



(12)发明专利

(10)授权公告号 CN 105376581 B

(45)授权公告日 2018.10.16

(21)申请号 201510787965.4

(22)申请日 2015.11.17

(65)同一申请的已公布的文献号
申请公布号 CN 105376581 A

(43)申请公布日 2016.03.02

(73)专利权人 复旦大学
地址 200433 上海市杨浦区邯郸路220号

(72)发明人 范益波 黄磊磊 刘淑君 曾晓洋

(74)专利代理机构 上海正旦专利代理有限公司
31200
代理人 陆飞 盛志范

(51)Int.Cl.
H04N 19/426(2014.01)
H04N 19/625(2014.01)

(56)对比文件

CN 102110058 A,2011.06.29,
US 2005018907 A1,2005.01.27,
CN 103327331 A,2013.09.25,
CN 104253998 A,2014.12.31,
“张淑芳,李华,刘义明,刘亚欣”.“基于
H.264的复杂度可分级的DCT算法研究”.《电子测
量技术》.2009,第32卷(第1期),

审查员 陈瑞宁

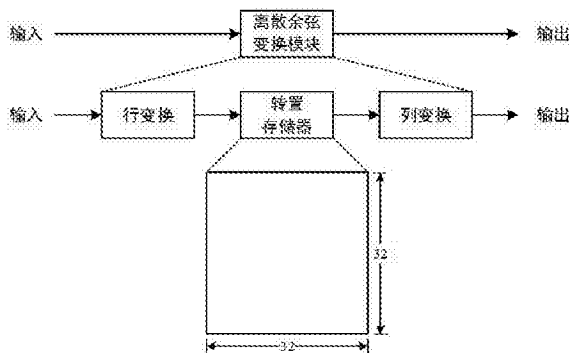
权利要求书1页 说明书4页 附图2页

(54)发明名称

一种适用于HEVC标准的基于指针的两级DCT
系数存储方法

(57)摘要

本发明属于高清数字视频压缩编解码技术
领域,具体为一种适用于HEVC标准的基于指针的
两级DCT系数存储方法。本发明将DCT系数分成三
个部分:符号位、高位数据和低位数据,并将数据
位的较高部分划归为高位数据,剩余比特划归为
低位数据;同时采用SRAM作为存储层次中的第一
级,用来存储符号位、低位数据和能够指向高位
数据的高位指针;采用寄存器作为存储层次中的
第二级,用来存储一些需要存储的高位数据;并
且使用计数器推算高位指针并进行索引。本发明
通过层次化的存储策略减少硬件代价。



1. 一种适用于HEVC标准的基于指针的两级DCT系数存储方法,其特征在于具体步骤为:
首先,将DCT系数分成三个部分:符号位、高位数据和低位数据;在补码表示的方式下,HEVC标准中的系数用16比特表示,其中,最高位是符号位,剩余比特是数据位;现将数据位的较高部分划归为高位数据,剩余比特划归为低位数据;

接着,SRAM作为存储层次中的第一级,用来存储符号位、低位数据和能够指向高位数据的高位指针;采用寄存器作为存储层次中的第二级,用来存储一些需要存储的高位数据;

当高位指针为一个特定的值0时,高位数据就是符号位的扩展;而对于需要存储高位数据,则它们将通过第二级存储层次即寄存器存储,并且使用计数器推算高位指针并进行索引。

2. 根据权利要求1所述的适用于HEVC标准的基于指针的两级DCT系数存储方法,其特征在于,仍假定高位指针等于0时,该指针所对应的高位数据是符号位的扩展;那么,在建立索引关系时,硬件使用一个计数器计数需要存储的高位数据的数目,该计数器从1开始记录,当且仅当一个新的需要存储的高位数据出现,硬件处理器将该数据存储于当前计数器所对应的寄存器中,接着计数器自增1。

一种适用于HEVC标准的基于指针的两级DCT系数存储方法

技术领域

[0001] 本发明属于高清数字视频压缩编解码技术领域,具体为一种适用于HEVC标准的两级DCT系数存储方法。

背景技术

[0002] HEVC(High Efficiency Video Coding)是由国际电信组织(ITU)和运动图像专家组(MPEG)联合成立的组织JCTVC提出的下一代视频编解码标准。目标是在相同的视觉效果的前提下,相比于上一代标准,即H.264/AVC标准,压缩率提高一倍。

[0003] 基于HEVC的视频编码器,主要由以下模块组成:帧内预测、帧间预测、变换、量化、反量化、反变换、重建、去方块滤波器、自适应样点补偿等。其中,离散余弦变换(DCT)是编码器必不可少的一部分,负责集中残差数据的能量,以供随后的量化和熵编码。由于HEVC标准下图像处理块的大小已经到达了 64×64 ,离散余弦变换(DCT)处理块的大小也到达了 32×32 。这使得常用的行列变换架构中所需的转置存储器变得异常庞大,如图1所示。对于如此庞大的存储需求,传统的基于寄存器的和基于静态随机存储器的存储方式已经变得不再适用。

发明内容

[0004] 本发明的目的在于提出一种可以克服现有技术不足的、能有效适用于HEVC标准的两级离散余弦变换(DCT)系数存储方法。

[0005] 本发明提出的适用于HEVC标准的两级DCT系数存储方法,是基于指针的,具体步骤为:

[0006] 首先,将离散余弦变换(DCT)系数(以下简称系数),分成三个部分:符号位,高位数据和低位数据。在补码表示的方式下,HEVC标准中的系数可以用16比特表示。其中,最高位是符号位,剩余比特是数据位。而更进一步地本发明将数据位的较高部分划归为高位数据,剩余比特划归为低位数据。具体的划分方法可以选择不同的比重,如图2所示的将高11位划为高位数据,低4位划为低位数据。

[0007] 接着,采用静态随机访问存储器(SRAM)作为存储层次中的第一级,这一级用来存储符号位、低位数据和能够指向高位数据的高位指针;采用寄存器作为存储层次中的第二级,这一级用来存储一些需要存储的高位数据。

[0008] 此处,需要存储与否取决于该高位内的所有比特是否都等于符号位。在补码表示下,如果一个数接近于0,那么该数较高几个比特都将是符号位的扩展。而残差经过离散余弦变换(DCT)之后,大量的数据都将接近于0。对于这些数,只要知道了其符号位,就可以得到高位数据,因此,他们是没有必要被存储的。本发明约定在这种情况下,高位指针被设置为一个特定的值,如0。那么,当高位指针的值为该值,即0时,高位数据就是符号位的扩展。而对于需要存储高位数据,则他们将通过第二级存储层次,即寄存器存储,并且使用计数器推算高位指针并进行索引。以上描述的存储层次如图3所示。

[0009] 为了简单地建立索引关系,本发明可以采用(但不仅限于)使用计数器。仍假定高位指针等于0时,该指针所对应的高位数据是符号位的复制。那么,在建立索引关系时,硬件可以使用一个计数器计数需要存储的高位数据的数目。该计数器从1开始记录,当且仅当一个新的需要存储的高位数据出现,硬件处理器就将该数据存储于当前计数器所对应的寄存器中,接着计数器自增1。

[0010] 本发明通过层次化的存储策略减少硬件代价。

附图说明

[0011] 图1为离散余弦变换(DCT)与转置存储器。

[0012] 图2为对于系数的划分。

[0013] 图3为两级存储策略图示。

具体实施方式

[0014] 下面通过实例,进一步具体描述本发明方法。

[0015] 假设当前需要存储的16×16系数矩阵如下表所示,且按照光栅顺序存储。

[0016]

24	2	-7	-1	-1	0	4	-2	0	0	2	-1	0	-1	0	0
7	-17	21	1	2	0	-2	0	0	1	-1	1	-3	1	0	0
0	33	-5	8	-3	-1	2	1	-1	-2	3	-4	2	-2	1	0
-4	4	1	-6	3	1	-1	0	0	3	-1	1	0	-2	-1	0
-19	5	4	1	-5	0	-1	-1	-1	2	0	-1	3	0	1	0
6	-3	1	-3	8	-3	2	0	2	-1	1	-1	-1	0	0	-1
0	-2	2	-1	-2	3	-1	1	3	2	0	1	0	0	0	0
8	-5	-3	8	-1	-3	0	0	-3	0	1	0	1	1	1	-1
-2	7	-4	-3	2	5	0	0	-1	0	1	-1	-1	0	0	0
-1	0	-1	3	-5	-2	4	0	0	1	-2	1	0	1	0	0
-1	4	1	-5	7	3	-2	-1	0	0	-1	0	0	0	0	0
1	-6	8	0	-10	4	2	1	0	-1	-1	1	0	-1	0	0
1	-1	-4	8	-1	-3	-2	3	-1	0	0	0	0	0	0	0
1	0	-2	1	-2	4	0	-2	1	-1	1	0	0	0	0	0
-2	2	-5	4	0	-7	2	3	0	0	0	0	0	0	0	0
0	0	-1	-2	6	0	-2	-4	1	0	0	0	0	0	0	0

[0017] 那么,本发明将有如下写入过程:

[0018] 初始情况下,所有静态随机存储器(SRAM)和寄存器的存储空间都是0,计数器是1。

[0019] 对于第0行第0列的系数24,它所对应的二进制补码是0000_0000_0001_1000,其中,符号位为0,高位数据为000_0000_0001,低位数据为1000。由于高位数据不是符号位的扩展,因此,高位数据被存储在1号寄存器中;符号位0,指针1和低位数据1000被存储在静态随机存储器(SRAM)的0地址中;计数器随后自增1,变为2。

[0020] 对于第0行第1列的系数2,它所对应的二进制补码是0000_0000_0000_0010,其中,符号位为0,高位数据为000_0000_0000,低位数据为0010。由于高位数据是符号位的扩展,

因此,不需要存储高位数据;符号位0,指针0和低位数据0010被存储在静态随机存储器(SRAM)的1地址中;计数器不变。

[0021] 对于第0行第2列的系数-7,它所对应的二进制补码是1111_1111_1111_1001,其中,符号位为1,高位数据为111_1111_1111,低位数据为1001。由于高位数据是符号位的扩展,因此,不需要存储高位数据;符号位1,指针0和低位数据1001被存储在静态随机存储器(SRAM)的2地址中;计数器不变。

[0022] 对于第0行第3列的系数-1,它所对应的二进制补码是1111_1111_1111_1111,其中,符号位为1,高位数据为111_1111_1111,低位数据为1111。由于高位数据是符号位的扩展,因此,不需要存储高位数据;符号位1,指针0和低位数据1111被存储在静态随机存储器(SRAM)的3地址中;计数器不变。

[0023] ……。

[0024] 对于第1行第0列的系数7,它所对应的二进制补码是0000_0000_0000_0111,其中,符号位为0,高位数据为000_0000_0000,低位数据为0111。由于高位数据是符号位的扩展,因此,不需要存储高位数据;符号位0,指针0和低位数据0111被存储在静态随机存储器(SRAM)的16地址中;计数器不变。

[0025] 对于第1行第1列的系数-17,它所对应的二进制补码是1111_1111_1110_1111,其中,符号位为1,高位数据为111_1111_1110,低位数据为1111。由于高位数据不是符号位的扩展,因此,高位数据被存储在2号寄存器中;符号位1,指针2和低位数据1111被存储在静态随机存储器(SRAM)的17地址中;计数器随后自增1,变为3。

[0026] 对于第1行第2列的系数21,它所对应的二进制补码是0000_0000_0001_0101,其中,符号位为0,高位数据为000_0000_0001,低位数据为0101。由于高位数据不是符号位的扩展,因此,高位数据被存储在3号寄存器中;符号位0,指针3和低位数据0101被存储在静态随机存储器(SRAM)的18地址中;计数器随后自增1,变为4。

[0027] ……。

[0028] 本发明将有如下读出过程:

[0029] 对于第0行第0列的系数24,它被存储在静态随机存储器(SRAM)的0地址,符号位是0,指针是1,低位数据是1000;由于指针是1,因此,高位数据被存储在1号寄存器中,内容是000_0000_0001。所以,最后的值是0_000_0000_0001_1000,也就是24。

[0030] 对于第0行第1列的系数2,它被存储在静态随机存储器(SRAM)的1地址,符号位是0,指针是0,低位数据是0010;由于指针是0,因此,高位数据就是符号位的扩展,也就是000_0000_0000。所以,最后的值是0_000_0000_0000_0010,也就是2。

[0031] 对于第0行第2列的系数-7,它被存储在静态随机存储器(SRAM)的2地址,符号位是1,指针是0,低位数据是1001;由于指针是0,因此,高位数据就是符号位的扩展,也就是111_1111_1111。所以,最后的值是1_111_1111_1111_1001,也就是-7。

[0032] 对于第0行第3列的系数-1,它被存储在静态随机存储器(SRAM)的3地址,符号位是1,指针是0,低位数据是1111;由于指针是0,因此,高位数据就是符号位的扩展,也就是111_1111_1111。所以,最后的值是1_111_1111_1111_1111,也就是-1。

[0033] ……。

[0034] 对于第1行第0列的系数7,它被存储在静态随机存储器(SRAM)的16地址,符号位0,

指针是0,低位数据是0111;由于指针是0,因此,高位数据就是符号位的扩展,也就是000_0000_0000。所以,最后的值是0_000_0000_0000_0000,也就是7。

[0035] 对于第1行第1列的系数-17,它被存储在静态随机存储器(SRAM)的17地址,符号位1,指针是2,低位数据是1111;由于指针是2,因此,高位数据被存储在2号寄存器中,内容是111_1111_1110。所以,最后的值是1_111_1111_1110_1111,也就是-17。

[0036] 对于第1行第2列的系数21,它被存储在静态随机存储器(SRAM)的18地址,符号位0,指针是3,低位数据是0101;由于指针是3,因此,高位数据被存储在3号寄存器中,内容是000_0000_0001。所以,最后的值是0_000_0000_0001_0101,也就是21。

[0037] ……。

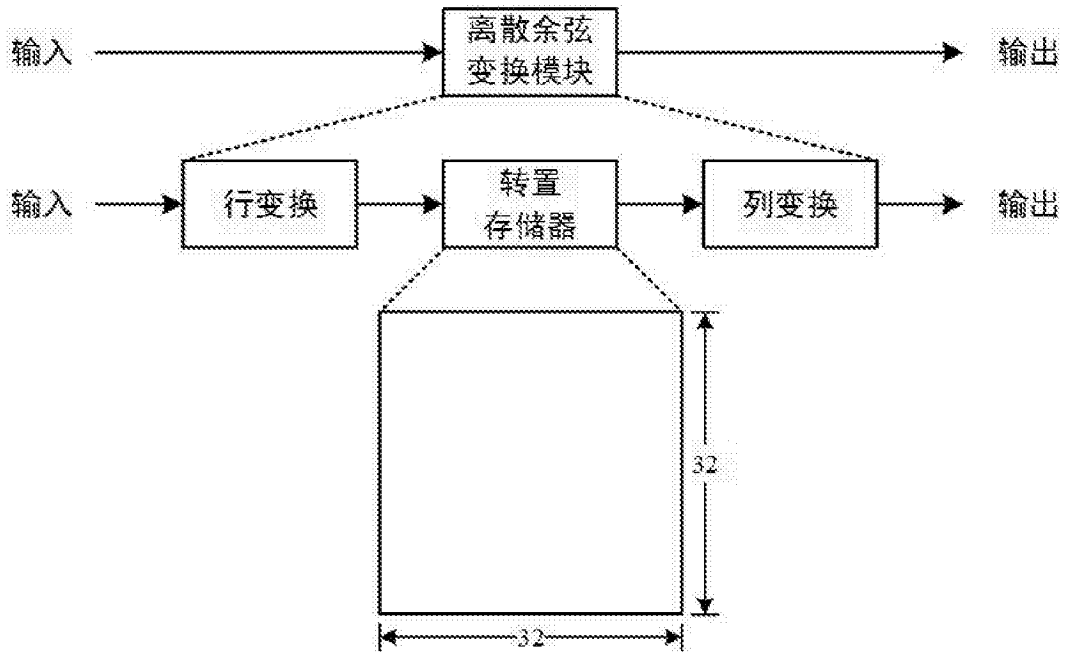


图1

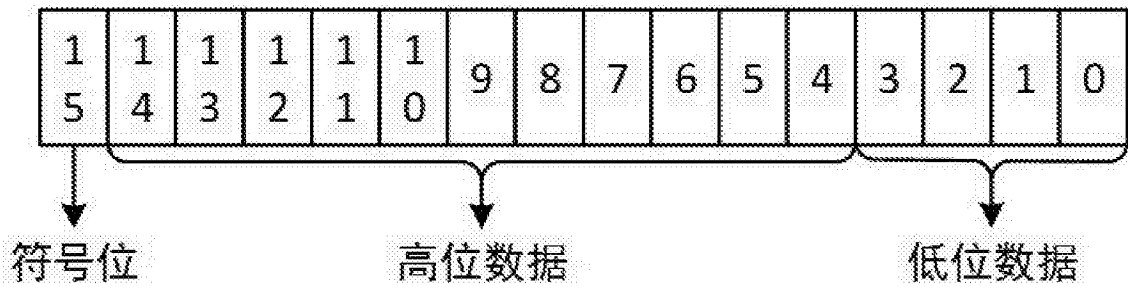


图2

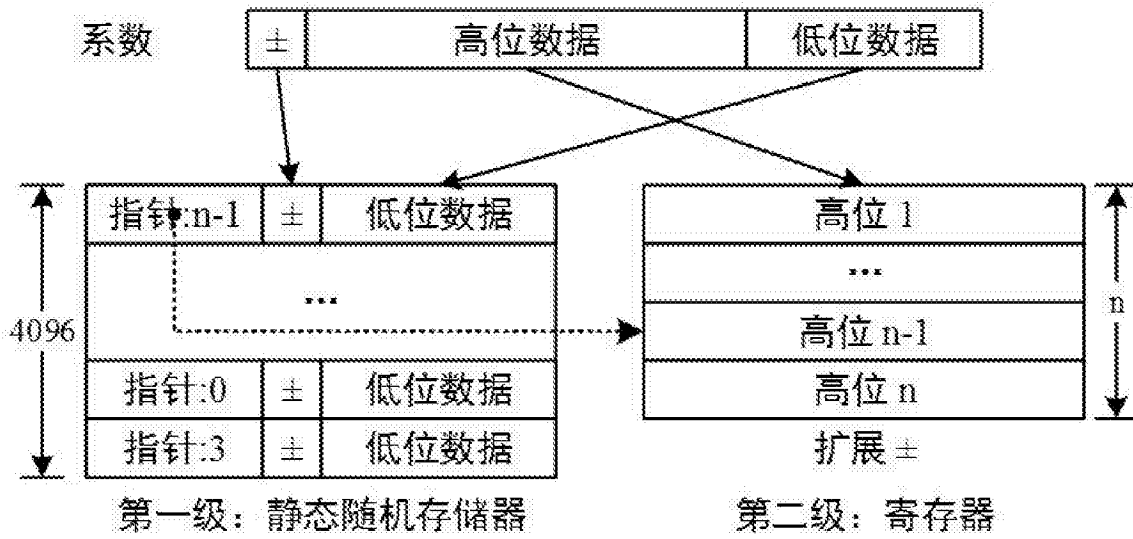


图3