(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 113170149 B
(45)授权公告日 2024.06.25

(21)申请号 201980079768.3
(22)申请日 2019.12.02
(65)同一申请的已公布的文献号
申请公布号 CN 113170149 A
(43)申请公布日 2021.07.23
(30)优先权数据
18306603.4 2018.12.03 EP
18306619.0 2018.12.05 EP
(85)PCT国际申请进入国家阶段日
2021.06.02
(86)PCT国际申请的申请数据
PCT/US2019/063913 2019.12.02
(87)PCT国际申请的公布数据
WO2020/117628 EN 2020.06.11
(73)专利权人 交互数字VC控股公司
地址 美国特拉华州
(72)发明人 K.纳泽 F.莱林内克 T.波伊里尔
(74)专利代理机构 北京市柳沈律师事务所 11105
专利代理师 赵碧洋
(51)Int.Cl.
H04N 19/176(2006.01)
H04N 19/70(2006.01)
H04N 19/119(2006.01)
H04N 19/122(2006.01)
H04N 19/625(2006.01)
(56)对比文件
CN 103931182 A,2014.07.16
US 2016044314 A1,2016.02.11
审查员 杜乾敏

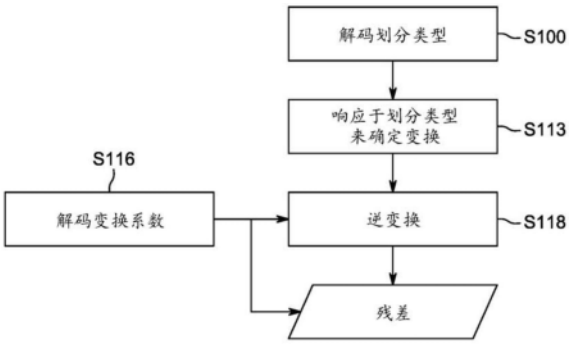
权利要求书4页 说明书20页 附图18页

(54)发明名称

用于图片编码和解码的方法和设备

(57)摘要

提出了一种解码方法。首先对块到变换单元的划分的类型进行解码。然后,响应于划分的类型,为块的每个变换单元确定变换。最后,使用所确定的变换对变换单元的经解码的变换系数进行逆变换。



CN 113170149 B

1.一种解码方法,包括:

- 对块到多个子块的划分的类型进行解码;

- 基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换;和

- 使用所确定的变换对所述子块的经解码的变换系数进行逆变换;其中基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换包括:

- 解码代表所述块的第一子块的第一水平变换的信息和代表所述块的第二子块的第一垂直变换的信息;和

- 基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换。

2.根据权利要求1所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一水平变换相同的第二水平变换。

3.根据权利要求1所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一垂直变换互补的互补垂直变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

4.根据权利要求1所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一垂直变换相同的第二垂直变换。

5.根据权利要求1所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一水平变换互补的互补水平变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

6.根据权利要求3或5所述的方法,其中,所述块由比特流的至少一个比特流部分表示,所述互补垂直变换或所述互补水平变换由所述至少一个比特流部分中的二进制值指示。

7.一种解码装置,包括一个或多个处理器,所述处理器被配置为至少执行:

-对块到多个子块的划分的类型进行解码;

-基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换;和

-使用所确定的变换对所述子块的经解码的变换系数进行逆变换;其中,为了基于所述划分的类型来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为执行:

-对代表所述块的第一子块的第一水平变换的信息和代表所述块的第二子块的第一垂直变换的信息进行确定和解码;和

-基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换。

8.根据权利要求7所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一水平变换相同的第二水平变换。

9.根据权利要求7所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一垂直变换互补的互补垂直变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

10.根据权利要求7所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一垂直变换相同的第二垂直变换。

11.根据权利要求7所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一水平变换互补的互补水平变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

12.根据权利要求9或11所述的装置,其中,所述块由比特流的至少一个比特流部分表示,所述互补垂直变换或所述互补水平变换由所述至少一个比特流部分中的二进制值指示。

13.一种编码方法,包括:

- 对块到多个子块的划分的类型进行确定和编码;
- 基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换;和
- 使用所确定的变换来变换所述子块的残差;其中基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换包括:
- 对代表所述块的第一子块的第一水平变换的信息和代表所述块的第二子块的第一垂直变换的信息进行确定和编码;和
- 基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换。

14.根据权利要求13所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一水平变换相同的第二水平变换。

15.根据权利要求13所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第

一垂直变换互补的互补垂直变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

16.根据权利要求13所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一垂直变换相同的第二垂直变换。

17.根据权利要求13所述的方法,其中,基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换包括:在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一水平变换互补的互补水平变换,其中,如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而与第二变换的第一基函数匹配,则第一变换与第二变换互补。

18.一种编码装置,包括一个或多个处理器,所述处理器被配置为至少执行:

-对块到多个子块的划分的类型进行确定和编码;

-基于所述划分的类型,确定所述多个子块中的子块的一个或多个变换;

-使用所确定的变换来变换所述子块的残差;其中,为了基于所述划分的类型来确定所述块的每个子块的一个或多个变换,所述一个或多个处理器还被配置为执行:

-对代表所述块的第一子块的第一水平变换的信息和代表所述块的第二子块的第一垂直变换的信息进行确定和编码;和

-基于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述多个子块中的子块的一个或多个变换。

19.根据权利要求18所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述另一子块的第一水平变换相同的第二水平变换。

20.根据权利要求18所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是水平的情况下,为所述子块确定与所述子块的第一垂直变换互补的互补垂直变换,其中如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而匹配第二变换的第一基函数,则第一变换与第二变换互补。

21.如权利要求18所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一垂直变换相同的第二垂直变换。

22.根据权利要求18所述的装置,其中,为了基于所述第一水平变换和所述第一垂直变换中的一个或多个来确定所述多个子块中的子块的一个或多个变换,所述一个或多个处理器还被配置为在与另一子块相邻的子块和所述另一子块之间的划分线是垂直的情况下,为所述子块确定与所述另一子块的第一水平变换互补的互补水平变换,其中如果第一变换的第一基函数通过处于第二变换的第一基函数的延续中而匹配第二变换的第一基函数,则第

一变换与第二变换互补。

用于图片编码和解码的方法和设备

技术领域

[0001] 本文的实施例中的至少一个总体上涉及用于图片编码和解码的方法和设备,更具体地,涉及使用变换(即空间变换)进行图片编码和解码的方法和设备。

背景技术

[0002] 为了实现高压缩效率,图像和视频编码方案通常采用预测和变换来利用视频内容中的空间和时间冗余。通常,帧内或帧间预测用于利用帧内或帧间相关性,然后原始图像块和预测图像块之间的差异(通常表示为预测误差、预测残差或预测子)被变换、量化和熵编码。在编码期间,原始图像块通常(可能使用四叉树分割)被分割/划分成子块。为了重建视频,通过对应于预测、变换、量化和熵编码的逆过程来解码压缩数据。

发明内容

[0003] 根据至少一个实施例的一般方面,提出了一种解码方法,包括:

[0004] -对块到变换单元的划分的类型进行解码;

[0005] -响应于所述划分的类型,确定所述块的每个变换单元的变换;和

[0006] -使用所确定的变换对所述变换单元的经解码的变换系数进行逆变换。

[0007] 根据至少一个实施例的一般方面,提供了一种解码装置,其包括一个或多个处理器,所述处理器被配置为执行:

[0008] -对块到变换单元的划分的类型进行解码;

[0009] -响应于所述划分的类型,确定所述块的每个变换单元的变换;和

[0010] -使用所确定的变换对所述变换单元的经解码的变换系数进行逆变换。

[0011] 根据至少一个实施例的一般方面,提出了一种编码方法,包括:

[0012] -对块到变换单元的划分的类型进行确定和编码;

[0013] -响应于所述划分的类型,确定所述块的每个变换单元的变换;

[0014] -使用所确定的变换来变换所述变换单元的残差。

[0015] 根据至少一个实施例的一般方面,提出了一种编码装置,包括一个或多个处理器,所述处理器被配置为执行:

[0016] -对块到变换单元的划分的类型进行确定和编码;

[0017] -响应于所述划分的类型,确定所述块的每个变换单元的变换;

[0018] -使用所确定的变换来变换所述变换单元的残差。

[0019] 根据至少一个实施例的另一个一般方面,比特流被格式化为包括根据上述编码方法生成的信号。

[0020] 本发明的实施例中的一个或多个实施例还提供了一种计算机可读存储介质,其上存储有用于根据上述方法中的任何方法的至少一部分对视频数据进行编码或解码的指令。一个或多个实施例还提供了一种其上存储有根据上述编码方法生成的比特流的计算机可读存储介质。一个或多个实施例还提供了用于发送或接收根据上述编码方法生成的比特流

的方法和装置。一个或多个实施例还提供了一种计算机程序产品,包括用于执行上述方法中的任何方法的至少一部分的指令。

附图说明

[0021] 图1描绘了被垂直(图1的顶部)或水平(图1的底部)划分成变换单元(TU)的块;

[0022] 图2示出了根据各种四叉树、二叉树和三叉树划分类型将块划分成TU;

[0023] 图3示出了根据T型划分将块划分成TU;

[0024] 图4示出了将块垂直和水平划分成四个TU;

[0025] 图5描绘了被垂直(图5(a))和水平(图5(b))划分成两个TU的块;

[0026] 图6、7和8示出了互补变换的不同情况;

[0027] 图9、10和11示出了对各种划分类型的变换的确定;

[0028] 图12-14是根据各种实施例的解码方法的流程图;

[0029] 图15-17、19-20、23-28和30-33在左边示出了变换选择,并在右边示出了各种划分类型的所有TU的2D第一基函数;

[0030] 图18、21-22和29示出了各种划分类型的所有TU的2D第一基函数;

[0031] 图34和35是根据各种实施例的编码方法的流程图;

[0032] 图36示出了根据一个实施例的视频编码器的框图;

[0033] 图37示出了根据一个实施例的视频解码器的框图;和

[0034] 图38示出了其中实施了各种方面和实施例的系统的示例的框图。

具体实施方式

[0035] 在HEVC编码中,图片被分割成大小可配置(通常为64x64)的正方形的CTU。在其他视频编码标准中,大小可能是128x128或256x256。CTU是进行分割成4个大小相等的正方形编码单元(CU)(即宽度和高度是父块大小的一半)的分割的四叉树的根。四叉树是这样一种树,其中父节点可以被划分成四个子节点,每个子节点可以成为针对进行划分成四个子节点的另一划分的父节点。在HEVC中,编码块(CB)被分割成一个或多个预测块(PB),并形成进行分割成变换块(TB)的分割的四叉树的根。对应于编码块、预测块和变换块,编码单元(CU)包括预测单元(PU)和树结构的变换单元(TU)集,PU包括所有颜色分量的预测信息,且TU包括每个颜色分量的残差编码语法结构。亮度分量的CB、PB和TB的大小适用于相应的CU、PU和TU。

[0036] 在最近的编码系统中,例如VVC(通用视频编码),CTU是进行分割成编码单元(CU)的分割的编码树的根。编码树是这样一种树,其中父节点(通常对应于块)可以被划分成子节点(例如划分成2个、3个或4个子节点),每个子节点可以成为针对进行划分成子节点的另一划分的父节点。除了四叉树划分模式,还定义了新的划分模式(二叉树对称划分模式、二叉树不对称划分模式和三叉树划分模式),增加了可能的划分模式的总数。编码树具有唯一的根节点,例如CTU。编码树的叶子是树的终止节点。编码树的每个节点代表可以被进一步划分成更小块(也称为子块)的块。一旦确定了CTU到CU的分割,与编码树的叶子相对应的CU被编码。可以在编码器侧通过率失真优化过程来确定CTU到CU的分割以及用于编码每个CU(对应于编码树的叶子)的编码参数。

[0037] 在本申请中,术语“块”或“图片块”可以用来指CTU、CU、PU、TU、CB、PB和TB中的任何一个。此外,术语“块”或“图片块”可以用来指在H.264/AVC或其他视频编码标准中规定的宏块、分区和子块,并且更一般地指大小多样的样点阵列。

[0038] 在本申请中,术语“重建的”和“解码的”可以互换使用,术语“像素”和“样点”可以互换使用,术语“图像”、“图片”和“帧”可以互换使用。通常,但不是必须的,术语“重建的”用于编码器侧,而“解码的”用于解码器侧。

[0039] 在HEVC中,预测单元使用称为残差四叉树的技术被划分成变换单元。该技术为编码系统提供了灵活性,以便在存在某些不连续性时更好地拟合残差信号属性。因此,使用该工具可以获得显著的比特率增益。

[0040] 在VVC中,除了经典的DCT2变换之外,还使用DST7和DCT8。因此,可以通过水平地应用变换中的一个和垂直地应用变换中的一个来变换整个CU/PU。

[0041] 在一个变型中,CU/PU可以首先被划分成TU,然后被变换。因此,图1描绘了被垂直(图1的顶部)或水平(图1的底部)划分成TU的块(例如PU)。在该图中,块只被垂直或水平划分成两个TU,并且对于水平变换和垂直变换的选择是预定义的。因此,将块划分成TU的方式是受限的。此外,只有一种方式来选择不同TU的变换类型。例如,在块被垂直划分成两个TU的情况下。通过在第一TU的行上应用DCT-8和在列上应用DST-7来变换第一TU(在块的左边),而通过在第二TU的行上应用DST-7和在列上应用DST-7来变换第二TU(在块的右边)。

[0042] 至少一个实施例能够根据各种划分类型将块(例如,CU或PU)划分成多个TU。

[0043] 当划分发生时,编码器有几个选项来选择每个TU的变换类型。这导致显著的比特成本,因为必须为每个TU用信号通知变换类型。因此,至少一个实施例提出限制对TU的变换选择,即在允许几种划分类型并且使用除经典DCT2之外的各种变换的情况下。

[0044] 在一个实施例中,响应于为空间相邻的TU确定的变换,为TU确定变换。

[0045] 图2、3和4描绘了块到TU的各种划分类型。图2示出了根据四叉树(SPLIT_QT)、根据二叉树垂直划分(SPLIT_BT_VER)、根据二叉树水平划分(SPLIT_BT_HOR)、根据三叉树垂直划分(SPLIT_TT_VER)和根据三叉树水平划分(SPLIT_TT_VER)将块(例如CU或PU)划分成TU。在SPLIT_BT_VER和SPLIT_BT_HOR的情况下,块被对称地划分成两个TU。在SPLIT_TT_VER的情况下,宽度为W的块被划分成宽度分别等于0.25*W、0.5*W和0.25*W的三个TU。在SPLIT_TT_HOR的情况下,高度为H的块被划分成三个高度分别等于0.25*H、0.5*H和0.25*H的TU。

[0046] 图3示出了根据T型划分将块(例如,CU或PU)划分成TU。在SPLIT_T_TOP的情况下,首先将块水平划分成相同大小的两个子块,其中顶部子块进一步垂直划分成相同大小的两个TU。因此,该块最终被划分为三个TU。以同样的方式,在SPLIT_T_BOTTOM的情况下,首先将块水平划分成相同大小的两个子块,其中底部子块进一步垂直划分成相同大小的两个TU。在SPLIT_T_RIGHT的情况下,首先将块垂直划分成相同大小的两个子块,其中右子块进一步水平划分成相同大小的两个TU。因此,该块最终被划分为三个TU。以同样的方式,在SPLIT_T_LEFT的情况下,块首先被垂直划分成相同大小的两个子块,其中左子块被进一步水平划分成相同大小的两个TU。

[0047] 图4示出了将块(例如CU或PU)垂直(SPLIT_Q_VER)和水平(SPLIT_Q_HOR)划分成四个TU。在SPLIT_Q_VER的情况下,宽度为W的块被划分成四个TU,每个TU的宽度等于0.25*W。在SPLIT_QT_HOR的情况下,高度为H的块被划分成四个TU,每个TU的高度等于0.25*H。

[0048] 根据图2、3和4的划分类型之一,从单个块/PU/CU,通过划分块/PU/CU,可以获得多个TU。

[0049] 图5描绘了被垂直(图5(a))和水平(图5(b))划分成两个TU的块。每当出现划分时,划分边界周围很可能存在一些不连续性,即很可能存在边。在图5(a)中,出现了垂直划分。因此,在垂直划分线周围很可能存在一些不连续性,而TU1和TU2的垂直特性可以假设是相似的。因此,在一个实施例中,选择Tr1V和Tr2V,使得它们是相似的变换。在一种变型中,选择Tr2V和Tr1V,使得它们是相同的。另一方面,Tr2H被选择为Tr1H的互补变换。

[0050] 在图5(b)中,出现了水平划分。因此,划分线周围很可能存在一些不连续性,而TU1和TU2的水平特性可以假设是相似的。

[0051] 因此,在一个实施例中,选择Tr1H和Tr2HV,使得它们是相似的变换。在一种变型中,选择Tr1H和Tr2H,使得它们是相同的。另一方面,Tr2V被选择为Tr1V的互补变换。

[0052] 更一般地,在一个实施例中,选择沿着划分线来应用的变换,使得它们相似或者甚至相同,而选择垂直于(多个)划分线来应用的变换,使得它们互补。

[0053] 原始变换的互补变换是其第一基函数与原始变换的第一基函数良好匹配(即处于原始变换的第一基函数的延续中,如图6-8所示)的变换。事实上,第一基函数是表示变换的最重要的基函数。它表示所考虑的变换在最低频率下的基函数。互补函数在表1中定义。

[0054]

变换	互补变换
递增第一基函数(例如 DST7)	递减第一基函数(例如 DCT8)或常数第一基函数
递减第一基函数	递增第一基函数或零第一基函数
常数第一基函数	递减第一基函数
零第一基函数	递增第一基函数

[0055] 表1

[0056] 零变换是基函数为零的变换。换句话说,零变换将残差变换成等于零的系数。

[0057] 如表2所示,DCT/DST变换族中的一些可分类为递增、递减和常数的第一基函数。

[0058]

变换	第一基类型
DCT1	常数
DCT2	常数
DCT3	递减
DCT4	递减
DCT5	常数
DCT6	常数
DCT7	递减
DCT8	递减
DST1	常数
DST2	常数

DST3	递增
DST4	递增
DST7	递增
DST8	递增

[0059] 表2

[0060] 例如,DCT2基函数$T_i(j)$,i,j=0,1,…,N-1定义如下:

[0061] $$T_i(j)=\omega_0\cdot\sqrt{\frac{2}{N}}\cdot\cos\left(\frac{\pi\cdot i\cdot(2j+1)}{2N}\right),\text{其中}\omega_0=\begin{cases}\sqrt{\frac{2}{N}} & i=0\\ 1 & i\neq 0\end{cases}$$

[0062] 第一基函数是针对i=0的,因此对于DCT2是常数。

[0063] 再例如,DCT8基函数$T_i(j)$,i,j=0,1,…,N-1定义如下:

[0064] $$T_i(j)=\sqrt{\frac{4}{2N+1}}\cdot\cos\left(\frac{\pi\cdot(2i+1)\cdot(2j+1)}{4N+2}\right)$$

[0065] 第一基函数是针对i=0的,因此对于DCT8是递减的。

[0066] 再例如,DST7基函数$T_i(j)$,i,j=0,1,…,N-1定义如下:

[0067] $$T_i(j)=\sqrt{\frac{4}{2N+1}}\cdot\sin\left(\frac{\pi\cdot(2i+1)\cdot(j+1)}{2N+1}\right)$$

[0068] 第一基函数是针对i=0的,因此对于DST7是递增的。

[0069] 图6、7和8示出了互补变换的不同情况(根据表1)。在每个图形上,绘制了变换的第一基函数(在左边),然后是互补变换的第一基函数(在右边)。在图6中,绘制了DCT8变换的第一基函数(递增基函数),然后是互补变换的第一基函数,其在图6a中为DST7(递减基函数),而在图6b中为DCT2(常数基函数)。

[0070] 在图7中,绘制了DCT7变换的第一基函数(递减基函数),然后是互补变换的第一基函数,其在图7a中为DST8(递增基函数),而在图7b中为零基函数。

[0071] 在图8(a)中,绘制了DCT2变换的第一基函数(常数基函数),然后是互补变换的第一基函数,即DST7(递减基函数)。在图8(b)中,绘制了零基函数,然后是互补变换的第一基函数,即DCT8(递增基函数)。

[0072] 参考回图5,在第一实施例中,沿着划分线(即,平行于划分线)而应用的变换被选择为使得它们相同。因此,由于只有一个变换(分别为:Tr1V;Tr1H)需要被用信号通知,而不是两个(分别为:Tr1V和Tr2V;Tr1H和Tr2H),所以信令成本减少了。

[0073] 图9、10和11示出了针对图2-4所示的各种划分类型的第一实施例,其中沿着划分线(即平行于划分线)而应用的变换是相同的。例如,在图9中,对于划分类型SPLIT_BT_VER,平行于垂直划分线应用了表示为V的相同变换。同样的原理也适用于划分类型SPLIT_BT_HOR,其中,平行于水平划分线应用了表示为H的相同变换。

[0074] 参考回图5,在第二实施例中,垂直于划分线而应用的变换被选择为使得它们是互补的。因此,由于只有一个变换(分别为:Tr1H;Tr1V)需要用信号通知,而不是两个(分别为:Tr1H和Tr2H;Tr1V和Tr2V),所以信令成本减少了。

[0075] 图9、10和11还示出了针对图2-4所示的各种划分类型的第二实施例,其中垂直于划分线而应用的变换是互补的。例如,在图9中,对于划分类型SPLIT_BT_VER,垂直于垂直划分线应用了表示为H和H_c的互补变换。同样的原理也适用于划分类型SPLIT_BT_HOR,其中垂直于水平划分线应用了表示为V和V_c的互补变换。

[0076] 对于划分类型SPLIT_TT_VER,垂直于前两个块(TU1和TU2)上的第一垂直划分线应用了表示为H和H_c的互补变换,并且垂直于后两个块(TU2和TU3)上的第二垂直划分线应用了表示为H_c和H_c_c的互补变换。同样的原理适用于划分类型SPLIT_TT_VER,其中垂直于前两个块(TU1和TU2)上的第一水平划分线应用了表示为V和V_c的互补变换,垂直于后两个块(TU2和TU3)上的第二垂直划分线应用了表示为V_c和V_c_c的互补变换。在TT划分的情况下,有3个水平(垂直)TU。第一水平(垂直)TU1被认为是第二个TU2的参考,并且第二个TU2被认为是第三个TU3的参考。

[0077] 在图10中,对于划分类型SPLIT_Q_HOR,垂直于垂直划分线应用了互补变换。更具体地,将V及其互补变换V_c应用于前两个块,接下来应用作为应用于紧接上面的相邻块的V_c的互补变换的V_c_c,接下来应用作为应用于紧接上面的相邻块的V_c_c的互补变换的V_c_c_c。同样的原理也适用于划分类型SPLIT_Q_VER。H_c是H的互补变换,H_c_c是H_c的互补变换,H_c_c_c是H_c_c的互补变换。在图11中,特定的变换Const在平行于较大划分方向的方向上被应用于较大的TU。Const是一种变换,其第一基函数是常数(例如DCT2)。

[0078] 第一和第二实施例可以独立使用或者可以组合使用。

[0079] 图12是根据一个实施例的解码方法的流程图。在S100,划分类型被解码。在块被划分的情况下(SPLIT),第一水平变换Tr1H和垂直变换Tr1V被解码,更精确地,标识这些变换的索引被解码(S110)。在块中使用的其他变换进一步从这两个变换中的至少一个导出(S112)。例如,在S112中,确定平行于划分线而应用的变换,使得它们在划分线是水平的情况下与Tr1H相同。例如,在S112中,确定平行于划分线而应用的变换,使得它们在划分线是垂直的情况下与Tr1V相同。例如,在S112中,在图5(a)的情况下,Tr2V被确定为与经解码的Tr1V相同。在该实施例中,也可以在S110解码另一变换Tr2H。

[0080] 在一个变型中,块中使用的其他变换进一步从这两个变换中导出。在该变型中,垂直于划分线而应用的变换被确定为互补的。例如,在S112中,在图5(a)的情况下,Tr2V被确定为与经解码的Tr1V相同,并且Tr2H被选择为与经解码的Tr1H互补。因此,从在S110解码的变换中确定变换Tr2H和Tr2V(未被解码)。

[0081] 在变换具有多于一个互补变换的情况下,附加的比特被解码以指示使用哪个互补变换。例如,在经解码的变换Tr1H的第一基函数递减的情况下(例如,Tr1H=DCT8),它可以具有如表1所示的两个互补变换:一个具有递增的第一基函数,另一个具有零第一基函数。二进制值0可以表示第一互补变换(递增第一基函数,例如DST7),值1可以表示零第一基函数互补变换。

[0082] 在经解码的变换第一基函数是零或常数的情况下,存在如表1所示的单个互补变换,在这种情况下没有附加的二进制值被解码。

[0083] 对于QT和T划分类型,最多解码2个附加的二进制值,以指示互补的水平(H_c)变换和垂直(V_c)变换。对于BT,最多解码1个附加的二进制值,以指示互补的水平或垂直变换(分别为针对SPLIT_BT_HOR的V_c和针对SPLIT_BT_VER的H_c)。对于TT划分类型,最多2个附

加的二进制值被解码以指示第一和第二互补变换(针对SPLIT_TT_VER的H_c和H_c_c以及针对SPLIT_TT_HOR的V_c和V_c_c),并且对于Quarter划分类型,最多3个附加的二进制值被解码(针对SPLIT_Q_VER的H_c、H_c_c和H_c_c_c以及针对SPLIT_Q_HOR的V_c、V_c_c和V_c_c_c_c)。

[0084] 回到图12,在块未被划分的情况下(NO_SPLIT),针对整个块解码单个变换(S114)。更具体地,至少一个索引(MTS标志)被解码(S114)以识别所使用的变换。在特定且非限制性的实施例中,S114包括解码MTS标志(S1140)。在MTS=0的情况下,则使用DCT2。在MTS=1的情况下,取决于另一经解码的索引来使用DCT8或DST7(S1142)。可以使用S114的其他变型。例如,单个标志可以被解码以直接指示针对整个块所使用的变换。

[0085] 一旦确定了要使用的变换,就使用所确定的变换对在S116解码的变换系数进行逆变换(S118,S120),以获得残差。

[0086] 在一个变型中,互补变换的数量被限制为一个。在这种情况下,在S112中没有附加的二进制值被解码,因为一旦变换已知,其互补变换也已知,如下表3所示。

[0087]

变换	互补变换
递增第一基函数(例如DST7)	递减第一基函数(例如DCT8)
递减第一基函数	递增第一基函数
常数第一基函数	递减第一基函数
零第一基函数	递增第一基函数

[0088] 表3

[0089] 例如,在经解码的变换Tr1H的第一基函数递减的情况下(例如,Tr1H=DCT8),它具有如表2所示的单个互补变换,即具有递增的第一基函数的变换。该实施例提高了编码效率,因为没有附加的比特被解码。逆变换是互补变换的特例。逆变换是其第一基函数具有与原始变换相反的性质(递增对递减)的变换。即,其第一基函数递增的变换(例如DST7),其逆变换的第一基函数是递减的(例如DCT8)。

[0090] 在一个变型中,表3的最后两行被移除。

[0091] 在图13中,块的第一个TU被视为参考TU,其水平和垂直变换(Tr1H和Tr1V)被解码。右侧、底部和右下的TU可以通过重复使用相同的变换、它的逆变换或T型划分的常数变换来获得(如图11所示)。

[0092] 图14是根据另一实施例的解码方法的流程图。与图12的步骤相同的步骤用相同的附图标记标识,并且不再进一步公开。在这个实施例中,所有的TU变换类型的信令都被移除,包括针对第一个TU的信令。因此,图12的S110被移除。在S113中,响应于经解码的SPLIT类型,直接确定变换。

[0093] 为此,为每个划分类型定义初始变换。然后,到其他变换的映射与图9-11所示的相同。选择与划分的类型非常相似的变换集。换句话说,如图15-33所示,所有TU的2D第一基函数显示出与所需划分的相似性。

[0094] 图15在左边示出了变换选择,在右边显示了在划分类型为SPLIT_QT的情况下所有TU的2D第一基函数。

[0095] TU1水平变换具有递减的(Dec)第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0096] 其他变换是遵循上面定义的规则而确定的。

[0097] TU2水平变换具有递增的(Inc)第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0098] TU3水平变换具有递减的第一基函数(例如DCT8),TU2垂直变换具有递增的第一基函数(例如DST7)。

[0099] TU4水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递增的第一基函数(例如DST7)。

[0100] 图15的左边示出了变换选择,图15的右边显示了所有TU的2D第一基函数。

[0101] 图16在左边示出了变换选择,在右边显示了在划分类型为SPLIT_BT_VER的情况下所有TU的2D第一基函数。

[0102] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0103] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0104] 图17在左边示出了变换选择,在右边显示了在划分类型为SPLIT_BT_HOR的情况下所有TU的2D第一基函数。

[0105] TU1垂直变换具有递减的第一基函数(例如DCT8),TU1水平变换具有递减的第一基函数(例如DCT8)。

[0106] TU2垂直变换具有递增的第一基函数(例如DST7),TU2水平变换具有递减的第一基函数(例如DCT8)。

[0107] 可替换地,对于垂直划分,第一垂直变换可以具有常数的第一基函数,或者对于水平划分,第一水平变换可以具有常数的第一基函数。这是为了获得更锐利的边缘并且更集中于中心,如图18所示。

[0108] 图19在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_VER的情况下所有TU的2D第一基函数。

[0109] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0110] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0111] TU3水平变换具有递减的第一基函数(例如DCT8),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0112] 图20在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_HOR的情况下所有TU的2D第一基函数。

[0113] TU1垂直变换具有递减的第一基函数(例如DCT8),TU1水平变换具有递减的第一基函数(例如DCT8)。

[0114] TU2垂直变换具有递增的第一基函数(例如DST7),TU2水平变换具有递减的第一基函数(例如DCT8)。

[0115] TU3垂直变换具有递减的第一基函数(例如DCT8),TU2水平变换具有递减的第一基函数(例如DCT8)。

[0116] 可替换地,对于垂直划分,第一垂直变换可以具有常数的第一基函数,或者对于水平划分,第一水平变换可以具有常数的第一基函数。这是为了获得更锐利的边缘并且更集中于中心,如图21所示。

[0117] 为了获得跨越TT划分的两个边界的对称性,中心变换可以被认为是常数变换。也就是说,对于垂直TT,可以使用以下变换:

[0118] TU1水平变换具有递增的第一基函数(例如DST7),TU1垂直变换具有常数的第一基函数(例如DCT2)。

[0119] TU2水平变换具有常数的第一基函数(例如DCT2),TU2垂直变换具有常数的第一基函数(例如DCT2)。

[0120] TU3水平变换具有递减的第一基函数(例如DCT8),TU3垂直变换具有常数的第一基函数(例如DCT2)。

[0121] 对于水平TT,以下变换可以是:

[0122] TU1垂直变换具有递增的第一基函数(例如DST7),TU1水平变换具有常数的第一基函数(例如DCT2)。

[0123] TU2垂直变换具有常数的第一基函数(例如DCT2),TU2水平变换具有常数的第一基函数(例如DCT2)。

[0124] TU3垂直变换具有递减的第一基函数(例如DCT8),TU3水平变换具有常数的第一基函数(例如DCT2)。

[0125] 这产生了如图22所示的基函数。显然,这是对TT划分的更好的匹配。

[0126] 一些TT情况可以非常类似于非对称二叉树(ABT),它将块划分成两部分:1/4和3/4。

[0127] 图23在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_VER的情况下所有TU的2D第一基函数。具有左边缘的垂直ABT(ABT_VER_LEFT)可以通过使用以下变换来实现:

[0128] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有常数的第一基函数(例如DCT2)。

[0129] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有常数的第一基函数(例如DCT2)。

[0130] TU3水平变换具有常数的第一基函数(例如DCT2),TU3垂直变换具有常数的第一基函数(例如DCT2)。

[0131] 图24在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_VER的情况下所有TU的2D第一基函数。具有左边缘的垂直ABT(ABT_VER_RIGHT)可以通过使用以下变换来实现:

[0132] TU1水平变换具有常数的第一基函数(例如DCT2),TU1垂直变换具有常数的第一基函数(例如DCT2)。

[0133] TU2水平变换具有递减的第一基函数(例如DCT8),TU2垂直变换具有常数的第一基函数(例如DCT2)。

[0134] TU3水平变换具有递增的第一基函数(例如DST7),TU3垂直变换具有常数的第一基函数(例如DCT2)。

[0135] 图25在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_HOR的情况下所有TU的2D第一基函数。具有左边缘的垂直ABT(ABT_HOR_TOP)可以通过使用以下变换来实现:

[0136] TU1垂直变换具有递减的第一基函数(例如DCT8),TU1水平变换具有常数的第一基函数(例如DCT2)。

[0137] TU2垂直变换具有递增的第一基函数(例如DST7),TU2水平变换具有常数的第一基函数(例如DCT2)。

[0138] TU3垂直变换具有常数的第一基函数(例如DCT2),TU3水平变换具有常的数第一基函数(例如DCT2)。

[0139] 图26在左边示出了变换选择,在右边显示了在划分类型为SPLIT_TT_HOR的情况下所有TU的2D第一基函数。具有左边缘的垂直ABT(ABT_HOR_BOT)可以通过使用以下变换来实现:

[0140] TU1垂直变换具有常数的第一基函数(例如DCT2),TU1水平变换具有常数的第一基函数(例如DCT2)。

[0141] TU2垂直变换具有递减的第一基函数(例如DCT8),TU2水平变换具有常数的第一基函数(例如DCT2)。

[0142] TU3垂直变换具有递增的第一基函数(例如DST7),TU3水平变换具有常数的第一基函数(例如DCT2)。

[0143] 图27在左边示出了变换选择,在右边显示了在划分类型为SPLIT_Q_VER的情况下所有TU的2D第一基函数。

[0144] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0145] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0146] TU3水平变换具有递减的第一基函数(例如DCT8),TU3垂直变换具有递减的第一基函数(例如DCT8)。

[0147] TU4水平变换具有递增的第一基函数(例如DST7),TU4垂直变换具有递减的第一基函数(例如DCT8)。

[0148] 图28在左边示出了变换选择,在右边显示了在划分类型为SPLIT_Q_HOR的情况下所有TU的2D第一基函数。

[0149] TU1垂直变换具有递减的第一基函数(例如DCT8),TU1水平变换具有递减的第一基函数(例如DCT8)。

[0150] TU2垂直变换具有递增的第一基函数(例如DST7),TU2水平变换具有递减的第一基函数(例如DCT8)。

[0151] TU3垂直变换具有递减的第一基函数(例如DCT8),TU3水平变换具有递减的第一基函数(例如DCT8)。

[0152] TU4垂直变换具有递增的第一基函数(例如DST7),TU4水平变换具有递减的第一基函数(例如DCT8)。

[0153] 可替换地,常数变换可用于水平划分的水平变换和垂直划分的垂直变换,如图29

所示。

[0154] 图30在左边示出了变换选择,在右边显示了在划分类型为SPLIT_T_TOP的情况下所有TU的2D第一基函数。

[0155] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0156] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0157] TU3水平变换具有递增的第一基函数(例如DST7),TU3垂直变换具有常数的第一基函数(例如DCT2)。

[0158] 图31在左边示出了变换选择,在右边显示了在划分类型为SPLIT_T_BOTTOM的情况下所有TU的2D第一基函数。

[0159] TU1垂直变换具有递减的第一基函数(例如DCT8),TU1水平变换具有常数的第一基函数(例如DCT2)。

[0160] TU2垂直变换具有递增的第一基函数(例如DST7),TU2水平变换具有递减的第一基函数(例如DCT8)。

[0161] TU3垂直变换具有递增的第一基函数(例如DST7),TU3水平变换具有递增的第一基函数(例如DST7)。

[0162] 图32在左边示出了变换选择,在右边显示了在划分类型为SPLIT_T_RIGHT的情况下所有TU的2D第一基函数。

[0163] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有常数的第一基函数(例如DCT2)。

[0164] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有递减的第一基函数(例如DCT8)。

[0165] TU3水平变换具有递增的第一基函数(例如DST7),TU3垂直变换具有递增的第一基函数(例如DST7)。

[0166] 图33在左边示出了变换选择,在右边显示了在划分类型为SPLIT_T_LEFT的情况下所有TU的2D第一基函数。

[0167] TU1水平变换具有递减的第一基函数(例如DCT8),TU1垂直变换具有递减的第一基函数(例如DCT8)。

[0168] TU2水平变换具有递增的第一基函数(例如DST7),TU2垂直变换具有常数第一基函数(例如DCT2)。

[0169] TU3水平变换具有递减的第一基函数(例如DCT8),TU3垂直变换具有递增的第一基函数(例如DST7)。

[0170] 在替代实施例中,可以认为TU划分与PU或CU划分相同。例如,如果水平二进制划分发生在PU或CU中,则相应的TU划分也是水平二进制的,并且如上述实施例中所解释的那样获得单独的变换。

[0171] 上述实施例可以扩展到递归划分发生的情况。为此,相邻TU的变换之间的关系需要符合本发明中提出的方法。也就是说,划分边界周围的变换需要被取为彼此互补,并且其他方向的变换是相似或相同的。以这种方式,本发明适用于递归和非递归场景的所有可能

的TU划分。

[0172] 图34是根据一个实施例的编码方法的流程图。关于解码方法而公开的实施例和变型也适用于编码方法。在S200,确定并编码划分类型。划分类型通常在编码器侧由率失真优化过程(RDO)确定。在块被划分的情况下(SPLIT),例如使用RDO来确定第一水平变换Tr1H和垂直变换Tr1V。更精确地,标识这些变换的索引被获得和编码(S210)。在块中使用的其他变换进一步从这两个变换中的至少一个导出(S212)。例如,在S112中,确定平行于划分线而应用的变换,使得它们在划分线是水平的情况下与Tr1H相同。例如,在S112中,确定平行于划分线而应用的变换,使得它们在划分线是垂直的情况下与Tr1V相同。例如,在S112中,在图5(a)的情况下,Tr2V被确定为与经解码的Tr1V相同。在该实施例中,也可以在S110解码另一变换Tr2H。

[0173] 在一个变型中,块中使用的其他变换进一步从这两个变换中导出。在该变型中,垂直于划分线而应用的变换被确定为互补的。例如,在S212中,在图5(a)的情况下,Tr2V被确定为与经解码的Tr1V相同,并且Tr2H被选择为与经解码的Tr1H互补。因此,变换Tr2H和Tr2V(未被解码)是从在S210确定的变换中确定的。

[0174] 在变换具有多于一个互补变换的情况下,附加的比特被编码以指示使用哪个互补变换。

[0175] 回到图12,在块未被划分的情况下(NO_SPLIT),针对整个块来确定单个变换(S214)。

[0176] 一旦确定了要使用的变换,就使用所确定的变换来变换在S216确定的残差,以获得变换系数。

[0177] 图35是根据另一实施例的解码方法的流程图。与图34的步骤相同的步骤用相同的附图标记标识,并且不再进一步公开。在这个实施例中,所有的TU变换类型的信令都被移除,包括针对第一个TU的信令。在S213中,响应于经解码的划分类型,直接确定变换。该实施例对应于图14的解码器实施例。

[0178] 本申请描述了各种方面,包括工具、特征、实施例、模型、方法等。这些方面中的许多方面被具体地描述,并且至少为了显示单独的特征,通常以听起来可能是限制性的方式来描述。然而,这是为了描述清楚,并不限制那些方面的应用或范围。事实上,所有不同的方面可以被组合和互换以提供进一步的方面。此外,这些方面也可以与早期文件中描述的方面相结合和互换。

[0179] 本申请中描述和预期的方面可以以许多不同的形式实现。下面的图36、37和38提供了一些实施例,但是其他实施例也是可以预期的,并且图36、37和38的讨论并不限制实现的广度。至少一个方面通常涉及视频编码和解码,并且至少一个其他方面通常涉及发送被生成或编码的比特流。这些和其他方面可以实现为方法、装置、其上存储有根据所描述的任何方法对视频数据进行编码或解码的指令的计算机可读存储介质、和/或其上存储有根据所描述的任何方法生成的比特流的计算机可读存储介质。

[0180] 本文描述了各种方法,并且每种方法包括用于实现所述方法的一个或多个步骤或动作。除非该方法的正确操作需要步骤或动作的特定顺序,否则可以修改或组合特定步骤和/或动作的顺序和/或使用。

[0181] 本申请中描述的各种方法和其他方面可用于修改如图36和图37所示的视频编码

器100和解码器200的模块,例如变换/逆变换模块(125、150、250)。此外,本文的方面不限于VVC或HEVC,并且可以应用于例如其他标准和建议,无论是现有的还是未来开发的,以及任何这样的标准和建议(包括VVC和HEVC)的扩展。除非另有说明,或者技术上排除,否则本申请中描述的方面可以单独使用或者组合使用。

[0182] 在本申请中使用了各种数值,例如,其中块被划分的TU的数量。特定值是为了示例的目的,并且所描述的方面不限于这些特定值。此外,图2-4中的划分类型只是为了举例。可以使用其他划分类型。另一方面,可以使用图2-4的划分类型的子集。

[0183] 图36示出了编码器100。该编码器100的变型是被预期的,但是为了清楚起见,下面描述编码器100,而没有描述所有期望的变型。

[0184] 在被编码之前,视频序列可以经过预编码处理(101),例如,对输入彩色图片应用颜色变换(例如,从RGB 4:4:4到YCbCr 4:2:0的变换),或者执行输入图片分量的重新映射,以便获得对压缩更有弹性的信号分布(例如,使用对颜色分量之一的直方图均衡)。元数据可以与预处理相关联,并附加到比特流。

[0185] 在编码器100中,图片由如下所述的编码器元件编码。要编码的图片被分割(102)并以例如CU为单元进行处理。使用例如帧内或帧间模式对每个单元进行编码。当单元以帧内模式被编码时,它执行帧内预测(160)。在帧间模式中,执行运动估计(175)和补偿(170)。编码器决定(105)将帧内模式或帧间模式中的哪一个用于编码该单元,并通过例如预测模式标志来指示帧内/帧间决定。例如,通过从原始图像块中减去(110)预测块来计算预测残差。

[0186] 预测残差然后被变换(125)和量化(130)。量化的变换系数以及运动矢量和其他语法元素被熵编码(145)以输出比特流。编码器可以跳过变换,直接对未变换的残差信号进行量化。编码器可以绕过变换和量化,即直接对残差进行编码,而无需应用变换或量化过程。

[0187] 编码器对经编码的块进行解码,来为进一步的预测提供参考。对经量化的变换系数进行去量化(140)和逆变换(150)以解码预测残差。通过组合(155)经解码的预测残差和预测块,重建图像块。环路内滤波器(165)被应用于重建图片,以执行例如去方块/采样自适应偏移(SAO)滤波来减少编码伪像。滤波后的图像存储在参考图片缓冲器(180)中。

[0188] 图37示出了视频解码器200的框图。在解码器200中,比特流由如下所述的解码器元件解码。视频解码器200通常执行与编码遍次相逆的解码遍次,如图37所示。编码器100通常还执行视频解码作为编码视频数据的一部分。

[0189] 特别地,解码器的输入包括视频比特流,其可以由视频编码器100生成。比特流首先被熵解码(230)以获得变换系数、运动矢量和其他经编码的信息。图片分割信息指示图片是如何被分割的。因此,解码器可以根据经解码的图片分割信息来拆分(235)图片。对变换系数进行去量化(240)和逆变换(250)以解码预测残差。通过组合(255)经解码的预测残差和预测块,重建图像块。预测块可以从帧内预测(260)或运动补偿预测(即帧间预测)(275)中获得(270)。环路内滤波器(265)被应用于重建图像。滤波后的图像存储在参考图片缓冲器(280)中。

[0190] 经解码的图片可以进一步经过后解码处理(285),例如逆颜色变换(例如,从YCbCr 4:2:0到RGB 4:4:4的转换)或逆重新映射,该逆新重映射执行预编码处理(101)中执行的重新映射过程的逆。后解码处理可以使用在预编码处理中导出并在比特流中用信号通知的元

数据。

[0191] 图38示出了其中实现了各种方面和实施例的系统的示例的框图。系统1000可以被实现为包括以下描述的各种组件的设备,并且被配置为执行本文档中描述的方面中的一个或多个方面。这种设备的示例包括但不限于各种电子设备,诸如个人计算机、膝上型计算机、智能电话、平板计算机、数字多媒体机顶盒、数字电视接收机、个人视频记录系统、联网家用电器和服务器。系统1000的元件可以单独或组合地体现在单个集成电路(IC)、多个IC和/或分立组件中。例如,在至少一个实施例中,系统1000的处理元件和编码器/解码器元件分布在多个IC和/或分立组件上。在各种实施例中,系统1000经由例如通信总线或通过专用输入和/或输出端口通信耦合到一个或多个其他系统或其他电子设备。在各种实施例中,系统1000被配置成实现本文档中描述的方面中的一个或多个方面。

[0192] 系统1000包括至少一个处理器1010,处理器1010被配置为执行加载在其中的指令,用于实现例如本文档中描述的各个方面。处理器1010可以包括嵌入式存储器、输入输出接口和本领域已知的各种其他电路系统1000包括至少一个存储器1020(例如,易失性存储设备和/或非易失性存储设备)。系统1000包括存储设备1040,该存储设备1040可以包括非易失性存储器和/或易失性存储器,包括但不限于电可擦除可编程只读存储器(EEPROM)、只读存储器(ROM)、可编程只读存储器(PROM)、随机存取存储器(RAM)、动态随机存取存储器(DRAM)、静态随机存取存储器(SRAM)、闪存、磁盘驱动器、和/或光盘驱动器。作为非限制性示例,存储设备1040可以包括内部存储设备、附接存储设备(包括可拆卸和不可拆卸存储设备)、和/或网络可访问存储设备。

[0193] 系统1000包括编码器/解码器模块1030,其被配置为例如处理数据以提供经编码的视频或经解码的视频,并且编码器/解码器模块1030可以包括其自己的处理器和存储器。编码器/解码器模块1030代表可以被包括在设备中以执行编码和/或解码功能的模块。众所周知,设备可以包括编码和解码模块中的一个或两者。另外,编码器/解码器模块1030可以被实现为系统1000的独立元件,或者可以作为本领域技术人员已知的硬件和软件的组合被结合在处理器1010内

[0194] 要加载到处理器1010或编码器/解码器1030上以执行本文档中描述的各个方面的程序代码可以存储在存储设备1040中,并随后加载到存储器1020上以供处理器1010执行。根据各种实施例,处理器1010、存储器1020、存储设备1040和编码器/解码器模块1030中的一个或多个可以在执行本文档中描述的过程期间存储各种项目中的一个或多个。这种存储的项目可以包括但不限于输入视频、经解码的视频或经解码的视频的部分、比特流、矩阵、变量,以及来自等式、公式、运算和运算逻辑的处理的中间结果或最终结果。

[0195] 在一些实施例中,处理器1010和/或编码器/解码器模块1030内部的存储器用于存储指令,并为编码或解码期间所需的处理提供工作存储器。然而,在其他实施例中,处理设备(例如,处理设备可以是处理器1010或编码器/解码器模块1030)外部的存储器用于这些功能中的一个或多个。外部存储器可以是存储器1020和/或存储设备1040,例如动态易失性存储器和/或非易失性闪存。在几个实施例中,外部非易失性闪存用于存储例如电视的操作系统。在至少一个实施例中,诸如RAM的快速外部动态易失性存储器被用作视频编码和解码操作的工作存储器,例如用于MPEG-2(MPEG指的是运动图片专家组,MPEG-2也称为ISO/IEC13818,13818-1也称为H.222,13818-2也称为H.262)、HEVC(HEVC指的是高效视频编码,

也称为H.265和MPEG-HPart2)或VVC(通用视频编码,联合视频专家组JVET开发的一种新的标准)。

[0196] 如框1130所示,可以通过各种输入设备来提供对系统1000的元件的输入。这种输入设备包括但不限于:(i)接收例如由广播公司通过空中发送的射频(RF)信号的RF部分,(ii)组件(COMP)输入端子(或一组COMP输入端子),(iii)通用串行总线(USB)输入端子,和/或(iv)高清晰度多媒体接口(HDMI)输入端子。图10中未示出的其他示例包括复合视频。

[0197] 在各种实施例中,框1130的输入设备具有本领域已知的相关联的相应输入处理元件例如,RF部分可以与适合于以下各项的元件相关联:(i)选择期望的频率(也称为选择信号,或者将信号频带限制到频率的频带),(ii)对选择的信号进行下变频,(iii)再次进行频带限制到更窄频率的频带以选择(例如)在某些实施例中可以称为信道的信号频带,(iv)解调经下变频和频带限制的信号,(v)执行纠错,以及(vi)解复用以选择期望的数据分组流。各种实施例的RF部分包括一个或多个执行这些功能的元件,例如频率选择器、信号选择器、频带限制器、信道选择器、滤波器、下变频器、解调器、纠错器和解复用器。RF部分可以包括调谐器,该调谐器执行各种这些功能,包括例如将接收信号下变频到较低频率(例如,中频或近基带频率)或基带。在一个机顶盒实施例中,RF部分及其相关联的输入处理元件接收通过有线(例如,电缆)介质发送的RF信号,并通过滤波、下变频和再次滤波到期望的频带来执行频率选择。各种实施例重新排列上述(和其他)元素的顺序,移除这些元素中的一些,和/或添加执行相似或不同功能的其他元素。添加元件可以包括在现有元件之间插入元件,例如插入放大器和模数转换器。在各种实施例中,RF部分包括天线。

[0198] 此外,USB和/或HDMI端子可以包括各自的接口处理器,用于通过USB和/或HDMI连接将系统1000连接到其他电子设备。应当理解,输入处理的各个方面,例如里德-所罗门纠错,可以根据需要在例如单独的输入处理IC或处理器1010中实现。类似地,USB或HDMI接口处理的各方面可以根据需要在单独的接口IC内或处理器1010内实现。解调、纠错和解复用的流被提供给各种处理元件,包括例如与存储器和存储元件组合操作的处理器1010和编码器/解码器1030,以根据需要处理数据流,来呈现在输出设备上。

[0199] 系统1000的各种元件可以设置在集成外壳内,在集成外壳内,各种元件可以使用合适的连接布置(例如本领域已知的内部总线,包括IC间(I2C)总线、布线和印刷电路板)相互连接并在其间发送数据。

[0200] 系统1000包括能够经由通信信道1060与其他设备通信的通信接口1050。通信接口1050可以包括但不限于被配置为通过通信信道1060发送和接收数据的收发器。通信接口1050可以包括但不限于调制解调器或网卡,并且通信信道1060可以例如在有线和/或无线介质中实现。

[0201] 在各种实施例中,使用诸如Wi-Fi网络的无线网络,例如IEEE 802.11(IEEE指电气和电子工程师协会),将数据流式传输或以其他方式提供给系统1000。这些实施例的Wi-Fi信号是通过适于Wi-Fi通信的通信信道1060和通信接口1050接收的。这些实施例的通信信道1060通常连接到接入点或路由器,该接入点或路由器提供对包括互联网在内的外部网络的访问,以允许流应用和其他超顶通信。其他实施例使用机顶盒向系统1000提供流式数据,机顶盒通过输入块1130的HDMI连接传送数据。还有其他实施例使用输入块1130的RF连接向系统1000提供流式数据。如上所述,各种实施例以非流式方式提供数据。另外,各种实施例

使用除Wi-Fi之外的无线网络,例如蜂窝网络或蓝牙网络。

[0202] 系统1000可以将输出信号提供给各种输出设备,包括显示器1100、扬声器1110和其他外围设备1120。各种实施例的显示器1100包括例如触摸屏显示器、有机发光二极管(OLED)显示器、弯曲显示器、和/或可折叠显示器中的一个或多个。显示器1100可以用于电视机、平板计算机、膝上型计算机、蜂窝电话(移动电话)或其他设备。显示器1100也可以与其他组件集成(例如,如在智能电话中),或者分离(例如,用于膝上型计算机的外部监视器)。在实施例的各种示例中,其他外围设备1120包括独立数字视频盘(或数字多功能盘)(对两个术语来说都是DVR)、盘播放器、立体声系统和/或照明系统中的一个或多个。各种实施例使用一个或多个外围设备1120,其基于系统1000的输出提供功能。例如,盘播放器执行播放系统1000的输出的功能。

[0203] 在各种实施例中,使用诸如AV.Link、消费电子控制(CEC)或支持用户干预或不干预的设备到设备控制的其他通信协议的信令在系统1000和显示器1100、扬声器1110或其他外围设备1120之间传送控制信号。输出设备可以通过各自的接口1070、1080和1090经由专用连接通信地耦合到系统1000。可替换地,输出设备可以经由通信接口1050使用通信信道1060连接到系统1000。显示器1100和扬声器1110可以与电子设备(例如电视机)中的系统1000的其他组件集成在单个单元中。在各种实施例中,显示接口1070包括显示驱动器,例如时序控制器(TCon)芯片。

[0204] 可替换地,显示器1100和扬声器1110可以与其他组件中的一个或多个分离,例如,如果输入1130的RF部分是独立机顶盒的一部分。在显示器1100和扬声器1110是外部组件的各种实施例中,输出信号可以经由专用输出连接(包括例如HDMI端口、USB端口或COMP输出)来提供。

[0205] 这些实施例可以通过由处理器1010实现的计算机软件或者通过硬件或者通过硬件和软件的组合来实现。作为非限制性示例,实施例可以由一个或多个集成电路来实现。作为非限制性示例,存储器1020可以是适合于技术环境的任何类型,并且可以使用任何适当的数据存储技术来实现,例如光存储设备、磁存储设备、基于半导体的存储设备、固定存储器和可移动存储器。作为非限制性示例,处理器1010可以是适合于技术环境的任何类型,并且可以包括微处理器、通用计算机、专用计算机和基于多核架构的处理器中的一个或多个。

[0206] 各种实现涉及解码。本申请中使用的“解码”可以包括例如对接收的经编码的序列执行的全部或部分过程,以便产生适于显示的最终输出。在各种实施例中,这些过程包括通常由解码器执行的一个或多个过程,例如熵解码、逆量化、逆变换和差分解码。在各种实施例中,这样的过程也包括或者可替换地包括由本申请中描述的各种实施方式的解码器执行的过程,例如,确定要应用于块的TU上的变换。

[0207] 作为进一步的示例,在一个实施例中,“解码”仅指熵解码,在另一实施例中,“解码”仅指差分解码,而在另一实施例中,“解码”指熵解码和差分解码的组合。基于具体描述的上下文,短语“解码过程”旨在具体指操作的子集还是泛指更广泛的解码过程将是清楚的,并且被认为是本领域技术人员所熟知的

[0208] 各种实施方式涉及编码。以类似于上面关于“解码”的讨论的方式,本申请中使用的“编码”可以包括例如对输入视频序列执行的全部或部分过程,以便产生经编码的比特流。在各种实施例中,这些过程包括通常由编码器执行的一个或多个过程,例如,分割、差分

编码、变换、量化和熵编码。在各种实施例中,这样的过程也包括或者可替换地包括由本申请中描述的各种实施方式的编码器执行的过程,例如,例如,确定要应用于块的TU上的变换。

[0209] 作为进一步的示例,在一个实施例中,“编码”仅指熵编码,在另一实施例中,“编码”仅指差分编码,在另一实施例中,“编码”指差分编码和熵编码的组合。基于具体描述的上下文,短语“编码过程”旨在具体指操作的子集还是泛指更广泛的编码过程将是清楚的,并且被认为是本领域技术人员所熟知的注意,这里使用的语法元素,例如MTS标志/索引,是描述性术语。因此,它们不排除使用其他语法元素名称。

[0210] 当图作为流程图呈现时,应该理解它也提供了相应装置的框图。类似地,当图被呈现为框图时,应该理解它也提供了相应方法/过程的流程图。

[0211] 各种实施例涉及率失真优化。特别是,在编码过程中,通常会考虑速率和失真之间的平衡或权衡,通常会受到计算复杂性的限制。率失真优化通常被表述为最小化率失真函数,它是速率和失真的加权和。有不同的方法来解决率失真优化问题。例如,这些方法可以基于对所有编码选项的广泛测试,包括所有考虑的模式或编码参数值,以及对它们的编码成本和编码和解码后重建信号的相关失真的完整评估。还可以使用更快的方法来节省编码复杂度,特别是基于预测或预测残差信号而不是重建信号来计算近似失真。也可以混合使用这两种方法,例如只对一些可能的编码选项使用近似失真,而对其他编码选项使用完全失真。其他方法仅评估可能编码选项的子集。更一般地,许多方法采用各种技术中的任何一种来执行优化,但是优化不一定是对编码成本和相关失真的完整评估。

[0212] 这里描述的实施方式和方面可以在例如方法或过程、装置、软件程序、数据流或信号中实现。即使仅在单一形式的实施方式的上下文中讨论(例如,仅作为方法讨论),所讨论的特征的实施方式也可以以其他形式实现(例如,装置或程序)。装置可以用例如适当的硬件、软件和固件来实现。方法可以在例如处理器中实现,处理器通常指处理设备,包括例如计算机、微处理器、集成电路或可编程逻辑设备。处理器还包括通信设备,例如计算机、蜂窝电话、便携式/个人数字助理(“PDA”)和其他便于终端用户之间信息通信的设备。

[0213] 提及“一个实施例”或“实施例”或“一个实施方式”或“实施方式”及其其他变型,意味着结合该实施例描述的特定特征、结构、特性等被包括在至少一个实施例中。因此,短语“在一个实施例中”或“在实施例中”或“在一个实施方式中”或“在实施方式中”的出现,以及在本申请各处出现的任何其他变型,不一定都指同一实施例。

[0214] 此外,本申请可以指“确定”各条信息。确定信息可以包括例如估计信息、计算信息、预测信息或从存储器中检索信息中的一个或多个。

[0215] 此外,本申请可以指“访问”各种信息。访问信息可以包括例如接收信息、检索信息(例如,从存储器)、存储信息、移动信息、复制信息、计算信息、确定信息、预测信息或估计信息中的一个或多个。

[0216] 此外,本申请可以指“接收”各条信息。与“访问”一样,接收意指广义的术语。接收信息可以包括例如访问信息或检索信息(例如,从存储器)中的一个或多个。此外,“接收”通常以这样或那样的方式涉及诸如存储信息、处理信息、发送信息、移动信息、复制信息、擦除信息、计算信息、确定信息、预测信息或估计信息的操作期间。

[0217] 应当理解,例如在“A/B”、“A和/或B”和“A和B中的至少一个”的情况下,使用以下

“/”、“和/或”、和“……中的至少一个”意在包括仅选择第一个列出的选项(A),或仅选择第二个列出的选项(B),或选择两个选项(A和B)。再例如,在“A、B和/或C”和“A、B和C中的至少一个”的情况下,这样的措辞旨在包括仅选择第一个列出的选项(A),或仅选择第二个列出的选项(B),或仅选择第三个列出的选项(C),或仅选择第一个和第二个列出的选项(A和B),或者仅选择第一和第三个列出的选项(A和C),或者仅选择第二和第三个列出的选项(B和C),或者选择所有三个选项(A和B和C)。正如本领域和相关领域的普通技术人员所清楚的那样,这可以扩展到所列出的许多项目。

[0218] 此外,如这里所使用的,词语“信号”尤其是指向相应的解码器指示某些东西。例如,在某些实施例中,编码器用信号通知特定的一个互补变换。这样,在实施例中,在编码器侧和解码器侧使用相同的参数。因此,例如,编码器可以向解码器发送(显式信令)特定参数,使得解码器可以使用相同的特定参数。相反,如果解码器已经具有特定参数以及其他参数,则可以使用信令而不发送(隐式信令)来简单地允许解码器知道并选择特定参数。通过避免任何实际函数的传输,在各种实施例中实现了比特节省。应当理解,信令可以以多种方式实现。例如,在各种实施例中,一个或多个语法元素、标志等用于向相应的解码器发信号通知信息。虽然前面提到了词语“信号”的动词形式,但是词语“信号”在这里也可以用作名词。

[0219] 对于本领域的普通技术人员来说显而易见的是,实施方式可以产生各种信号,这些信号被格式化以携带例如可以被存储或发送的信息。该信息可以包括例如用于执行方法的指令,或者由所描述的实施方式之一产生的数据。例如,信号可以被格式化以携带所描述的实施例的比特流。这种信号可以被格式化为例如电磁波(例如,使用频谱的射频部分)或基带信号。格式化可以包括,例如,编码数据流和用经编码的数据流调制载波。信号携带的信息可以是例如模拟或数字信息。众所周知,信号可以通过各种不同的有线或无线链路发送。信号可以存储在处理器可读介质上。

[0220] 我们描述了许多实施例。这些实施例的特征可以在各种权利要求类别和类型之间单独提供或者以任意组合的方式提供。此外,实施例可以单独或以任意组合包括跨越各种权利要求类别和类型的以下特征、设备或方面中的一个或多个:

[0221] • 包括所描述的语法元素(例如指示互补变换的索引或其变型)中的一个或多个的比特流或信号。

[0222] • 在信令中插入语法元素,该语法元素使得解码器能够以与编码器所使用的方式相对应的方式对经解码的系数进行逆变换。

[0223] • 创建和/或发送和/或接收和/或解码包括所描述的语法元素或其变型中的一个或多个的比特流或信号。

[0224] • TV、机顶盒、蜂窝电话、平板计算机或根据所描述的任何实施例执行变换确定的其他电子设备。

[0225] • TV、机顶盒、蜂窝电话、平板计算机或根据所描述的任何实施例来执行变换确定并且显示(例如,使用监视器、屏幕或其他类型的显示器)结果图像的其他电子设备。

[0226] • TV、机顶盒、蜂窝电话、平板计算机或调谐(例如,使用调谐器)信道以接收包括经编码的图像的信号并根据所描述的任何实施例来执行变换确定的其他电子设备。

[0227] • TV、机顶盒、蜂窝电话、平板计算机或通过空中接收(例如,使用天线)包括经编

码的图像的信号并根据所描述的任何实施例执行变换确定的其他电子设备。

[0228] 公开了一种解码方法,包括:

[0229] -对块到变换单元的划分的类型进行解码;

[0230] -响应于所述划分的类型,确定所述块的每个变换单元的变换;和

[0231] -使用所确定的变换对所述变换单元的经解码的变换系数进行逆变换。

[0232] 公开了一种编码方法,包括:

[0233] -对块到变换单元的划分的类型进行确定和编码;

[0234] -响应于所述划分的类型,确定所述块的每个变换单元的变换;和

[0235] -使用所确定的变换来变换所述变换单元的残差。

[0236] 公开了一种解码装置,包括一个或多个处理器,被配置为执行:

[0237] -对块到变换单元的划分的类型进行解码;

[0238] -响应于所述划分的类型,确定所述块的每个变换单元的变换;

[0239] -使用所确定的变换对所述变换单元的经解码的变换系数进行逆变换。

[0240] 公开了一种编码装置,包括一个或多个处理器,被配置为执行:

[0241] -对块到变换单元的划分的类型进行确定和编码;

[0242] -响应于所述划分的类型,确定所述块的每个变换单元的变换;

[0243] -使用所确定的变换来变换所述变换单元的残差。

[0244] 本发明的实施例中的一个或多个实施例还提供了一种计算机可读存储介质,其上存储有用于根据上述方法中的任何方法的至少一部分对视频数据进行编码或解码的指令。一个或多个实施例还提供了一种其上存储有根据上述编码方法生成的比特流的计算机可读存储介质。一个或多个实施例还提供了用于发送或接收根据上述编码方法生成的比特流的方法和装置。一个或多个实施例还提供了一种计算机程序产品,包括用于执行上述方法中的任何方法的至少一部分的指令。

[0245] 在特定实施例中,响应于所述划分的类型,确定所述块的每个变换单元的变换包括:

[0246] -解码所述块的第一变换单元的第一水平变换和所述块的第二变换单元的第一垂直变换;和

[0247] -响应于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述块的所有变换单元的变换。

[0248] 在特定实施例中,所述第一和第二变换单元是所述块的相同变换单元。

[0249] 在特定实施例中,响应于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述块的所有变换单元的变换包括:在与另一变换单元相邻的变换单元和所述另一变换单元之间的划分线是水平的情况下,为所述变换单元确定与所述另一变换单元的水平变换相同的水平变换。

[0250] 在特定实施例中,响应于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述块的所有变换单元的变换包括:在与另一变换单元相邻的变换单元和所述另一变换单元之间的划分线是水平的情况下,为所述变换单元确定与所述另一变换的垂直变换互补的垂直变换,其中两个变换在其第一基函数匹配的情况下是互补的。

[0251] 在特定实施例中,响应于所述第一水平变换和所述第一垂直变换中的一个或多

个,确定所述块的所有变换单元的变换包括:在与另一变换单元相邻的变换单元和所述另一变换单元之间的划分线是垂直的情况下,为所述变换单元确定与所述另一变换单元的垂直变换相同的垂直变换。

[0252] 在特定实施例中,响应于所述第一水平变换和所述第一垂直变换中的一个或多个,确定所述块的所有变换单元的变换包括:在与另一变换单元相邻的变换单元和所述另一变换单元之间的划分线是垂直的情况下,为所述变换单元确定与所述另一变换的水平变换互补的水平变换,其中两个变换在其第一基函数匹配的情况下是互补的。

[0253] 在特定实施例中,所述互补变换由经编码/解码的二进制值指示。

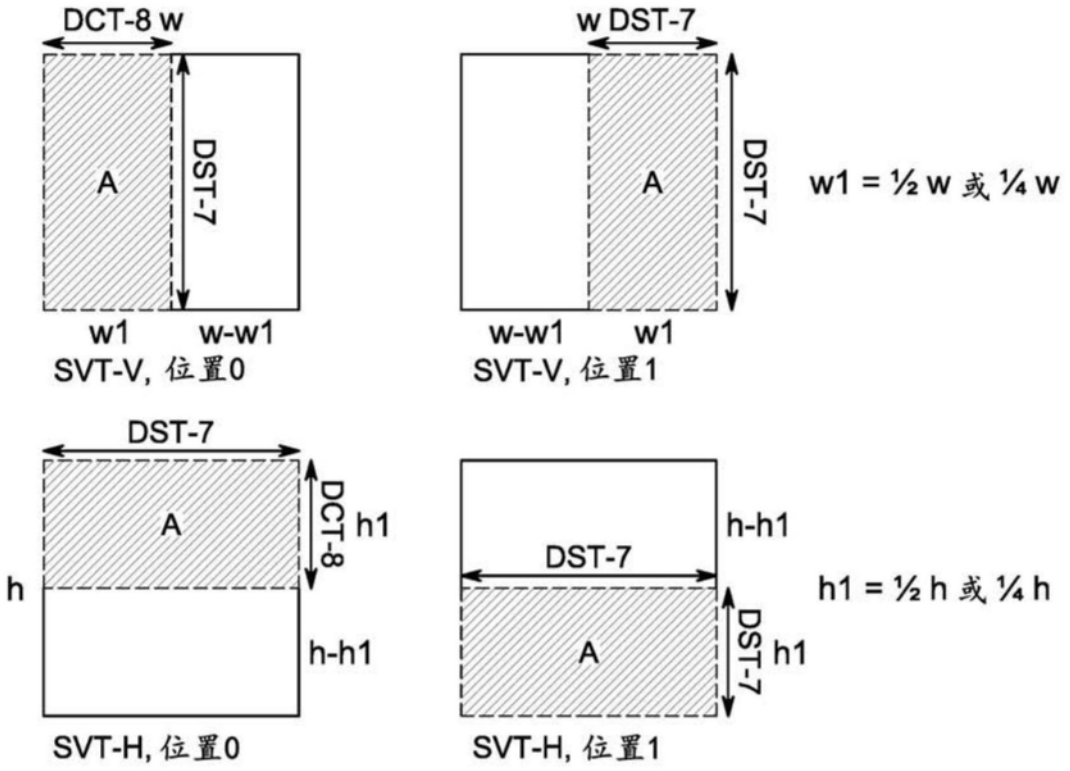

DCT-8 w
A
DST-7
w1
w-w1
SVT-V, 位置0
w DST-7
A
DST-7
w-w1
w1
SVT-V, 位置1
w1 = ½ w 或 ¼ w
DST-7
A
DCT-8
h1
h
h-h1
SVT-H, 位置0
h-h1
DST-7
A
DST-7
h1
SVT-H, 位置1
h1 = ½ h 或 ¼ h

图1

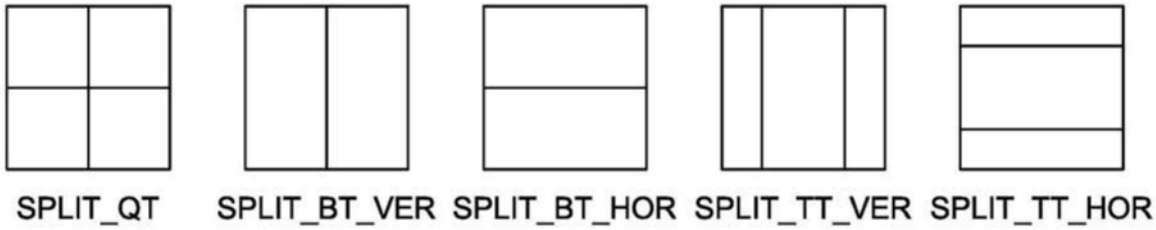

SPLIT_QT
SPLIT_BT_VER
SPLIT_BT_HOR
SPLIT_TT_VER
SPLIT_TT_HOR

图2

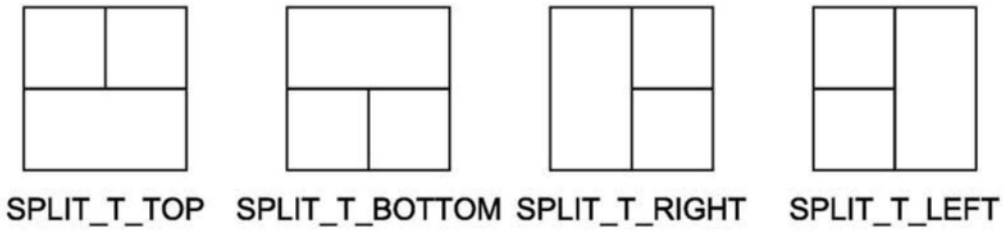

SPLIT_T_TOP
SPLIT_T_BOTTOM
SPLIT_T_RIGHT
SPLIT_T_LEFT

图3

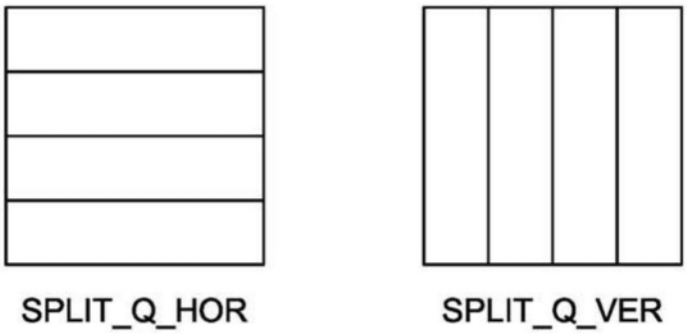

SPLIT_Q_HOR
SPLIT_Q_VER

图4

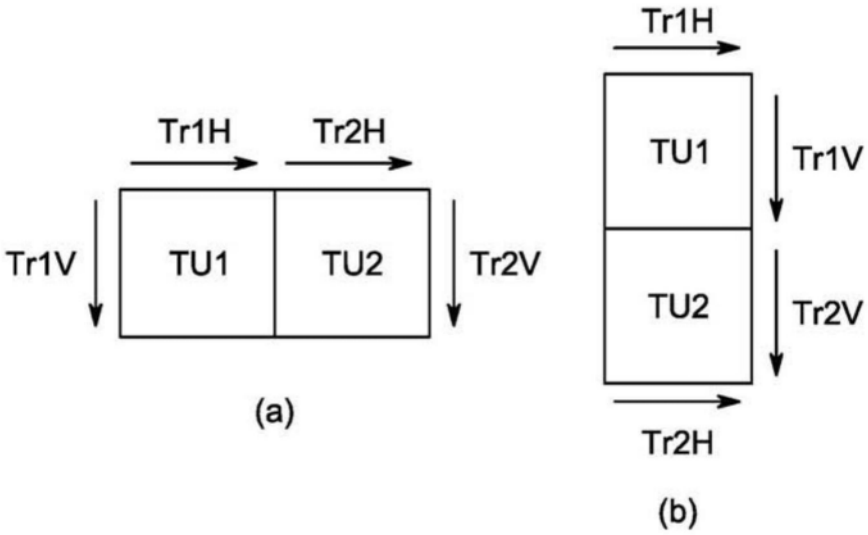
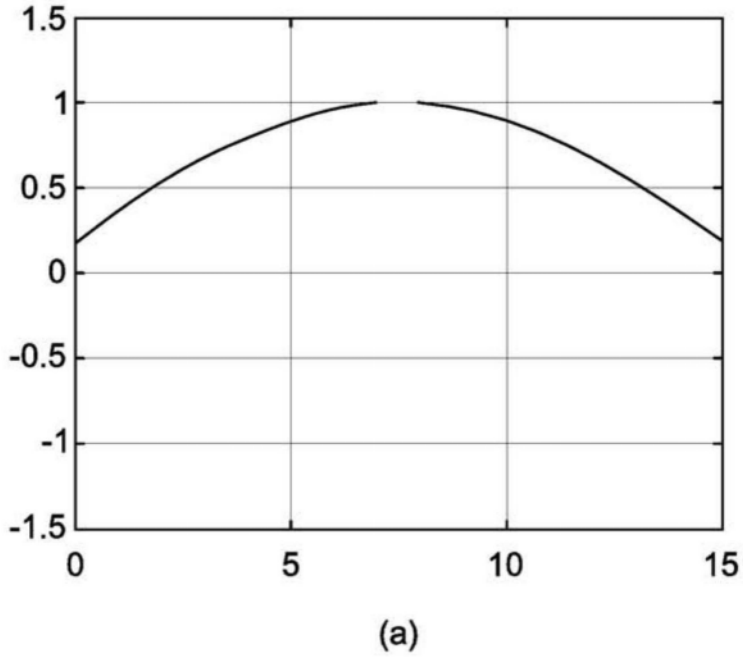
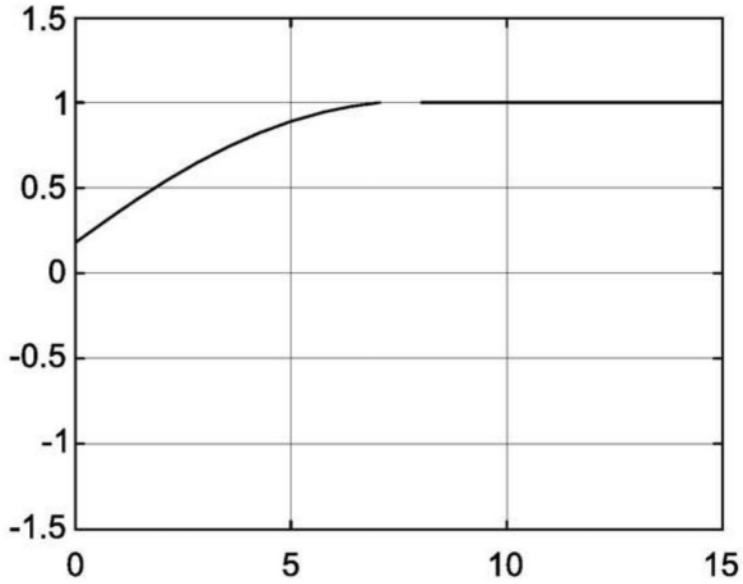

Tr1H
Tr2H
Tr1V
TU1
TU2
Tr2V
(a)
Tr1H
TU1
Tr1V
TU2
Tr2V
Tr2H
(b)

图5

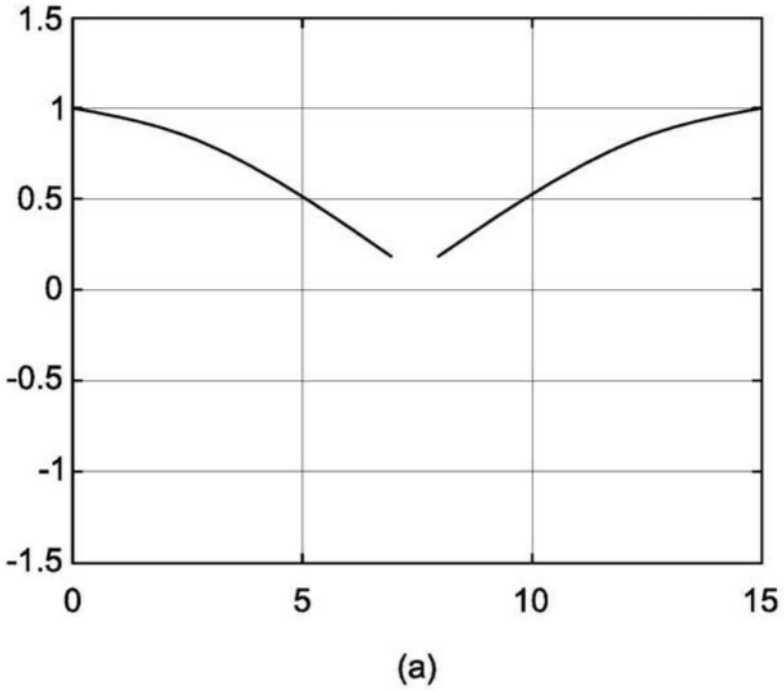


(a)

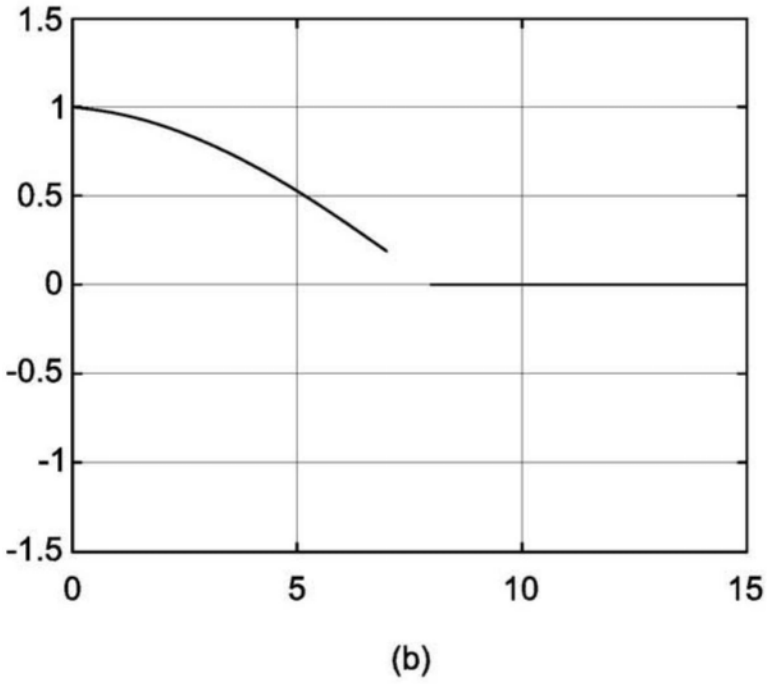


(b)

图6

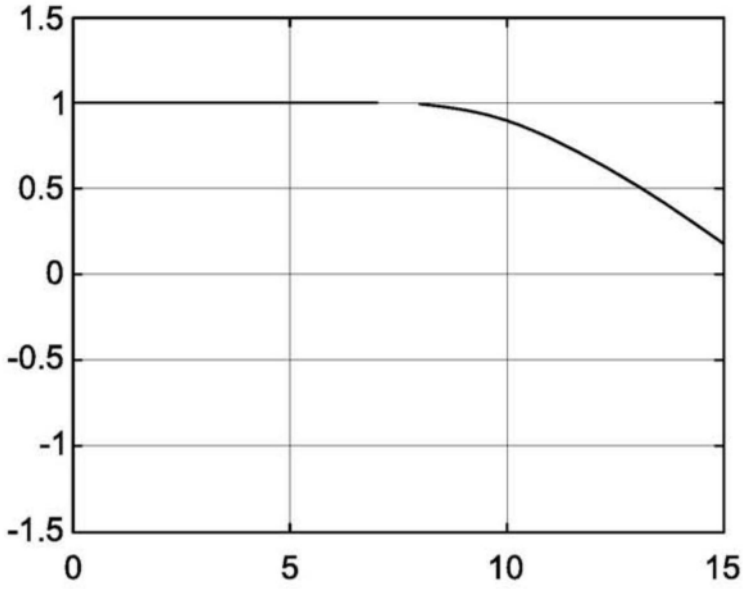


(a)

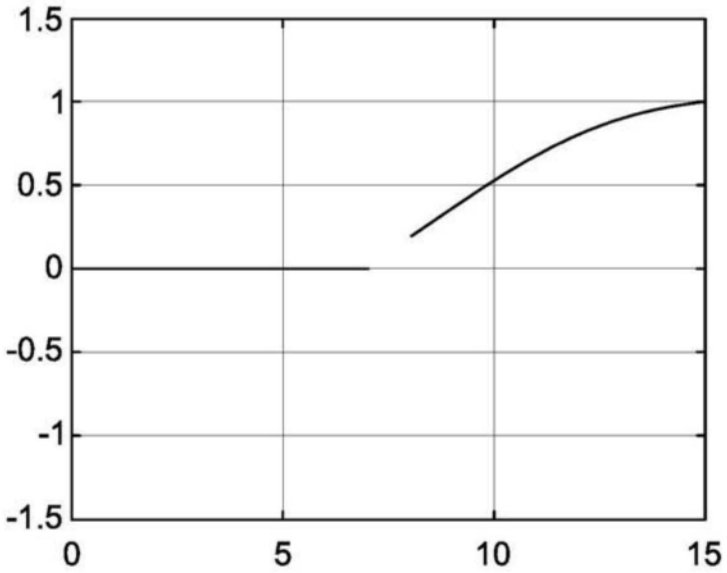


(b)

图7



(a)



(b)

图8

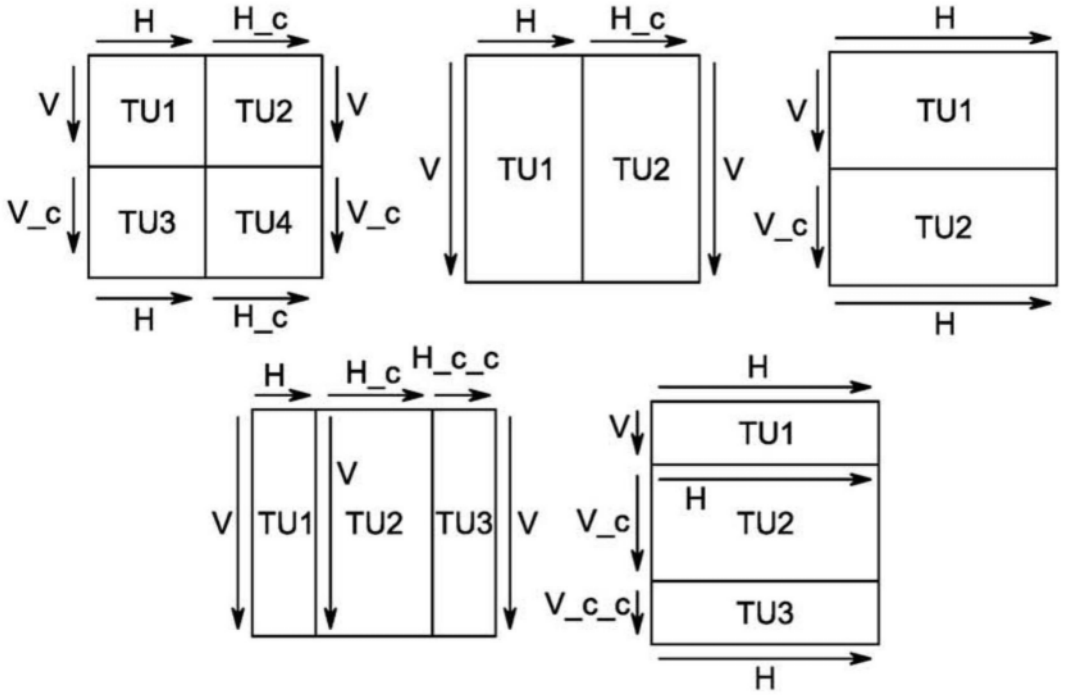

H
H_c
V
TU1
TU2
V_c
TU3
TU4
H_c_c
TU1
TU2
TU3

图9

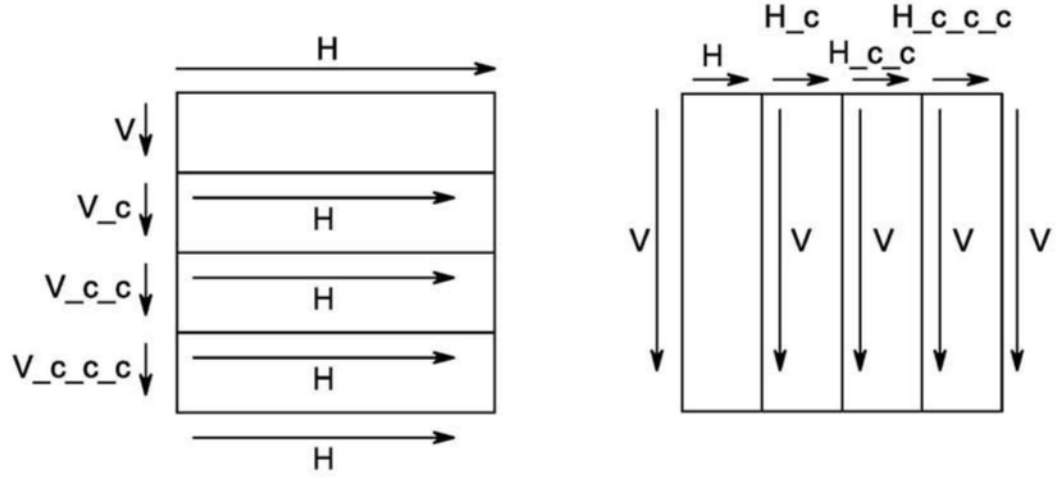

H
V
V_c
V_c_c
V_c_c_c
H_c
H_c_c_c
H_c_c

图10

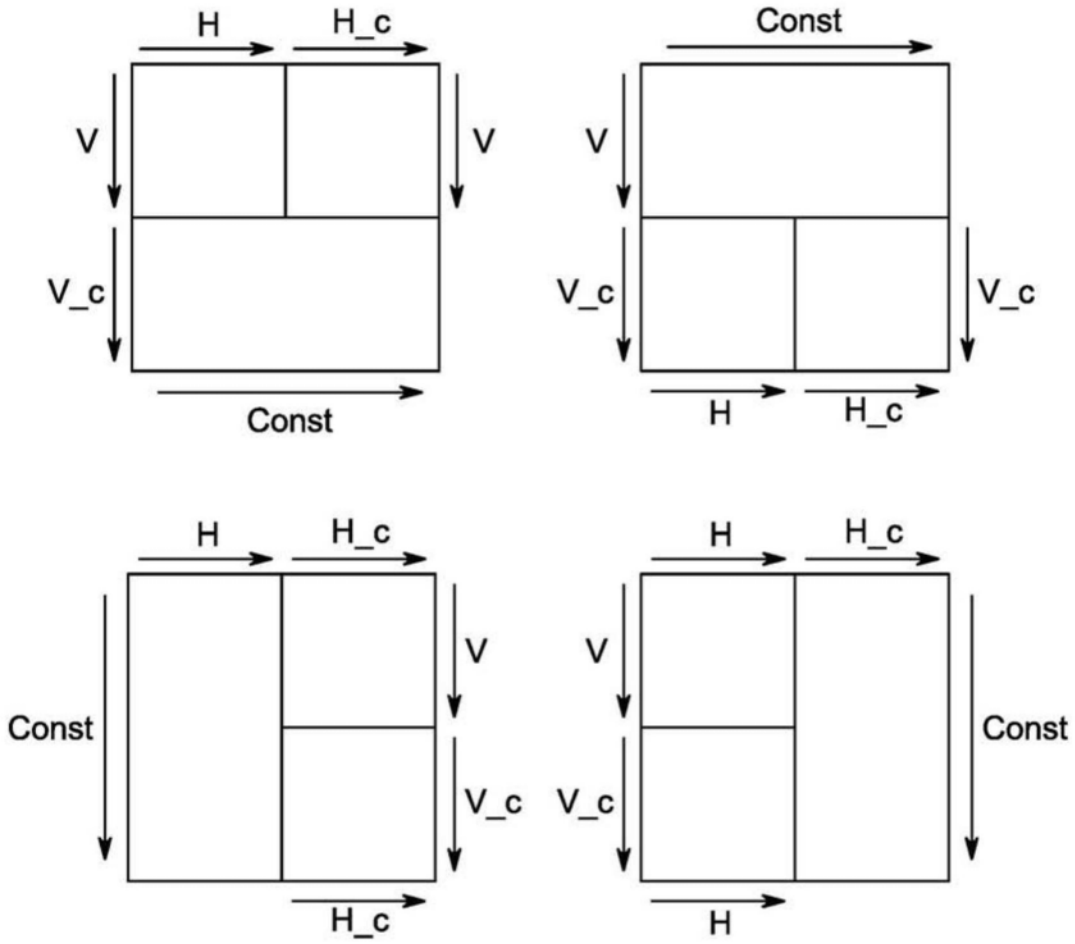

H
H_c
V
V
V_c
Const
Const
V
V_c
V_c
H
H_c
H
H_c
Const
V
V_c
H_c
H
H_c
V
V_c
Const
H

图11

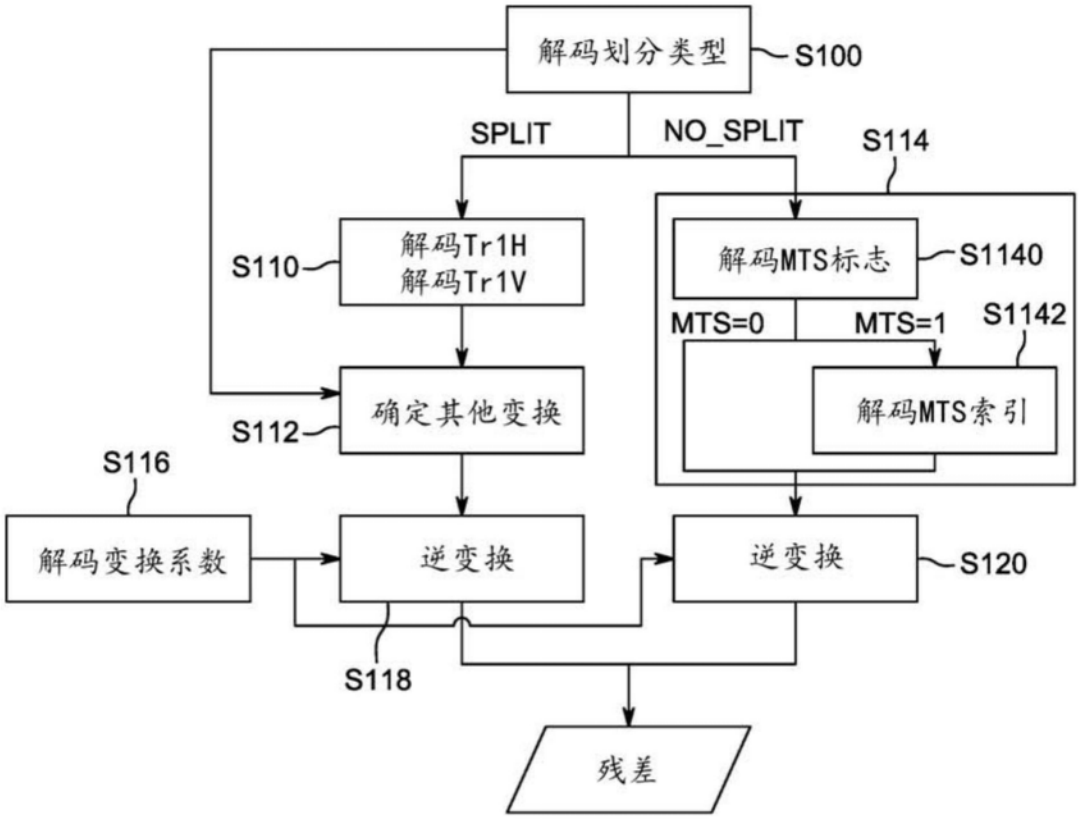

解码划分类型
S100
SPLIT
NO_SPLIT
S114
解码Tr1H
解码Tr1V
S110
解码MTS标志
S1140
MTS=0
MTS=1
S1142
确定其他变换
S112
解码MTS索引
S116
解码变换系数
逆变换
逆变换
S120
S118
残差

图12

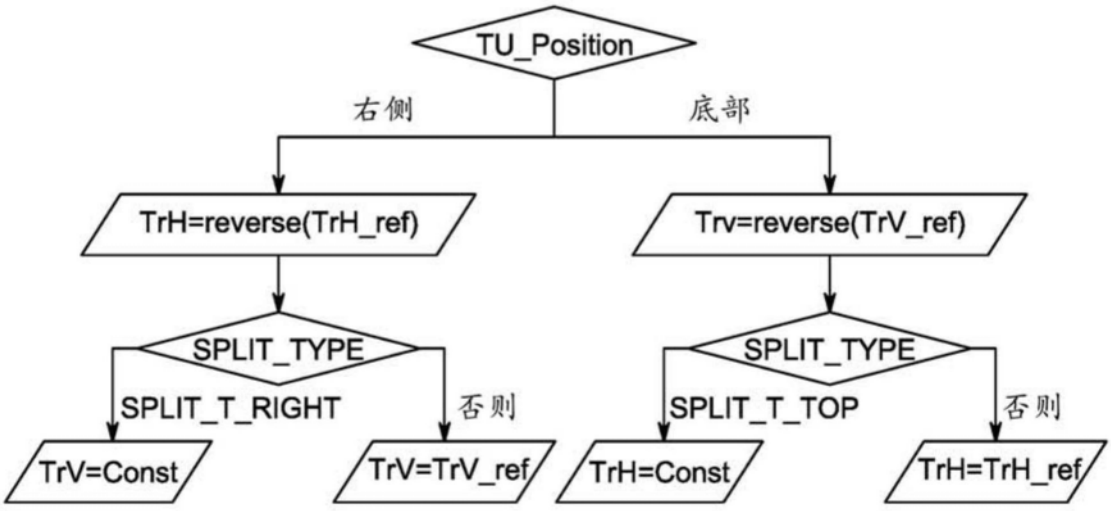

TU_Position
右侧
底部
TrH=reverse(TrH_ref)
Trv=reverse(TrV_ref)
SPLIT_TYPE
SPLIT_TYPE
SPLIT_T_RIGHT
否则
SPLIT_T_TOP
否则
TrV=Const
TrV=TrV_ref
TrH=Const
TrH=TrH_ref

图13

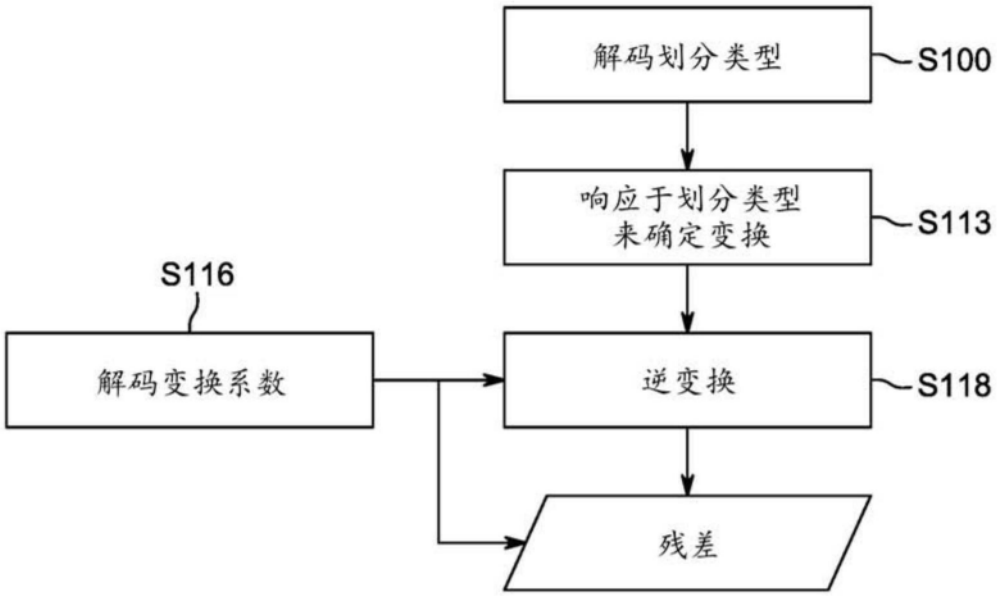

解码划分类型
S100
响应于划分类型
来确定变换
S113
S116
解码变换系数
逆变换
S118
残差

图14

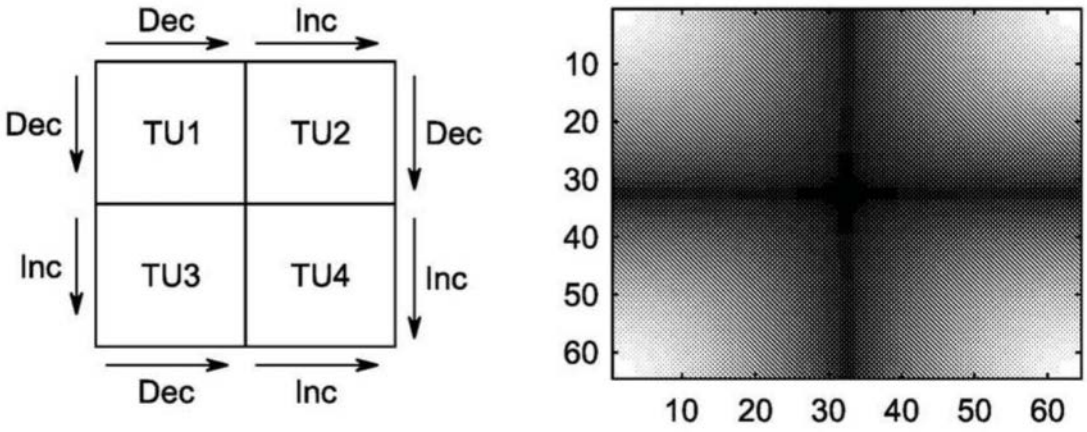

Dec
Inc
Dec
TU1
TU2
Dec
Inc
TU3
TU4
Inc
Dec
Inc
10
20
30
40
50
60
10
20
30
40
50
60

图15

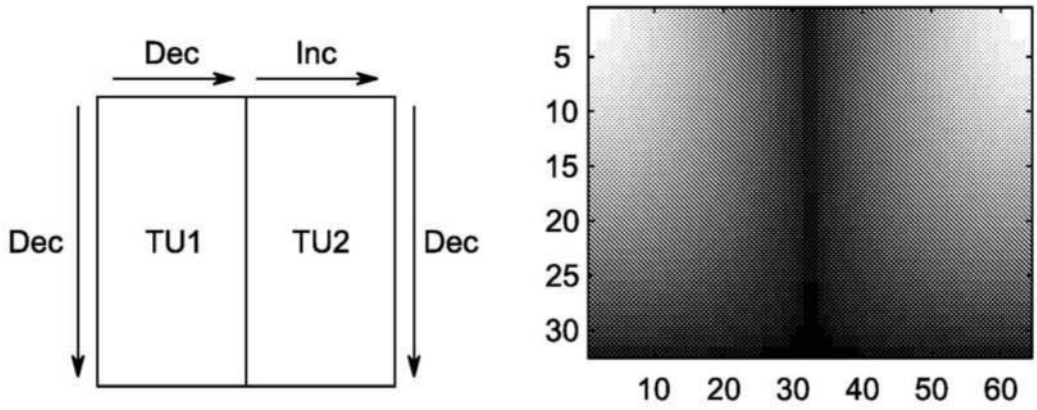

Dec
Inc
Dec
TU1
TU2
Dec
5
10
15
20
25
30
10
20
30
40
50
60

图16

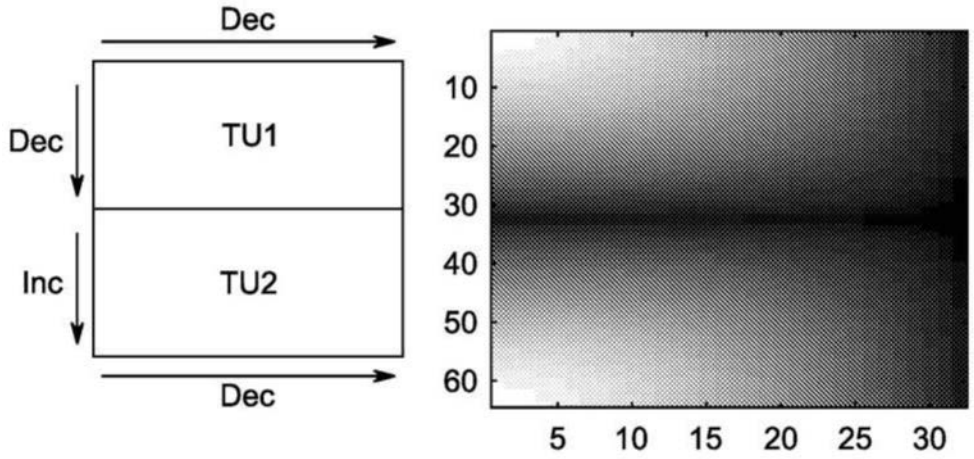

Dec
Dec
TU1
Inc
TU2
Dec
10
20
30
40
50
60
5
10
15
20
25
30

图17

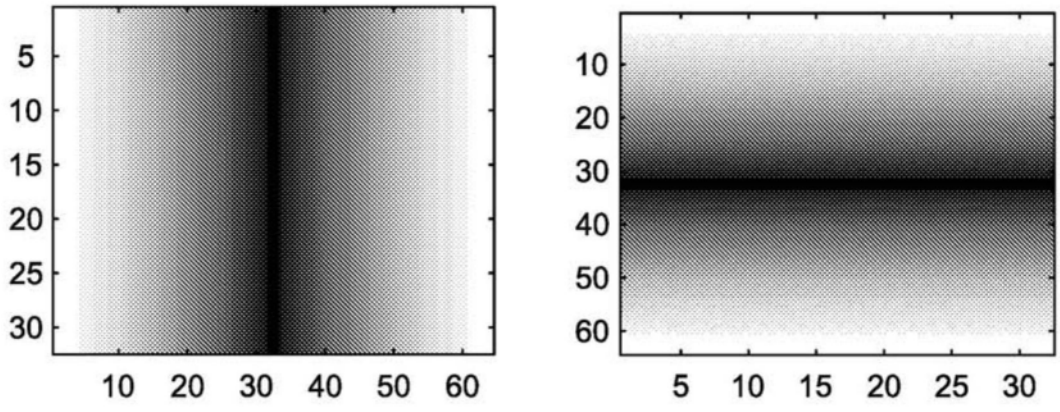

5
10
15
20
25
30
10
20
30
40
50
60
10
20
30
40
50
60
5
10
15
20
25
30

图18

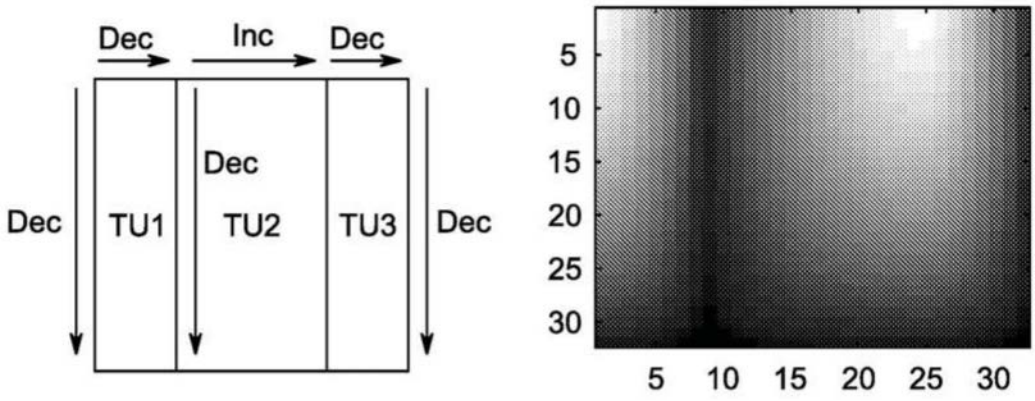

Dec
Inc
Dec
Dec
Dec
TU1
TU2
TU3
Dec
5
10
15
20
25
30
5
10
15
20
25
30

图19

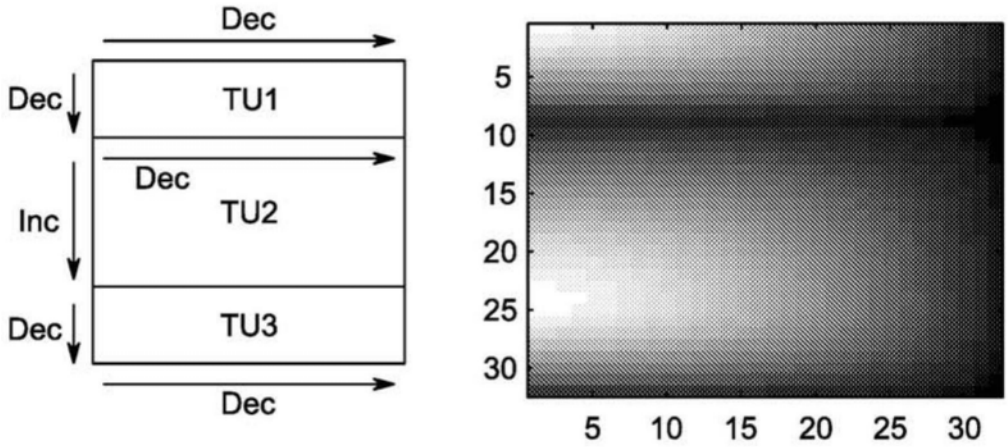

Dec
Dec
TU1
Dec
Inc
TU2
Dec
TU3
Dec
5
10
15
20
25
30
5
10
15
20
25
30

图20

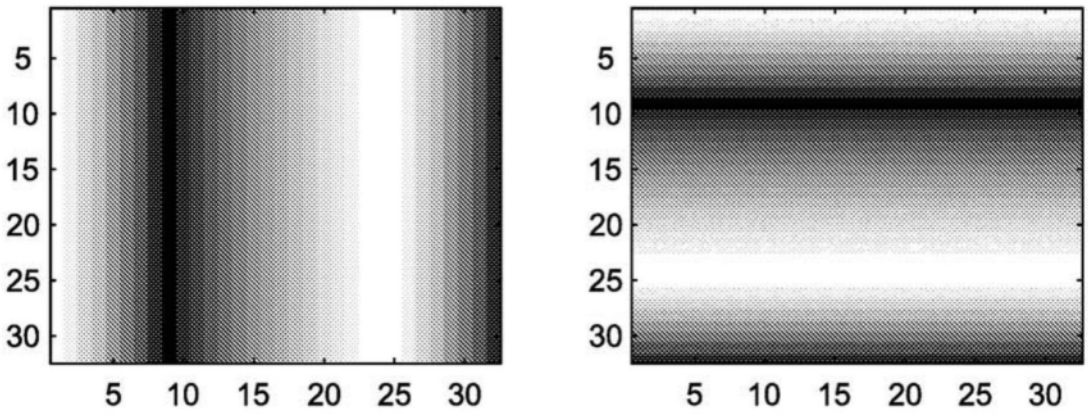

5
10
15
20
25
30
5
10
15
20
25
30
5
10
15
20
25
30
5
10
15
20
25
30

图21

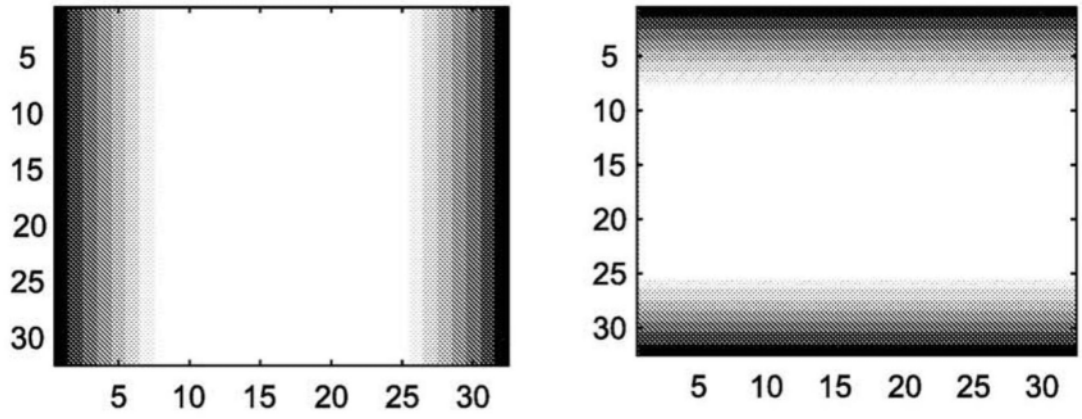

5
10
15
20
25
30
5 10 15 20 25 30
5
10
15
20
25
30
5 10 15 20 25 30

图22

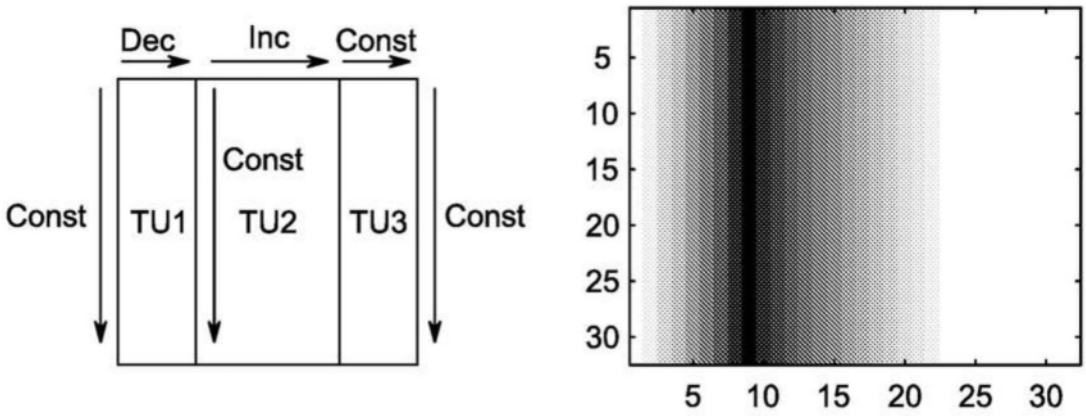

Dec
Inc
Const
Const
Const
TU1
TU2
TU3
Const
5
10
15
20
25
30
5 10 15 20 25 30

图23

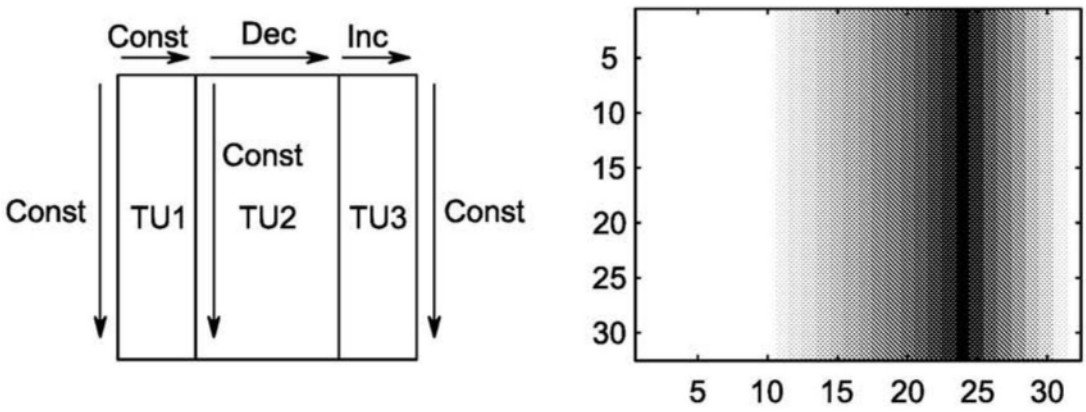

Const
Dec
Inc
Const
Const
TU1
TU2
TU3
Const
5
10
15
20
25
30
5 10 15 20 25 30

图24

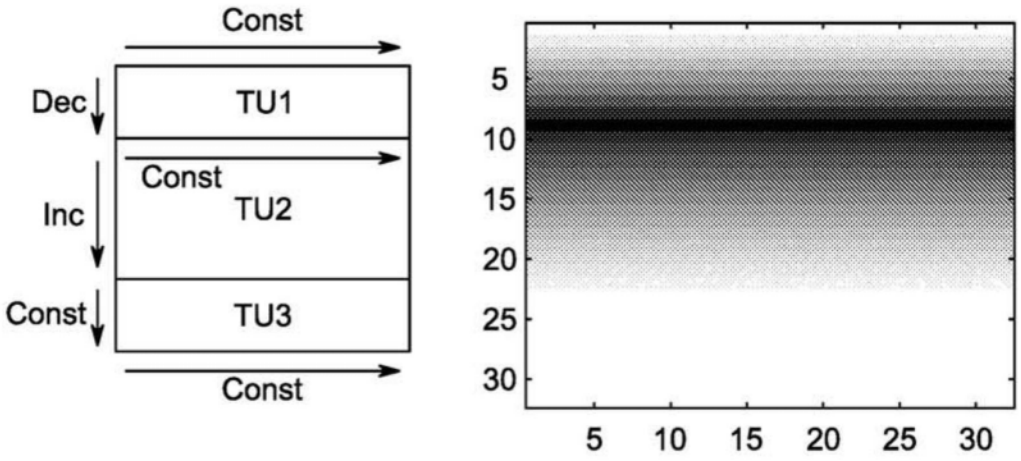

Const
Dec
TU1
Const
Inc
TU2
Const
TU3
Const
5
10
15
20
25
30
5
10
15
20
25
30

图25

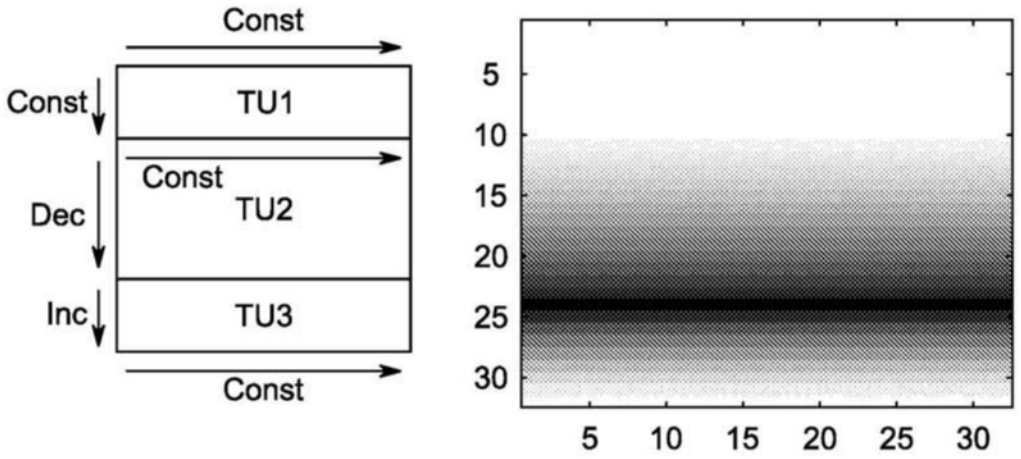

Const
Const
TU1
Const
Dec
TU2
Inc
TU3
Const
5
10
15
20
25
30
5
10
15
20
25
30

图26

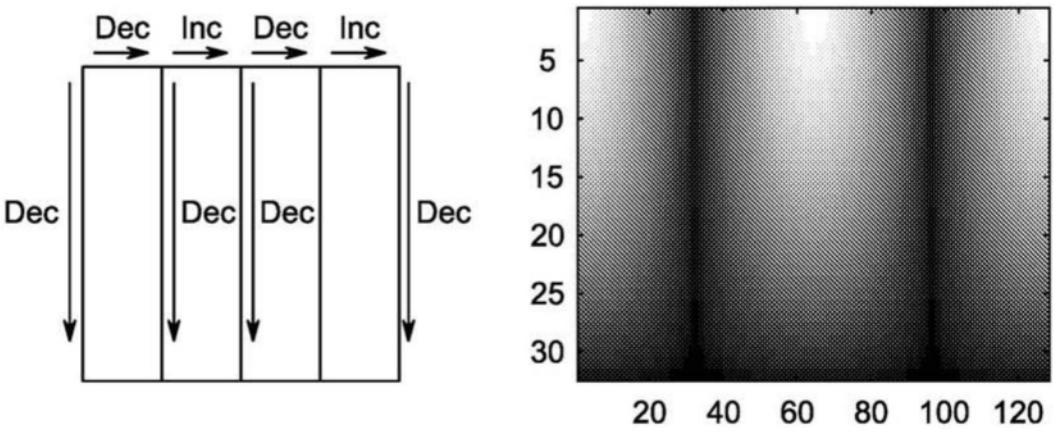

Dec
Inc
Dec
Inc
Dec
Dec
Dec
Dec
5
10
15
20
25
30
20
40
60
80
100
120

图27

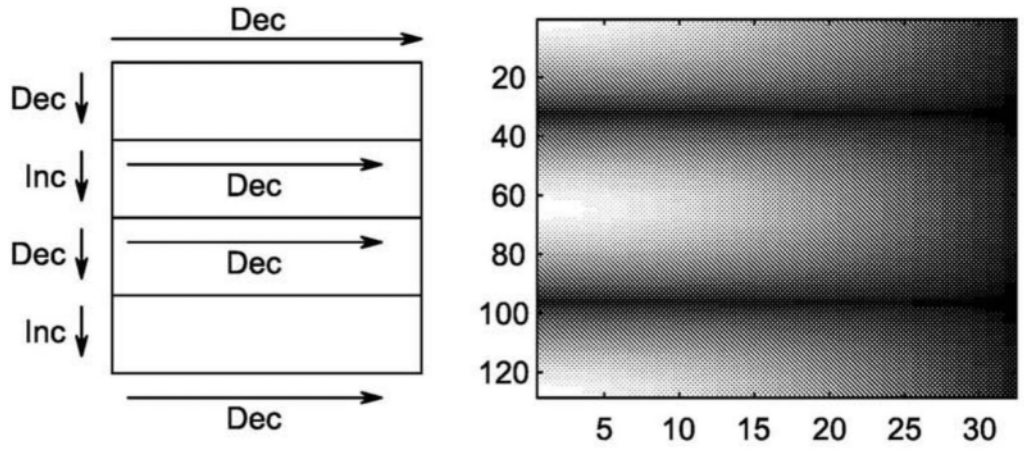

Dec
Dec
Inc
Dec
Dec
Dec
Inc
Dec
20
40
60
80
100
120
5
10
15
20
25
30

图28

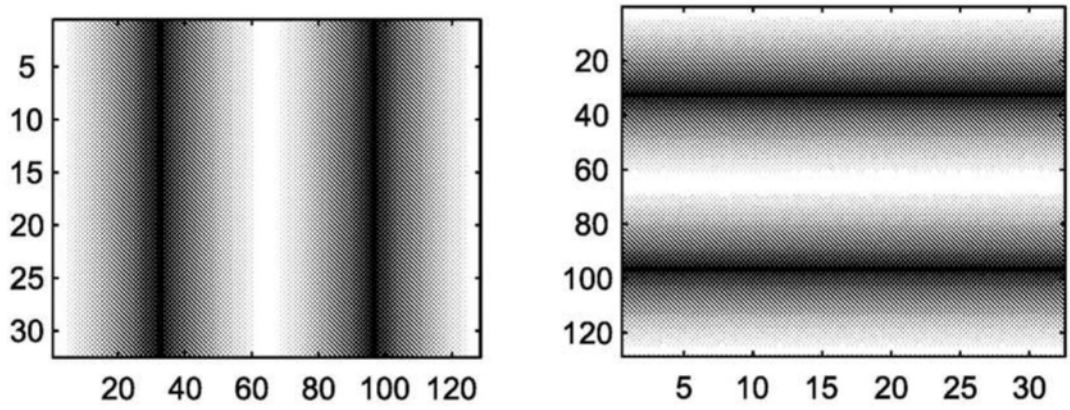

5
10
15
20
25
30
20
40
60
80
100
120
20
40
60
80
100
120
5
10
15
20
25
30

图29

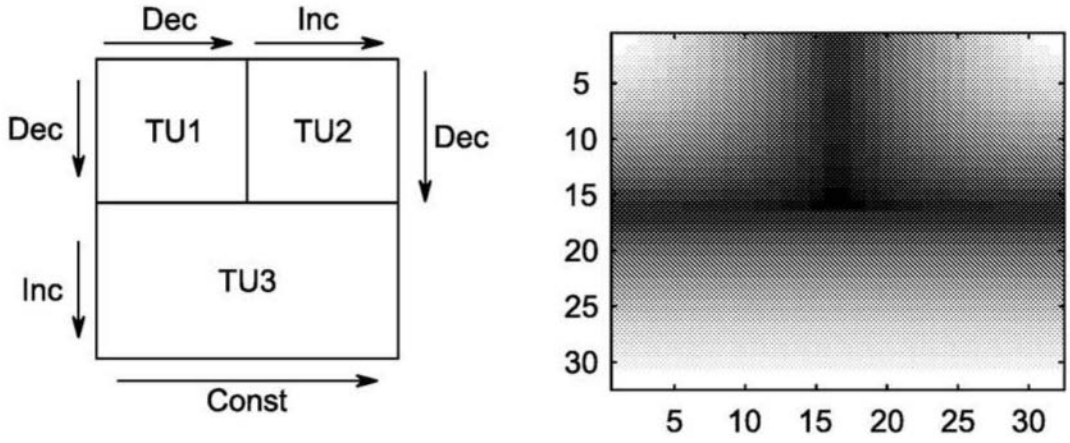

Dec
Inc
Dec
TU1
TU2
Dec
Inc
TU3
Const
5
10
15
20
25
30
5
10
15
20
25
30

图30

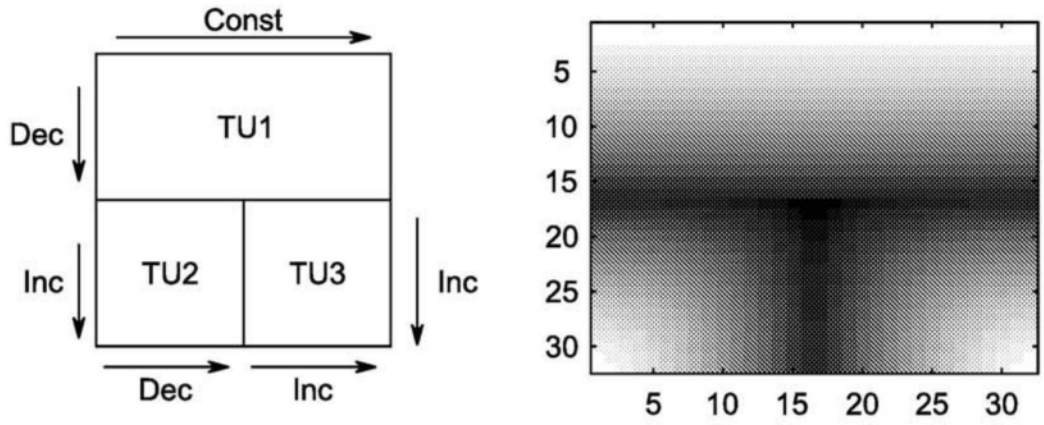

Const
Dec
TU1
TU2
TU3
Inc
Inc
Dec
Inc
5
10
15
20
25
30
5
10
15
20
25
30

图31

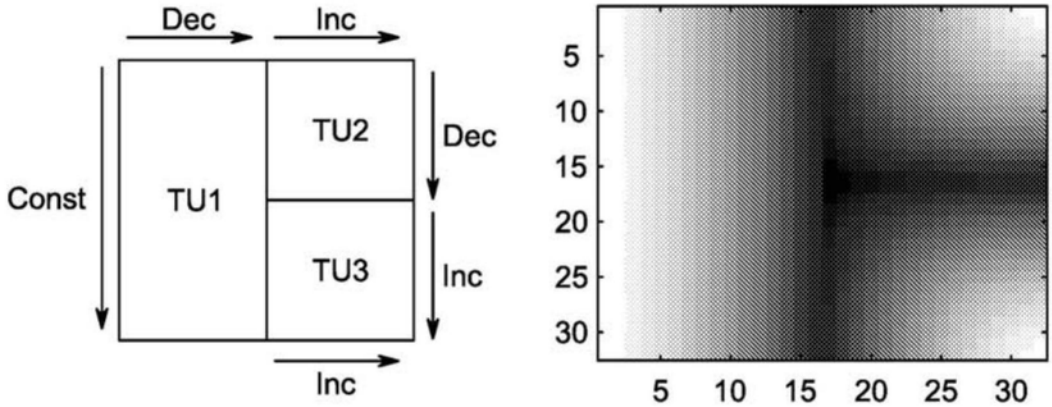

Dec
Inc
Const
TU1
TU2
TU3
Dec
Inc
Inc
5
10
15
20
25
30
5
10
15
20
25
30

图32

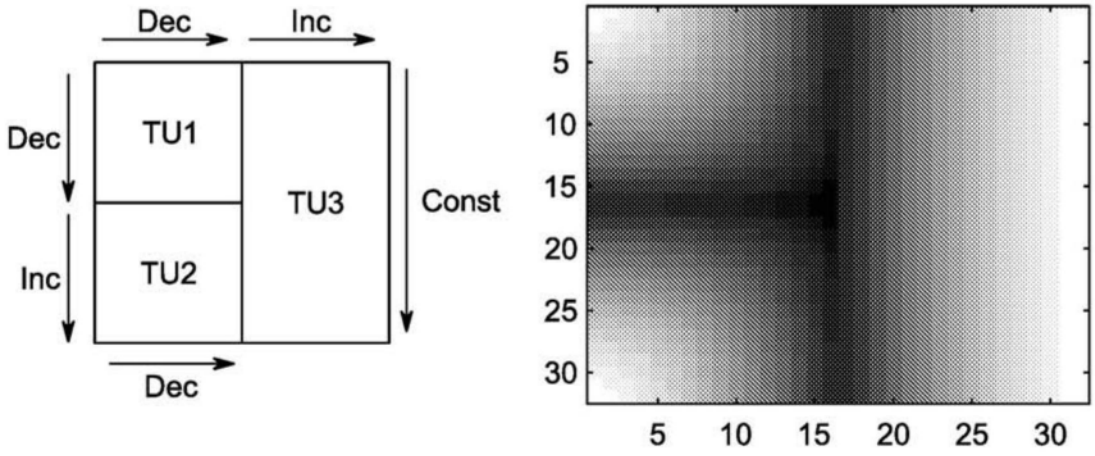

Dec
Inc
Dec
TU1
TU3
Const
Inc
TU2
Dec
5
10
15
20
25
30
5
10
15
20
25
30

图33

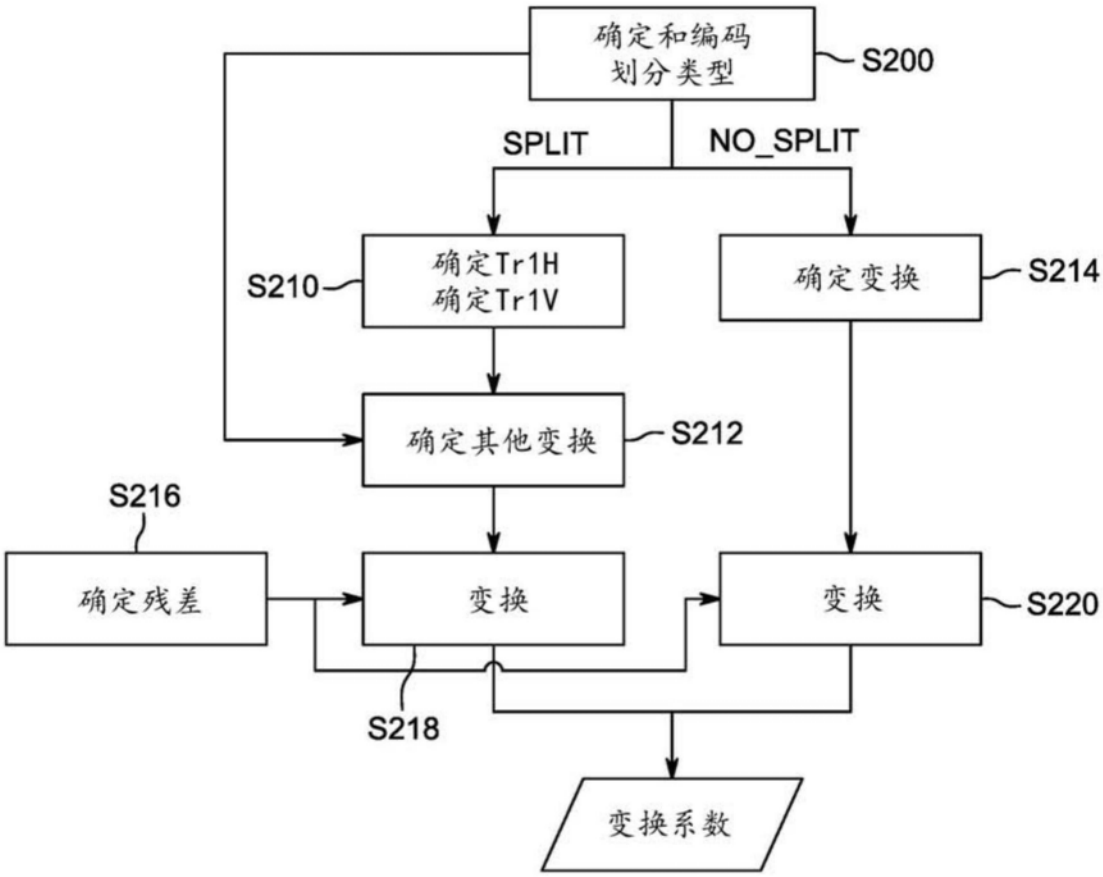

确定和编码
划分类型
S200
SPLIT
NO_SPLIT
S210
确定Tr1H
确定Tr1V
确定变换
S214
确定其他变换
S212
S216
确定残差
变换
变换
S220
S218
变换系数

图34

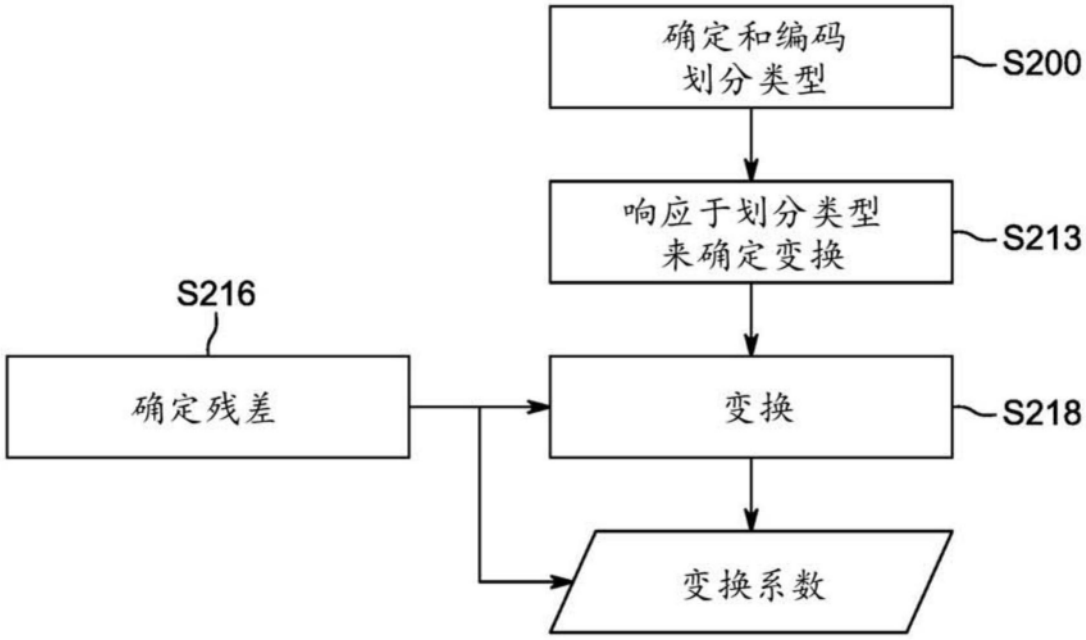

确定和编码
划分类型
S200
响应于划分类型
来确定变换
S213
S216
确定残差
变换
S218
变换系数

图35

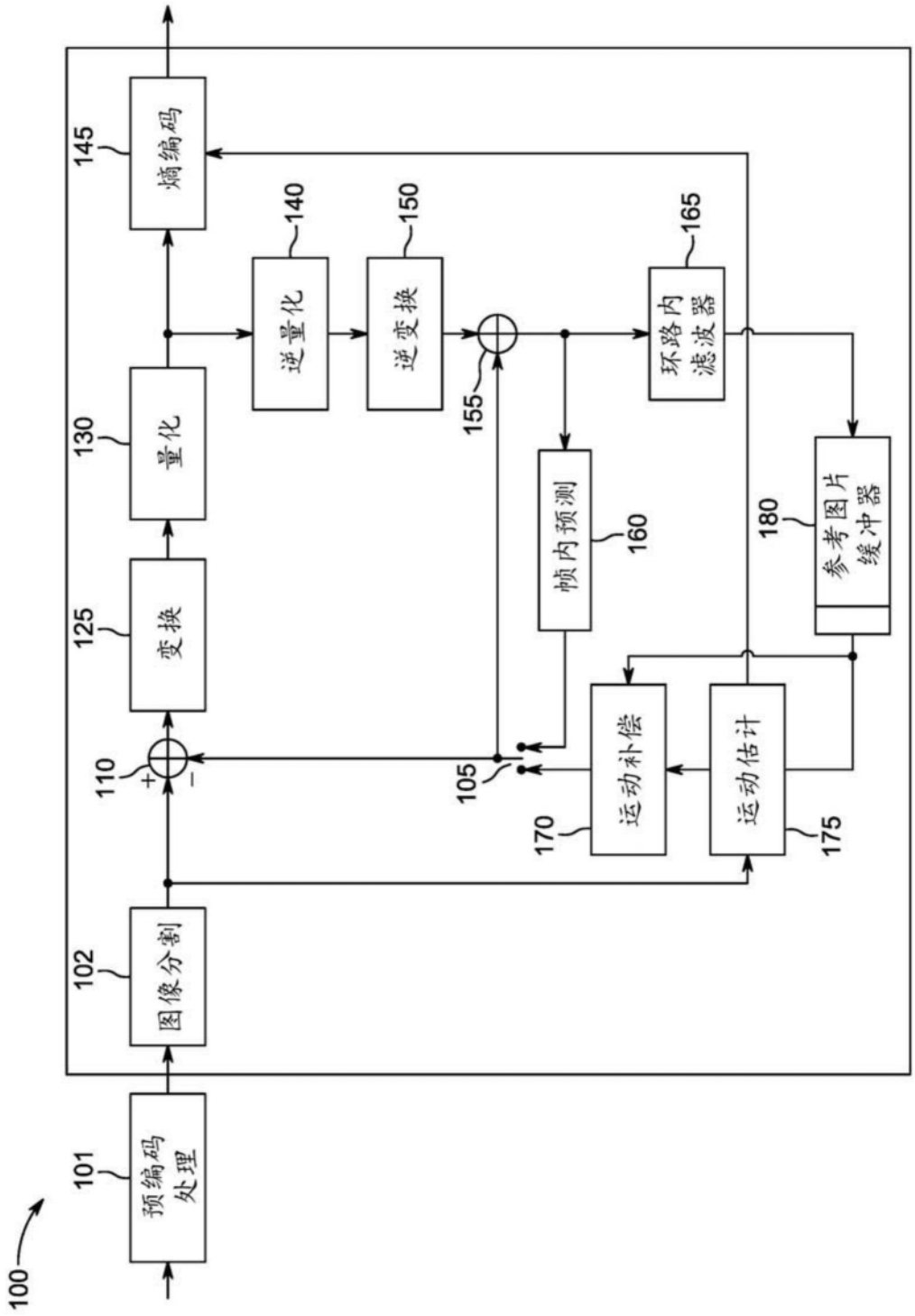

100
101
预编码处理
102
图像分割
110
+
−
125
变换
130
量化
145
熵编码
140
逆量化
150
逆变换
155
165
环路内滤波器
180
参考图片缓冲器
160
帧内预测
105
170
运动补偿
175
运动估计

图36

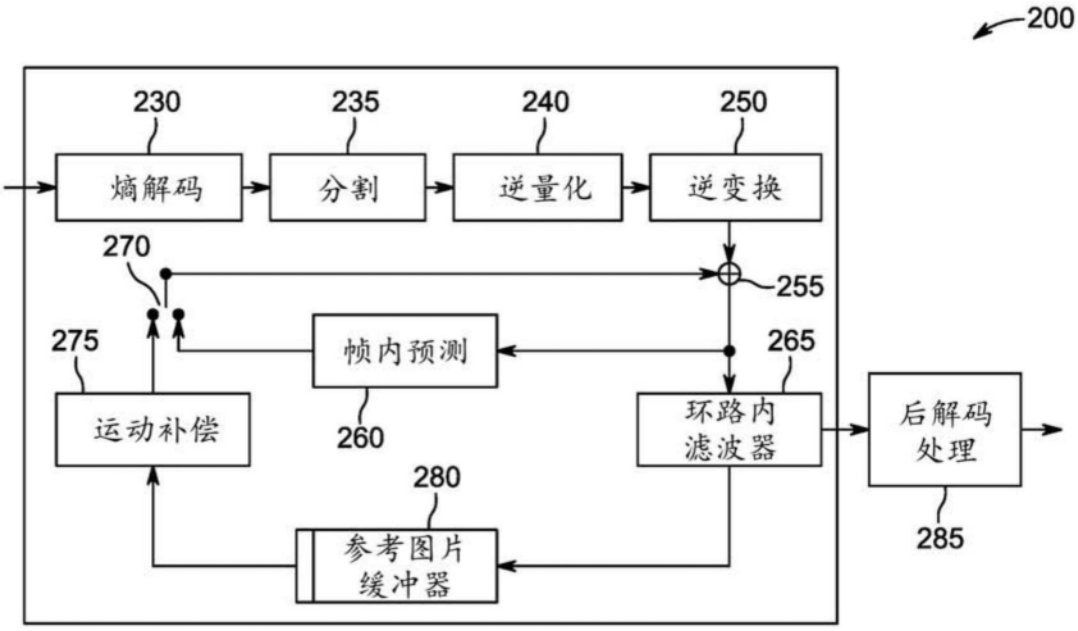

200
230
235
240
250
熵解码
分割
逆量化
逆变换
270
255
275
帧内预测
265
运动补偿
260
环路内
滤波器
后解码
处理
280
285
参考图片
缓冲器

图37

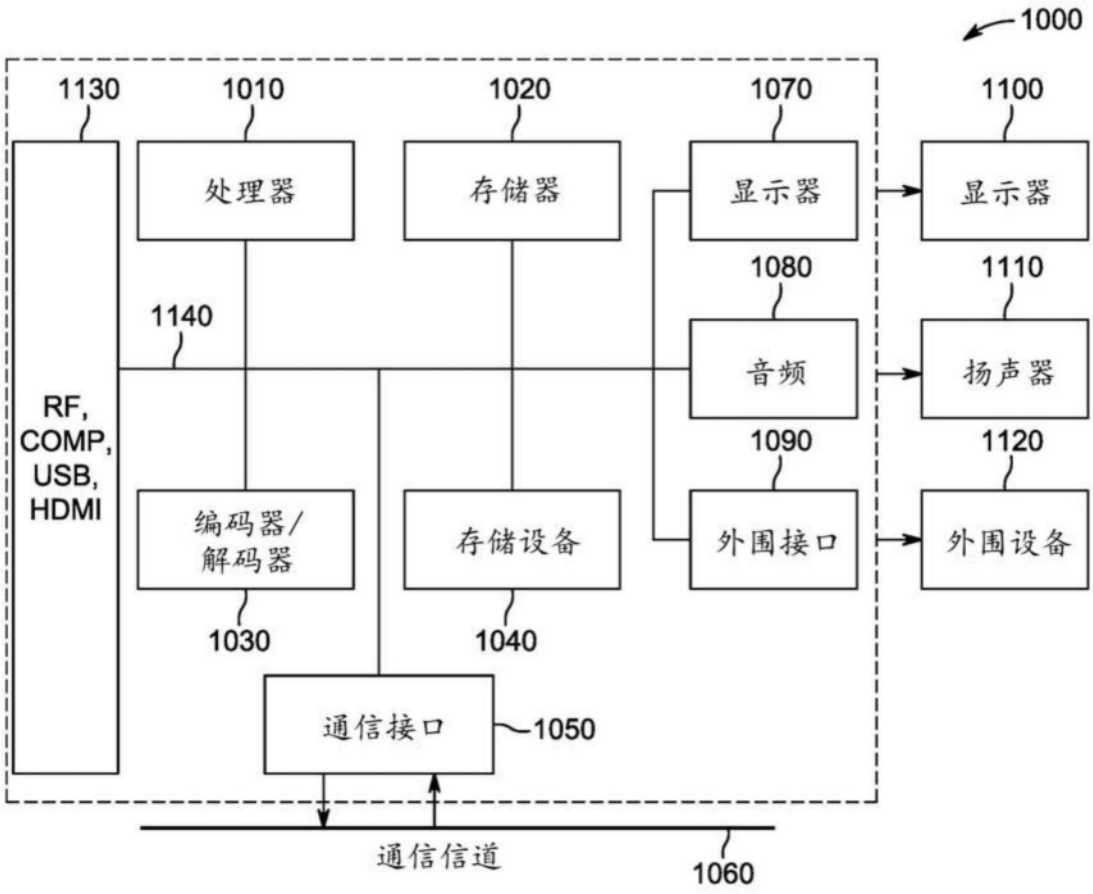

1000
1130
1010
1020
1070
1100
处理器
存储器
显示器
显示器
1080
1110
1140
音频
扬声器
RF,
COMP,
USB,
HDMI
1090
1120
编码器/
解码器
存储设备
外围接口
外围设备
1030
1040
通信接口
1050
通信信道
1060

图38