



## (12) 发明专利申请

(10) 申请公布号 CN 111903133 A

(43) 申请公布日 2020.11.06

(21) 申请号 201980021364.9

(22) 申请日 2019.02.15

(30) 优先权数据

62/631,047 2018.02.15 US

(85) PCT国际申请进入国家阶段日

2020.09.23

(86) PCT国际申请的申请数据

PCT/US2019/018279 2019.02.15

(87) PCT国际申请的公布数据

W02019/161248 EN 2019.08.22

(71) 申请人 艾锐势有限责任公司

地址 美国佐治亚州

(72) 发明人 克里特·帕努索波内 王利民

(74) 专利代理机构 中原信达知识产权代理有限公司 11219

代理人 张伟峰 夏凯

(51) Int.Cl.

H04N 19/543 (2006.01)

H04N 19/587 (2006.01)

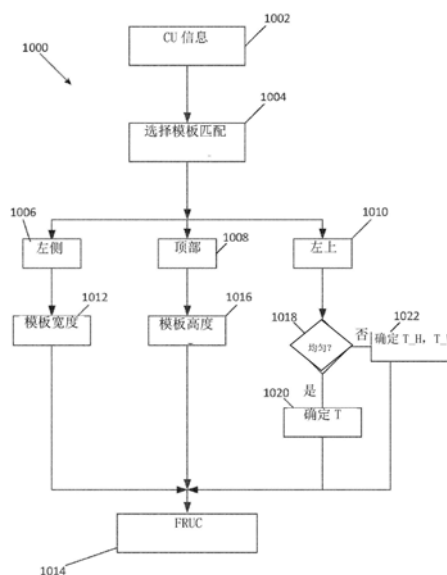
权利要求书2页 说明书15页 附图10页

### (54) 发明名称

用于模板匹配的可变模板尺寸

### (57) 摘要

公开了一种帧间编码系统和方法,其中采用了可变尺寸的模板匹配。可以定义顶部模板、左侧模板和左上模板,其中,顶部模板的宽度等于编码块的宽度,左侧模板的高度等于编码块的高度,但模板的第二维度(高度和宽度)是可变的。接着识别当前编码块之间的最佳匹配,然后使用FRUC对编码块进行编码。



1. 一种帧间编码方法,包括:  
识别编码单元;  
确定与编码单元相关联的信息;  
定义与所述编码单元相邻的像素的编码模板,其中,所述编码模板至少部分地基于所述编码单元的宽度和高度中的至少一者;以及  
至少部分地基于所述编码模板对所述编码单元编码。
2. 根据权利要求1所述的帧间编码方法,其中,所述编码模板由位于所述编码单元左侧的像素构成。
3. 根据权利要求2所述的帧间编码方法,其中,所述编码模板具有等于所述编码单元的所述高度的高度。
4. 根据权利要求3所述的帧间编码方法,其中,所述编码模板具有等于或小于所述编码单元的所述宽度的宽度。
5. 根据权利要求4所述的帧间编码方法,其中,所述编码模板的所述宽度是可变的。
6. 根据权利要求1所述的帧间编码方法,其中,所述编码模板由位于所述编码单元上方的像素构成。
7. 根据权利要求6所述的帧间编码方法,其中,所述编码模板具有等于所述编码单元的所述宽度的宽度。
8. 根据权利要求7所述的帧间编码方法,其中,所述编码模板具有等于或小于所述编码单元的所述高度的高度。
9. 根据权利要求8所述的帧间编码方法,其中,所述编码模板的所述高度是可变的。
10. 根据权利要求1所述的帧间编码方法,其中,所述编码模板由位于所述编码单元上方和左侧的像素构成。
11. 根据权利要求10所述的帧间编码方法,其中,所述编码模板具有等于或小于所述编码单元的所述高度的厚度。
12. 根据权利要求11所述的帧间编码方法,其中,所述编码模板具有等于或小于所述编码单元的所述宽度的厚度。
13. 根据权利要求12所述的帧间编码方法,其中,所述编码模板的所述厚度是可变的。
14. 一种帧间编码系统,包括:  
在存储器中接收编码单元;  
确定与编码单元相关联的信息并且将其存储在存储器中;  
定义与所述编码单元相邻的像素的编码模板并且将其存储在存储器中,其中,所述编码模板至少部分地基于所述编码单元的宽度和高度中的至少一者;以及  
至少部分地基于所述编码模板在利用帧速率上转换的信号中对所述编码单元编码。
15. 根据权利要求14所述的帧间编码系统,其中,所述编码模板由位于所述编码单元左侧的像素构成;  
其中,所述编码模板具有等于所述编码单元的所述高度的高度;并且  
其中,所述编码模板具有等于或小于所述编码单元的所述宽度的宽度。
16. 根据权利要求14所述的帧间编码系统,其中,所述编码模板由位于所述编码单元上方的像素构成;

其中,所述编码模板具有等于所述编码单元的所述宽度的宽度;并且

其中,所述编码模板具有等于或小于所述编码单元的所述高度的高度。

17.根据权利要求14所述的帧间编码系统,其中,所述编码模板由位于所述编码单元上方和左侧的像素构成;并且

其中,所述编码模板具有等于或小于所述编码单元的所述高度的厚度。

18.根据权利要求17所述的帧间编码系统,其中,所述编码模板具有等于或小于所述编码单元的所述宽度的厚度。

19.根据权利要求17所述的帧间编码系统,其中,所述编码模板的所述厚度是可变的。

## 用于模板匹配的可变模板尺寸

[0001] 优先权要求

[0002] 本申请依据35 U.S.C.§119(e) 要求2018年2月15日提交的较早提交的美国临时申请序列号62/631,047的优先权,其全部内容在此通过引用并入本文。

### 技术领域

[0003] 本公开涉及视频编码领域,尤其涉及与使用模板匹配相关联的编码效率提高,其中模板尺寸可以变化。

### 背景技术

[0004] 不断发展的视频编码标准的技术改进展现出提高编码效率的趋势,以实现更高的比特率、更高的分辨率和更好的视频质量。联合视频探索团队开发了一种被称为JVET的新型视频编码方案,并正在开发一种被称为多功能视频编码(VVC)的更新型视频编码方案—由JVET于2018年10月1日发布的标题为多功能视频编码(草案2)的该标准的草案2的VVC第7版的完整内容据此以引用方式并入本文。类似于像HEVC(高效视频编码)的其他视频编码方案,JVET和VVC都是基于块的混合空域和时域预测编码方案。然而,相对于HEVC,JVET和VVC包括对比特流结构、语法、约束条件以及用于生成解码图片的映射的很多修改。JVET已经在联合探索模型(JEM)编码器和解码器中实施,但VVC预计要到2020年初才能实现。

### 发明内容

[0005] 一个或多个计算机的系统可被配置成通过在该系统上安装软件、固件、硬件或它们的组合来执行特定操作或动作,所述软件、固件、硬件或它们的组合在操作中使该系统执行特定所述动作。一个或多个计算机程序可被配置为通过包括指令来执行特定操作或动作,所述指令当由数据处理装置执行时使该装置执行所述动作。一个一般性方面包括:识别编码单元;确定与编码单元相关联的信息;定义与所述编码单元相邻的像素的编码模板,其中,所述编码模板至少部分地基于所述编码单元的宽度和高度中的至少一者。该方法还包括至少部分地基于所述编码模板对所述编码单元进行编码。该方面的其他实施方案包括相应的计算机系统、装置和记录在一个或多个计算机存储设备上的计算机程序,它们中的每一个都被配置为执行方法的动作。

[0006] 各个实施方案可包括以下特征中一个或多个特征:该帧间编码方法,其中,所述编码模板由位于编码单元左侧的像素构成。该帧间编码方法,其中,所述编码模板具有等于所述编码单元的所述高度的高度。该帧间编码方法,其中,所述编码模板具有等于或小于所述编码单元的所述宽度的宽度。该帧间编码方法,其中,所述编码模板的所述宽度是可变的。该帧间编码方法,其中,所述编码模板由位于编码单元上方的像素构成。该帧间编码方法,其中,所述编码模板由位于编码单元上方和左侧的像素构成。所述技术的实施方式可包括硬件、方法或过程,或计算机可访问介质上的计算机软件。

[0007] 一个一般性方面包括一种帧间编码系统,其包括:在存储器中接收编码单元;确定

与编码单元相关联的信息并将其存储在存储器中;定义与所述编码单元相邻的像素的编码模板并将其存储在存储器中,其中,所述编码模板至少部分地基于所述编码单元的宽度和高度中的至少一者;以及至少部分地基于所述编码模板在利用帧速率上转换的信号中对所述编码单元进行编码。该方面的其他实施方案包括相应的计算机系统、装置和记录在一个或多个计算机存储设备上的计算机程序,它们中的每一个都被配置为执行方法的动作。

[0008] 附加或替代实施方案可包括以下特征中的一个或多个特征。该帧间编码系统,其中,所述编码模板由位于编码单元左侧的像素构成。该系统还可以包括条件,其中,所述编码模板具有等于所述编码单元的所述高度的高度。该系统还可以包括条件,其中,所述编码模板具有等于或小于所述编码单元的所述宽度的宽度。该帧间编码系统还可以包括条件,其中,所述编码模板可以由位于编码单元上方的像素构成,或者所述编码模板具有等于所述编码单元的所述宽度的宽度。所述技术的实施方案可包括硬件、方法或过程,或计算机可访问介质上的计算机软件。

## 附图说明

[0009] 借助于附图解释本发明的其他细节,在附图中:

[0010] 图1描绘了将帧划分成多个编码树单元(CTU)。

[0011] 图2a-2c描绘了CTU被示范性分割成编码单元(CU)。

[0012] 图3描绘了图2的CU分割的四叉树加二叉树(QTBT)表示。

[0013] 图4描绘了JVET或VVC编码器中用于CU编码的简化框图。

[0014] 图5描绘了用于JVET或VVC中的亮度分量的可能帧内预测模式。

[0015] 图6描绘了JVET或VVC解码器中用于CU编码的简化框图。

[0016] 图7描绘了编码单元以及具有可变高度/宽度的关联顶部和左侧模板的实施方案。

[0017] 图8-9描绘了编码单元与具有可变宽度/高度的关联顶部和左侧模板的替代实施方案。

[0018] 图10描绘了在编码中利用可变模板尺寸的方法的实施方案。

[0019] 图11描绘了适于并被配置为提供可变模板尺寸以用于模板匹配的计算机系统的实施方案。

[0020] 图12描绘了适于并被配置为提供可变模板尺寸以用于模板匹配的视频编码器/解码器的实施方案。

## 具体实施方式

[0021] 图1描绘了将帧划分成多个编码树单元(CTU) 100。帧可以是视频序列中的图像。帧可以包括矩阵或矩阵的集合,以像素值表示图像中的强度度量。因此,这些矩阵的集合可以生成视频序列。可以定义像素值以表示全色视频编码中的颜色和亮度,其中像素被划分成三个通道。例如,在YCbCr颜色空间中,像素可以具有表示图像中的灰度水平强度的亮度值Y,以及表示颜色从灰色到蓝色和红色的差异程度的两个色度值Cb和Cr。在其他实施方案中,可以用不同颜色空间或模型中的值来表示像素值。视频的分辨率可以确定帧中的像素的数量。更高的分辨率可能表示更多像素和更好的图像清晰度,但也可能导致更高的带宽、存储和传输需求。

[0022] 可以使用JVET对视频序列的帧进行编码和解码。JVET是联合视频探索团队开发的视频编码方案。已经在JEM(联合探索模型)编码器和解码器中实施了多个版本的JVET。类似于像HEVC(高效视频编码)的其他视频编码方案, JVET是一种基于块的混合空域和时域预测编码方案。在利用JVET进行编码期间, 首先将帧分成被称为CTU 100的正方形块, 如图1所示。例如, CTU 100可以是 $128 \times 128$ 像素的块。

[0023] 图2描绘了CTU 100被示范性地分割成CU 102。帧中的每个CTU 100可以被分割成一个或多个CU(编码单元) 102。如下所述, CU 102可以用于预测和变换。与HEVC不同的是, 在JVET中, CU 102可以是矩形或正方形, 并可以被编码而无需进一步分割成预测单元或变换单元。CU 102可以与其根CTU 100一样大, 或者是与 $4 \times 4$ 块一样小的根CTU 100的更小细分。

[0024] 在JVET中, 可以根据四叉树加二叉树(QTBT)方案将CTU 100分割成CU 102, 其中, 可以根据四叉树将CTU 100递归地划分成正方形块, 然后可以根据二叉树水平或垂直递归地划分那些正方形块。可以设置参数以根据QTBT控制划分, 这些参数是例如CTU尺寸、四叉树和二叉树叶节点的最小尺寸、二叉树根节点的最大尺寸, 以及二叉树的最大深度。在VVC中, 也可以利用三叉树划分将CTU 100分割成CU。

[0025] 作为非限制性示例, 图2a示出了分割成CU 102的CTU 100, 其中实线表示四叉树划分, 虚线表示二叉树划分。如图所示, 二叉树划分允许水平划分和垂直划分以定义CTU及其细分成CU的结构。图2b和2c描绘了CU的三叉树划分的替代非限制性示例, 其中, CU的细分是不均等的。

[0026] 图3描绘了图2的分割的QTBT表示。四叉树根节点代表CTU 100, 其中四叉树部分中的每个子节点代表从正方形父块划分的四个正方形块中的一个。然后可以使用二叉树将四叉树叶节点代表的正方形块划分零次或多次, 其中四叉树叶节点为二叉树的根节点。在二叉树部分的每个层级, 可以垂直或水平地对块进行划分。设置为“0”的标志表示水平划分块, 而设置为“1”的标志表示垂直划分块。

[0027] 在四叉树划分和二叉树划分之后, 由QTBT的叶节点代表的块代表要例如使用帧间预测或帧内预测编码来编码的最终CU 102。对于用帧间预测编码的条带或完整帧, 可以为亮度和色度分量使用不同的分割结构。例如, 对于帧间条带, CU 102可以具有用于不同颜色分量的编码块(CB), 例如一个亮度CB和两个色度CB。对于用帧内预测编码的条带或完整帧, 对于亮度和色度分量, 分割结构可以是相同的。

[0028] 图4描绘了JVET编码器中用于CU编码的简化框图。视频编码的主要阶段包括: 如上所述的分割以识别CU 102, 接着在404或406处使用预测对CU 102编码, 在408处生成残差CU 410, 在412处进行变换, 在416处进行量化, 以及在420处进行熵编码。图4中所示的编码器和编码过程还包括下文更详细描述的了解码过程。

[0029] 给定当前CU 102, 编码器可以在404处使用帧内预测在空域上或在406处使用帧间预测在时域上获得预测CU 402。预测编码的基本理念是在初始信号与针对初始信号的预测之间传输差分或残余信号。在接收机侧, 可以通过将残余和预测相加来重构初始信号, 如下文将描述的。因为差分信号的相关性低于原始信号, 所以其传输所需的比特更少。

[0030] 完全用帧内预测CU编码的条带, 例如, 整个图片或图片的一部分, 可以是无需参考其他条带就可以编码的I条带, 并且因此可以是解码能够开始的可能点。用至少一些帧间预测CU编码的条带可以是基于一个或多个参考图片解码的预测(P)或双向预测(B)条带。

P条带可以与先前编码的条带一起使用帧内预测和帧间预测。例如,可以使用帧间预测对P条带进行比I条带更进一步的压缩,但需要先前编码的条带的编码来对它们进行编码。B条带可以使用帧内预测或帧间预测,使用来自两个不同帧的插值预测,来使用来自先前和/或后续条带的数据进行其编码,从而提高运动估计过程的精确度。在一些情况下,也可以或可以替代地使用帧内块复制来对P条带和B条带进行编码,其中,使用来自相同条带的其他部分的数据。

[0031] 如下文将要论述的,可以基于从先前编码的CU 102 (例如,相邻CU 102或参考图片中的CU 102) 重构的CU 434进行帧内预测或帧间预测。

[0032] 当在404处用帧内预测在空域上对CU 102进行编码时,可以找到一种帧内预测模式,该模式基于来自图片中的相邻CU 102的样点来最佳地预测CU 102的像素值。

[0033] 在对CU的亮度分量编码时,编码器可以生成候选帧内预测模式的列表。尽管HEVC对于亮度分量具有35种可能的帧内预测模式,但在JVET中,对于亮度分量有67种可能的帧内预测模式,在VVC中,有85种预测模式。这些模式包括平面模式、DC模式、图5中所示的65种定向模式,以及18种宽角预测模式,平面模式使用从相邻像素生成的值的三维平面,DC模式使用相邻像素的平均值,定向模式使用沿实线指示的方向从相邻像素复制的值,宽角预测模式可以与非正方形块一起使用。

[0034] 在为CU的亮度分量生成候选帧内预测模式的列表时,列表上的候选模式的数量可以取决于CU的尺寸。候选列表可以包括:具有最低SATD (绝对变换差之和) 成本的HEVC的35种模式的子集;为JVET添加的与从HEVC模式发现的候选相邻的新定向模式;以及来自基于用于先前编码的相邻块的帧内预测模式识别的CU 102的六个最可能模式 (MPM) 的集合以及默认模式列表中的模式。

[0035] 在对CU的色度分量编码时,也可以生成候选帧内预测模式的列表。候选模式的列表可以包括从亮度样点利用跨分量线性模型投影生成的模式、为亮度CB,尤其是色度块中的共位位置发现的帧内预测模式,以及先前为相邻块发现的色度预测模式。编码器可以找到列表上具有最低速率失真成本的候选模式,并在对CU的亮度和色度分量编码时使用那些帧内预测模式。可以在指示用于对每个CU 102进行编码的帧内预测模式的比特流中对语法进行编码。

[0036] 在已经选择了用于CU 102的最佳帧内预测模式之后,编码器可以使用那些模式生成预测CU 402。当选定模式是定向模式时,可以使用4抽头滤波器来提高定向精确度。可以利用边界预测滤波器,例如2抽头或3抽头滤波器调节预测块的顶部或左侧的列或行。

[0037] 可以利用位置相关帧内预测组合 (PDPC) 过程对预测CU 402进行进一步平滑化,该过程使用相邻块的未滤波样点调节基于相邻块的滤波样点生成的预测CU 402,或者使用3抽头或5抽头低通滤波器进行自适应参考样点平滑化以处理参考样点。

[0038] 当在406处利用帧间预测在时域上对CU 102编码时,可以找到一组运动矢量 (MV), 其指向参考图片中对CU 102的像素值做出最佳预测的样点。帧间预测通过表示条带中像素块的位移来利用条带之间的时间冗余性。通过称为运动补偿的过程,根据先前或之后条带中的像素值确定位移。可以在比特流中向解码器提供表示相对于特定参考图片的像素位移的运动矢量和关联的参考索引,连带提供初始像素与经运动补偿的像素之间的残差。解码器可以使用残差和信令通知的运动矢量和参考索引来在重构条带中重构像素块。

[0039] 在JVET中,可以以1/16像素的精确度存储运动矢量,并可以利用四分之一像素或整数像素分辨率对运动矢量与CU的预测运动矢量之间的差值编码。

[0040] 在JVET中,可以使用各种技术为CU 102内的多个子CU找到运动矢量,所述技术是例如高级时域运动矢量预测(ATMVP)、空时运动矢量预测(STMVP)、仿射运动补偿预测、模式匹配的运动矢量推导(PMMVD)和/或双向光流(BIO)。

[0041] 使用ATMVP,编码器可以为CU 102找到指向参考图片中对应块的时域矢量。可以基于为先前编码的相邻CU 102找到的运动矢量和参考图片找到时域矢量。使用整个CU 102的时域矢量指向的参考块,可以为CU 102内的每个子CU找到运动矢量。

[0042] STMVP可以通过对为先前利用帧间预测编码的相邻块找到的运动矢量进行缩放和平均,来为子CU找到运动矢量,并一起找到时域矢量。

[0043] 可以使用仿射运动补偿预测,基于为块的顶角找到的两个控制运动矢量,为该块中的每个子CU预测运动矢量场。例如,可以基于为CU 102内的每个 $4 \times 4$ 块找到的顶角运动矢量,推导出子CU的运动矢量。

[0044] PMMVD可以使用双边匹配或模板匹配为当前CU 102找到初始运动矢量。双边匹配可以沿运动轨迹查看两个不同参考图片中的当前CU 102和参考块,而模板匹配可以查看当前CU 102和参考图片中由模板标识的对应块。然后可以针对每个子CU逐个细化为CU 102找到的初始运动矢量。

[0045] 在利用双向预测基于更早和更晚参考图片执行帧间预测时可以使用BIO,并且BIO允许基于两个参考图片之间的差值梯度为子CU找到运动矢量。

[0046] 在一些情况下,可以在CU级别使用局部照明补偿(LIC),以基于与当前CU 102相邻的样点和与候选运动矢量标识的参考块相邻的对应样点,找到缩放因子参数和偏移参数的值。在JVET中,LIC参数可以变化并在CU级别被信令通知。

[0047] 对于以上方法中的一些,可以在CU级别将为CU的每个子CU找到的运动矢量用信号发送到解码器。对于其他方法,例如PMMVD和BIO,不在比特流中信令通知运动信息以节省开销,并且解码器可以通过相同过程推导出运动矢量。

[0048] 在已经找到了CU 102的运动矢量之后,编码器可以使用那些运动矢量生成预测CU 402。在一些情况下,在已经为各个子CU找到了运动矢量时,在通过组合那些运动矢量与先前为一个或多个相邻子CU找到的运动矢量来生成预测CU 402时,可以使用重叠块运动补偿(OBMC)。

[0049] 在使用双向预测时,JVET可以使用解码器侧运动矢量细化(DMVR)来找到运动矢量。DMVR允许使用双向模板匹配过程,基于为双向预测找到的两个运动矢量来找到运动矢量。在DMVR中,可以找到利用两个运动矢量中的每个运动矢量生成的预测CU 402的加权组合,并且可以通过用最佳地指向组合的预测CU 402的新运动矢量替换这两个运动矢量来细化这两个运动矢量。可以使用两个细化的运动矢量来生成最终预测CU 402。

[0050] 在408处,如上所述,一旦已经在404处用帧内预测或在406处用帧间预测找到了预测CU 402,编码器就可以从当前CU 102减去预测CU 402,以找到残差CU 410。

[0051] 编码器可以在412处使用一个或多个变换操作来将残差CU 410变换成在变换域中表达残差CU 410的变换系数414,例如,使用离散余弦块变换(DCT变换)将数据转换到变换域中。与HEVC相比,JVET允许更多类型的变换操作,包括DCT-II、DST-VII、DST-VII、DCT-



VIII、DST-I和DCT-V操作。可以将允许的变换操作分组成子集,并可以由编码器信令通知使用了哪些子集以及那些子集中的哪些特定操作的指示。在一些情况下,可以使用大块尺寸变换来将大于某个尺寸的CU 102中的高频变换系数归零,使得仅为那些CU 102保持低频变换系数。

[0052] 在一些情况下,可以在正向内核变换之后向低频变换系数414应用模式相关的非独立二次变换(MDNSST)。MDNSST操作可以使用基于旋转数据的Hypercube-Givens变换(HyGT)。在使用时,可以由编码器信令通知标识特定MDNSST操作的索引值。

[0053] 在416处,编码器可以将变换系数414量化成量化变换系数416。可以通过将系数值除以量化步长来计算每个系数的量化,该量化步长是从量化参数(QP)推导出的。在一些实施方案中, Qstep被定义为 $2^{(QP-4)/6}$ 。因为可以将高精度变换系数414转换成具有有限数量的可能值的量化变换系数416,所以量化可以有助于数据压缩。于是,变换系数的量化可以限制变换过程生成和发送的比特量。不过,尽管量化是有损操作,并且量化的损失不能恢复,但量化过程在重构序列的质量与表示该序列所需的信息量之间进行权衡。例如,较低的QP值可以产生质量更好的解码视频,尽管需要更大量的数据才能表示和传输。相反,高QP值可以产生质量较低的重构视频序列,但数据和带宽需求较低。

[0054] JVET可以利用基于方差的自适应量化技术,这允许每个CU 102为其编码过程使用不同的量化参数(而不是在对帧的每个CU 102编码时使用相同的帧QP)。基于方差的自适应量化技术自适应地降低某些块的量化参数,同时在其他块中增大量化参数。为了为CU 102选择特定QP,计算该CU的方差。简而言之,如果CU的方差高于帧的平均方差,可以为该CU 102设置比帧的QP更高的QP。如果CU 102呈现出比帧的平均方差更低的方差,则可以分配更低的QP。

[0055] 在420处,编码器可以通过对量化变换系数418进行熵编码来找到最终压缩比特422。熵编码旨在消除要传输的信息的统计冗余。在JVET中,可以使用CABAC(上下文自适应二进制算术编码)对量化变换系数418编码,该技术使用概率度量来消除统计冗余。对于具有非零量化变换系数418的CU 102,可以将量化变换系数418转换成二进制。然后可以使用上下文模型对二进制表示的每个比特(“二进制位”)编码。CU 102可以被分解成三个区域,每个区域具有其自己的一组上下文模型以用于该区域内的像素。

[0056] 可以执行多个扫描轮次以对二进制位进行编码。在对前三个二进制位(bin0、bin1和bin2)编码的轮次期间,可以通过找到该二进制位在多达五个由模板标识的先前编码的相邻量化变换系数418中的位置之和,来找到指示为该二进制位使用哪个上下文模型的索引值。

[0057] 上下文模型可以基于二进制位的值为“0”或“1”的概率。在对值进行编码时,可以基于遇到值“0”和“1”的实际数量来更新上下文模型中概率。尽管HEVC使用固定表格来针对每个新图片对上下文模型进行重新初始化,但在JVET中,可以基于为先前编码的帧间预测图片开发的上下文模型对用于新帧间预测图片的上下文模型的概率进行初始化。

[0058] 编码器可以产生比特流,该比特流包含残差CU 410的熵编码的比特422、诸如选定的帧内预测模式或运动矢量的预测信息、如何根据QTBT结构从CTU 100分割CU 102的指示符,和/或关于编码视频的其他信息。比特流可以由解码器解码,如下所述。

[0059] 除了使用量化变换系数418找到最终压缩比特422之外,编码器还可以通过遵循与

解码器将用来生成重构的CU 434的解码过程相同的解码过程,使用量化变换系数418来生成重构的CU 434。于是,一旦变换系数已经被编码器计算并量化,就可以将量化变换系数418传输到编码器中的解码环路。在量化CU的变换系数之后,解码环路允许编码器生成与解码器在解码过程中生成的相同的重构的CU 434。因此,在对新CU 102执行帧内预测或帧间预测时,编码器可以使用解码器会用于相邻CU 102或参考图片的相同的重构的CU 434。重构的CU 102、重构条带或完整的重构帧可以充当其他预测阶段的参考。

[0060] 在编码器的解码环路处(并且参见下文,对于解码器中相同的操作),为了获得重构图像的像素值,可以执行去量化过程。为了对帧进行去量化,例如,将帧的每个像素的量化值乘以量化步长,例如上述(Qstep),以获得重构的去量化变换系数426。例如,在图4中所示的解码过程中,在编码器中,可以在424处对残差CU 410的量化变换系数418进行去量化以找到去量化变换系数426。如果在编码期间执行MDNSST操作,则在去量化之后可以对该操作进行反向操作。

[0061] 在428处,可以例如通过向值应用DCT来获得重构图像,从而对去量化变换系数426进行逆变换以找到重构的残差CU 430。在432处,可以将重构的残差CU 430添加到在404处利用帧内预测或在406处利用帧间预测找到的对应预测CU 402,以便找到重构的CU 434。

[0062] 在436处,可以在图片级别或CU级别在解码过程期间(在编码器中,或者如下文所述,在解码器中)向重构数据应用一个或多个滤波器。例如,编码器可以应用去方块滤波器、样点自适应偏移(SAO)滤波器和/或自适应环形滤波器(ALF)。编码器的解码过程可以实现滤波器,以估计可以解决重构图像中的潜在人工痕迹的最佳滤波器参数并将其传输到解码器。这样的改进提高了重构视频的客观和主观质量。在去方块滤波中,可以修改子CU边界附近的像素,而在SAO中,可以使用边缘偏移或频带偏移分类修改CTU 100中的像素。JVET的ALF可以使用对于每个 $2 \times 2$ 块具有圆形对称形状的滤波器。可以信令通知用于每个 $2 \times 2$ 块的滤波器的尺寸和身份的指示。

[0063] 如果重构图片是参考图片,可以将它们存储在参考缓冲器438中,以在406处对将来的CU 102进行帧间预测。

[0064] 在以上步骤期间,JVET允许使用内容自适应裁剪操作来调整颜色值,以匹配在上裁剪边界与下裁剪边界之间。裁剪边界可以针对每个条带改变,并且可以在比特流中信令通知标识边界的参数。

[0065] 图6描绘了JVET解码器中用于CU编码的简化框图。JVET解码器可以接收包含关于已编码CU 102的信息的比特流。比特流可以指示如何根据QTBT结构从CTU 100分割出图片的CU 102,CU 102的预测信息(例如,帧内预测模式或运动矢量),以及表示熵编码残差CU的比特602。

[0066] 在604处,解码器可以使用编码器在比特流中信令通知的CABAC上下文模型对熵编码比特602解码。解码器可以使用编码器信令通知的参数,来以与在编码期间更新上下文模型的概率相同的方式更新上下文模型的概率。

[0067] 在604处对熵编码进行逆操作以找到量化变换系数606之后,解码器可以在608对它们进行去量化,以找到去量化变换系数610。如果在编码期间执行MDNSST操作,则在去量化之后可以由解码器对该操作进行逆操作。

[0068] 在612处,可以对去量化变换系数610进行逆变换以找到重构的残差CU 614。在616

处,可以将重构的残差CU 614添加到在622处利用帧内预测或在624处利用帧间预测找到的对应预测CU 626,以便找到重构的CU 618。

[0069] 在620处,可以在图片级别或CU级别向重构数据应用一个或多个滤波器。例如,解码器可以应用去方块滤波器、样点自适应偏移(SAO)滤波器和/或自适应环形滤波器(ALF)。如上所述,可以使用位于编码器的解码环路中的环内滤波器来估计最优滤波器参数,以提高帧的客观和主观质量。这些参数被传输到解码器以在620处对重构帧滤波,以与编码器中的经滤波的重构帧。

[0070] 在通过找到重构的CU 618并应用信令通知的滤波器生成重构图片之后,解码器可以输出重构图片作为输出视频628。如果重构图片要被用作参考图片,可以将它们存储在参考缓冲器630中,以在624处对将来的CU 102进行帧间预测。

[0071] 帧速率上转换(FRUC)是一种帧间编码工具。在使用FRUC模式对CU编码时,在解码器侧推导出其运动矢量。在比特流中包括信令通知以指示推导过程。与HEVC merge模式(其中,推导出的运动矢量(MV)限于候选MV列表内的MV)不同,FRUC通过避免表达MV信令通知来改善编码效率。具体而言,FRUC利用模式匹配运动矢量推导方法,该方法可以基于匹配成本从搜索窗口内的MV候选确定MV。在一些实施方案中,可以基于可以预定的FRUC模式和搜索模式指定匹配模式。因此,解码器可以遵循相同过程来推导FRUC MV。

[0072] 在一些实施方案中,FRUC可能有3种模式:AMVP(高级运动矢量预测子)模板匹配、Merge模板匹配和Merge双边匹配。可以将模板匹配模式用作AMVP模式的选项以确定CU的MV或用作merge模式的选项以确定CU的MV。对于模板匹配而言,可以将模板用作CU的代表,并且可以使用来自编码帧中的相邻块的重构像素来形成模板。在一些实施方案中,编码器和解码器都使用相同的搜索模式在参考帧中的搜索窗口内搜索候选模板。然后,可以将匹配最好的候选模板的偏移用作MV。

[0073] 双边匹配是可以用于merge模式以确定CU的MV的另一种FRUC模式。双边匹配可以采用来自两个参考帧的重构像素来确定MV,而不是像模板匹配中那样依赖于来自编码帧的重构像素来推导MV。在双边匹配的一些实施方案中,可以假设连续的运动轨迹,并且可以将指向匹配最好的块对的两个MV(在轨迹约束下)用作merge的MV。

[0074] 图7描绘了编码单元700以及具有可变高度/宽度的关联的顶部模板702和左侧模板704的实施方案。模板配置在使用模板匹配进行编码时发挥重要作用。图7描绘了在一些编码实施方案中使用的尺寸为W 706乘H 708的CU 700的模板配置。在一些实施方案中,模板可以包括两个部分,即顶部模板702和左侧模板704。可以使用来自与编码块或编码单元700的顶行相邻的相邻块的四行重构像素形成顶部模板702。在图7所示的实施方案中,顶部模板702可以具有与编码块/编码单元(CU)相同的宽度706。此外,在图7所示的实施方案中,可以使用来自与编码块(CU)的左侧列相邻的相邻块的四列重构像素形成左侧模板704,使得左侧模板704可以具有与编码块(CU)相同的高度。尽管图7描绘了具有4行的顶部模板702和具有4列的左侧模板704,但在其替代实施方案中,可以与顶部模板702和左侧模板704相关联地使用任何已知的、方便和/或期望数量的行列。

[0075] 在图7所示的实施方案中,采用与CU相关的模板配置作为在模板匹配中被用作CU的代表的模板配置。在一些实施方案中,模板可以具有类似于CU的特性以实现的高预测精确度。在模板尺寸过小的实施方案中,模板可能无法提供关于CU的重要细节。相反,大的模

板尺寸可以包括与CU无关的额外信息,并由于来自额外/不必要信息的“噪声”而导致不必要的系统负担和/或导致结果不佳。鉴于此,如在JEM7中所使用的,固定模板尺寸(顶部模板702为4行,且左侧模板704为4列)在相关性方面是次优的。于是,需要一种能够利用与CU中的特性匹配的可变模板尺寸的系统和方法。在一些实施方案中,模板(顶部模板702和/或左侧模板704)尺寸可以是完全灵活的。但是,应理解,完全尺寸灵活性可能需要大量的开销,这对于系统操作而言可能成本过高。在一些实施方案中,可以使用一些编码信息来确定模板尺寸。然而,在一些实施方案中,可以通过管理和/或减小尺寸确定步骤的复杂性来使系统负担最小化。在一些实施方案中,模板尺寸可以至少部分地基于编码块(CU)尺寸。亦即,在编码块(CU)尺寸小时,模板尺寸702、704也可以小,以减小包括错误或不必要信息的可能性。相反,在一些实施方案中,在编码块(CU)尺寸大时,模板尺寸702、704可以更大时,从而模板可以避免被约束在局部极小值中。

[0076] 在系统和方法的一些实施方案中,其中,CU具有尺寸W乘H,其中W是编码块706的宽度,H是编码块708的高度,顶部模板702尺寸可以被定义为W乘X,左侧模板704尺寸可以被定义为Y乘H。然而,替代实施方案可以包括并支持多种模板尺寸,如下面的方程所示,其中,顶部模板的高度X和左侧模板的宽度Y可以如下计算:

[0077]  $X = \text{VerSize1}$ , 在  $H < \text{VerThreshold}(1)$  时

[0078]  $X = \text{VerSize2}$ , 在  $H < \text{VerThreshold}(2)$  时

[0079]  $X = \text{VerSize3}$ , 在  $H < \text{VerThreshold}(3)$  时

[0080] ...

[0081]  $X = \text{VerSizeN}$ , 在  $H \geq \text{VerThreshold}(N-1)$  时

[0082] 以及

[0083]  $Y = \text{HorSize1}$ , 在  $W < \text{HorThreshold}(1)$  时

[0084]  $Y = \text{HorSize2}$ , 在  $W < \text{HorThreshold}(2)$  时

[0085]  $Y = \text{HorSize3}$ , 在  $W < \text{HorThreshold}(3)$  时

[0086] ...

[0087]  $Y = \text{HorSizeN}$ , 在  $W \geq \text{HorThreshold}(N-1)$  时

[0088] 其中, VerSize和HorSize分别是行和列的模板尺寸参数; VerThreshold和HorThreshold分别是行和列的编码块尺寸参数的阈值。

[0089] 在一些实施方案中,可以将HorSize1和VerSize1设置为1,可以将HorSize2和VerSize2设置为2,并且可以将HorSize3和VerSize3设置为3。在这样的配置中,可以将HorThreshold(1)和VerThreshold(1)设置为8,可以将HorThreshold(2)和VerThreshold(2)设置为16,并且可以将HorThreshold(3)和VerThreshold(3)设置为32。然而,在替代实施方案中,可以使用任何大于或小于32的已知的、方便的和/或期望的值。

[0090] 图8-9描绘了编码单元700与关联的左上模板802的替代实施方案。图8描绘了模板配置的示例,所述模板配置包括来自编码块的左上相邻块的重构像素,其中T是模板的厚度804。在图8所描绘的实施方案中,模板的宽度为W+T,模板的高度为H+T,模板尺寸可以是灵活的,并且也可以在方便和/或期望的情况下,根据W、H和T的值(可以被约束)应用。

[0091] 通过图9所描绘的非限制性示例,可以通过为宽度和高度采用不同的厚度参数来影响模板尺寸灵活性,其中,可以至少部分地基于编码块尺寸来确定参数。图9描绘了能够

实现这样的模板尺寸灵活性的模板的实施方案。在图9所描绘的实施方案中,  $T_W$ 代表厚度参数902,  $T_H$ 代表模板802的高度参数904。因此, 对于不同的编码块尺寸, 可以通过 $T_W$ 和 $T_H$ 定义具有参数的结构, 如下所示:

[0092]  $T_H = \text{VerSize1}$ , 在  $H < \text{VerThreshold}(1)$  时

[0093]  $T_H = \text{VerSize2}$ , 在  $H < \text{VerThreshold}(2)$  时

[0094]  $T_H = \text{VerSize3}$ , 在  $H < \text{VerThreshold}(3)$  时

[0095] ...

[0096]  $T_H = \text{VerSizeN}$ , 在  $H \geq \text{VerThreshold}(N-1)$  时

[0097] 以及

[0098]  $T_W = \text{HorSize1}$ , 在  $W < \text{HorThreshold}(1)$  时

[0099]  $T_W = \text{HorSize2}$ , 在  $W < \text{HorThreshold}(2)$  时

[0100]  $T_W = \text{HorSize3}$ , 在  $W < \text{HorThreshold}(3)$  时

[0101] ...

[0102]  $T_W = \text{HorSizeN}$ , 在  $W \geq \text{HorThreshold}(N-1)$  时

[0103] 其中,  $\text{VerSize}$ 和 $\text{HorSize}$ 分别是行和列的模板尺寸参数;  $\text{VerThreshold}$ 和 $\text{HorThreshold}$ 分别是行和列的编码块尺寸参数的阈值。

[0104] 作为非限制性示例, 在实施图9所描绘的系统和方法的一种可能配置中, 可以将 $\text{HorSize1}$ 和 $\text{VerSize1}$ 设置为1, 可以将 $\text{HorSize2}$ 和 $\text{VerSize2}$ 设置为2, 并可以将 $\text{HorSize3}$ 和 $\text{VerSize3}$ 设置为3。在这样的配置中, 可以将 $\text{HorThreshold}(1)$ 和 $\text{VerThreshold}(1)$ 设置为8, 可以将 $\text{HorThreshold}(2)$ 和 $\text{VerThreshold}(2)$ 设置为16, 并且可以将 $\text{HorThreshold}(3)$ 和 $\text{VerThreshold}(3)$ 设置为32。然而, 在替代实施方案中, 可以使用任何大于或小于32的已知的、方便的和/或期望的值。

[0105] 在一些实施方案中, 模板702、704、802的最小和最大尺寸可以至少部分地基于编码块(CU)的尺寸、与实施硬件相关联的约束条件、与可用带宽相关联的约束条件或传输约束条件和/或任何其他已知的、方便和/或期望的条件。作为非限制性示例, 在一些实施方案中, 模板702、704、802的模板最大尺寸可以固定在块尺寸的1/4处。然而, 在替代实施方案中, 可以使用任何已知的、方便的和/或期望的值。

[0106] 图10描绘了在编码1000中利用可变模板尺寸的方法的实施方案。在图10所描绘的实施方案中, 在步骤1002中获得编码单元信息。然后, 在步骤1004中, 判断要使用的模板是否是左侧模板1006、顶部模板1008和/或左上模板1010中的一个。在一些实施方案中, 确定使用哪个模板1006、1008、1010可以基于当前编码块(CU)与顶部模板1006、左侧模板1008和/或左上模板1010之间标准的最佳匹配。如果要使用左侧模板, 那么在步骤1012中, 可以确定模板的宽度, 并且该框可以进行到FRUC步骤1014。如果要使用顶部模板, 那么在步骤1016中, 可以确定模板的高度, 并且该框可以进行到FRUC步骤1014。如果确定要使用左上模板, 那么在步骤1018中, 可以确定左上模板是否具有均匀深度 $T$ 。如果要使用的左上模板具有均匀深度, 那么可以在步骤1020中定义该模板, 并且该框可以进行到FRUC步骤1014。如果在步骤1018中确定左上模板没有均匀深度, 那么可以在步骤1022中定义模板尺寸 $T_H$ 和 $T_W$ , 并且该框可以进行到FRUC步骤1014。

[0107] 实施实施方案所需的指令序列的执行可以由图11中所示的计算机系统1100执行。

在实施方案中,指令序列的执行是由单个计算机系统1100执行的。根据其他实施方案,通过通信链路1115耦合的两个或更多计算机系统1100可以彼此协调地执行指令序列。尽管下文将仅给出一个计算机系统1100的描述,但是,应当理解,可以采用任意数量的计算机系统1100来实践实施方案。

[0108] 现在将参考图11描述根据实施方案的计算机系统1100,该图是计算机系统1100的功能部件的框图。如本文所使用的,术语计算机系统1100被广泛地用于描述可以存储并独立运行一个或多个程序的任何计算设备。

[0109] 每个计算机系统1100可以包括耦合到总线1106的通信接口1114。通信接口1114在计算机系统1100之间提供双向通信。各个计算机系统1100的通信接口1114传输和接收电、电磁或光信号,这些信号包括代表各种类型的信号信息(例如,指令、消息和数据)的数据流。通信链路1115将一个计算机系统1100链接到另一个计算机系统1100。例如,通信链路1115可以是LAN,在这种情况下,通信接口1114可以是LAN卡;或者通信链路1115可以是PSTN,在这种情况下,通信接口1114可以是集成服务数字网络(ISDN)卡或调制解调器;或者通信链路1115可以是因特网,在这种情况下,通信接口1114可以是拨号、电缆或无线调制解调器。

[0110] 计算机系统1100可以通过其各自的通信链路1115和通信接口1114传输和接收消息、数据和指令,包括程序,即应用程序、代码。所接收的程序代码可以在被接收时由相应的处理器1107执行,和/或存储在存储设备1110中,或存储在其他关联的非易失性存储介质中,以用于稍后执行。

[0111] 在实施方案中,计算机系统1100协同数据存储系统1131一起工作,所述数据存储系统是例如包含容易被计算机系统1100访问的数据库1132的数据存储系统1131。计算机系统1100通过数据接口1133与数据存储系统1131通信。耦合到总线1106的数据接口1133传输和接收电、电磁或光信号,这些信号包括代表各种类型的信号信息(例如,指令、消息和数据)的数据流。在实施方案中,数据接口1133的功能可以由通信接口1114执行。

[0112] 计算机系统1100包括用于传送指令、消息和数据(统称为信息)的总线1106或其他通信机构,以及与总线1106耦合用于处理信息的一个或多个处理器1107。计算机系统1100还包括耦合到总线1106用于存储将由处理器1107执行的动态数据和指令的主存储器1108,诸如随机存取存储器(RAM)或其他动态存储设备。在由处理器1107执行指令期间,主存储器1108还可用于存储临时数据,即,变量或其他中间信息。

[0113] 计算机系统1100还可以包括耦合到总线1106用于存储处理器1107的静态数据和指令的只读存储器(ROM)1109或其他静态存储设备。存储设备1110(诸如,磁盘或光盘)也可以被提供并耦合到总线1106以用于存储处理器1107的数据和指令。

[0114] 计算机系统1100可以经由总线1106耦合到显示设备1111,例如但不限于,阴极射线管(CRT)或液晶显示器(LCD),以用于向用户显示信息。输入设备1112,例如数字字母键和其他按键,可以耦合到总线1106以用于向处理器1107传送信息和命令选择。

[0115] 根据一个实施方案,各个计算机系统1100通过各自的处理器1107执行包含在主存储器1108中的一个或多个指令的一个或多个序列而执行具体操作。此类指令可以从另一个计算机可用介质,诸如ROM 1109或存储设备1110,被读入主存储器1108。包含在主存储器1108中的指令的序列的执行使得处理器1107执行本文所述的过程。在替代实施方案中,硬

连线电路可以取代或结合软件指令使用。因此,实施方案不限于硬件电路和/或软件的任何特定组合。

[0116] 如本文所使用的,术语“计算机可用介质”是指提供信息或可由处理器1107使用的任何介质。这样的介质可以采取很多形式,包括但不限于,非易失性介质、易失性介质和传输介质。非易失性介质,即,在没电的情况下可以保持信息的介质,包括ROM 1109、CD ROM、磁带和磁盘。易失性介质,即,在没电的情况下不可以保持信息的介质,包括主存储器1108。传输介质包括同轴电缆、铜线和光纤,其包括具有总线1106的电线。传输介质还可以采取载波(即,可以在频率、幅度或相位方面被调制以传输信息信号的电磁波)的形式。此外,传输介质可以采取声波或光波(诸如,在无线电波和红外线数据通信期间产生的那些)的形式。

[0117] 在以上说明书中,已经参考各实施方案的具体元素描述了各实施方案。但是,将显而易见的是,在不脱离实施方案的更广泛的实质和范围的情况下,可对其进行各种修改和变更。例如,读者要理解,本文所述的过程流程图中所示的过程动作的具体排序和组合仅仅是示例性的,并且可以使用不同或额外的过程动作或过程动作的不同组合或排序来实践这些实施方案。因此,说明书和附图应被视为是示例性的而非限制性的。

[0118] 还应该指出的是,可以在各种计算机系统中实施本发明。本文所述的各种技术可以在硬件或软件或两者的组合中实现。优选地,在可编程计算机上执行的计算机程序中实施这些技术,所述可编程计算机各自包括处理器、可由处理器读取的存储介质(包括易失性和非易失性存储器和/或存储元件)、至少一个输入设备和至少一个输出设备。可以向使用输入设备输入的数据应用程序代码以执行上文描述的功能并产生输出信息。输出信息被应用到一个或多个输出设备。优选地以高阶程序编程语言或面向对象的编程语言实现每个程序,以与计算机系统通信。然而,如果需要的话,可以用汇编语言或机器语言来实现程序。在任何情况下,该语言可以是编译或解释语言。每个这样的计算机程序优选地存储在可由通用或专用可编程计算机读取的存储介质或设备(例如,ROM或磁盘)上,以在计算机读取存储介质或设备时配置并操作计算机以执行上述程序。还可以考虑将该系统实现为配置有计算机程序的计算机可读存储介质,其中如此配置的存储介质使计算机以特定的预定义方式运行。此外,示范性计算应用的存储元件可以是关系型或顺序(平坦文件)型计算数据库,其能够以各种组合和配置存储数据。

[0119] 图12是可以并入本文所述的系统和设备的特征的源设备1212和目的地设备1210的高水平视图。如图12中所示,示例性视频编码系统1210包括源设备1212和目的地设备1214,其中,在本示例中,源设备1212生成编码视频数据。因此,源设备1212可被称为视频编码设备。目的地设备1214可以对源设备1212生成的编码视频数据解码。因此,目的地设备1214可被称为视频解码设备。源设备1212和目的地设备1214可以是视频编码设备的示例。

[0120] 目的地设备1214可以经由信道1216从源设备1212接收编码视频数据。信道1216可以包括能够将编码视频数据从源设备1212移动到目的地设备1214的一种介质或设备。在一个示例中,信道1216可以包括通信介质,该通信介质使得源设备1212能够实时地将编码视频数据直接传输到目的地设备1214。

[0121] 在本示例中,源设备1212可以根据通信标准(例如,无线通信协议)调制编码视频数据,并且可以向目的地设备1214传输调制的视频数据。通信介质可以包括无线或有线通信介质,例如射频(RF)频谱或一个或多个物理传输线。通信介质可以形成诸如局域网、广域

网的基于分组的网络或诸如因特网的全球网络的一部分。通信介质可以包括路由器、交换机、基站或促成从源设备1212到目的地设备1214的通信的其他设备。在另一个示例中,信道1216可以对应于存储由源设备1212生成的编码视频数据的存储介质。

[0122] 在图12的示例中,源设备1212包括视频源1218、视频编码器1220和输出接口1228。在一些情况下,输出接口1228可以包括调制器/解调器(调制解调器)和/或发射器。在源设备1212中,视频源1218可以包括源,例如,视频捕获设备(例如摄像机)、包含先前捕获的视频数据的视频档案、从视频内容提供商接收视频数据的视频馈送接口和/或用于生成视频数据的计算机图形系统,或这些源的组合。

[0123] 视频编码器1220可以对捕获的、预先捕获的或计算机生成的视频数据编码。输入图像可以由视频编码器1220接收并存储在输入帧存储器1221中。通用处理器1223可以从这里加载信息并执行编码。可以从存储设备,例如图12中描绘的示例性存储器模块,加载用于驱动通用处理器的程序。通用处理器可以使用处理存储器1222进行编码,并且通用处理器对编码信息的输出可以被存储在缓冲器,例如输出缓冲器1226中。

[0124] 视频编码器1220可以包括重采样模块1225,其可以被配置成在可缩放视频编码方案中对视频数据编码(code)(例如,编码(encode)),该可缩放视频编码方案定义至少一个基础层和至少一个增强层。作为编码过程的一部分,重采样模块1225可以对至少一些视频数据进行重采样,其中,可以使用重采样滤波器以自适应方式执行重采样。

[0125] 可以经由源设备1212的输出接口1228直接向目的地设备1214传输编码视频数据,例如,编码比特流。在图12的示例中,目的地设备1214包括输入接口1238、视频解码器1230和显示设备1232。在一些情况下,输入接口1238可以包括接收器和/或调制解调器。目的地设备1214的输入接口1238通过信道1216接收编码视频数据。编码视频数据可以包括由视频编码器1220生成的代表视频数据的各种语法元素。这样的语法元素可以与通信介质上传输的编码视频数据一起被包括,存储在存储介质上或存储在文件服务器上。

[0126] 编码视频数据也可以存储到存储介质或文件服务器上,以供目的地设备1214稍晚访问以进行解码和/或回放。例如,编码比特流可以临时存储在输入缓冲器1231中,然后加载到通用处理器1233中。可以从存储设备或存储器加载用于驱动通用处理器的程序。通用处理器可以使用处理存储器1222来执行解码。视频编码器1230还可以包括类似于视频编码器1220中采用的重采样模块1225的重采样模块1235。

[0127] 图12描绘了与通用处理器1233分开的重采样模块1235,但本领域的技术人员将理解,重采样功能可以由通用处理器执行的程序执行,并且视频编码器中的处理可以使用一个或多个处理器来完成。解码的图像可以存储在输出帧缓冲器1236中,并且然后被发送到输入接口1238。

[0128] 显示设备1232可以与目的地设备1214集成或者可以在其外部。在一些示例中,目的地设备1214可以包括集成显示设备,并且还可以被配置为与外部显示设备接合。在其他示例中,目的地设备1214可以是显示设备。通常,显示设备1232向用户显示解码视频数据。

[0129] 视频编码器1220和视频解码器1230可以根据视频压缩标准操作。ITU-T VCEG (Q6/16) 和ISO/IEC MPEG (JTC 1/SC 29/WG 11) 正在研究对未来视频编码技术的标准化的潜在需求,该技术的压缩能力显著地超过当前的高效视频编码HEVC标准(包括其对屏幕内容编码和高动态范围编码的当前扩展和近期扩展)。各工作组正在共同努力一起开展这项探索



活动(被称为联合视频探索团队(JVET)),以评估这个领域中其专家提出的压缩技术设计。在作者为J.Chen,E.Alshina,G.Sullivan,J.Ohm,J.Boyce的“Algorithm Description of Joint Exploration Test Model 5(JEM 5)”,JVET-E1001-V2中描述了JVET开发的最近情况。

[0130] 此外或替代地,视频编码器1220和视频解码器1230可以根据与所公开的JVET特征一起工作的其他专利或行业标准而操作。因而,其他标准是例如ITU-T H.264标准,或称为MPEG-4,部分10,高级视频编码(AVC)或此类标准的扩展。因此,尽管是为JVET新开发的,但本公开的技术不限于任何特定的编码标准或技术。视频压缩标准和技术和其他示例包括MPEG-2、ITU-T H.263和专有或开源压缩格式和相关格式。

[0131] 视频编码器1220和视频解码器1230可以在硬件、软件、固件或它们的任意组合中实现。例如,视频编码器1220和解码器1230可以采用一个或多个处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、离散逻辑或其任意组合。在视频编码器1220和解码器1230部分地中软件中实现时,设备可以将用于软件的指令存储在合适的非暂态计算机可读存储介质中,并可以在硬件中使用一个或多个处理器执行该指令以执行本公开的技术。视频编码器1220和视频解码器1230中的每一个都可以包括在一个或多个编码器或解码器中,编码器或解码器的任一者都可以被集成为相应设备中的组合编码器/解码器(CODEC)的部分。

[0132] 可以在由计算机执行的计算机可执行指令(例如,程序模块)的一般上下文中描述本文所述主题的各方面,计算机是例如上文所述的通用处理器1223和1233。通常,程序模块包括执行特定任务或实施特定抽象数据类型的例程、程序、对象、部件、数据结构等。也可以在分布式计算环境中实践本文所述的主题的各方面,其中任务是由通过通信网络链接的远程处理设备执行的。在分布式计算环境中,程序模块可位于包括存储设备的本地和远程计算机存储介质。

[0133] 存储器的示例包括随机存取存储器(RAM)、只读存储器(ROM)或两者。存储器可以存储指令,例如,源代码或二进制代码,以用于执行上文所述的技术。在执行要由诸如处理器1223和1233的处理器执行的指令期间,存储器还可用于存储变量或其他中间信息。

[0134] 存储设备还可以存储指令,例如,源代码或二进制代码,以用于执行上文所述的技术。存储设备可以额外地存储由计算机处理器使用 and 操控的数据。例如,视频编码器1220或视频解码器1230中的存储设备可以是由计算机系统1223或1233访问的数据库。存储设备的其他示例包括随机存取存储器(RAM)、只读存储器(ROM)、硬盘驱动器、磁盘、光盘、CD-ROM、DVD、闪存存储器、USB存储卡或任何其他计算机可以读取的介质。

[0135] 存储器或存储设备可以是由或结合视频编码器和/或解码器使用的非暂态计算机可读存储介质的示例。非暂态计算机可读存储介质包含用于控制计算机系统的指令,所述计算机系统要被配置为执行特定实施方案描述的功能。所述指令在由一个或多个计算机处理器执行时,可以被配置成执行在特定实施方案中描述的功能。

[0136] 而且,要指出的是,已经将一些实施方案描述为可以被描绘为流程图或框图的过程。虽然每者可将操作描述为顺序的过程,但是这些操作中的多个操作可并行执行或同时执行。此外,操作的顺序可被重新布置。过程可以具有附图中未包括的额外步骤。

[0137] 特定实施方案可以在非暂态计算机可读存储介质中实现,以由或结合指令执行系

统、装置、系统或机器使用。计算机可读存储介质包含用于控制计算机系统以执行特定实施方案描述的方法的指令。计算机系统包括一个或多个计算设备。指令在由一个或多个计算机处理器执行时,可以被配置成执行在特定实施方案中描述的功能。

[0138] 如本文中的说明书和随后的整个权利要求书中所使用的,除非上下文明确指出其他表述,否则“一(a,an)”和“该”包括复数引用。而且,如本文中的说明书和随后的整个权利要求书中所使用的,“在……中”的含义包括“在……中”和“在……上”,除非上下文明确指出其他表述。

[0139] 尽管已经用以上结构特征和/或方法动作特有的语言详细描述了本发明的示范性实施方案,但要理解的是,本领域的技术人员将容易认识到,在实质上不背离本发明的新颖教导和优点的情况下,在示范性实施方案中很多额外修改是可能的。此外,应当理解,所附权利要求中定义的主题未必限于上述具体特征或动作。因此,这些和所有这样的修改都意在包括在根据所附权利要求的宽度和范围解释的本发明的范围内。

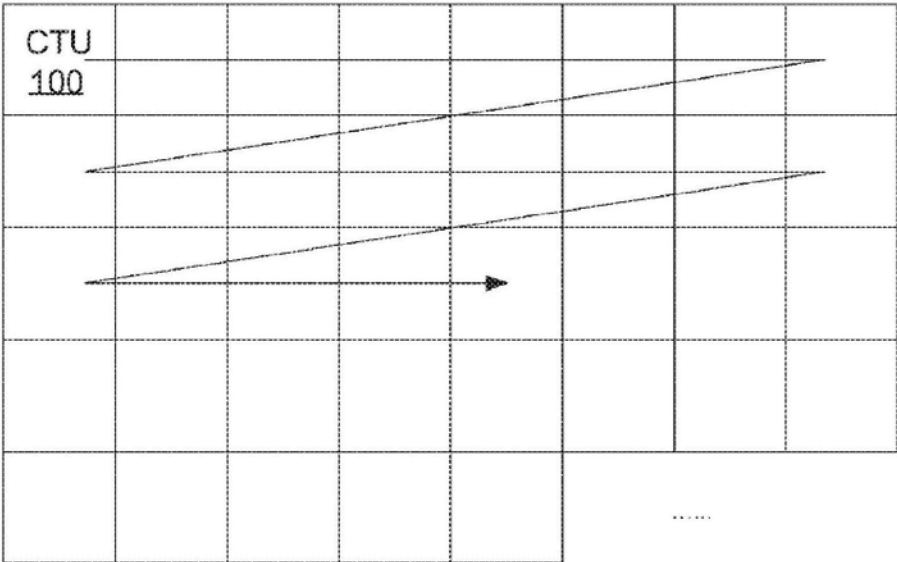


图1

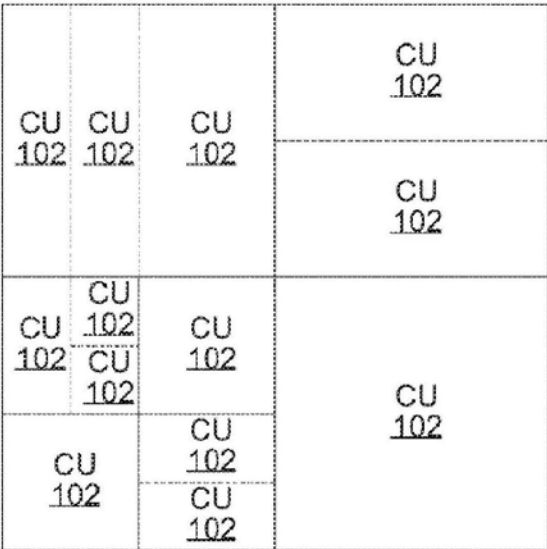


图2a

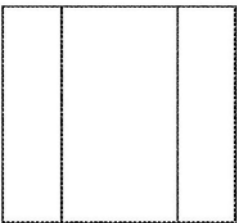


图2b



图2c

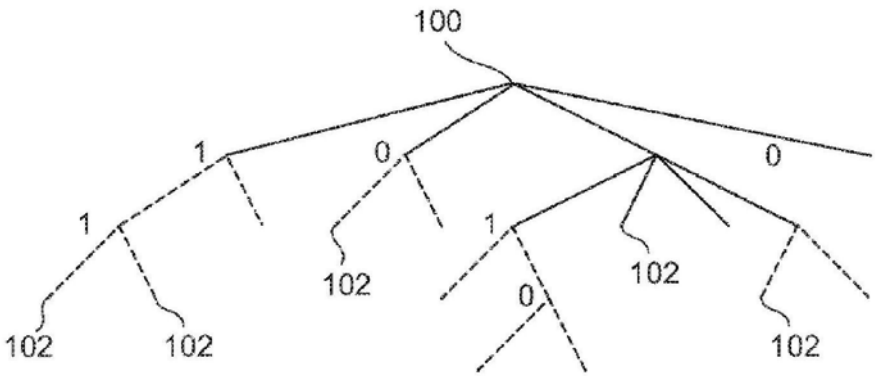


图3

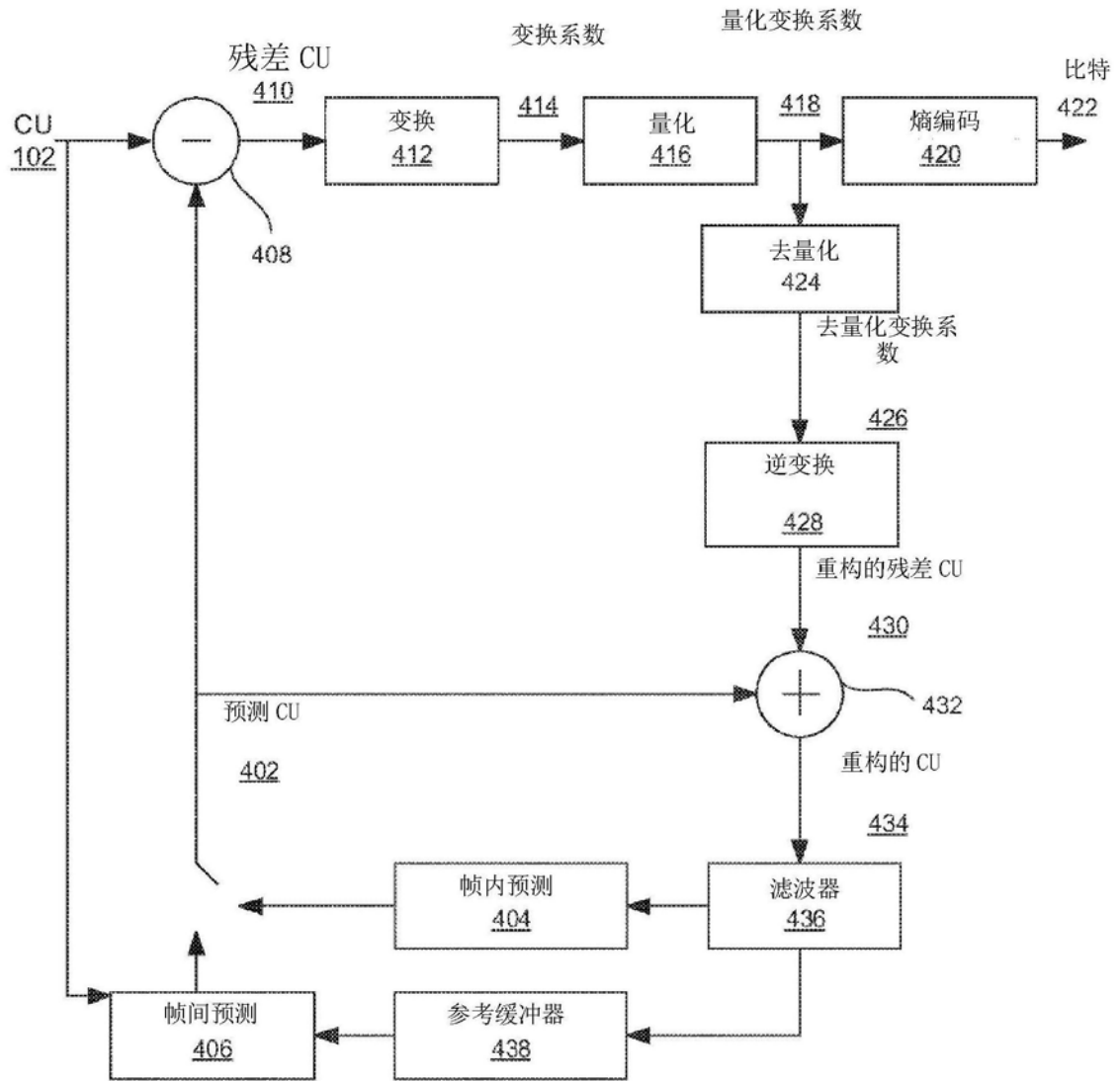


图4

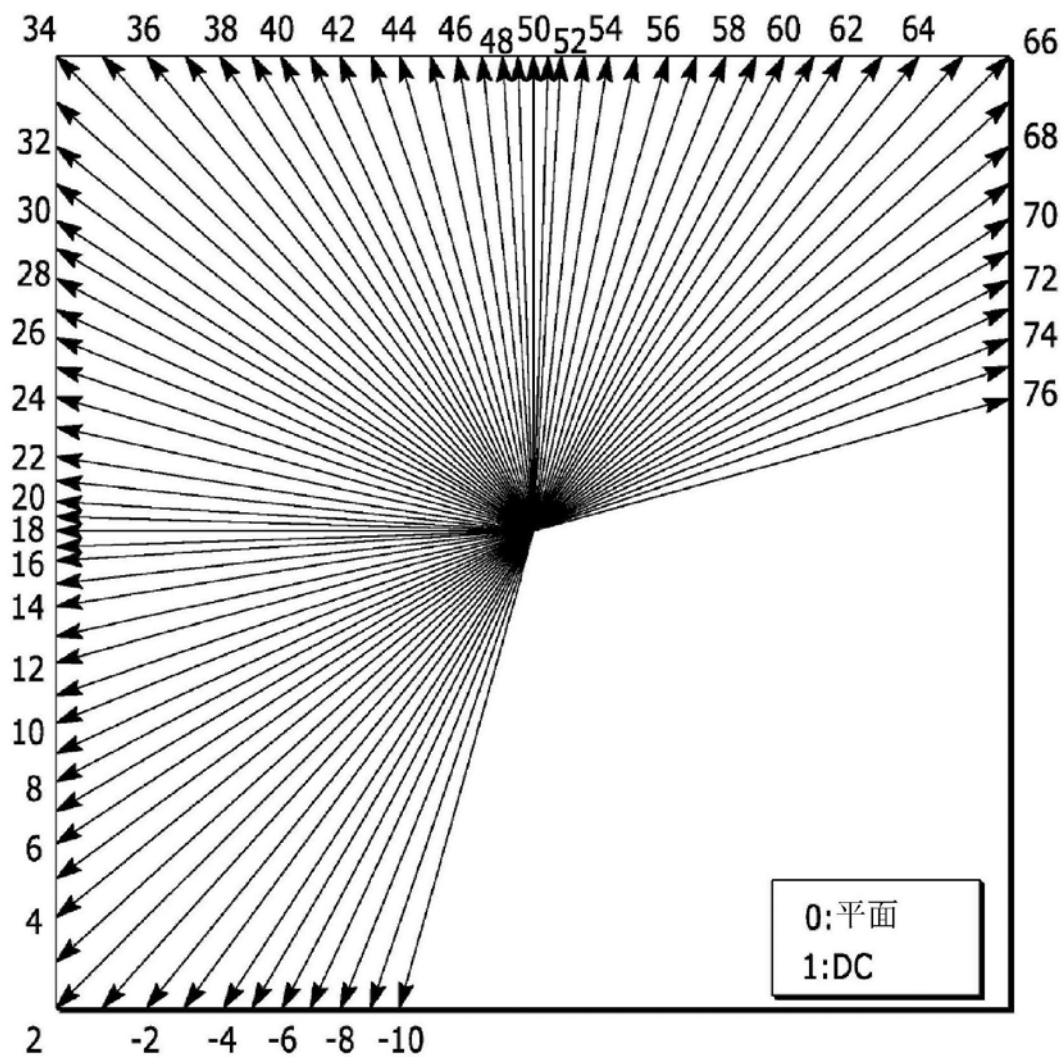


图5

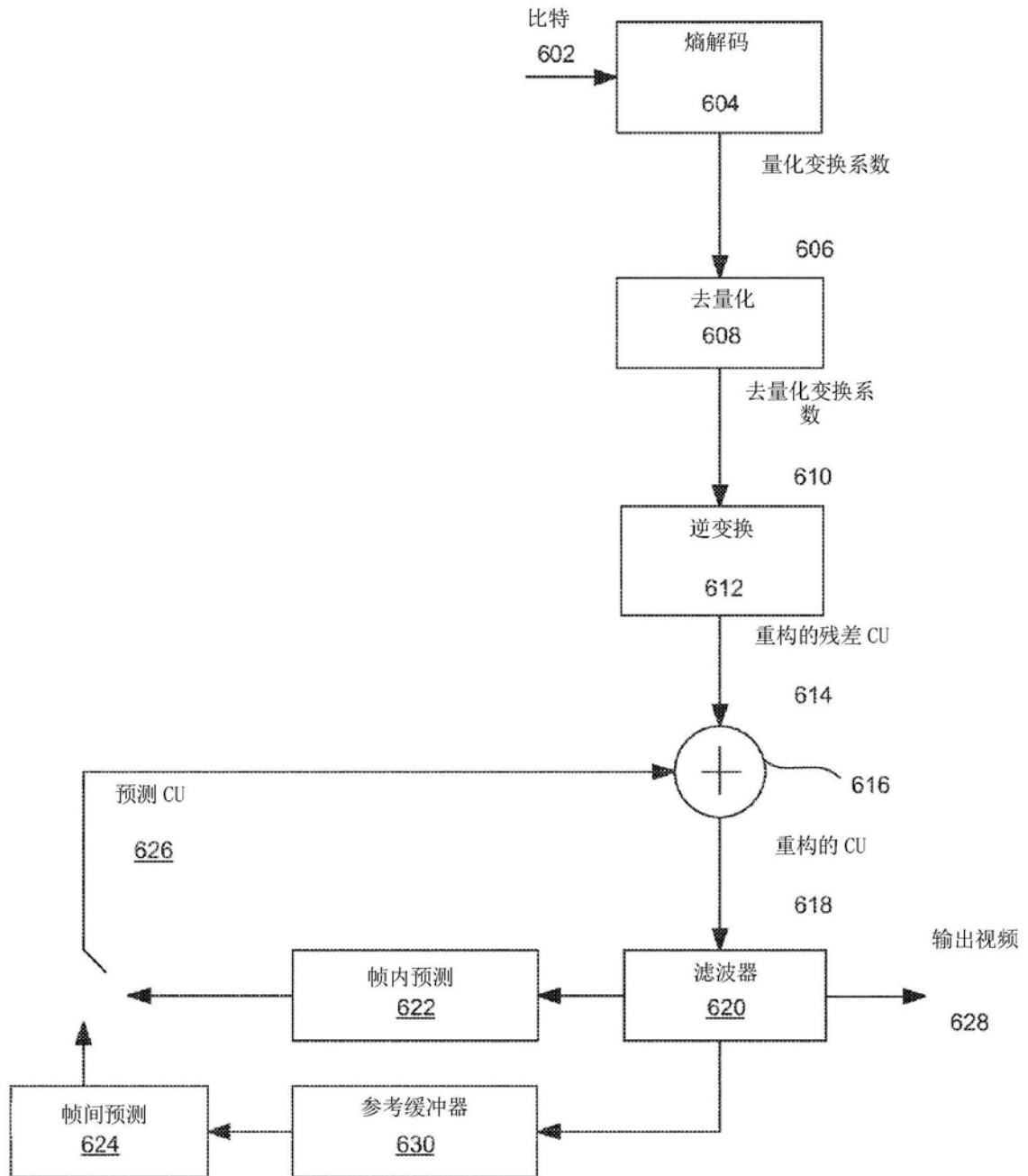


图6

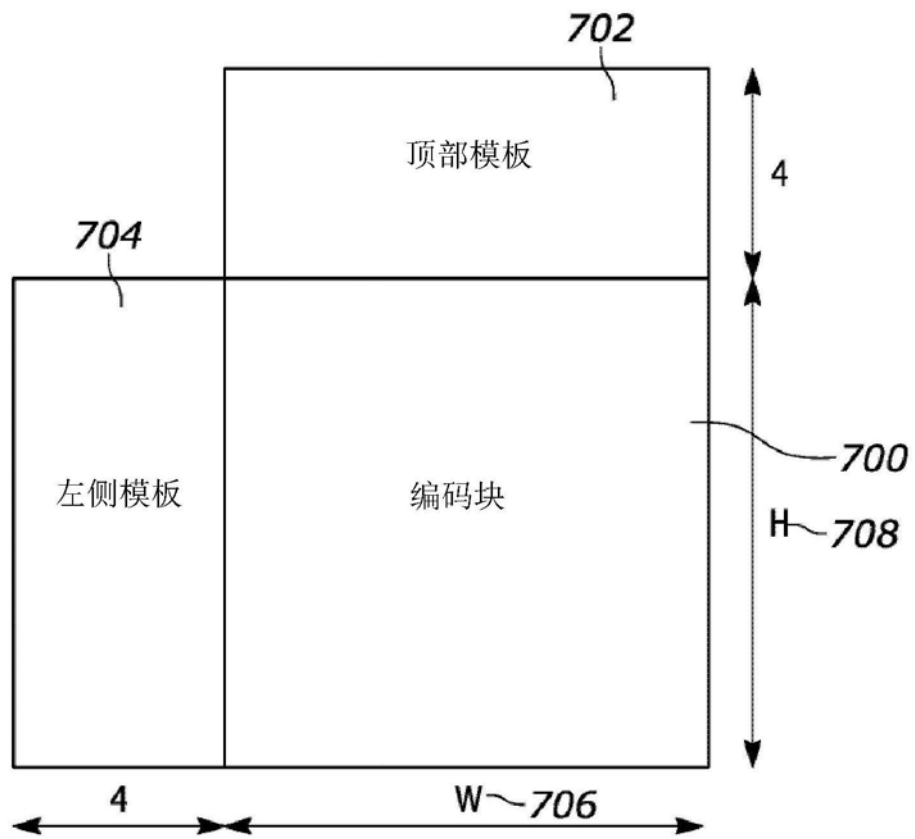


图7



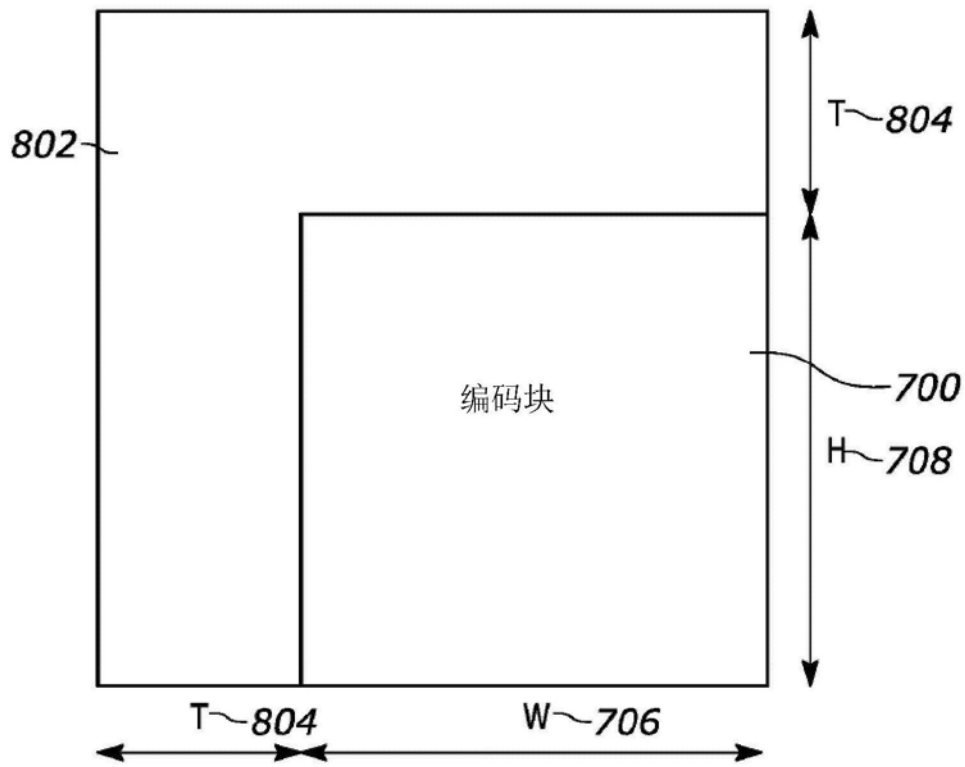


图8

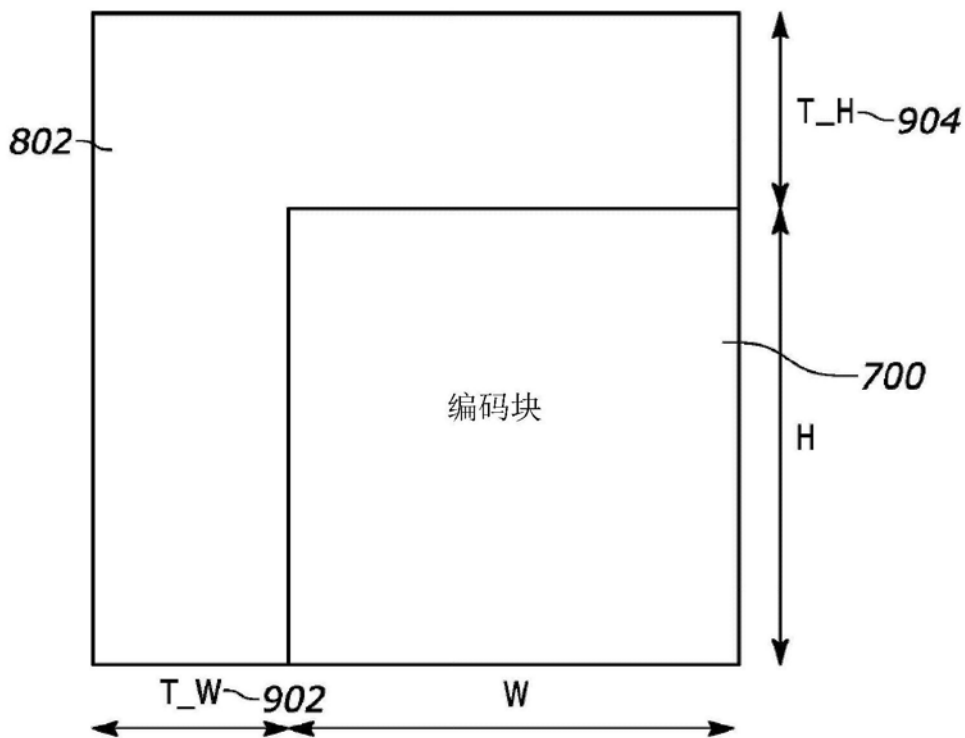


图9

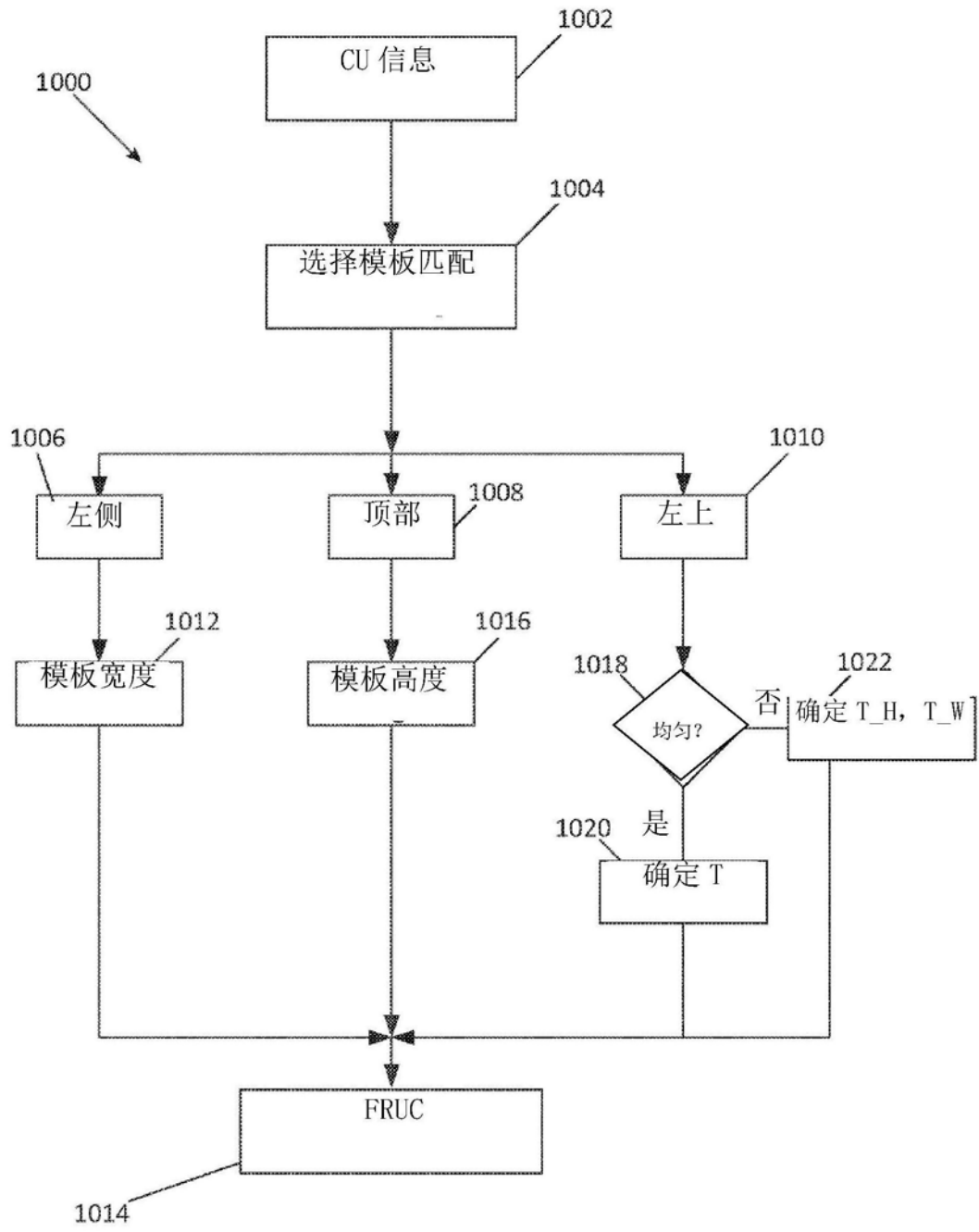


图10

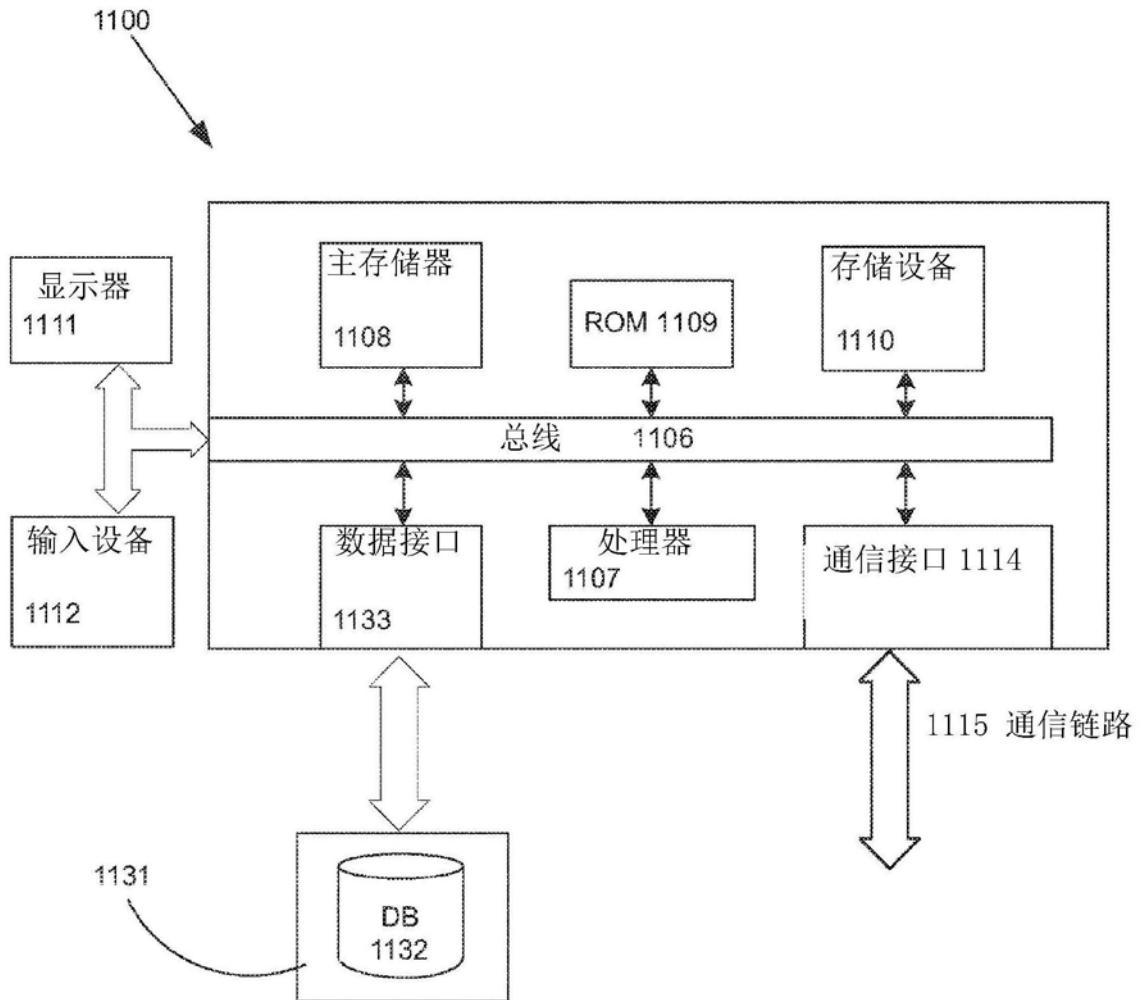


图11

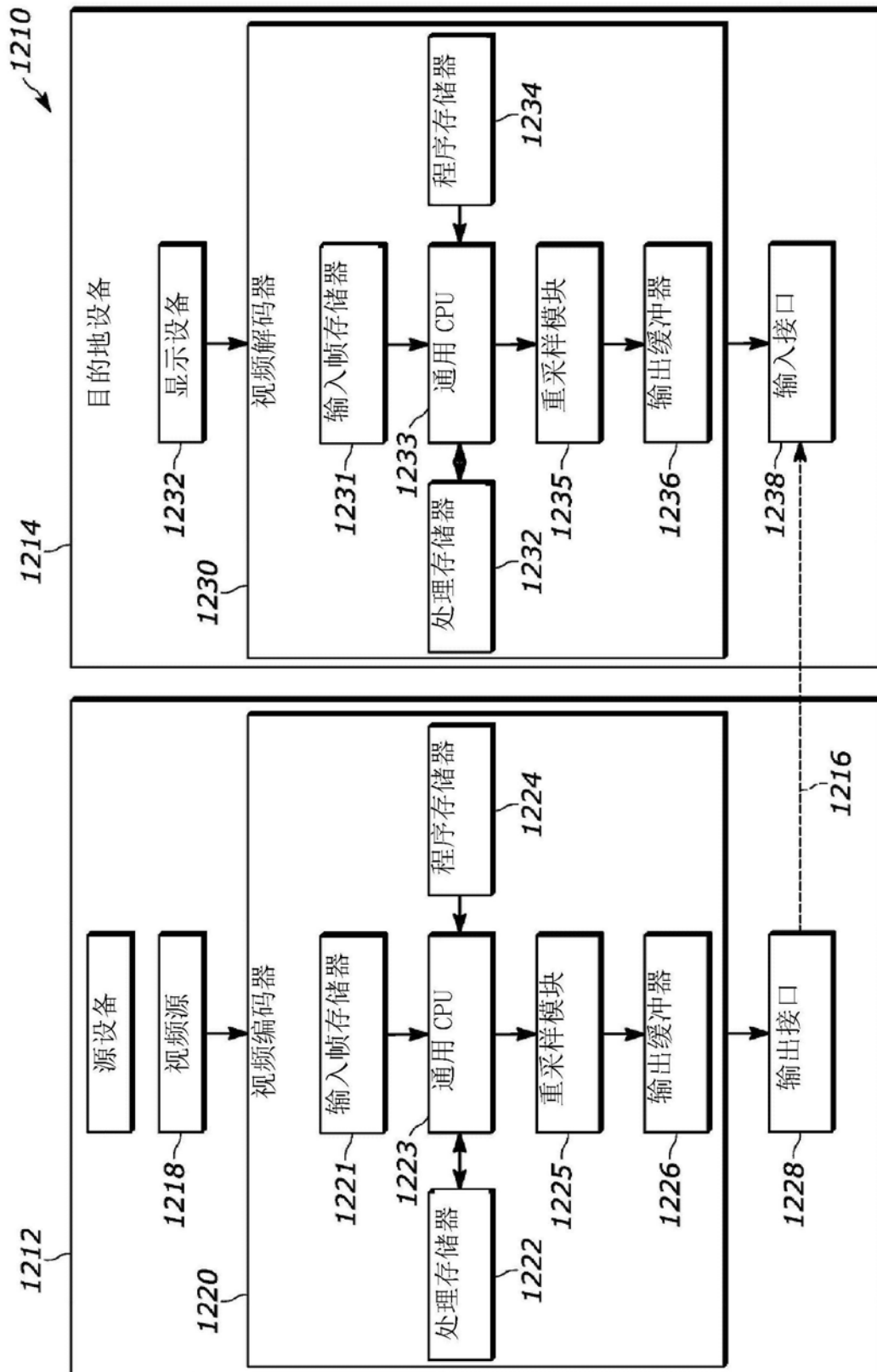


图12