

文章编号 1004-924X(2014)05-1322-09

高效视频编码的快速编码单元深度 遍历选择和早期编码单元裁剪

蒋刚毅^{1, 2*}, 杨小祥¹, 彭宗举¹, 郁梅^{1, 2}, 邵枫¹, 陈芬¹

(1. 宁波大学 信息科学与工程学院, 浙江 宁波 315211;

2. 南京大学 计算机软件新技术国家重点实验室, 江苏 南京 210093)

摘要:为了降低新一代高效视频编码 (High Efficiency Video Coding, HEVC) 标准的编码复杂度,提出了一种基于四叉树结构类型分析和早期编码单元(Coding Unit, CU)裁剪的 HEVC 快速编码新算法。首先,通过分析已编码的最大编码单元(Largest CU, LCU)四叉树结构类型,确定其深度遍历区间(Depth Range, DR)类型。然后,利用相邻已编码的 LCU 以及前向参考帧和后向参考帧中坐标位置相同的 LCU 的 DR 类型来预测当前 LCU 的 DR 类型,并根据预测得到的 DR 类型对当前 LCU 设定 CU 深度遍历区间。最后,采用贝叶斯决策原理获取阈值,并利用该阈值在 CU 分割过程进行早期 CU 裁剪。实验结果表明:相对于原始 HEVC 编码结构,本文算法在随机访问模式下编码时间平均减少 41.55%,BDBR(Bjontegaard Delta Bit Rate)只增加约 1.94%,BDPSNR(Bjontegaard Delta Peak Signal-to-noise Rate)只降低了 0.06 dB;与 Shen 方案相比,该算法可以降低 12%左右的计算复杂度,BDBR 只增加约 1.09%,BDPSNR 只降低了 0.03 dB。

关键词:高效视频编码;编码单元(CU);CU 深度遍历选择;CU 裁剪;四叉树结构类型分析

中图分类号:TN919.81 **文献标识码:**A **doi:**10.3788/OPE.20142205.1322

Fast CU depth range selection and early CU pruning for HEVC

JIANG Gang-yi^{1, 2*}, YANG Xiao-xiang¹, PENG Zong-ju¹, YU Mei^{1, 2}, SHAO Feng¹, CHEN Fen¹

(1. Faculty of Information Science and Engineering, Ningbo University, Ningbo 315211, China;

2. National Key Laboratory of Software New Technology, Nanjing University, Nanjing 210093, China)

* Corresponding author, E-mail: jianggangyi@126.com

Abstract: To reduce the computational complexity of the emerging High Efficiency Video Coding (HEVC), a novel fast coding algorithm based on quad-tree type analysis and Coding Unit (CU) pruning was proposed. First, the CU Depth Range (DR) of an encoded Largest CU (LCU) was determined by analyzing its quad-tree type. Then, the DR of the current LCU was predicted based on those of neighboring LCUs and the co-located LCUs in the previously coded reference frame which was before or after current frame. The CU depth range of the current LCU could be determined based on the predicted DR. The proposed method also took advantage of Bayesian decision to obtain thresholds, by

收稿日期:2013-09-09;修订日期:2013-10-30.

基金项目:国家自然科学基金资助项目(No. 61171163, No. 61271270, No. 61271021, No. 61311140262, No. U1301257)

which early CU pruning could be determined. Experimental results show that the proposed method reduces encoding time by 41.55% on average in a random access case. Compared with the original HEVC encoding scheme, Bjontegaard delta bitrate (BDBR) increment is 1.94% and BD-PSNR loss is 0.06 dB. Compared with Shen's algorithm, it saves 12% encoding time with a BDBR increment of 1.09% and a BD-PSNR loss of 0.03 dB.

Key words: High Efficiency Video Coding (HEVC); Coding Unit (CU); CU Depth Range Selection (CUDR); CU pruning; quad-tree type analysis

1 引言

随着网络技术和终端处理能力的不断发展,远程监控、视频流媒体等领域对视频压缩的要求越来越高^[1-4]。视频编码联合专家组正致力于研究新一代高效视频编码标准(High Efficiency Video Coding, HEVC)^[5],其核心目标是在 H.264/AVC high profile^[6]的基础上,将压缩效率提高一倍。HEVC 在编码结构上采用超大尺寸四叉树结构的编码单元(Coding Unit, CU),预测单元(Prediction Unit, PU)和变换单元(Transform Unit, TU)^[7],显著地提高了压缩效率,同时也导致编码复杂度的增加。

HEVC 测试模型(HEVC Test Model, HM)^[8]采用递归的方式对最大编码单元(Largest CU, LCU)进行四叉树结构划分^[9]。在一个 LCU 块中,它使用递归的方式遍历所有深度的 CU,根据率失真优化^[10]进行最佳 CU 尺寸的选择,以实现 LCU 四叉树结构的划分及相关的 PU 块划分,这使得编码计算复杂度非常巨大。为了降低复杂度,Li 等人^[11]提出了一种自适应 CU 深度遍历(Adaptive CU Depth Range, ACUDR)方法。该方法主要通过空域相关性来预测当前 LCU 的 CU 深度,而大多数的 LCU 还是要遍历 3 个 CU 深度以上。因此,该算法节省时间相当有限。Shen 等人^[12]采用时空域联合预测的方式,通过相邻 LCU 深度值加权方式预测当前 LCU 的 CU 深度值,来确定其相应的深度遍历区间。该方法虽然很大程度上减少了遍历范围,但未考虑到视频内容及其序列的差异性,其固定权重导致预测深度值仍然与其给定的遍历区间存在相当大的改善空间。Cho 等人^[13]和 Tan 等人^[14]分别从帧内预测和帧间预测的角度,提出对 CU 裁剪进行预判以提前终止 CU 分割

过程,但是二者均未考虑在限制深度遍历区间的基础上进行提前终止 CU 分割过程。Sun 等人^[15]指出 CU 裁剪过程中总的率失真代价并不直接等于 4 个子 CU 的率失真代价之和,而是存在一定可能会造成误判的偏差,对于该偏差的获取,文中缺乏合适的理论建模过程。Jim 等人^[16]利用贝叶斯原理训练率失真代价阈值来进行 HEVC 帧内预测块选择,而该算法训练得到的阈值只对帧内预测的 CU 尺寸选择过程有效。

本文提出了一种新的 HEVC 快速编码算法。该算法通过对四叉树结构类型进行分析,确定其深度遍历区间,然后利用时空域相关性原理,预测当前 LCU 的深度遍历区间;同时,采用贝叶斯决策原理获取阈值,并利用该阈值在 CU 分割过程进行早期 CU 裁剪预判来进一步降低编码复杂度。

2 HEVC 编码复杂度分析

HEVC 采用率失真优化方式决定 LCU 最终四叉树结构的分割形式,这样,可以在保证视频质量的前提下获得最小的码率。其率失真代价函数 $J(CU_d)$ 为:

$$J(CU_d) = SSE_d + \lambda \times B_d, \quad (1)$$

其中: d 表示当前 CU 的深度值, B_d 代表当前 CU 经过各种 PU 预测和模式选择后编码所需的比特数, λ 为拉格朗日算子,其具体计算方法详见文献[7]; SSE_d (Sum of Square Error, SSE) 表示原始图像块和重建图像块之间的误差平方和,如式(2)所示:

$$\begin{cases} SSE = \sum_{i,j} Diff(i,j)^2 \\ Diff(i,j) = I_0(i,j) - I_R(i,j) \end{cases}, \quad (2)$$

其中: $I_0(i,j)$ 为原始图像块的像素值, $I_R(i,j)$ 为

重建图像块的像素值。

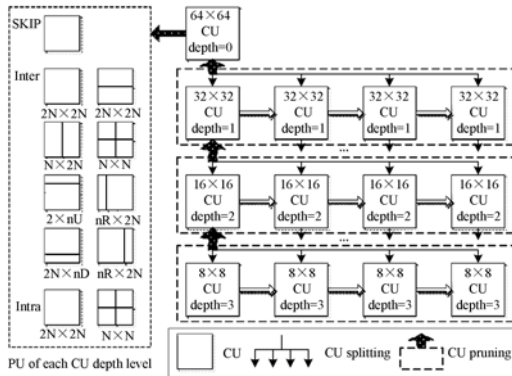


图 1 HM 中 CU 分割和裁剪过程示意图

Fig. 1 Illustration of CU splitting and pruning process in HM

如图 1 所示, HM 中的一个 LCU 的四叉树结构的划分过程如下: 首先进行 CU 的分割过程, 当 LCU 停止划分时, 尺寸为 64×64 的 CU ($\text{depth}=0$) 开始对该 CU 进行预测编码, 同时, 该 CU 还要进行各种 PU 预测和相应的模式选择, 最终得到其率失真代价 $J(\text{CU}_0)$ 。然后, 将该 CU 进一步划分为 4 个子 CU, 每个子 CU 的尺寸为 32×32 , 深度 depth 为 1。同理, 可得到每个子 CU 的率失真代价 $J(\text{CU}_{1,i})$ ($i=0, 1, 2, 3$)。如此递归地划分下去, 直至子 CU 的尺寸为 8×8 , 即 depth 为 3 时, 不再继续划分。接着, 从 depth 为 3 的 CU 开始进行 CU 块裁剪过程, 如果 4 个 8×8 大小的子 CU 的率失真代价之和小于其对应的 16×16 大小的 CU 的率失真代价, 则选择 8×8 的 CU 类型进行裁剪; 否则, 选择 16×16 的 CU 类型。如此类推, 直到选到 depth 为 0 的 CU。至此, 当前 LCU 的分割和裁剪过程完成。

根据上述分析, 确定一个 LCU 的最终分割形式, 需要对 CU 深度进行 $0 \sim 3$ 次的全遍历, 总共需要进行 $1+4+4 \times 4+4 \times 4 \times 4=85$ 次 CU 尺寸选择的率失真代价计算, 而每个 CU 还要进行各种 PU 预测和模式选择的率失真代价计算, 这使得编码计算复杂度非常大。

3 HEVC 快速 CU 深度遍历选择和早期 CU 裁剪新算法

在充分考虑到 HEVC 的四叉树编码结构和视频的时空域相关性的基础上, 本文提出图 2 所

示的快速剪裁算法。首先通过对四叉树结构类型进行分析, 确定其深度遍历区间 (Depth Range, DR) 类型, 利用参考帧相同位置的 LCU 和相邻 LCU 的 DR 类型来预测当前 LCU 的 CU 深度遍历区间; 然后, 再根据预测的深度区间进行编码。同时, 采用贝叶斯决策训练原理获取阈值, 并利用该阈值对 CU 分割进行早期 CU 裁剪。

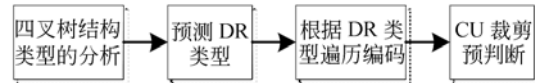


图 2 本文算法框架示意图

Fig. 2 Block diagram of proposed algorithm

3.1 DR 类型的确定

HEVC 采用四叉树结构进行编码, 一个 LCU 最终的分割类型是通过其 CU 深度值确定的, 大小为 64×64 的 LCU 包括 256 个 4×4 块, 每个 4×4 块都用一个深度值 depth ($\text{depth} \in [0, 3]$) 表示。不同四叉树结构的 LCU, 其 CU 深度值的分布情况也不同, 如图 3 所示。LCU 的四叉树结构类型可以分为 8 种 (即 A, B, C, D, E, F, G, H)。若能在 LCU 四叉树结构划分前对其类型进行预测, 通过该类型设定 CU 深度遍历范围, 就可以减少不必要的 CU 深度遍历, 从而降低其编码复杂度。但过于精细的预测遍历区间会很容易造成误判。因此, 本文将深度值分布相近的四叉树结构类型进行合并, 并定义如图 3 所示的 T_1 、 T_2 、 T_3 和 T_4 种 DR 类型。例如, T_2 包括 C 和 D 2 种四叉树结构类型, 其中: C 的深度值为 1 和 2, D 的深度值则全为 2, 故将这 2 种四叉树结构的 DR 类型定为 T_2 。本文算法根据 DR 类型进行 CU 深度遍历, 相比原始 $0 \sim 3$ 的全遍历, 明显减少了编码复杂度。

为了快速确定当前 LCU 的 DR 类型, 本文定义一种绝对 CU 深度误差和 (Sum of Absolute CU Depth Difference, SACUDD) 的代价函数, 计算如下:

$$\text{SACUDD}_d = \frac{1}{N} \sum_{i=0}^{N-1} |\text{depth}_i - d|, \quad (3)$$

其中: d 代表基准深度值, 其值为 0, 1, 2 和 3; N 为该 LCU 中 4×4 块的个数, depth_i 则表示该 LCU 中第 i 个 4×4 块所对应的深度值, 其取值范围为 $[0, 3]$ 。

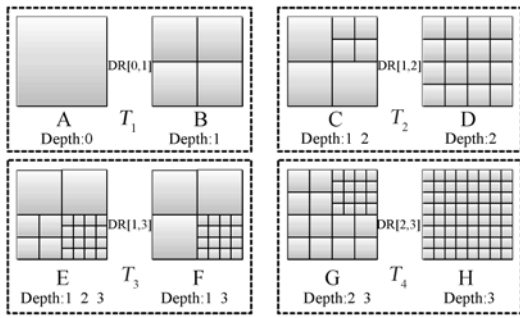


图 3 8 种二叉树结构类型及其所对应的 DR 类型
Fig. 3 Eight kinds of quad-tree structures and their DR types

表 1 A、B、C、D、E、F、G、H 所对应的 SACUDD 值
Tab. 1 SACUDDs of A, B, C, D, E, F, G, H

SACUDD 值	T_1		T_2		T_3		T_4	
	A	B	C	D	E	F	G	H
SACUDD ₀	0	1	1.25	2	1.75	1.5	2.25	3
SACUDD ₁	1	0	0.25	1	0.75	0.5	1.25	2
SACUDD ₂	2	1	0.75	0	0.75	1	0.25	1
SACUDD ₃	3	2	1.75	1	1.25	1.5	0.75	0

表 1 为图 3 中 A、B、C、D、E、F、G、H 8 种二叉树结构分割形式所对应的 SACUDD_d 值。表中阴影部分为 8 种常见二叉树结构分割形式对应的最小的 2 个 SACUDD_d 值。当出现 SACUDD_d 值相同的情况时(例如 B 和 D, 次小值 1 均出现 2 次), 则选择下标较小的 SACUDD_d 项作为参选项。

一个已编码 LCU 的 DR 类型 T_{cur} 可按照公式(4)所示方式进行判断, 即计算该 LCU 的 2 个参选项之和是否为 1, 若为 1 则其 DR 类型为 T_1 、 T_2 、 T_4 ; 由于 T_3 的 2 种二叉树结构类型的深度值都有 1 和 3, 因此, 若参选项之和大于 1, 则确定其为 T_3 类型。 T_{cur} 计算如下:

$$T_{cur} = \begin{cases} T_1, & \text{if } SACUDD_0 + SACUDD_1 = 1 \\ T_2, & \text{if } SACUDD_1 + SACUDD_2 = 1 \\ T_3, & \text{if } SACUDD_1 + SACUDD_2 > 1 \\ T_4, & \text{if } SACUDD_2 + SACUDD_3 = 1 \end{cases} \quad (4)$$

3.2 利用时空相关性的中值 DR 类型预测

在视频序列中, 当前编码 LCU 与相邻已编码的 LCU 以及前一帧和后一帧中与当前编码 LCU 位置相同的 LCU(以下简称为对应 LCU)之间有很大的相关性。这里, 通过图 4 中 Left LCU、Top LCU、Left-top LCU、Col1 LCU 和 Col2 LCU 共 5 个参选预测 LCU 的 DR 类型来预测当前编码 LCU 的 DR 类型。具体步骤为: 先根据上述方法获得 5 个参选预测 LCU 的 DR 类型, 再按表 2 中的映射关系确定 DR 值的候选列表, 将该列表进行从小到大排序, 取排序后的中值作为当前 LCU 预测的 DR 类型值 DR_{pred}。

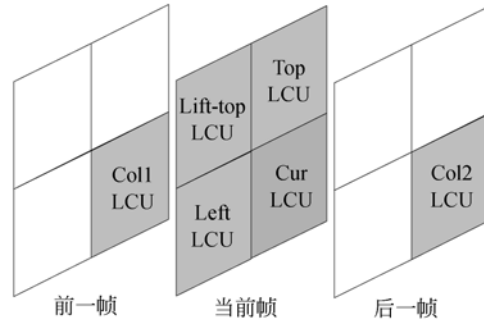


图 4 LCU 的时空域相关性
Fig. 4 Temporal and spatial correlation of LCUs

表 2 中在 4 种 DR 类型的基础上增加了 2 个特殊的 DR 值(1 和 5), 其中: $DR = 5$ ($depth \in [0, 3]$) 主要用于上述 5 个参选预测 LCU 不同时存在, 需要进行全遍历的情况, 而 $DR = 1$ ($depth \in [0, 2]$) 主要用来修正 DR 参考列表, 以减少误判。直接将 5 个参选预测 LCU 的 DR 值进行排序, 并取中值。但这样对于窄区间(例如: $[0, 1]$, $[1, 2]$, $[2, 3]$)而言, 可能会造成一定程度上的误判。假如: 假定 DR 参选列表为 $[0, 0, 0, 2, 2]$, 则其预测 DR 值为零, 而全遍历后的 DR 类型有可能是 T_2 , 从而造成误判。为了减少这种情况, 在 DR 参选列表中增加 2 个候选 DR 值(1 和 3), 此时, 上述 DR 参选列表扩展为 $[0, 0, 0, 1, 2, 2, 3]$, 其预测 DR 值为 1 (即 $depth \in [0, 2]$), 这将在一定程度上减少 T_2 类型的误判。当有 4 个以上参选 DR 值为零时, 说明此时预测 LCU 处在平坦区域, 故 $DR = 0$ 的可能性很大。同理, 增加 $DR = 3$ 是为了处理纹理复杂区域情况时, 减少对 T_4 类型的误判。因此,

在 DR 类型预测前对 DR 参选列表初始化为 [1, 5, 5, 5, 5, 5, 3]。表 3 给出了直接采用 5 个参选预测 LCU 的 DR 参选值和采用扩充的 7 个 DR 参选值时, DR 类型预测与原始 HM9.0 算法结果的相似性比较。由表 3 可得, 相比 5 个 DR 参选值而言, 7 个 DR 参选值列表的 DR 类型与原始 HM9.0 算法的相似度更高, 可在很大程度上减少误判, 这进一步验证了本方案的可行性。

表 2 DR 类型及其所对应的 DR 值

Tab. 2 DR types and their corresponding values

DR 类型	DR 值	[Depth _{min} , Depth _{max}]
T ₁	0	[0, 1]
—	1	[0, 2]
T ₂	2	[1, 2]
T ₃	3	[1, 3]
T ₄	4	[2, 3]
—	5	[0, 3]

表 3 DR 类型分布的统计分析结果

Tab. 3 Statistical analysis of DR type distributions

测试序列	5 个 DR 参选值/%				7 个 DR 参选值%			
	T ₁ [0, 1]	T ₂ [1, 2]	T ₃ [1, 3]	T ₄ [2, 3]	T ₁ [0, 1]	T ₂ [1, 2]	T ₃ [1, 3]	T ₄ [2, 3]
Traffic	80.3	82.1	86.8	85.5	88.5	90.6	92.2	89.0
Vidyo1	85.5	86.6	89.7	90.2	90.2	86.6	95.3	94.2
Cactus	86.6	84.5	86.2	82.6	86.6	93.4	92.1	87.6
Johnny	85.8	85.5	90.5	87.1	96.5	95.9	98.4	92.9
Kimono1	91.3	88.9	92.8	88.6	95.6	92.5	97.6	90.1
BQTerrace	92.6	90.5	91.5	94.3	97.3	98.2	95.6	99.3
FourPepple	86.2	82.9	88.6	90.2	90.6	86.5	90.1	89.6
ParkScene	87.3	86.9	87.2	88.9	92.1	90.3	95.3	91.0
平均	87.0	86.0	89.2	88.4	92.2	91.7	94.6	91.7

3.3 基于贝叶斯决策的早期 CU 裁剪预判

如图 5 所示, 在 LCU 的块划分过程中, 若当前 CU(depth ∈ [1, 2]) 为其上一层 CU 的第 4 个分割 CU (图 5 中的 CU_{1,3} 和 CU_{2,0,3}), 则在计算完其率失真代价 J 后, 就有可能通过预判提前进行早期 CU 裁剪过程。以 CU_{1,3} 为例, 若满足 $J(CU_0) < \sum_{i=0}^3 J(CU_{1,i})$, 则说明不分割块 CU₀ 比将其分割成 4 个分割块 CU_{1,i} (i = 0, 1, 2, 3) 更优。因此, CU_{1,3} 没有必要进行进一步分割, 可以提前对其进行 CU 裁剪。实际上, 完整 CU 裁剪过程中与 J(CU₀) 最终比较的率失真代价和 $\sum_{i=0}^3 J(CU_{1,i})$ 之间存在一定的差值

ΔJ。

本文采用公式(5)的判决条件对 CU 裁剪提前进行预判, 从而可以提前终止不必要的分割过程。

$$\begin{cases} \text{不分割, if } J(CU_{d-1}) < \sum_{i=0}^3 J(CU_{d,i}) + \Delta J_{Th} \\ \text{分割, 其它} \end{cases} \quad (5)$$

其中: d 表示当前 CU 所对应的深度; J(CU_{d-1}) 表示上一深度 CU 的率失真代价; $\sum_{i=0}^3 J(CU_{d,i})$ 表示 4 个子 CU 的率失真代价之和; ΔJ_{Th} 表示用贝叶斯决策训练出来的 ΔJ。

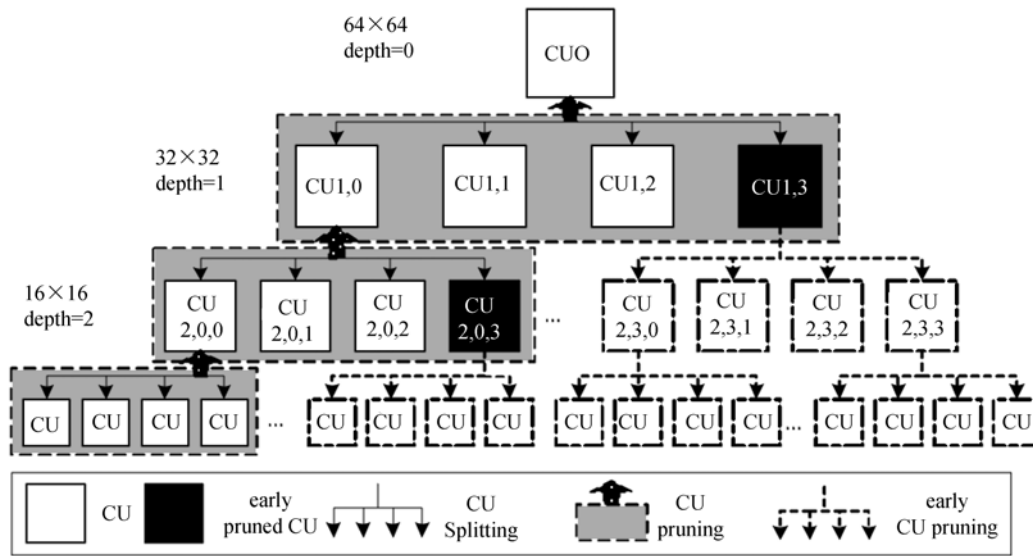


图 5 早期 CU 裁剪示例图

Fig. 5 Examples of early CU pruning

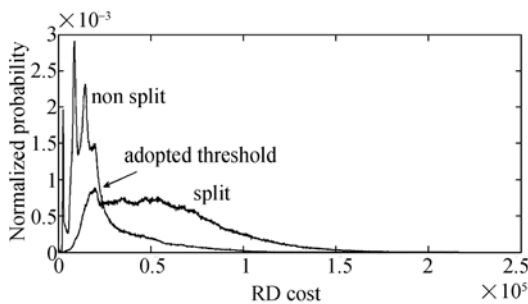


图 6 HM 中 CU 分割与不分割的归一化概率图

Fig. 6 Normalization probability of CU with/without splitting in HM

文献[15]提出采用贝叶斯决策原理在帧内模式选择过程中训练基于 SATD(Hadamard transformed SAD)的率失真代价最佳阈值。本文主要在 CU 的分割和裁剪过程中训练基于 SSE 的率失真代价最佳阈值 ΔJ_{Th} , 如式(6)所示。定义 ΔT 表示 $J(CU_{d-1})$ 和 $\sum_{i=0}^3 J(CU_{d,i})$ 之间的差值, 主要统计当前 CU 在分割和不分割条件下, 误差率最小时所对应的 ΔT 即为最佳阈值 ΔJ_{Th} 。

$$E(T) = \int_{\Delta T=0}^T P(\text{分割} | \Delta T) + \int_{\Delta T=T}^{\infty} P(\text{不分割} | \Delta T), \quad (6)$$

其中: E 表示误差率; T 表示最佳阈值; $P(\text{分割} |$

$\Delta T)$ 和 $P(\text{不分割} | \Delta T)$ 分别代表当前 CU 在分割和不分割条件下的归一化条件概率分布函数, 如图 6 所示, 图中 2 条曲线的交点所对应的率失真代价即为最佳阈值。经过反复采用式(6)所示方法, 得到以下测试结果: 当 $\text{depth}=1$ 时, ΔJ_{Th} 为 65; 当 $\text{depth}=2$ 时, ΔJ_{Th} 为 8。

4 实验结果及分析

为了验证算法的有效性, 在 HEVC 参考软件 HM-9.0^[17] 上实现本文算法并进行测试; 实验平台的硬件配置为 Intel(R) Core(TM) i7-2600 CPU, 主频为 3.40 GHz, 内存为 16.0 GB; 操作系统为 Windows 7; 开发工具为 Microsoft Visual Studio 2008。实验的主要编码参数^[18]为: 高效率配置, 随机访问模式, 编码帧数为 100 frame, 量化参数(Quantization Parameters, QP)分别为 22、27、32 和 37, GOP(Group of Pictures)大小为 8。采用 BDPSNR(Bjontegaard Delta Peak Signal-to-Noise Rate) 和 BDBR(Bjontegaard Delta Bit Rate)^[19] 分别表示相同码率条件下 PSNR 的变化量和相同 PSNR 条件下码率的变化百分比, 编码时间变化的百分比用 $\Delta T(\%)$ 表示:

$$\Delta T(\%) = \frac{T_{\text{pro}} - T_{\text{HM}}}{T_{\text{HM}}} \times 100\%, \quad (7)$$

式中: T_{pro} 、 T_{HM} 分别为本文所提算法和 HM 方法的编码时间。

实验中采用了通用测试条件^[18]中的 10 个测试序列进行测试, 本文除了与原始 HM9.0 方案进行比较外, 还选择在 HM 9.0 基础上对 Shen 方案^[10]的 ACUDR 算法进行横向对比, 结果如表 4 所示。由表 4 可得, Shen 方案的编码时间平均减少了 28.24%, BDPSNR 平均下降了 0.03 dB, 而 BDBR 平均只增加了 0.85%。本文采用二叉树结构类型分析改进的 ACUDR 算法可知, 相对于 HM9.0 方案, 平均编码时间减少达 37.88%, 与 Shen 方案相比, 其可进一步降低编码复杂度, 而 BDPSNR 平均下降了 0.01 dB, BDBR 平均只增加了 0.59%。总体算法时间平均减少可达 41.55%, BDPSNR 平均下降了 0.06 dB, BDBR 平均只增加了 1.94%。

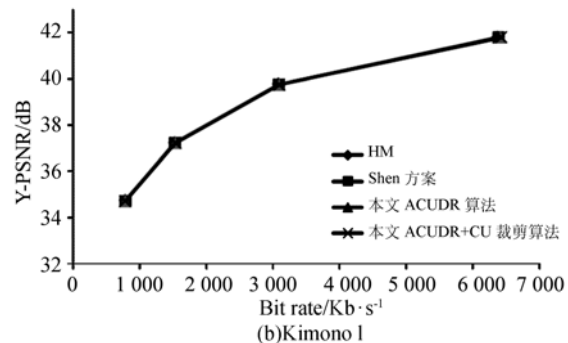
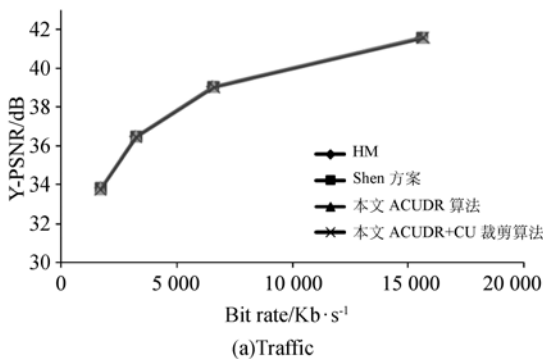
图 7 给出了在 Traffic、Kimono1、Johnny、BasketballDrill 4 个序列上 3 种方案 (Shen 方案、本文 ACUDR 算法和本文总体算法) 的率失真性能比较图。图 8 给出了上述 3 种方案的时间节省曲线。由图 7 和图 8 可得: (1) 本文 ACUDR 算法相比 Shen 方案可以进一步降低 10% 左右的计算复杂度, 其主要原因是本文通过二叉树类型分析预测 DR 类型, 从而有效地减少了深度遍历范围; (2) 本文总体算法对于不同的测试序列均能节省 45% 左右的计算复杂度; (3) 可以直观地看出 3 种方案的率失真性能相当。

图 9(a)~(b) 分别表示原始 HM9.0 方案和本文算法编码后的二叉树分割结果示意图, 其中, 方框代表二者的不匹配之处。由图 9 可得, 该区域在整个图像中占少数, 且分割深度相近, 因此, 整体率失真性能相对稳定。

表 4 各算法与 HM 对比的实验结果

Tab. 4 Comparison results of several algorithms and HM

类型	测试序列	Shen 方案			本文算法					
		BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)	ACUDR+CU 裁剪		
					BDBR (%)	BDPSNR (dB)	ΔT (%)	BDBR (%)	BDPSNR (dB)	ΔT (%)
Class A (2 560×1 600)	Traffic	0.82	-0.03	-31.26	1.73	-0.05	-40.73	2.39	-0.08	-44.04
	People On Street	1.53	-0.05	-20.30	2.23	-0.06	-35.06	2.78	-0.09	-36.59
Class B (1 920×1 080)	Kimono1	0.25	-0.01	-30.08	0.91	-0.03	-43.12	0.95	-0.03	-45.50
	BQ Terrace	0.15	0.00	-30.55	0.75	-0.01	-39.16	0.86	-0.02	41.29
Class C (832×480)	Basketball Drill	1.65	-0.05	-20.97	2.52	-0.07	-30.63	2.85	-0.08	-33.71
	Race Horses	0.86	-0.03	-19.30	1.15	-0.04	-31.95	1.28	-0.05	34.37
Class D (416×240)	BQ Square	0.86	-0.03	-25.47	1.64	-0.03	-35.58	2.07	-0.04	-40.56
	Basketball Pass	1.32	-0.04	-29.62	1.40	-0.04	-34.82	1.55	-0.05	-45.48
Class E (1280×720)	Johnny	0.51	-0.01	-38.49	0.85	-0.02	-44.05	1.95	-0.05	-47.28
	Four People	0.60	-0.02	-36.37	1.20	-0.04	-43.69	2.76	-0.10	-46.70
	平均	0.85	-0.03	-28.24	1.44	-0.04	-37.88	1.94	-0.06	-41.55



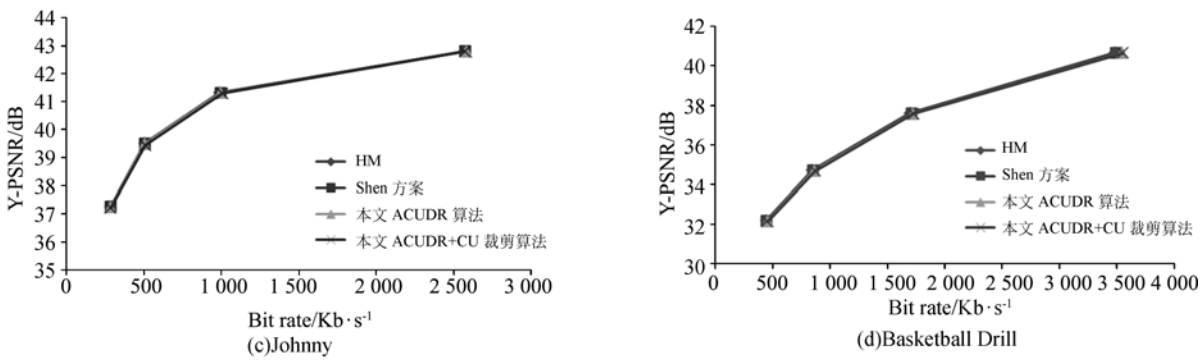


图 7 各方案的率失真性能

Fig. 7 RD performance of several algorithms

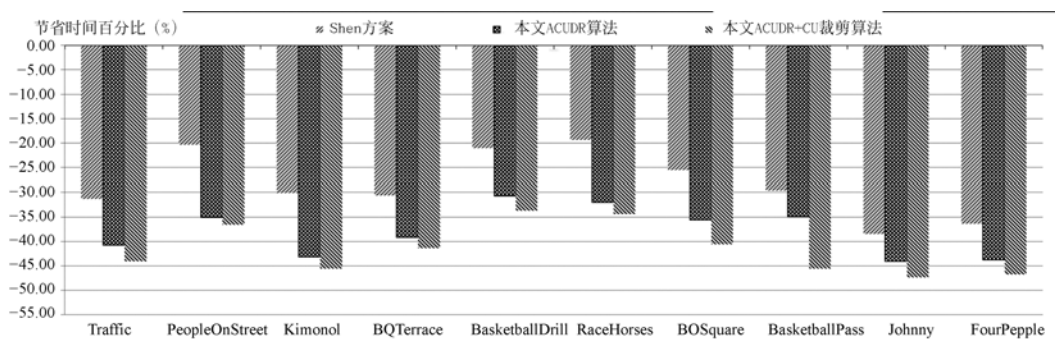


图 8 各算法相对于 HM9.0 的时间节省比重

Fig. 8 Time saving proportion of each algorithm by comparing with HM9.0

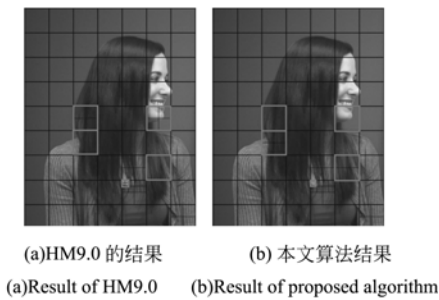


图 9 KristenAndSara 在 QP=32 条件下的四叉树分割结果

Fig. 9 Quar-tree partitioning results of KristenAndSara sequence with QP of 32

5 结论

本文提出了一种基于四叉树结构类型分析的

参考文献:

[1] 吴银花, 金龙旭, 张宁, 等. 针对 H.264 快速整像素运动估计算法[J]. 光学精密工程, 2013, 21(4): 1017-1025.
WU Y H, JIN L X, ZHANG N, et al.. Improve-

ment of fast integer pixel motion estimation algorithm for H.264 [J]. *Opt. Precision Eng.*, 2013, 21(4):1017-1025. (in Chinese)
[2] 祝世平, 张玲. 基于分形和 H.264 的视频编码系统 [J]. 光学精密工程, 2013, 21(3):774-781.

ACUDR 和 CU 裁剪的 HEVC 快速编码方法, 该方法的优点主要包括两个方面:其一为通过对四叉树结构类型的分析,提出了一种快速确定 LCU 的 DR 类型的方法,并利用该方法对现有的 ACUDR 算法进行了改进;其二为利用贝叶斯决策原理统计阈值,并利用该阈值在 CU 分割过程中进行早期 CU 裁剪预判。实验结果表明,本文提出的 ACUDR 算法可以有效地减少 CU 深度遍历范围,编码复杂度比 Shen 方案减少了 12% 左右,而 BDBR 只增加了约 1.09%,BDPSNR 降低了 0.03 dB,视频质量基本保持不变,总体算法的编码时间相对于原始的 HM9.0 算法平均减少了 41.55%,有效地降低了编码复杂度。下一步的研究可从帧间模式选择入手,减少预测选择范围,进一步降低编码复杂度。

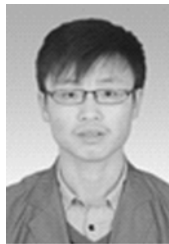
- ZHU SH P, ZHANG L. Video coding system based on fractal and H.264 [J]. *Opt. Precision Eng.*, 2013, 21(3):774-781. (in Chinese)
- [3] 张永祥, 卢岩, 栾中, 等. 视频图像运动补偿系统的设计与实现 [J]. *液晶与显示*, 2013, 28(3):424-428.
ZHANG Y X, LU Y, LUAN ZH, *et al.*. Design and implementation of motion compensation system based on FPGA [J]. *Chinese Journal of Liquid Crystals and Displays*, 2013, 28(3):424-428. (in Chinese)
- [4] 吴银花, 金龙旭, 张柯, 等. 应用 J_{mode} 值单调性的快速 P 帧模式选择算法 [J]. *液晶与显示*, 2013, 28(2):266-272.
WU Y H, JIN L X, ZHANG K. Fast P-frame mode decision algorithm using monotonicity of J_{mode} [J]. *Chinese Journal of Liquid Crystals and Displays*, 2013, 28(2):266-272. (in Chinese)
- [5] WIEGAND T, OHM J R, SULLIVAN G J, *et al.*. Special section on the joint call for proposals on High Efficiency Video Coding (HEVC) standardization [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2010, 20(12):1661-1666.
- [6] WIEGAND T, SULLIVAN G J, BJONTEGAARD G, *et al.*. Overview of the H.264/AVC video coding standard [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2012, 13(7):560-576.
- [7] BROSS B, HAN W J, OHM J R, *et al.*. High Efficiency Video Coding (HEVC) text specification draft 10 (for FDIS & Consent) [C]. *12th JCT-VC Meeting, JCTVC-L1003, Geneva*, 2013.
- [8] MCCANN K, BROSS B, HAN W J, *et al.*. High Efficiency Video Coding (HEVC) Test Model 10 (HM10) encoder description [C]. *12th JCT-VC Meeting, JCTVC-L1002, Geneva*, 2013.
- [9] SULLIVAN G J, OHM J, HAN W J, *et al.*. Overview of the High Efficiency Video Coding (HEVC) standard [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2012, 22(12):1649-1668.
- [10] SULLIVAN G J, WIEGAND T. Rate-distortion optimization for video compression [J]. *IEEE Signal Process Magazine*, 1998, 15(6):74-90.
- [11] LI X, AN J C, GUO X, *et al.*. Adaptive CU depth range [C]. *5th JCT-VC Meeting, JCTVC-E090, Geneva*, 2011.
- [12] SHEN L Q, LIU Z, ZHANG X P, *et al.*. An effective CU size decision method for HEVC encoders [J]. *IEEE Transactions on Multimedia*, 2013, 15(2):465-470.
- [13] CHO S, KIM M. Fast CU splitting and pruning for suboptimal CU partitioning in HEVC intra coding [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2013.
- [14] TAN H L, LIU F J, TAN Y H, *et al.*. On fast coding tree block and mode decision for High-Efficiency Video Coding (HEVC) [C]. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Kyoto*, 2012:825-828.
- [15] SUN H M, ZHOU D J, GOTO S S. A low-complexity HEVC intra prediction algorithm based on level and mode filtering [C]. *2012 IEEE International Conference on Multimedia and Expo (ICME), Melbourne*, 2012:1085-1090.
- [16] KIM J, CHOE Y, KIM Y G. Fast Coding Unit size decision algorithm for intra coding in HEVC [C]. *2013 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas*, 2013:637-638.
- [17] HM software [OL]. [2012-12-19] https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-9.0/.
- [18] BOSSEN F. Common test conditions and software reference configurations [C]. *12th JCT-VC Meeting, JCTVC-L1100, Geneva*, 2013.
- [19] BJONTEGAARD G. Calculation of average PSNR differences between RD-Curves [C]. *13th Video Coding Experts Group (VCEG) Meeting, VCEG-M33, Austin*, 2001.

导师简介:



蒋刚毅(1964—),男,浙江绍兴人,博士,教授,博士生导师,2000年于韩国 Ajou 大学获得博士学位,主要研究领域包括多媒体信号处理与通信研究、3D 视频系统编码与传输、图像与视频质量评价、数码显微信号处理等。E-mail:jianggangyi@126.com

作者简介:



杨小祥(1989—),男,湖南衡阳人,硕士研究生,主要从事新一代编码技术 HEVC、多媒体信息处理等方面的研究。E-mail:yangxiaoxiang@163.com