-0.026	0.01	-44.85
	QP = 32	-0.007	0.01	-49.31
Racehorse	QP = 24	-0.023	0.09	-26.12
	QP = 32	-0.026	-0.02	-28.14

注:正值代表该算法与 HM5.0 相比增长值,负值代表该算法与 HM5.0 相比减少值,下同。

从表1和表2可以看出,与 HM5.0 相比,峰值信噪比平均减少 0.024 9 dB,比特率略微增加 0.186%,编码时间平均减少 40.95%。

4 结论

通过计算编码块的 SAMD 值和比较当前编码

表2 各个序列在 2 个 QP 下的平均数据

序列名	峰值信噪比差值/dB	比特率差值/%	编码时间差值/%
Vidyo1	-0.038 5	0.585	-51.97
Traffic	-0.02	0.115	-37.615
BasketballDrive	-0.016 5	0.01	-47.08
Racehorse	-0.024 5	0.035	-27.13
平均	-0.024 9	0.186	-40.95

CU 和相邻 CU 的深度,提出了一种基于图像特征的 HEVC 快速帧内预测算法.该算法可以跳过不必要的块预测,节省编码时间.与原算法相比,在保持视频质量基本不变的提前下,码率略增的基础上,减少了编码时间,有效地提高了编码效率。

参考文献:

- [1] 刘昱,胡晓爽,段继忠.新一代视频编码技术 HEVC 算法分析及比较[J].电视技术,2012,36(20):45.
- [2] 成益龙,滕国伟,石旭利,等.一种快速 HEVC 帧内预测算法[J].电视技术,2012,36(21):4.
- [3] Ugur K, Andersson K, Fuldseth A, et al. High performance, low complexity video coding and the emerging HEVC standard[J]. IEEE Trans Circuits and Systems for Video Technology, 2010, 20(12):1688.
- [4] Zhang L, Ma S W, Zhao D B. Fast mode decision algorithm for intra prediction in HEVC[C]//Proceedings of Visual Communications and Image Processing (VCIP), Tainan: IEEE, 2011.
- [5] 赵文强,沈礼权,张兆杨. HEVC 帧内预测算法的优化[J].电视技术,2012,36(8):31.
- [6] 陈王焕.新一代高效视频编码帧内预测快速算法研究[D].广州:中山大学,2012.
- [7] 谢翠兰,郑艺玲.基于 SAD 和 SATD 的 H.264 快速帧内预测算法[J].计算机工程,2008,34(10):215.