

一种基于纹理方向自适应预测和游程哥伦布编码的帧存无损压缩算法

罗瑜¹, 唐博²

(1. 陕西中医药大学基础医学院, 陕西西安 712046; 2. 电子科技大学航天航空学院, 四川成都 611731)

摘要: 为了进一步提高参考帧无损压缩的压缩性能, 本文提出了一种基于纹理方向预测和游程哥伦布编码的帧存无损压缩算法. 本算法首先采用双扫描和自适应预测的方法, 按纹理方向, 为每个像素选取最优的参考像素, 并进行预测以获得预测残差; 然后对预测残差进行游程混合熵编码, 从而提高了参考帧无损压缩的压缩性能. 实验结果显示, 与帧内预测游程哥伦布编码算法相比, 本文算法不但平均压缩率提高了 16%, 而且降低了平均编码时间.

关键词: 参考帧; 无损压缩; 预测; 熵编码

中图分类号: TN919.81

文献标识码: A

文章编号: 0372-2112 (2018)04-0969-06

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2018.04.027

A Frame Memory Lossless Compression Using Grain Directional Adaptive Prediction and Golomb Run-Length Entropy Coding

LUO Yu¹, TANG Bo²

(1. Department of Basic Medicine, Shaanxi University of Chinese Medicine, Xi'an, Shaanxi 712046, China;

2. School of Aeronautics and Astronautics, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China)

Abstract: To improve the compression performance, a frame memory lossless compression using grain-directional adaptive prediction and Golomb run-length entropy coding is proposed. First, the optimal reference pixels were obtained through double-scan and multi-directional prediction; then, the pixel differences were calculated and entropy-coded through Golomb and run-length. In this way, the compression performance is improved. Simulation results show that compared with those of reference frame lossless compression based on intra prediction and Golomb entropy coding, the average compression ratio is increased by 16% while the average coding time is reduced when using the proposed algorithm.

Key words: reference frame; lossless compression; prediction; entropy coding

1 引言

随着人们对视频质量需求的逐渐增加, 视频的图像分辨率作为视频质量的重要特性之一, 已经从 720p 和 1080p 过渡到目前市场主流的 4K 视频分辨率. 为了解决大分辨率视频传输造成的带宽紧张问题, 2013 年最新的 H. 265 视频编码标准^[1]被推出, 该编码标准以 H. 264 视频编码标准^[2]为基础, 并且较 H. 264 视频编码标准, 以增加 3 倍复杂度的代价, 换取了 2 倍的压缩效率.

与 H. 264 类似, H. 265 标准中存在大量的参考帧. 在硬件实现中, 过多的参考帧存储 (以下简称帧存) 不但会造成芯片面积成本的大幅度增加, 而且在帧间预

测中, 帧存的读写也会对总线带宽和功耗带来很大的冲击. 为了解决这一问题, 压缩技术^[3-5]中出现了一种新的分支, 即帧存压缩技术. 帧存压缩分为有损压缩和无损压缩两种, 为了保证编解码的一致性, 有损压缩大多同时应用于视频编解码器的两端; 而当视频编码器或解码器中任意一方没有采用有损压缩技术时, 另一方只能采用无损压缩. 在监控、电视等领域中, 为了降低芯片的成本和功耗, 帧存无损压缩技术被广泛的应用.

针对帧存无损压缩, 越来越多的技术^[6,7]被提出, 基于 MMSQ 定长编码^[8,9]和基于帧内预测游程变长编码的帧存无损压缩 (A Frame Memory Lossless Compression Using Intra Prediction And Golomb Entropy Cod-

ing, IP_GC)的改进技术^[10,11]成为主流研究方向,其中 IP_GC 的压缩率最为突出. 为了进一步提高帧内预测哥伦布编码的压缩率,本文提出了一种基于纹理方向预测和游程哥伦布编码的帧内无损压缩算法(A Frame Memory Lossless Compression Using Grain-Directional Adaptive Prediction And Golomb Run-Length Entropy Coding, GDP_GRLC). 仿真实验结果显示,本文算法较 IP_GC,在编码时间减少的前提下,压缩率有大幅度的提升;较 IP_GC 的改进算法,在压缩率几乎相等的情况下,编码时间大幅度减少.

2 IP_GC 算法分析

IP_GC 将预测单元(Prediction Unit, PU)、熵编码单元(Entropy Coding Unit, EU)和压缩单元(Compression Unit, CU)设为同样大小. 首先采用帧内预测方法得到预测残差(Prediction Difference, PD),然后通过哥伦布编码进行熵编码.

2.1 帧内预测方法

帧内预测方法在已知已编码 PU 与当前 PU 的相邻边界的前提下,根据水平垂直相关性,自适应的进行预测以得到 PD. 如图 1 所示,该方法具体步骤如下:

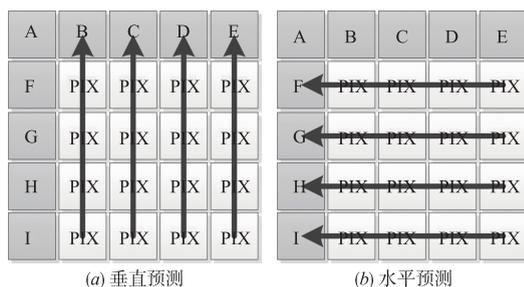


图1 预测模式

(1)进行垂直预测. PU 内的所有像素,参考当前 PU 的上相邻 PU 的最后一行,得到像素对应的 PD;并累加 PU 内所有 PD 的绝对值,得到垂直预测的绝对残差和(Sum of Absolute Difference, SAD).

(2)进行水平预测. PU 内的所有像素,参考当前 PU 的左相邻 PU 的最后一列,得到像素对应的 PD;并累加 PU 内所有 PD 的绝对值,得到水平预测 SAD.

(3)在水平预测 SAD 和垂直预测 SAD 之间,选取 SAD 最小的预测模式为最终预测模式,采用该预测模式计算得到 PD,同时将预测方向的标准位写入码流.

2.2 哥伦布编码

哥伦布编码将 PD 通过 k 值,拆分为商和余数 2 个分量,并分别进行熵编码. PD 的拆分具体步骤如下:

(1)将 PD 根据式(1)进行映射,以消除符号位.

$$PD = (PD < 0) ? (2 \times PD) : (2 \times PD + 1) \quad (1)$$

(2)将 PD 除以 2^k 得到余数和商,并根据表 1 将余数

和商分别进行编码. 如表 1 所示,商最大值为 8;当商为 8 时,余数的码字为 PD 且码长为 PD 的最大比特位宽.

表 1 哥伦布码表

商			余数	
商	码字	码长	码字	码长
0	0	0	余数	k
1	10	1	余数	k
2	110	2	余数	k
3	1110	3	余数	k
4	11110	4	余数	k
5	111110	5	余数	k
6	1111110	6	余数	k
7	11111110	7	余数	k
$> = 8$	111111110	8	PD	PD 的比特位宽

IP_GC 对于 CU 中每个 PD 采用同样 k 值进行编码. 对 EU 遍历所有 k 值后,以选出产生比特数最少的 k ,然后进行最终的熵编码. k 值范围为 $[0, 8]$.

2.3 算法分析

根据以上介绍的算法原理可以发现,传统的帧内预测和哥伦布编码都具有一定的缺陷,具体如下:

(1)因为 CU 中存在多种纹理的方向性,但是传统的帧内预测仅仅涉及到水平和垂直 2 种纹理的方向,所以导致纹理方向获取不充分;又因为在纹理方向上 CU 内的像素值也存在逐变的可能,传统的帧内预测将一行或一列的所有像素都参考同一个像素,所以导致 PD 不能进一步减小.

(2)对于图像平坦区域,大部分的 PD 为 0,甚至经常出现 PD 连续为 0 的情况. 但是传统的哥伦布编码并没有考虑到这一特征,而是对每个为 0 的 PD 都要消耗 1bit 进行编码,导致压缩效率的降低. 又因为传统哥伦布编码的商的码表,会随着商的增大而逐级增大,从而在残差较大时不但不能有效压缩,反而会出现膨胀.

(3)传统的哥伦布编码采用遍历 k 的方式,该方式不但增加了编码时间,而且还要为传递 k 值消耗比特.

2.4 IP_GC 改进算法分析

为改善以上缺点,Yongje Lee 在 2016 年提出了一种 IP_GC 改进算法(Advanced IP_GC, A_IP_GC^[12]),该算法首先采用 8 方向预测差分脉冲编码计算 PD,然后根据纹理相关性计算 k 值,最终采用 3 种哥伦布码表对 PD 进行编码. 目前,该算法效果最优.

A_IP_GC 相对于 IP_GC 算法,采用了更多的预测方向得到更优的 PD,并且针对 PD 的不同采用不同的哥伦布码表,最终大幅度提高了压缩率. 但是该算法存在如下缺陷:

(1)过多的预测方向判定和 3 种哥伦布码表的选择造成了过多的运算量.

(2)对于平坦区域残差几乎为 0 的特点没有针对

性设计,导致对于简单图像的压缩率不能进一步提高.

3 GDP_GRLC 算法

为了改进以上算法缺点,本文在不过分增加算法复杂度的基础上,提出一种 GDP_GRLC 算法,进一步提高了参考帧的压缩率.首先,该算法充分利用 PU 的纹理,为其中每个像素自适应的选择相关性最优的参考像素;其次,在得到 PD 后,根据 PD 的分布择游程或改进的哥伦布编码进行熵编码;最终在编码时间减少的情况下,进一步的提高了压缩率.

3.1 纹理方向自适应预测

为了更加充分的利用纹理方向性和渐变性,本文算法在不过分增加计算复杂度的前提下,采用双扫描方法,并利用水平、垂直、45 度和 135 度这 4 种像素相关性,为当前像素自适应的选取最优的参考像素.详细步骤描述如下:

步骤 1 由于纹理区域和边界区域存在一定的纹理相关性,所以采用双扫描的方式可以更大概率的获得更小的 PD(实验结果中进行测试比对).如图 2(a)所示,对当前 PU 进行正扫描;对于 PU 中任一像素,根据其周围相邻像素,计算得到水平、垂直、45 度和 135 度这 4 个方向的梯度值,并选取最小梯度值对应的方向,且与当前预测像素最为相邻的点作为参考像素.

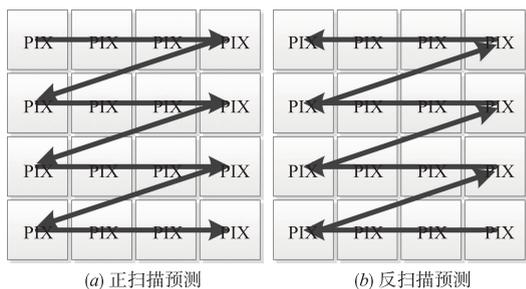


图2 扫描模式

如图 3 所示,若 $ABS(E-A)$ 最小,即 135 度纹理,那么参考像素为 B;若 $ABS(E-B)$ 最小,即垂直纹理,那么参考像素为 C;若 $ABS(E-C)$ 最小,即 45 度纹理,那么参考像素为 D;若 $ABS(C-B)$ 最小,即水平纹理,那么参考像素为 E.

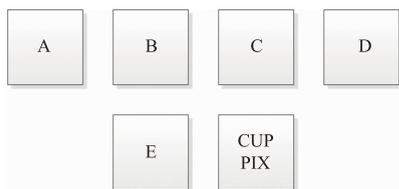


图3 参考点选取示意图

在参考像素点确定后,根据参考像素计算得到 PD,并根据式(1)消除 PD 的符号位.最终累加正扫描模式

下 PU 内所有的 PD,得到 $SAD_{forward}$.

步骤 2 如图 2(b)所示,按照反扫描模式重复步骤 1,得到 $SAD_{reverse}$.

步骤 3 在 $SAD_{forward}$ 和 $SAD_{reverse}$ 中,选择 SAD 最小的扫描方式,即为最终的扫描方式.并采用该方式计算得到 PU 内所有的 PD,同时将扫描方式的标准位写入码流,在解码端通过解析该标志位,获取扫描方式的信息,以用于解码和图像重建.

由于 PU 越小,PU 内纹理方向和像素渐变越一致,从而导致图像中为传输扫描方向标志位所消耗的总比特数越多,所以本算法限定 PU 单元大小为 4×4 或 8×8 .

3.2 游程哥伦布熵编码

本文算法采用游程和哥伦布混编的方法,该方法针对图像中平坦区域 PD 值大多为 0 的特点,大大提高了平坦区域的压缩率;其次本文算法利用一种新的哥伦布码表,极大程度缓解了 PD 膨胀的情况;最终本文算法提出自适应 k 值的方法,不但降低了传输 k 值的比特消耗,而且节约了编码时间.算法步骤如下:

步骤 1 若 EU 内所有 PD 都为 0,那么将游程标志位置为 1,并传送到解码端.

步骤 2 若 EU 内有任何 PD 不为 0,那么将游程标志位置为 0,并传送到解码端;同时,对 EU 内的 PD 逐个进行哥伦布编码.考虑到残差间的相关性,为了不传输 k 值,EU 的第一个 PD 的 k 设置为 0,后续 PD 的 k 根据上一个 PD 进行选取,选取规则如表 2 所示.由于图像中 PD 大部分较小,所以为了进一步减少 PD 传输 bit 数,本文提出一种新的码表,如表 3 所示.

表 2 k 值与 PD 的映射查询

上一个 PD 值	当前 PD 的 k
[0 - 1]	0
[2 - 3]	1
[4 - 7]	2
[8 - 15]	3
[16 - 31]	4
[32 - 63]	5
[64 - ∞]	6

表 3 一种新的哥伦布码表

商			余数	
商	码字	码长	码字	码长
0	1	1	余数	k
1	010	3	余数	k
2	110	3	余数	k
3	0000	4	余数	k
4	1000	4	余数	k
5	0100	4	余数	k
>5	1100	4	PD	PD 的比特位宽

由于 EU 越小时,选择游程模式的机率越大,而图

像中传输游程标志位所消耗的总比特数越多,所以本算法限定 EU 单元大小为 4 或 8.

4 实验结果与分析

本文的测试平台为 HM10.0^[13], 视频压缩标准为 H. 265, 测试序列根据纹理复杂度分为简单、中等、复杂 3 种共 9 个序列, 测试方法主要包括平均压缩率 (Average Compression Ratio, AVE_CR) 和平均相对编码时间 (Average Relative Encoding Time, AVE_AET). 为了充分验证本文算法理论, 我们进行 5 组实验, 分别为:

- (1) 最优的 PU 和 EU 选择测试;
- (2) 游程模式有效性测试分析;
- (3) 单双向扫描性能对比;
- (4) 不同算法实验结果对比;
- (5) 预测的方向有效性测试分析.

为了证明 A_IP_GC 中 8 种预测方向比本文算法 4 种预测方向的编码性能提升较小, 我们以 A_IP_GC 为基础版本, 去掉次要的 4 种角度 (22.5、67.5、112.5 和 157.5), 仅保留主要的 4 种角度 (45、90、135 和 180). 对比测试 8 种预测方向和 4 种预测方向的 AVE_CR, 实验结果如表 4 所示, 可以发现图像越简单, 4 种次要的预测方向越不能产生应有的压缩效果. 总体来看, 去掉 4 种次要预测方向, A_IP_GC 的 AVE_CR 仅降低 0.89%. 因此本文算法为了在不过分影响编码效率的前提下降低运算复杂度, 可以去除 4 种次要预测方向.

表 4 预测方向使用比例对比

序列	4 种预测方向 AVE_CR (%)	8 种预测方向 AVE_CR (%)	8 种 - 4 种 AVE_CR (%)
ducks_take_off	79.15	79.31	0.16
blue_sky	78.20	78.25	0.05
Johnny	76.94	76.98	0.04
crowd_run	71.98	72.56	0.58
old_town_cross	72.71	73.11	0.4
stockholm	68.95	70.01	1.06
rush_field_cuts	61.26	63.45	2.19
riverbed	63.34	65.12	1.78
mobcal	65.21	66.98	1.77
平均	70.86	71.75	0.89

最优的 PU 和 EU 选择测试: 为选出本文算法中最优的 PU 和 EU, 我们将 PU 和 EU 的大小进行两两组合, 形成 4 组参数进行测试. 由于同一图像中 4 组参数遍历像素数量和像素计算复杂度基本类似, 所以本文仅仅将 AVE_CR 作为衡量条件进行测试, 测试结果如表 5 所示. AVE_CR 计算过程为如式 (2) 所示, 其中, $Size_{is}$ 为压缩后的码流大小, $Size_{yuv}$ 为压缩前的原始 YUV 文件大小.

$$AVE_CR = (1 - Size_{is}/Size_{yuv}) \times 100\% \quad (2)$$

通过表 5 的实验数据可以发现, 相同 EU 下 PU 为 4×4 时最优; 相同 PU 下 EU 为 8 时最优. 当 PU 为 4×4

且 EU 为 8 时, 平均压缩率最高, 平均为 71.91%. 该结论与本文算法原理一致: 首先 PU 越小, 覆盖的纹理方向和渐变越一致, 预测效果越好; 然后由于大部分 PD 为 0 的区域是连续的, 因此 EU4 和 EU8 大部分重合, 且 U8 付出的游程标志位比特数比 EU4 减少 75%, 因此 EU8 要优于 EU4.

游程模式有效性测试分析: 为了测试本文算法的游程模式有效性, 我们统计不同纹理图像中游程模式采用的比例, 实验结果如表 6 所示. 可以发现, 图像越纹理越简单, 游程模式采用比例越高, 即纹理复杂图像比纹理简单图像, 游程模式采用的比例大幅度降低.

表 5 本文算法中 PU 和 EU 交叉组合压缩率对比

序列	PU8 × 8 EU8	PU8 × 8 EU4	PU4 × 4 EU8	PU4 × 4 EU4
	AVE_CR (%)	AVE_CR (%)	AVE_CR (%)	AVE_CR (%)
ducks_take_off	80.23	78.12	81.35	80.01
blue_sky	81.12	79.02	82.02	80.98
Johnny	80.06	78.34	79.32	79.01
crowd_run	69.23	67.23	72.15	69.91
old_town_cross	70.12	68.98	73.91	70.23
stockholm	68.16	66.65	69.98	67.12
rush_field_cuts	59.01	56.01	62.27	58.23
riverbed	59.42	55.78	62.75	58.46
mobcal	59.76	56.29	63.45	59.00
平均	69.68	67.38	71.91	69.21

表 6 游程模式使用比例对比

序列	哥伦布模式采用比例 (%)	游程模式采用比例 (%)
ducks_take_off	79.83	20.17
blue_sky	80.86	19.14
Johnny	80.48	19.52
crowd_run	91.55	8.45
old_town_cross	92.76	7.24
stockholm	90.64	9.36
rush_field_cuts	97.88	2.12
riverbed	96.55	3.45
mobcal	97.64	2.36
平均	89.80	10.20

单双向扫描性能对比: 为比较双扫描模式和单向正扫描模式的优劣, 我们在比较压缩率的同时, 也进行 AVE_AET 的比较测试, 测试结果如表 7 所示. AVE_AET 的计算过程如式 (3) 所示, 其中 $Time_{1scan}$ 为 H. 265 加本文算法单向正扫描的编码时间, $Time_{2scan}$ 为 H. 265 加本文算法双向扫描的编码时间.

$$\text{AVE_AET} = (\text{Time}_{2\text{scan}} / \text{Time}_{1\text{scan}} - 1) \times 100\% \quad (3)$$

表 7 本文算法双向扫描和单向正扫描的性能对比

序列	AVE_CR (%) 双向扫描	AVE_CR (%) 单向正扫描	AVE_AET (%) 双扫描较单扫描 运算时间增幅
ducks_take_off	81.25	81.26	27.36
blue_sky	82.02	82.23	27.36
Johnny	79.32	79.14	27.36
crowd_run	72.15	70.26	27.36
old_town_cross	73.91	71.11	27.36
Stockholm	69.98	68.48	27.36
rush_field_cuts	62.27	59.12	27.36
riverbed	62.75	58.99	27.36
mobcal	63.45	59.23	27.36
平均	71.91	69.98	27.36

通过表 7 的实验数据可以发现,首先由于预测复杂度的增加,双扫描方式较单向正扫描方式的平均运算时间增加 27.36%。其次,双扫描方式较单向正扫描方式的平均压缩率平均增长约 2%,并且图像越复杂,

AVE_CR 增长越多,该结论与本文算法原理一致。

不同算法实验结果对比:为了比较本文算法和 IP_GC、A_IP_GC 的压缩性能,我们同时进行了 AVE_CR 和 AVE_AET 的测试实验,实验结果如表 8 所示。此时 AVE_AET 计算公式如式(4)所示,其中 Time_{org} 为 H.265 不加帧存压缩的编码时间, $\text{Time}_{\text{other}}$ 为 H.265 加 GDP_GRLC、IP_GC 或 A_IP_GC 的编码时间。

$$\text{AVE_AET} = (\text{Time}_{\text{other}} / \text{Time}_{\text{org}} - 1) \times 100\% \quad (4)$$

根据表 8 的实验数据,相对于 IP_GC,本文算法因为采用了双扫描的像素级自适应预测,可以为每个像素选取最优的参考像素,又因为通过游程和哥伦布混编的方法,可以使熵进一步逼近理论极限,所以压缩比大幅度提高,最终较 IP_GC, AVE_CR 提高了 16%。对于 AVE_AET,虽然本文算法对像素选取参考点的判断条件更多,但是由于熵编码阶段不需要 k 值遍历,所以减少了运算复杂度;并且由于游程方法的使用,使越简单的序列花费的编码时间越少。

表 8 本文算法较 IP_GC 和 A_IP_GC 算法压缩性能对比

序列	AVE_CR (%)	AVE_CR (%)	AVE_CR (%)	AVE_AET (%)	AVE_AET (%)	AVE_AET (%)
	本文算法	A_IP_GC	IP_GC	本文算法	A_IP_GC	IP_GC
ducks_take_off	81.25	79.31	65.26	2.35	36.56	6.72
blue_sky	82.02	78.25	67.23	2.41	36.56	6.72
Johnny	79.32	76.98	65.14	2.32	36.56	6.72
crowd_run	72.15	72.56	55.76	2.45	36.56	6.72
old_town_cross	73.91	73.11	55.91	2.87	36.56	6.72
Stockholm	69.98	70.01	53.48	2.98	36.56	6.72
rush_field_cuts	62.27	63.45	46.43	3.61	36.56	6.72
riverbed	62.75	65.12	46.71	3.70	36.56	6.72
mobcal	63.45	66.98	47.34	3.67	36.56	6.72
平均	71.91	71.75	55.91	2.93	36.56	6.72

对于 Johnny 序列,本文算法的相对编码时间仅为 2.32%,而 IP_GC 的相对编码时间为 6.72%。最终较 IP_GC,本文算法的相对编码时间有大幅度降低。

相对于 A_IP_GC, A_IP_GC 过多的预测方向对于平坦区域有些冗余,并且本文算法的游程模式对于平坦区域有更好的压缩效果,因此对于简单序列,本文算法的 AVE_CR 较优。对于复杂纹理区域,更多的预测方向和多种码表可以起到更好的压缩效果,因此对于复杂序列, A_IP_GC 的 AVE_CR 较优。最终对于多序列平均 AVE_CR,本文算法和 A_IP_GC 几乎一致。但是对于 AVE_AET,由于 A_IP_GC 的过多预测方向和 3 种码表的判定选择,导致了该算法的计算复杂度过高,最终 A_IP_GC 的 AVE_AET 达到了 36.56%,而本文算法的 AVE_AET 仅有 2.93%,即较 IP_GC,本文算法的相对编码时间有大幅度降低。

5 总结

为了进一步减少 H.265 的参考帧存储,针对 IP_GC 和 A_IP_GC 的缺点,本文提出了 GDP_GRLC 算法。GDP_GRLC 算法首先充分利用 CU 的纹理方向一致性和渐变性,通过双扫描和像素级自适应预测的方法,将像素间的空间冗余度大幅度降低;然后采用游程哥伦布混编、自适应 k 和新的哥伦布码表 3 种方法进行熵编码,使编码熵进一步逼近理论熵的极限。实验结果表明,相对 IP_GC,本文算法的平均压缩率提高了 16%,同时相对编码时间减少了 3.79%。相对 A_IP_GC,本文算法在平均压缩率持平的情况下,相对编码时间减少了 33.63%。

参考文献

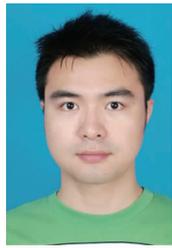
[1] 万帅,杨付正.新一代高效视频编码 H.265/HEVC:原

- 理、标准与实现[M]. 北京:电子工业出版社,2014. 94-265.
- [2] SCHWARZ H, MARPE D, WIEGAND T. Overview of the scalable video coding extension of the h. 26/avc standard [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2007, 17(9): 1103-1120.
- [3] 王亚文, 陈鸿昶, 等. 基于自适应分层结构的压缩分布场跟踪算法[J]. 电子学报, 2017, 45(4): 799-804.
WANG Ya-wen, CHEN Hong-chang, et al. Object tracking by compressive distribution fields with adaptive hierarchical structure[J]. Acta Electronica Sinica, 2017, 45(4): 799-804. (in Chinese)
- [4] 翟锐, 吕科, 等. 基于地形高度域的数据压缩算法研究[J]. 电子学报, 2016, 44(12): 2894-2899.
ZHAI rui, LÜ ke, et al. Research on terrain height field compression algorithm[J]. Acta Electronica Sinica, 2016, 44(12): 2894-2899. (in Chinese)
- [5] 胡春筠, 胡斌杰. 基于伪随机码置乱的分布式视频残差编码端码率控制算法[J]. 电子学报, 2016, 44(6): 1490-1495.
HU Chun-jun, HU Bin-jie. Encoder rate control algorithm based on scrambling with pseudo-random code for distributed residual coding of video[J]. Acta Electronica Sinica, 2016, 44(6): 1490-1495. (in Chinese)
- [6] Fan Y, Shang Q, Zeng X. In-block prediction-based mixed lossy and lossless reference frame recompression for next generation video encoding[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, 25(1): 112-124.
- [7] SILVERIA D, POVALA G, AMARAL L, et al. A low complexity and lossless reference frame encoder algorithm for video coding[A]. IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)[C]. Danvers: IEEE, 2014. 7408-7412.
- [8] GUPTA A D, AMRUTUR B, MEHENDALE M M, et al. Memory bandwidth and power reduction using lossy reference frame compression in video encoding[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2015, 21(20): 225-230.
- [9] Ma Y, Kang L. Adaptive granularity selection in reference picture memory compression[A]. Industrial and Control Engineering (MEIC)[C]. Turkey, 2015. 1158-1161.
- [10] Lee Y. A new frame recompression algorithm integrated with h. 264 video compression[A]. International Symposium on Circuits and Systems (ISCAS)[C]. Nagoya: IEEE, 2007. 1621-1624.
- [11] SAMPIO F, ZATT B, SHAFIQUE M, AGOSTINI L, et al. Content-adaptive reference frame compression based on intra-frame prediction for multi view video coding[A]. IEEE International Conference on Image Processing (ICIP)[C]. Melbourne: IEEE, 2013. 1831-1835.
- [12] LIAN X, LIU Z, ZHOU W, et al. Lossless frame memory compression using pixel-grain prediction and dynamic order entropy coding[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2016, 26(1): 223-235.
- [13] JCT-VC. HM10. 0 reference software [OL]. https://hevc.hhi.fraunhofer.de/svn/svn_hevcsoftware/trunk/, 2013-04-08/2016-08-22.

作者简介



罗 瑜 女, 1984 年 3 月出生, 陕西省咸阳人. 现为陕西中医药大学讲师, 研究方向为图形图像处理.
E-mail: luoyu2010@163.com



唐 博 男, 1983 年 1 月出生, 重庆大足人. 现为电子科技大学航空航天学院副教授, 主要研究方向为图形图像处理及通信信号处理相关工作.
E-mail: nnp1@163.com