



(12) 发明专利申请

(10) 申请公布号 CN 113841400 A

(43) 申请公布日 2021. 12. 24

(21) 申请号 202080018276.6

(74) 专利代理机构 北京德琦知识产权代理有限公司 11018

(22) 申请日 2020.03.09

代理人 徐文静 陈世华

(30) 优先权数据

62/816,124 2019.03.09 US

16/812,000 2020.03.06 US

(51) Int.Cl.

H04N 19/12 (2006.01)

H04N 19/122 (2006.01)

H04N 19/176 (2006.01)

H04N 19/132 (2006.01)

H04N 19/44 (2006.01)

H04N 19/102 (2006.01)

H04N 19/88 (2006.01)

H04N 19/166 (2006.01)

H04N 19/00 (2006.01)

(85) PCT国际申请进入国家阶段日

2021.09.02

(86) PCT国际申请的申请数据

PCT/US2020/021659 2020.03.09

(87) PCT国际申请的公布数据

W02020/185656 EN 2020.09.17

(71) 申请人 腾讯美国有限责任公司

地址 美国加利福尼亚州帕洛阿尔托公园大道2747号

(72) 发明人 赵欣 李翔 刘杉

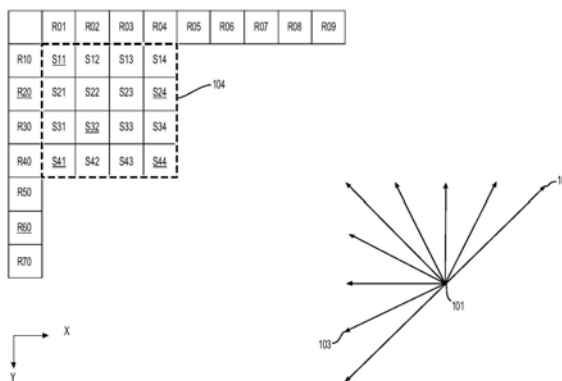
权利要求书4页 说明书41页 附图17页

(54) 发明名称

视频编解码的方法及装置

(57) 摘要

本公开的各方面提供视频编码/解码的方法和装置。在一些示例中,视频解码的装置包括处理电路。处理电路可以对的系数块的编码信息进行解码,所述系数块来自自己编码视频码流。编码信息可以指示系数块的大小。处理电路可以基于系数块的大小,确定要对系数块的变换系数执行主逆变换中的水平逆变换和垂直逆变换的顺序,以获得残差块的残差数据。当系数块的大小满足条件时,在对系数块的变换系数执行水平逆变换之后执行垂直逆变换。处理电路可以基于残差数据来重建残差块中的样本。



1. 一种在解码器中进行视频解码的方法,其特征在于,包括:

对系数块的编码信息进行解码,所述系数块来自自己编码视频码流,所述编码信息指示所述系数块的大小;

基于所述系数块的所述大小,确定要对所述系数块的变换系数执行主逆变换中的水平逆变换和垂直逆变换的顺序,以获得残差块的残差数据,其中,

当所述系数块的所述大小满足条件时,在对所述系数块的所述变换系数执行所述水平逆变换之后执行所述垂直逆变换;以及

基于所述残差数据来重建所述残差块中的样本。

2. 根据权利要求1所述的方法,其特征在于,所述条件包括所述系数块的所述大小为 32×64 。

3. 根据权利要求1所述的方法,其特征在于,所述条件包括 $M \times N$ 的所述系数块的高度 N 大于所述系数块的宽度 M 。

4. 根据权利要求1所述的方法,其特征在于:

所述系数块的所述大小为 $M \times N$, M 和 N 是正整数;

所述残差块中 $m \times n$ 区域内的第一残差数据将通过所述主逆变换进行计算,并且所述残差块中所述 $m \times n$ 区域外的第二残差数据将不通过所述主逆变换进行计算, m 小于或等于 M ,并且 n 小于或等于 N ;并且

所述条件包括比率 m/M 大于或等于比率 n/N 。

5. 一种在解码器中进行视频解码的方法,其特征在于,包括:

对系数块的编码信息进行解码,所述系数块来自自己编码视频码流,所述编码信息指示所述系数块的大小;

基于所述系数块的所述大小,确定是否减少主逆变换中的水平逆变换和垂直逆变换其中之一的计算次数,所述垂直逆变换将所述系数块的变换系数变换成中间块的中间数据,所述水平逆变换将所述中间数据变换成残差块中的残差数据;

执行所述主逆变换,包括:

当确定减少所述垂直逆变换中的所述计算次数时,通过所述垂直逆变换计算所述中间块中的顶部16行的所述中间数据,且所述中间块中的剩余中间数据为零;并且

当确定减少所述水平逆变换中的所述计算次数时,通过所述水平逆变换计算所述残差块中的左侧16列的所述残差数据,且所述残差块中的剩余残差数据为零;以及

基于所述残差数据来重建所述残差块中的样本。

6. 根据权利要求5所述的方法,其特征在于:

所述系数块的所述大小为 32×64 ;

所述水平逆变换和垂直逆变换之一是所述垂直逆变换;

所述确定包括:当所述系数块的所述大小为 32×64 时,确定减少所述垂直逆变换中的所述计算次数;并且

所述执行所述主逆变换包括执行所述主逆变换,通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据,并且所述中间块中的所述剩余中间数据为零。

7. 根据权利要求5所述的方法,其特征在于:

所述系数块的所述大小为 32×64 ;

所述水平逆变换和垂直逆变换之一是所述水平逆变换；

所述确定包括：当所述系数块的所述大小为 32×64 时，确定减少所述水平逆变换中的所述计算次数；并且

所述执行所述主逆变换包括执行所述主逆变换，通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据，并且所述残差块中的所述剩余残差数据为零。

8. 根据权利要求5所述的方法，其特征在于：

所述系数块的所述大小为 32×32 ；

所述水平逆变换和垂直逆变换之一是所述水平逆变换；

所述确定包括：当所述系数块的所述大小为 32×32 时，确定减少所述水平逆变换中的所述计算次数；并且

所述执行所述主逆变换包括执行所述主逆变换，通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据，所述残差块中的所述剩余残差数据为零，并且通过所述垂直逆变换计算所述中间块中的所述中间数据。

9. 根据权利要求5所述的方法，其特征在于：

所述系数块的所述大小为 32×32 ；

所述水平逆变换和垂直逆变换之一是所述垂直逆变换；

所述确定包括：当所述系数块的所述大小为 32×32 时，确定减少所述垂直逆变换中的所述计算次数；并且

所述执行所述主逆变换包括执行所述主逆变换，通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据，所述中间块中的所述剩余中间数据为零，并且通过所述水平逆变换计算所述残差块中的所述残差数据。

10. 一种视频解码的装置，其特征在于，包括处理电路，所述处理电路被配置为：

对系数块的编码信息进行解码，所述系数块来自已编码视频码流，所述编码信息指示所述系数块的大小；

基于所述系数块的所述大小，确定要对所述系数块的变换系数执行主逆变换中的水平逆变换和垂直逆变换的顺序，以获得残差块的残差数据，其中

当所述系数块的所述大小满足条件时，在对所述系数块的所述变换系数执行所述水平逆变换之后执行所述垂直逆变换；并且

基于所述残差数据来重建所述残差块中的样本。

11. 根据权利要求10所述的装置，其特征在于，所述条件包括所述系数块的所述大小为 32×64 。

12. 根据权利要求10所述的装置，其特征在于，所述条件包括 $M \times N$ 的所述系数块的高度 N 大于所述系数块的宽度 M 。

13. 根据权利要求10所述的装置，其特征在于：

所述系数块的所述大小为 $M \times N$ ， M 和 N 是正整数；

所述残差块中 $m \times n$ 区域内的第一残差数据将通过所述主逆变换进行计算，并且所述残差块中所述 $m \times n$ 区域外的第二残差数据将不通过所述主逆变换计算， m 小于或等于 M ，并且 n 小于或等于 N ；并且

所述条件包括比率 m/M 大于或等于比率 n/N 。

14. 一种视频解码的装置,其特征在於,包括处理电路,所述处理电路被配置为:

对系数块的编码信息进行解码,所述系数块来自已编码视频码流,所述编码信息指示所述系数块的大小;

基于所述系数块的所述大小,确定是否减少主逆变换中的水平逆变换和垂直逆变换其中之一之计算次数,所述垂直逆变换将所述系数块的变换系数变换成中间块的中间数据,所述水平逆变换将所述中间数据变换成残差块中的残差数据;

执行所述主逆变换,其中:

当确定要减少所述垂直逆变换中的所述计算次数时,通过所述垂直逆变换计算所述中间块中的顶部16行的所述中间数据,且所述中间块中的剩余中间数据为零;并且

当确定要减少所述水平逆变换中的所述计算次数时,通过所述水平逆变换计算所述残差块中的左侧16列的所述残差数据,且所述残差块中的剩余残差数据为零;并且

基于所述残差数据来重建所述残差块中的样本。

15. 根据权利要求14所述的装置,其特征在於:

所述系数块的所述大小为 32×64 ;

所述水平逆变换和垂直逆变换之一是所述垂直逆变换;并且

所述处理电路进一步被配置为:

当所述系数块的所述大小为 32×64 时,确定要减少所述垂直逆变换中的所述计算次数;并且

执行所述主逆变换,其中

通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据,并且所述中间块中的所述剩余中间数据为零。

16. 根据权利要求14所述的装置,其特征在於:

所述系数块的所述大小为 32×64 ;

所述水平逆变换和垂直逆变换之一是所述水平逆变换;并且

所述处理电路进一步被配置为:

当所述系数块的所述大小为 32×64 时,确定要减少所述水平逆变换中的所述计算次数;并且

执行所述主逆变换,其中,

通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据,并且所述残差块中的所述剩余残差数据为零。

17. 根据权利要求14所述的装置,其特征在於:

所述系数块的所述大小为 32×32 ;

所述水平逆变换和垂直逆变换之一是所述水平逆变换;并且

所述处理电路进一步被配置为:

当所述系数块的所述大小为 32×32 时,确定要减少所述水平逆变换中的所述计算次数;并且

执行所述主逆变换,其中,

通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据,所述残差块中的所述剩余残差数据为零,并且通过所述垂直逆变换来计算所述中间块中的所述中间数据。

18. 根据权利要求14所述的装置,其特征在于:

所述系数块的所述大小为 32×32 ;

所述水平逆变换和垂直逆变换之一是所述垂直逆变换;并且

所述处理电路进一步被配置为:

当所述系数块的所述大小为 32×32 时,确定要减少所述垂直逆变换中的所述计算次数;并且

执行所述主逆变换,其中

通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据,所述中间块中的所述剩余中间数据为零,并且通过所述水平逆变换来计算所述残差块中的所述残差数据。

视频编解码的方法及装置

[0001] 交叉引用

[0002] 本申请要求于2020年3月6日提交的美国专利申请第16/812,000号,“视频编解码的方法及装置(Method and Apparatus for Video Coding)”的优先权,该申请要求于2019年3月9日提交的美国临时申请第62/816,124号,“自适应变换系数零输出(Adaptive Transform Coefficient Zero-Out)”的优先权。这些在先申请的全部内容通过引用结合到本文中。

技术领域

[0003] 本申请描述总体上涉及视频编解码的实施例。

背景技术

[0004] 本文所提供的背景描述旨在整体呈现本申请的背景。在背景技术部分以及本说明书的各个方面中所描述的目前已署名的发明人的工作所进行的程度,并不表明其在本申请提交时作为现有技术,且从未明示或暗示其被承认为本申请的现有技术。

[0005] 通过具有运动补偿的帧间图片预测技术,可以进行视频编码和解码。未压缩的数字视频可包括一系列图片,每个图片具有例如 1920×1080 亮度样本及相关色度样本的空间维度。所述系列图片具有固定的或可变的图片速率(也非正式地称为帧率),例如每秒60个图片或60Hz。未压缩的视频具有非常大的比特率要求。例如,每个样本8比特的1080p60 4:2:0的视频(1920×1080 亮度样本分辨率,60Hz帧率)要求接近1.5Gbit/s带宽。一小时这样的视频就需要超过600GB的存储空间。

[0006] 视频编码和解码的一个目的,是通过压缩减少输入视频信号的冗余信息。视频压缩可以帮助降低对上述带宽或存储空间的要求,在某些情况下可降低两个或更多数量级。无损和有损压缩,以及两者的组合均可采用。无损压缩是指从压缩的原始信号中重建原始信号精确副本的技术。当使用有损压缩时,重建信号可能与原始信号不完全相同,但是原始信号和重建信号之间的失真足够小,使得重建信号可用于预期应用。有损压缩广泛应用于视频。容许的失真量取决于应用。例如,相比于电视应用的用户,某些消费流媒体应用的用户可以容忍更高的失真。可实现的压缩比反映出:较高的允许/容许失真可产生较高的压缩比。

[0007] 视频编码器和解码器可利用几大类技术,例如包括:运动补偿、变换、量化和熵编码。

[0008] 视频编解码器技术可包括已知的帧内编码技术。在帧内编码中,在不参考先前重建的参考图片的样本或其它数据的情况下表示样本值。在一些视频编解码器中,图片在空间上被细分为样本块。当所有的样本块都以帧内模式编码时,该图片可以为帧内图片。帧内图片及其衍生(例如独立解码器刷新图片)可用于复位解码器状态,并且因此可用作编码视频比特流和视频会话中的第一图片,或用作静止图像。帧内块的样本可用于变换,且可在熵编码之前量化变换系数。帧内预测可以是使预变换域中的样本值最小化的技术。在某些情

形下,变换后的DC值越小,且AC系数越小,则在给定的量化步长尺寸下需要越少的比特来表示熵编码之后的块。

[0009] 如同从诸如MPEG-2代编码技术中所获知的,传统帧内编码不使用帧内预测。然而,一些较新的视频压缩技术包括:试图从例如周围样本数据和/或元数据中得到数据块的技术,其中周围样本数据和/或元数据是在空间相邻的编码/解码期间、且在解码顺序之前获得的。这种技术后来被称为“帧内预测”技术。需要注意的是,至少在某些情形下,帧内预测仅使用正在重建的当前图片的参考数据,而不使用参考图片的参考数据。

[0010] 可以存在许多不同形式的帧内预测。当在给定的视频编码技术中可以使用超过一种这样的技术时,所使用的技术可以按帧内预测模式进行编码。在某些情形下,模式可具有子模式和/或参数,且这些模式可单独编码或包含在模式码字中。将哪个码字用于给定模式/子模式/参数组合会通过帧内预测影响编码效率增益,因此用于将码字转换成比特流的熵编码技术也会出现这种情况。

[0011] H.264引入了一种帧内预测模式,其在H.265中进行了改进,且在诸如被称为联合探索模型(JEM,joint exploration model)/下一代视频编码(VVC,versatile video coding)/基准集合(BMS,benchmark set)的更新的编码技术中进一步被改进。通过使用属于已经可用的样本的相邻样本值可以形成预测块。在一些示例中,将相邻样本的样本值按照某一方向复制到预测块中。对所使用方向的引用可以被编码在比特流中,或者本身可以被预测。

[0012] 参照图1A,右下方描绘了来自H.265的33个可能的预测方向(对应于35种帧内模式的33个角度模式)中已知的九个预测方向的子集合。箭头会聚的点(101)表示正在被预测的样本。箭头表示样本正在被预测的方向。例如,箭头(102)表示根据右上方与水平方向成45度角的一个或多个样本,预测样本(101)。类似地,箭头(103)表示根据左下方与水平方向成22.5度角的一个或多个样本,预测样本(101)。

[0013] 仍然参考图1A,在左上方示出了一个包括 4×4 个样本的正方形块(104)(由粗虚线表示)。正方形块(104)包括16个样本,每个样本用“S”、以及其在Y维度上的位置(例如,行索引)和在X纬度上的位置(例如,列索引)来标记。例如,样本S21是Y维度上的第二个样本(从顶部开始)和X维度上的第一个(从左侧开始)样本。类似地,样本S44在X维度和Y维度上都是块(104)中的第四个样本。由于该块为 4×4 大小的样本,因此S44位于右下角。还示出了遵循类似编号方案的参考样本。参考样本用“R”、以及其相对于块(104)的Y位置(例如,行索引)和X位置(例如,列索引)来标记。在H.264与H.265中,预测样本与正在重建的块相邻,因此不需要使用负值。

[0014] 通过从信号通知的预测方向所占用的相邻样本来复制参考样本值,可以进行帧内图片预测。例如,假设编码视频比特流包括信令,对于该块,该信令指示与箭头(102)一致的预测方向,即,根据右上方与水平方向成45度角的一个或多个预测样本来预测样本。在这种情况下,根据同一参考样本R05,预测样本S41、S32、S23和S14。根据参考样本R08,预测样本S44。

[0015] 在某些情况下,例如通过内插,可以合并多个参考样本的值,以便计算参考样本,尤其是当方向不能被45度整除时。

[0016] 随着视频编码技术的发展,可能的方向的数量已经增加了。在H.264(2003年)中,

可以表示九种不同的方向。在H.265(2013年)和JEM/VVC/BMS中增加到了33个,而在此申请时,可以支持多达65个方向。已经进行了实验来识别最可能的方向,并且熵编码中的某些技术被用于使用少量比特来表示那些可能的方向,对于较不可能的方向则接受某些代价。此外,有时可以根据在相邻的、已经解码的块中所使用的相邻方向来预测方向本身。

[0017] 图1B示出了用于描绘根据JEM的65个帧内预测方向的示意图(180),以示出随着时间推移预测方向的增加数量。

[0018] 表示方向的编码视频比特流中的帧内预测方向比特的映射可以因视频编码技术的不同而不同,并且,例如可以从对帧内预测模式到码字的预测方向的简单直接映射,到包括最可能的模式和类似技术的复杂的自适应方案。然而,在所有情况下,视频内容中可能存在某些方向,其在统计学上比其它方向更不可能出现。由于视频压缩的目的是减少冗余,所以在运行良好的视频编码技术中,与更可能的方向相比,那些不太可能的方向将使用更多数量的比特来表示。

[0019] 通过具有运动补偿的帧间图片预测技术,可以进行视频编码和解码。运动补偿可以是一种有损压缩技术,且可涉及如下技术:来自先前重建的图片或重建图片一部分(参考图片)的样本数据块在空间上按运动矢量(下文称为MV)指示的方向移位后,用于新重建的图片或图片部分的预测。在某些情况下,参考图片可与当前正在重建的图片相同。MV可具有两个维度X和Y,或者三个维度,其中第三个维度表示使用中的参考图片(后者间接地可为时间维度)。

[0020] 在一些视频压缩技术中,应用于某个样本数据区域的MV可根据其它MV来预测,例如根据与正在重建的区域空间相邻的另一个样本数据区域相关的、且按解码顺序在该MV前面的那些MV。这样做可以大大减少编码MV所需的数据量,从而消除冗余信息并增加压缩量。MV预测可以有效地进行,例如,当对从相机导出的输入视频信号(称为自然视频)进行编码时,存在一种统计上的可能性,即面积大于单个MV适用区域的区域,会朝着类似的方向移动,因此,在某些情况下,可以用邻近区域的MV导出的相似运动矢量进行预测。这导致针对给定区域发现的MV与根据周围MV预测的MV相似或相同,并且在熵编码之后,又可以用比直接编码MV时使用的比特数更少的比特数来表示。在某些情况下,MV预测可以是对从原始信号(即样本流)导出的信号(即MV)进行无损压缩的示例。在其它情况下,MV预测本身可能是有损的,例如由于根据几个周围MV计算预测值时产生的取整误差。

[0021] H.265/HEVC(ITU-T Rec.H.265,“高效视频编码”,2016年12月)描述了各种MV预测机制。在H.265所提供的多种MV预测机制中,本文描述的是一种下文称为“空间合并”的技术。

[0022] 参考图2,当前块(201)包括编码器在运动搜索过程中发现的样本,所述样本可以根据空间移动了相同大小的先前块进行预测。不直接对该MV进行编码,而是通过使用与五个周围样本中的任何一个相关联的MV,从与一个或多个参考图片相关联的元数据中导出该MV,例如从最近的(按解码顺序)参考图片中导出该MV。其中,五个周围样本分别用A0、A1和B0、B1、B2(202到206)表示。在H.265中,MV预测可使用相邻块正在使用的同一参考图片的预测值。

发明内容

[0023] 本公开的各方面提供视频编码/解码的方法和装置。在一些示例中,视频解码的装置包括处理电路。处理电路可以对系数块的编码信息进行解码,所述系数块来自自己编码视频码流,所述编码信息指示所述系数块的大小。处理电路可以基于所述系数块的所述大小,确定要对所述系数块的变换系数执行主逆变换中的水平逆变换和垂直逆变换的顺序,以获得残差块的残差数据。当所述系数块的所述大小满足条件时,在对所述系数块的所述变换系数执行所述水平逆变换之后执行所述垂直逆变换。处理电路可以基于残差数据来重建残差块中的样本。在示例中,条件是系数块的大小为 32×64 。在示例中,条件是 $M \times N$ 的系数块的高度 N 大于系数块的宽度 M 。在示例中,所述系数块的所述大小为 $M \times N$, M 和 N 是正整数;所述残差块中 $m \times n$ 区域内的第一残差数据将通过所述主逆变换进行计算,并且所述残差块中所述 $m \times n$ 区域外的第二残差数据将不通过所述主逆变换计算, m 小于或等于 M ,并且 n 小于或等于 N ;并且所述条件是比率 m/M 大于或等于比率 n/N 。

[0024] 本公开的各方面提供视频编码/解码的方法和装置。在一些示例中,视频解码的装置包括处理电路。处理电路对系数块的编码信息进行解码,所述系数块来自自己编码视频码流,所述编码信息指示所述系数块的大小。处理电路基于所述系数块的所述大小,确定是否减少主逆变换中的水平逆变换和垂直逆变换其中之一的计算次数,所述垂直逆变换将所述系数块的变换系数变换成中间块的中间数据,所述水平逆变换将所述中间数据变换成残差块中的残差数据。处理电路可以执行主逆变换。当确定要减少所述垂直逆变换中的所述计算次数时,通过所述垂直逆变换计算所述中间块中的顶部16行的所述中间数据,并且所述中间块中的剩余中间数据为零;并且当确定要减少所述水平逆变换中的所述计算次数时,通过所述水平逆变换计算所述残差块中的左侧16列的所述残差数据,并且所述残差块中的剩余残差数据为零。处理电路可以基于残差数据来重建残差块中的样本。

[0025] 在实施例中,系数块的大小为 32×64 。水平逆变换和垂直逆变换之一是垂直逆变换。当系数块的大小为 32×64 时,处理电路可以确定要减少所述垂直逆变换中的所述计算次数。处理电路可以执行主逆变换,其中可以通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据,并且所述中间块中的所述剩余中间数据为零。

[0026] 在实施例中,系数块的大小为 32×64 。水平逆变换和垂直逆变换之一是水平逆变换。当所述系数块的所述大小为 32×64 时,确定要减少所述水平逆变换中的所述计算次数;并且执行所述主逆变换,其中,通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据,并且所述残差块中的所述剩余残差数据为零。

[0027] 在实施例中,系数块的大小为 32×32 。水平逆变换和垂直逆变换之一是水平逆变换。当所述系数块的所述大小为 32×32 时,确定要减少所述水平逆变换中的所述计算次数;并且执行所述主逆变换,其中,通过所述水平逆变换计算所述残差块中的所述左侧16列残差数据,所述残差块中的所述剩余残差数据为零,并且通过所述垂直逆变换来计算所述中间块中的所述中间数据。

[0028] 在实施例中,系数块的大小为 32×32 。水平逆变换和垂直逆变换之一是垂直逆变换。当所述系数块的所述大小为 32×32 时,确定要减少所述垂直逆变换中的所述计算次数;并且执行所述主逆变换,其中通过所述垂直逆变换计算所述中间块中的所述顶部16行的中间数据,所述中间块中的所述剩余中间数据为零,并且通过所述水平逆变换来计算所述残

差块中的所述残差数据。

附图说明

[0029] 通过以下详细描述和附图,所公开的主题的其它特征、性质及各种优点将更加明显,其中:

- [0030] 图1A是帧内预测模式的示例性子集的示意图。
- [0031] 图1B是示例性帧内预测方向的图示。
- [0032] 图2是一个示例中的当前块及其周围的空间合并候选的示意图。
- [0033] 图3是根据实施例的通信系统的简化框图的示意图。
- [0034] 图4是根据实施例的通信系统的简化框图的示意图。
- [0035] 图5是根据实施例的解码器的简化框图的示意图。
- [0036] 图6是根据实施例的编码器的简化框图的示意图。
- [0037] 图7示出了根据另一实施例的编码器的框图。
- [0038] 图8示出了根据另一实施例的解码器的框图。
- [0039] 图9示出了根据实施例的变换单元语法的示例。
- [0040] 图10A至图10C示出了根据实施例的残差编解码语法的示例。
- [0041] 图11A至图11B示出了根据实施例的主变换的示例。
- [0042] 图12A至图12E示出了根据实施例的示例性变换过程。
- [0043] 图13示出了根据另一实施例的残差编解码语法的示例。
- [0044] 图14示出了根据实施例的概述方法的流程图。
- [0045] 图15示出了根据实施例的概述方法的流程图。
- [0046] 图16是根据实施例的计算机系统的示意图。

具体实施方式

[0047] 图3是根据本申请公开的实施例的通信系统(300)的简化框图。通信系统(300)包括多个终端装置,所述终端装置可通过例如网络(350)彼此通信。举例来说,通信系统(300)包括通过网络(350)互连的第一对终端装置(310)和(320)。在图3的实施例中,第一对终端装置(310)和(320)执行单向数据传输。举例来说,终端装置(310)可对视频数据(例如由终端装置(310)采集的视频图片流)进行编码以通过网络(350)传输到另一终端装置(320)。已编码的视频数据以一个或多个已编码视频码流形式传输。终端装置(320)可从网络(350)接收已编码视频数据,对已编码视频数据进行解码以恢复视频数据,并根据恢复的视频数据显示视频图片。单向数据传输在媒体服务等应用中是较常见的。

[0048] 在另一实施例中,通信系统(300)包括执行已编码视频数据的双向传输的第二对终端装置(330)和(340),所述双向传输可例如在视频会议期间发生。对于双向数据传输,终端装置(330)和(340)中的每个终端装置可对视频数据(例如由终端装置采集的视频图片流)进行编码,以通过网络(350)传输到终端装置(330)和(340)中的另一终端装置。终端装置(330)和(340)中的每个终端装置还可接收由终端装置(330)和终端装置(340)中的另一终端装置传输的已编码视频数据,且可对所述已编码视频数据进行解码以恢复视频数据,且可根据恢复的视频数据在可访问的显示装置上显示视频图片。

[0049] 在图2的实施例中,终端装置(310)、(320)、(330)和(340)可为服务器、个人计算机和智能电话,但本申请公开的原理可不限于此。本申请公开的实施例适用于膝上型计算机、平板电脑、媒体播放器和/或专用视频会议设备。网络(350)表示在终端装置(310)、(320)、(330)和(340)之间传送已编码视频数据的任何数目的网络,包括例如有线(连线的)和/或无线通信网络。通信网络(350)可在电路交换和/或分组交换信道中交换数据。该网络可包括电信网络、局域网、广域网和/或互联网。出于本申请的目的,除非在下文中有所解释,否则网络(350)的架构和拓扑对于本申请公开的操作来说可能是无关紧要的。

[0050] 作为实施例,图4示出视频编码器和视频解码器在流式传输环境中的放置方式。本申请所公开主题可同等地适用于其它支持视频的应用,包括例如视频会议、数字TV、在包括CD、DVD、存储棒等的数字介质上存储压缩视频等等。

[0051] 流式传输系统可包括采集子系统(413),所述采集子系统可包括数码相机等视频源(401),所述视频源创建未压缩的视频图片流(402)。在实施例中,视频图片流(402)包括由数码相机拍摄的样本。相较于已编码的视频数据(404)(或已编码的视频码流),视频图片流(402)被描绘为粗线以强调高数据量的视频图片流,视频图片流(402)可由电子装置(420)处理,所述电子装置(420)包括耦接到视频源(401)的视频编码器(403)。视频编码器(403)可包括硬件、软件或软硬件组合以实现或实施如下文更详细地描述的所公开主题的各方面。相较于视频图片流(402),已编码的视频数据(404)(或已编码的视频码流(404))被描绘为细线以强调较低数据量的已编码的视频数据(404)(或已编码的视频码流(404)),其可存储在流式传输服务器(405)上以供将来使用。一个或多个流式传输客户端子系统,例如图4中的客户端子系统(406)和客户端子系统(408),可访问流式传输服务器(405)以检索已编码的视频数据(404)的副本(407)和副本(409)。客户端子系统(406)可包括例如电子装置(430)中的视频解码器(410)。视频解码器(410)对已编码的视频数据的传入副本(407)进行解码,且产生可在显示器(412)(例如显示屏)或另一呈现装置(未描绘)上呈现的输出视频图片流(411)。在一些流式传输系统中,可根据某些视频编码/压缩标准对已编码的视频数据(404)、视频数据(407)和视频数据(409)(例如视频码流)进行编码。这些标准的实施例包括ITU-T H.265。在实施例中,正在开发的视频编码标准非正式地称为下一代视频编码(Versatile Video Coding,VVC),本申请可用于VVC标准的上下文中。

[0052] 应注意,电子装置(420)和电子装置(430)可包括其它组件(未示出)。举例来说,电子装置(420)可包括视频解码器(未示出),且电子装置(430)还可包括视频编码器(未示出)。

[0053] 图5是根据本申请公开的实施例的视频解码器(510)的框图。视频解码器(510)可设置在电子装置(530)中。电子装置(530)可包括接收器(531)(例如接收电路)。视频解码器(510)可用于代替图4实施例中的视频解码器(410)。

[0054] 接收器(531)可接收将由视频解码器(510)解码的一个或多个已编码视频序列;在同一实施例或另一实施例中,一次接收一个已编码视频序列,其中每个已编码视频序列的解码独立于其它已编码视频序列。可从信道(501)接收已编码视频序列,所述信道可以是通向存储已编码的视频数据的存储装置的硬件/软件链路。接收器(531)可接收已编码的视频数据以及其它数据,例如,可转发到它们各自的使用实体(未标示)的已编码音频数据和/或辅助数据流。接收器(531)可将已编码视频序列与其它数据分开。为了防止网络抖动,缓冲

存储器(515)可耦接在接收器(531)与熵解码器/解析器(520)(此后称为“解析器(520)”)之间。在某些应用中,缓冲存储器(515)是视频解码器(510)的一部分。在其它情况下,所述缓冲存储器(515)可设置在视频解码器(510)外部(未标示)。而在其它情况下,视频解码器(510)的外部设置缓冲存储器(未标示)以例如防止网络抖动,且在视频解码器(510)的内部可配置另一缓冲存储器(515)以例如处理播出定时。而当接收器(531)从具有足够带宽和可控性的存储/转发装置或从等时同步网络接收数据时,也可能不需要配置缓冲存储器(515),或可以将所述缓冲存储器做得较小。当然,为了在互联网等业务分组网络上使用,也可能需要缓冲存储器(515),所述缓冲存储器可相对较大且可具有自适应性大小,且可至少部分地实施于操作系统或视频解码器(510)外部的类似元件(未标示)中。

[0055] 视频解码器(510)可包括解析器(520)以根据已编码视频序列重建符号(521)。这些符号的类别包括用于管理视频解码器(510)的操作的信息,以及用以控制显示装置(512)(例如,显示屏)等显示装置的潜在信息,所述显示装置不是电子装置(530)的组成部分,但可耦接到电子装置(530),如图5中所示。用于显示装置的控制信息可以是辅助增强信息(Supplemental Enhancement Information,SEI消息)或视频可用性信息(Video Usability Information,VUI)的参数集片段(未标示)。解析器(520)可对接收到的已编码视频序列进行解析/熵解码。已编码视频序列的编码可根据视频编码技术或标准进行,且可遵循各种原理,包括可变长度编码、霍夫曼编码(Huffman coding)、具有或不具有上下文灵敏度的算术编码等等。解析器(520)可基于对应于群组的至少一个参数,从已编码视频序列提取用于视频解码器中的像素的子群中的至少一个子群的子群参数集。子群可包括图片群组(Group of Pictures,GOP)、图片、图块、切片、宏块、编码单元(Coding Unit,CU)、块、变换单元(Transform Unit,TU)、预测单元(Prediction Unit,PU)等等。解析器(520)还可从已编码视频序列提取信息,例如变换系数、量化器参数值、运动矢量等等。

[0056] 解析器(520)可对从缓冲存储器(515)接收的视频序列执行熵解码/解析操作,从而创建符号(521)。

[0057] 取决于已编码视频图片或一部分已编码视频图片(例如:帧间图片和帧内图片、帧间块和帧内块)的类型以及其它因素,符号(521)的重建可涉及多个不同单元。涉及哪些单元以及涉及方式可由解析器(520)从已编码视频序列解析的子群控制信息控制。为了简洁起见,未描述解析器(520)与下文的多个单元之间的此类子群控制信息流。

[0058] 除已经提及的功能块以外,视频解码器(510)可在概念上细分成如下文所描述的数个功能单元。在商业约束下运行的实际实施例中,这些单元中的许多单元彼此紧密交互并且可以彼此集成。然而,出于描述所公开主题的目的,概念上细分成下文的功能单元是适当的。

[0059] 第一单元是缩放器/逆变换单元(551)。缩放器/逆变换单元(551)从解析器(520)接收作为符号(521)的量化变换系数以及控制信息,包括使用哪种变换方式、块大小、量化因子、量化缩放矩阵等。缩放器/逆变换单元(551)可输出包括样本值的块,所述样本值可输入到聚合器(555)中。

[0060] 在一些情况下,缩放器/逆变换单元(551)的输出样本可属于帧内编码块;即:不使用来自自先前重建的图片的预测性信息,但可使用来自当前图片的先前重建部分的预测性信息的块。此类预测性信息可由帧内图片预测单元(552)提供。在一些情况下,帧内图片预测

单元 (552) 采用从当前图片缓冲器 (558) 提取的已重建信息生成大小和形状与正在重建的块相同的周围块。举例来说, 当前图片缓冲器 (558) 缓冲部分重建的当前图片和/或完全重建的当前图片。在一些情况下, 聚合器 (555) 基于每个样本, 将帧内预测单元 (552) 生成的预测信息添加到由缩放器/逆变换单元 (551) 提供的输出样本信息中。

[0061] 在其它情况下, 缩放器/逆变换单元 (551) 的输出样本可属于帧间编码和潜在运动补偿块。在此情况下, 运动补偿预测单元 (553) 可访问参考图片存储器 (557) 以提取用于预测的样本。在根据符号 (521) 对提取的样本进行运动补偿之后, 这些样本可由聚合器 (555) 添加到缩放器/逆变换单元 (551) 的输出 (在这种情况下被称作残差样本或残差信号), 从而生成输出样本信息。运动补偿预测单元 (553) 从参考图片存储器 (557) 内的地址获取预测样本可受到运动矢量控制, 且所述运动矢量以所述符号 (521) 的形式而供运动补偿预测单元 (553) 使用, 所述符号 (521) 例如是包括 X、Y 和参考图片分量。运动补偿还可包括在使用子样本精确运动矢量时, 从参考图片存储器 (557) 提取的样本值的内插、运动矢量预测机制等等。

[0062] 聚合器 (555) 的输出样本可在环路滤波器单元 (556) 中被各种环路滤波技术采用。视频压缩技术可包括环路内滤波器技术, 所述环路内滤波器技术受控于包括在已编码视频序列 (也称作已编码视频码流) 中的参数, 且所述参数作为来自解析器 (520) 的符号 (521) 可用于环路滤波器单元 (556)。然而, 在其他实施例中, 视频压缩技术还可响应于在解码已编码图片或已编码视频序列的先前 (按解码次序) 部分期间获得的元信息, 以及响应于先前重建且经过环路滤波的样本值。

[0063] 环路滤波器单元 (556) 的输出可以是样本流, 所述样本流可输出到显示装置 (512) 以及存储在参考图片存储器 (557), 以用于后续的帧间图片预测。

[0064] 一旦完全重建, 某些已编码图片就可用作参考图片以用于将来预测。举例来说, 一旦对应于当前图片的已编码图片被完全重建, 且已编码图片 (通过例如解析器 (520)) 被识别为参考图片, 则当前图片缓冲器 (558) 可变为参考图片存储器 (557) 的一部分, 且可在开始重建后续已编码图片之前重新分配新的当前图片缓冲器。

[0065] 视频解码器 (510) 可根据例如 ITU-T H.265 标准中的预定视频压缩技术执行解码操作。在已编码视频序列遵循视频压缩技术或标准的语法以及视频压缩技术或标准中记录的配置文件的意义下, 已编码视频序列可符合所使用的视频压缩技术或标准指定的语法。具体地说, 配置文件可从视频压缩技术或标准中可用的所有工具中选择某些工具作为在所述配置文件下可供使用的仅有工具。对于合规性, 还要求已编码视频序列的复杂度处于视频压缩技术或标准的层级所限定的范围内。在一些情况下, 层级限制最大图片大小、最大帧率、最大重建取样率 (以例如每秒兆 (mega) 个样本为单位进行测量)、最大参考图片大小等。在一些情况下, 由层级设定的限制可通过假想参考解码器 (Hypothetical Reference Decoder, HRD) 规范和已在已编码视频序列中用信号表示的 HRD 缓冲器管理的元数据来进一步限定。

[0066] 在实施例中, 接收器 (531) 可连同已编码视频一起接收附加 (冗余) 数据。所述附加数据可以是已编码视频序列的一部分。所述附加数据可由视频解码器 (510) 用以对数据进行适当解码和/或较准确地重建原始视频数据。附加数据可呈例如时间、空间或信噪比 (signal noise ratio, SNR) 增强层、冗余切片、冗余图片、前向纠错码等形式。

[0067] 图6是根据本申请公开的实施例的视频编码器(603)的框图。视频编码器(603)设置于电子装置(620)中。电子装置(620)包括传输器(640)(例如传输电路)。视频编码器(603)可用于代替图4实施例中的视频编码器(403)。

[0068] 视频编码器(603)可从视频源(601)(并非图6实施例中的电子装置(620)的一部分)接收视频样本,所述视频源可采集将由视频编码器(603)编码的视频图像。在另一实施例中,视频源(601)是电子装置(620)的一部分。

[0069] 视频源(601)可提供将由视频编码器(603)编码的呈数字视频样本流形式的源视频序列,所述数字视频样本流可具有任何合适位深度(例如:8位、10位、12位……)、任何色彩空间(例如BT.601Y CrCb、RGB……)和任何合适取样结构(例如Y CrCb 4:2:0、Y CrCb 4:4:4)。在媒体服务系统中,视频源(601)可以是存储先前已准备的视频的存储装置。在视频会议系统中,视频源(601)可以是采集本地图像信息作为视频序列的相机。可将视频数据提供为多个单独的图片,当按顺序观看时,这些图片被赋予运动。图片自身可构建为空间像素阵列,其中取决于所用的取样结构、色彩空间等,每个像素可包括一个或多个样本。所属领域的技术人员可以很容易理解像素与样本之间的关系。下文侧重于描述样本。

[0070] 根据实施例,视频编码器(603)可实时或在由应用所要求的任何其它时间约束下,将源视频序列的图片编码且压缩成已编码视频序列(643)。施行适当的编码速度是控制器(650)的一个功能。在一些实施例中,控制器(650)控制如下文所描述的其它功能单元且在功能上耦接到这些单元。为了简洁起见,图中未标示耦接。由控制器(650)设置的参数可包括速率控制相关参数(图片跳过、量化器、率失真优化技术的 λ 值等)、图片大小、图片群组(group of pictures,GOP)布局,最大运动矢量搜索范围等。控制器(650)可用于具有其它合适的功能,这些功能涉及针对某一系统设计优化的视频编码器(603)。

[0071] 在一些实施例中,视频编码器(603)在编码环路中进行操作。作为简单的描述,在实施例中,编码环路可包括源编码器(630)(例如,负责基于待编码的输入图片和参考图片创建符号,例如符号流)和嵌入于视频编码器(603)中的(本地)解码器(633)。解码器(633)以类似于(远程)解码器创建样本数据的方式重建符号以创建样本数据(因为在本申请所考虑的视频压缩技术中,符号与已编码视频码流之间的任何压缩是无损的)。将重建的样本流(样本数据)输入到参考图片存储器(634)。由于符号流的解码产生与解码器位置(本地或远程)无关的位精确结果,因此参考图片存储器(634)中的内容在本地编码器与远程编码器之间也是按比特位精确对应的。换句话说,编码器的预测部分“看到”的参考图片样本与解码器将在解码期间使用预测时所“看到”的样本值完全相同。这种参考图片同步性基本原理(以及在例如因信道误差而无法维持同步性的情况下产生的漂移)也用于一些相关技术。

[0072] “本地”解码器(633)的操作可与例如已在上文结合图5详细描述视频解码器(510)的“远程”解码器相同。然而,另外简要参考图5,当符号可用且熵编码器(645)和解析器(520)能够无损地将符号编码/解码为已编码视频序列时,包括缓冲存储器(515)和解析器(520)在内的视频解码器(510)的熵解码部分,可能无法完全在本地解码器(633)中实施。

[0073] 此时可以观察到,除存在于解码器中的解析/熵解码之外的任何解码器技术,也必定以基本上相同的功能形式存在于对应的编码器中。出于此原因,本申请侧重于解码器操作。可简化编码器技术的描述,因为编码器技术与全面地描述的解码器技术互逆。仅在某些区域中需要更详细的描述,并且在下文提供。

[0074] 在操作期间,在一些实施例中,源编码器(630)可执行运动补偿预测编码。参考来自视频序列中被指定为“参考图片”的一个或多个先前已编码图片,所述运动补偿预测编码对输入图片进行预测性编码。以此方式,编码引擎(632)对输入图片的像素块与参考图片的像素块之间的差异进行编码,所述参考图片可被选作所述输入图片的预测参考。

[0075] 本地视频解码器(633)可基于源编码器(630)创建的符号,对可指定为参考图片的图片的已编码视频数据进行解码。编码引擎(632)的操作可为有损过程。当已编码视频数据可在视频解码器(图6中未示)处被解码时,重建的视频序列通常可以是带有一些误差的源视频序列的副本。本地视频解码器(633)复制解码过程,所述解码过程可由视频解码器对参考图片执行,且可使重建的参考图片存储在参考图片高速缓存(634)中。以此方式,视频编码器(603)可在本地存储重建的参考图片的副本,所述副本与将由远端视频解码器获得的重建参考图片具有共同内容(不存在传输误差)。

[0076] 预测器(635)可针对编码引擎(632)执行预测搜索。即,对于将要编码的新图片,预测器(635)可在参考图片存储器(634)中搜索可作为所述新图片的适当预测参考的样本数据(作为候选参考像素块)或某些元数据,例如参考图片运动矢量、块形状等。预测器(635)可基于样本块逐像素块操作,以找到合适的预测参考。在一些情况下,根据预测器(635)获得的搜索结果,可确定输入图片可具有从参考图片存储器(634)中存储的多个参考图片取得的预测参考。

[0077] 控制器(650)可管理源编码器(630)的编码操作,包括例如设置用于对视频数据进行编码的参数和子群参数。

[0078] 可在熵编码器(645)中对所有上述功能单元的输出进行熵编码。熵编码器(645)根据例如霍夫曼编码、可变长度编码、算术编码等技术对各种功能单元生成的符号进行无损压缩,从而将所述符号转换成已编码视频序列。

[0079] 传输器(640)可缓冲由熵编码器(645)创建的已编码视频序列,从而为通过通信信道(660)进行传输做准备,所述通信信道可以是通向将存储已编码的视频数据的存储装置的硬件/软件链路。传输器(640)可将来自视频编码器(603)的已编码视频数据与要传输的其它数据合并,所述其它数据例如是已编码音频数据和/或辅助数据流(未示出来源)。

[0080] 控制器(650)可管理视频编码器(603)的操作。在编码期间,控制器(650)可以为每个已编码图片分配某一已编码图片类型,但这可能影响可应用于相应的图片的编码技术。例如,通常可将图片分配为以下任一种图片类型:

[0081] 帧内图片(I图片),其可以是不将序列中的任何其它图片用作预测源就可被编码和解码的图片。一些视频编解码器容许不同类型的帧内图片,包括例如独立解码器刷新(Independent Decoder Refresh,“IDR”)图片。所属领域的技术人员了解I图片的变体及其相应的应用和特征。

[0082] 预测性图片(P图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,所述帧内预测或帧间预测使用至多一个运动矢量和参考索引来预测每个块的样本值。

[0083] 双向预测性图片(B图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,所述帧内预测或帧间预测使用至多两个运动矢量和参考索引来预测每个块的样本值。类似地,多个预测性图片可使用多于两个参考图片和相关联元数据以用于重建单个块。

[0084] 源图片通常可在空间上细分成多个样本块(例如,4×4、8×8、4×8或16×16个样

本的块),且逐块进行编码。这些块可参考其它(已编码)块进行预测编码,根据应用于块的相应图片的编码分配来确定所述其它块。举例来说,I图片的块可进行非预测编码,或所述块可参考同一图片的已经编码的块来进行预测编码(空间预测或帧内预测)。P图片的像素块可参考一个先前编码的参考图片通过空间预测或通过时域预测进行预测编码。B图片的块可参考一个或两个先前编码的参考图片通过空间预测或通过时域预测进行预测编码。

[0085] 视频编码器(603)可根据例如ITU-T H.265建议书的预定视频编码技术或标准执行编码操作。在操作中,视频编码器(603)可执行各种压缩操作,包括利用输入视频序列中的时间和空间冗余的预测编码操作。因此,已编码视频数据可符合所用视频编码技术或标准指定的语法。

[0086] 在实施例中,传输器(640)可在传输已编码的视频时传输附加数据。源编码器(630)可将此类数据作为已编码视频序列的一部分。附加数据可包括时间/空间/SNR增强层、冗余图片和切片等其它形式的冗余数据、SEI消息、VUI参数集片段等。

[0087] 采集到的视频可作为呈时间序列的多个源图片(视频图片)。帧内图片预测(常常简化为帧内预测)利用给定图片中的空间相关性,而帧间图片预测则利用图片之间的(时间或其它)相关性。在实施例中,将正在编码/解码的特定图片分割成块,正在编码/解码的特定图片被称作当前图片。在当前图片中的块类似于视频中先前已编码且仍被缓冲的参考图片中的参考块时,可通过称作运动矢量的矢量对当前图片中的块进行编码。所述运动矢量指向参考图片中的参考块,且在使用多个参考图片的情况下,所述运动矢量可具有识别参考图片的第三维度。

[0088] 在一些实施例中,双向预测技术可用于帧间图片预测中。根据双向预测技术,使用两个参考图片,例如按解码次序都在视频中的当前图片之前(但按显示次序可能分别是过去和将来)第一参考图片和第二参考图片。可通过指向第一参考图片中的第一参考块的第一运动矢量和指向第二参考图片中的第二参考块的第二运动矢量对当前图片中的块进行编码。具体来说,可通过第一参考块和第二参考块的组合来预测所述块。

[0089] 此外,合并模式技术可用于帧间图片预测中以改善编码效率。

[0090] 根据本申请公开的一些实施例,帧间图片预测和帧内图片预测等预测的执行以块为单位。举例来说,根据HEVC标准,将视频图片序列中的图片分割成编码树单元(coding tree unit,CTU)以用于压缩,图片中的CTU具有相同大小,例如 64×64 像素、 32×32 像素或 16×16 像素。一般来说,CTU包括三个编码树块(coding tree block,CTB),所述三个编码树块是一个亮度CTB和两个色度CTB。更进一步的,还可将每个CTU以四叉树拆分为一个或多个编码单元(coding unit,CU)。举例来说,可将 64×64 像素的CTU拆分为一个 64×64 像素的CU,或4个 32×32 像素的CU,或16个 16×16 像素的CU。在实施例中,分析每个CU以确定用于CU的预测类型,例如帧间预测类型或帧内预测类型。此外,取决于时间和/或空间可预测性,将CU拆分为一个或多个预测单元(prediction unit,PU)。通常,每个PU包括亮度预测块(prediction block,PB)和两个色度PB。在实施例中,编码(编码/解码)中的预测操作以预测块为单位来执行。以亮度预测块作为预测块为例,预测块包括像素值(例如,亮度值)的矩阵,例如 8×8 像素、 16×16 像素、 8×16 像素、 16×8 像素等等。

[0091] 图7是根据本申请公开的另一实施例的视频编码器(703)的图。视频编码器(703)用于接收视频图片序列中的当前视频图片内的样本值的处理块(例如预测块),且将所述处

理块编码到作为已编码视频序列的一部分的已编码图片中。在本实施例中，视频编码器(703)用于代替图4实施例中的视频编码器(403)。

[0092] 在HEVC实施例中，视频编码器(703)接收用于处理块的样本值的矩阵，所述处理块为例如 8×8 样本的预测块等。视频编码器(703)使用例如率失真(rate-distortion, RD)优化来确定是否使用帧内模式、帧间模式或双向预测模式来编码所述处理块。当在帧内模式中编码处理块时，视频编码器(703)可使用帧内预测技术以将处理块编码到已编码图片中；且当在帧间模式或双向预测模式中编码处理块时，视频编码器(703)可分别使用帧间预测或双向预测技术将处理块编码到已编码图片中。在某些视频编码技术中，合并模式可以是帧间图片预测子模式，其中，在不借助预测值外部的已编码运动矢量分量的情况下，从一个或多个运动矢量预测值导出运动矢量。在某些其它视频编码技术中，可存在适用于主题块的运动矢量分量。在实施例中，视频编码器(703)包括其它组件，例如用于确定处理块模式的模式决策模块(未示出)。

[0093] 在图7的实施例中，视频编码器(703)包括如图7所示的耦接到一起的帧间编码器(730)、帧内编码器(722)、残差计算器(723)、开关(726)、残差编码器(724)、通用控制器(721)和熵编码器(725)。

[0094] 帧间编码器(730)用于接收当前块(例如处理块)的样本、比较所述块与参考图片中的一个或多个参考块(例如先前图片和后来图片中的块)、生成帧间预测信息(例如根据帧间编码技术的冗余信息描述、运动矢量、合并模式信息)、以及基于帧间预测信息使用任何合适的技术计算帧间预测结果(例如已预测块)。在一些实施例中，参考图片是基于已编码的视频信息解码的已解码参考图片。

[0095] 帧内编码器(722)用于接收当前块(例如处理块)的样本、在一些情况下比较所述块与同一图片中已编码的块、在变换之后生成量化系数、以及在一些情况下还(例如根据一个或多个帧内编码技术的帧内预测方向信息)生成帧内预测信息。在实施例中，帧内编码器(722)还基于帧内预测信息和同一图片中的参考块计算帧内预测结果(例如已预测块)。

[0096] 通用控制器(721)用于确定通用控制数据，且基于所述通用控制数据控制视频编码器(703)的其它组件。在实施例中，通用控制器(721)确定块的模式，且基于所述模式将控制信号提供到开关(726)。举例来说，当所述模式是帧内模式时，通用控制器(721)控制开关(726)以选择供残差计算器(723)使用的帧内模式结果，且控制熵编码器(725)以选择帧内预测信息且将所述帧内预测信息添加在码流中；以及当所述模式是帧间模式时，通用控制器(721)控制开关(726)以选择供残差计算器(723)使用的帧间预测结果，且控制熵编码器(725)以选择帧间预测信息且将所述帧间预测信息添加在码流中。

[0097] 残差计算器(723)用于计算所接收的块与选自帧内编码器(722)或帧间编码器(730)的预测结果之间的差(残差数据)。残差编码器(724)用于基于残差数据操作，以对残差数据进行编码以生成变换系数。在实施例中，残差编码器(724)用于将残差数据从时域转换到频域，且生成变换系数。变换系数接着经由量化处理以获得量化的变换系数。在各种实施例中，视频编码器(703)还包括残差解码器(728)。残差解码器(728)用于执行逆变换，且生成已解码残差数据。已解码残差数据可适当地由帧内编码器(722)和帧间编码器(730)使用。举例来说，帧间编码器(730)可基于已解码残差数据和帧间预测信息生成已解码块，且帧内编码器(722)可基于已解码残差数据和帧内预测信息生成已解码块。适当处理已解码

块以生成已解码图片,且在一些实施例中,所述已解码图片可在存储器电路(未示出)中缓冲并用作参考图片。

[0098] 熵编码器(725)用于将码流格式化以产生已编码的块。熵编码器(725)根据HEVC标准等合适标准产生各种信息。在实施例中,熵编码器(725)用于获得通用控制数据、所选预测信息(例如帧内预测信息或帧间预测信息)、残差信息和码流中的其它合适的信息。应注意,根据所公开的主题,当在帧间模式或双向预测模式的合并子模式中对块进行编码时,不存在残差信息。

[0099] 图8是根据本申请公开的另一实施例的视频解码器(810)的图。视频解码器(810)用于接收作为已编码视频序列的一部分的已编码图像,且对所述已编码图像进行解码以生成重建的图片。在实施例中,视频解码器(810)用于代替图4实施例中的视频解码器(410)。

[0100] 在图8实施例中,视频解码器(810)包括如图8中所示耦接到一起的熵解码器(871)、帧间解码器(880)、残差解码器(873)、重建模块(874)和帧内解码器(872)。

[0101] 熵解码器(871)可用于根据已编码图片来重建某些符号,这些符号表示构成所述已编码图片的语法元素。此类符号可包括例如用于对所述块进行编码的模式(例如帧内模式、帧间模式、双向预测模式、后两者的合并子模式或另一子模式)、可分别识别供帧内解码器(872)或帧间解码器(880)用以进行预测的某些样本或元数据的预测信息(例如帧内预测信息或帧间预测信息)、呈例如量化的变换系数形式的残差信息等等。在实施例中,当预测模式是帧间或双向预测模式时,将帧间预测信息提供到帧间解码器(880);以及当预测类型是帧内预测类型时,将帧内预测信息提供到帧内解码器(872)。残差信息可经由逆量化并提供到残差解码器(873)。

[0102] 帧间解码器(880)用于接收帧间预测信息,且基于所述帧间预测信息生成帧间预测结果。

[0103] 帧内解码器(872)用于接收帧内预测信息,且基于所述帧内预测信息生成预测结果。

[0104] 残差解码器(873)用于执行逆量化以提取解量化的变换系数,且处理所述解量化的变换系数,以将残差从频域转换到空间域。残差解码器(873)还可能某些控制信息(用以获得量化器参数QP),且所述信息可由熵解码器(871)提供(未标示数据路径,因为这仅仅是低量控制信息)。

[0105] 重建模块(874)用于在空间域中组合由残差解码器(873)输出的残差与预测结果(可由帧间预测模块或帧内预测模块输出)以形成重建的块,所述重建的块可以是重建的图片的一部分,所述重建的图片继而可以是重建的视频的一部分。应注意,可执行解块操作等其它合适的操作来改善视觉质量。

[0106] 应注意,可使用任何合适的技术来实施视频编码器(403)、视频编码器(603)和视频编码器(703)以及视频解码器(410)、视频解码器(510)和视频解码器(810)。在实施例中,可使用一个或多个集成电路来实施视频编码器(403)、视频编码器(603)和视频编码器(703)以及视频解码器(410)、视频解码器(510)和视频解码器(810)。在另一实施例中,可使用执行软件指令的一个或多个处理器来实施视频编码器(403)、视频编码器(603)和视频编码器(703)以及视频解码器(410)、视频解码器(510)和视频解码器(810)。

[0107] 在一些实施例中,例如在HEVC中,主变换可以包括4点、8点、16点和32点离散余弦

变换 (DCT, Discrete Cosine Transform) 类型2 (DCT-2), 并且变换核心矩阵可以使用8比特整数表示 (即, 8比特变换核心 (8-bit transform core))。较小DCT-2的变换核心矩阵是较大DCT-2的变换核心矩阵的一部分, 如附录I所示。

[0108] DCT-2核心矩阵显示对称/反对称特性。因此, 可以支持“部分蝶式”实施方式来减少操作次数 (例如, 乘法、加法、减法、移位等), 并且可以使用部分蝶式来获得矩阵乘法的相同结果。

[0109] 在一些实施例中, 例如在VVC中, 除了上述的4点、8点、16点和32点的DCT-2变换之外, 还可以包括附加的2点DCT-2和64点DCT-2。在附录II中以 64×64 矩阵的形式显示了例如在VVC中使用的64点DCT-2核心的示例。

[0110] 除了例如在HEVC中使用的DCT-2和 4×4 DST-7之外, 自适应多变换 (AMT, Adaptive Multiple Transform) (也称为增强型多变换 (EMT, Enhanced Multiple Transform) 或多变换选择 (MTS, Multiple Transform Selection)) 方案可以在例如VVC中使用, 以用于帧间和帧内编码块的残差编解码。除了HEVC中的当前变换之外, AMT方案可以使用从DCT/DST族选择的多个变换。新引入的变换矩阵是DST-7、DCT-8。表1示出了针对N点输入的选择的DST/DCT的基函数的示例。

[0111]	变换类型 (Transform Type)	基函数 (Basis function) $T_i(j), i, j=0, 1, \dots, N-1$
[0112]	DCT-2	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot i \cdot (2j+1)}{2N}\right)$ 其中, $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i=0 \\ 1 & i \neq 0 \end{cases}$
	DCT-8	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \cos\left(\frac{\pi \cdot (2i+1) \cdot (2j+1)}{4N+2}\right)$
	DST-7	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \sin\left(\frac{\pi \cdot (2i+1) \cdot (j+1)}{2N+1}\right)$

[0113] 表1

[0114] 例如在VVC中使用的主变换矩阵可以以8比特表示进行使用。在实施例中, AMT将变换矩阵应用于宽度和高度都小于或等于32的CU。是否应用AMT可以由标志 (例如, mts_flag) 控制。当mts_flag等于0时, 在一些示例中, 仅应用DCT-2来对残差数据进行编解码。当mts_flag等于1时, 使用2个二进制数进一步用信号通知索引 (例如, mts_idx), 以识别例如根据表2要使用的水平和垂直变换, 其中, 类型值1表示使用DST-7, 并且类型值2表示使用DCT-8。在表2中, trTypeHor和trTypeVer的规格取决于mts_idx[x][y][cIdx]。

[0115]	mts_idx[xTbY][yTbY][cIdx]	trTypeHor	trTypeVer
	-1	0	0
	0	1	1
	1	2	1

2	1	2
3	2	2

[0116] 表2

[0117] 在一些实施例中,当不使用上述基于信令的MTS(即,显式MTS)时,可以应用隐式MTS。利用隐式MTS,根据块宽度和高度而不是信令来进行变换的选择。例如,利用隐式MTS,选择DST-7以用于 $M \times N$ 的块的较短侧(即, M 和 N 中的最小一侧),并且选择DCT-2以用于该块的较长侧(即, M 和 N 中的最大一侧)。

[0118] 在附录III中图示了DST-7和DCT-8的示例性变换核心,每个示例性变换核心是由基矢量组成的矩阵。

[0119] 在一些示例中,例如在VVC中,当编解码块的高度和宽度都小于或等于64时,TB大小与编解码块大小相同。当编解码块的高度或宽度大于64时,在进行变换(例如,逆变换、主逆变换等)或帧内预测时,编解码块被进一步分割成多个子块,其中,每一子块的宽度和高度小于或等于64。可以对每一子块执行一次变换。

[0120] VVC中的一些示例中的MTS的相关语法和语义可以在下面的图9和图10A至图10C中描述(使用标签910和标签1010突出显示)。图9示出了变换单元语法的示例。图10A至图10C示出了残差编解码语法的示例。

[0121] 变换单元语义的示例如下。`cu_mts_flag[x0][y0]`等于1指示将多变换选择应用于关联的亮度变换块的残差样本。`cu_mts_flag[x0][y0]`等于0指示不将多变换选择应用于关联的亮度变换块的残差样本。数组索引 x_0 、 y_0 指示所考虑的变换块的左上角(top-left)的亮度样本相对于图片的左上角的亮度样本的位置(x_0, y_0)。当`cu_mts_flag[x0][y0]`不存在时,推断其等于0。

[0122] 残差编解码语义的示例如下。`mts_idx[x0][y0]`指示沿当前变换块的水平和垂直方向将哪些变换核应用于亮度残差样本。数组索引 x_0 、 y_0 指示所考虑的变换块的左上角的亮度样本相对于图片的左上角的亮度样本的位置(x_0, y_0)。当`mts_idx[x0][y0]`不存在时,推断其等于-1。

[0123] 图11A示出了由编码器执行的示例性正向变换(也称为正向主变换)。正向变换可以包括正向水平变换和正向垂直变换。首先将正向水平变换应用于具有残差数据的残差块(1110)以获得中间块(intermediate block)。随后,将正向垂直变换应用于中间块以获得具有变换系数的系数块(1112)。

[0124] 图11B示出了由解码器执行的示例性反向变换(也称为主逆变换或逆变换)。一般而言,逆变换与正向变换相匹配。主逆变换可以包括主水平逆变换(也称为水平逆变换)和主垂直逆变换(也称为垂直逆变换)。为了匹配正向变换,在逆变换中切换应用水平逆变换和垂直逆变换的顺序。因此,将垂直逆变换首先应用于系数块(1122)以获得中间块。随后,将水平逆变换应用于中间块以获得残差块(1120)。

[0125] 主变换可以指正向主变换或主逆变换。水平变换可以指水平逆变换或正向水平变换。类似地,垂直变换可以指垂直逆变换或正向垂直变换。

[0126] 在示例中,例如在VVC中,在解码器处,首先执行垂直主逆变换(也称为垂直逆变换),然后在应用垂直逆变换之后,再执行水平主逆变换(也称为水平逆变换),如图11B所述和使用图12A至图12E中的标签1210至1211突出显示的文本所示。图12A至图12E示出了经缩

放的变换系数的变换过程的示例。

[0127] 主变换,例如正向主变换或主逆变换,可以利用如下面描述的零输出(zero-out)方法或零输出方案。在一些示例中,例如在VVC中,对于64点(或64长度)DCT-2,仅计算前32个系数,并且将剩余系数设定为0。因此,对于使用DCT-2变换进行编解码的 $M \times N$ 块,计算左上角(top-left)的 $\min(M, 32) \times \min(N, 32)$ 低频系数。剩余系数被设定为0并且不发信号通知。在示例中,不计算剩余系数。可通过将系数块大小设定为 $\min(M, 32) \times \min(N, 32)$ 来进行系数块的熵编解码,以将 $M \times N$ 块的系数编解码视为 $\min(M, 32) \times \min(N, 32)$ 的系数块。

[0128] 在使用MTS的一些示例中,对于32点DST-7或DCT-8,仅计算前16个系数,并且将剩余系数设定为0。因此,对于使用DST-7或DCT-8变换进行编码的 $M \times N$ 块,保持左上角的 $\min(M, 16) \times \min(N, 16)$ 低频系数。剩余的系数可以被设定为0并且不发信号通知。然而,与应用64点零输出(64-point zero-out)DCT-2时使用的系数编解码方案不同,对于32点MTS,即使当 M 或 N 大于16时,仍对整个 $M \times N$ 块执行系数编解码。然而,当系数群组(CG,coefficient group)在左上角的 16×16 低频区域之外(即,系数群组在零输出区域中)时,不发信号通知用于指示系数群组是否具有非零系数的标志(例如,coded_sub_block_flag)。零输出区域是指系数块中系数为零的区域,因此,零输出区域中的系数为零。下文描述残差编解码语法的示例,其由图13中的标签1310进行突出显示的文本来指示。

[0129] 主变换(例如主逆变换或正向主变换)的一个主要复杂性方面是每个样本的乘法的平均数(MPS,multiplications per sample)。例如,对于 8×8 主变换,如果将使用全矩阵乘法的DST-7应用于 8×8 块的水平和垂直变换,则水平变换需要8次乘法来计算每个系数,并且垂直变换需要8次乘法来计算每个系数,因此每个样本总共使用16次乘法。

[0130] 在一些示例中,最坏情况的MPS是 32×32 DCT2和 32×64 DCT-2,如下所述。MPS也可以称为每个系数的乘法。

[0131] 对于 $M \times N$ 的TU,如果对于TU的左上角的 $m \times n$ (即, m 乘以 n)区域保持非零变换系数,则可以通过 $Muls_per_coeff_{directMatrixMultiply} = \frac{m}{M}n + m$ 计算在直接矩阵乘法结构中实现的逆变换的每个系数的乘法数(number of multiplications)。当 n 小于 N 或 m 小于 M 时,应用零输出方法,并且可以减少每个系数的乘法数。

[0132] 同样,可以通过 $Muls_per_coeff_{halfButterFly} = \frac{m}{M}(\frac{n}{2} + \log_2 \frac{n}{2}) + (\frac{m}{2} + \log_2 \frac{m}{2})$ 计算在半蝶形结构和半直接矩阵乘法结构中实现的逆变换的每个系数的乘法数。

[0133] 在一些示例中,例如在VVC中,针对主变换(例如,4点DCT-2、8点DCT-2、16点DCT-2、32点DCT-2和64点DCT-2)定义了高达32点的DST7/DCT8变换(例如,4点DST-7、8点DST-7、16点DST-7、32点DST-7、4点DCT-8、8点DCT-8、16点DCT-8和32点DCT-8)和高达64点的DCT2变换。对于64点主变换(例如64点DCT-2),可以应用变换系数零输出方案(也称为零输出方法、零输出方案)。对于零输出方案,对于64点主变换,在一些示例中,仅保持前32个系数,并且将剩余的系数设定为0。例如,对于 64×64 块,保持左上角的 32×32 系数,并且将剩余的系数设定为0。对于 $64 \times N$ 块,其中 N 小于或等于32,可以保持左边的 $32 \times N$ 系数,并且将剩余的系数设定为0。对于 $M \times 64$ 块,其中 M 小于或等于32,可以保持顶部的 $M \times 32$ 系数,并且将剩余的系数设定为0。

[0134] 表3示出了每个系数的乘法数,其中 M 是块宽度, N 是块高度, m 是保持在水平方向

(即,在每行中)的系数的数量,并且n是保持在垂直方向(即,在每列中)的系数的数量。如表3所图示,当DST-7/DCT-8的左上角的16x16系数保持不变(即m=n=16)时,最坏的情况(例如,每个系数的乘法数是最大次数)包括32×64DCT-2和32×32DCT-2。例如,对于32×64DCT-2和32×32DCT-2,每个系数的乘法数是40(使用半蝶形),并且对应于最差情况。

[0135] 在一些示例中,在视频编解码中可以采用不具有零输出且使用半蝶形的32×32DCT2、具有零输出(m=n=16)的32×32DST7/DCT8、具有零输出(m=n=16)的32×16DST7/DCT8、具有零输出(m=n=16)的16×32DST7/DCT8。

	M	N	m	n	每个系数的乘法数(直接矩阵乘法)	每个系数的乘法数(半蝶形)
DCT2	TU 大小 M*N, 并且保持左上角的 m*n 个非零系数					
	64	64	32	32	48	30
	64	32	32	32	48	30
	32	64	32	32	64	40
	32	32	32	32	64	40
DST7/DCT8	TU 大小 M*N, 并且保持左上角的 m*n 个非零系数					
	32	32	32	32	64	
	32	16	32	16	48	
	16	32	16	32	48	
	32	32	16	16	24	
	32	16	16	16	24	
	16	32	16	16	32	

[0137] 表3

[0138] 在实施例中,可以在编码器侧的正向核心变换与量化之间及在解码器侧的解量化与逆向核心变换之间使用模式相关的不可分离二次变换(NSST, non-separable secondary transform)。例如,为了保持低复杂度,在主变换(或核心变换)之后将NSST应用于低频系数。当变换系数块的宽度(W)和高度(H)两者均大于或等于8时,对变换系数块的左上角8×8区域应用8×8NSST。否则,当变换系数块的宽度W或高度H为4时,应用4×4NSST,并且对变换系数块的左上角min(8,W)×min(8,H)区域执行4×4NSST。将上述变换选择方法应用于亮度分量和色度分量两者。

[0139] 下面使用4×4输入块作为示例来描述NSST的矩阵乘法实施方式。4×4输入块X在等式(1)中示出为:

$$X = \begin{bmatrix} X_{00} & X_{01} & X_{02} & X_{03} \\ X_{10} & X_{11} & X_{12} & X_{13} \\ X_{20} & X_{21} & X_{22} & X_{23} \\ X_{30} & X_{31} & X_{32} & X_{33} \end{bmatrix} \quad (1)$$

[0141] 在等式(2)中输入块X可以表示为向量 \vec{X} ,其中:

$$\vec{X} = [X_{00} X_{01} X_{02} X_{03} X_{10} X_{11} X_{12} X_{13} X_{20} X_{21} X_{22} X_{23} X_{30} X_{31} X_{32} X_{33}]^T \quad (2)$$

[0143] 不可分离变换可以计算为 $\vec{F} = T \cdot \vec{X}$, 其中, \vec{F} 指示变换系数矢量, 并且T为 16×16 变换矩阵。随后使用输入块X的扫描顺序 (例如, 水平扫描顺序、垂直扫描顺序或对角线扫描顺序) 将 16×1 变换系数矢量 \vec{F} 重组为 4×4 块。可以将具有较小索引的系数与较小扫描索引一起放置在 4×4 系数块中。在一些实施例中, 可以使用具有蝶形实施方式的超立方体-吉文斯 (Hypercube-Givens) 变换 (HyGT) 来代替上述矩阵乘法, 以降低NSST的复杂度。

[0144] 在示例中, 35×3 不可分离二次变换可用于 4×4 块大小和 8×8 块大小两者, 其中, 35是与帧内预测模式相关联的变换集的数量, 并且3是每个帧内预测模式的NSST候选的数量。表4示出了从帧内预测模式到各个变换集的示例性映射。根据示出了从帧内预测模式到变换集索引的映射的表4, 由对应的亮度/色度帧内预测模式指示应用于亮度/色度变换系数的变换集。对于对应于对角线预测方向的大于34的帧内预测模式, 在编码器/解码器的NSST之前/之后分别对变换系数块进行转置。

[0145] 对于每个变换集, 选择的NSST候选可以进一步由显式地用信号通知的CU级的NSST索引来指示。在变换系数之后, 在每个帧内编码CU的码流中用信号通知CU级NSST索引, 并且将截断的一元二进制定用于CU级NSST索引。例如, 对于平面模式或DC模式, 截断值为2, 并且对于角度帧内预测模式, 截断值为3。在示例中, 仅在CU中存在一个以上非零系数时, 发信号通知CU级NSST索引。默认值为零并且没有发信号通知, 这指示NSST没有被应用于CU。值1至3中的每一个值指示将从变换集中应用哪一个NSST候选。

[0146]	帧内模式	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	集	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	帧内模式	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
	集	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
[0147]	帧内模式	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
	集	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
	帧内模式	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67(LM)
	集	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	Null

[0148] 表4

[0149] 在一些实施例中, NSST不应用于以变换跳过模式编码的块。当为CU发信号通知CU级NSST索引并且CU级NSST索引不等于零时, 不将NSST用于在CU中以变换跳过模式编码的块。当具有所有分量的块的CU以变换跳过模式编码或者非变换跳过模式CB的多个非零系数小于2时, 不为CU发信号通知CU级NSST索引。

[0150] 参考表3, 当在主变换中使用DCT-2时, 用于主变换的最坏的情况MPS可以对应于大小为 32×32 或 32×64 的TB。在示例中, 最坏的情况MPS相对于第二坏的情况 (例如, 当使用DCT2时, TB大小为 64×64 或 64×32) 多使用1/3的乘法。本公开的各方面包括当使用DCT-2执

行主变换(例如主逆变换)时减小大小为 32×32 或 32×64 的TB的MPS的方法。本文的实施例还可以应用于减小任何合适的TB大小的MPS并且使用不限于DCT-2的任何合适的变换。合适的变换可以使用DCT和/或DST族中的变换矩阵,包括DCT-2、DST-7、DST-4、DCT-8、DCT-4等。

[0151] 本文中描述的实施例可以单独使用或以任何顺序组合使用。进一步地,可以通过处理电路(例如,至少一个处理器或者至少一个集成电路)来实现实施例。在一个示例中,至少一个处理器可以执行存储在非易失性计算机可读介质中的程序。

[0152] 在本公开中,适用于MTS候选的DST-7的实施例可适用于DST-4,反之亦然。类似地,适用于MTS候选的DCT-8的实施例也适用于DCT-4,反之亦然。

[0153] 在本公开中,适用于NSST的实施例可适用于作为NSST的替代设计的减少二次变换(RST, Reduced Secondary Transform)。二次变换可以指NSST或RST。

[0154] 如上所述,在图11B中,在TB上应用主逆变换的第一顺序是首先应用垂直逆变换,然后应用水平逆变换。根据一些实施例,第二顺序可用于应用主逆变换,其中,首先应用水平逆变换,并且在应用水平逆变换之后应用垂直逆变换。因此,第一顺序可以切换到第二顺序。通常,主逆变换可以使用任何合适类型的主变换,例如包括DCT-2、DST-7、DST-4、DCT-8、DCT-4等的DCT和/或DST族中的变换矩阵。在一些实施例中,利用DCT-2变换矩阵来实现主逆变换。在示例中,利用直接矩阵乘法结构来实现主逆变换。在另一示例中,利用半蝶式和半直接矩阵乘法的结构来实现主逆变换。

[0155] 在实施例中,对于某些TB大小使用第二顺序(例如,将第一顺序切换到第二顺序)。因此,应用水平逆变换和垂直逆变换的顺序可以取决于TB大小。TB大小可以指TB的面积,例如 $M \times N$ 、TB的宽度、TB的高度等。在示例中,当TB大小为 32×64 时,使用第二顺序(即,在垂直逆变换之前应用水平逆变换)。当TB大小不同于 32×64 时,使用第一顺序(即,在水平逆变换之前应用垂直逆变换)。

[0156] 在实施例中,当 $M \times N$ 的TB的高度 N 大于TB的宽度 M 时,可以使用第二顺序,以在垂直逆变换之前应用水平逆变换。否则,当TB的高度 N 小于或等于TB的宽度 M 时,可以使用第一顺序,使得在水平逆变换之前应用垂直逆变换。

[0157] 可以执行水平逆变换以将 $M \times N$ 的第一块中的第一元素变换成 $M \times N$ 的第二块中的第二元素。水平逆变换可以称为一维变换,例如 M 长度变换,其中,水平逆变换可以逐行应用于第一块,其中,每行具有 M 个第一元素。在实施例中,通过水平逆变换来计算第二块中左侧 m 列中的第二元素,而不计算第二块中右侧 $(M-m)$ 列中的第二元素,并且所述第二元素为零。当水平逆变换包括零输出方法时, m 是小于 M 的正整数,上述水平逆变换计算第二块的每行中的前 m 个第二元素,而每行中剩余的 $(M-m)$ 个第二元素不进行计算并且为零。因此,通过使用零输出方法的 M 长度变换来保持(每行中的)前 m 个第二元素。因此,可以通过包括或使用零输出方法来减少计算次数(以及要计算和存储的数据量)。当水平逆变换不包括零输出方法时, m 可以等于 M 。

[0158] 可以执行垂直逆变换以将 $M \times N$ 的第三块中的第三元素变换成 $M \times N$ 的第四块中的第四元素。可以将垂直逆变换称为一维变换,例如 N 长度变换,其中,垂直逆变换可以逐列应用于第三块,其中,每列具有 N 个第三元素。在实施例中,通过垂直逆变换来计算第四块中顶部 n 行中的第四元素,而第四块中底部 $(N-n)$ 行中的第四元素不进行计算并且为零。当垂直逆变换包括零输出方法时, n 是小于 N 的正整数,上述垂直逆变换计算第四块的每列中的前 n

个第四元素,而每列中剩余的(N-n)个第四元素不进行计算并且为零。因此,通过使用零输出方法的N长度变换来保持(每列中的)前n个第四元素。因此,可以通过包括零输出方法来减少计算次数(以及要计算和存储的数据量)。当垂直逆变换不包括零输出方法时,n可以等于N。

[0159] 主逆变换(例如,DCT-2)可应用于具有 $M \times N$ 的TB大小的系数块,以获得 $M \times N$ 的残差块。根据本公开的实施例,残差块中的 $m \times n$ 区域中的第一残差数据将由主逆变换确定,且残差块中的 $m \times n$ 区域外的第二残差数据将不由主逆变换计算,其中,m小于或等于M,且n小于或等于N。如上所述,当水平逆变换包括零输出方法时,m小于M。当垂直逆变换包括零输出方法时,n小于N。当比率 m/M 大于或等于比率 n/N 时,在应用垂直逆变换之前应用水平逆变换。否则,当比率 m/M 小于比率 n/N 时,在应用水平逆变换之前应用垂直逆变换。

[0160] 通常,执行主逆变换以将系数块中的变换系数变换为残差块中的残差数据。TB可以指系数块和/或残差块。TB大小、系数块的大小和残差块的大小为 $M \times N$ 。当比率 m/M 大于或等于比率 n/N 时,在应用垂直逆变换之前应用水平逆变换。因此,第一块是 $M \times N$ 的系数块,并且第一数据是变换系数。第二块是 $M \times N$ 的中间块,并且第二数据是中间数据。第三块与第二块相同,并且第三数据与第二数据相同。第四块是 $M \times N$ 的残差块,并且第四数据是残差数据。

[0161] 根据一些实施例,可以基于TB大小来确定一维变换中计算或保持的数据量,所述一维变换通过使用零输出方法应用于TB。如上所述,一维变换可以指水平逆变换或垂直逆变换。在一些实施例中,使用图11B所示的第一顺序执行主逆变换,以首先应用垂直逆变换,随后应用水平逆变换。当TB大小为 32×64 且宽度为32且高度为64时,通过零输出应用64长度垂直逆变换,以通过计算顶部16行中间块(即,每列中的前16段中间数据)来将具有变换系数的 32×64 的系数块变换成具有中间数据的中间块。中间块中的剩余中间数据不被计算并且为零(例如,剩余中间数据可以被设定为零)。在一些示例中,当TB大小不同于 32×64 (例如 16×64)时,64长度垂直逆变换计算中间块中的顶部32行中间数据,并且中间块中的剩余中间数据不进行计算并且为零。

[0162] 当TB大小为 32×64 并且使用第一顺序时,可以通过计算左边16列残差块(即,每一行中的前16段残差数据)来通过零输出应用32长度水平逆变换,以将具有中间数据的 32×64 的中间块变换成具有残差数据的残差块。残差块中的剩余残差数据不被计算并且为零(例如,剩余残差数据可以被设定为零)。

[0163] 在一些示例中,当TB大小为 $M \times 32$,其中,M小于32时,32长度垂直逆变换不应用零输出方案。在一些示例中,当TB大小为 $32 \times N$,其中,N小于32时,32长度水平逆变换不应用零输出方案。

[0164] 根据一些实施例,可以基于一维变换是水平逆变换还是垂直逆变换来确定要使用一维变换为行或列计算的数据量。例如,零输出方案可以仅用于一维变换中的一种。在实施例中,使用第一顺序执行主逆变换,以首先应用垂直逆变换,随后应用水平逆变换。TB大小为 32×32 。在示例中,32长度垂直变换不应用零输出方法。因此,32长度垂直逆变换计算或保持中间块的32行中的所有中间数据。32长度水平逆变换使用零输出方法计算或保持残差块的左侧16列残差块。残差块中的剩余残差数据不进行计算并且为零(例如,剩余残差数据可以被设定为零)。在示例中,32长度垂直逆变换使用零输出方法计算或保持中间块的顶部

16行。中间块中的剩余中间数据不进行计算并且为零(例如,剩余中间数据可以被设定为零)。同时,32长度水平逆变换不应用零输出方法。因此,32长度水平逆变换计算或保持残差块的32列中的所有残差数据。

[0165] 当第一主逆变换应用于TB而没有二次逆变换(inverse secondary transform)时,第一MPS归因于第一主逆变换(first inverse primary transform)。当对TB应用第二主逆变换(second inverse primary transform)和二次逆变换时,第二MPS(也称为组合MPS)可以包括第二主逆变换和二次逆变换两者中的乘法。当第一主逆变换与第二主逆变换相同时,第二MPS可以大于第一MPS,因为在二次逆变换中使用附加乘法。可以实现下面的实施例以将组合的MPS约束或减小到MPS限制内(例如,表3中示出的最坏情况的MPS),从而提高编解码效率。

[0166] 根据本公开的各方面,使用零输出方案保持的非零变换系数的数量和/或是否在主逆变换中使用零输出方法可取决于是否应用二次变换。因此,可以减少主逆变换和二次变换的组合的MPS。返回到第一主逆变换和第二主逆变换,根据本公开的方面,第一主逆变换可以计算或保持第一数量的残差数据,并且第二主逆变换可以计算或保持第二数量的残差数据,其中,第二数量的残差数据小于第一数量的残差数据,使得组合的MPS(即,第二MPS)类似于第一MPS。在示例中,第二主逆变换的至少一个一维变换(例如,水平逆变换或垂直逆变换)包括零输出方法。第一主逆变换的一维变换还可以包括零输出方法,然而,第二主逆变换可以计算或保持比第一主逆变换少的残差数据。

[0167] 如上所述,主逆变换和二次逆变换可以应用于TB并产生组合的MPS。通常,主逆变换可以包括不同类型(例如,DCT-2、DST-7、DST-4、DCT-8、DCT-4等)。根据本公开的各方面,可以基于主逆变换的类型来确定在二次逆变换中使用零输出方法时是否使用零输出方法和/或要计算或保持在系数块中的数据量,从而可减小组合的MPS。例如,当主逆变换更复杂时(例如,主逆变换的MPS相对较大),可以进一步减少在二次逆变换中使用零输出方法时在系数块中计算或保持的数据量。

[0168] 根据本公开的各方面,可以根据用于TB的帧内预测模式来执行零输出方法。在一些实施例中,可以基于帧内预测模式来确定是否使用零输出方法以及在使用零输出方法时要计算或保持的数据量。

[0169] 在实施例中,利用零输出方法执行水平逆变换还是垂直逆变换可以取决于用于TB的帧内预测方向。帧内预测方向对应于帧内预测模式。在示例中,当帧内预测方向接近水平预测方向时,将零输出方法应用于水平逆变换。在示例中,当帧内预测方向接近垂直预测方向时,对垂直逆变换应用零输出方法。例如,水平预测方向由值Hor索引,并且垂直预测方向由值Ver索引。可以定义阈值thres以确定帧内预测模式与对应于水平预测方向的水平模式或对应于垂直预测方向的垂直模式的接近程度。确定在 $[\text{Hor} - \text{thres}, \text{Hor} + \text{thres}]$ 的范围内索引的帧内预测模式(或帧内预测方向)接近水平预测方向。确定在 $[\text{Ver} - \text{thres}, \text{Ver} + \text{thres}]$ 的范围内索引的帧内预测模式接近垂直预测方向。阈值thres的示例值包括但不限于1、2、3、4、...和16。

[0170] 在实施例中,可以基于TB的帧内预测方向来确定在主逆变换和/或二次变换中使用零输出方法时要计算或保持的数据量。

[0171] 图14示出了概述根据本公开的实施例的方法(1400)的流程图。可以在以帧内模式

编码的块的重建中使用方法(1400),以为重建中的块生成预测块。在一些示例中,该方法(1400)可以用于以帧间模式编码的块的重建。在各种实施例中,该方法(1400)由处理电路执行,例如终端设备(310)、(320)、(330)和(340)中的处理电路,执行视频编码器(403)的功能的处理电路,执行视频解码器(410)的功能的处理电路,执行视频解码器(510)的功能的处理电路,执行视频编码器(603)的功能的处理电路等。在一些实施例中,该方法(1400)以软件指令实现,因此当处理电路执行这些软件指令时,处理电路执行方法(1400)。该方法开始于(S1401)并进行到(S1410)。

[0172] 在(S1410)处,对TB(例如系数块)的编码信息进行解码,所述TB来自自己编码视频码流。所述编码信息可以指示TB大小(或称为系数块的大小)。如上所述,TB大小可以指TB的面积、宽度、高度等。

[0173] 在(S1420)处,可以基于系数块的大小来确定要执行主逆变换的水平逆变换和垂直逆变换的顺序。可以通过主逆变换将系数块的变换系数变换成残差块的残差数据。如上所述,在示例中,该顺序是第一顺序。在示例中,该顺序是第二顺序。

[0174] 在(S1430)处,如上所述,可以确定TB大小是否满足条件。条件可以是TB大小是否为 32×64 。条件可以是 $M \times N$ 的系数块的高度 N 大于系数块的宽度 M 。条件可以比较第一比率与第二比率,其中,第一比率(例如, m/M)是针对一行计算的数据量相对于该行中的总数据量,并且第二比率(n/N)是针对一列计算的数据量相对于该列中的总数据量。当确定TB大小满足条件时,方法(1400)进行到(S1440)。否则,方法(1400)进行到(S1450)。

[0175] 在(S1440)处,在对系数块的变换系数执行水平逆变换之后执行垂直逆变换,并且因此使用第二顺序。

[0176] 在(S1450)处,在对系数块的变换系数执行垂直逆变换之后执行水平逆变换,并且因此使用第一顺序。

[0177] 在(S1460)处,可以基于残差数据来重建残差块中的样本。

[0178] 可以适当地修改方法(1400)。例如,可以修改、省略或组合至少一个步骤。例如,可以组合(S1420)和(S1430)。可以添加一个或多个附加步骤。还可以修改执行方法(1400)的顺序。

[0179] 图15示出了概述根据本公开的实施例的方法(1500)的流程图。可以在以帧内模式编码的块的重建中使用方法(1500),以便为重建中的块生成预测块。在一些示例中,该方法(1500)可以用于以帧间模式编码的块的重建。在各种实施例中,方法(1500)由处理电路执行,例如终端设备(310)、(320)、(330)和(340)中的处理电路,执行视频编码器(403)的功能的处理电路,执行视频解码器(410)的功能的处理电路,执行视频解码器(510)的功能的处理电路,执行视频编码器(603)的功能的处理电路等。在一些实施例中,该方法(1500)以软件指令实现,因此当处理电路执行这些软件指令时,处理电路执行方法(1500)。该方法开始于(S1501)并进行到(S1510)。

[0180] 在(S1510)处,对TB(例如系数块)的编码信息进行解码,所述TB来自自己编码视频码流。编码信息可以指示TB大小(或称为系数块的大小)。

[0181] 在(S1520)处,可以基于系数块的大小来确定是否减少主逆变换中的水平逆变换和垂直逆变换之一的计算次数(或要计算的数据量)。可以通过主逆变换将系数块的变换系数变换成残差块的残差数据。在实施例中,使用第一顺序。垂直逆变换可以将系数块的变换

系数变换成中间块的中间数据,并且水平逆变换可以将中间数据变换成残差块中的残差数据。在示例中,减少一维变换(例如,水平逆变换或垂直逆变换)中的计算次数意味着一维变换包括零输出方法,其中,计算数据的前16行(用于水平逆变换)或列(用于垂直逆变换)且不计算剩余数据。例如,当TB大小为 32×64 和 32×32 之一时,确定一维变换中的计算次数为减少。当确定一维变换中的计算次数要减少时,方法(1500)进行到(S1530)。

[0182] 在(S1530)处,可以确定是否要减少水平逆变换中的计算次数。当要减少水平逆变换中的计算次数时,方法(1500)进行到(S1540)。否则,方法(1500)进行到(S1550)。

[0183] 可以适当地修改(S1530)。例如,代替在(S1530)处确定是否要减少水平逆变换中的计算次数,可以确定是否要减少垂直逆变换中的计算次数。

[0184] 在(S1540)处,执行主逆变换,其中,水平逆变换包括零输出方法,并且如上所述,通过水平逆变换计算残差块中的左侧16列残差数据,并且残差块中的剩余残差数据不进行计算并且为零。

[0185] 在示例中,在(S1540)处,垂直逆变换还包括零输出方法。

[0186] 在(S1550)处,执行主逆变换,其中,垂直逆变换包括零输出方法,并且因此通过垂直逆变换计算中间块中的顶部16行中间数据,并且中间块中的剩余中间数据不进行计算并且为零,如上所述。

[0187] 在(S1560)处,可以基于残差数据来重建残差块。

[0188] 可以适当地修改方法(1500)。例如,可以修改、省略或组合至少一个步骤。例如,步骤(S1520)和(S1530)可以组合成单个步骤。可以添加一个或多个附加步骤。还可以修改执行方法(1500)的顺序。

[0189] 在示例中,可以修改和组合方法(1400)中的一个或多个步骤和方法(1500)中的一个或多个步骤。

[0190] 上述技术可以通过计算机可读指令实现为计算机软件,并且物理地存储在一个或多个计算机可读介质中。例如,图16示出了计算机系统(1600),其适于实现所公开主题的某些实施例。

[0191] 所述计算机软件可通过任何合适的机器代码或计算机语言进行编码,通过汇编、编译、链接等机制创建包括指令的代码,所述指令可由一个或多个计算机中央处理单元(CPU),图形处理单元(GPU)等直接执行或通过译码、微代码等方式执行。

[0192] 所述指令可以在各种类型的计算机或其组件上执行,包括例如个人计算机、平板电脑、服务器、智能手机、游戏设备、物联网设备等。

[0193] 图16所示的用于计算机系统(1600)的组件本质上是示例性的,并不用于对实现本申请实施例的计算机软件的使用范围或功能进行任何限制。也不应将组件的配置解释为与计算机系统(1600)的示例性实施例中所示的任一组件或其组合具有任何依赖性 or 要求。

[0194] 计算机系统(1600)可以包括某些人机界面输入设备。这种人机界面输入设备可以通过触觉输入(如:键盘输入、滑动、数据手套移动)、音频输入(如:声音、掌声)、视觉输入(如:手势)、嗅觉输入(未示出),对一个或多个人类用户的输入做出响应。所述人机界面设备还可用于捕获某些媒体,气与人类有意识的输入不必直接相关,如音频(例如:语音、音乐、环境声音)、图像(例如:扫描图像、从静止影像相机获得的摄影图像)、视频(例如二维视频、包括立体视频的三维视频)。

[0195] 人机界面输入设备可包括以下中的一个或多个(仅绘出其中一个):键盘(1601)、鼠标(1602)、触控板(1603)、触摸屏(1610)、数据手套(未示出)、操纵杆(1605)、麦克风(1606)、扫描仪(1607)、照相机(1608)。

[0196] 计算机系统(1600)还可以包括某些人机界面输出设备。这种人机界面输出设备可以通过例如触觉输出、声音、光和嗅觉/味觉来刺激一个或多个人类用户的感受。这样的人机界面输出设备可包括触觉输出设备(例如通过触摸屏(1610)、数据手套(未示出)或操纵杆(1605)的触觉反馈,但也可以有不用作输入设备的触觉反馈设备)、音频输出设备(例如,扬声器(1609)、耳机(未示出))、视觉输出设备(例如,包括阴极射线管屏幕、液晶屏幕、等离子屏幕、有机发光二极管屏的屏幕(1610),其中每一个都具有或没有触摸屏输入功能、每一个都具有或没有触觉反馈功能——其中一些可通过诸如立体画面输出的手段输出二维视觉输出或三维以上的输出;虚拟现实眼镜(未示出)、全息显示器和放烟箱(未示出))以及打印机(未示出)。

[0197] 计算机系统(1600)还可以包括人可访问的存储设备及其相关介质,如包括具有CD/DVD的高密度只读/可重写式光盘(CD/DVD ROM/RW)(1620)或类似介质(1621)的光学介质、拇指驱动器(1622)、可移动硬盘驱动器或固体状态驱动器(1623),诸如磁带和软盘(未示出)的传统磁介质,诸如安全软件保护器(未示出)等的基于ROM/ASIC/PLD的专用设备,等等。

[0198] 本领域技术人员还应当理解,结合所公开的主题使用的术语“计算机可读介质”不包括传输介质、载波或其它瞬时信号。

[0199] 计算机系统(1600)还可以包括通往一个或多个通信网络的接口。例如,网络可以是无线的、有线的、光学的。网络还可为局域网、广域网、城域网、车载网络和工业网络、实时网络、延迟容忍网络等等。网络还包括以太网、无线局域网、蜂窝网络(GSM、3G、4G、5G、LTE等)等局域网、电视有线或无线广域数字网络(包括有线电视、卫星电视、和地面广播电视)、车载和工业网络(包括CANBus)等等。某些网络通常需要外部网络接口适配器,用于连接到某些通用数据端口或外围总线(1649)(例如,计算机系统(1600)的USB端口);其它系统通常通过连接到如下所述的系统总线集成到计算机系统(1600)的核心(例如,以太网接口集成到PC计算机系统或蜂窝网络接口集成到智能电话计算机系统)。通过使用这些网络中的任何一个,计算机系统(1600)可以与其它实体进行通信。所述通信可以是单向的,仅用于接收(例如,无线电视),单向的仅用于发送(例如CAN总线到某些CAN总线设备),或双向的,例如通过局域或广域数字网络到其它计算机系统。上述的每个网络和网络接口可使用某些协议和协议栈。

[0200] 上述的人机界面设备、人可访问的存储设备以及网络接口可以连接到计算机系统(1600)的核心(1640)。

[0201] 核心(1640)可包括一个或多个中央处理单元(CPU)(1641)、图形处理单元(GPU)(1642)、以现场可编程门阵列(FPGA)(1643)形式的专用可编程处理单元、用于特定任务的硬件加速器(1644)等。这些设备以及只读存储器(ROM)(1645)、随机存取存储器(1646)、内部大容量存储器(例如内部非用户可存取硬盘驱动器、固态硬盘等)(1647)等可通过系统总线(1648)进行连接。在某些计算机系统中,可以以一个或多个物理插头的形式访问系统总线(1648),以便可通过额外的中央处理单元、图形处理单元等进行扩展。外围装置可直接附

接到核心的系统总线(1648),或通过外围总线(1649)进行连接。外围总线的体系结构包括外部控制器接口PCI、通用串行总线USB等。

[0202] CPU(1641)、GPU(1642)、FPGA(1643)和加速器(1644)可以执行某些指令,这些指令组合起来可以构成上述计算机代码。该计算机代码可以存储在ROM(1645)或RAM(1646)中。过渡数据也可以存储在RAM(1646)中,而永久数据可以存储在例如内部大容量存储器(1647)中。通过使用高速缓冲存储器可实现对任何存储器设备的快速存储和检索,高速缓冲存储器可与一个或多个CPU(1641)、GPU(1642)、大容量存储器(1647)、ROM(1645)、RAM(1646)等紧密关联。

[0203] 所述计算机可读介质上可具有计算机代码,用于执行各种计算机实现的操作。介质和计算机代码可以是为本申请的目的而特别设计和构造的,也可以是计算机软件领域的技术人员所熟知和可用的介质和代码。

[0204] 作为实施例而非限制,具有体系结构(1600)的计算机系统,特别是核心(1640),可以作为处理器(包括CPU、GPU、FPGA、加速器等)提供执行包含在一个或多个有形的计算机可读介质中的软件的功能。这种计算机可读介质可以是与上述的用户可访问的大容量存储器相关联的介质,以及具有非易失性的核心(1640)的特定存储器,例如核心内部大容量存储器(1647)或ROM(1645)。实现本申请的各种实施例的软件可以存储在这种设备中并且由核心(1640)执行。根据特定需要,计算机可读介质可包括一个或一个以上存储设备或芯片。该软件可以使得核心(1640)特别是其中的处理器(包括CPU、GPU、FPGA等)执行本文所述的特定过程或特定过程的特定部分,包括定义存储在RAM(1646)中的数据结构以及根据软件定义的过程来修改这种数据结构。另外或作为替代,计算机系统可以提供逻辑硬连线或以其它方式包含在电路(例如,加速器(1644))中的功能,该电路可以代替软件或与软件一起运行以执行本文所述的特定过程或特定过程的特定部分。在适当的情况下,对软件的引用可以包括逻辑,反之亦然。在适当的情况下,对计算机可读介质的引用可包括存储执行软件的电路(如集成电路(IC)),包含执行逻辑的电路,或两者兼备。本申请包括任何合适的硬件和软件组合。

[0205] 附录A:首字母缩略词

[0206] JEM:联合探索模型

[0207] VVC:下一代视频编码

[0208] BMS:基准集合

[0209] MV:运动向量

[0210] HEVC:高效视频编码

[0211] SEI:补充增强信息

[0212] VUI:视频可用性信息

[0213] GOPs:图片组

[0214] TUs:变换单元

[0215] PUs:预测单元

[0216] CTUs:编码树单元

[0217] CTBs:编码树块

[0218] PBs:预测块

- [0219] HRD:假想参考解码器
- [0220] SNR:信噪比
- [0221] CPUs:中央处理单元
- [0222] GPUs:图形处理单元
- [0223] CRT:阴极射线管
- [0224] LCD:液晶显示
- [0225] OLED:有机发光二极管
- [0226] CD:光盘
- [0227] DVD:数字化视频光盘
- [0228] ROM:只读存储器
- [0229] RAM:随机存取存储器
- [0230] ASIC:专用集成电路
- [0231] PLD:可编程逻辑设备
- [0232] LAN:局域网
- [0233] GSM:全球移动通信系统
- [0234] LTE:长期演进
- [0235] CANBus:控制器局域网总线
- [0236] USB:通用串行总线
- [0237] PCI:外围设备互连
- [0238] FPGA:现场可编程门阵列
- [0239] SSD:固态驱动器
- [0240] IC:集成电路
- [0241] CU:编码单元

[0242] 虽然本申请已对多个示例性实施例进行了描述,但实施例的各种变更、排列和各种等同替换均属于本申请的范围内。因此应理解,本领域技术人员能够设计多种系统和方法,所述系统和方法虽然未在本文中明确示出或描述,但其体现了本申请的原则,因此属于本申请的精神和范围之内。

- [0243] 附录I
- [0244] 4x4变换
- [0245] {64, 64, 64, 64}
- [0246] {83, 36, -36, -83}
- [0247] {64, -64, -64, 64}
- [0248] {36, -83, 83, -36}
- [0249] 8x8变换
- [0250] {64, 64, 64, 64, 64, 64, 64, 64}
- [0251] {89, 75, 50, 18, -18, -50, -75, -89}
- [0252] {83, 36, -36, -83, -83, -36, 36, 83}
- [0253] {75, -18, -89, -50, 50, 89, 18, -75}
- [0254] {64, -64, -64, 64, 64, -64, -64, 64}

- [0285] {82 22-54-90-61 13 78 85 31-46-90-67 4 73 88 38-38-88-73 -4 67 90 46-31-85-78-13 61 90 54-22-82}
- [0286] {80 9-70-87-25 57 90 43-43-90-57 25 87 70 -9-80-80 -9 70 87 25-57-90-43 43 90 57-25-87-70 9 80}
- [0287] {78 -4-82-73 13 85 67-22-88-61 31 90 54-38-90-46 46 90 38-54-90-31 61 88 22-67-85-13 73 82 4-78}
- [0288] {75-18-89-50 50 89 18-75-75 18 89 50-50-89-18 75 75-18-89-50 50 89 18-75-75 18 89 50-50-89-18 75}
- [0289] {73-31-90-22 78 67-38-90-13 82 61-46-88 -4 85 54-54-85 4 88 46-61-82 13 90 38-67-78 22 90 31-73}
- [0290] {70-43-87 9 90 25-80-57 57 80-25-90-9 87 43-70-70 43 87 -9-90-25 80 57-57-80 25 90 9-87-43 70}
- [0291] {67-54-78 38 85-22-90 4 90 13-88-31 82 46-73-61 61 73-46-82 31 88-13-90 -4 90 22-85-38 78 54-67}
- [0292] {64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64}
- [0293] {61-73-46 82 31-88-13 90 -4-90 22 85-38-78 54 67-67-54 78 38-85-22 90 4-90 13 88-31-82 46 73-61}
- [0294] {57-80-25 90 -9-87 43 70-70-43 87 9-90 25 80-57-57 80 25-90 9 87-43-70 70 43-87 -9 90-25-80 57}
- [0295] {54-85 -4 88-46-61 82 13-90 38 67-78-22 90-31-73 73 31-90 22 78-67-38 90-13-82 61 46-88 4 85-54}
- [0296] {50-89 18 75-75-18 89-50-50 89-18-75 75 18-89 50 50-89 18 75-75-18 89-50-50 89-18-75 75 18-89 50}
- [0297] {46-90 38 54-90 31 61-88 22 67-85 13 73-82 4 78-78 -4 82-73-13 85-67-22 88-61-31 90-54-38 90-46}
- [0298] {43-90 57 25-87 70 9-80 80 -9-70 87-25-57 90-43-43 90-57-25 87-70 -9 80-80 9 70-87 25 57-90 43}
- [0299] {38-88 73 -4-67 90-46-31 85-78 13 61-90 54 22-82 82-22-54 90-61-13 78-85 31 46-90 67 4-73 88-38}
- [0300] {36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36}
- [0301] {31-78 90-61 4 54-88 82-38-22 73-90 67-13-46 85-85 46 13-67 90-73 22 38-82 88-54 -4 61-90 78-31}
- [0302] {25-70 90-80 43 9-57 87-87 57 -9-43 80-90 70-25-25 70-90 80-43-9 57-87 87-57 9 43-80 90-70 25}
- [0303] {22-61 85-90 73-38 -4 46-78 90-82 54-13-31 67-88 88-67 31 13-54 82-90 78-46 4 38-73 90-85 61-22}
- [0304] {18-50 75-89 89-75 50-18-18 50-75 89-89 75-50 18 18-50 75-89 89-75

ci, -cb, -bu, -bn, -bg, -bk, -br, -by, -cf, cj, cc, bv, bo, bh, bj, bq, bx, ce, -ck, -cd, -bw, -bp, -bi}

[0319] {ad, ae, af, ag, -ag, -af, -ae, -ad, -ad, -ae, -af, -ag, ag, af, ae, ad, ad, ae, af, ag, -ag, -af, -ae, -ad, -ad, -ae, -af, -ag, ag, af, ae, ad, ad, ae, af, ag, -ag, -af, -ae, -ad, -ad, -ae, -af, -ag, ag, af, ae, ad, ad, ae, af, ag, -ag, -af, -ae, -ad, -ad, -ae, -af, -ag, ag, af, ae, ad}

[0320] {bj, bs, cb, ck, -cc, -bt, -bk, -bi, -br, -ca, -cj, cd, bu, bl, bh, bq, bz, ci, -ce, -bv, -bm, -bg, -bp, -by, -ch, cf, bw, bn, bf, bo, bx, cg, -cg, -bx, -bo, -bf, -bn, -bw, -cf, ch, by, bp, bg, bm, bv, ce, -ci, -bz, -bq, -bh, -bl, -bu, -cd, cj, ca, br, bi, bk, bt, cc, -ck, -cb, -bs, -bj}

[0321] {ar, aw, bb, -bd, -ay, -at, -ap, -au, -az, -be, ba, av, aq, as, ax, bc, -bc, -ax, -as, -aq, -av, -ba, be, az, au, ap, at, ay, bd, -bb, -aw, -ar, -ar, -aw, -bb, bd, ay, at, ap, au, az, be, -ba, -av, -aq, -as, -ax, -bc, bc, ax, as, aq, av, ba, -be, -az, -au, -ap, -at, -ay, -bd, bb, aw, ar}

[0322] {bk, bv, cg, -ce, -bt, -bi, -bm, -bx, -ci, cc, br, bg, bo, bz, ck, -ca, -bp, -bf, -bq, -cb, cj, by, bn, bh, bs, cd, -ch, -bw, -bl, -bj, -bu, -cf, cf, bu, bj, bl, bw, ch, -cd, -bs, -bh, -bn, -by, -cj, cb, bq, bf, bp, ca, -ck, -bz, -bo, -bg, -br, -cc, ci, bx, bm, bi, bt, ce, -cg, -bv, -bk}

[0323] {ai, al, ao, -am, -aj, -ah, -ak, -an, an, ak, ah, aj, am, -ao, -al, -ai, -ai, -al, -ao, am, aj, ah, ak, an, -an, -ak, -ah, -aj, -am, ao, al, ai, ai, al, ao, -am, -aj, -ah, -ak, -an, an, ak, ah, aj, am, -ao, -al, -ai, -ai, -al, -ao, am, aj, ah, ak, an, -an, -ak, -ah, -aj, -am, ao, al, ai}

[0324] {bl, by, -ck, -bx, -bk, -bm, -bz, cj, bw, bj, bn, ca, -ci, -bv, -bi, -bo, -cb, ch, bu, bh, bp, cc, -cg, -bt, -bg, -bq, -cd, cf, bs, bf, br, ce, -ce, -br, -bf, -bs, -cf, cd, bq, bg, bt, cg, -cc, -bp, -bh, -bu, -ch, cb, bo, bi, bv, ci, -ca, -bn, -bj, -bw, -cj, bz, bm, bk, bx, ck, -by, -bl}

[0325] {as, az, -bd, -aw, -ap, -av, -bc, ba, at, ar, ay, -be, -ax, -aq, -au, -bb, bb, au, aq, ax, be, -ay, -ar, -at, -ba, bc, av, ap, aw, bd, -az, -as, -as, -az, bd, aw, ap, av, bc, -ba, -at, -ar, -ay, be, ax, aq, au, bb, -bb, -au, -aq, -ax, -be, ay, ar, at, ba, -bc, -av, -ap, -aw, -bd, az, as}

[0326] {bm, cb, -cf, -bq, -bi, -bx, cj, bu, bf, bt, ci, -by, -bj, -bp, -ce, cc, bn, bl, ca, -cg, -br, -bh, -bw, ck, bv, bg, bs, ch, -bz, -bk, -bo, -cd, cd, bo, bk, bz, -ch, -bs, -bg, -bv, -ck, bw, bh, br, cg, -ca, -bl, -bn, -cc, ce, bp, bj, by, -ci, -bt, -bf, -bu, -cj, bx, bi, bq, cf, -cb, -bm}

[0327] {ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab}

[0328] {bn, ce, -ca, -bj, -br, -ci, bw, bf, bv, -cj, -bs, -bi, -bz, cf, bo, bm, cd, -cb, -bk, -

bq, -ch, bx, bg, bu, -ck, -bt, -bh, -by, cg, bp, bl, cc, -cc, -bl, -bp, -cg, by, bh, bt, ck, -bu, -bg, -bx, ch, bq, bk, cb, -cd, -bm, -bo, -cf, bz, bi, bs, cj, -bv, -bf, -bw, ci, br, bj, ca, -ce, -bn}

[0329] {at, bc, -ay, -ap, -ax, bd, au, as, bb, -az, -aq, -aw, be, av, ar, ba, -ba, -ar, -av, -be, aw, aq, az, -bb, -as, -au, -bd, ax, ap, ay, -bc, -at, -at, -bc, ay, ap, ax, -bd, -au, -as, -bb, az, aq, aw, -be, -av, -ar, -ba, ba, ar, av, be, -aw, -aq, -az, bb, as, au, bd, -ax, -ap, -ay, bc, at}

[0330] {bo, ch, -bv, -bh, -ca, cc, bj, bt, -cj, -bq, -bm, -cf, bx, bf, by, -ce, -bl, -br, -ck, bs, bk, cd, -bz, -bg, -bw, cg, bn, bp, ci, -bu, -bi, -cb, cb, bi, bu, -ci, -bp, -bn, -cg, bw, bg, bz, -cd, -bk, -bs, ck, br, bl, ce, -by, -bf, -bx, cf, bm, bq, cj, -bt, -bj, -cc, ca, bh, bv, -ch, -bo}

[0331] {aj, ao, -ak, -ai, -an, al, ah, am, -am, -ah, -al, an, ai, ak, -ao, -aj, -aj, -ao, ak, ai, an, -al, -ah, -am, am, ah, al, -an, -ai, -ak, ao, aj, aj, ao, -ak, -ai, -an, al, ah, am, -am, -ah, -al, an, ai, ak, -ao, -aj, -aj, -ao, ak, ai, an, -al, -ah, -am, am, ah, al, -an, -ai, -ak, ao, aj}

[0332] {bp, ck, -bq, -bo, -cj, br, bn, ci, -bs, -bm, -ch, bt, bl, cg, -bu, -bk, -cf, bv, bj, ce, -bw, -bi, -cd, bx, bh, cc, -by, -bg, -cb, bz, bf, ca, -ca, -bf, -bz, cb, bg, by, -cc, -bh, -bx, cd, bi, bw, -ce, -bj, -bv, cf, bk, bu, -cg, -bl, -bt, ch, bm, bs, -ci, -bn, -br, cj, bo, bq, -ck, -bp}

[0333] {au, -be, -at, -av, bd, as, aw, -bc, -ar, -ax, bb, aq, ay, -ba, -ap, -az, az, ap, ba, -ay, -aq, -bb, ax, ar, bc, -aw, -as, -bd, av, at, be, -au, -au, be, at, av, -bd, -as, -aw, bc, ar, ax, -bb, -aq, -ay, ba, ap, az, -az, -ap, -ba, ay, aq, bb, -ax, -ar, -bc, aw, as, bd, -av, -at, -be, au}

[0334] {bq, -ci, -bl, -bv, cd, bg, ca, -by, -bi, -cf, bt, bn, ck, -bo, -bs, cg, bj, bx, -cb, -bf, -cc, bw, bk, ch, -br, -bp, cj, bm, bu, -ce, -bh, -bz, bz, bh, ce, -bu, -bm, -cj, bp, br, -ch, -bk, -bw, cc, bf, cb, -bx, -bj, -cg, bs, bo, -ck, -bn, -bt, cf, bi, by, -ca, -bg, -cd, bv, bl, ci, -bq}

[0335] {ae, -ag, -ad, -af, af, ad, ag, -ae, -ae, ag, ad, af, -af, -ad, -ag, ae, ae, -ag, -ad, -af, af, ad, ag, -ae, -ae, ag, ad, af, -af, -ad, -ag, ae, ae, -ag, -ad, -af, af, ad, ag, -ae, -ae, ag, ad, af, -af, -ad, -ag, ae, ae, -ag, -ad, -af, af, ad, ag, -ae, -ae, ag, ad, af, -af, -ad, -ag, ae}

[0336] {br, -cf, -bg, -cc, bu, bo, -ci, -bj, -bz, bx, bl, ck, -bm, -bw, ca, bi, ch, -bp, -bt, cd, bf, ce, -bs, -bq, cg, bh, cb, -bv, -bn, cj, bk, by, -by, -bk, -cj, bn, bv, -cb, -bh, -cg, bq, bs, -ce, -bf, -cd, bt, bp, -ch, -bi, -ca, bw, bm, -ck, -bl, -bx, bz, bj, ci, -bo, -bu, cc, bg, cf, -br}

[0337] {av, -bb, -ap, -bc, au, aw, -ba, -aq, -bd, at, ax, -az, -ar, -be, as, ay, -ay, -as, be, ar, az, -ax, -at, bd, aq, ba, -aw, -au, bc, ap, bb, -av, -av, bb, ap, bc, -au, -aw, ba, aq, bd, -at, -ax, az, ar, be, -as, -ay, ay, as, -be, -ar, -az, ax, at, -bd, -aq, -ba, aw, au, -bc, -ap, -bb, av}

[0338] {bs,-cc,-bi,-cj,bl,bz,-bv,-bp,cf,bf,cg,-bo,-bw,by,bm,-ci,-bh,-cd,br, bt,-cb,-bj,-ck,bk,ca,-bu,-bq,ce,bg,ch,-bn,-bx,bx,bn,-ch,-bg,-ce,bq,bu,-ca,- bk,ck,bj,cb,-bt,-br,cd,bh,ci,-bm,-by,bw,bo,-cg,-bf,-cf,bp,bv,-bz,-bl,cj,bi, cc,-bs}

[0339] {ak,-am,-ai,ao,ah,an,-aj,-al,al,aj,-an,-ah,-ao,ai,am,-ak,-ak,am,ai,- ao,-ah,-an,aj,al,-al,-aj,an,ah,ao,-ai,-am,ak,ak,-am,-ai,ao,ah,an,-aj,-al,al, aj,-an,-ah,-ao,ai,am,-ak,-ak,am,ai,-ao,-ah,-an,aj,al,-al,-aj,an,ah,ao,-ai,- am,ak}

[0340] {bt,-bz,-bn,cf,bh,ck,-bi,-ce,bo,by,-bu,-bs,ca,bm,-cg,-bg,-cj,bj,cd,- bp,-bx,bv,br,-cb,-bl,ch,bf,ci,-bk,-cc,bq,bw,-bw,-bq,cc,bk,-ci,-bf,-ch,bl,cb,- br,-bv,bx,bp,-cd,-bj,cj,bg,cg,-bm,-ca,bs,bu,-by,-bo,ce,bi,-ck,-bh,-cf,bn,bz,- bt}

[0341] {aw,-ay,-au,ba,as,-bc,-aq,be,ap,bd,-ar,-bb,at,az,-av,-ax,ax,av,-az,- at,bb,ar,-bd,-ap,-be,aq,bc,-as,-ba,au,ay,-aw,-aw,ay,au,-ba,-as,bc,aq,-be,- ap,-bd,ar,bb,-at,-az,av,ax,-ax,-av,az,at,-bb,-ar,bd,ap,be,-aq,-bc,as,ba,-au,- ay,aw}

[0342] {bu,-bw,-bs,by,bq,-ca,-bo,cc,bm,-ce,-bk,cg,bi,-ci,-bg,ck,bf,cj,-bh,- ch,bj,cf,-bl,-cd,bn,cb,-bp,-bz,br,bx,-bt,-bv,bv,bt,-bx,-br,bz,bp,-cb,-bn,cd, bl,-cf,-bj,ch,bh,-cj,-bf,-ck,bg,ci,-bi,-cg,bk,ce,-bm,-cc,bo,ca,-bq,-by,bs, bw,-bu}

[0343] {aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa, aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,- aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa,aa,aa,-aa,-aa, aa}

[0344] {bv,-bt,-bx,br,bz,-bp,-cb,bn,cd,-bl,-cf,bj,ch,-bh,-cj,bf,-ck,-bg,ci, bi,-cg,-bk,ce,bm,-cc,-bo,ca,bq,-by,-bs,bw,bu,-bu,-bw,bs,by,-bq,-ca,bo,cc,- bm,-ce,bk,cg,-bi,-ci,bg,ck,-bf,cj,bh,-ch,-bj,cf,bl,-cd,-bn,cb,bp,-bz,-br,bx, bt,-bv}

[0345] {ax,-av,-az,at,bb,-ar,-bd,ap,-be,-aq,bc,as,-ba,-au,ay,aw,-aw,-ay,au, ba,-as,-bc,aq,be,-ap,bd,ar,-bb,-at,az,av,-ax,-ax,av,az,-at,-bb,ar,bd,-ap,be, aq,-bc,-as,ba,au,-ay,-aw,aw,ay,-au,-ba,as,bc,-aq,-be,ap,-bd,-ar,bb,at,-az,- av,ax}

[0346] {bw,-bq,-cc,bk,ci,-bf,ch,bl,-cb,-br,bv,bx,-bp,-cd,bj,cj,-bg,cg,bm,- ca,-bs,bu,by,-bo,-ce,bi,ck,-bh,cf,bn,-bz,-bt,bt,bz,-bn,-cf,bh,-ck,-bi,ce,bo,- by,-bu,bs,ca,-bm,-cg,bg,-cj,-bj,cd,bp,-bx,-bv,br,cb,-bl,-ch,bf,-ci,-bk,cc, bq,-bw}

[0347] {al,-aj,-an,ah,-ao,-ai,am,ak,-ak,-am,ai,ao,-ah,an,aj,-al,-al,aj,an,- ah,ao,ai,-am,-ak,ak,am,-ai,-ao,ah,-an,-aj,al,al,-aj,-an,ah,-ao,-ai,am,ak,- ak,-am,ai,ao,-ah,an,aj,-al,-al,aj,an,-ah,ao,ai,-am,-ak,ak,am,-ai,-ao,ah,-an,-

aj,al}

[0348] {bx,-bn,-ch,bg,-ce,-bq,bu,ca,-bk,-ck,bj,-cb,-bt,br,cd,-bh,ci,bm,-by,-bw,bo,cg,-bf,cf,bp,-bv,-bz,bl,cj,-bi,cc,bs,-bs,-cc,bi,-cj,-bl,bz,bv,-bp,-cf,bf,-cg,-bo,bw,by,-bm,-ci,bh,-cd,-br,bt,cb,-bj,ck,bk,-ca,-bu,bq,ce,-bg,ch,bn,-bx}

[0349] {ay,-as,-be,ar,-az,-ax,at,bd,-aq,ba,aw,-au,-bc,ap,-bb,-av,av,bb,-ap,bc,au,-aw,-ba,aq,-bd,-at,ax,az,-ar,be,as,-ay,-ay,as,be,-ar,az,ax,-at,-bd,aq,-ba,-aw,au,bc,-ap,bb,av,-av,-bb,ap,-bc,-au,aw,ba,-aq,bd,at,-ax,-az,ar,-be,-as,ay}

[0350] {by,-bk,cj,bn,-bv,-cb,bh,-cg,-bq,bs,ce,-bf,cd,bt,-bp,-ch,bi,-ca,-bw,bm,ck,-bl,bx,bz,-bj,ci,bo,-bu,-cc,bg,-cf,-br,br,cf,-bg,cc,bu,-bo,-ci,bj,-bz,-bx,bl,-ck,-bm,bw,ca,-bi,ch,bp,-bt,-cd,bf,-ce,-bs,bq,cg,-bh,cb,bv,-bn,-cj,bk,-by}

[0351] {af,-ad,ag,ae,-ae,-ag,ad,-af,-af,ad,-ag,-ae,ae,ag,-ad,af,af,-ad,ag,ae,-ae,-ag,ad,-af,-af,ad,-ag,-ae,ae,ag,-ad,af,af,-ad,ag,ae,-ae,-ag,ad,-af,-af,ad,-ag,-ae,ae,ag,-ad,af,af,-ad,ag,ae,-ae,-ag,ad,-af,-af,ad,-ag,-ae,ae,ag,-ad,af}

[0352] {bz,-bh,ce,bu,-bm,cj,bp,-br,-ch,bk,-bw,-cc,bf,-cb,-bx,bj,-cg,-bs,bo,ck,-bn,bt,cf,-bi,by,ca,-bg,cd,bv,-bl,ci,bq,-bq,-ci,bl,-bv,-cd,bg,-ca,-by,bi,-cf,-bt,bn,-ck,-bo,bs,cg,-bj,bx,cb,-bf,cc,bw,-bk,ch,br,-bp,-cj,bm,-bu,-ce,bh,-bz}

[0353] {az,-ap,ba,ay,-aq,bb,ax,-ar,bc,aw,-as,bd,av,-at,be,au,-au,-be,at,-av,-bd,as,-aw,-bc,ar,-ax,-bb,aq,-ay,-ba,ap,-az,-az,ap,-ba,-ay,aq,-bb,-ax,ar,-bc,-aw,as,-bd,-av,at,-be,-au,au,be,-at,av,bd,-as,aw,bc,-ar,ax,bb,-aq,ay,ba,-ap,az}

[0354] {ca,-bf,bz,cb,-bg,by,cc,-bh,bx,cd,-bi,bw,ce,-bj,bv,cf,-bk,bu,cg,-bl,bt,ch,-bm,bs,ci,-bn,br,cj,-bo,bq,ck,-bp,bp,-ck,-bq,bo,-cj,-br,bn,-ci,-bs,bm,-ch,-bt,bl,-cg,-bu,bk,-cf,-bv,bj,-ce,-bw,bi,-cd,-bx,bh,-cc,-by,bg,-cb,-bz,bf,-ca}

[0355] {am,-ah,al,an,-ai,ak,ao,-aj,aj,-ao,-ak,ai,-an,-al,ah,-am,-am,ah,-al,-an,ai,-ak,-ao,aj,-aj,ao,ak,-ai,an,al,-ah,am,am,-ah,al,an,-ai,ak,ao,-aj,aj,-ao,-ak,ai,-an,-al,ah,-am,-am,ah,-al,-an,ai,-ak,-ao,aj,-aj,ao,ak,-ai,an,al,-ah,am}

[0356] {cb,-bi,bu,ci,-bp,bn,-cg,-bw,bg,-bz,-cd,bk,-bs,-ck,br,-bl,ce,by,-bf,bx,cf,-bm,bq,-cj,-bt,bj,-cc,-ca,bh,-bv,-ch,bo,-bo,ch,bv,-bh,ca,cc,-bj,bt,cj,-bq,bm,-cf,-bx,bf,-by,-ce,bl,-br,ck,bs,-bk,cd,bz,-bg,bw,cg,-bn,bp,-ci,-bu,bi,-cb}

[0357] {ba,-ar,av,-be,-aw,aq,-az,-bb,as,-au,bd,ax,-ap,ay,bc,-at,at,-bc,-ay,ap,-ax,-bd,au,-as,bb,az,-aq,aw,be,-av,ar,-ba,-ba,ar,-av,be,aw,-aq,az,bb,-as,

au, -bd, -ax, ap, -ay, -bc, at, -at, bc, ay, -ap, ax, bd, -au, as, -bb, -az, aq, -aw, -be, av, -ar, ba}

[0358] {cc, -bl, bp, -cg, -by, bh, -bt, ck, bu, -bg, bx, ch, -bq, bk, -cb, -cd, bm, -bo, cf, bz, -bi, bs, -cj, -bv, bf, -bw, -ci, br, -bj, ca, ce, -bn, bn, -ce, -ca, bj, -br, ci, bw, -bf, bv, cj, -bs, bi, -bz, -cf, bo, -bm, cd, cb, -bk, bq, -ch, -bx, bg, -bu, -ck, bt, -bh, by, cg, -bp, bl, -cc}

[0359] {ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac}

[0360] {cd, -bo, bk, -bz, -ch, bs, -bg, bv, -ck, -bw, bh, -br, cg, ca, -bl, bn, -cc, -ce, bp, -bj, by, ci, -bt, bf, -bu, cj, bx, -bi, bq, -cf, -cb, bm, -bm, cb, cf, -bq, bi, -bx, -cj, bu, -bf, bt, -ci, -by, bj, -bp, ce, cc, -bn, bl, -ca, -cg, br, -bh, bw, ck, -bv, bg, -bs, ch, bz, -bk, bo, -cd}

[0361] {bb, -au, aq, -ax, be, ay, -ar, at, -ba, -bc, av, -ap, aw, -bd, -az, as, -as, az, bd, -aw, ap, -av, bc, ba, -at, ar, -ay, -be, ax, -aq, au, -bb, -bb, au, -aq, ax, -be, -ay, ar, -at, ba, bc, -av, ap, -aw, bd, az, -as, as, -az, -bd, aw, -ap, av, -bc, -ba, at, -ar, ay, be, -ax, aq, -au, bb}

[0362] {ce, -br, bf, -bs, cf, cd, -bq, bg, -bt, cg, cc, -bp, bh, -bu, ch, cb, -bo, bi, -bv, ci, ca, -bn, bj, -bw, cj, bz, -bm, bk, -bx, ck, by, -bl, bl, -by, -ck, bx, -bk, bm, -bz, -cj, bw, -bj, bn, -ca, -ci, bv, -bi, bo, -cb, -ch, bu, -bh, bp, -cc, -cg, bt, -bg, bq, -cd, -cf, bs, -bf, br, -ce}

[0363] {an, -ak, ah, -aj, am, ao, -al, ai, -ai, al, -ao, -am, aj, -ah, ak, -an, -an, ak, -ah, aj, -am, -ao, al, -ai, ai, -al, ao, am, -aj, ah, -ak, an, an, -ak, ah, -aj, am, ao, -al, ai, -ai, al, -ao, -am, aj, -ah, ak, -an, -an, ak, -ah, aj, -am, -ao, al, -ai, ai, -al, ao, am, -aj, ah, -ak, an}

[0364] {cf, -bu, bj, -bl, bw, -ch, -cd, bs, -bh, bn, -by, cj, cb, -bq, bf, -bp, ca, ck, -bz, bo, -bg, br, -cc, -ci, bx, -bm, bi, -bt, ce, cg, -bv, bk, -bk, bv, -cg, -ce, bt, -bi, bm, -bx, ci, cc, -br, bg, -bo, bz, -ck, -ca, bp, -bf, bq, -cb, -cj, by, -bn, bh, -bs, cd, ch, -bw, bl, -bj, bu, -cf}

[0365] {bc, -ax, as, -aq, av, -ba, -be, az, -au, ap, -at, ay, -bd, -bb, aw, -ar, ar, -aw, bb, bd, -ay, at, -ap, au, -az, be, ba, -av, aq, -as, ax, -bc, -bc, ax, -as, aq, -av, ba, be, -az, au, -ap, at, -ay, bd, bb, -aw, ar, -ar, aw, -bb, -bd, ay, -at, ap, -au, az, -be, -ba, av, -aq, as, -ax, bc}

[0366] {cg, -bx, bo, -bf, bn, -bw, cf, ch, -by, bp, -bg, bm, -bv, ce, ci, -bz, bq, -bh, bl, -bu, cd, cj, -ca, br, -bi, bk, -bt, cc, ck, -cb, bs, -bj, bj, -bs, cb, -ck, -cc, bt, -bk, bi, -br, ca, -cj, -cd, bu, -bl, bh, -bq, bz, -ci, -ce, bv, -bm, bg, -bp, by, -ch, -cf, bw, -bn, bf, -bo, bx, -cg}

[0367] {ag, -af, ae, -ad, ad, -ae, af, -ag, -ag, af, -ae, ad, -ad, ae, -af, ag, ag, -af, ae, -

ad,ad,-ae,af,-ag,-ag,af,-ae,ad,-ad,ae,-af,ag,ag,-af,ae,-ad,ad,-ae,af,-ag,-ag,af,-ae,ad,-ad,ae,-af,ag,ag,-af,ae,-ad,ad,-ae,af,-ag,-ag,af,-ae,ad,-ad,ae,-af,ag}

[0368] {ch,-ca,bt,-bm,bf,-bl,bs,-bz,cg,ci,-cb,bu,-bn,bg,-bk,br,-by,cf,cj,-cc,bv,-bo,bh,-bj,bq,-bx,ce,ck,-cd,bw,-bp,bi,-bi,bp,-bw,cd,-ck,-ce,bx,-bq,bj,-bh,bo,-bv,cc,-cj,-cf,by,-br,bk,-bg,bn,-bu,cb,-ci,-cg,bz,-bs,bl,-bf,bm,-bt,ca,-ch}

[0369] {bd,-ba,ax,-au,ar,-ap,as,-av,ay,-bb,be,bc,-az,aw,-at,aq,-aq,at,-aw,az,-bc,-be,bb,-ay,av,-as,ap,-ar,au,-ax,ba,-bd,-bd,ba,-ax,au,-ar,ap,-as,av,-ay,bb,-be,-bc,az,-aw,at,-aq,aq,-at,aw,-az,bc,be,-bb,ay,-av,as,-ap,ar,-au,ax,-ba,bd}

[0370] {ci,-cd,by,-bt,bo,-bj,bf,-bk,bp,-bu,bz,-ce,cj,ch,-cc,bx,-bs,bn,-bi,bg,-bl,bq,-bv,ca,-cf,ck,cg,-cb,bw,-br,bm,-bh,bh,-bm,br,-bw,cb,-cg,-ck,cf,-ca,bv,-bq,bl,-bg,bi,-bn,bs,-bx,cc,-ch,-cj,ce,-bz,bu,-bp,bk,-bf,bj,-bo,bt,-by,cd,-ci}

[0371] {ao,-an,am,-al,ak,-aj,ai,-ah,ah,-ai,aj,-ak,al,-am,an,-ao,-ao,an,-am,al,-ak,aj,-ai,ah,-ah,ai,-aj,ak,-al,am,-an,ao,ao,-an,am,-al,ak,-aj,ai,-ah,ah,-ai,aj,-ak,al,-am,an,-ao,-ao,an,-am,al,-ak,aj,-ai,ah,-ah,ai,-aj,ak,-al,am,-an,ao}

[0372] {cj,-cg,cd,-ca,bx,-bu,br,-bo,bl,-bi,bf,-bh,bk,-bn,bq,-bt,bw,-bz,cc,-cf,ci,ck,-ch,ce,-cb,by,-bv,bs,-bp,bm,-bj,bg,-bg,bj,-bm,bp,-bs,bv,-by,cb,-ce,ch,-ck,-ci,cf,-cc,bz,-bw,bt,-bq,bn,-bk,bh,-bf,bi,-bl,bo,-br,bu,-bx,ca,-cd,cg,-cj}

[0373] {be,-bd,bc,-bb,ba,-az,ay,-ax,aw,-av,au,-at,as,-ar,aq,-ap,ap,-aq,ar,-as,at,-au,av,-aw,ax,-ay,az,-ba,bb,-bc,bd,-be,-be,bd,-bc,bb,-ba,az,-ay,ax,-aw,av,-au,at,-as,ar,-aq,ap,-ap,aq,-ar,as,-at,au,-av,aw,-ax,ay,-az,ba,-bb,bc,-bd,be}

[0374] {ck,-cj,ci,-ch,cg,-cf,ce,-cd,cc,-cb,ca,-bz,by,-bx,bw,-bv,bu,-bt,bs,-br,bq,-bp,bo,-bn,bm,-bl,bk,-bj,bi,-bh,bg,-bf,bf,-bg,bh,-bi,bj,-bk,bl,-bm,bn,-bo,bp,-bq,br,-bs,bt,-bu,bv,-bw,bx,-by,bz,-ca,cb,-cc,cd,-ce,cf,-cg,ch,-ci,cj,-ck}

[0375] }

[0376] 其中,

[0377] {aa,ab,ac,ad,ae,af,ag,ah,ai,aj,ak,al,am,an,ao,ap,aq,ar,as,at,au,av,aw,ax,ay,az,ba,bb,bc,bd,be,bf,bg,bh,bi,bj,bk,bl,bm,bn,bo,bp,bq,br,bs,bt,bu,bv,bw,bx,by,bz,ca,cb,cc,cd,ce,cf,cg,ch,ci,cj,ck} =

[0378] {64,83,36,89,75,50,18,90,87,80,70,57,43,25,9,90,90,88,85,82,78,73,67,61,54,46,38,31,22,13,4,91,90,90,90,88,87,86,84,83,81,79,77,73,71,69,65,62,59,56,52,48,44,41,37,33,28,24,20,15,11,7,2}

- [0379] 附录III
- [0380] 4-点DST-7
- [0381] {a,b,c,d}
- [0382] {c,c,0,-c}
- [0383] {d,-a,-c,b}
- [0384] {b,-d,c,-a}
- [0385] 其中, {a,b,c,d} = {29,55,74,84}
- [0386] 8-点DST-7:
- [0387] {a,b,c,d,e,f,g,h,}
- [0388] {c,f,h,e,b,-a,-d,-g,}
- [0389] {e,g,b,-c,-h,-d,a,f,}
- [0390] {g,c,-d,-f,a,h,b,-e,}
- [0391] {h,-a,-g,b,f,-c,-e,d,}
- [0392] {f,-e,-a,g,-d,-b,h,-c,}
- [0393] {d,-h,e,-a,-c,g,-f,b,}
- [0394] {b,-d,f,-h,g,-e,c,-a,}
- [0395] 其中, {a,b,c,d,e,f,g,h} = {17,32,46,60,71,78,85,86}
- [0396] 16-点DST-7
- [0397] {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,}
- [0398] {c,f,i,l,o,o,l,i,f,c,0,-c,-f,-i,-l,-o,}
- [0399] {e,j,o,m,h,c,-b,-g,-l,-p,-k,-f,-a,d,i,n,}
- [0400] {g,n,l,e,-b,-i,-p,-j,-c,d,k,o,h,a,-f,-m,}
- [0401] {i,o,f,-c,-l,-l,-c,f,o,i,0,-i,-o,-f,c,l,}
- [0402] {k,k,0,-k,-k,0,k,k,0,-k,-k,0,k,k,0,-k,}
- [0403] {m,g,-f,-n,-a,l,h,-e,-o,-b,k,i,-d,-p,-c,j,}
- [0404] {o,c,-l,-f,i,i,-f,-l,c,o,0,-o,-c,l,f,-i,}
- [0405] {p,-a,-o,b,n,-c,-m,d,l,-e,-k,f,j,-g,-i,h,}
- [0406] {n,-e,-i,j,d,-o,a,m,-f,-h,k,c,-p,b,l,-g,}
- [0407] {l,-i,-c,o,-f,-f,o,-c,-i,l,0,-l,i,c,-o,f,}
- [0408] {j,-m,c,g,-p,f,d,-n,i,a,-k,l,-b,-h,o,-e,}
- [0409] {h,-p,i,-a,-g,o,-j,b,f,-n,k,-c,-e,m,-l,d,}
- [0410] {f,-l,o,-i,c,c,-i,o,-l,f,0,-f,l,-o,i,-c,}
- [0411] {d,-h,l,-p,m,-i,e,-a,-c,g,-k,o,-n,j,-f,b,}
- [0412] {b,-d,f,-h,j,-l,n,-p,o,-m,k,-i,g,-e,c,-a,}
- [0413] 其中, {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p} = {9,17,25,33,41,49,56,62,66,72,77,81,83,87,89,90}
- [0414] 32-点DST-7
- [0415] {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,A,B,C,D,E,F,}
- [0416] {c,f,i,l,o,r,u,x,A,D,F,C,z,w,t,q,n,k,h,e,b,-a,-d,-g,-j,-m,-p,-s,-v,-

- y, -B, -E,}
- [0417] {e, j, o, t, y, D, D, y, t, o, j, e, 0, -e, -j, -o, -t, -y, -D, -D, -y, -t, -o, -j, -e, 0, e, j, o, t, y, D,}
- [0418] {g, n, u, B, D, w, p, i, b, -e, -l, -s, -z, -F, -y, -r, -k, -d, c, j, q, x, E, A, t, m, f, -a, -h, -o, -v, -C,}
- [0419] {i, r, A, C, t, k, b, -g, -p, -y, -E, -v, -m, -d, e, n, w, F, x, o, f, -c, -l, -u, -D, -z, -q, -h, a, j, s, B,}
- [0420] {k, v, F, u, j, -a, -l, -w, -E, -t, -i, b, m, x, D, s, h, -c, -n, -y, -C, -r, -g, d, o, z, B, q, f, -e, -p, -A,}
- [0421] {m, z, z, m, 0, -m, -z, -z, -m, 0, m, z, z, m, 0, -m, -z, -z, -m, 0, m, z, z, m, 0, -m, -z, -z, -m, 0, m, z,}
- [0422] {o, D, t, e, -j, -y, -y, -j, e, t, D, o, 0, -o, -D, -t, -e, j, y, y, j, -e, -t, -D, -o, 0, o, D, t, e, -j, -y,}
- [0423] {q, E, n, -c, -t, -B, -k, f, w, y, h, -i, -z, -v, -e, l, C, s, b, -o, -F, -p, a, r, D, m, -d, -u, -A, -j, g, x,}
- [0424] {s, A, h, -k, -D, -p, c, v, x, e, -n, -F, -m, f, y, u, b, -q, -C, -j, i, B, r, -a, -t, -z, -g, l, E, o, -d, -w,}
- [0425] {u, w, b, -s, -y, -d, q, A, f, -o, -C, -h, m, E, j, -k, -F, -l, i, D, n, -g, -B, -p, e, z, r, -c, -x, -t, a, v,}
- [0426] {w, s, -d, -A, -o, h, E, k, -l, -D, -g, p, z, c, -t, -v, a, x, r, -e, -B, -n, i, F, j, -m, -C, -f, q, y, b, -u,}
- [0427] {y, o, -j, -D, -e, t, t, -e, -D, -j, o, y, 0, -y, -o, j, D, e, -t, -t, e, D, j, -o, -y, 0, y, o, -j, -D, -e, t,}
- [0428] {A, k, -p, -v, e, F, f, -u, -q, j, B, a, -z, -l, o, w, -d, -E, -g, t, r, -i, -C, -b, y, m, -n, -x, c, D, h, -s,}
- [0429] {C, g, -v, -n, o, u, -h, -B, a, D, f, -w, -m, p, t, -i, -A, b, E, e, -x, -l, q, s, -j, -z, c, F, d, -y, -k, r,}
- [0430] {E, c, -B, -f, y, i, -v, -l, s, o, -p, -r, m, u, -j, -x, g, A, -d, -D, a, F, b, -C, -e, z, h, -w, -k, t, n, -q,}
- [0431] {F, -a, -E, b, D, -c, -C, d, B, -e, -A, f, z, -g, -y, h, x, -i, -w, j, v, -k, -u, l, t, -m, -s, n, r, -o, -q, p,}
- [0432] {D, -e, -y, j, t, -o, -o, t, j, -y, -e, D, 0, -D, e, y, -j, -t, o, o, -t, -j, y, e, -D, 0, D, -e, -y, j, t, -o,}
- [0433] {B, -i, -s, r, j, -A, -a, C, -h, -t, q, k, -z, -b, D, -g, -u, p, l, -y, -c, E, -f, -v, o, m, -x, -d, F, -e, -w, n,}
- [0434] {z, -m, -m, z, 0, -z, m, m, -z, 0, z, -m, -m, z, 0, -z, m, m, -z, 0, z, -m, -m, z, 0, -z, m, m, -z, 0, z, -m,}
- [0435] {x, -q, -g, E, -j, -n, A, -c, -u, t, d, -B, m, k, -D, f, r, -w, -a, y, -p, -h, F, -i, -o, z, -b, -v, s, e, -C, l,}

- [0436] {v,-u,-a,w,-t,-b,x,-s,-c,y,-r,-d,z,-q,-e,A,-p,-f,B,-o,-g,C,-n,-h,D,-m,-i,E,-l,-j,F,-k,}
- [0437] {t,-y,e,o,-D,j,j,-D,o,e,-y,t,0,-t,y,-e,-o,D,-j,-j,D,-o,-e,y,-t,0,t,-y,e,o,-D,j,}
- [0438] {r,-C,k,g,-y,v,-d,-n,F,-o,-c,u,-z,h,j,-B,s,-a,-q,D,-l,-f,x,-w,e,m,-E,p,b,-t,A,-i,}
- [0439] {p,-F,q,-a,-o,E,-r,b,n,-D,s,-c,-m,C,-t,d,l,-B,u,-e,-k,A,-v,f,j,-z,w,-g,-i,y,-x,h,}
- [0440] {n,-B,w,-i,-e,s,-F,r,-d,-j,x,-A,m,a,-o,C,-v,h,f,-t,E,-q,c,k,-y,z,-l,-b,p,-D,u,-g,}
- [0441] {l,-x,C,-q,e,g,-s,E,-v,j,b,-n,z,-A,o,-c,-i,u,-F,t,-h,-d,p,-B,y,-m,a,k,-w,D,-r,f,}
- [0442] {j,-t,D,-y,o,-e,-e,o,-y,D,-t,j,0,-j,t,-D,y,-o,e,e,-o,y,-D,t,-j,0,j,-t,D,-y,o,-e,}
- [0443] {h,-p,x,-F,y,-q,i,-a,-g,o,-w,E,-z,r,-j,b,f,-n,v,-D,A,-s,k,-c,-e,m,-u,C,-B,t,-l,d,}
- [0444] {f,-l,r,-x,D,-C,w,-q,k,-e,-a,g,-m,s,-y,E,-B,v,-p,j,-d,-b,h,-n,t,-z,F,-A,u,-o,i,-c,}
- [0445] {d,-h,l,-p,t,-x,B,-F,C,-y,u,-q,m,-i,e,-a,-c,g,-k,o,-s,w,-A,E,-D,z,-v,r,-n,j,-f,b,}
- [0446] {b,-d,f,-h,j,-l,n,-p,r,-t,v,-x,z,-B,D,-F,E,-C,A,-y,w,-u,s,-q,o,-m,k,-i,g,-e,c,-a,}
- [0447] 其中, {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,A,B,C,D,E,F} = {4,9,13,17,21,26,30,34,38,42,45,50,53,56,60,63,66,68,72,74,77,78,80,82,84,85,86,88,88,89,90,90}
- [0448] 4-点DCT-8
- [0449] {a,b,c,d,}
- [0450] {b,0,-b,-b,}
- [0451] {c,-b,-d,a,}
- [0452] {d,-b,a,-c,}
- [0453] 其中, {a,b,c,d} = {84,74,55,29}
- [0454] 8-点DCT-8:
- [0455] {a,b,c,d,e,f,g,h,}
- [0456] {b,e,h,-g,-d,-a,-c,-f,}
- [0457] {c,h,-e,-a,-f,g,b,d,}
- [0458] {d,-g,-a,-h,c,e,-f,-b,}
- [0459] {e,-d,-f,c,g,-b,-h,a,}
- [0460] {f,-a,g,e,-b,h,d,-c,}
- [0461] {g,-c,b,-f,-h,d,-a,e,}

- [0462] {h,-f,d,-b,a,-c,e,-g,}
- [0463] 其中, {a,b,c,d,e,f,g,h} = {86,85,78,71,60,46,32,17}
- [0464] 16-点DCT-8
- [0465] {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,}
- [0466] {b,e,h,k,n,0,-n,-k,-h,-e,-b,-b,-e,-h,-k,-n,}
- [0467] {c,h,m,-p,-k,-f,-a,-e,-j,-o,n,i,d,b,g,l,}
- [0468] {d,k,-p,-i,-b,-f,-m,n,g,a,h,o,-l,-e,-c,-j,}
- [0469] {e,n,-k,-b,-h,0,h,b,k,-n,-e,-e,-n,k,b,h,}
- [0470] {f,0,-f,-f,0,f,f,0,-f,-f,0,f,f,0,-f,-f,}
- [0471] {g,-n,-a,-m,h,f,-o,-b,-l,i,e,-p,-c,-k,j,d,}
- [0472] {h,-k,-e,n,b,0,-b,-n,e,k,-h,-h,k,e,-n,-b,}
- [0473] {i,-h,-j,g,k,-f,-l,e,m,-d,-n,c,o,-b,-p,a,}
- [0474] {j,-e,-o,a,-n,-f,i,k,-d,-p,b,-m,-g,h,l,-c,}
- [0475] {k,-b,n,h,-e,0,e,-h,-n,b,-k,-k,b,-n,-h,e,}
- [0476] {l,-b,i,o,-e,f,-p,-h,c,-m,-k,a,-j,-n,d,-g,}
- [0477] {m,-e,d,-l,-n,f,-c,k,o,-g,b,-j,-p,h,-a,i,}
- [0478] {n,-h,b,-e,k,0,-k,e,-b,h,-n,-n,h,-b,e,-k,}
- [0479] {o,-k,g,-c,b,-f,j,-n,-p,l,-h,d,-a,e,-i,m,}
- [0480] {p,-n,l,-j,h,-f,d,-b,a,-c,e,-g,i,-k,m,-o,}
- [0481] 其中, {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p} = {90,89,87,83,81,77,72,66,62,56,49,41,33,25,17,9}
- [0482] 32-点DCT-8
- [0483] {a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,A,B,C,D,E,F,}
- [0484] {b,e,h,k,n,q,t,w,z,C,F,-E,-B,-y,-v,-s,-p,-m,-j,-g,-d,-a,-c,-f,-i,-l,-o,-r,-u,-x,-A,-D,}
- [0485] {c,h,m,r,w,B,0,-B,-w,-r,-m,-h,-c,-c,-h,-m,-r,-w,-B,0,B,w,r,m,h,c,c,h,m,r,w,B,}
- [0486] {d,k,r,y,F,-A,-t,-m,-f,-b,-i,-p,-w,-D,C,v,o,h,a,g,n,u,B,-E,-x,-q,-j,-c,-e,-l,-s,-z,}
- [0487] {e,n,w,F,-y,-p,-g,-c,-l,-u,-D,A,r,i,a,j,s,B,-C,-t,-k,-b,-h,-q,-z,E,v,m,d,f,o,x,}
- [0488] {f,q,B,-A,-p,-e,-g,-r,-C,z,o,d,h,s,D,-y,-n,-c,-i,-t,-E,x,m,b,j,u,F,-w,-l,-a,-k,-v,}
- [0489] {g,t,0,-t,-g,-g,-t,0,t,g,g,t,0,-t,-g,-g,-t,0,t,g,g,t,0,-t,-g,-g,-t,0,t,g,g,t,}
- [0490] {h,w,-B,-m,-c,-r,0,r,c,m,B,-w,-h,-h,-w,B,m,c,r,0,-r,-c,-m,-B,w,h,h,w,-B,-m,-c,-r,}
- [0491] {i,z,-w,-f,-l,-C,t,c,o,F,-q,-a,-r,E,n,d,u,-B,-k,-g,-x,y,h,j,A,-v,-e,-m,-D,s,b,p,}

- [0492] {j,C,-r,-b,-u,z,g,m,F,-o,-e,-x,w,d,p,-E,-l,-h,-A,t,a,s,-B,-i,-k,-D,q,c,v,-y,-f,-n,}
- [0493] {k,F,-m,-i,-D,o,g,B,-q,-e,-z,s,c,x,-u,-a,-v,w,b,t,-y,-d,-r,A,f,p,-C,-h,-n,E,j,l,}
- [0494] {l,-E,-h,-p,A,d,t,-w,-a,-x,s,e,B,-o,-i,-F,k,m,-D,-g,-q,z,c,u,-v,-b,-y,r,f,C,-n,-j,}
- [0495] {m,-B,-c,-w,r,h,0,-h,-r,w,c,B,-m,-m,B,c,w,-r,-h,0,h,r,-w,-c,-B,m,m,-B,-c,-w,r,h,}
- [0496] {n,-y,-c,-D,i,s,-t,-h,E,d,x,-o,-m,z,b,C,-j,-r,u,g,-F,-e,-w,p,l,-A,-a,-B,k,q,-v,-f,}
- [0497] {o,-v,-h,C,a,D,-g,-w,n,p,-u,-i,B,b,E,-f,-x,m,q,-t,-j,A,c,F,-e,-y,l,r,-s,-k,z,d,}
- [0498] {p,-s,-m,v,j,-y,-g,B,d,-E,-a,-F,c,C,-f,-z,i,w,-l,-t,o,q,-r,-n,u,k,-x,-h,A,e,-D,-b,}
- [0499] {q,-p,-r,o,s,-n,-t,m,u,-l,-v,k,w,-j,-x,i,y,-h,-z,g,A,-f,-B,e,C,-d,-D,c,E,-b,-F,a,}
- [0500] {r,-m,-w,h,B,-c,0,c,-B,-h,w,m,-r,-r,m,w,-h,-B,c,0,-c,B,h,-w,-m,r,r,-m,-w,h,B,-c,}
- [0501] {s,-j,-B,a,-C,-i,t,r,-k,-A,b,-D,-h,u,q,-l,-z,c,-E,-g,v,p,-m,-y,d,-F,-f,w,o,-n,-x,e,}
- [0502] {t,-g,0,g,-t,-t,g,0,-g,t,t,-g,0,g,-t,-t,g,0,-g,t,t,-g,0,g,-t,-t,g,0,-g,t,t,-g,}
- [0503] {u,-d,B,n,-k,-E,g,-r,-x,a,-y,-q,h,-F,-j,o,A,-c,v,t,-e,C,m,-l,-D,f,-s,-w,b,-z,-p,i,}
- [0504] {v,-a,w,u,-b,x,t,-c,y,s,-d,z,r,-e,A,q,-f,B,p,-g,C,o,-h,D,n,-i,E,m,-j,F,l,-k,}
- [0505] {w,-c,r,B,-h,m,0,-m,h,-B,-r,c,-w,-w,c,-r,-B,h,-m,0,m,-h,B,r,-c,w,w,-c,r,B,-h,m,}
- [0506] {x,-f,m,-E,-q,b,-t,-B,j,-i,A,u,-c,p,F,-n,e,-w,-y,g,-l,D,r,-a,s,C,-k,h,-z,-v,d,-o,}
- [0507] {y,-i,h,-x,-z,j,-g,w,A,-k,f,-v,-B,l,-e,u,C,-m,d,-t,-D,n,-c,s,E,-o,b,-r,-F,p,-a,q,}
- [0508] {z,-l,c,-q,E,u,-g,h,-v,-D,p,-b,m,-A,-y,k,-d,r,-F,-t,f,-i,w,C,-o,a,-n,B,x,-j,e,-s,}
- [0509] {A,-o,c,-j,v,F,-t,h,-e,q,-C,-y,m,-a,l,-x,-D,r,-f,g,-s,E,w,-k,b,-n,z,B,-p,d,-i,u,}
- [0510] {B,-r,h,-c,m,-w,0,w,-m,c,-h,r,-B,-B,r,-h,c,-m,w,0,-w,m,-c,h,-r,B,B,-r,h,-c,m,-w,}
- [0511] {C,-u,m,-e,d,-l,t,-B,-D,v,-n,f,-c,k,-s,A,E,-w,o,-g,b,-j,r,-z,-F,x,-p,

$h, -a, i, -q, y, \}$

[0512] $\{D, -x, r, -l, f, -a, g, -m, s, -y, E, C, -w, q, -k, e, -b, h, -n, t, -z, F, B, -v, p, -j, d, -c, i, -o, u, -A, \}$

[0513] $\{E, -A, w, -s, o, -k, g, -c, b, -f, j, -n, r, -v, z, -D, -F, B, -x, t, -p, l, -h, d, -a, e, -i, m, -q, u, -y, C, \}$

[0514] $\{F, -D, B, -z, x, -v, t, -r, p, -n, l, -j, h, -f, d, -b, a, -c, e, -g, i, -k, m, -o, q, -s, u, -w, y, -A, C, -E, \}$

[0515] 其中, $\{a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F\} = \{90, 90, 89, 88, 88, 86, 85, 84, 82, 80, 78, 77, 74, 72, 68, 66, 63, 60, 56, 53, 50, 45, 42, 38, 34, 30, 26, 21, 17, 13, 9, 4\}$

	R01	R02	R03	R04	R05	R06	R07	R08	R09
R10	<u>S11</u>	S12	S13	S14					
<u>R20</u>	S21	S22	S23	<u>S24</u>					
R30	S31	<u>S32</u>	S33	S34					
R40	<u>S41</u>	S42	S43	<u>S44</u>					
R50									
<u>R60</u>									
R70									

104

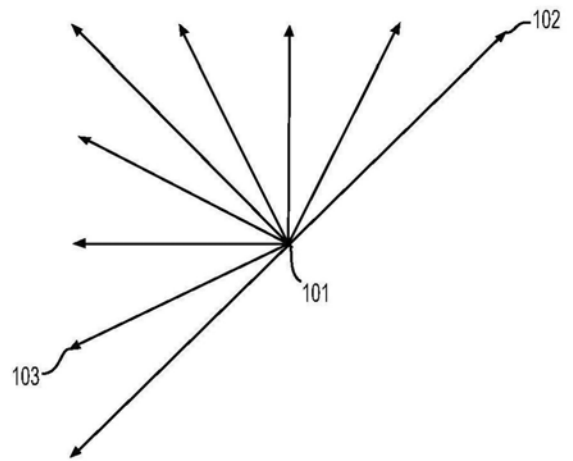
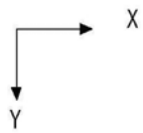


图1A

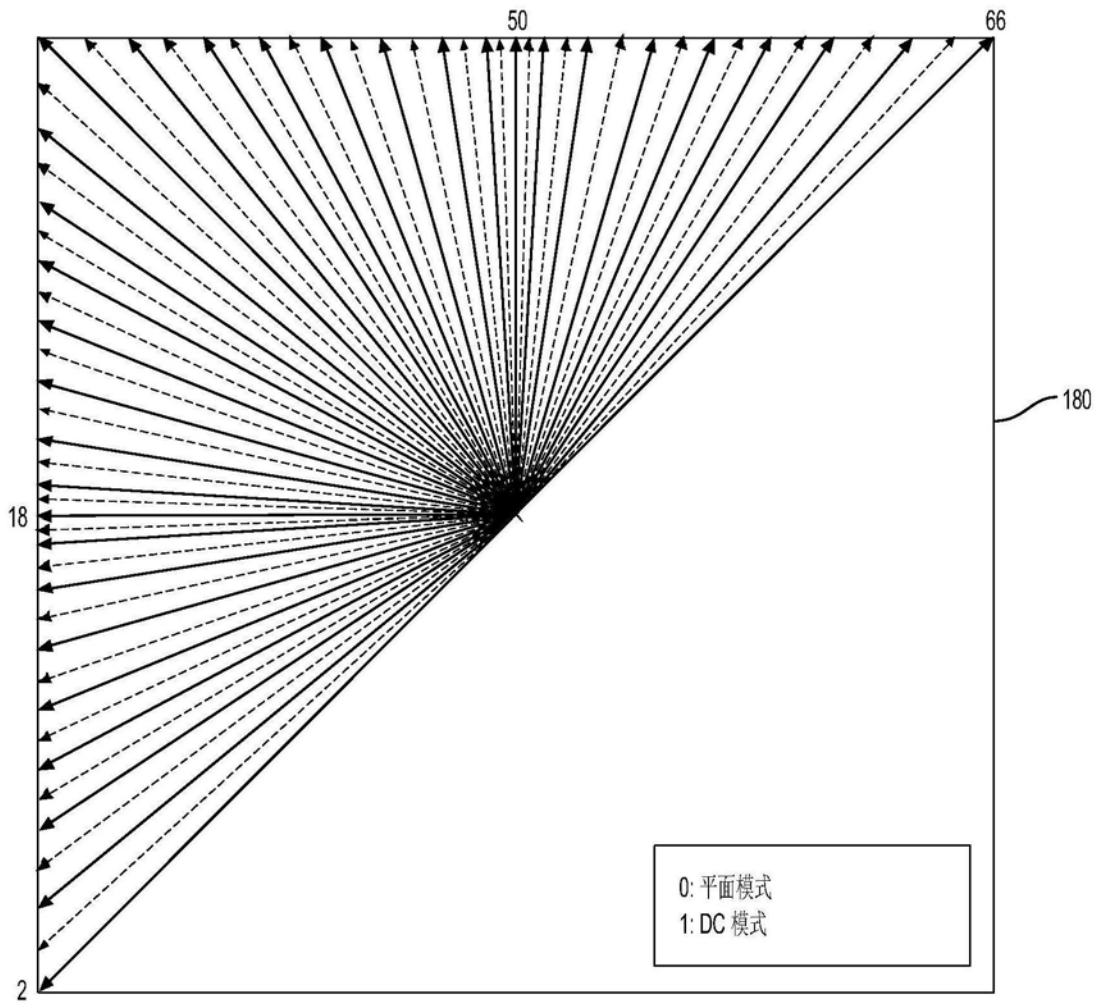


图1B

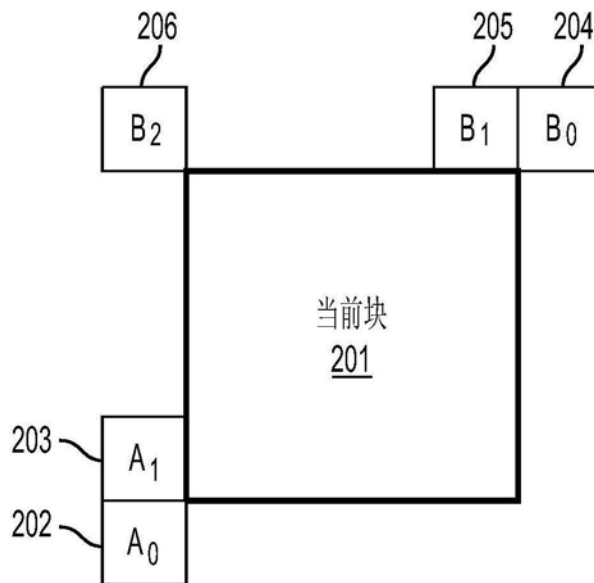


图2 (相关技术)

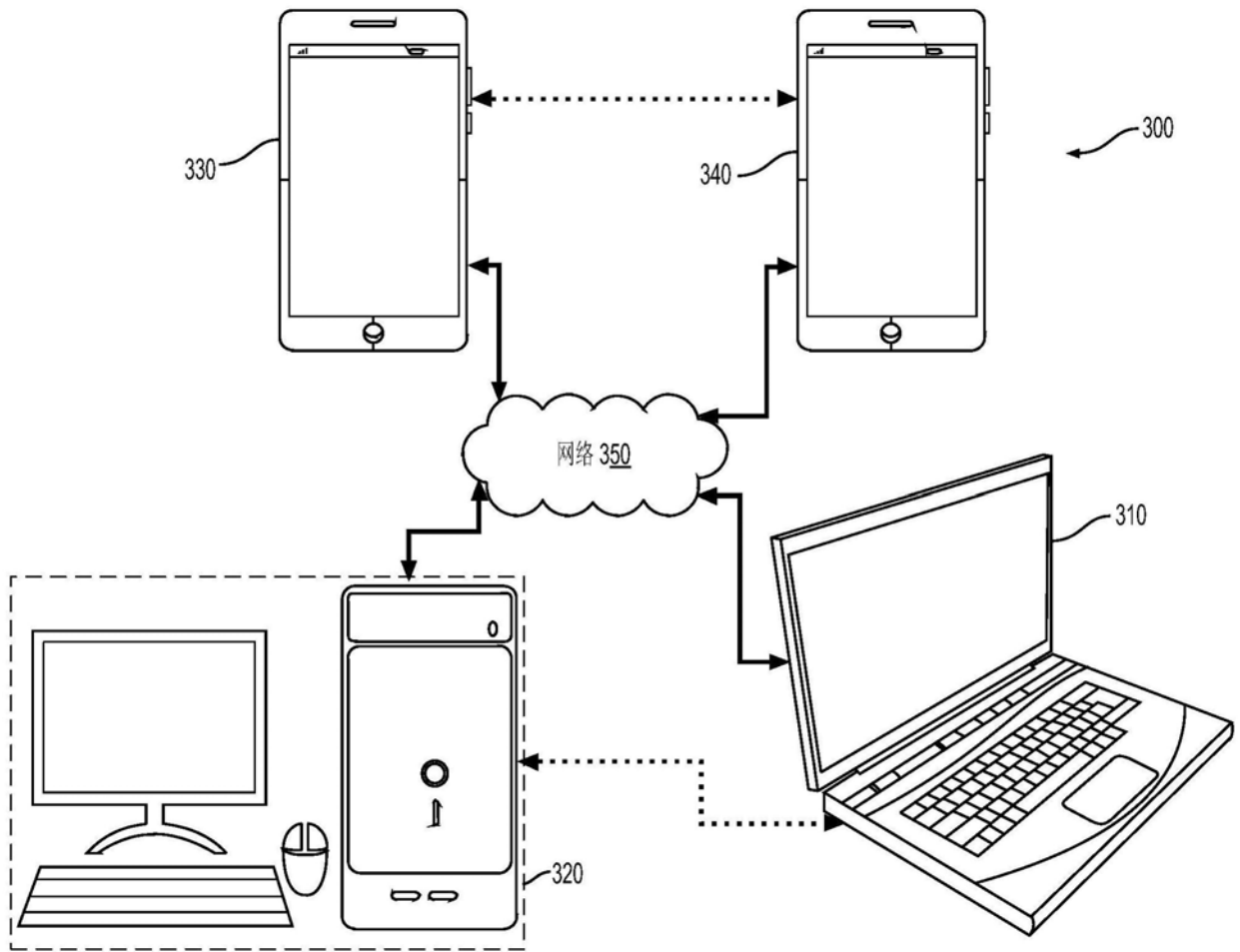


图3

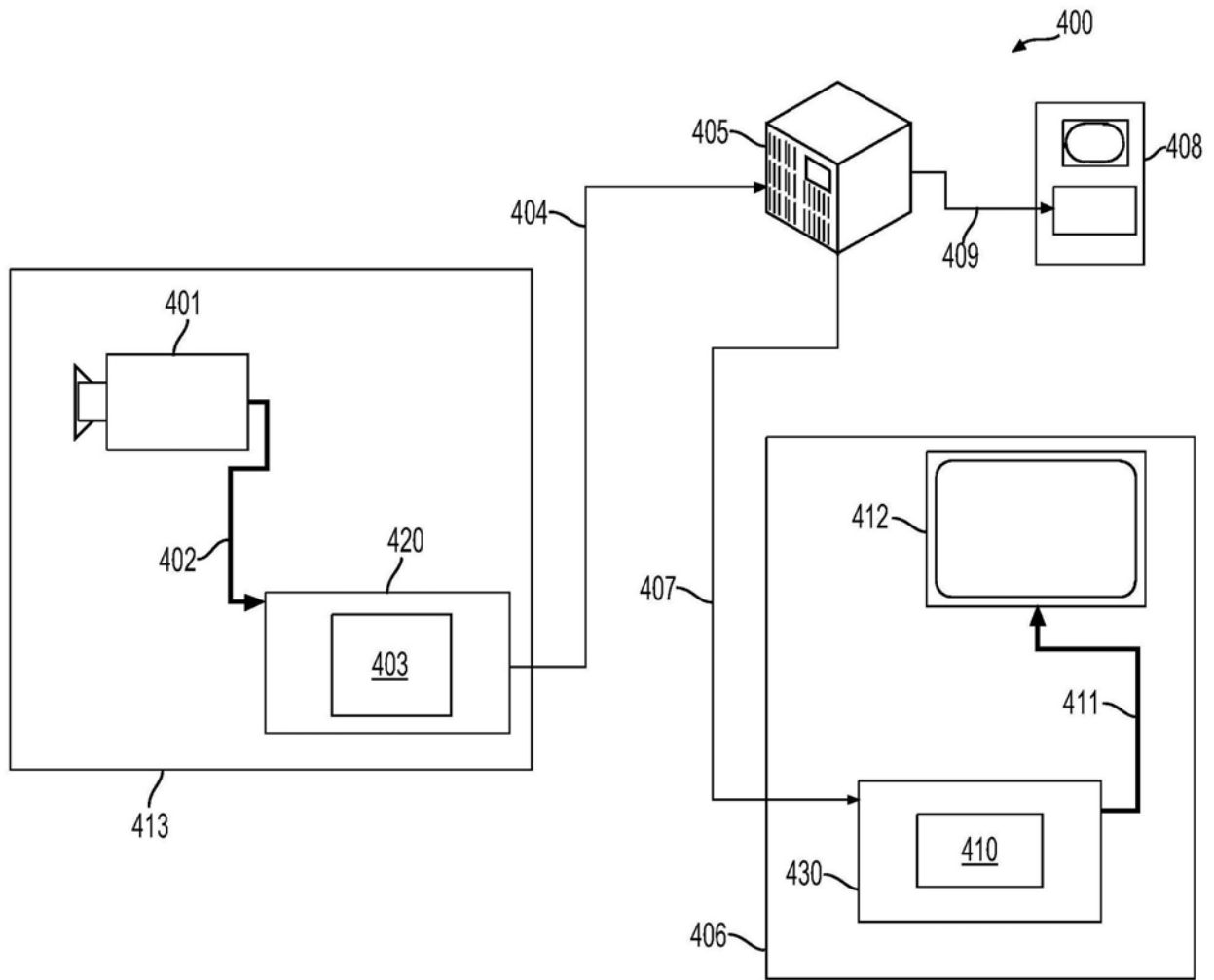


图4

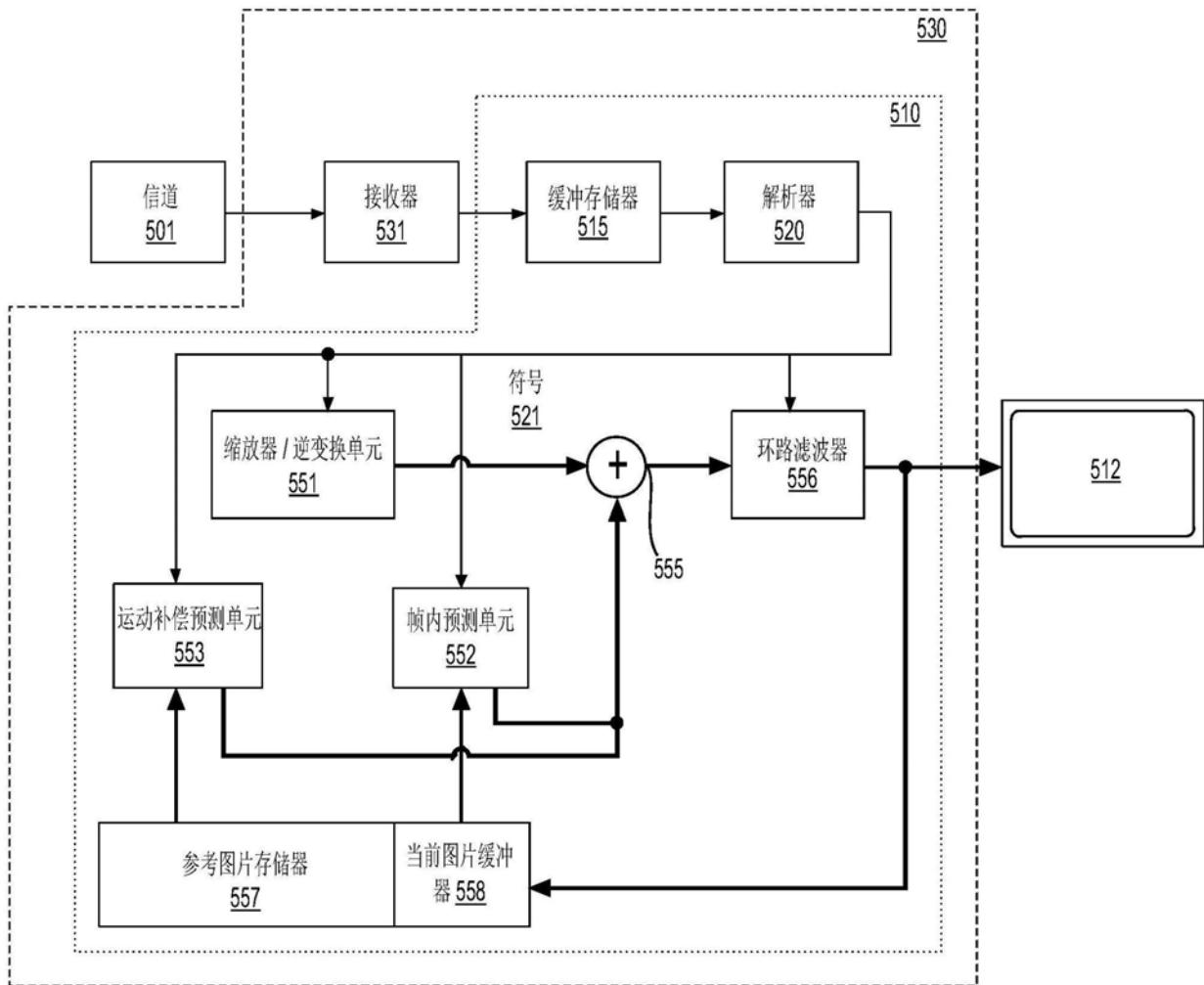


图5

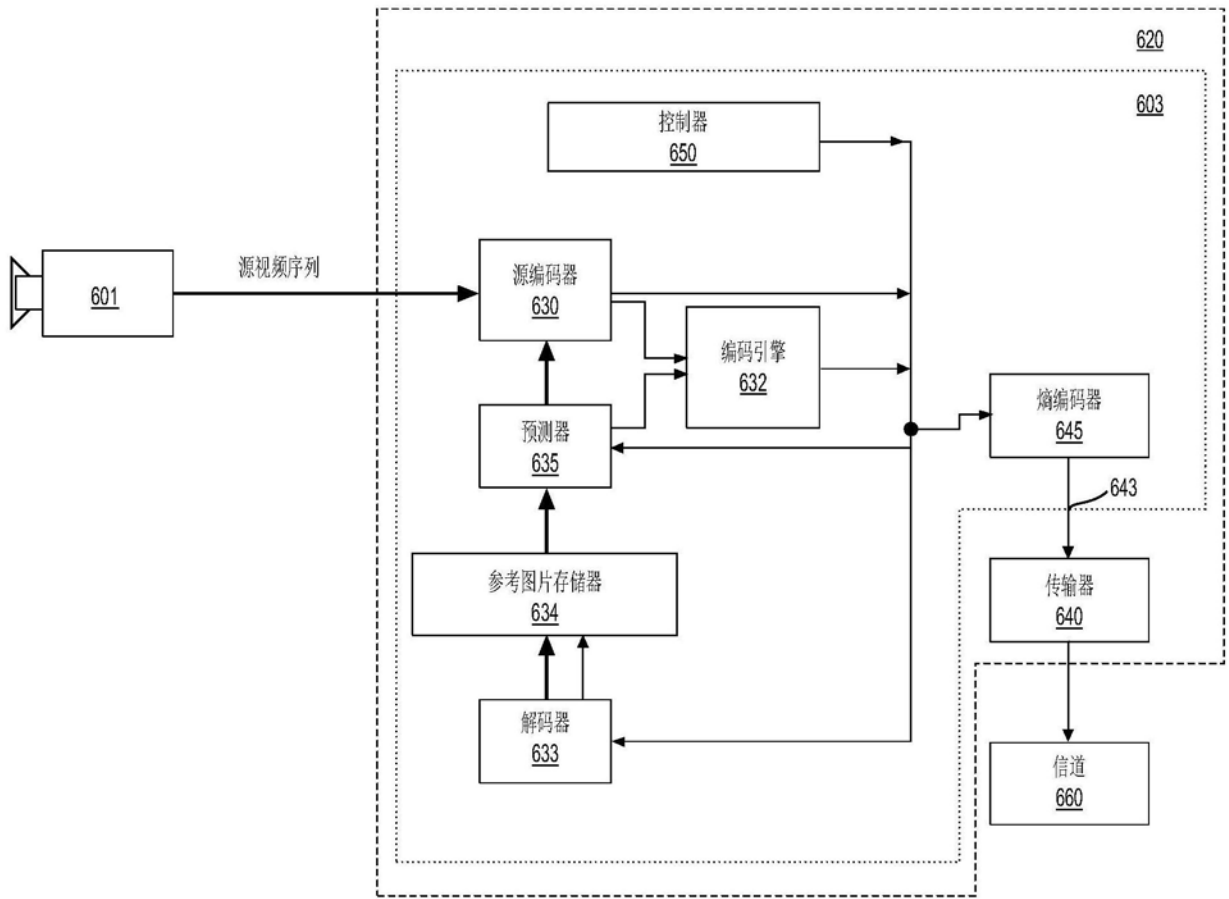


图6

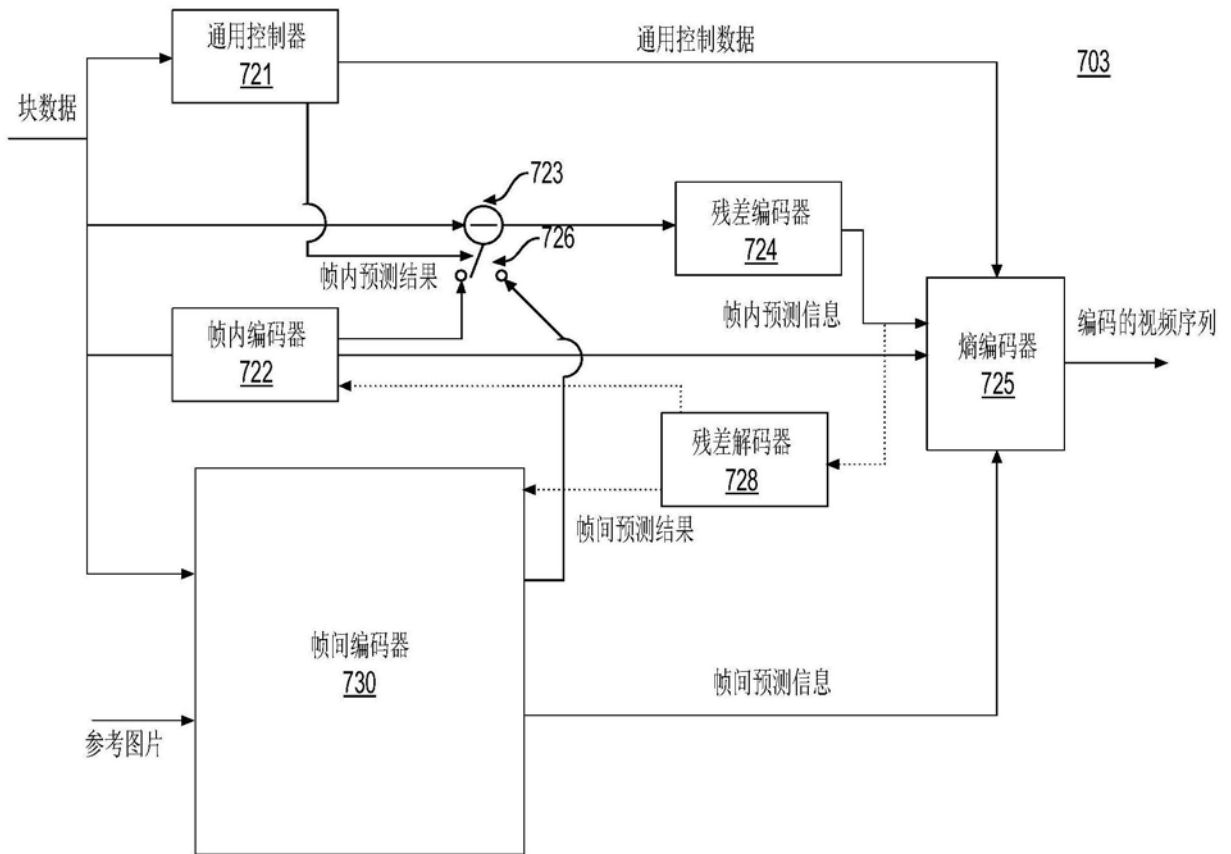


图7

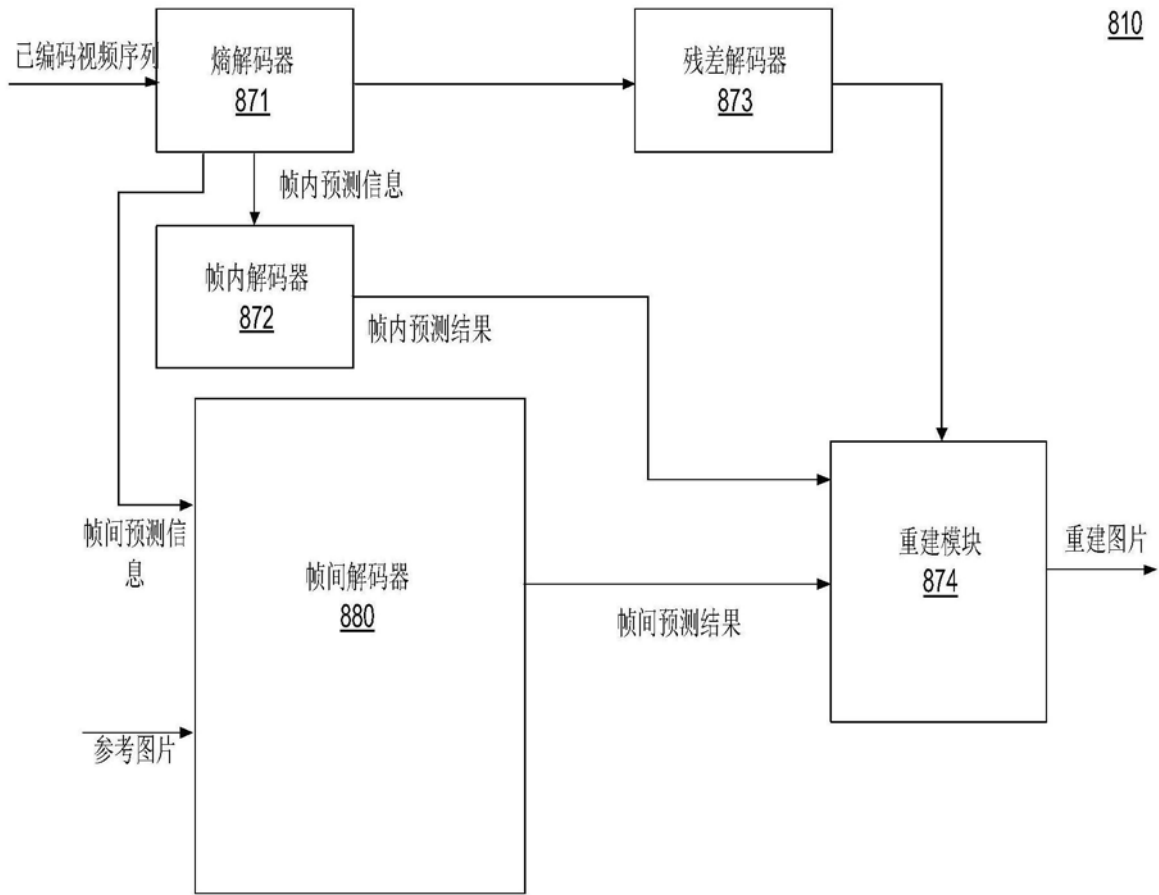


图8

	描述符 (Descriptor)
<code>transform_unit(x0, y0, tbWidth, tbHeight, treeType) {</code>	
<code> if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA)</code>	
<code> tu_cbf_luma[x0][y0]</code>	<code>ae(v)</code>
<code> if(treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA) {</code>	
<code> tu_cbf_cb[x0][y0]</code>	<code>ae(v)</code>
<code> tu_cbf_cr[x0][y0]</code>	<code>ae(v)</code>
<code> }</code>	
<code> if(</code> <code> ((CuPredMode[x0][y0] == MODE_INTRA) && sps_mts_intra_enabled_flag) </code> <code> ((CuPredMode[x0][y0] == MODE_INTER) && sps_mts_inter_enabled_flag))</code> <code> && tu_cbf_luma[x0][y0] && treeType != DUAL_TREE_CHROMA</code> <code> && (tbWidth <= 32) && (tbHeight <= 32))</code>	
<code> cu_mts_flag[x0][y0]</code>	<code>ae(v)</code>
<code> if(tu_cbf_luma[x0][y0])</code>	
<code> residual_coding(x0, y0, log2(tbWidth), log2(tbHeight), 0)</code>	
<code> if(tu_cbf_cb [x0][y0])</code>	
<code> residual_coding(x0, y0, log2(tbWidth/ 2), log2(tbHeight/ 2), 1)</code>	
<code> if(tu_cbf_cr[x0][y0])</code>	
<code> residual_coding(x0, y0, log2(tbWidth/ 2), log2(tbHeight/ 2), 2)</code>	
<code>}</code>	

910

图9

	描述符 (Descriptor)
<code>residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {</code>	
<code> if(transform_skip_enabled_flag && (cIdx != 0 cu_mts_flag[x0][y0] == 0) &&</code> <code> (log2TbWidth <= 2) && (log2TbHeight <= 2))</code>	
<code> transform_skip_flag[x0][y0][cIdx]</code>	<code>ae(v)</code>
<code> last_sig_coeff_x_prefix</code>	<code>ae(v)</code>
<code> last_sig_coeff_y_prefix</code>	<code>ae(v)</code>
<code> if(last_sig_coeff_x_prefix > 3)</code>	
<code> last_sig_coeff_x_suffix</code>	<code>ae(v)</code>
<code> if(last_sig_coeff_y_prefix > 3)</code>	
<code> last_sig_coeff_y_suffix</code>	<code>ae(v)</code>
<code> </code>	

图10A

从图10A继续

if(dep_quant_enabled_flag) {	
QState = startQStateSb	
for(n = numSbCoeff - 1; n >= 0; n- -) {	
xC = (xS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if(sig_coeff_flag[xC][yC])	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
(2 * AbsLevel[xC][yC] - (QState > 1 ? 1 : 0)) *	
(1 - 2 * coeff_sign_flag[n])	
QState = QStateTransTable[QState][par_level_flag[n]]	
} else {	
sumAbsLevel = 0	
for(n = numSbCoeff - 1; n >= 0; n- -) {	
xC = (xS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if(sig_coeff_flag[xC][yC]) {	

图10B

从图10B继续

TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
AbsLevel[xC][yC] * (1 - 2 * coeff_sign_flag[n])	
if(signHidden) {	
sumAbsLevel += AbsLevel[xC][yC]	
if((n == firstSigScanPosSb) && (sumAbsLevel%2 == 1))	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
- TransCoeffLevel[x0][y0][cIdx][xC][yC]	
}	
}	
}	
}	
}	
if(cu_mts_flag[x0][y0] && (cIdx == 0) &&	
! transform_skip_flag[x0][y0][cIdx] &&	
((CuPredMode[x0][y0] == MODE_INTRA && numSigCoeff > 2)	
(CuPredMode[x0][y0] == MODE_INTER)) {	
mts_idx[x0][y0]	ae(v)
}	

1010

图10C

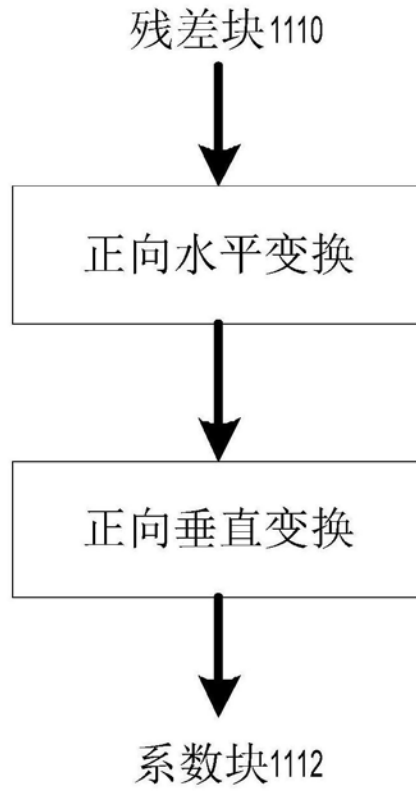


图11A

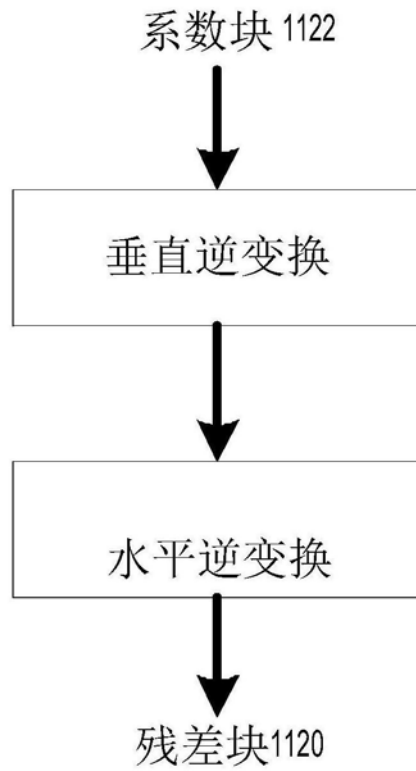


图11B

该过程的输入为:

- 亮度位置 (xTbY, yTbY) 指示当前亮度变换块的左上角的样本相对于当前图片的左上角的亮度样本,
- 变量 nTbW 指示当前变换块的宽度,
- 变量 nTbH 指示当前变换块的高度,
- 变量 cIdx 指定当前块的颜色分量,
- 缩放的变换系数的(nTbW)x(nTbH) 的数组d[x][y], x=0...nTbW-1, y=0...nTbH-1.

此过程的输出是残差样本的(nTbW)x(nTbH) 的数组r[x][y], x=0...nTbW-1, y=0...nTbH-1.

图12A

从图12A继续

变量implicitMtsEnabled 推导如下所示:

- 如果sps_mts_enabled_flag等于1并且以下条件之一为真, 则将 implicitMtsEnabled设置为等于1:
 - IntraSubPartitionsSplitType 不等于 ISP_NO_SPLIT
 - cu_sbt_flag 等于1, 并且 Max(nTbW, nTbH)小于等于32
 - sps_explicit_mts_intra_enabled_flag和 sps_explicit_mts_inter_enabled_flag 均等于0, 以及 CuPredMode[xTbY][yTbY] 等于 MODE_INTRA
- 否则, 将implicitMtsEnabled 设置为 0.

图12B

从图12B继续

指示水平变换核的变量trTypeHor和指示垂直变换核的变量trTypeVer的变量推导如下:

- 如果 cIdx 大于 0, 则将trTypeHor 和trTypeVer 设置为 0.
- 否则, 如果 implicitMtsEnabled等于1, 适用以下内容:
 - 如果IntraSubPartitionsSplitType 不等于 ISP_NO_SPLIT, 根据 intraPredMode指定trTypeHor和trTypeVer
 - 否则, 如果 cu_sbt_flag等于1, 根据cu_sbt_horizontal_flag and cu_sbt_pos_flag指定trTypeHor 和 trTypeVer
 - 否则 (sps_explicit_mts_intra_enabled_flag和sps_explicit_mts_inter_enabled_flag等于0), trTypeHor和trTypeVer 推导如下:

$$\text{trTypeHor} = (\text{nTbW} \geq 4 \ \&\& \ \text{nTbW} \leq 16 \ \&\& \ \text{nTbW} \leq \text{nTbH}) ? 1 : 0$$

$$\text{trTypeVer} = (\text{nTbH} \geq 4 \ \&\& \ \text{nTbH} \leq 16 \ \&\& \ \text{nTbH} \leq \text{nTbW}) ? 1 : 0$$

图12C

从图12C继续

- 否则，根据 $tu_mts_idx[xTbY][yTbY]$ 指定 $trTypeHor$ 和 $trTypeVer$ 变量 $nonZeroW$ 和 $nonZeroH$ 推导如下：

$$nonZeroW = \text{Min}(nTbW, (trTypeHor > 0) ? 16 : 32)$$

$$nonZeroH = \text{Min}(nTbH, (trTypeVer > 0) ? 16 : 32)$$

图12D

从图12D继续

残差样本的 $(nTbW) \times (nTbH)$ 数组 如下推导：

1. 当 $nTbH$ 大于 1时, 通过变换块的高度 $nTbH$, 缩放的变换系数的非零高度 $nonZeroH$, 列表 $d[x][y]$, $y=0..nTbH-1$ 以及设定等于 $trTypeVer$ 的变换类型变量 $trType$ 作为输入, 调用每个列 $x=0..nonZeroW-1$ 的一维变换过程, 将缩放的变换系数的每一列(垂直的) $d[x][y]$, $x=0..nonZeroW-1$, $y=0..nonZeroH-1$, 变换为 $e[x][y]$, $x=0..nonZeroW-1$, $y=0..nTbH-1$ 1210
2. 当 $nTbH$ 和 $nTbW$ 均大于 1时, 中间样本值 $g[x][y]$ 如下推导, $x=0..nonZeroW-1$, $y=0..nTbH-1$:
 $g[x][y] = \text{Clip3}(\text{CoeffMin}, \text{CoeffMax}, (e[x][y] + 64) \gg 7)$
3. 当 $nTbW$ 大于 1时, 通过变换块的宽度 $nTbW$, 生成的数组 $g[x][y]$ 的非零宽度 $nonZeroW$, 列表 $g[x][y]$, $x=0..nonZeroW-1$ 以及设定等于 $trTypeHor$ 的变换类型变量 $trType$ 作为输入, 调用每个行 $y=0..nTbH-1$ 的一维变换过程, 将生成的数组的每一行(水平的) $g[x][y]$, $x=0..nonZeroW-1$, $y=0..nonZeroH-1$, 变换为 $r[x][y]$, $x=0..nTbW-1$, $y=0..nTbH-1$ 1211
 输出为列表 $r[x][y]$, $x=0..nTbW-1$

图12E

	描述符 Descriptor
residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	
if(transform_skip_enabled_flag && (cIdx!= 0 tu_mts_flag[x0][y0] == 0) && (log2TbWidth <= 2) && (log2TbHeight <= 2))	
transform_skip_flag [x0][y0][cIdx]	ae(v)
last_sig_coeff_x_prefix	ae(v)
last_sig_coeff_y_prefix	ae(v)
if(last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
.....	
for(i = lastSubBlock; i >= 0?i?) {	
.....	
if((i < lastSubBlock) && (i > 0)) {	
if(((tu_mts_flag[x0][y0] == 0 cIdx != 0) && (xS << log2SbSize) < 32 && (yS << log2SbSize) < 32) (tu_mts_flag[x0][y0] == 1 && (xS << log2SbSize) < 16 && (yS << log2SbSize) < 16))	
coded_sub_block_flag [xS][yS]	ae(v)
inferSbDcSigCoeffFlag = 1	
}	
.....	
}	
if(tu_mts_flag[x0][y0] && (cIdx == 0))	
mts_idx [x0][y0][cIdx]	ae(v)
}	

1310

图13

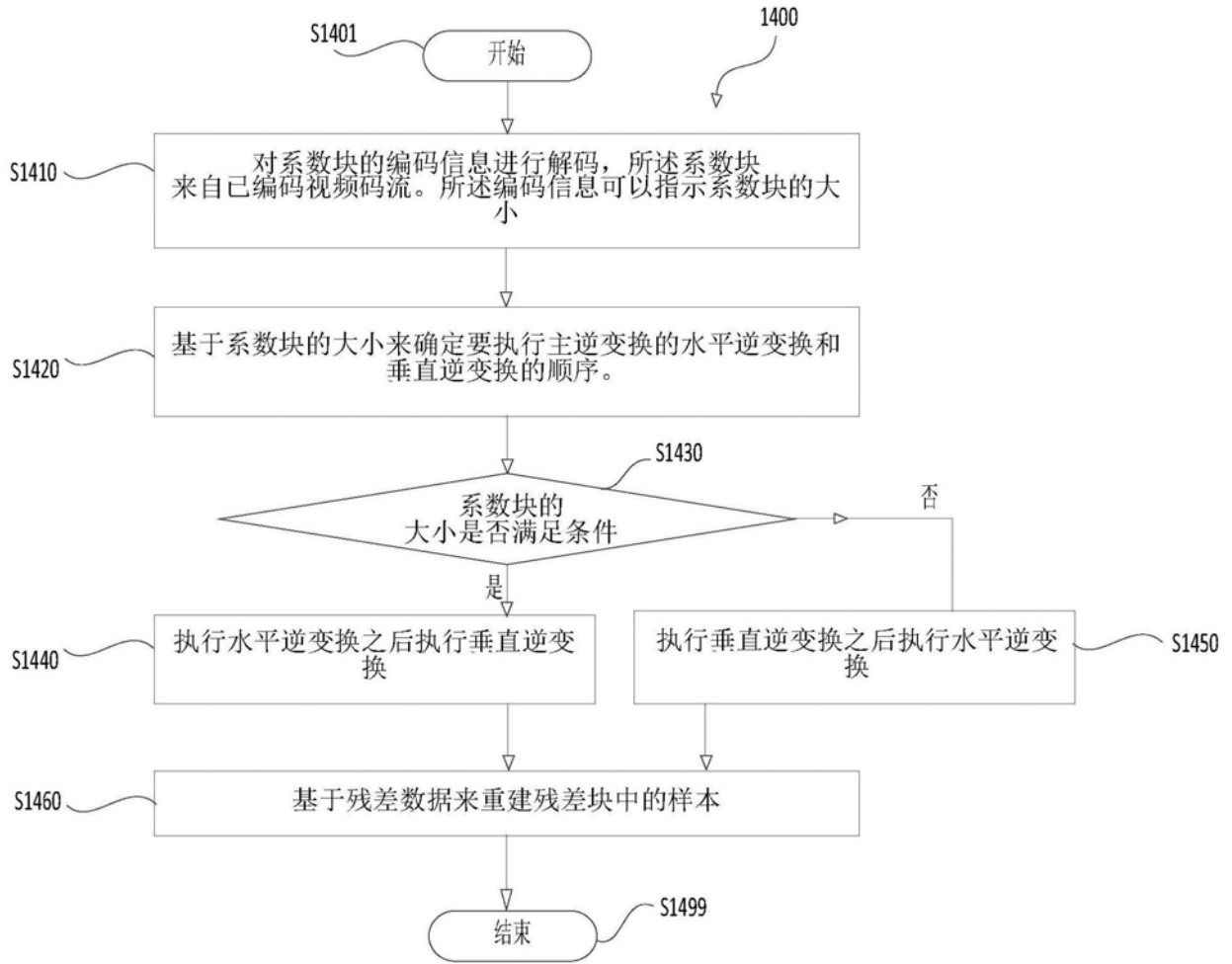


图14

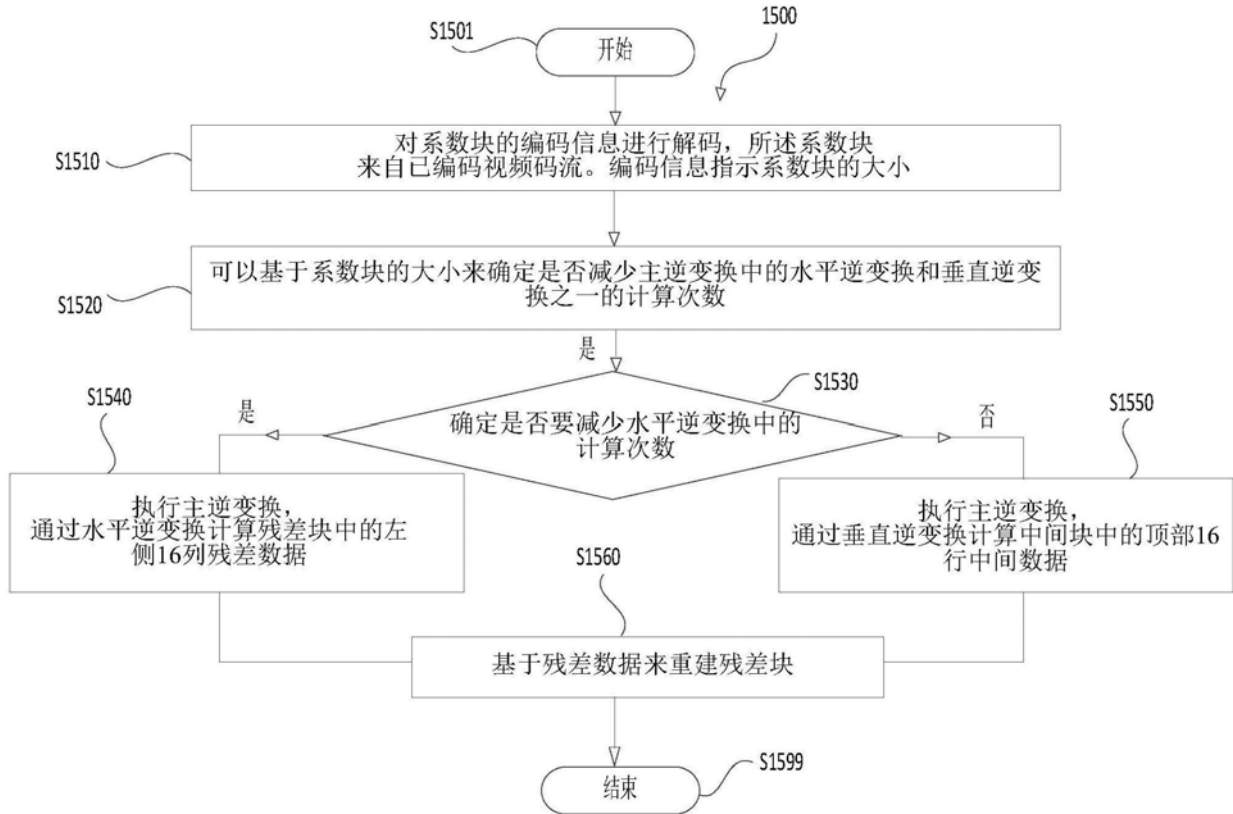


图15

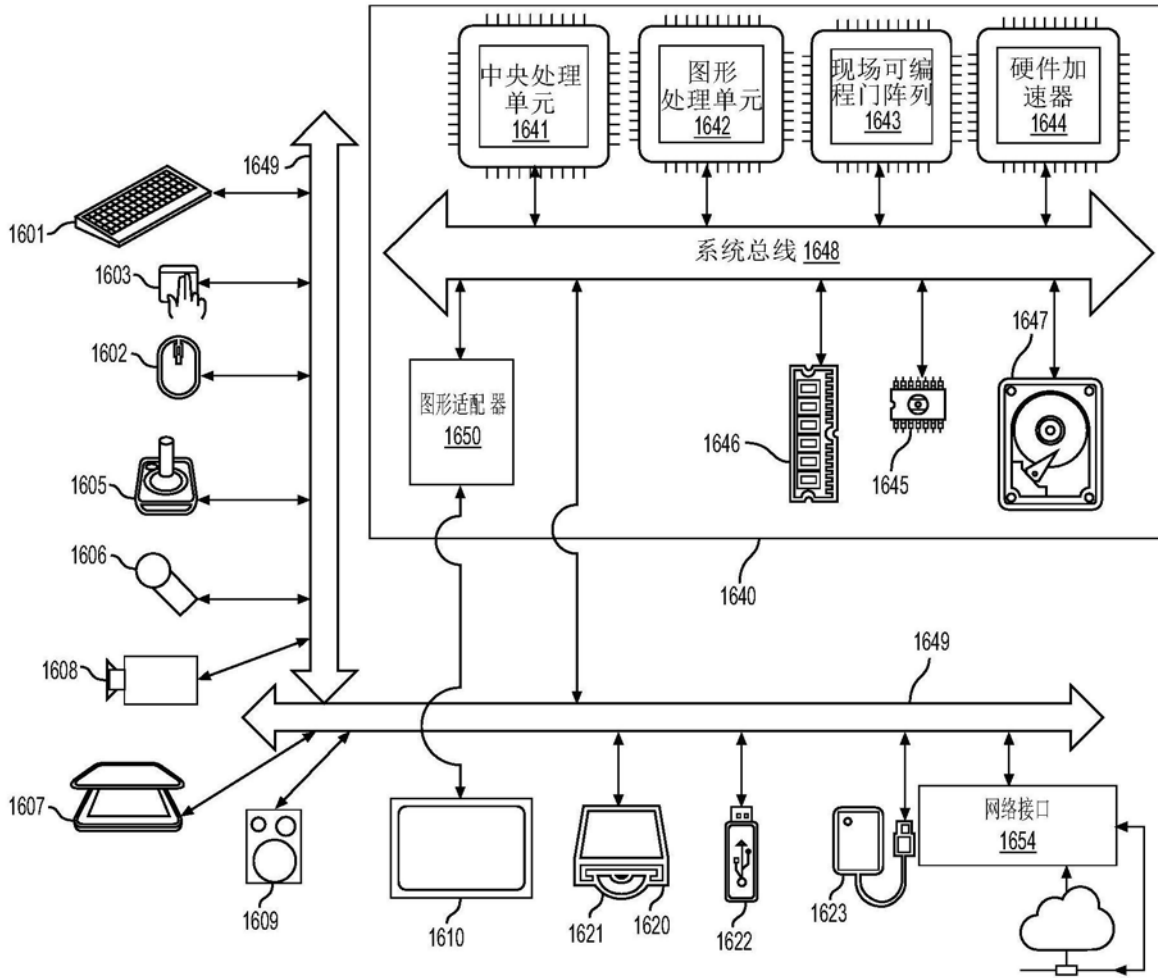


图16