



(12) 发明专利申请

(10) 申请公布号 CN 120302070 A

(43) 申请公布日 2025. 07. 11

(21) 申请号 202510641812.2

H04N 19/119 (2014.01)

(22) 申请日 2020.08.04

H04N 19/61 (2014.01)

(30) 优先权数据

H04N 19/46 (2014.01)

2019232801 2019.09.17 AU

H04N 19/159 (2014.01)

G06T 9/40 (2006.01)

(62) 分案原申请数据

H04N 19/18 (2014.01)

202080062643.2 2020.08.04

H04N 19/122 (2014.01)

(71) 申请人 佳能株式会社

H04N 19/169 (2014.01)

地址 日本

H04N 19/70 (2014.01)

H04N 19/186 (2014.01)

(72) 发明人 克里斯托弗·詹姆斯·罗斯沃恩

H04N 19/132 (2014.01)

伊夫特哈尔·艾哈迈德

H04N 19/176 (2014.01)

(74) 专利代理机构 北京魏启学律师事务所

11398

专利代理师 陈涛

(51) Int. Cl.

H04N 19/96 (2014.01)

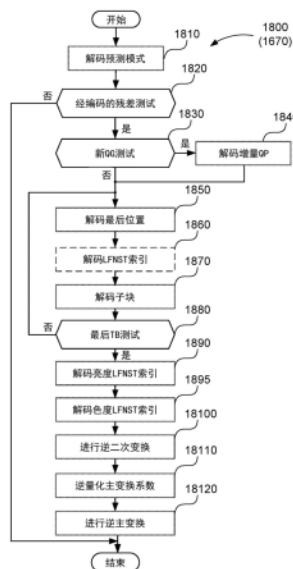
权利要求书3页 说明书39页 附图22页

(54) 发明名称

从位流中解码编码单元的方法和设备、将编码单元编码在位流中的方法和设备、存储介质和计算机程序产品

(57) 摘要

提供了一种从位流中解码编码单元的方法和设备、将编码单元编码在位流中的方法和设备、存储介质和计算机程序产品。具体地,一种从视频位流中解码来自图像帧的编码树单元的编码树的编码单元的系统和方法,编码单元具有主颜色通道和至少一个二次颜色通道。该方法包括:根据编码树单元的经解码拆分标志来确定包括主颜色通道和至少一个二次颜色通道的编码单元;对第一索引进行解码以选择用于主颜色通道的核,并且对第二索引进行解码以选择用于至少一个二次颜色通道的核;根据第一索引选择第一核,并且根据第二索引选择第二核;以及通过将第一核应用于主颜色通道的残差系数并且将第二核应用于至少一个二次颜色通道的残差系数来对编码单元进行解码。



1. 一种从位流中解码图像的编码树单元中的编码单元的方法,所述编码单元具有亮度通道和色度通道,所述方法包括:

根据所述编码树单元的一个或多个拆分标志来确定具有所述亮度通道和所述色度通道的所述编码单元,其中所述编码单元能够是从所述编码树单元中的一个或多个拆分所获得的多个编码单元其中之一,所述一个或多个拆分能够包括水平二元拆分;

从所述位流中解码用于选择用于所述亮度通道的不可分离变换核的第一索引;

根据所述第一索引来选择所述不可分离变换核;

从所述位流中解码所述编码单元中的所述亮度通道的亮度变换块的系数和所述编码单元中的所述色度通道的色度变换块的系数;

通过应用所选择的不可分离变换核来对所述亮度变换块的系数进行不可分离变换以导出所述亮度变换块的不可分离变换后的系数;以及

通过对所述亮度变换块的不可分离变换后的系数以及对所述色度变换块的系数进行可分离变换来对所述编码单元进行解码,

其中,在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树相同的情况下,(a)能够对用于选择用于所述亮度通道的不可分离变换核的所述第一索引进行解码,(b)不对用于选择用于所述色度通道的不可分离变换核的第二索引进行解码,(c)能够仅对所述编码单元中的亮度变换块的系数进行所述不可分离变换,并且(d)不对所述编码单元中的色度变换块的系数进行所述不可分离变换,所述色度变换块的宽度和高度都等于或大于4,以及

其中,在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树分离、所述编码树单元中的给定区域被拆分成亮度编码块、并且存在与所述给定区域相对应的色度编码块的情况下,(a)用于选择用于所述亮度通道的不可分离变换核的所述第一索引能够针对所述亮度编码块中的各个亮度编码块单独存在,并且(b)用于选择用于所述色度通道的不可分离变换核的所述第二索引能够针对与所述给定区域相对应的色度编码块存在。

2. 一种将图像的编码树单元中的编码单元编码在位流中的方法,所述编码单元具有亮度通道和色度通道,所述方法包括:

确定具有所述亮度通道和所述色度通道的编码单元,其中所述编码单元能够是从所述编码树单元中的一个或多个拆分所获得的多个编码单元其中之一,所述一个或多个拆分能够包括水平二元拆分;

(a)对所述编码单元中的所述亮度通道的亮度变换块的系数进行可分离变换以导出所述亮度变换块的可分离变换后的系数,并且(b)对所述编码单元中的所述色度通道的色度变换块的系数进行可分离变换以导出所述色度变换块的可分离变换后的系数;

选择用于所述亮度通道的不可分离变换核;

通过应用所选择的不可分离变换核来对所述亮度变换块的可分离变换后的系数进行不可分离变换;以及

将用于选择用于所述亮度通道的不可分离变换核的第一索引编码在所述位流中,

其中,在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树相同的情况下,(a)能够对用于选择用于所述亮度通道的不可分离变换核的所述第一

索引进行编码, (b) 不对用于选择用于所述色度通道的不可分离变换核的第二索引进行编码, (c) 能够仅对所述编码单元中的亮度变换块的可分离变换后的系数进行所述不可分离变换, 并且 (d) 不对所述编码单元中的色度变换块的可分离变换后的系数进行所述不可分离变换, 所述色度变换块的宽度和高度都等于或大于4, 以及

其中, 在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树分离、所述编码树单元中的给定区域被拆分成亮度编码块、并且存在与所述给定区域相对应的色度编码块的情况下, (a) 用于选择用于所述亮度通道的不可分离变换核的所述第一索引能够针对所述亮度编码块中的各个亮度编码块单独存在, 并且 (b) 用于选择用于所述色度通道的不可分离变换核的所述第二索引能够针对与所述给定区域相对应的色度编码块存在。

3. 一种从位流中解码图像的编码树单元中的编码单元的设备, 所述编码单元具有亮度通道和色度通道, 所述设备包括:

确定单元, 其被配置为根据所述编码树单元的一个或多个拆分标志来确定具有所述亮度通道和所述色度通道的编码单元, 其中所述编码单元能够是从所述编码树单元中的一个或多个拆分所获得的多个编码单元其中之一, 所述一个或多个拆分能够包括水平二元拆分;

第一解码单元, 其被配置为从所述位流中解码用于选择用于所述亮度通道的不可分离变换核的第一索引;

选择单元, 其被配置为根据所述第一索引来选择所述不可分离变换核;

第二解码单元, 其被配置为从所述位流中解码所述编码单元中的所述亮度通道的亮度变换块的系数和所述编码单元中的所述色度通道的色度变换块的系数;

执行单元, 其被配置为通过应用所选择的不可分离变换核来对所述亮度变换块的系数进行不可分离变换以导出所述亮度变换块的不可分离变换后的系数; 以及

第三解码单元, 其被配置为通过对所述亮度变换块的不可分离变换后的系数以及对所述色度变换块的系数进行可分离变换来对所述编码单元进行解码,

其中, 在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树相同的情况下, (a) 能够对用于选择用于所述亮度通道的不可分离变换核的所述第一索引进行解码, (b) 不对用于选择用于所述色度通道的不可分离变换核的第二索引进行解码, (c) 能够仅对所述编码单元中的亮度变换块的系数进行所述不可分离变换, 并且 (d) 不对所述编码单元中的色度变换块的系数进行所述不可分离变换, 所述色度变换块的宽度和高度都等于或大于4, 以及

其中, 在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树分离、所述编码树单元中的给定区域被拆分成亮度编码块、并且存在与所述给定区域相对应的色度编码块的情况下, (a) 用于选择用于所述亮度通道的不可分离变换核的所述第一索引能够针对所述亮度编码块中的各个亮度编码块单独存在, 并且 (b) 用于选择用于所述色度通道的不可分离变换核的所述第二索引能够针对与所述给定区域相对应的色度编码块存在。

4. 一种将图像的编码树单元中的编码单元编码在位流中的设备, 所述编码单元具有亮度通道和色度通道, 所述设备包括:

确定单元,其被配置为确定具有所述亮度通道和所述色度通道的编码单元,其中所述编码单元能够是从所述编码树单元中的一个或多个拆分所获得的多个编码单元其中之一,所述一个或多个拆分能够包括水平二元拆分;

第一执行单元,其被配置为(a)对所述编码单元中的所述亮度通道的亮度变换块的系数进行可分离变换以导出所述亮度变换块的可分离变换后的系数,并且(b)对所述编码单元中的所述色度通道的色度变换块的系数进行可分离变换以导出所述色度变换块的可分离变换后的系数;

选择单元,其被配置为选择用于所述亮度通道的不可分离变换核;

第二执行单元,其被配置为通过应用所选择的不可分离变换核对所述亮度变换块的可分离变换后的系数进行不可分离变换;以及

编码单元,其被配置为将用于选择用于所述亮度通道的不可分离变换核的第一索引编码在所述位流中,

其中,在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树相同的情况下,(a)能够对用于选择用于所述亮度通道的不可分离变换核的所述第一索引进行编码,(b)不对用于选择用于所述色度通道的不可分离变换核的第二索引进行编码,(c)能够仅对所述编码单元中的亮度变换块的可分离变换后的系数进行所述不可分离变换,并且(d)不对所述编码单元中的色度变换块的可分离变换后的系数进行所述不可分离变换,所述色度变换块的宽度和高度都等于或大于4,以及

其中,在所述编码树单元中的亮度通道的编码树与所述编码树单元中的色度通道的编码树分离、所述编码树单元中的给定区域被拆分成亮度编码块、并且存在与所述给定区域相对应的色度编码块的情况下,(a)用于选择用于所述亮度通道的不可分离变换核的所述第一索引能够针对所述亮度编码块中的各个亮度编码块单独存在,并且(b)用于选择用于所述色度通道的不可分离变换核的所述第二索引能够针对与所述给定区域相对应的色度编码块存在。

5.一种非暂时性计算机可读存储介质,其包含计算机可执行指令,所述计算机可执行指令使计算机进行根据权利要求1所述的方法。

6.一种非暂时性计算机可读存储介质,其包含计算机可执行指令,所述计算机可执行指令使计算机进行根据权利要求2所述的方法。

7.一种计算机程序产品,其包括程序,所述程序在被计算机执行时使所述计算机进行根据权利要求1所述的方法。

8.一种计算机程序产品,其包括程序,所述程序在被计算机执行时使所述计算机进行根据权利要求2所述的方法。

从位流中解码编码单元的方法和设备、将编码单元编码在位流中的方法和设备、存储介质和计算机程序产品

[0001] (本申请是申请日为2020年8月4日、申请号为2020800626432、发明名称为“用于编码和解码图像的编码树单元中的编码单元的方法和设备、非暂时性计算机可读存储介质和计算机程序产品”的申请的分案申请。)

技术领域

[0002] 本发明通常涉及数字视频信号处理,尤其涉及用于对视频样本的块进行编码和解码的方法、设备和系统。本发明还涉及包括记录有用于对视频样本的块进行编码和解码的计算机程序的计算机可读介质的计算机程序产品。

背景技术

[0003] 当前存在包括用于传输和存储视频数据的应用的许多视频编码用的应用。还开发了许多视频编码标准并且其它视频编码标准当前正在开发中。视频编码标准化的最新进展已导致形成被称为“联合视频专家组”(JVET)的组。该联合视频专家组(JVET)包括:还已知为“视频编码专家组”(VCEG)的国际电信联盟(ITU)的电信标准化部门(ITU-T)的研究组16、问题6(SG16/Q6)的成员;以及还已知为“运动图片专家组”(MPEG)的国际标准化组织/国际电工协会联合技术协会1/小组协会29/工作组11(ISO/IEC JTC1/SC29/WG11)的成员。

[0004] 联合视频专家组(JVET)发布了提案征集(CfP),并在美国圣地亚哥市举行的第10次会议上对答复进行了分析。所提交的答复表明,视频压缩能力明显优于当前最先进的视频压缩标准(即,“高效率视频编码”(HEVC))的视频压缩能力。基于该优异表现,决定开始用以开发命名为“通用视频编码”(VVC)的新视频压缩标准的项目。预计VVC将特别是随着视频格式的能力的增加(例如,具有更高的分辨率和更高的帧频)解决针对甚至更高的压缩性能的持续需求、以及解决针对通过WAN的服务提供(其中,带宽成本相对较高)的日益增长的市场需求。诸如沉浸式视频等的用例需要实时编码和解码这种较高的格式,例如,立方体地图投影(CMP)可以使用8K格式,即使最终渲染的“视口”利用较低的分辨率。VVC必须可在当代硅工艺中实现,并且在所实现的性能与实现成本之间提供可接受的折衷。例如,在硅面积、CPU处理器负荷、存储器利用率和带宽的一个或多于一个方面可以考虑实现成本。可以通过将帧区域分割成部分并且并行处理各个部分来处理较高视频格式。由压缩帧的多个部分构建的位流仍适合于由“单核”解码器解码,即根据应用需要将帧级约束(包括位率)分配给各个部分。

[0005] 视频数据包括图像数据的帧序列,各个帧包括一个或多于一个颜色通道。通常,需要一个主颜色通道和两个二次颜色通道。主颜色通道通常被称为“亮度”通道,并且(一个或多于一个)二次颜色通道通常被称为“色度”通道。尽管视频数据通常在RGB(红-绿-蓝)颜色空间中显示,但该颜色空间在三个相应分量之间具有高度相关性。编码器或解码器所看到的视频数据表示通常使用诸如YCbCr等的颜色空间。YCbCr将发光度(根据变换方程映射到“亮度”)集中在Y(主)通道中,并且将色度集中在Cb和Cr(二次)通道中。由于使用去相关的

YCbCr信号,因此亮度通道的统计与色度通道的统计明显不同。主要差异在于:在量化之后,与相应亮度通道块的系数相比,色度通道针对给定块包含相对较少的有效系数。此外,可以与亮度通道相比更低的速率(例如,在水平方向上为一半且在垂直方向上为一半(被称为“4:2:0色度格式”))对Cb和Cr通道进行空间采样。4:2:0色度格式通常用于“消费者”应用中,诸如互联网视频流式传输、广播电视以及蓝光™盘上的存储。在水平方向上以半速率对Cb和Cr通道进行子采样而不是垂直地进行子采样被称为“4:2:2色度格式”。4:2:2色度格式通常用于专业应用中,包括用于电影制作等的镜头的捕获。4:2:2色度格式的较高采样速率使得所得视频对编辑操作(诸如颜色分级等)更有弹性。在分发到消费者之前,4:2:2色度格式材料经常被转换为4:2:0色度格式,然后被编码以分发到消费者。除了色度格式之外,视频还由分辨率和帧频来表征。示例分辨率是具有 3840×2160 的分辨率的超高清清晰度(UD)或具有 7680×4320 的分辨率的“8K”,并且示例帧频是60Hz或120Hz。亮度样本速率的范围可以从约500兆样本/秒到数千兆样本/秒。对于4:2:0色度格式,各个色度通道的采样率是亮度采样率的四分之一,并且对于4:2:2色度格式,各个色度通道的采样率是亮度采样率的一半。

[0006] VVC标准是一种“基于块”的编解码器,其中,帧首先被分割成称为“编码树单元”(CTU)的正方形区域阵列。CTU通常占据相对大的区域,诸如 128×128 个亮度样本等。然而,各帧的右和底边缘的CTU的区域可能较小。与各CTU相关联的是针对亮度通道和色度通道这两者的“编码树”(“共用树”)或者各自针对亮度通道和色度通道的单独树。编码树定义了将CTU的区域分解为一组区域,也称为“编码块”(CB)。当使用共用树时,单个编码树指定针对亮度通道和色度通道这两者的块,在这种情况下,并置的编码块的集合被称为“编码单元”(CU),即,各个CU具有用于各个颜色通道的编码块。按特定顺序处理CB以进行编码或解码。作为使用4:2:0色度格式的结果,包括 128×128 亮度样本区域的亮度编码树的CTU具有与 128×128 亮度样本区域并置的 64×64 色度样本区域的相应色度编码树。当单个编码树用于亮度通道和色度通道时,给定区域的并置块的集合通常称为“单元”,例如上述CU以及“预测单元”(PU)和“变换单元”(TU)。具有跨越4:2:0色度格式视频数据的颜色通道的CU的单个树产生相应亮度块的一半宽度和高度的色度块。当针对给定区域使用单独的编码树时,将使用上述CB以及“预测块”(PB)和“变换块”(TB)。

[0007] 尽管在“单元”和“块”之间有上述区别,但是术语“块”可以用作针对将操作应用于所有颜色通道的帧的区域(area)或区(region)的通用术语。

[0008] 对于各CU,生成帧数据的相应区域的内容(样本值)的预测单元(PU) (“预测单位”)。此外,形成了在编码器的输入处看到的预测与区域内容之间的差(或空间域的“残差”)的表示。各颜色通道的差可以被变换并编码为残差系数的序列,从而形成给定CU的一个或多个TU。所应用的变换可以是应用于残差值的各个块的离散余弦变换(DCT)或其它变换。该变换是分开应用的,即分两遍进行二维变换。首先通过对块中的各行样本应用一维变换来对块进行变换。然后,通过对部分结果的各列应用一维变换来对部分结果进行变换,以产生基本上对残差样本进行去相关的变换系数的最终块。VVC标准支持各种大小的变换,包括矩形块(各边尺寸为2的幂)的变换。量化变换系数以用于将熵编码在位流中。

[0009] VVC的特征是帧内预测和帧间预测。帧内预测涉及使用正使用的帧中的先前处理的样本来生成对该帧中的当前样本块的预测。帧间预测涉及使用从先前解码的帧获得的样

本块来生成对帧中的当前样本块的预测。从先前解码的帧获得的样本块根据运动矢量从当前块的空间位置偏移,该运动矢量通常已经应用了滤波。帧内预测块可以是 (i) 均匀样本值 (“DC帧内预测”)、(ii) 具有偏移以及水平梯度和垂直梯度的平面 (“平面帧内预测”)、(iii) 具有在特定方向上应用的相邻样本的块的群体 (“角度帧内预测”) 或者 (iv) 使用相邻样本和所选择的矩阵系数的矩阵乘法的结果。通过将 ‘残差’ 编码在位流中,可以在某种程度上校正预测块与相应输入样本之间的进一步差异。通常将残差从空间域变换到频域以形成残差系数(在 “主变换域” 中),可以通过应用 “二次变换” 进一步变换残差系数(以产生 “二次变换域” 中的残差系数)。根据量化参数量化残差系数,导致在解码器处产生的样本的重建的精度损失,而位流内的位率也减小。量化参数可以在帧之间以及在各个帧内变化。对于 “速率控制” 的编码器,帧内量化参数的变化是典型的。不管所接收的输入样本的统计数据(诸如噪声性质、运动程度等)如何,速率控制的编码器都尝试产生具有大体上恒定位率的位流。由于位流通常通过具有有限带宽的网络传送,因此速率控制是用以不管输入到编码器的原始帧的变化如何都确保网络上的可靠性能的普遍技术。在帧以并行区段编码的情况下,速率控制的使用的灵活性是所期望的,因为不同区段在所期望的保真度方面可能具有不同的要求。

发明内容

[0010] 本发明的目的是基本上克服或至少改善现有布置的一个或多于一个缺点。

[0011] 本公开的一个方面提供了一种从视频位流中解码来自图像帧的编码树单元的编码树的编码单元的方法,该编码单元具有主颜色通道和至少一个二次颜色通道,该方法包括:根据编码树单元的经解码拆分标志来确定包括主颜色通道和至少一个二次颜色通道的编码单元;对第一索引进行解码以选择用于所述主颜色通道的核,并且对第二索引进行解码以选择用于所述至少一个二次颜色通道的核;根据所述第一索引选择第一核,并且根据所述第二索引选择第二核;以及通过将第一核应用于主颜色通道的残差系数并且将第二核应用于至少一个二次颜色通道的残差系数来对编码单元进行解码。

[0012] 根据另一方面,在对编码单元的最后有效残差系数的位置进行解码之后紧接着对第一索引或第二索引进行解码。

[0013] 根据另一方面,针对多个二次颜色通道解码单个残差系数。

[0014] 根据另一方面,针对单个二次颜色通道解码单个残差系数。

[0015] 根据另一方面,第一索引和第二索引彼此独立。

[0016] 根据另一方面,第一核和第二核分别取决于用于主颜色通道和至少一个二次颜色通道的帧内预测模式。

[0017] 根据另一方面,第一核和第二核分别与主通道的块大小和至少一个二次颜色通道的块大小相关。

[0018] 根据另一方面,第二核与编码位流的色度子采样率相关。

[0019] 根据另一方面,核中的各个核实现不可分离的二次变换。

[0020] 根据另一方面,编码单元包括两个二次颜色通道,并且针对二次颜色通道中的各个二次颜色通道解码单独的索引。

[0021] 本公开的另一方面提供了一种从视频位流中解码来自图像帧的编码树单元的编

码树的编码单元的方法,所述编码单元具有主颜色通道和至少一个二次颜色通道,所述方法包括:根据所述编码树单元的经解码拆分标志来确定包括所述主颜色通道和所述至少一个二次颜色通道的编码单元;根据所述主颜色通道的经解码索引选择不可分离变换核;将所选择的不可分离变换核应用于所述主颜色通道的解码残差以产生二次变换系数;以及通过将可分离变换核应用于二次变换系数并且将可分离变换核应用于至少一个二次颜色通道的解码残差来对编码单元进行解码。

[0022] 本公开的另一方面提供了一种非暂时性计算机可读介质,其上存储有计算机程序以实现从视频位流中解码来自图像帧的编码树单元的编码树的编码单元的方法,该编码单元具有主颜色通道和至少一个二次颜色通道,该方法包括:根据编码树单元的经解码拆分标志来确定包括主颜色通道和至少一个二次颜色通道的编码单元;对第一索引进行解码以选择用于所述主颜色通道的核,并且对第二索引进行解码以选择用于所述至少一个二次颜色通道的核;根据所述第一索引选择第一核,并且根据所述第二索引选择第二核;以及通过将第一核应用于主颜色通道的残差系数并且将第二核应用于至少一个二次颜色通道的残差系数来对编码单元进行解码。

[0023] 本公开的另一方面提供了一种视频解码器,该视频解码器被配置为实现从视频位流中解码来自图像帧的编码树单元的编码树的编码单元的方法,该编码单元具有主颜色通道和至少一个二次颜色通道,该方法包括:根据编码树单元的经解码拆分标志来确定包括主颜色通道和至少一个二次颜色通道的编码单元;对第一索引进行解码以选择用于所述主颜色通道的核,并且对第二索引进行解码以选择用于所述至少一个二次颜色通道的核;根据所述第一索引选择第一核,并且根据所述第二索引选择第二核;以及通过将第一核应用于主颜色通道的残差系数并且将第二核应用于至少一个二次颜色通道的残差系数来对编码单元进行解码。

[0024] 本公开的另一方面提供了一种系统,其包括:存储器;以及处理器,其中,所述处理器被配置为执行存储在所述存储器上的代码,以实现从视频位流中解码来自图像帧的编码树单元的编码树的编码单元的方法,所述编码单元具有主颜色通道和至少一个二次颜色通道,所述方法包括:根据所述编码树单元的经解码拆分标志来确定包括所述主颜色通道和所述至少一个二次颜色通道的编码单元;对第一索引进行解码以选择用于所述主颜色通道的核,并且对第二索引进行解码以选择用于所述至少一个二次颜色通道的核;根据所述第一索引选择第一核,并且根据所述第二索引选择第二核;以及通过将第一核应用于主颜色通道的残差系数并且将第二核应用于至少一个二次颜色通道的残差系数来对编码单元进行解码。

[0025] 本公开的另一方面提供了一种从位流中解码多个编码单元以产生图像帧的方法,所述编码单元是编码树单元的分解的结果,所述多个编码单元形成位流的一个或多个连续部分,所述方法包括:确定位流的一个或多个连续部分中的各个连续部分的细分级别,各个细分级别适用于位流的相应连续部分的编码单元;对多个区域中的各个区域的量化参数增量进行解码,各个区域基于将编码树单元分解为位流的各个连续部分的编码单元和相应的所确定的细分级别;根据区域的经解码增量量化参数和所述图像帧的较早编码单元的量化参数来确定各个区域的量化参数;使用各个区域的所确定的量化参数对多个编码单元进行解码以产生图像帧。

[0026] 根据另一方面,各个区域基于与编码单元相关联的细分级别和相应连续部分的所确定细分级别的比较。

[0027] 根据另一方面,针对各个区域确定量化参数增量,其中相应的编码树具有小于或等于相应连续部分的所确定的细分级别的细分级别。

[0028] 根据另一方面,针对具有小于或等于相应所确定细分级别的细分级别的编码树单元中的任何节点设置新区域。

[0029] 根据另一方面,针对各个连续部分确定的细分级别包括针对连续部分的亮度编码单元的第一细分级别和针对连续部分的色度编码单元的第二细分级别。

[0030] 根据另一方面,第一细分级别和第二细分级别是不同的。

[0031] 根据另一方面,该方法还包括:解码指示可重写与位流相关联的序列参数集的分区分区约束的标志。

[0032] 根据另一方面,针对一个或多于一个连续部分中的各个连续部分的所确定细分级别包括针对该区域的最大亮度编码单元深度。

[0033] 根据另一方面,针对一个或多于一个连续部分中的各个连续部分的所确定细分级别包括针对相应区域的最大色度编码单元深度。

[0034] 根据另一方面,调整针对连续部分之一的所确定细分级别以维持相对于针对位流的分区分区约束所解码的最深允许细分级别的偏移。

[0035] 本公开的另一方提供了一种非暂时性计算机可读介质,其上存储有计算机程序以实现从位流中解码多个编码单元以产生图像帧的方法,所述编码单元是编码树单元的分解的结果,所述多个编码单元形成所述位流的一个或多于一个连续部分,所述方法包括:确定所述位流的一个或多于一个连续部分中的各个连续部分的细分级别,各个细分级别适用于所述位流的相应连续部分的编码单元;对多个区域中的各个区域的量化参数增量进行解码,各个区域基于将编码树单元分解为位流的各个连续部分的编码单元和相应的所确定的细分级别;根据区域的经解码增量量化参数和所述图像帧的较早编码单元的量化参数来确定各个区域的量化参数;以及使用各个区域的所确定的量化参数对多个编码单元进行解码以产生图像帧。

[0036] 本公开的另一方提供了一种视频解码器,其被配置为实现从位流中解码多个编码单元以产生图像帧的方法,所述编码单元是编码树单元的分解的结果,所述多个编码单元形成位流的一个或多于一个连续部分,所述方法包括:确定位流的一个或多于一个连续部分中的各个连续部分的细分级别,各个细分级别适用于位流的相应连续部分的编码单元;对多个区域中的各个区域的量化参数增量进行解码,各个区域基于将编码树单元分解为位流的各个连续部分的编码单元和相应的所确定的细分级别;根据区域的经解码增量量化参数和图像帧的较早编码单元的量化参数来确定各个区域的量化参数;以及使用各个区域的所确定的量化参数对多个编码单元进行解码以产生图像帧。

[0037] 本公开的另一方提供了一种系统,其包括:存储器;以及处理器,其中所述处理器被配置为执行存储在所述存储器上的代码,以实现从位流中解码多个编码单元以产生图像帧的方法,所述编码单元是编码树单元的分解结果,所述多个编码单元形成所述位流的一个或多于一个连续部分,所述方法包括:确定所述位流的一个或多于一个连续部分中的各个连续部分的细分级别,各个细分级别适用于所述位流的相应连续部分的编码单元;对

多个区域中的各个区域的量化参数增量进行解码,各个区域基于将编码树单元分解为位流的各个连续部分的编码单元和相应的所确定的细分级别;根据区域的经解码增量量化参数和图像帧的较早编码单元的量化参数来确定各个区域的量化参数;以及使用各个区域的所确定的量化参数对多个编码单元进行解码以产生图像帧。

[0038] 还公开了其它方面。

附图说明

[0039] 现在将参考以下附图和附录描述本发明的至少一个实施例,其中:

[0040] 图1是示出视频编码和解码系统的示意性框图;

[0041] 图2A和2B构成可以实践图1的视频编码和解码系统的其中一个或这两者的通用计算机系统的示意框图;

[0042] 图3是示出视频编码器的功能模块的示意框图;

[0043] 图4是示出视频解码器的功能模块的示意框图;

[0044] 图5是示出通用视频编码的树结构中的块向一个或多个于一个块的可用分割的示意框图;

[0045] 图6是用以在通用视频编码的树结构中实现块向一个或多个于一个块的许可分割的数据流的示意图;

[0046] 图7A和7B示出编码树单元 (CTU) 向多个编码单元 (CU) 的示例分割;

[0047] 图8A、8B和8C示出由编码树中的拆分得到的细分级别及其对将编码树单元分割成量化组的影响;

[0048] 图9A和9B示出 4×4 变换块扫描模式以及相关的主变换系数和二次变换系数;

[0049] 图9C和9D示出 8×8 变换块扫描模式以及相关的主变换系数和二次变换系数;

[0050] 图10示出针对各种大小的变换块应用二次变换的区域;

[0051] 图11示出具有多个条带的位流的句法结构,各个条带包括多个编码单元;

[0052] 图12示出具有针对编码树单元的亮度和色度编码块的共用树的位流的句法结构;

[0053] 图13示出将帧编码在包括一个或多个于一个条带作为编码单元序列的位流中的方法;

[0054] 图14示出将条带头部编码在位流中的方法;

[0055] 图15示出将编码单元编码在位流中的方法;

[0056] 图16示出从作为布置成条带的编码单元序列的位流中解码帧的方法;

[0057] 图17示出从位流中解码条带头部的的方法;

[0058] 图18示出从位流中解码编码单元的方法;以及

[0059] 图19A和19B示出用于将二次变换应用或绕过亮度和色度通道的规则。

具体实施方式

[0060] 在任一个或多个于一个附图中参考具有相同附图标记的步骤和/或特征的情况下,除非出现相反意图,否则这些步骤和/或特征为了本说明书的目的而具有相同的(一个或多个于一个)功能或(一个或多个于一个)操作。

[0061] 速率控制的视频编码器需要以适合于块分区约束的粒度来调整量化参数的灵活

性。块分区约束可能在帧的一部分与另一部分之间有所不同,例如,其中多个视频编码器并行操作以压缩各个帧。需要量化参数调整的区域的粒度相应地变化。此外,在生成正被变换的残差的预测信号的范围内应用对所应用的变换选择(包括二次变换的潜在应用)的控制。特别地,对于帧内预测,单独模式可用于亮度块和色度块,因为其可以使用不同的帧内预测模式。

[0062] 视频的一些部分比其它部分对渲染视口的保真度做出更大的贡献,并且可以被分配更大的位率以及在块结构和量化参数的方差方面的更大的灵活性。可以利用更简单的块结构来压缩对所渲染的视口的保真度贡献很小的部分(诸如在所渲染的视图的侧面或后面的那些部分),以减少编码工作量并且在量化参数的控制方面具有较小的灵活性。通常,选择较大的值以更粗略地量化较低位率的变换系数。另外,变换选择的应用在亮度通道与色度通道之间可以是独立的,以通过避免需要联合考虑亮度和色度以进行变换选择来进一步简化编码处理。特别地,在单独考虑亮度和色度的帧内预测模式之后,避免了联合考虑亮度和色度以进行二次变换选择的需要。

[0063] 图1是示出视频编码和解码系统100的功能模块的示意性框图。系统100可以改变在帧的不同部分中调整量化参数的区域,以适应在帧的各个部分中可能有效的不同块分区约束。

[0064] 系统100包括源装置110和目的地装置130。通信通道120用于从源装置110向目的地装置130通信编码视频信息。在一些配置中,源装置110和目的地装置130中的一个或两个分别可以包括移动电话手机或“智能电话”,其中在这种情况下,通信通道120是无线通道。在其它配置中,源装置110和目的地装置130可以包括视频会议设备,其中在这种情况下,通信通道120通常是诸如因特网连接等的有线通道。此外,源装置110和目的地装置130可以包括范围广泛的任意装置,其中这些装置包括支持空中电视广播、有线电视应用、因特网视频应用(包括流传输)、以及在某些计算机可读存储介质(诸如文件服务器中的硬盘驱动器等)上捕获编码视频数据的应用的装置。

[0065] 如图1所示,源装置110包括视频源112、视频编码器114和发送器116。视频源112通常包括所捕获视频帧数据(表示为113)的源,诸如摄像传感器、存储在非暂时性记录介质上的先前捕获到的视频序列、或者来自远程摄像传感器的视频馈送。视频源112也可以是计算机显卡的输出(例如,显示操作系统和在计算装置(例如,平板计算机)上执行的各种应用的视频输出)。可以包括摄像传感器作为视频源112的源装置110的示例包括智能电话、视频摄录机、专业摄像机和网络视频照相机。

[0066] 视频编码器114将来自视频源112的(由箭头113指示的)所捕获帧数据转换(或“编码”)成(由箭头115指示的)位流。位流115由发送器116经由通信通道120作为编码视频数据(或“编码视频信息”)进行发送。位流115也可以存储在诸如“闪存”存储器或硬盘驱动器等非暂时性存储装置122中,直到随后通过通信通道120发送或者作为通过通信通道120的发送的代替为止。例如,经编码视频数据可以在需要时经由广域网(WAN)供应给顾客以用于视频流式传输应用。

[0067] 目的地装置130包括接收器132、视频解码器134和显示装置136。接收器132从通信通道120接收编码视频数据并将所接收到的视频数据作为(由箭头133指示的)位流传递至视频解码器134。然后,视频解码器134将(由箭头135指示的)解码后的帧数据输出至显示装

置136。解码后的帧数据135具有与帧数据113相同的色度格式。显示装置136的示例包括阴极射线管、液晶显示器(诸如在智能电话、平板计算机、计算机监视器、或者单机型电视机中等)。也可以将源装置110和目的地装置130各自的功能体现在单个装置中,该单个装置的示例包括移动电话手机和平板计算机。经解码帧数据可在呈现给用户之前进一步变换。例如,可以使用投影格式从经解码帧数据渲染具有特定纬度和经度的“视口”以表示场景的360°视图。

[0068] 尽管以上说明了示例装置,但源装置110和目的地装置130各自通常经由硬件组件和软件组件的组合可以配置在通用计算机系统内。图2A示出这种计算机系统200,该计算机系统200包括:计算机模块201;诸如键盘202、鼠标指示器装置203、扫描器226、可被配置为视频源112的照相机227、以及麦克风280等的输入装置;以及包括打印机215、可被配置为显示装置136的显示装置214、以及扬声器217的输出装置。计算机模块201可以使用外部调制器-解调器(调制解调器)收发器装置216来经由接线221与通信网络220进行通信。可以表示通信通道120的通信网络220可以是WAN,诸如因特网、蜂窝电信网络或私有WAN等。在接线221是电话线的情况下,调制解调器216可以是传统的“拨号上网”调制解调器。可替代地,在接线221是高容量(例如,线缆或光学的)接线的情况下,调制解调器216可以是宽带调制解调器。还可以使用无线调制解调器来进行向通信网络220的无线连接。收发器装置216可以提供发送器116和接收器132的功能,并且通信通道120可以体现在接线221中。

[0069] 计算机模块201通常包括至少一个处理器单元205和存储器单元206。例如,存储器单元206可以具有半导体随机存取存储器(RAM)和半导体只读存储器(ROM)。计算机模块201还包括多个输入/输出(I/O)接口,其中这多个输入/输出(I/O)接口包括:音频-视频接口207,其连接至视频显示器214、扬声器217和麦克风280;I/O接口213,其连接至键盘202、鼠标203、扫描器226、照相机227以及可选的操纵杆或其它人机接口装置(未示出);以及外部调制解调器216和打印机215所用的接口208。从音频-视频接口207向计算机监视器214的信号通常是计算机显卡的输出。在一些实现中,调制解调器216可以内置于计算机模块201内,例如内置于接口208内。计算机模块201还具有本地网络接口211,其中该本地网络接口211允许计算机系统200经由接线223连接至已知为局域网(LAN)的局域通信网络222。如图2A所示,局域通信网络222还可以经由接线224连接至广域网220,其中该局域通信网络222通常包括所谓的“防火墙”装置或具有相似功能的装置。本地网络接口211可以包括以太网(Ethernet™)电路卡、蓝牙(Bluetooth™)无线配置或IEEE 802.11无线配置;然而,对于接口211,可以实践多种其它类型的接口。本地网络接口211还可以提供发送器116和接收器132的功能,并且通信通道120也可以体现在局域通信网络222中。

[0070] I/O接口208和213可以提供串行连接和并行连接中的任一个或这两者,其中前者通常根据通用串行总线(USB)标准来实现并且具有相应的USB连接器(未示出)。设置有存储装置209,并且存储装置209通常包括硬盘驱动器(HDD)210。还可以使用诸如软盘驱动器和磁带驱动器等的其它存储装置(未示出)。通常设置有光盘驱动器212以用作数据的非易失性源。可以使用例如光盘(例如,CD-ROM、DVD、蓝光盘(Blu ray Disc™))、USB-RAM、便携式外部硬盘驱动器和软盘等的便携式存储器装置作为针对计算机系统200的数据的适当源。通常,HDD 210、光盘驱动器212、网络220和222中的任意还可被配置成作为视频源112进行工作、或者作为为了经由显示器214进行再现所要存储的解码视频数据的目的地而进行工作。

系统100的源装置110和目的地装置130可以体现在计算机系统200中。

[0071] 计算机模块201的组件205-213通常经由互连总线204并且以得到相关领域技术人员已知的计算机系统200的传统操作模式的方式进行通信。例如,处理器205使用接线218连接至系统总线204。同样,存储器206和光盘驱动器212通过接线219连接至系统总线204。可以实践所述配置的计算机的示例包括IBM-PC和兼容机、Sun SPARCstation、Apple MacTM或相似的计算机系统。

[0072] 在适当或期望的情况下,可以使用计算机系统200来实现视频编码器114和视频解码器134以及以下所述的方法。特别地,可以将视频编码器114、视频解码器134和要说明的方法作为在计算机系统200内可执行的一个或多个软件应用程序233来实现。特别地,利用软件233中的在计算机系统200内执行的指令231(参考图2B)来实现视频编码器114、视频解码器134和所述方法的步骤。可以将软件指令231形成各自用于进行一个或多个特定任务的一个或多个代码模块。还可以将软件分割成两个单独部分,其中第一部分和相应的代码模块进行所述方法,并且第二部分和相应的代码模块管理第一部分和用户之间的用户界面。

[0073] 例如,可以将软件存储在包括以下所述的存储装置的计算机可读介质中。将软件从计算机可读介质载入计算机系统200,然后由计算机系统200来执行。具有这样的软件的计算机可读介质或者该计算机可读介质上所记录的计算机程序是计算机程序产品。在计算机系统200中使用该计算机程序产品优选地实现了用于实施视频编码器114、视频解码器134和所述方法的有利设备。

[0074] 通常将软件233存储在HDD 210或存储器206中。将该软件从计算机可读介质载入计算机系统200,并且由计算机系统200来执行。因而,例如,可以将软件233存储在光盘驱动器212所读取的光学可读盘存储介质(例如,CD-ROM) 225上。

[0075] 在一些实例中,将应用程序233以编码在一个或多个CD-ROM 225上并且经由相应的驱动器212进行读取的方式供给至用户,或者可替代地,可以由用户从网络220或222读取应用程序233。更进一步地,还可以将软件从其它计算机可读介质载入计算机系统200。计算机可读存储介质是指将所记录的指令和/或数据提供至计算机系统200以供执行和/或处理的任何非暂时性有形存储介质。这种存储介质的示例包括软盘、磁带、CD-ROM、DVD、蓝光盘(Blu-ray DiscTM)、硬盘驱动器、ROM或集成电路、USB存储器、磁光盘、或者诸如PCMCIA卡等的计算机可读卡等,而与这些装置在计算机模块201的内部还是外部无关。还可以参与将软件、应用程序、指令和/或视频数据或编码视频数据提供至计算机模块401的暂时性或非有形计算机可读传输介质的示例包括:无线电或红外线传输通道及向着其它计算机或联网装置的网络接线、以及包括电子邮件发送和网站上所记录的信息等的因特网或内联网。

[0076] 可以执行上述的应用程序233的第二部分和相应的代码模块来实现要绘制或以其它方式呈现在显示器214上的一个或多个图形用户界面(GUI)。通过典型地对键盘202和鼠标203进行操作,计算机系统200的用户和应用可以在功能上可适用的方式对界面进行操作,以将控制命令和/或输入提供至与这些(一个或多个)GUI相关联的应用。还可以实现在功能上可适用的其它形式的用户界面,诸如利用经由扬声器217所输出的语音提示和经由麦克风280所输入的用户声音命令的音频界面等。

[0077] 图2B是处理器205和“存储器”234的详细示意框图。存储器234表示图2A中的计算

机模块201可以访问的(包括HDD 209和半导体存储器206的)所有存储器模块的逻辑聚合。

[0078] 在初始对计算机模块201通电的情况下,执行上电自检(power-on self-test, POST)程序250。通常将POST程序250存储在图2A的半导体存储器206的ROM 249中。有时将诸如存储有软件的ROM 249等的硬件装置称为固件。POST程序250检查计算机模块201内的硬件以确保适当工作,并且通常检查处理器205、存储器234(209,206)和通常还存储在ROM 249中的基本输入-输出系统软件(BIOS)模块251,以进行正确操作。一旦POST程序250成功运行,BIOS 251启动图2A的硬盘驱动器210。启动硬盘驱动器210使得经由处理器205执行驻留在硬盘驱动器210上的引导装入程序252。这样将操作系统253载入RAM存储器206,其中在该RAM存储器206上,操作系统253开始工作。操作系统253是处理器205可执行的系统级应用,以实现包括处理器管理、存储器管理、装置管理、存储管理、软件应用接口和通用用户界面等的各种高级功能。

[0079] 操作系统253管理存储器234(209,206),以确保计算机模块201上运行的各处理或应用具有在不会与分配至其它处理的存储器冲突的情况下执行的充足存储器。此外,必须适当使用图2A的计算机系统200中可用的不同类型的存储器,以使得各处理可以高效地运行。因此,聚合存储器234并不意图例示如何分配存储器的特定分段(除非另外说明),而是提供计算机系统200可访问的存储器的概述图以及如何使用该存储器。

[0080] 如图2B所示,处理器205包括多个功能模块,其中这多个功能模块包括控制单元239、算术逻辑单元(ALU)240和有时称为高速缓冲存储器的本地或内部存储器248。高速缓冲存储器248在寄存器区段中通常包括多个存储寄存器244-246。一个或多个内部总线241从功能上使这些功能模块相互连接。处理器205通常还具有用于使用接线218经由系统总线204与外部装置进行通信的一个或多个接口242。存储器234使用接线219连接至总线204。

[0081] 应用程序233包括可以包含条件分支指令和循环指令的指令序列231。程序233还可以包括执行程序233时所使用的数据232。将指令231和数据232分别存储在存储器位置228、229、230和235、236、237中。根据指令231和存储器位置228-230的相对大小,如存储器位置230中示出的指令所描述的,可以将特定指令存储在单个存储器位置中。可选地,如存储器位置228和229中示出的指令段所描述的,可以将指令分割成各自被存储在单独的存储器位置的多个部分。

[0082] 通常,向处理器205赋予一组指令,其中在该处理器205内执行该组指令。处理器205等待随后输入,其中处理器205通过执行另一组指令来对该随后输入作出反应。可以从多个源中的一个或多个源提供各输入,其中该输入包括输入装置202、203中的一个或多个所生成的数据、从外部源经由网络220、202其中之一所接收到的数据、从存储装置206、209其中之一所检索到的数据或者从插入相应的读取器212内的存储介质225所检索到的数据(所有这些均在图2A中示出)。执行一组指令在一些情况下可能会导致输出数据。执行还可能涉及将数据或变量存储至存储器234。

[0083] 视频编码器114、视频解码器134和所述方法可以使用存储器234内的相应存储器位置255、256、257中所存储的输入变量254。视频编码器114、视频解码器134和所述方法产生存储器234内的相应存储器位置262、263、264中所存储的输出变量261。可以将中间变量258存储在存储器位置259、260、266和267中。

[0084] 参考图2B的处理器205,寄存器244、245、246、算术逻辑单元(ALU)240和控制单元239一起工作以进行微操作序列,其中这些微操作序列是针对构成程序233的指令集中的各指令进行“提取、解码和执行”周期所需的。各提取、解码和执行周期包括:

[0085] 提取操作,用于从存储器位置228、229、230提取或读取指令231;

[0086] 解码操作,其中在该解码操作中,控制单元239判断提取了哪个指令;以及

[0087] 执行操作,其中在该执行操作中,控制单元239和/或ALU 240执行该指令。

[0088] 之后,可以执行针对下一指令的进一步提取、解码和执行周期。同样,可以进行存储周期,其中通过该存储周期,控制单元239将值存储至或写入存储器位置232。

[0089] 要说明的图13至图18的方法中的各步骤或子处理与程序233的一个或多个于一个区段相关联,并且通常通过处理器205中的寄存器部244、245、247、ALU 240和控制单元239一起工作以针对程序233的所述分段的指令集中的各指令进行提取、解码和执行周期,来进行该步骤或子处理。

[0090] 图3是示出视频编码器114的功能模块的示意框图。图4是示出视频解码器134的功能模块的示意框图。通常,数据以样本或系数的组(诸如块向固定大小的子块的分割等)或者作为阵列在视频编码器114和视频解码器134内的功能模块之间传递。如图2A和2B所示,可以使用通用计算机系统200来实现视频编码器114和视频解码器134,其中可以利用计算机系统200内的专用硬件、利用计算机系统200内可执行的软件(诸如驻留在硬盘驱动器205上并且由处理器205控制其执行的软件应用程序233的一个或多个于一个软件代码模块等),来实现各种功能模块。可替代地,可以利用在计算机系统200内可执行的专用硬件和软件的组合来实现视频编码器114和视频解码器134。可以可替代地在诸如进行所述方法的功能或子功能的一个或多个于一个集成电路等的专用硬件中实现视频编码器114、视频解码器134和所述方法。这种专用硬件可以包括图形处理单元(GPU)、数字信号处理器(DSP)、专用标准产品(ASPP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或者一个或多个于一个微处理器和关联存储器。特别地,视频编码器114包括模块310-390,并且视频解码器134包括模块420-496,其中这些模块各自可被实现为软件应用程序233的一个或多个于一个软件代码模块。

[0091] 尽管图3的视频编码器114是通用视频编码(VVC)视频编码流水线的示例,但也可以使用其它视频编解码器来进行这里所述的处理阶段。视频编码器114接收诸如一系列帧(各帧包括一个或多个于一个颜色通道)等的所捕获帧数据113。帧数据113可以处于任何色度格式,例如4:0:0、4:2:0、4:2:2或4:4:4色度格式。块分区器310首先将帧数据113分割成CTU,CTU的形状通常为正方形,并且被配置成使得使用CTU的特定大小。例如,CTU的大小可以是 64×64 、 128×128 或 256×256 个亮度样本。块分区器310进一步将各CTU分割成与亮度编码树或色度编码树相对应的一个或多个于一个CB。亮度通道还可以被称为主颜色通道。各个色度通道还可以被称为二次颜色通道。CB具有各种大小并且可以包括正方形和非正方形长宽比这两者的。参考图13-15进一步描述块分区器310的操作。然而,在VVC标准中,CB、CU、PU和TU总是具有2的幂的边长。因而,从块分区器310输出当前CB(表示为312),从而根据对CTU的一个或多个于一个块的迭代、根据CTU的亮度编码树和色度编码树而前进。以下参考图5和6来进一步说明用于将CTU分区为CB的选项。尽管操作大体上是以CTU为单位描述的,但视频编码器114和视频解码器134可以在较小大小的区域上操作以减少存储器消耗。例如,各

个CTU可以被分割成较小的区域,被称为大小为 64×64 的“虚拟流水线数据单元”(VPDU)。VPDU形成更适于在硬件架构中的流水线处理的数据粒度,其中,与对完整CTU的操作相比,存储器占用面积的减少减小了硅面积并且因此减小了成本。

[0092] 从帧数据113的第一次分割得到的CTU可以是按光栅扫描顺序扫描的,并可以被分成组成一个或多个“条带(slice)”。条带可以是“帧内”(或“I”)条带。帧内条带(I条带)指示条带中的每个CU都是帧内预测的。可选地,条带可以是单预测或双预测的(分别为“P”或“B”条带),分别指示条带中的单预测和双预测的附加可用性。

[0093] 在I条带中,各个CTU的编码树可以在 64×64 层级以下发散成两个单独的编码树,一个用于亮度而另一个用于色度。使用单独的树允许不同的块结构存在于CTU的亮度 64×64 区域内的亮度与色度之间。例如,大的色度CB可与许多较小的亮度CB并置,反之亦然。在P或B条带中,CTU的单个编码树定义亮度和色度共同的块结构。单个树的所得块可以是帧内预测或帧间预测的。

[0094] 对于各CTU,视频编码器114在两个阶段中操作。在第一阶段(称为“搜索”阶段),块分区器310测试编码树的各种潜在配置。编码树的各个潜在配置具有关联的“候选”CB。第一阶段涉及测试各种候选CB,以选择提供相对高的压缩效率和相对低的失真的CB。该测试通常涉及拉格朗日优化,由此基于速率(编码成本)和失真(关于输入帧数据113的误差)的加权组合来评价候选CB。选择“最佳”候选CB(具有最低评价速率/失真的CB)以用于随后编码在位流115中。候选CB的评价中包括如下选项:将CB用于给定区域,或者根据各种拆分选项来拆分该区域并利用其它CB来编码各个较小所得区域或更进一步拆分区。结果,在搜索阶段中选择编码树和CB本身这两者。

[0095] 视频编码器114针对各CB(例如,CB 312)产生由箭头320指示的预测块(PB)。PB 320是关联的CB 312的内容的预测。减法器模块322产生PB 320和CB 312之间的表示为324的差(或“残差”,其是指差在空间域中)。差324是PB 320和CB 312中的相应样本之间的块大小差。差324被变换、量化和表示为由箭头336指示的变换块(TB)。通常从多个可能候选CB之一(例如,基于所评价的成本或失真)选择PB 320和关联的TB 336。

[0096] 候选编码块(CB)是针对关联的PB和所得到的残差从视频编码器114可用的预测模式其中之一得到的CB。当与视频解码器114中的预测PB组合时,TB 336以位流中的附加信令为代价来减少解码CB和原始CB 312之间的差。

[0097] 因而,各候选编码块(CB)(即预测块(PB)与变换块(TB)的组合)具有关联的编码成本(或“速率”)和关联的差(或“失真”)。CB的失真通常被估计为样本值的差,诸如绝对差和(SAD)或平方差和(SSD)等。模式选择器386可以使用差324来确定从各候选PB得到的估计,以确定预测模式387。预测模式387指示针对当前CB使用特定预测模式(例如,帧内预测或帧间预测)的决定。可以以与残差的熵编码相比明显更低的成本来进行与各候选预测模式相关联的编码成本和相应的残差编码的估计。因此,即使在实时视频编码器中,也可以评价多个候选模式,以确定率失真意义上的最佳模式。

[0098] 确定率失真方面的最佳模式通常是使用拉格朗日优化的变形来实现的。

[0099] 可以采用拉格朗日或类似的优化处理来进行(利用块分区器310的)CTU向CB的最佳分区的选择以及从多个可能性中的最佳预测模式的选择这两者。通过将候选模式的拉格朗日优化处理应用在模式选择器模块386中,选择成本测量最低的帧内预测模式作为最佳

模式。最低成本模式是所选择的二次变换索引388,并且也由熵编码器338编码在位流115中。

[0100] 在视频编码器114的操作的第二阶段(称为“编码”阶段)中,在视频编码器114中进行对各个CTU的(一个或多个)所确定编码树的迭代。对于使用单独树的CTU,针对CTU的每个 64×64 亮度区域,首先对亮度编码树进行编码,随后对色度编码树进行编码。在亮度编码树内仅亮度CB被编码,并且在色度编码树内仅色度CB被编码。对于使用共用树的CTU,单个树根据共用树的公共块结构来描述CU,即,亮度CB和色度CB。

[0101] 熵编码器338支持句法元素的可变长度编码和句法元素的算术编码这两者。诸如“参数集”等的位流的部分(例如,序列参数集(SPS)和图片参数集(PPS))使用固定长度码字和可变长度码字的组合。条带(也称为连续部分)具有使用可变长度编码的条带头部,随后是使用算术编码的条带数据。条带头部限定当前条带特有的参数,诸如条带级量化参数偏移等。条带数据包括条带中的各个CTU的句法元素。使用可变长度编码和算术编码需要在位流的各个部分内的顺序解析。可以用起始码描述这些部分以形成“网络抽象层单元”或“NAL单元”。使用上下文自适应二进制算术编码处理来支持算术编码。算术编码的句法元素由一个或多个“bin(二进制文件)”的序列组成。与位一样,bin的值为“0”或“1”。然而,bin未作为离散位编码在位流115中。bin具有关联预测(或“可能”或“最大概率”)值和关联概率(被称为“上下文”)。当要编码的实际bin与预测值匹配时,对“最大概率符号”(MPS)进行编码。对最大概率符号进行编码在位流115中的消耗位(包括总计小于一个离散位的成本)方面相对便宜。当要编码的实际bin与可能值不匹配时,对“最小概率符号”(LPS)进行编码。对最小概率符号进行编码在消耗位方面具有相对高的成本。bin编码技术使得能够使“0”vs“1”的概率歪斜的bin进行高效编码。对于具有两个可能值的句法元素(即,“flag(标志)”),单个bin就足够了。对于具有许多可能值的句法元素,需要bin的序列。

[0102] 可以基于该序列中的较前bin的值来确定该序列中的较后bin的存在。另外,各bin可以与多于一个的上下文相关联。可以根据句法元素中的较前bin和相邻句法元素的bin值(即,来自相邻块的bin值)等来选择特定上下文。每次对上下文编码bin进行编码时,以反映出新bin值的方式来更新针对该bin(如果有)选择的上下文。如此,二进制算术编码方案被认为是自适应的。

[0103] 视频编码器114也支持缺少上下文的bin(“旁路bin”)。假定“0”和“1”之间的等概率分布来对旁路bin进行编码。因而,各个bin具有位流115中的一位的编码成本。不存在上下文节省了存储器并降低了复杂度,因而使用特定bin的值的分布未歪斜的旁路bin。采用上下文和自适应的熵编码器的一个示例在本领域中被称为CABAC(上下文自适应二进制算术编码器),并且在视频编码中采用了该编码器的许多变体。

[0104] 熵编码器338使用上下文编码的bin和旁路编码的bin的组合来编码量化参数392,并且如果用于当前CB,则编码LFNST索引388。量化参数392是使用“增量QP”编码的。在称为“量化组”的各个区域中,最多用信号通知一次增量QP。量化参数392被应用于亮度CB的残差系数。经调整的量化参数被应用于并置的色度CB的残差系数。经调整的量化参数可以包括根据映射表及选自偏移列表的CU层级偏移而从亮度量化参数392进行映射。当与变换块相关联的残差仅在经受通过应用二次变换而变换成主系数的那些系数位置中包括有效残差系数时,用信号通知二次变换索引388。

[0105] 多路复用器模块384根据从各个候选CB的测试预测模式中选择所确定的最佳帧内预测模式,从帧内预测模块364输出PB 320。候选预测模式不需要包括由视频编码器114支持的每个可设想预测模式。帧内预测分为三种类型。“DC帧内预测”涉及用表示附近重建样本的平均值的单个值来填充PB。“平面帧内预测”涉及用根据平面的样本来填充PB,其中,DC偏移以及垂直和水平梯度从附近的重建的相邻样本导出。附近的重建样本通常包括在当前PB之上的一行重建样本(向PB的右侧延伸一定程度)和在当前PB的左侧的一列重建样本(向PB之外向下延伸一定程度)。“角度帧内预测”涉及利用重建的相邻样本来填充PB,该重建的相邻样本被滤波并且在特定方向(或“角度”)上跨PB传播。在VVC中,支持65个角度,其中矩形块能够利用正方形块所不可用的额外角度来产生总共87个角度。第四类型的帧内预测可用于色度PB,从而根据“跨分量线性模型”(CCLM)模式从并置的亮度重建样本来生成PB。三种不同的CCLM模式是可用的,各个模式使用从相邻的亮度和色度样本导出的不同模型。所导出的模型用以从并置的亮度样本生成用于色度PB的样本块。

[0106] 在先前重建的样本不可用的情况下(例如,在帧的边缘处),使用样本范围的一半的默认半色调值。例如,对于10位视频,使用512的值。由于没有先前样本可用于位于帧的左上位置处的CB,因此角度和平面帧内预测模式产生与DC预测模式相同的输出,即具有半色调值作为幅度的样本的平坦平面。

[0107] 对于帧间预测,通过运动补偿模块380使用来自按位流中的编码帧的顺序在当前帧之前的一个或两个帧的样本来产生预测块382并且由多路复用器模块384输出为PB 320。此外,对于帧间预测,单个编码树通常用于亮度通道和色度通道这两者。位流中的编码帧的顺序可能不同于当捕获或显示时的帧的顺序。当一个帧用于预测时,该块被称为“单预测”并且具有两个关联的运动矢量。当两个帧用于预测时,该块被称为“双预测”并且具有两个关联的运动矢量。对于P条带,各CU可以被帧内预测或单预测。对于B条带,各CU可以被帧内预测、单预测或双预测。通常使用“图片组”结构来编码帧,从而实现帧的时间层级结构。帧可以被分割成多个条带,各个条带编码帧的一部分。帧的时间层级结构允许帧参考按显示帧的顺序的先前和后续的图片。图像按确保满足解码各帧的依赖关系所必需的顺序来编码。

[0108] 根据运动矢量378和参考图片索引来选择样本。运动矢量378和参考图片索引适用于所有颜色通道,因而主要就对PU而非PB的操作来描述帧间预测,即,使用单个编码树来描述将各个CTU分解成一个或多于一个帧间预测块。帧间预测方法在运动参数的数量及其精度上可能变化。运动参数通常包括参考帧索引(其指示将使用来自参考帧列表的哪些参考帧加上参考帧各自的空间平移),但可以包括更多帧、特殊帧、或者诸如缩放和旋转等的复杂仿射参数。另外,可以应用预先确定的运动细化处理以基于参考样本块生成密集运动估计。

[0109] 在确定并选择了PB 320、并在减法器322处从原始样本块中减去了PB 320的情况下,获得编码成本最低的残差(表示为324)并对该残差进行有损压缩。有损压缩处理包括变换、量化和熵编码的步骤。正向主变换模块326对差324应用正向变换,从而将差324从空间域转换到频率域并且产生由箭头328表示的主变换系数。一维中的最大主变换大小是32点DCT-2或64点DCT-2变换。如果正编码的CB大于表示为块大小的最大支持主变换大小(即,64×64或32×32),则以区块方式应用主变换326以变换差324的所有样本。变换326的应用得

到CB的多个TB。在各个变换应用在比 32×32 大(例如, 64×64)的差324的TB上操作的情况下,在TB的左上 32×32 区域之外的所有得到的主变换系数328被设置为零,即丢弃。剩余的主变换系数328被传递至量化器模块334。根据与CB相关联的量化参数392量化主变换系数328以产生主变换系数332。量化参数392可以针对亮度CB相对于各个色度CB而不同。通过进行不可分离的二次变换(NSST)操作或绕过二次变换,将主变换系数332传递到正向二次变换模块330以产生由箭头336表示的变换系数。正向主变换通常是可分离的,变换各个TB的一组行,然后变换一组列。对于宽度和高度不超过16个样本的亮度TB,正向主变换模块326在水平和垂直方向上使用II型离散余弦变换(DCT-2),或者在水平和垂直方向上绕过变换,或者在水平或垂直方向上使用VII型离散正弦变换(DST-7)和VIII型离散余弦变换(DCT-8)的组合。DST-7和DCT-8的组合的使用在VVC标准中被称为“多变换选择集合”(MTS)。

[0110] 模块330的正向二次变换通常为不可分离变换,其仅应用于经帧内预测CU的残差且仍可被绕过。正向二次变换对16个样本(被布置为主变换系数328的左上 4×4 子块)或48个样本(被布置为主变换系数328的左上 8×8 系数中的三个 4×4 子块)进行操作以产生二次变换系数集合。二次变换系数集合的数量可以少于导出其的主变换系数集合的数量。由于二次变换仅应用于彼此相邻并包括DC系数的系数集合,因此二次变换被称为“低频不可分离的二次变换”(LFNST)。此外,当应用LFNST时,TB中的所有剩余系数必须在主变换域和二次变换域两者中都为零。

[0111] 量化参数392对于给定TB是恒定的,并且因此得到在TB的主变换域中产生的残差系数的均匀缩放。量化参数392可以通过用信号通知的“增量量化参数”定期地变化。对于包含在给定区域内的CU(称为“量化组”),用信号通知增量量化参数(增量QP)一次。如果CU大于量化组大小,则通过CU的TB之一用信号通知增量QP一次。也就是说,针对CU的第一量化组,熵编码器338用信号通知增量QP一次,并且针对CU的任何后续量化组,不用信号通知增量QP。通过应用“量化矩阵”,非均匀缩放也是可能的,由此应用于各个残差系数的缩放因子源自量化参数392与缩放矩阵中的相应条目的组合。缩放矩阵可以具有小于TB的大小的尺寸,并且当应用于TB时,最近邻方法被用于根据大小小于TB大小的缩放矩阵来提供针对各个残差系数的缩放值。残差系数336被供应至熵编码器338以在位流115中编码。通常,根据扫描模式,扫描TU的具有至少一个有效残差系数的各个TB的残差系数以产生值的有序列表。扫描模式通常扫描作为 4×4 “子块”的序列的TB,以 4×4 组残差系数的粒度提供规则的扫描操作,其中子块的布置取决于TB的大小。各个子块内的扫描和从一个子块到下一子块的进展通常遵循后向对角线扫描模式。另外,量化参数392使用增量QP句法元素被编码在位流115中,并且二次变换索引388在参考图13至图15所描述的条件下被编码在位流115中。

[0112] 如上所述,视频编码器114需要访问与在视频解码器134中看到的经编码帧表示相对应的帧表示。因而,残差系数336通过逆二次变换模块344(根据二次变换系数388操作)以产生由箭头342表示的中间逆变换系数。中间逆变换系数346根据量化参数392由去量化模块340逆量化以产生由箭头346表示的残差样本。中间逆变换系数346被传递到逆主变换模块348以产生TU的由箭头350表示的残差样本。逆二次变换模块344进行的逆变换类型与由正向二次变换模块330进行的正向变换类型相对应。逆主变换模块348进行的逆变换类型与由主变换模块326进行的主变换类型相对应。求和模块352将残差样本350和PU 320相加以产生CU的重建样本(由箭头354指示)。

[0113] 重建样本354被传递到参考样本高速缓冲存储器356和环内滤波器模块368。通常使用ASIC上的静态RAM实现(因此避免了昂贵的片外存储器访问)的参考样本高速缓冲存储器356提供了满足用于为帧中的后续CU生成帧内PB的依赖关系所需的最小样本存储。最小依赖关系通常包括沿着一行CTU的底部的样本的“线缓冲器”,以供下一行CTU以及范围由CTU的高度设置的列缓冲使用。参考样本高速缓冲存储器356将参考样本(由箭头358表示)供给至参考样本滤波器360。样本滤波器360应用平滑操作以产生滤波参考样本(由箭头362指示)。滤波参考样本362由帧内预测模块364使用以产生由箭头366表示的样本的帧内预测块。对于各候选帧内预测模式,帧内预测模块364产生样本块即366。样本块366由模块364使用诸如DC、平面或角度帧内预测等的技术来生成。

[0114] 环内滤波器模块368对重建样本354应用数个滤波阶段。滤波阶段包括“去块滤波器”(DBF),该DBF应用与CU边界对齐的平滑化,以减少由不连续而产生的伪影。环内滤波器模块368中存在的另一滤波阶段是“自适应环路滤波器”(ALF),该ALF应用基于Wiener的自适应滤波器以进一步降低失真。环内滤波器模块368中的另一可用滤波阶段是“样本自适应偏移”(SAO)滤波器。SAO滤波器通过首先将重建样本分类为一个或多个类别、并且根据所分配的类别在样本级别应用偏移来工作。

[0115] 从环内滤波器模块368输出由箭头370表示的滤波样本。滤波样本370被存储在帧缓冲器372中。帧缓冲器372通常具有存储数个(例如,多达16个)图片的容量,因而存储在存储器206中。由于所需的大存储器消耗,因此帧缓冲器372通常不使用片上存储器来存储。如此,对帧缓冲器372的访问在存储器带宽方面是昂贵的。帧缓冲器372将参考帧(由箭头374表示)提供至运动估计模块376和运动补偿模块380。

[0116] 运动估计模块376估计多个“运动矢量”(表示为378),其各自是相对于当前CB的位置的笛卡尔空间偏移,从而参考帧缓冲器372中的参考帧其中之一中的块。针对各运动矢量产生参考样本的滤波块(表示为382)。滤波参考样本382形成可供模式选择器386的潜在选择用的进一步候选模式。此外,对于给定CU,PB 320可以使用一个参考块(“单预测”)形成,或者可以使用两个参考块(“双预测”)形成。对于所选择的运动矢量,运动补偿模块380根据支持运动矢量中的子像素精度的滤波处理来产生PU 320。如此,运动估计模块376(其对许多候选运动矢量进行操作)与运动补偿模块380(其仅对所选择的候选进行操作)相比可以进行简化滤波处理,以实现降低的计算复杂度。当视频编码器114针对CU选择帧间预测时,运动矢量378被编码在位流115中。

[0117] 尽管参考通用视频编码(VVC)说明了图3的视频编码器114,但其它视频编码标准或实现也可以采用模块310-390的处理阶段。帧数据113(和位流115)也可以从存储器206、硬盘驱动器210、CD-ROM、蓝光盘(Blue-ray disk™)或其它计算机可读存储介质中读取(或者被写入存储器206、硬盘驱动器210、CD-ROM、蓝光盘或其它计算机可读存储介质)。另外,帧数据113(和位流115)可以从外部源(诸如连接至通信网络220的服务器或者射频接收器等)接收(或者被发送至该外部源)。通信网络220可以提供有限的带宽,从而需要在视频编码器114中使用速率控制以避免在帧数据113难以压缩时使网络饱和。此外,可以从表示帧数据113的空间部分(CTU集合)的一个或多个条带构建位流115,这些条带由视频编码器114的一个或多个实例产生、在处理器205的控制下以协调方式操作。在本发明的上下文下,条带还可被称为位流的“连续部分”。条带在位流内是连续的,并且(例如如果正使

用并行处理)可编码或解码为单独部分。

[0118] 在图4中示出视频解码器134。尽管图4的视频解码器134是通用视频编码(VVC)视频解码流水线的示例,但其它视频编解码器也可用于进行本文所述的处理阶段。如图4所示,位流133被输入到视频解码器134。位流133可以从存储器206、硬盘驱动器210、CD-ROM、蓝光盘或其它非暂时性计算机可读存储介质中读取。可替代地,位流133可以从外部源(诸如连接至通信网络220的服务器或者射频接收器等)接收。位流133包含表示要解码的所捕获帧数据的编码句法元素。

[0119] 位流133被输入到熵解码器模块420。熵解码器模块420通过解码“bin”序列从位流133中提取句法元素并将句法元素的值传递到视频解码器134中的其它模块。熵解码器模块420使用可变长度和固定长度解码来解码SPS、PPS或条带头部,使用算术解码引擎来将条带数据的句法元素解码为一个或多个bin的序列。各个bin可以使用一个或多个“上下文”,其中上下文描述用于为bin编码“一”和“零”值的概率水平。在多个上下文可用于给定bin的情况下,进行“上下文建模”或“上下文选择”步骤以选择可用上下文之一来解码bin。对bin进行解码的处理形成了一个顺序反馈循环,因此,各个条带可以由给定的熵解码器420实例整体解码。单个(或几个)高执行熵解码器420实例可以从位流115中解码帧的所有条带,多个低执行熵解码器420实例可以同时从位流133中解码帧的条带。

[0120] 熵解码器模块420应用算术编码算法,例如“上下文自适应二进制算术编码”(CABAC),以从位流133解码句法元素。所解码的句法元素用于重建视频解码器134内的参数。参数包括残差系数(由箭头424表示)、量化参数474、二次变换索引470以及诸如帧内预测模式等的模式选择信息(由箭头458表示)。模式选择信息还包括诸如运动矢量等的信息、以及各CTU向一个或多个CB的分区。参数用于通常与来自先前解码的CB的样本数据组合生成PB。

[0121] 残差系数424被传递到逆二次变换模块436,其中根据参考图16至18描述的方法应用二次变换或不进行操作(旁路)。逆二次变换模块436从二次变换域系数产生重建变换系数432,即主变换域系数。重建变换系数432被输入到去量化器模块428。去量化器模块428对残差系数432进行逆量化(或“缩放”),即,在主变换系数域中,以根据量化参数474创建由箭头440表示的重建中间变换系数。如果在位流133中指示使用非均匀逆量化矩阵,则视频解码器134从位流133读取量化矩阵作为缩放因子序列,并且根据将缩放因子排列成矩阵。逆缩放将量化矩阵与量化参数组合使用以创建重建中间变换系数440。

[0122] 重建变换系数440被传递到逆主变换模块444。模块444将系数440从频率域变换回到空间域。模块444的操作的结果是由箭头448表示的残差样本的块。残差样本448的块在大小上等于相应的CB。残差样本448的块被供给至求和模块450。在求和模块450处,将残差样本448加到表示为452的解码PB,以产生由箭头456表示的重建样本的块。重建样本456被供给至重建样本高速缓冲存储器460和环内滤波模块488。环内滤波模块488产生表示为492的帧样本的重建块。帧样本492被写入帧缓冲器496。

[0123] 重建样本高速缓冲存储器460以类似于视频编码器114的重建样本高速缓冲存储器356的方式操作。重建样本高速缓冲存储器460在无存储器206的情况下(例如,通过作为代替使用通常是片上存储器的数据232)为对后续CB进行帧内预测所需的重建样本提供存储。由箭头464表示的参考样本是从重建样本高速缓冲存储器460获得的,并被供给至参考

样本滤波器468以产生由箭头472表示的滤波参考样本。滤波参考样本472被供给至帧内预测模块476。模块476根据在位流133中表示的并由熵解码器420解码的帧内预测模式参数458,产生由箭头480表示的帧内预测样本的块。使用诸如DC、平面或角度帧内预测等的模式来生成样本480的块。

[0124] 当在位流133中指示CB的预测模式使用帧内预测时,帧内预测样本480经由复用器模块484形成解码的PB 452。帧内预测产生样本的预测块(PB),即,使用同一颜色分量中的“相邻样本”导出的一个颜色分量中的块。相邻样本是与当前块相邻的样本,并且由于在块解码顺序中处于前面因此已经被重建。在亮度和色度块并置的情况下,亮度和色度块可以使用不同的帧内预测模式。然而,两个色度通道共用相同的帧内预测模式。

[0125] 当在位流133中指示CB的预测模式是帧内预测时,运动补偿模块434使用运动矢量(通过熵解码器420从位流133中解码)和参考帧索引从帧缓冲器496中选择和过滤样本块498,来产生表示为438的帧间预测样本的块。样本块498是从帧缓冲器496中所存储的先前解码帧获得的。为了进行双预测,产生两个样本块并将这两个样本块混合在一起以产生解码PB 452的样本。帧缓冲器496由来自环内滤波模块488的滤波块数据492填充。与视频编码器114的环内滤波模块368一样,环内滤波模块488应用DBF、ALF和SAO滤波操作中的任意。通常,运动矢量被应用于亮度和色度通道这两者,但在亮度和色度通道中用于子样本插值的滤波处理是不同的。

[0126] 图5是示出通用视频编码的树结构中的区域向一个或多个子区域的可用分割或拆分的集合500的示意框图。如参考图3所述,集合500中示出的分割可供编码器114的块分区器310利用以根据如通过拉格朗日优化所确定的编码数将各CTU分割成一个或多个一个CU或CB。

[0127] 尽管集合500仅示出将正方形区域分割成其它可能非正方形子区域,但应当理解,集合500正示出编码树中的父节点到编码树中的子节点的潜在分割,并且未要求父节点对应于正方形区域。如果包含区域为非正方形,则根据包含块的长宽比来对从分割得到的块的尺寸进行缩放。一旦区域未被进一步拆分,即,在编码树的叶节点处,CU占据该区域。

[0128] 将区域子分割成子区域的处理必须在所得到的子区域达到最小CU大小(通常为4×4亮度样本)时终止。除了将CU约束为禁止块区域小于例如16个样本的预定最小大小之外,将CU约束为具有为四的最小宽度或高度。就宽度和高度或者就宽度或高度这两而言其它最小值也是可能的。子分割处理也可以在分解的最深层之前终止,从而得到大于最小CU大小的CU。有可能不发生拆分,从而得到占据整个CTU的单个CU。占据整个CTU的单个CU是最大可用编码单元大小。由于使用经子采样的色度格式(诸如4:2:0等),视频编码器114和视频解码器134的布置可以比亮度通道中更早地终止色度通道中的区域的拆分,包括定义亮度和色度通道的块结构的共用编码树的情况。当单独的编码树用于亮度和色度时,对可用拆分操作的约束确保16个样本的最小色度CB区域,即使这种CB与更大的亮度区域(例如,64个亮度样本)并置也是如此。

[0129] 在不存在进一步子分割的情况下,在编码树的叶节点处存在CU。例如,叶节点510包含一个CU。在编码树的非叶节点处,存在向两个或更多个其它节点的拆分,其中各节点可以是形成一个CU的叶节点、或者是包含向更小区域的进一步拆分的非叶节点。在编码树的各个叶节点处,针对各个颜色通道存在一个编码块。针对亮度和色度这两者在相同的深度

处终止的拆分得到三个并置CB。针对亮度比针对色度在更深的深度处终止的拆分得到与色度通道的CB并置的多个亮度CB。

[0130] 如图5所示,四叉树拆分512将包含区域分割成四个大小相等的区域。与HEVC相比,通用视频编码(VVC)通过附加拆分(包括水平二元拆分514和垂直二元拆分516)实现了附加的灵活性。拆分514和516各自将包含区域分割成两个大小相等的区域。分割沿着包含块内的水平边界(514)或垂直边界(516)。

[0131] 在通用视频编码中通过添加三元水平拆分518和三元垂直拆分520实现了进一步的灵活性。三元拆分518和520将块分割成沿着包含区域宽度或高度的1/4和3/4在水平方向(518)或垂直方向(520)上形成界限的三个区域。四叉树、二叉树和三叉树的组合被称为“QTBT”。“树的根包括零个或多个四叉树拆分(树的“QT”部分)。一旦QT部分终止,可以发生零个或多个二元或三元拆分(“多树”或树的“MT”部分),最终在树的叶节点处的CB或CU中结束。在树描述所有颜色通道的情况下,树的叶节点为CU。在树描述亮度通道或色度通道的情况下,树的叶节点为CB。

[0132] 与仅支持四叉树、因而仅支持正方形块的HEVC相比,QTBT特别是考虑到二叉树和/或三叉树拆分的可能递归应用而得到更多可能的CU大小。当仅四叉树拆分可用时,编码树深度的各个增加对应于CU大小减小到父区域的大小的四分之一。在VVC中,二元和三元拆分的可用性意味着编码树深度不再直接对应于CU区域。可以通过约束拆分选项以消除将得到小于四个样本的块宽度或高度或者将得到不是四个样本的倍数的拆分来降低异常(非正方形)块大小的可能性。通常,约束将在考虑亮度样本时适用。然而,在所描述的布置中,约束可以单独应用于色度通道的块。拆分选项对色度通道的约束的应用可能针对亮度vs色度(例如,在帧数据采用4:2:0色度格式或4:2:2色度格式时)得到不同的最小块大小。各拆分产生具有相对于包含区域不变的、二等分的或四等分的边尺寸的子区域。然后,由于CTU大小为2的幂,因此所有CU的边尺寸也为2的幂。

[0133] 图6是示出在通用视频编码中使用的QTBT(或“编码树”)结构的数据流600的示意图。将QTBT结构用于各CTU以定义CTU向一个或多个CU的分割。各CTU的QTBT结构由视频编码器114中的块分区器310确定,并被编码到流115中或者由视频解码器134中的熵解码器420从流133解码。根据图5所示的分割,数据流600进一步表现可供块分区器310将CTU分割成一个或多个CU用的许可组合的特征。

[0134] 从层级结构的顶层开始、即在CTU处,首先进行零个或多个四叉树分割。具体地,由块分区器310作出四叉树(QT)拆分决策610。610处的决策返回“1”符号,这表明决定根据四叉树拆分512将当前节点拆分成四个子节点。结果是诸如在620处等生成四个新节点,并且针对各新节点,递归回到QT拆分决策610。各新节点均是按光栅(或Z扫描)顺序考虑的。可替代地,如果QT拆分决策610指示不进行进一步拆分(返回“0”符号),则四叉树分区停止,并且随后考虑多树(MT)拆分。

[0135] 首先,由块分区器310作出MT拆分决策612。在612处,指示进行MT拆分的决策。在决策612处返回“0”符号,这表明将不进行节点向子节点的进一步拆分。如果将不进行节点的进一步拆分,则节点是编码树的叶节点并且对应于CU。在622处输出叶节点。可替代地,如果MT拆分612指示决定进行MT拆分(返回“1”符号),则块分区器310进入方向决策614。

[0136] 方向决策614将MT拆分的方向指示为水平(“H”或“0”)或垂直(“V”或“1”)。如果决

策614返回指示水平方向的“0”，则块分区器310进入决策616。如果决策614返回指示垂直方向的“1”，则块分区器310进入决策618。

[0137] 在决策616和618各自中，在BT/TT拆分时将MT拆分的分区数量指示为两个（二元拆分或“BT”节点）或三个（三元拆分或“TT”）。也就是说，当从614指示的方向为水平时，由块分区器310作出BT/TT拆分决策616，并且当从614指示的方向为垂直时，由块分区器310作出BT/TT拆分决策618。

[0138] BT/TT拆分决策616指示水平拆分是通过返回“0”所指示的二元拆分514、还是通过返回“1”所指示的三元拆分518。当BT/TT拆分决策616指示二元拆分时，在生成HBT CTU节点的步骤625处，块分区器310根据二元水平拆分514生成两个节点。当BT/TT拆分616指示三元拆分时，在生成HTT CTU节点的步骤626处，块分区器310根据三元水平拆分518生成三个节点。

[0139] BT/TT拆分决策618指示垂直拆分是通过返回“0”所指示的二元拆分516、还是通过返回“1”所指示的三元拆分520。当BT/TT拆分618指示二元拆分时，在生成VBT CTU节点的步骤627处，块分区器310根据垂直二元拆分516生成两个节点。当BT/TT拆分618指示三元拆分时，在生成VTT CTU节点的步骤628处，块分区器310根据垂直三元拆分520生成三个节点。对于从步骤625-628得到的各节点，根据方向614按从左到右或从上到下的顺序来应用数据流600返回到MT拆分决策612的递归。结果，可以应用二叉树和三叉树拆分以生成具有各种大小的CU。

[0140] 图7A和7B提供CTU 710向多个CU或CB的示例分割700。在图7A中示出示例CU 712。图7A示出CTU 710中的CU的空间排列。示例分割700在图7B中也被示出为编码树720。

[0141] 在图7A的CTU 710中的各非叶节点（例如，节点714、716和718）处，按“Z顺序”扫描或遍历所包含的节点（其可以是进一步分割的或者可以是CU），以创建在编码树720中表示为列的节点列表。对于四叉树拆分，Z顺序扫描得到从左上方到右方之后是从左下方到右方的顺序。对于水平和垂直拆分，Z顺序扫描（遍历）分别简化为从上方到下方的扫描和从左方到右方的扫描。图7B的编码树720根据所应用的扫描顺序列出所有的节点和CU。各拆分在树的下一级别生成二、三或四个新节点的列表，直到到达叶节点（CU）为止。

[0142] 在如参考图3所述利用块分区器310将图像分解为CTU并进一步分解为CU、并且使用CU生成各残差块（324）的情况下，利用视频编码器114对残差块进行正向变换和量化。随后扫描如此得到的TB 336以形成残差系数的顺序列表，作为熵编码模块338的操作的一部分。在视频解码器134中进行等效处理以从位流133获得TB。

[0143] 图8A、8B和8C示出由编码树中的拆分得到的细分级别以及对将编码树单元分割成量化组的相应影响。针对每个量化组通过TB的残差用信号通知增量QP（392）至多一次。在HEVC中，量化组的定义对应于编码树深度，因为该定义得到固定大小的区域。在VVC中，附加拆分意味着编码树深度不再是CTU区域的合适代理。在VVC中，定义了“细分级别”，其中各个递增对应于所包含区域的一半。

[0144] 图8A示出编码树中的拆分的集合800和相应的细分级别。在编码树的根节点处，细分级别被初始化为零。当编码树包括四叉树拆分（例如810）时，细分级别针对其中包含的任何CU递增二。当编码树包括二元拆分（例如812）时，细分级别针对其中包含的任何CU递增1。当编码树包括三元拆分（例如814）时，细分级别针对外部两个CU递增2，并且针对由三元拆

分得到的内部CU递增1。当遍历各个CTU的编码树时,如参考图6所述,根据集合800确定各个所得CU的细分级别。

[0145] 图8B示出CU节点的示例集合840,并且示出拆分的效果。集合840的具有零细分级别的示例父节点820对应于图8B的示例中的大小为 64×64 的CTU。父节点820被三元拆分以产生大小分别为 16×64 、 32×64 和 16×64 的三个子节点821、822和823。子节点821、822和823分别具有细分级别2、1和2。

[0146] 在图8B的示例中,量化组阈值被设置为1,对应于 64×64 区域的一半,即对应于2048个样本的区域。标志跟踪新QG的开始。对于细分级别小于或等于量化组阈值的任何节点,重置跟踪新QG的标志。当遍历具有零细分级别的父节点820时,设置该标志。尽管大小为 32×64 的中心CU 822具有2048个样本的区域,但是两个兄弟CU 821和823具有细分级别2,即1024的区域,因此当遍历中心CU时不重置标志,并且量化组不在中心CU处开始。代替地,按照初始标志重置,标志如824所示在父节点处开始。有效地,QP可以仅在与量化组区域的倍数对齐的边界上改变。增量QP连同与CB相关联的TB的残差一起被用信号通知。如果不存在有效系数,则不存在编码增量QP的机会。

[0147] 图8C示出将CTU 862分割成多个CU和QG以图示细分级别、QG与增量QP的信令之间的关系示例860。垂直二元拆分将CTU 862分割成两半,左半部分870包含一个CU CU0,并且右半部分872包含若干CU (CU1-CU4)。在图8C的示例中,量化组阈值被设置为2,使得量化组通常具有等于CTU区域的四分之一的区域。由于父节点(即,编码树的根节点)的细分级别为零,因此QG标志被重置,并且新QG将从下一个编码CU(即,箭头868处的CU)开始。CU0(870)具有编码系数,因此增量QP 864与CU0的残差一起编码。右半部分872经过水平二元拆分并且在右半部分872的上部和下部中进一步拆分,从而得到CU1-CU4。与右半部分872的上部(包括CU1和CU2的877)和下部(包括CU3和CU4的878)部分相对应的编码树节点的细分级别为2。细分级别2等于量化组阈值2,因此新QG在各个部分开始,分别标记为874和876。CU1没有编码系数(没有残差),并且CU2是“跳过”CU,其也没有编码系数。因此,对于上部,不对增量QP进行编码。CU3是跳过CU,并且CU4具有编码残差,因此对于包括CU3和CU4的QG,利用CU4的残差对增量QP 866进行编码。

[0148] 图9A和9B示出 4×4 变换块扫描模式以及相关的主和二次变换系数。从视频编码器114角度来描述二次变换模块330对主残差系数的操作。根据后向对角线扫描模式910来扫描 4×4 TB 900。扫描模式910从“最后有效系数”位置朝向DC(左上)系数位置行进。未被扫描的所有系数位置,例如当考虑在正向方向上扫描时,位于最后有效系数位置之后的残差系数隐式地是非有效的。当使用二次变换时,所有剩余系数都是非有效的。也就是说,不进行二次变换的所有二次域残差系数都是非有效的,并且未通过二次变换的应用填充的所有主域残差系数都需要是非有效的。此外,在由模块330应用正向二次变换之后,可以存在比由二次变换模块330处理的主变换系数的数量更少的二次变换系数。例如,图9B示出了块的集合920。在图9B中,十六(16)个主系数被布置为一个 4×4 子块,即 4×4 TB 920的924。在图9B的示例中,可以对主残差系数进行二次变换以产生二次变换块926。二次变换块926包含八个二次变换系数928。八个二次变换系数928根据扫描模式910存储在TB中,从DC系数位置向前打包。 4×4 子块的剩余系数位置(被示出为区域930)包含来自主变换的量化残差系数,并且需要对于要应用的二次变换都是非有效的。因此, 4×4 TB的指定作为TB 920的前八个

扫描位置之一的系数的最后有效系数位置指示 (i) 应用二次变换, 或 (ii) 在量化之后, 主变换的输出不具有超出TB 920的第八扫描位置的有效系数。

[0149] 当可以对TB进行二次变换时, 二次变换索引(即388)被编码以指示二次变换的可能应用。二次变换索引还可以指示: 在多个变换核可用的情况下, 哪个核将在模块330处被应用为二次变换。相应地, 当最后有效系数位置位于保留用于保持二次变换系数(例如, 928)的任何一个扫描位置时, 视频解码器134对二次变换索引470进行解码。

[0150] 尽管已经描述了将16个主系数映射到八个二次系数的二次变换核, 但是不同的核是可能的, 包括映射到不同数量的二次变换系数的核。二次变换系数的数量可以与主变换系数的数量相同, 例如16。对于宽度为4且高度大于4的TB, 关于 4×4 TB情况描述的行为适用于TB的顶部子块。当应用二次变换时, TB的其它子块具有零值残差系数。对于宽度大于4且高度等于4的TB, 关于 4×4 TB情况描述的行为适用于TB的最左侧子块, 并且TB的其它子块具有零值残差系数, 从而允许使用最后有效系数位置来确定是否需要解码二次变换索引。

[0151] 图9C和9D示出 8×8 变换块扫描模式和示例相关的主和二次变换系数。图9C示出用于 8×8 TB 940的基于 4×4 子块的后向对角线扫描模式950。在基于 4×4 子块的后向对角线扫描模式950中扫描 8×8 TB 940。图9D示出集合960, 该集合960示出二次变换的操作效果。扫描950从最后有效系数位置返回到DC(左上)系数位置。当剩余的16个主系数(示出为964)为零值时, 将正向二次变换核应用于48个主系数(示出为940的区域962)是可能的。将二次变换应用于区域962得到示出为966的16个二次变换系数。TB的其它系数位置是零值, 标记为968。如果 8×8 TB 940的最后有效位置指示二次变换系数在966内, 则对二次变换索引388进行编码以指示模块330应用特定变换核(或绕过核)。视频解码器134使用TB的最后有效位置来确定是否解码二次变换索引, 即索引470。对于宽度或高度超过八个样本的变换块, 图9C和9D的方法应用于左上 8×8 区域, 即TB的左上 2×2 子块。

[0152] 如图9A至图9D中所描述, 两个大小的二次变换核是可用的。一个大小的二次变换核用于宽度或高度为4的变换块, 并且另一个大小的二次变换核用于宽度和高度大于4的变换块。在各个大小的核内, 二次变换核的多个集合(例如, 四个)是可用的。基于块的帧内预测模式选择一个集合, 该一个集合可在亮度块与色度块之间不同。在所选择的集合内, 一个或两个核可用。独立于属于编码树单元的共用树的编码单元中的亮度块和色度块, 经由二次变换索引用信号通知使用所选集合内的一个核或绕过二次变换。换句话说, 用于亮度通道的索引及用于色度通道的索引彼此独立。

[0153] 图10示出在通用视频编码(VVC)标准中可用的变换块的集合1000。图10还示出将二次变换应用于来自集合1000的变换块的残差系数的子集。图10示出宽度和高度为从4到32范围内的TB。然而, 宽度和/或高度为64的TB是可能的, 但是为了便于参考而未示出。

[0154] 将16点二次变换1052(用较暗的阴影示出)应用于 4×4 系数集合。16点二次变换1052应用于宽度或高度为4的TB, 例如, 4×4 TB 1010、 8×4 TB 1012、 16×4 TB 1014、 32×4 TB 1016、 4×8 TB 1020、 4×16 TB 1030和 4×32 TB 1040。如果64点主变换可用, 则将16点二次变换1052应用于大小为 4×64 和 64×4 的TB(图10中未示出)。对于宽度或高度为四但具有多于16个主系数的TB, 仅将16点二次变换应用于TB的左上 4×4 子块, 并且需要其它子块具有零值系数以应用二次变换。通常, 应用16点二次变换得到16个二次变换系数, 其被打包到TB中以编码在获得原始16个主变换系数的子块中。例如如参考图9B所述, 二次变换核可以使得

创建比应用二次变换的主变换系数的数量更少的二次变换系数。

[0155] 对于宽度和高度大于四的变换大小,如图10所示,48点二次变换1050(用较浅的阴影示出)可用于对变换块的左上 8×8 区域中的残差系数的三个 4×4 子块的应用。在每种情况下在用浅色阴影和虚线轮廓示出的区域中,48点二次变换1050应用于 8×8 变换块1022、 16×8 变换块1024、 32×8 变换块1026、 8×16 变换块1032、 16×16 变换块1034、 32×16 变换块1036、 8×32 变换块1042、 16×32 变换块1044和 32×32 变换块1046。如果64点主变换可用,则48点二次变换1050也适用于大小为 8×64 、 16×64 、 32×64 、 64×64 、 64×32 、 64×16 和 64×8 的TB(未示出)。48点二次变换核的应用通常导致产生少于48个二次变换系数。例如,可以产生8或16个二次变换系数。二次变换系数存储在左上区域中的变换块中,例如,图9D中示出了八个二次变换系数。未经受二次变换的主变换系数(“仅主系数”) (例如TB 1034的系数1066(类似于图9D的964))需要为零值,以应用二次变换。在正向方向上应用48点二次变换1050之后,可以包含有效系数的区域从48个系数减少到16个系数,从而进一步减少了可能包含有效系数的系数位置的数量。例如,968将仅包含非有效系数。对于逆二次变换,例如仅在TB的966中存在的经解码的有效系数被变换以产生任何一个在区域(例如962)中可能是有效的系数,然后对这些系数进行主逆变换。当二次变换将一个或多个子块减少为16个二次变换系数的集合时,仅左上 4×4 子块可以包含有效系数。位于可以存储二次变换系数的任何系数位置处的最后有效系数位置指示应用二次变换或仅应用主变换。然而,在量化之后,所得到的有效系数处于相同的区域中,就好像已经应用了二次变换核那样。

[0156] 当最后有效系数位置指示TB(例如922或962)中的二次变换系数位置时,需要用信号通知的二次变换索引来在应用二次变换核或绕过二次变换之间进行区分。尽管已经从视频编码器114的角度描述了将二次变换应用于图10中的各种大小的TB,但是在视频解码器134中进行相应的逆处理。视频解码器134首先解码最后有效系数位置。如果经解码的最后有效系数位置指示二次变换的潜在应用,即该位置在分别产生8或16个二次变换系数的二次变换核的928或966内,则解码二次变换索引以确定是应用还是绕过逆二次变换。

[0157] 图11示出具有多个条带的位流1101的句法结构1100。条带中的各条带包含多个编码单元。位流1101可以由视频编码器114产生为例如位流115,或者可以由视频解码器134解析为例如位流133。位流1101被分割成多个部分,例如网络抽象层(NAL)单元,其中通过在各个NAL单元之前设置NAL单元头部(诸如1108等)来实现描绘。序列参数集(SPS)1110定义序列级参数,诸如用于编码和解码位流的简档(工具集)、色度格式、样本位深和帧分辨率等。集1110中还包括约束各个CTU的编码树中不同类型的拆分的应用的参数。例如,使用用于块大小约束的 \log_2 基础并相对于其它参数(诸如最小CTU大小等)表达参数,可以优化约束拆分类型的参数的编码以用于更紧凑的表示。在SPS 1110中编码的若干参数如下:

[0158] • `log2_ctu_size_minus5`:指定CTU大小,其中经编码值0、1及2分别指定 32×32 、 64×64 及 128×128 的CTU大小。

[0159] • `partition_constraints_override_enabled_flag`:使得能够应用若干参数的条带级覆写,统称为分区约束参数1130。

[0160] • `log2_min_luma_coding_block_size_minus2`:指定(亮度样本中的)最小编码块大小,其中值0、1、2、...指定 4×4 、 8×8 、 16×16 、...的最小亮度CB大小。最大编码值由指定的CTU大小约束,即,使得`log2_min_luma_coding_block_size_minus2` \leq `log2_ctu_size`

minus5+3。可用色度块尺寸对应于可用亮度块尺寸,根据使用中的色度格式的色度通道子采样来缩放。

[0161] • `sps_max_mtt_hierarchy_depth_inter_slice`:指定多树型拆分(即,二元和三元拆分)的编码树中的编码单元相对于帧间(P或B)条带的编码树中的二叉树节点(即,一旦二叉树拆分在编码树中停止)的最大层级深度,并且是参数1130之一。

[0162] • `sps_max_mtt_hierarchy_depth_intra_slice_luma`:指定多树型拆分(即,二元和三元)的编码树中的编码单元相对于帧内(I)条带的编码树中的二叉树节点(即,一旦二叉树拆分在编码树中停止)的最大层级深度,并且是参数1130之一。

[0163] • `partition_constraints_override_flag`:当SPS中的`partition_constraints_override_enabled_flag`等于1时,在条带头部中用信号通知该参数,并且该参数指示在SPS中用信号通知的分区约束将针对相应条带被覆写。

[0164] 图片参数集(PPS)1112定义适用于零个或多个帧的参数集合。包括在PPS 1112中的参数包括用于将帧分割成一个或多个“区块”和/或“块”的参数。PPS1112的参数还可以包括CU色度QP偏移的列表,其中之一可以在CU级应用以从并置亮度CB的量化参数导出供色度块使用的量化参数。

[0165] 形成一个图片的条带序列被称为访问单元(AU),诸如AU 0 1114等。AU 0 1114包括三个条带,诸如条带0至2等。条带1被标记为1116。与其它条带一样,条带1(1116)包括条带头部1118和条带数据1120。

[0166] 条带头部包括分组为1134的参数。组1134包括:

[0167] • `slice_max_mtt_hierarchy_depth_luma`:当条带头部中的`partition_constraints_override_flag`等于1且覆写从SPS导出的值时,在条带头部1118中用信号通知。对于I条带,代替使用`sps_max_mtt_hierarchy_depth_intra_slice_luma`来设置1134处的MaxMttDepth,而是使用`slice_max_mtt_hierarchy_depth_luma`。对于P或B条带,代替使用`sps_max_mtt_hierarchy_depth_inter_slice`,而是使用`slice_max_mtt_hierarchy_depth_luma`。

[0168] 从解码自SPS1110的句法元素`sps_log2_diff_min_qt_min_cb_intra_slice_luma`导出变量MinQtLog2SizeIntraY(未示出),其指定由I条带的零个或多个二叉树拆分(即,在编码树中没有发生进一步的MTT拆分)得到的最小编码块大小。从解码自SPS1110的句法元素`sps_log2_diff_min_qt_min_cb_inter_slice`导出变量MinQtLog2SizeInterY(未示出)。变量MinQtLog2SizeInterY指定由P和B条带的零个或多个二叉树拆分(即,在编码树中没有发生进一步的MTT拆分)得到的最小编码块大小。由于由二叉树拆分得到的CU是正方形,因此变量MinQtLog2SizeIntraY和MinQtLog2SizeInterY各自指定宽度和高度两者(作为CU宽度/高度的log2)。

[0169] 参数`cu_qp_delta_subdiv`可以可选地在条带头部1118中用信号通知,并且指示在用于共用树或单独树条带中的亮度分支的编码树中用信号通知增量QP的最大细分级别。对于I条带,`cu_qp_delta_subdiv`的范围是0到 $2 * (\log_2 \text{ctu_size_minus5} + 5 - \text{MinQtLog2SizeIntraY} + \text{MaxMttDepthY} \ 1134)$ 。对于P或B条带,`cu_qp_delta_subdiv`的范围是0到 $2 * (\log_2 \text{ctu_size_minus5} + 5 - \text{MinQtLog2SizeInterY} + \text{MaxMttDepthY} \ 1134)$ 。由于`cu_qp_delta_subdiv`的范围取决于自从SPS1110或条带头部1118获得的分区约束导出的值

MaxMttDepthY 1134,因此不存在解析问题。

[0170] 参数cu_chroma_qp_offset_subdiv可以可选地在条带头部1118中用信号通知,并且指示在共用树中或在单独的树条带中的色度分支中用信号通知色度CU QP偏移的最大细分级别。对I或P/B条带的cu_chroma_qp_offset_subdiv的范围约束与对cu_qp_delta_subdiv的相应范围约束相同。

[0171] 针对条带1120中的CTU导出细分级别1136,其指定用于亮度CB的cu_qp_delta_subdiv和用于色度CB的cu_chroma_qp_offset_subdiv。如参考图8A-C所描述,细分级别用于确定在CTU中的哪些点处对增量QP句法元素进行编码。对于色度CB,也使用图8A-C的方法来用信号通知色度CU级偏移启用(以及索引,如果启用的话)。

[0172] 图12示出位流1101(例如115或133)的条带数据1120的句法结构1200,其具有用于编码树单元(诸如CTU 1210等)的亮度和色度编码块的共用树。CTU 1210包括一个或多个CU,示例示出为CU 1214。CU 1214包括用信号通知的预测模式1216a,接着是变换树1216b。当CU 1214的大小不超过最大变换大小(32×32或64×64)时,变换树1216b包括一个变换单元,示出为TU 1218。

[0173] 如果预测模式1216a指示针对CU 1214使用帧内预测,则指定亮度帧内预测模式和色度帧内预测模式。对于CU 1214的亮度CB,主变换类型也用信号通知为(i)水平和垂直DCT-2、(ii)水平和垂直变换跳过或(iii)水平和垂直DST-7和DCT-8的组合。如果用信号通知的亮度变换类型为水平和垂直DCT-2(选项(i)),则在如参考图9A-D所描述的条件下,在位流中用信号通知附加亮度二次变换类型1220,也称为“低频不可分离变换”(LFNST)索引。色度二次变换类型1221也用信号通知。色度二次变换类型1221是独立于亮度主变换类型是否为DCT-2而用信号通知的。

[0174] 共用编码树的使用得到包括被示出为亮度TB Y 1222、第一色度TB Cb1224和第二色度TB Cr 1226的用于各个颜色通道的TB的TU 1218。发送单个色度TB以指定Cb和Cr通道两者的色度残差的编码模式是可用的,称为“联合CbCr”编码模式。当启用联合CbCr编码模式时,对单个色度TB进行编码。

[0175] 不管颜色通道如何,各个TB包括最后位置1228。最后位置1228指示当考虑对角线扫描模式(其用于在前向方向上(即,从DC系数向前)串行化TB的系数阵列)中的系数时TB中的最后有效残差系数位置。如果TB的最后位置1228指示仅二次变换域中的系数(即将仅经受主变换的所有剩余系数)是有效的,则用信号通知二次变换索引以指定是否应用二次变换。

[0176] 如果要应用二次变换并且如果多于一个二次变换核可用,则二次变换索引指示选择哪个核。通常,在“候选集合”中,一个核可用,或者两个核可用。候选集合是从块的帧内预测模式确定的。通常,存在四个候选集合,但是可以存在更少的候选集合。如上所述,针对亮度和色度使用二次变换,并且因此所选择的核分别取决于用于亮度和色度通道的帧内预测模式。核还可以取决于相应的亮度和色度TB的块大小。针对色度选择的核还取决于位流的色度子采样率。如果仅一个核可用,则用信号通知被限制为应用或不应用二次变换(索引范围0至1)。如果两个核可用,则索引值为0(不应用)、1(应用第一核)或2(应用第二核)。对于色度,将相同的二次变换核应用于各个色度通道,因此Cb块1224和Cr块1226的残差仅需要包括经受二次变换的位置中的有效系数,如参考图9A-D所述。如果使用联合CbCr编码,则仅

包括经受二次变换的位置中的有效系数的要求仅适用于单个编码色度TB,因为所得的Cb和Cr残差仅在与联合编码TB中的有效系数相对应的位置中包含有效系数。如果给定二次索引的(一个或多个)适用颜色通道由单个TB(单个最后位置,例如1228)描述,即当使用联合CbCr编码时亮度总是仅需要一个TB并且色度需要一个TB,则可以在编码最后位置之后紧接着编码二次变换索引而不是在TU之后编码二次变换索引,即作为索引1230而不是1220(或1221)。在位流中较早地用信号通知二次变换允许视频解码器134在残差系数1232中的各个残差系数被解码时开始应用二次变换,从而减少系统100中的延迟。

[0177] 在视频编码器114和视频解码器134的布置中,当不使用联合CbCr编码时,针对各个色度TB(即1224和1226)用信号通知单独的二次变换索引,从而得到对各个颜色通道的二次变换的独立控制。如果独立地控制各个TB,则可以在亮度和色度的相应TB的最后位置之后紧接着用信号通知各个TB的二次变换索引(无论是否应用联合CbCr模式)。

[0178] 图13示出用于将帧数据113编码在位流115中的方法1300,位流115包括作为编码树单元序列的一个或多个条带。方法1300可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1300可以在处理器205的执行下由视频编码器114进行。由于对帧进行编码的工作负荷,方法1300的步骤可以在不同的处理器中进行以例如使用当代多核处理器分担工作负荷,使得不同的条带由不同的处理器编码。此外,在编码位流115的各个部分(条带)时,分区约束和量化组定义可以从一个条带到另一条带变化,这被认为对于速率控制目的是有益的。为了对各个编码单元的残差进行编码的附加灵活性,不仅量化组细分级别可以从一个条带到另一条带变化,而且二次变换的应用对于亮度和色度是独立可控的。因此,方法1300可以存储在计算机可读存储介质上和/或存储器206中。

[0179] 方法1300开始于编码SPS/PPS步骤1310。在步骤1310处,视频编码器114将SPS1110和PPS1112作为固定和可变长度编码参数序列编码在位流115中。partition_constraints_override_enabled_flag被编码为SPS1110的一部分,指示分区约束能够在相应条带(诸如1116等)的条带头部(1118)中被覆写。默认分区约束也由视频编码器114编码为SPS1110的一部分。

[0180] 方法1300从步骤1310继续到将帧分割成条带的步骤1320。在步骤1320的执行中,处理器205将帧数据113分割成一个或多个条带或连续部分。在需要并行性的情况下,视频编码器114的单独实例在某种程度上独立地对各个条带进行编码。单个视频编码器114可以顺序地处理各个条带,或者可以实现一些中间程度的并行性。通常,将帧分割成条带(连续部分)与将帧分割成称为“子图片”或区块等的区域的边界对准。

[0181] 方法1300从步骤1320继续到编码条带头部的步骤1330。在步骤1330处,熵编码器338将条带头部1118编码在位流115中。以下参考图14提供步骤1330的示例实现。

[0182] 方法1300从步骤1330继续到将条带分割成CTU的步骤1340。在步骤1340的执行中,视频编码器114将条带1116分割成CTU序列。条带边界与CTU边界对准,并且条带中的CTU根据CTU扫描顺序(通常为光栅扫描顺序)排序。将条带分割成CTU确定了帧数据113的哪一部分将由视频编码器113在编码当前条带时处理。

[0183] 方法1300从步骤1340继续到确定编码树的步骤1350。在步骤1350处,视频编码器114确定条带中当前选择的CTU的编码树。方法1300在步骤1350的第一次调用时从条带1116中的第一CTU开始,并且在后续调用时前进到条带1116中的后续CTU。在确定CTU的编码树

时,由块分区器310生成并测试四叉树、二元和三元拆分的各种组合。

[0184] 方法1300从步骤1350继续到确定编码单元的步骤1360。在步骤1360处,视频编码器114执行以使用已知方法确定从评估中的各种编码树得到的CU的“最佳”编码。确定最佳编码涉及确定预测模式(例如,具有特定模式的帧内预测或具有运动矢量的帧间预测)、变换选择(主变换类型和可选的二次变换类型)。如果亮度TB的主变换类型被确定为DCT-2或者不经受正向二次变换的任何量化的主变换系数是有效的,则亮度TB的二次变换索引可以指示二次变换的应用。否则,亮度的二次变换索引指示绕过二次变换。对于亮度通道,主变换类型被确定为色度通道的MTS选项之一、DCT-2或变换跳过,DCT-2是可用的变换类型。参考图19A和19B进一步描述二次变换类型的确定。确定编码还可以包括确定可以改变QP的量化参数,即在量化组边界处的量化参数。在确定各个编码单元时,还以联合方式确定最佳编码树。当使用帧内预测对编码单元进行编码时,确定亮度帧内预测模式和色度帧内预测。

[0185] 当在由DCT-2主变换的应用得到的主域残差中不存在“AC”(变换块的左上位置之外的位置中的系数)残差系数时,确定编码单元的步骤1360可以禁止测试二次变换的应用。如果针对仅包括DC系数的变换块测试二次变换应用(最后位置指示仅变换块的左上系数是有效的),则看到编码效率增加。仅存在DC主系数时测试二次变换的禁止跨越了二次变换索引适用的块,即,对单个索引进行编码时的共用树的Y、Cb和Cr(当Cb和Cr块是两个样本的宽度或高度时,仅Y通道)。尽管与具有至少一个AC系数的残差相比,仅具有DC系数的残差在编码成本上是低的,但对仅具有有效DC系数的残差应用二次变换也导致最终编码DC系数的幅度进一步减小。即使在编码之前的进一步量化和/或舍入操作之后,其它(AC)系数在二次变换之后的幅度不足以得到位流中的一个或多个有效编码残差系数。在共用或单独树编码树中,假设存在至少一个有效主系数,即使在二次变换索引的应用范围内仅存在相应变换块的一个或多个DC系数,视频编码器114也测试非零二次变换索引值的选择(即,二次变换的应用)。

[0186] 方法1300从步骤1360继续到对编码单元进行编码的步骤1370。在步骤1370处,视频编码器114将步骤1360的所确定的编码单元编码在位流115中。参考图15更详细地描述如何对编码单元进行编码的示例。

[0187] 方法1300从步骤1370继续到最后编码单元测试的步骤1380。在步骤1380处,处理器205测试当前编码单元是否是CTU中的最后编码单元。如果不是(步骤1380为“否”),则处理器205中的控制前进到确定编码单元的步骤1360。否则,如果当前编码单元是最后编码单元(步骤1380为“是”),则处理器205中的控制前进到最后CTU测试的步骤1390。

[0188] 在最后CTU测试的步骤1390处,处理器205测试当前CTU是否是条带1116中的最后CTU。如果不是条带1116中的最后CTU,则处理器205中的控制返回到确定编码树的步骤1350。否则,如果当前CTU是最后的(步骤1390为“是”),则处理器中的控制前进到最后条带测试的步骤13100。

[0189] 在最后条带测试的步骤13100处,处理器205测试正在编码的当前条带是否是帧中的最后条带。如果不是最后条带(步骤13100为“否”),则处理器205中的控制前进到编码条带头部的步骤1330。否则,如果当前条带是最后的并且所有条带(连续部分)已经被编码(步骤13100为“是”),则方法1300终止。

[0190] 图14示出如在步骤1330实现的用于将条带头部1118编码在位流115中的方法

1400。方法1400可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1400可以在处理器205的执行下由视频编码器114进行。因此,方法1400可以存储在计算机可读存储介质上和/或存储器206中。

[0191] 方法1400在分区约束覆写启用测试的步骤1410处开始。在步骤1410处,处理器205测试如在SPS1110中编码的分区约束覆写启用标志是否指示分区约束可以在条带级被覆写。如果分区约束可以在条带级被覆写(步骤1410为“是”),则处理器205中的控制前进到确定分区约束的步骤1420。否则,如果分区约束不能在条带级被覆写(步骤1410处为“否”),则处理器205中的控制前进到编码其它参数的步骤1480。

[0192] 在确定分区约束的步骤1420处,处理器205确定适合于当前条带1116的分区约束(例如,最大MTT拆分深度)。在一个示例中,帧数据310包含映射到2D框中且分割成若干子图片的场景的360度视图的投影。取决于所选视口,某些条带可能需要较高保真度且其它条带可能需要较低保真度。可以基于帧数据310的由给定条带编码的部分的保真度要求来设置该条带的分区约束(例如,按照步骤1340)。在较低保真度被认为是可接受的情况下,具有较大CU的较浅编码树是可接受的,因此最大MTT深度可以被设置为较低值。相应地至少在由所确定的最大MTT深度1134产生的范围中确定以标志`cu_qp_delta_subdiv`用信号通知的细分级别1136。还确定并用信号通知相应的色度细分级别。

[0193] 方法1400从步骤1420继续到编码分区约束覆写标志的步骤1430。在步骤1430处,熵编码器338将标志编码在位流115中,该标志指示如在SPS1110中用信号通知的分区约束是否将针对条带1116被覆写。如果在步骤1420处导出了特定于当前条带的分区约束,则标志值将指示分区约束覆写功能的使用。如果在步骤1420处确定的约束与已经在SPS1110中编码的约束匹配,则不需要覆写分区约束,因为没有要用信号通知的改变,并且相应地编码标志值。

[0194] 方法1400从步骤1430继续到分区约束覆写测试的步骤1440。在步骤1440处,处理器205测试在步骤1430处编码的标志值。如果标志指示分区约束将被覆写(步骤1440为“是”),则处理器205中的控制前进到编码条带分区约束的步骤1450。否则,如果分区约束不被覆写(步骤1440为“否”),则处理器205中的控制前进到编码其它参数的步骤1480。

[0195] 方法1400从步骤1440继续到编码条带分区约束的步骤1450。在步骤1450的执行中,熵编码器338将条带的所确定分区约束编码在位流115中。条带的分区约束包括“`slice_max_mtt_hierarchy_depth_luma`”,从其导出`MaxMttDepthY` 1134。

[0196] 方法1400从步骤1450继续到编码QP细分级别的步骤1460。在步骤1460处,熵编码器338使用“`cu_qp_delta_subdiv`”句法元素对亮度CB的细分级别进行编码,如参考图11所述。

[0197] 方法1400从步骤1460继续到编码色度QP细分级别的步骤1470。在步骤1470处,熵编码器338使用“`cu_chroma_qp_offset_subdiv`”句法元素对用于CU色度QP偏移的用信号通知的细分级别进行编码,如参考图11所述。

[0198] 步骤1460和1470进行操作以对帧的条带(连续部分)的整体QP细分级别进行编码。整体细分级别包括用于条带的亮度编码单元的细分级别和用于条带的色度编码单元的细分级别两者。例如由于对I条带中的亮度和色度使用单独的编码树,因此色度和亮度细分级别可以是不同的。

[0199] 方法1400从步骤1470继续到编码其它参数的步骤1480。在步骤1480处,熵编码器338将其它参数编码在条带头部1118中,诸如控制如去块、自适应环路滤波、从先前用信号通知的缩放列表中可选地选择缩放列表(用于将量化参数非均匀地应用于变换块)等的特定工具所需的参数等。方法1400在执行步骤1480时终止。

[0200] 图15示出用于将编码单元编码在位流115中的方法1500,其对应于图13的步骤1370。方法1500可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1500可以在处理器205的执行下由视频编码器114进行。因此,方法1500可以存储在计算机可读存储介质上和/或存储器206中。

[0201] 方法1500在编码预测模式的步骤1510处开始。在步骤1510处,熵编码器338将在步骤1360处确定的针对编码单元的预测模式编码在位流115中。“pred_mode”句法元素被编码以区分用于编码单元的帧内预测、帧间预测或其它预测模式的使用。如果帧内预测用于编码单元,则对亮度帧内预测模式进行编码,并且对色度帧内预测模式进行编码。如果帧间预测用于编码单元,则可以对“合并索引”进行编码以从相邻编码单元选择运动矢量以供该编码单元使用,可以对运动矢量增量进行编码以将偏移引入到从空间相邻块导出的运动矢量。对主变换类型进行编码以针对编码单元的亮度TB在水平和垂直使用DCT-2、水平和垂直使用变换跳过、或者水平和垂直使用DCT-8和DST-7的组合之间进行选择。

[0202] 方法1500从步骤1510继续到编码残差测试的步骤1520。在步骤1520处,处理器205确定是否需要针对编码单元对残差进行编码。如果对于编码单元存在要编码的任何有效残差系数(步骤1520为“是”),则处理器205中的控制前进到新QG测试步骤1530。否则,如果不存在用于编码的有效残差系数(步骤1520为“否”),则方法1500终止,因为对编码单元进行解码所需的所有信息都存在于位流115中。

[0203] 在新QG测试步骤1530处,处理器205确定编码单元是否对应于新的量化组。如果编码单元对应于新的量化组(步骤1530为“是”),则处理器205中的控制进行到编码增量QP的步骤1540。否则,如果编码单元与新的量化组不相关(步骤1530为“否”),则处理器205中的控制前进到进行主变换的步骤1550。在对各个编码单元进行编码时,在步骤1530处遍历CTU的编码树的节点。当根据“cu_qp_delta_subdiv”所确定的,当前节点的任何子节点具有小于或等于当前条带的细分级别1136的细分级别时,新的量化组在与该节点相对应的CTU区域中开始,并且步骤1530返回“是”。量化组中包括编码残差的第一CU还将包括编码增量QP,从而用信号通知适用于该量化组中的残差系数的量化参数的任何变化。

[0204] 在编码增量QP的步骤1540处,熵编码器338将增量QP编码在位流115中。增量QP对预测QP与在当前量化组中使用的预期QP之间的差进行编码。通过对相邻的较早(上方和左侧)量化组的QP求平均来导出预测QP。当细分级别较低时,量化组较大,并且较不频繁地对增量QP进行编码。增量QP的较不频繁的编码得到用于用信号通知QP中的变化的较低开销,但也导致速率控制中的较低灵活性。各个量化组的量化参数的选择由QP控制器模块390进行,QP控制器模块390通常实现速率控制算法以针对位流115的特定速率,这在某种程度上与底层帧数据113的统计量的变化无关。方法1500从步骤1540继续到进行主变换的步骤1550。

[0205] 在进行主变换的步骤1550处,正向主变换模块326根据编码单元的主变换类型进行主变换,从而得到主变换系数328。对各个颜色通道进行主变换,首先对亮度通道(Y)进行

主变换,然后在针对当前TU的步骤1550的后续调用时对Cb TB和Cr TB进行主变换。对于亮度通道,进行主变换类型(DCT-2、变换跳过、MTS选项),并且对于色度通道,进行DCT-2。

[0206] 方法1500从步骤1550继续到量化主变换系数的步骤1560。在步骤1560处,量化器模块334根据量化参数392对主变换系数328进行量化,以产生量化的主变换系数332。使用增量QP(当存在时)来编码变换系数328。

[0207] 方法1500从步骤1560继续到进行二次变换的步骤1570。在步骤1570处,二次变换模块330根据当前变换块的二次变换索引388对量化的主变换系数332进行二次变换,以产生二次变换系数336。尽管在量化之后进行二次变换,但是与量化参数392的最终预期量化器步长相比,主变换系数328可以保持更高的精度,例如幅度可以是直接由量化参数392的应用产生的幅度的16倍,即,将保留四个附加位的精度。在量化的主变换系数332中保留附加位的精度允许二次变换模块330对主系数域中的系数以更高的精度进行操作。在应用二次变换之后,步骤1560处的最终缩放(例如,右移四位)使得量化到量化参数392的预期量化器步长。对主变换系数(其对应于众所周知的变换基函数(DCT-2、DCT-8、DST-7))进行“缩放列表”的应用,而不是对由经训练的二次变换核产生的二次变换系数进行操作。当变换块的二次变换索引388指示不应用二次变换(索引值等于零)时,绕过二次变换。也就是说,主变换系数332不变地传播通过二次变换模块330以成为二次变换系数336。亮度二次变换索引与亮度帧内预测模式结合使用,以选择二次变换核来应用于亮度TB。色度二次变换索引与色度帧内预测模式结合使用,以选择二次变换核来应用于色度TB。

[0208] 方法1500从步骤1570继续到编码最后位置的步骤1580。在步骤1580处,熵编码器338将当前变换块的二次变换系数336中的最后有效系数的位置编码在位流115中。在第一次调用步骤1580时,考虑亮度TB,并且后续调用考虑Cb TB然后考虑Cr TB。

[0209] 在二次变换索引388紧接在最后位置之后被编码的布置中,方法1500继续到编码LFNST索引的步骤1590。如果基于在步骤1580处编码的最后位置未推断出二次变换索引为零,则在步骤1590处,熵编码器338使用截断一元码字将二次变换索引338作为“lfnst_index”编码在位流115中。各个CU具有一个亮度TB,从而允许对亮度块进行步骤1590,并且当“联合”编码模式用于色度时,对单个色度TB进行编码,因此可以对色度进行步骤1590。在解码各个残差系数之前知道二次变换索引,这使得能够在系数被解码时例如使用乘加逻辑来逐个系数地应用二次变换。方法1500从步骤1590继续到编码子块的步骤15100。

[0210] 如果二次变换索引388在最后位置之后没有紧接着被编码,则方法1500从步骤1580继续到编码子块的步骤15100。在编码子块的步骤15100处,当前变换块的残差系数(336)作为一系列子块被编码在位流115中。从包含最后有效系数位置的子块前进回到包含DC残差系数的子块,残差系数被编码。

[0211] 方法1500从步骤15100继续到最后TB测试的步骤15110。在该步骤处,处理器205测试当前变换块是否是颜色通道(即Y、Cb和Cr)上的进展中的最后变换块。如果刚刚编码的变换块用于Cr TB(步骤15110为“是”),则处理器205中的控制前进到编码亮度LFNST索引的步骤15120。否则,如果当前TB不是最后的(15110处为“否”),则处理器205中的控制返回到进行主变换的步骤1550和下一TB(选择Cb或Cr)。

[0212] 关于预测模式是帧内预测并使用DCT-2的共用编码树结构的示例来描述步骤1550至15110。除了使用已知方法的共用编码树结构之外,可以针对帧间预测模式或帧内预测模

式来实现诸如进行主变换(1550)、量化主变换系数(1560)和编码最后位置(1590)等的步骤的操作。无论预测模式或编码树结构如何,都可以实现步骤1510至1540。

[0213] 方法1500从步骤15110继续到编码亮度LFNST索引的步骤15120。在步骤15120处,如果应用于亮度TB的二次变换索引未被推断为零(不应用二次变换),则熵编码器338将其编码在位流115中。如果亮度TB的最后有效位置指示有效的仅主残差系数或者如果进行除DCT-2之外的主变换,则亮度二次变换索引被推断为零。另外,仅对于使用帧内预测和共用编码树结构的编码单元,将应用于亮度TB的二次变换索引编码在位流中。使用标志1220(或用于联合CbCr模式的标志1230)对应用于亮度TB的二次变换索引进行编码。

[0214] 方法1500从步骤15120继续到编码色度LFNST索引的步骤15130。在步骤1530处,如果应用于色度TB的二次变换索引未被推断为零(不应用二次变换),则由熵编码器338将色度二次变换索引编码在位流115中。如果任一色度TB的最后有效位置指示有效的仅主残差系数,则色度二次变换索引被推断为零。方法1500在执行步骤15130后终止,其中处理器205中的控制返回到方法1300。仅针对使用帧内预测和共用编码树结构的编码单元将应用于色度TB的二次变换索引编码在位流中。使用标志1221(或用于联合CbCr模式的标志1230)对应用于色度TB的二次变换索引进行编码。

[0215] 图16示出从作为布置成条带的编码单元序列的位流中解码帧的方法1600。方法1600可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1600可以在处理器205的执行下由视频解码器134进行。因此,方法1600可以存储在计算机可读存储介质上和/或存储器206中。

[0216] 方法1600对使用方法1300编码的位流进行解码,在方法1300中,分区约束和量化组定义可以从一个条带到另一条带变化,这被认为在编码位流115的各个部分(条带)时对于速率控制目的是有益的。不仅量化组细分级别可以从一个条带到另一条带变化,而且二次变换的应用对于亮度和色度是独立可控的。

[0217] 方法1600开始于解码SPS/PPS步骤1610。在步骤1610的执行中,视频解码器134从位流133将SPS1110和PPS1112解码为固定和可变长度参数的序列。partition_constraints_override_enabled_flag被解码为SPS1110的一部分,指示分区约束是否能够在相应条带(例如,1116)的条带头部(例如,1118)中被覆写。默认(即,如在SPS1110中用信号通知并且在没有后续覆写的情况下在条带中使用的)分区约束参数1130也由视频解码器134解码为SPS1110的一部分。

[0218] 方法1600从步骤1610继续到确定条带边界的步骤1620。在步骤1620的执行中,处理器205确定位流133中的当前访问单元中的条带的位置。通常,通过确定NAL单元边界(通过检测“起始码”)并且对于各个NAL单元读取包括“NAL单元类型”的NAL单元头部来识别条带。特定NAL单元类型标识条带类型,诸如“I条带”、“P条带”和“B条带”等。在识别出条带边界之后,应用233可以例如在多处理器架构中将方法1600的后续步骤的进行分布在不同的处理器上,以用于并行解码。多处理器系统中的各个处理器可以解码不同的条带以获得更高的解码吞吐量。

[0219] 方法1600从步骤1610继续到解码条带头部的步骤1630。在步骤1630处,熵解码器420从位流133解码条带头部1118。下面参考图17描述如在步骤1630处实现的从位流133解码条带头部1118的示例方法。

[0220] 方法1600从步骤1630继续到将条带分割成CTU的步骤1640。在步骤1640处,视频解码器134将条带1116分割成CTU序列。条带边界与CTU边界对准,并且条带中的CTU根据CTU扫描顺序排序。CTU扫描顺序通常是光栅扫描顺序。将条带分割成CTU确定了帧数据113的哪个部分将由视频解码器134在解码当前条带时处理。

[0221] 方法1600从步骤1640继续到解码编码树的步骤1650。在步骤1650的执行中,视频解码器133在第一次调用步骤1650时从条带1116中的第一CTU开始,从位流133中解码条带中的当前CTU的编码树。通过根据图6解码拆分标志来解码CTU的编码树。针对CTU在步骤1650的后续迭代中,对条带1116中的后续CTU进行解码。如果使用帧内预测模式和共用编码树结构对编码树进行编码,则编码单元具有主颜色通道(亮度或Y)和至少一个二次颜色通道(色度、Cb和Cr或CbCr)。在这种情况下,对编码树进行解码涉及根据编码树单元的拆分标志对包括主颜色通道和至少一个二次颜色通道的编码单元进行解码。

[0222] 方法1600从步骤1660继续到解码编码单元的步骤1670。在步骤1670处,视频解码器134从位流133解码编码单元。在下文中参考图18描述如是在步骤1670处实现的对编码单元进行解码的示例方法。

[0223] 方法1600从步骤1610继续到最后编码单元测试的步骤1680。在步骤1680处,处理器205测试当前编码单元是否是CTU中的最后编码单元。如果不是最后编码单元(步骤1680为“否”),则处理器205中的控制返回到解码编码单元的步骤1670以解码编码树单元的下一编码单元。如果当前编码单元是最后编码单元(步骤1680为“是”),则处理器205中的控制前进到最后CTU测试的步骤1690。

[0224] 在最后CTU测试的步骤1690处,处理器205测试当前CTU是否是条带1116中的最后CTU。如果不是条带中的最后CTU(步骤1690为“否”),则处理器205中的控制返回到解码编码树的步骤1650以解码条带1116的下一编码树单元。如果当前CTU是条带1116的最后CTU(步骤1690为“是”),则处理器205中的控制前进到最后条带测试的步骤16100。

[0225] 在最后条带测试步骤16100处,处理器205测试正在解码的当前条带是否是帧中的最后条带。如果不是帧中的最后条带(步骤16100为“否”),则处理器205中的控制返回到解码条带头部的步骤1630,并且步骤1630进行操作以解码帧中的下一条带(例如,图11的“条带2”)的条带头部。如果当前条带是帧中的最后条带(在步骤1600为“是”),则方法1600终止。

[0226] 如关于图1中的装置130所描述的,针对多个编码单元的操作方法1600进行操作以产生图像帧。

[0227] 图17示出如是在步骤1630实现的用于将条带头部解码到位流中的方法1700。方法1700可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1700可以在处理器205的执行下由视频解码器134进行。因此,方法1700可以存储在计算机可读存储介质上和/或存储器206中。

[0228] 类似于方法1500,针对帧(例如帧1101)中的当前条带或连续部分(1116)执行方法1700。方法1700开始于分区约束覆写启用测试步骤1710。在步骤1710处,处理器205测试如从SPS1110解码的分区约束覆写启用标志是否指示分区约束可以在条带级被覆写。如果分区约束可以在条带级被覆写(步骤1710为“是”),则处理器205中的控制前进到解码分区约束覆写标志的步骤1720。否则,如果分区约束覆写启用标志指示约束不可以在条带级被覆

写(步骤1710为“否”),则处理器205中的控制前进到解码其它参数的步骤1770。

[0229] 在解码分区约束覆写标志的步骤1720处,熵解码器420从位流133解码分区约束覆写标志。经解码的标志指示如在SPS1110中用信号通知的分区约束是否将针对当前条带1116被覆写。

[0230] 方法1700从步骤1720继续到分区约束覆写测试的步骤1730。在步骤1730的执行中,处理器205测试在步骤1720处解码的标志值。如果经解码的标志指示分区约束将被覆写(步骤1730为“是”),则处理器205中的控制前进到解码条带分区约束的步骤1740。否则,如果经解码的标志指示分区约束不会被覆写(步骤1730为“否”),则处理器205中的控制前进到解码其它参数的步骤1770。

[0231] 在解码条带分区约束的步骤1740处,熵解码器420从位流133解码所确定的针对条带的分区约束。针对条带的分区约束包括“slice_max_mtt_hierarchy_depth_luma”,从其导出MaxMttDepthY 1134。

[0232] 方法1700从步骤1740继续到解码QP细分级别的步骤1750。在步骤1720处,熵解码器420使用如参考图11所述的“cu_qp_delta_subdiv”句法元素对亮度CB的细分级别进行解码。

[0233] 方法1700从步骤1750继续到解码色度QP细分级别的步骤1760。在步骤1760处,熵解码器420使用如参考图11所述的“cu_chroma_qp_offset_subdiv”句法元素来解码用于用信号通知CU色度QP偏移的细分级别。

[0234] 步骤1750和1760进行操作以确定位流的特定连续部分(条带)的细分级别。步骤1630和16100之间的重复迭代进行操作以确定位流中各个连续部分(条带)的细分级别。如下所述,各个细分级别适用于相应条带(连续部分)的编码单元。

[0235] 方法1700从步骤1760继续到解码其它参数的步骤1770。在步骤1770处,熵解码器420解码来自条带头部1118的其它参数,诸如控制如去块、自适应环路滤波器、从先前用信号通知的缩放列表中可选地选择缩放列表(用于将量化参数非均匀地应用于变换块)等的特定工具所需的参数等。方法1700在执行步骤1770时终止。

[0236] 图18示出用于从位流解码编码单元的方法1800。方法1800可以由诸如经配置的FPGA、ASIC或ASSP等的设备来体现。另外,方法1800可以在处理器205的执行下由视频解码器134进行。因此,方法1800可以存储在计算机可读存储介质上和/或存储器206中。

[0237] 针对当前CTU(例如,条带1116的CTU0)的当前编码单元实现方法1800。方法1800开始于解码预测模式的步骤1810。在步骤1800处,熵解码器420从位流133解码如图13的步骤1360处确定的编码单元的预测模式。在步骤1810处对“pred_mode”句法元素进行解码,以区分针对编码单元的帧内预测、帧间预测或其它预测模式的使用。

[0238] 如果针对编码单元使用帧内预测,则在步骤1810处还对亮度帧内预测模式和色度帧内预测模式进行解码。如果针对编码单元使用帧间预测,则还可以在步骤1810处解码“合并索引”以确定来自相邻编码单元的运动矢量以供该编码单元使用,可以解码运动矢量增量以将偏移引入到从空间相邻块导出的运动矢量。还在步骤1810处对主变换类型进行解码,以针对编码单元的亮度TB在水平和垂直使用DCT-2、水平和垂直使用变换跳过、或者水平和垂直使用DCT-8和DST-7的组合之间进行选择。

[0239] 方法1800从步骤1810继续到编码残差测试的步骤1820。在步骤1820的执行中,处

处理器205通过使用熵解码器420对编码单元的“根编码块标志”进行解码来确定是否需要针对该编码单元对残差进行解码。如果存在要针对编码单元解码的任何有效残差系数(步骤1820为“是”),则处理器205中的控制前进到新QG测试的步骤1830。否则,如果不存在要解码的残差系数(步骤1820为“否”),则方法1800终止,因为已经在位流115中获得了该编码单元进行解码所需的所有信息。在方法1800终止时,进行诸如PB生成、应用环内滤波等的后续步骤,从而产生经解码样本,如参考图4所描述。

[0240] 在新QG测试步骤1830处,处理器205确定编码单元是否对应于新的量化组。如果编码单元对应于新的量化组(步骤1830为“是”),则处理器205中的控制进行到解码增量QP的步骤1840。否则,如果编码单元不对应于新的量化组(步骤1830为“否”),则处理器205中的控制前进到解码最后位置的步骤1850。新的量化组与当前模式或编码单元的细分级别有关。在解码各个编码单元时,遍历CTU的编码树的节点。在当前节点的任何子节点具有小于或等于当前条带的(即,如从“cu_qp_delta_subdiv”确定的)细分级别1136的细分级别时,新的量化组在与该节点相对应的CTU的区域中开始。量化组中包括编码残差系数的第一CU还将包括编码增量QP,从而用信号通知适用于该量化组中的残差系数的量化参数的任何变化。有效地,对各个区域(量化组)解码单个(至多一个)量化参数增量。如关于图8A至8C所描述的,各个区域(量化组)基于各个条带的编码树单元的分解和相应的细分级别(例如,如在步骤1460和1470处编码的)。换句话说,各个区域或量化组基于与编码单元相关联的细分级别与所确定的针对相应连续部分的细分级别的比较。

[0241] 在解码增量QP的步骤1840,熵解码器420从位流133解码增量QP。增量QP对预测QP与在当前量化组中使用的预期QP之间的差进行编码。通过对相邻(上方和左侧)量化组的QP求平均来导出预测QP。

[0242] 方法1800从步骤1840继续到解码最后位置的步骤1850。在执行步骤1850时,熵解码器420从位流133中解码当前变换块的二次变换系数424中的最后有效系数的位置。在第一次调用步骤1850时,针对亮度TB执行该步骤。在针对当前CU的步骤1850的后续调用中,针对Cb TB执行该步骤。如果最后位置指示有效系数在亮度块或色度块的二次变换系数集合之外(即,928或966之外),则亮度或色度通道的二次变换索引分别被推断为零。在针对Cb的迭代之后,在迭代中针对Cr TB实现该步骤。

[0243] 如关于图15的步骤1590所描述的,在一些布置中,在编码单元的最后有效系数位置之后紧接着对二次变换索引进行编码。在对同一编码单元进行解码时,如果基于在步骤1840解码的TB的最后位置的定位未推断出二次变换索引470为零,则在对该编码单元的最后有效残差系数的位置进行解码之后紧接着对二次变换索引470进行解码。在二次变换索引470在编码单元的最后有效系数位置之后紧接着被解码的布置中,方法1800从步骤1850继续到解码LFNST索引的步骤1860。在执行步骤1860时,熵解码器420在所有有效系数经受二次逆变换时(例如,在928或966内)使用截断一元码字将来自位流133的二次变换索引470解码为“lfnst_index”。当进行使用单个变换块的色度TB的联合编码时,可以针对亮度TB或色度来解码二次变换索引470。方法1800从步骤1860继续到解码子块的步骤1870。

[0244] 如果在编码单元的最后有效位置之后未紧接着解码二次变换索引470,则方法1800从步骤1850继续到解码子块的步骤1870。在步骤1870处,从位流133中解码当前变换块的残差系数(即424)作为一系列子块,从而从包含最后有效系数位置的子块前进回到包含

DC残差系数的子块。

[0245] 方法1800从步骤1870继续到最后TB测试的步骤1880。在步骤1880的执行中,处理器205测试当前变换块是否是颜色通道(即Y、Cb和Cr)上的进展中的最后变换块。如果刚解码的(当前)变换块用于Cr TB,则在处理器205中的控制中,所有TB已经被解码(步骤1880为“是”),方法1800前进到解码亮度LFNST索引的步骤1890。否则,如果TB尚未被解码(步骤1880为“否”),则处理器205中的控制返回到解码最后位置的步骤1850。在步骤1850的迭代中,选择下一TB(遵循Y、Cb、Cr的顺序)用于解码。

[0246] 方法1800从步骤1880继续到解码亮度LFNST索引的步骤1890。在步骤1890的执行中,如果亮度TB的最后位置在经受二次逆变换的系数集合(例如,928或966)内并且亮度TB正在水平和垂直地使用DCT-2作为主变换,则由熵解码器420从位流133解码要应用于亮度TB的二次变换索引470。如果亮度TB的最后有效位置指示在经受二次逆变换的系数集合之外(例如,在928或966之外)存在有效主系数,则亮度二次变换索引被推断为零(不应用二次变换)。步骤1890处解码的二次变换索引在图12中被指示为1220(或联合CbCr模式中的1230)。

[0247] 方法1800从步骤1890继续到解码色度LFNST索引的步骤1895。在步骤1895处,如果各个色度TB的最后位置在经受二次逆变换的系数集合(例如,928或966)内,则由熵解码器420从位流133中解码要应用于色度TB的二次变换索引470。如果任一色度TB的最后有效位置指示在经受二次逆变换的系数集合之外(例如,在928或966之外)存在有效主系数,则色度二次变换索引被推断为零(不应用二次变换)。步骤1895处解码的二次变换索引在图12中被指示为1221(或联合CbCr模式中的1230)。在解码亮度和色度的单独索引时,可以使用各个截断一元码字的单独算术上下文或者可以共用上下文,以使得亮度和色度截断一元码字中各自的第n bin共用相同的上下文。

[0248] 有效地,步骤1890和1895分别涉及:解码第一索引(诸如1220等)以选择用于亮度(主颜色)通道的核,以及解码第二索引(诸如1221等)以选择用于至少一个色度(二次颜色通道)的核。

[0249] 方法1800从步骤1895继续到进行逆二次变换的步骤18100。在该步骤处,逆二次变换模块436根据当前变换块的二次变换索引470对经解码的残差变换系数424进行逆二次变换,以产生二次变换系数432。步骤1890处解码的二次变换索引被应用于亮度TB,并且步骤1895处解码的二次变换索引被应用于色度TB。亮度和色度的核选择还分别取决于亮度帧内预测模式和色度帧内预测模式(其各自在步骤1810处被解码)。步骤18100根据亮度的LFNST索引来选择核,并且根据色度的LFNST索引来选择核。

[0250] 方法1800从步骤18100继续到逆量化主变换系数的步骤18110。在步骤18110处,逆量化器模块428根据量化参数474对二次变换系数432进行逆量化,以产生逆量化的主变换系数440。如果在步骤1840处解码了增量QP,则熵解码器420根据量化组(区域)的增量QP和图像帧的较早编码单元的量化参数来确定量化参数。如上所述,较早编码单元通常涉及相邻的左上编码单元。

[0251] 方法1800从步骤1870继续到进行主变换的步骤18120。在步骤18120处,逆主变换模块444根据编码单元的主变换类型进行逆主变换,使得变换系数440被转换为空间域的残差样本448。在各个颜色通道上进行逆主变换,首先对亮度通道(Y)进行逆主变换,然后在针

对当前TU的步骤1650的后续调用时对Cb和Cr TB执行逆主变换。步骤18100至18120有效地操作,以通过将在步骤1890处根据亮度的LFNST索引所选择的核应用于亮度通道的经解码残差系数,并且将在步骤1890处根据色度的LFNST索引所选择的核应用于至少一个色度通道的经解码残差系数,来解码当前编码单元。

[0252] 方法1800在执行步骤18120后终止,其中处理器205中的控制返回到方法1600。

[0253] 关于预测模式是帧内预测并且变换是DCT-2的共用编码树结构的示例来描述步骤1850至18120。例如,仅针对使用帧内预测和共用编码树结构的编码单元,从位流中解码应用于亮度TB的二次变换索引(1890)。类似地,仅针对使用帧内预测和共用编码树结构的编码单元,从位流中解码应用于色度TB的二次变换索引(1895)。除了使用已知方法的共用编码树结构之外,可以针对帧间预测模式或针对帧内预测模式来实现诸如解码子块(1870)、逆量化主变换系数(18110)和进行主变换等的步骤的操作。无论预测模式或结构如何,都以所描述的方式进行步骤1810至1840。

[0254] 一旦方法1800终止,就进行用于解码编码单元的后续步骤(包括由模块476生成帧内预测样本480、由模块450将经解码残差样本448与预测块452求和、以及应用环内滤波器模块488以产生滤波样本492),作为帧数据135输出。

[0255] 图19A和19B示出用于对亮度和色度通道应用或绕过二次变换的规则。图19A示出示例用于在由共用编码树产生的CU中的亮度和色度通道中应用二次变换的条件的表1900。

[0256] 如果亮度TB的最后有效系数位置指示不是由正向二次变换产生并且因此不经受逆二次变换的经解码有效系数,则存在条件1901。如果亮度TB的最后有效系数位置指示确实由正向二次变换产生并因此经受逆二次变换的经解码有效系数,则存在条件1902。另外,对于亮度通道,主变换类型需要是DCT-2以使条件1902存在,否则条件1901存在。

[0257] 如果一个或两个色度TB的最后有效系数位置指示不是由正向二次变换产生并且因此不经受逆二次变换的经解码有效系数,则存在条件1910。如果一个或两个色度TB的最后有效系数位置指示确实由正向二次变换产生并且因此经受逆二次变换的经解码有效系数,则存在条件1911。另外,色度块的宽度和高度需要为至少四个样本(例如,当使用4:2:0或4:2:2色度格式时的色度子采样可以得到两个样本的宽度或高度),以使条件1911存在。

[0258] 如果条件1901和1910存在,则不(独立地或联合地)用信号通知二次变换索引,并且不在亮度或色度中应用二次变换索引,即1920。如果条件1901和1911存在,则用信号通知一个二次变换索引以指示所选择的核的应用或仅针对亮度通道而绕过,即1921。如果条件1902和1910存在,则用信号通知一个二次变换索引以指示所选择的核的应用或仅针对色度通道而绕过,即1922。如果条件1911和1902存在,则具有独立信号通知的布置用信号通知两个二次变换索引,一个用于亮度TB,并且一个用于色度TB,即1923。当条件1902和1911存在时,具有单个用信号通知的二次变换索引的布置使用一个索引来控制亮度和色度的选择,尽管所选择的核也取决于亮度和色度帧内预测模式,其可能有所不同。将二次变换应用于亮度或色度(即1921和1922)的能力得到编码效率增加。

[0259] 图19B示出在步骤1360处视频编码器114可用的搜索选项的表1950。亮度(1952)和色度(1953)的二次变换索引分别被示出为1952和1953。索引值0指示绕过二次变换,并且索引值1和2指示针对从亮度或色度帧内预测模式导出的候选集合使用两个核中的哪一个。存在九种组合(“0,0”至“2,2”)的由此得到的搜索空间,其可能受到参考图19A描述的约束来

约束。与搜索所有可允许的组合相比,三个组合的简化搜索(1951)可以仅测试亮度和色度二次变换索引相同的组合,经受将最后有效系数位置指示存在仅主系数的通道的索引归零。例如,当条件1921存在时,选项“1,1”和“2,2”分别变为“0,1”和“0,2”(即,1954)。当条件1922存在时,选项“1,1”和“2,2”分别变为“1,0”和“2,0”(即1955)。当条件1920存在时,不需要用信号通知二次变换索引,并且使用选项“0,0”。实际上,条件1921和1922允许共用树CU中的选项“0,1”、“0,2”、“1,0”和“2,0”,从而得到更高的压缩效率。如果这些选项被禁止,则条件1901或1910中的任一个将使得条件1920(即,选项“1,1”和“2,2”)将被禁止,使得使用“0,0”(参见1956)。

[0260] 用信号通知条带头部中的量化组细分级别提供了图像级之下的更高粒度的控制。更高粒度的控制对于编码保真度要求从图像的一部分到另一部分变化的应用是有利的,并且特别是对于多个编码器可能需要稍微独立地操作以提供实时处理能力的应用是有利的。用信号通知条带头部中的量化组细分级别也与在用信号通知条带头部中的分区覆写设置和缩放列表应用设置一致。

[0261] 在视频编码器114和视频解码器134的一个布置中,色度帧内预测块的二次变换索引总是被设置为零,即,二次变换不应用于色度帧内预测块。在这种情况下,不需要用信号通知色度二次变换索引,因此可以省略步骤15130和1895,并且相应地简化步骤1360、1570和18100。

[0262] 如果共用树中的编码树中的节点具有64个亮度样本的区域,则用二元或四叉树拆分进一步拆分将产生较小亮度CB,诸如 4×4 块等,但将不产生较小色度CB。代替地,存在大小对应于64个亮度样本的区域的单个色度CB,诸如 4×4 色度CB等。类似地,具有128个亮度样本的区域并且经受三元拆分的编码树节点产生较小亮度CB和一个色度CB的集合。各个亮度CB具有相应的亮度二次变换索引,并且色度CB具有色度二次变换索引。

[0263] 当编码树中的节点具有64的区域并且用信号通知进一步拆分或者具有128个亮度样本的区域并且用信号通知三元拆分时,仅在亮度通道中应用拆分,并且所得CB(针对各个色度通道的若干亮度CB和一个色度CB)是全部帧内预测的或全部帧间预测的。当CU具有四个亮度样本的宽度或高度并且包括用于颜色通道(Y、Cb和Cr)中的各个的一个CB时,CU的色度CB具有两个样本的宽度或高度。宽度或高度为两个样本的CB不利用16点或48点LFNST核操作,因此不需要二次变换。对于宽度或高度为两个样本的块,不需要进行步骤15130、1895、1360、1570和18100。

[0264] 在视频编码器114和视频解码器134的另一布置中,当亮度和色度中的任一个或这两者仅在仅经受主变换的相应TB的区域中包含非有效残差系数时,用信号通知单个二次变换索引。如果亮度TB在经解码残差(例如1066、968)的非二次变换区域中包含有效残差系数,或者被指示为不使用DCT-2作为主变换,则所指示的二次变换核(或二次变换旁路)仅应用于色度TB。如果任一色度TB在经解码残差的非二次变换区域中包含有效残差系数,则所指示的二次变换核(或二次变换旁路)仅应用于亮度TB。即使当对于色度TB不可能时,应用二次变换对于亮度TB也变得可能,反之亦然,从而与要求在任何TB可以经受二次变换之前所有TB的最后位置都在二次系数域内相比,给出了编码效率增加。另外,针对共用编码树中的CU仅需要一个二次变换索引。当亮度主变换是DCT-2时,可以将二次变换对于色度以及对于亮度推断为禁用。

[0265] 在视频编码器114和视频解码器134的另一布置中,二次变换(分别由模块330和436)仅应用于CU的亮度TB,而不应用于该CU的任何色度TB。不存在用于色度通道的二次变换逻辑得到较低的复杂性,例如较低的执行时间或减小的硅面积。不存在用于色度通道的二次变换逻辑使得仅需要用信号通知一个二次变换索引,其可以在亮度TB的最后位置之后用信号通知。也就是说,代替步骤15120和1890,针对亮度TB进行步骤1590和1860。在这种情况下,省略步骤15130和1895。

[0266] 在视频编码器114和视频解码器134的另一布置中,在PPS1112中用信号通知定义量化组大小的句法元素(即,cu_chroma_qp_offset_subdiv和cu_qp_delta_subdiv)。即使分区约束在条带头部1118中被覆写,也根据在SPS 1110中用信号通知的分区约束来定义细分级别的值的范围。例如,cu_qp_delta_subdiv和cu_chroma_qp_offset_subdiv的范围被定义为 0 至 $2 * (\log_2 \text{ctu_size_minus}5 + 5 - (\text{MinQtLog}2\text{SizeInterY}$ 或 $\text{MinQtLog}2\text{SizeIntraY}) + \text{MaxMttDepthY_SPS})$ 。值MaxMttDepthY是从SPS1110导出的。也就是说,在当前条带是I条带时,将MaxMttDepthY设置为等于sps_max_mtt_hierarchy_depth_intra_slice_luma,并且在当前条带是P或B条带时,将MaxMttDepthY设置为等于sps_max_mtt_hierarchy_depth_inter_slice。对于分区约束被覆写为比SPS1110中用信号通知的深度更浅的条带,如果从PPS1112确定的量化组细分级别高于(深于)从条带头部确定的较浅编码树深度下的最高可实现细分级别,则条带的量化组细分级别被裁剪为等于条带的最高可实现细分级别。例如,对于特定条带,cu_qp_delta_subdiv和cu_chroma_qp_offset_subdiv被裁剪为在 0 至 $2 * (\log_2 \text{ctu_size_minus}5 + 5 - (\text{MinQtLog}2\text{SizeInterY}$ 或 $\text{MinQtLog}2\text{SizeIntraY}) + \text{MaxMttDepthY_slice_header})$ 内,并且裁剪后的值用于该条带。值MaxMttDepthY_slice_header是从条带头部1118导出的,即,MaxMttDepthY_slice_header被设置为等于slice_max_mtt_hierarchy_depth_luma。

[0267] 在视频编码器114和视频解码器134的又一布置中,根据从PPS1112解码的cu_chroma_qp_offset_subdiv和cu_qp_delta_subdiv来确定细分级别,以导出亮度和色度细分级别。当从条带头部1118解码的分区约束导致条带的细分级别的不同范围时,根据从SPS1110解码的分区约束,调整应用于条带的细分级别以维持相对于最深允许细分级别的相同偏移。例如,如果SPS1110指示最大细分级别为4且PPS1112指示细分级别为3且条带头部1118将最大值减小到3,则将条带内应用的细分级别设置为2(维持相对于最大允许细分级别的偏移为1)。调整量化组区域以对应于特定条带的分区约束的变化允许较不频繁地(即,在PPS级)用信号通知细分级别,同时提供适应于条带级分区约束变化的粒度。使用根据从SPS1110解码的分区约束所定义的范围在PPS1112中用信号通知细分级别的布置(其中,可能稍后基于从条带头部1118解码的覆写分区约束进行调整)避免了使PPS句法元素依赖于在条带头部1118中完成的分区约束的解析依赖性问题。

[0268] 产业上的可利用性

[0269] 所描述的布置适用于计算机和数据处理行业,并且特别地适用于对诸如视频和图像信号等的信号进行编码或解码的数字信号处理,从而实现高压压缩效率。

[0270] 这里描述的布置增加了在从传入视频数据生成高度压缩的位流时提供给视频编码器的灵活性。帧中的不同区域或子图片的量化能够以变化的粒度以及从一个区域到另一区域不同的粒度来控制,从而减少经编码的残差数据的量。在需要时,例如,对于如上所述

的360度图像,可以相应地实现更高的粒度。

[0271] 在一些布置中,如关于步骤15120和15130(以及相应地步骤1890和1895)所描述的,可以针对亮度和色度独立地控制二次变换的应用,从而实现进一步减少经编码的残差数据。描述了视频解码器,其具有对由这种视频编码器产生的位流进行解码的必要功能。

[0272] 前述仅说明本发明的一些实施例,并且可以在没有背离本发明的范围和精神的情况下对本发明进行修改和/或改变,其中这些实施例仅是示例性而非限制性的。

[0273] 相关申请的引用

[0274] 本申请根据35U.S.C.§119要求于2019年9月17日提交的澳大利亚专利申请2019232801的优先权益,这里出于所有目的将该专利申请的全部内容通过引用合并于此。

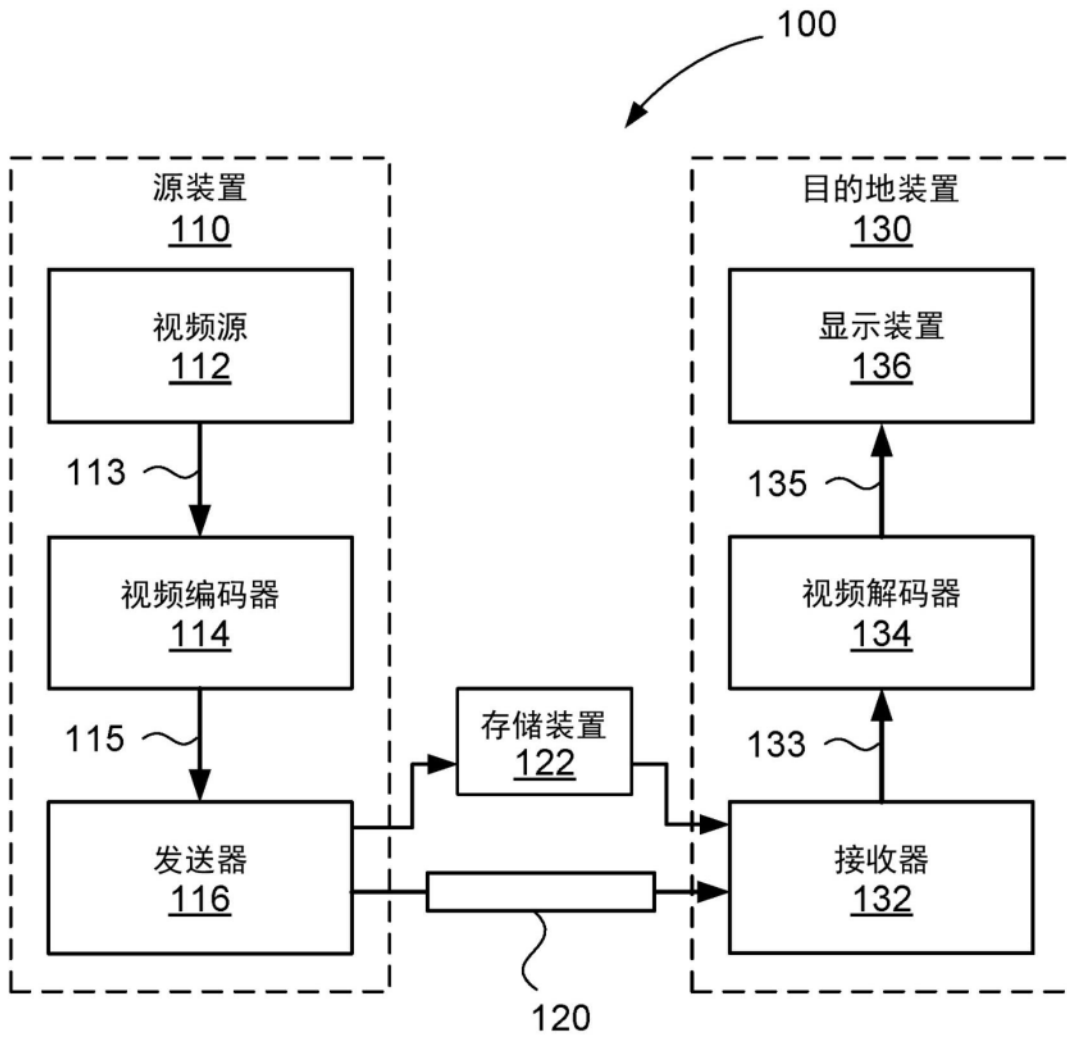


图1

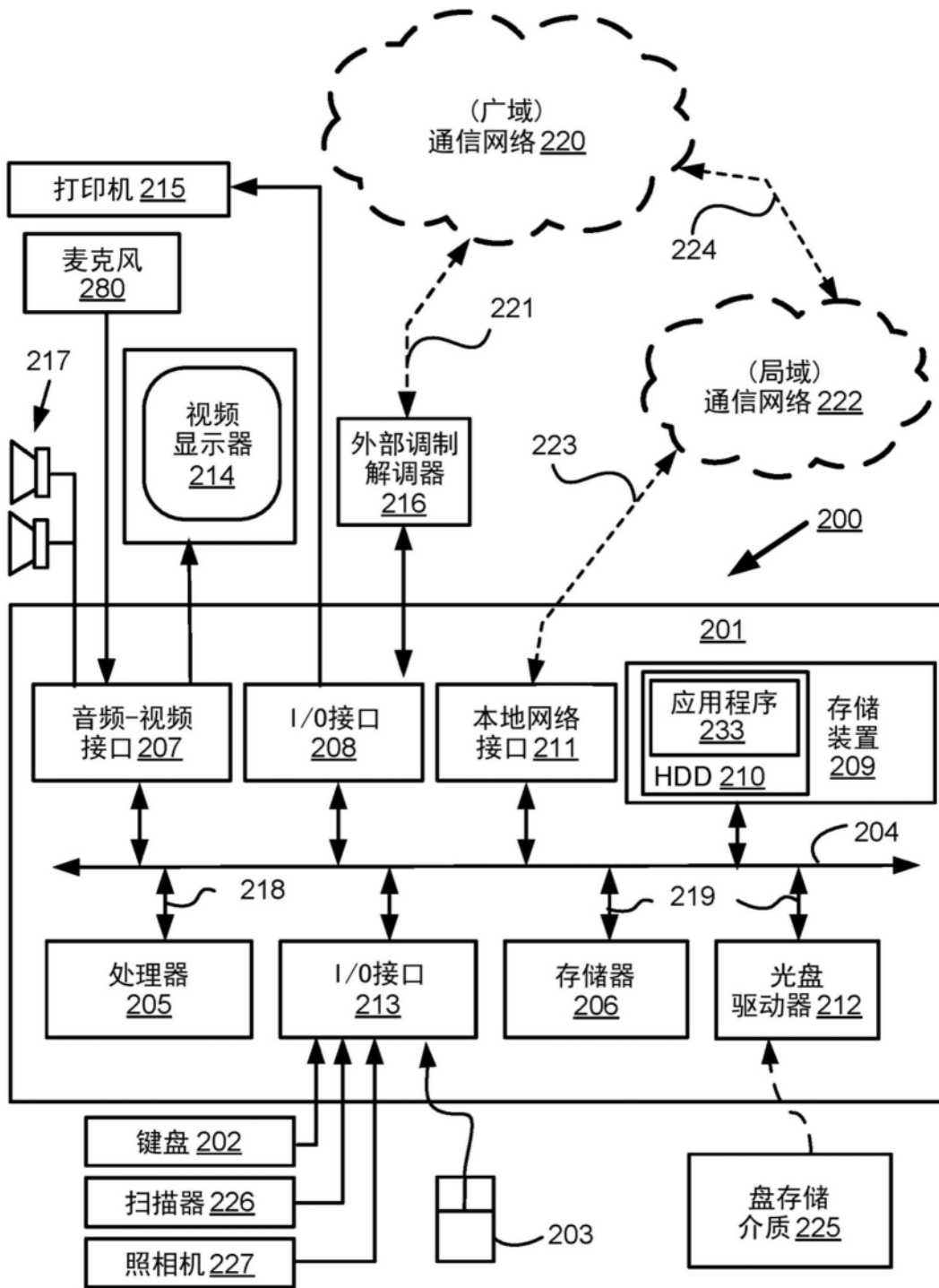


图2A

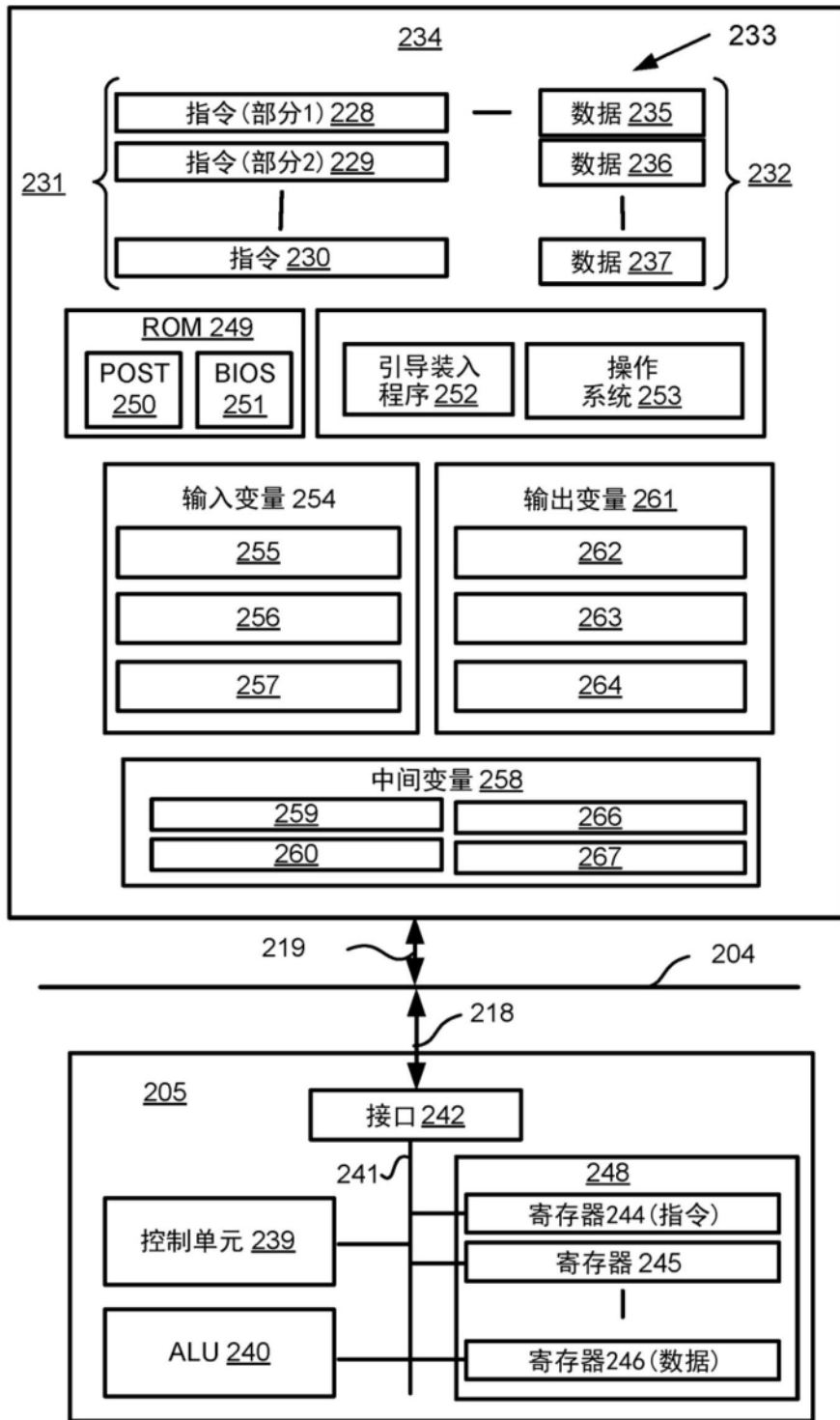


图2B

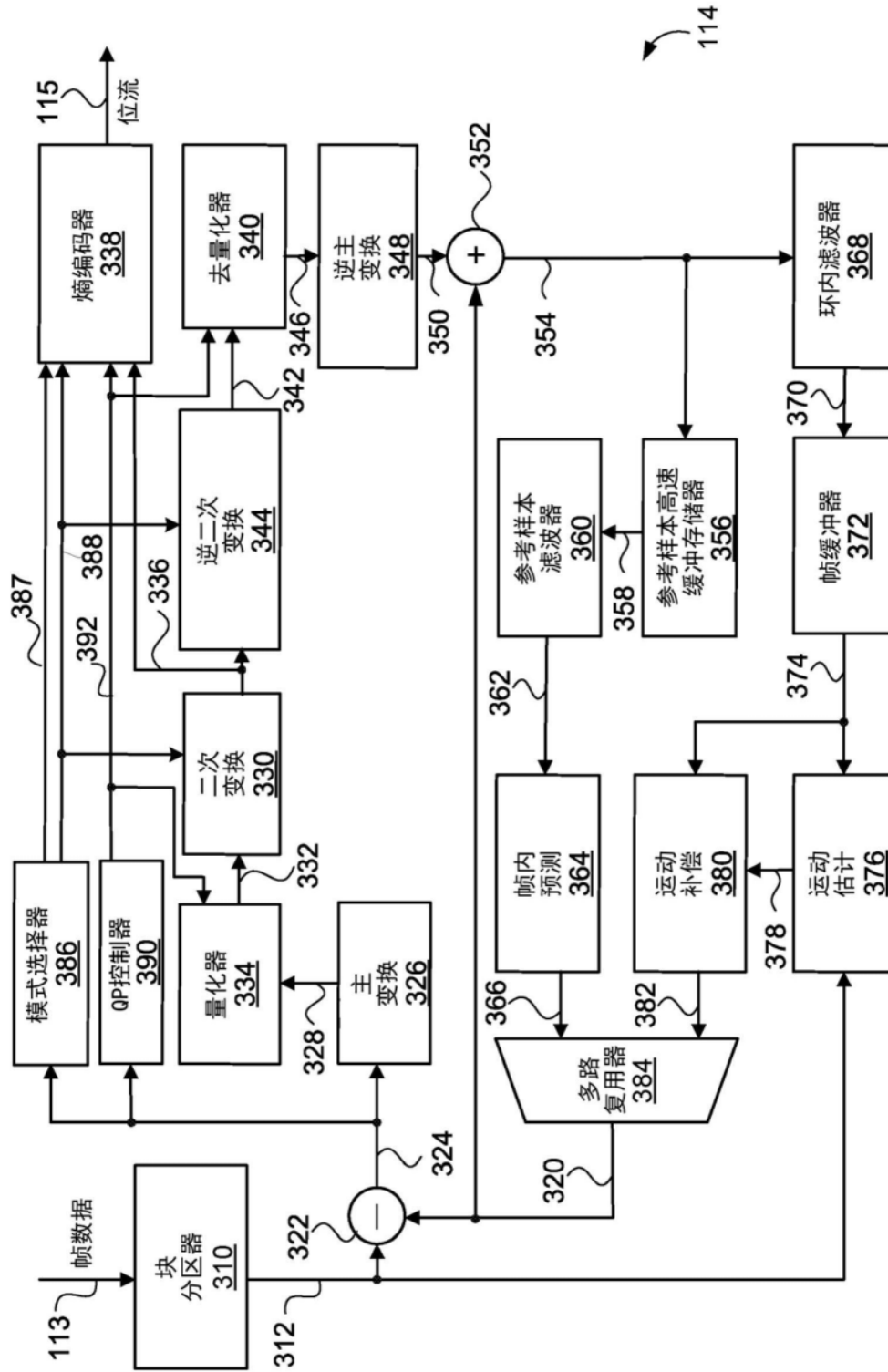


图3

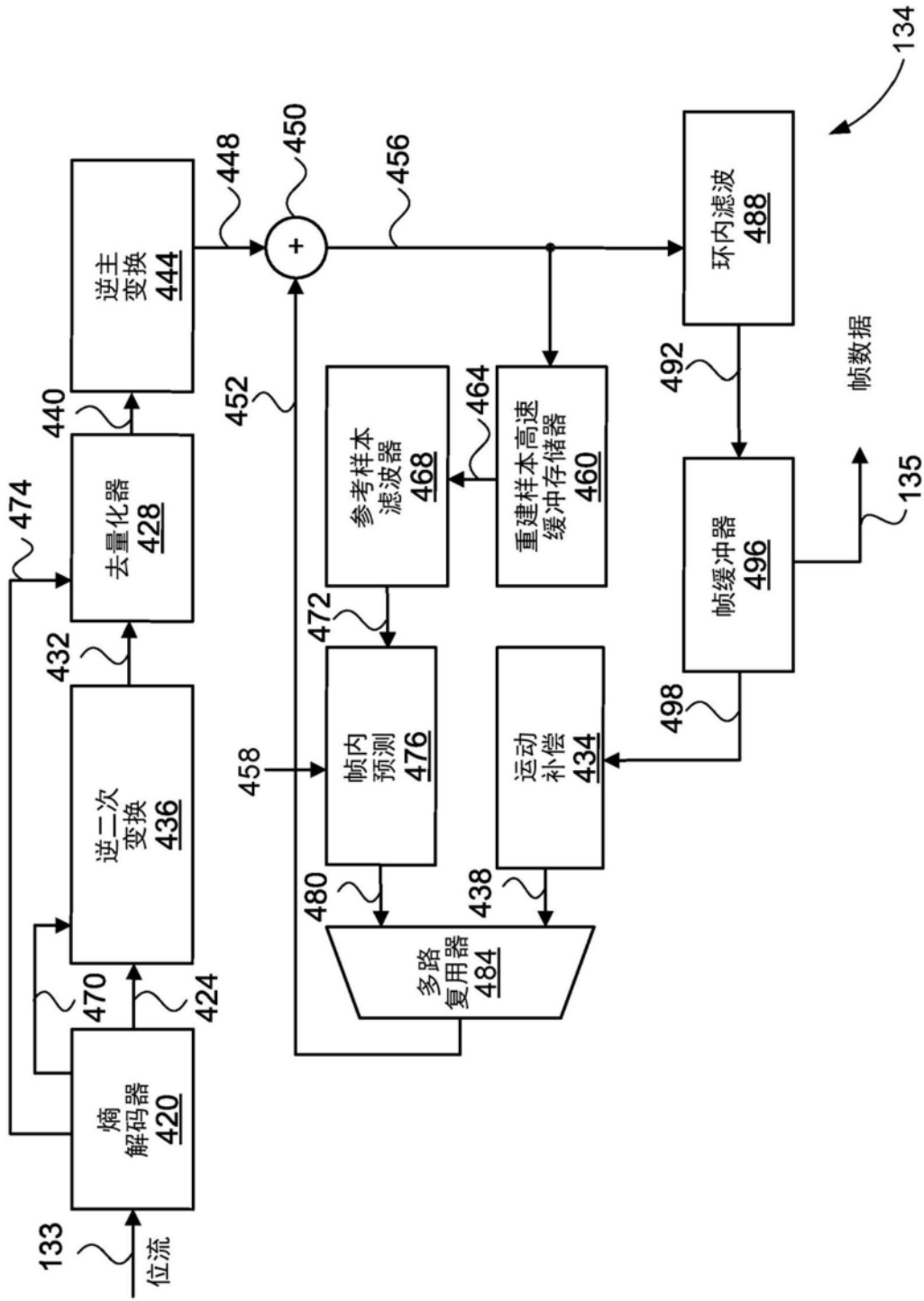


图4

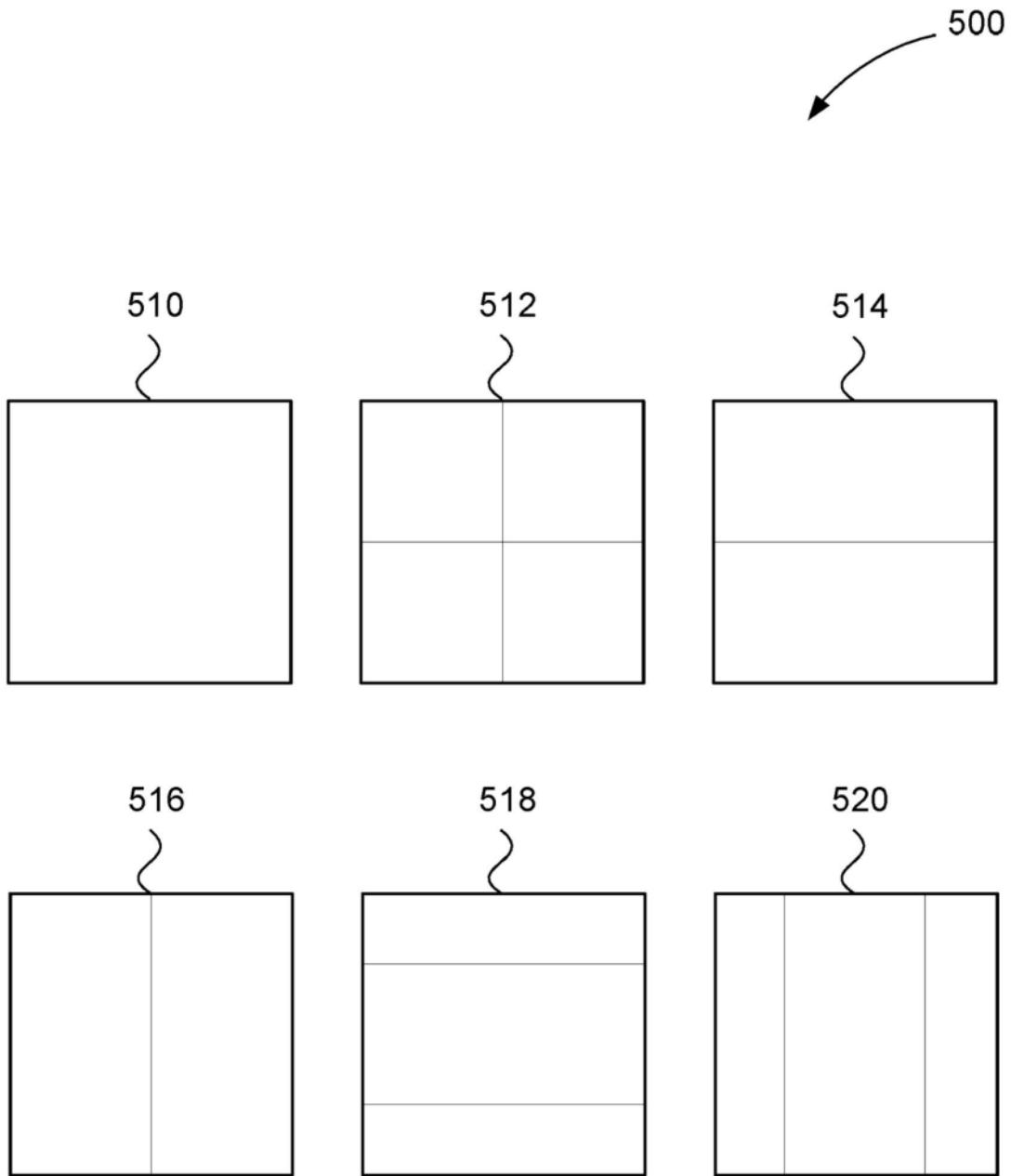


图5

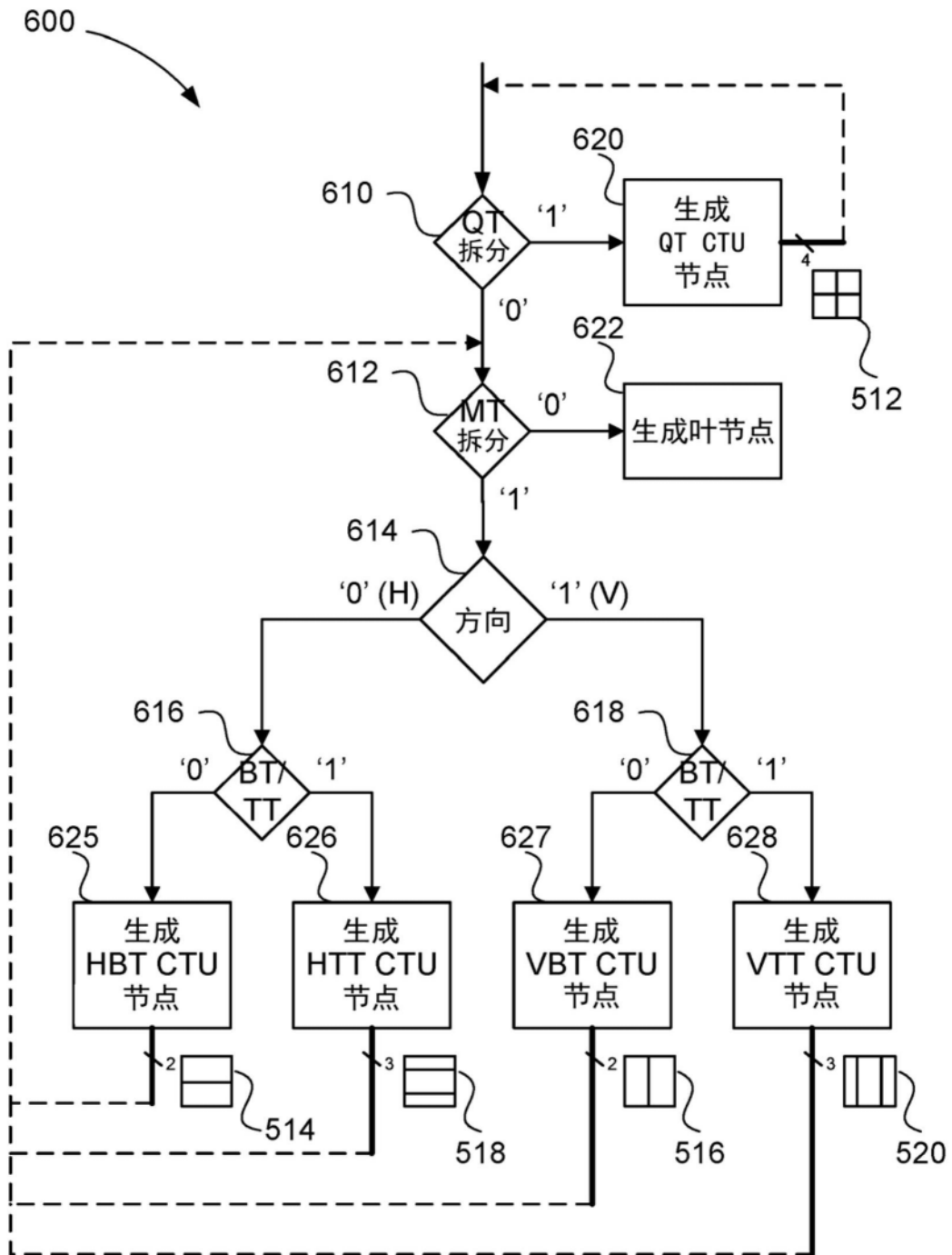


图6

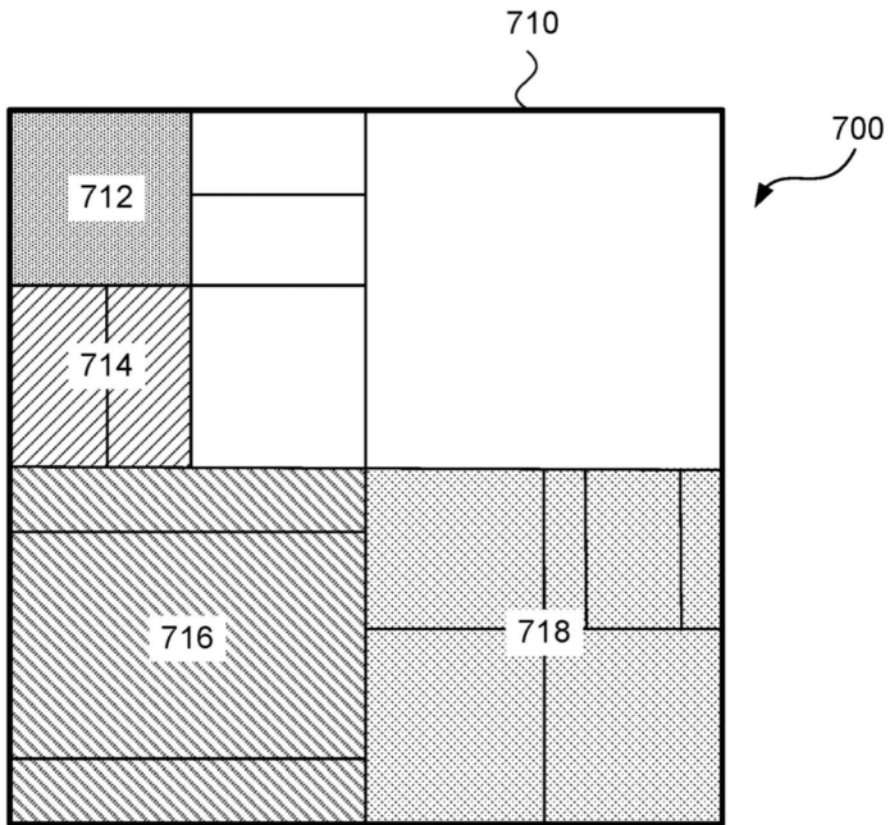


图7A

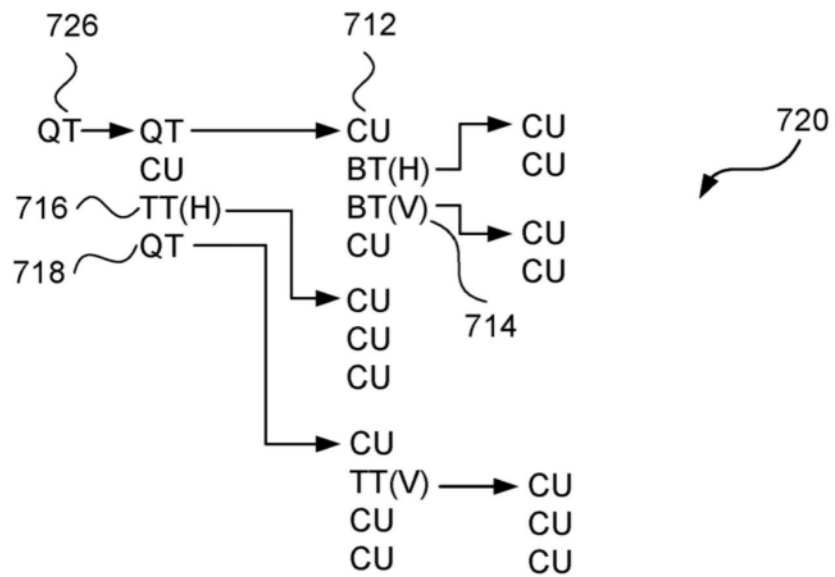


图7B

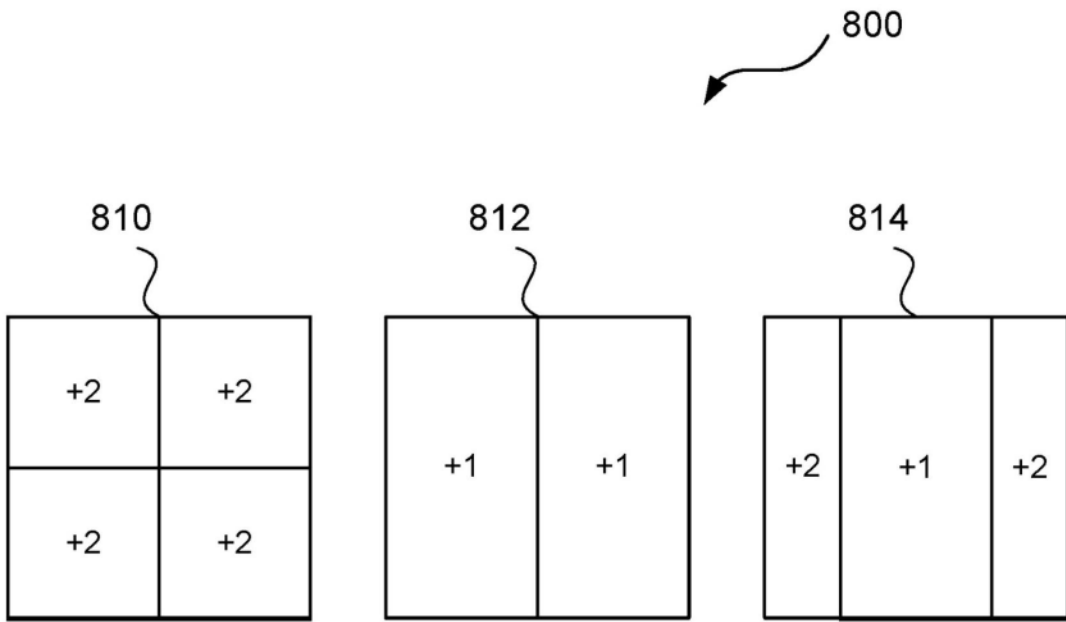


图8A

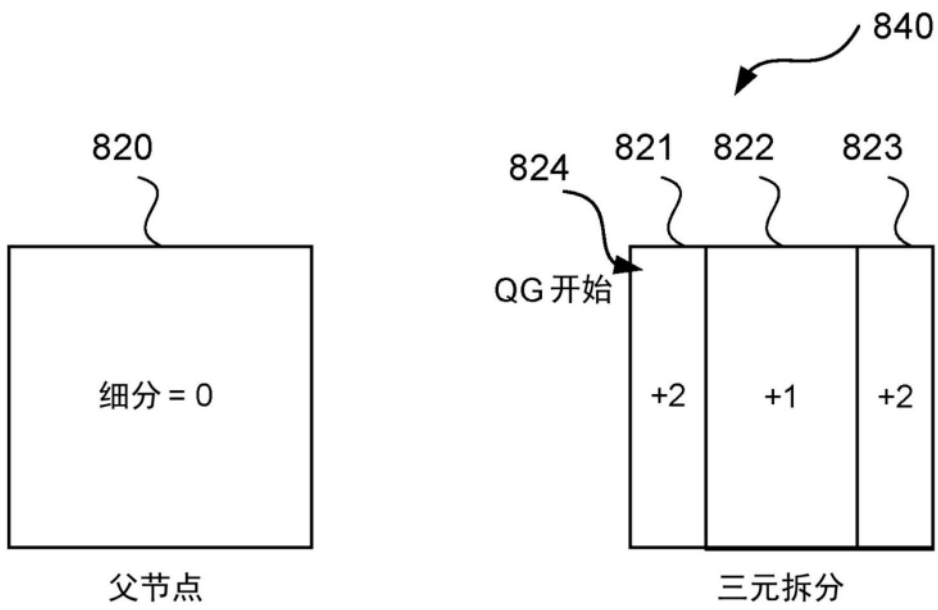


图8B

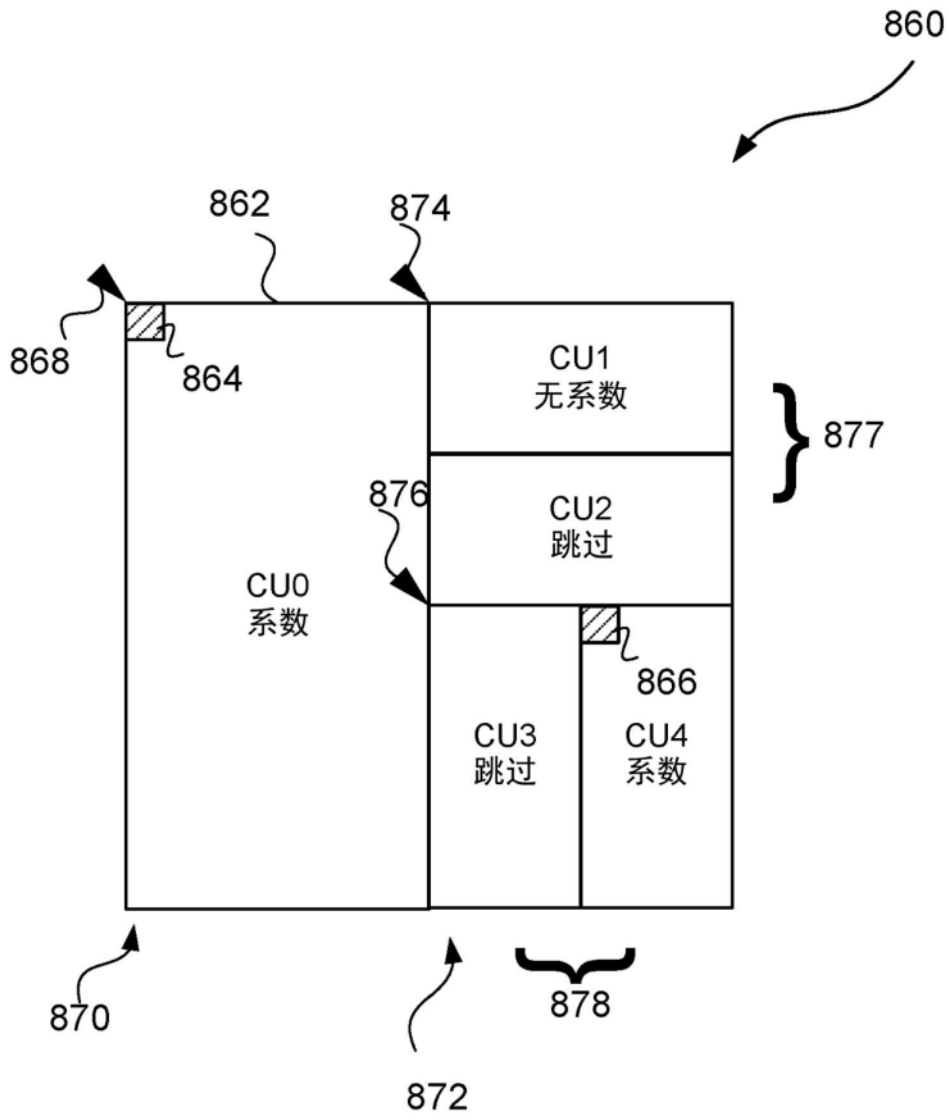


图8C

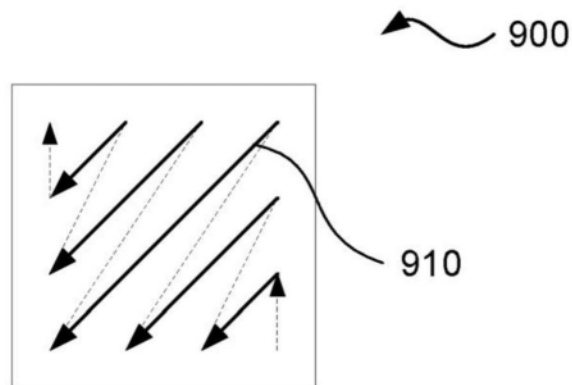


图9A

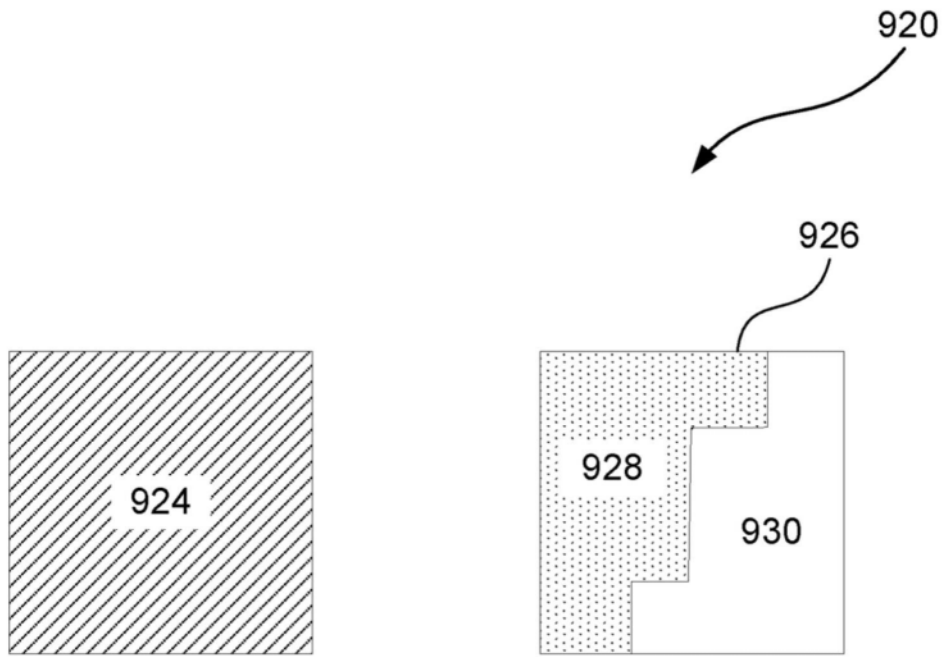


图9B

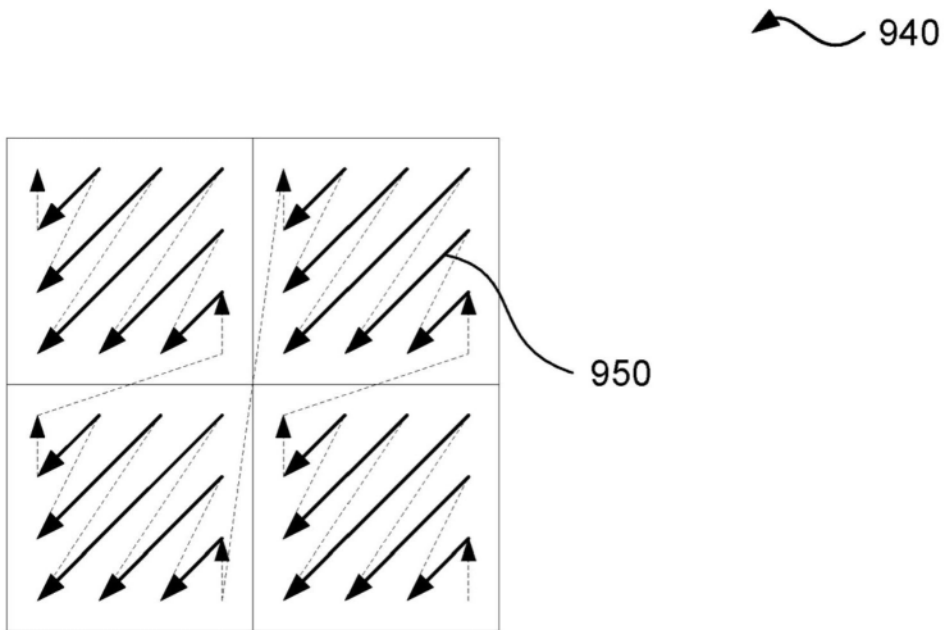


图9C

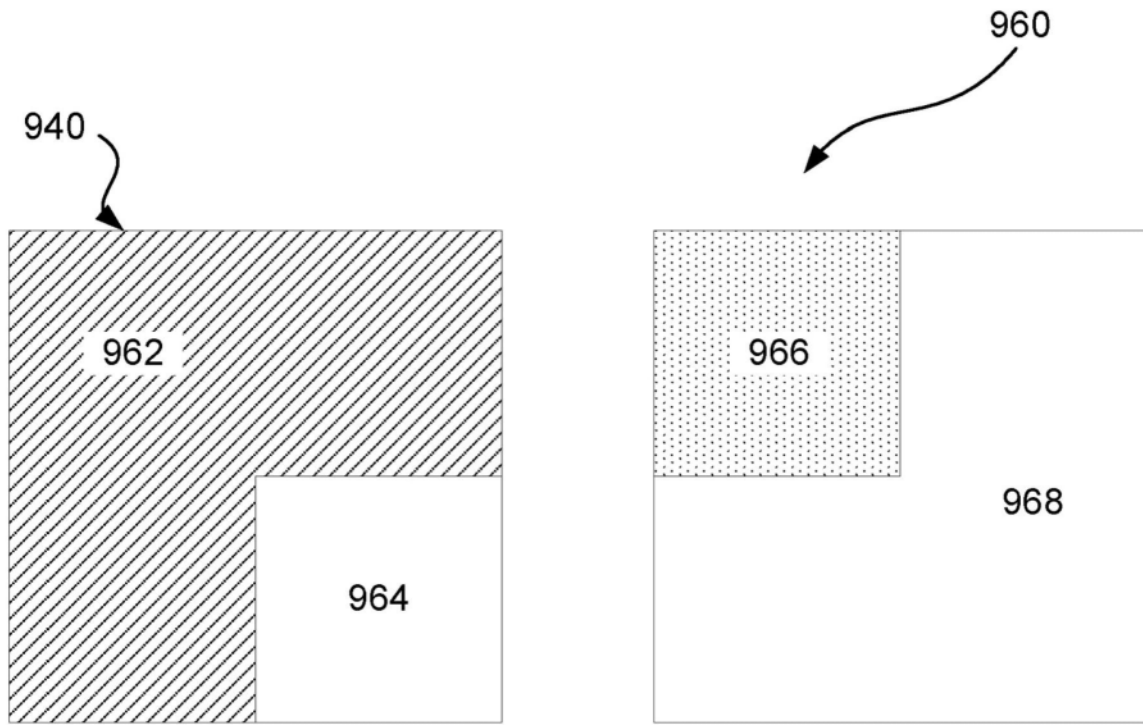


图9D

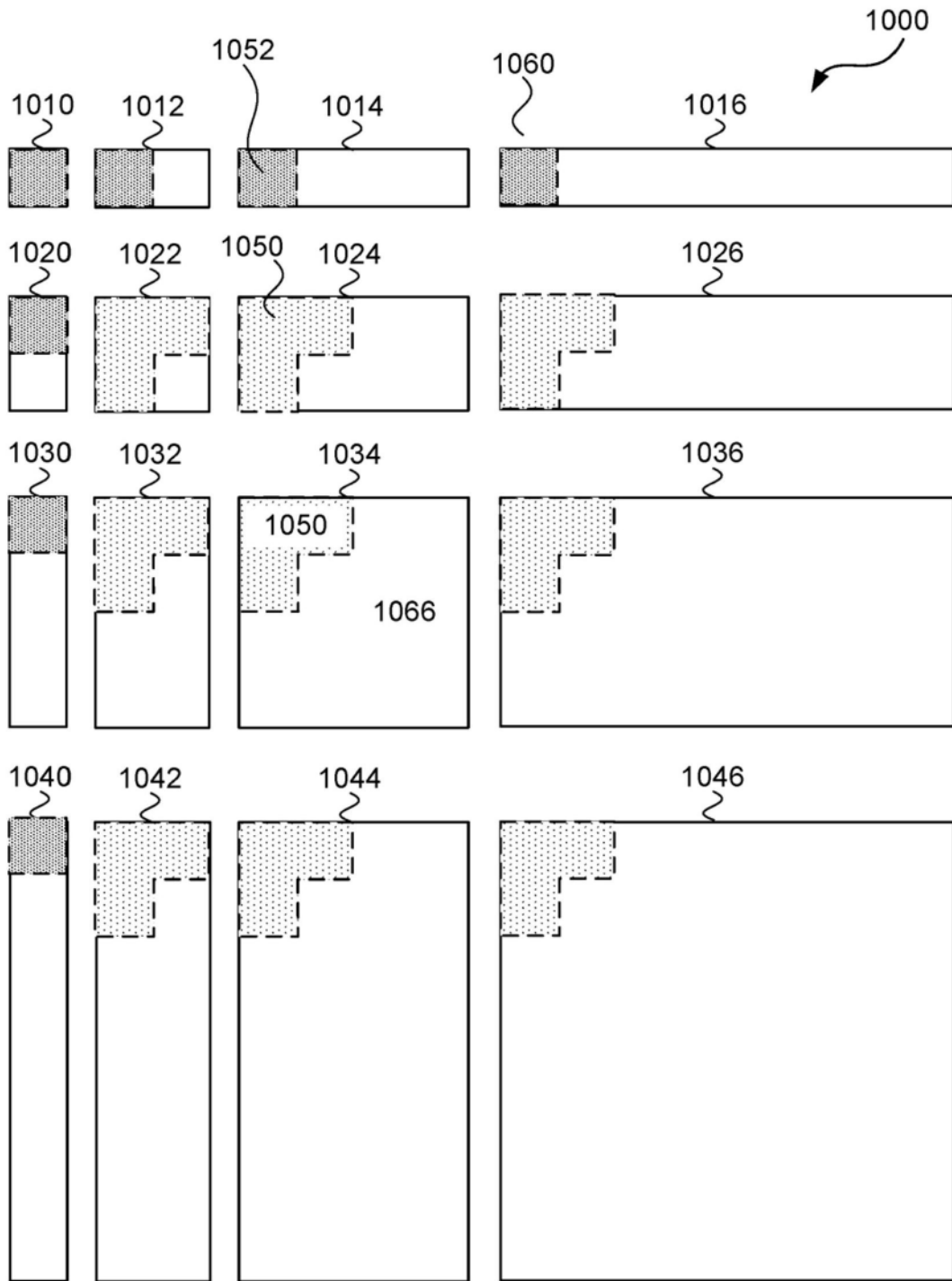


图10

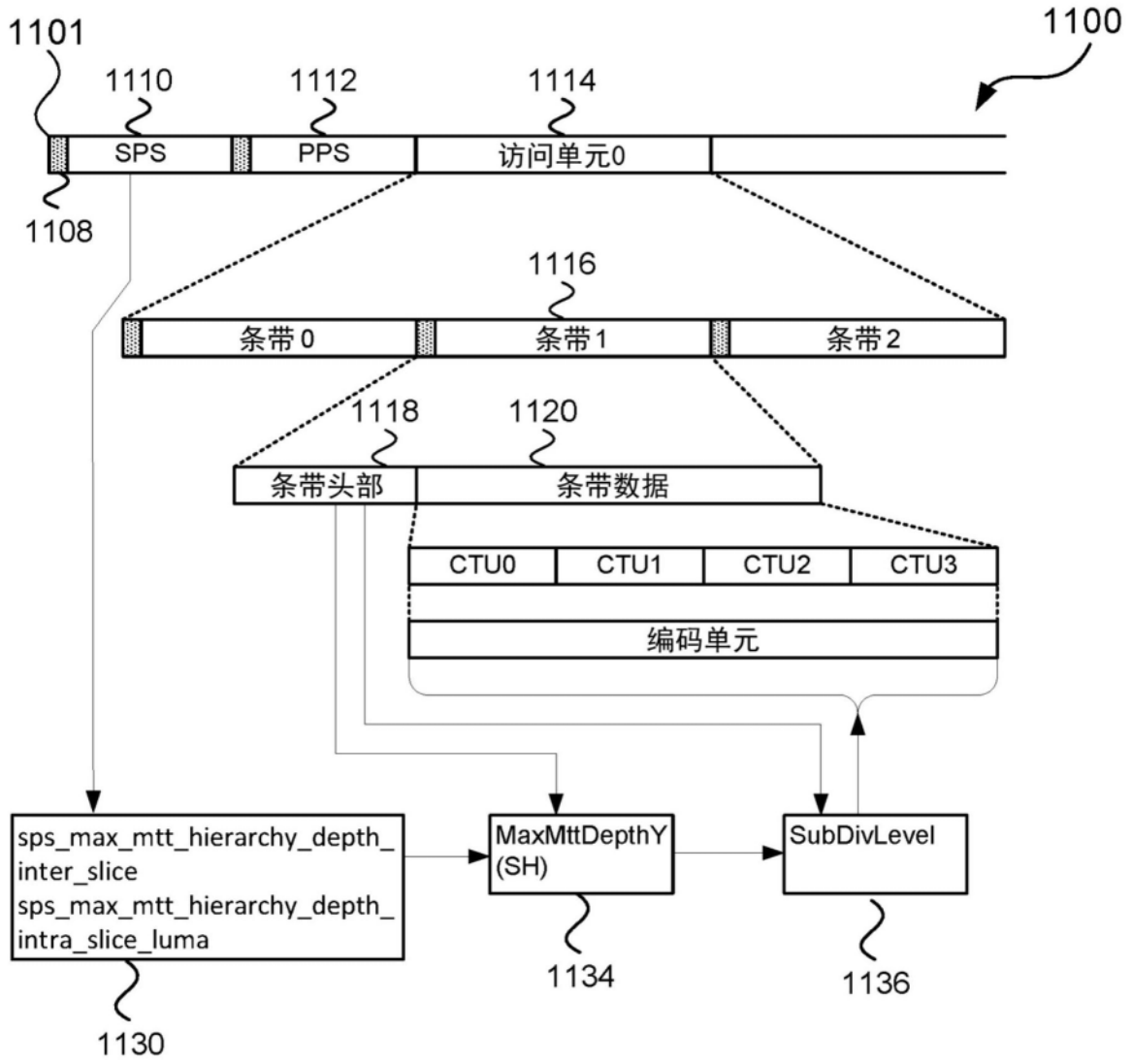


图11

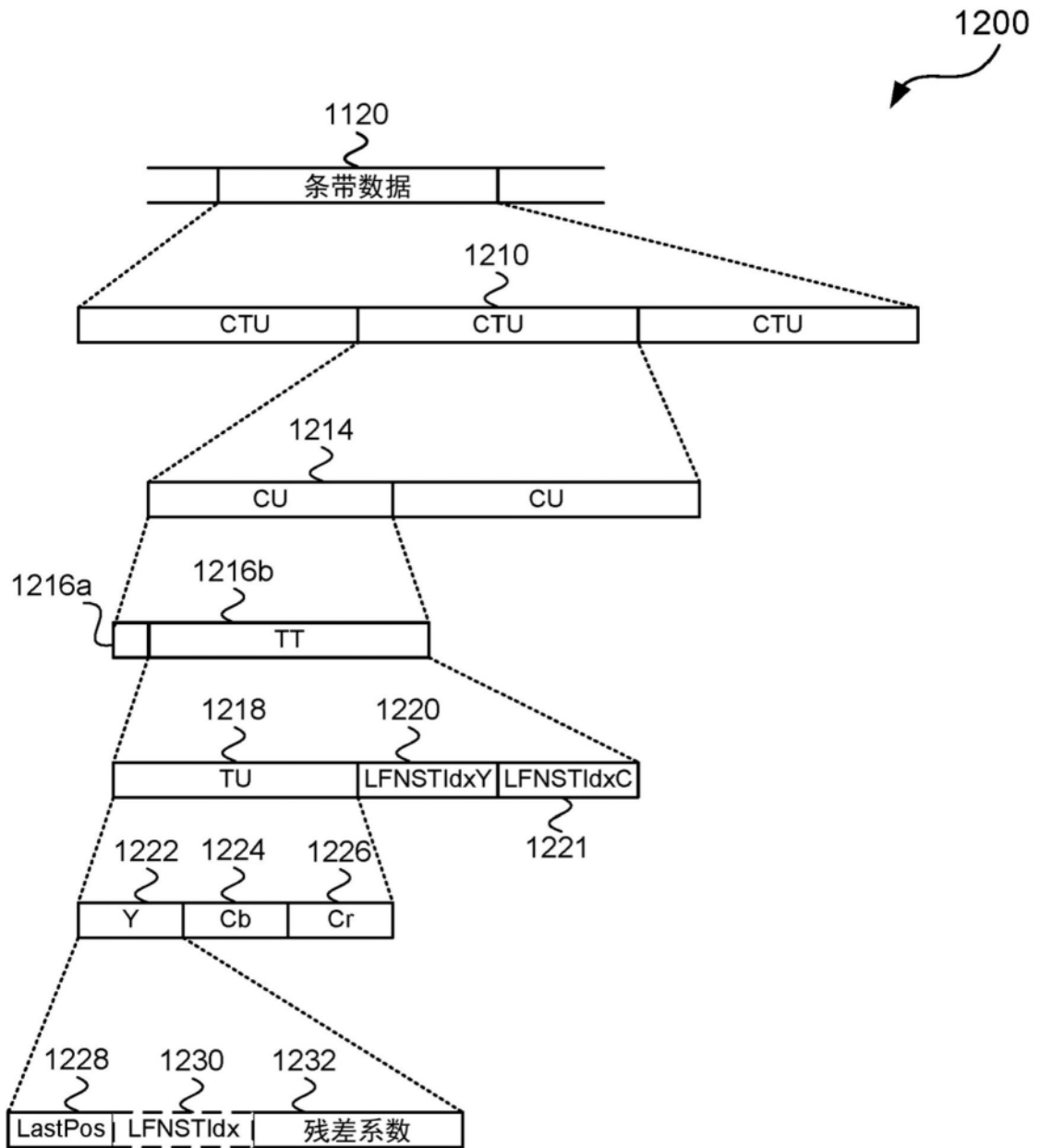


图12

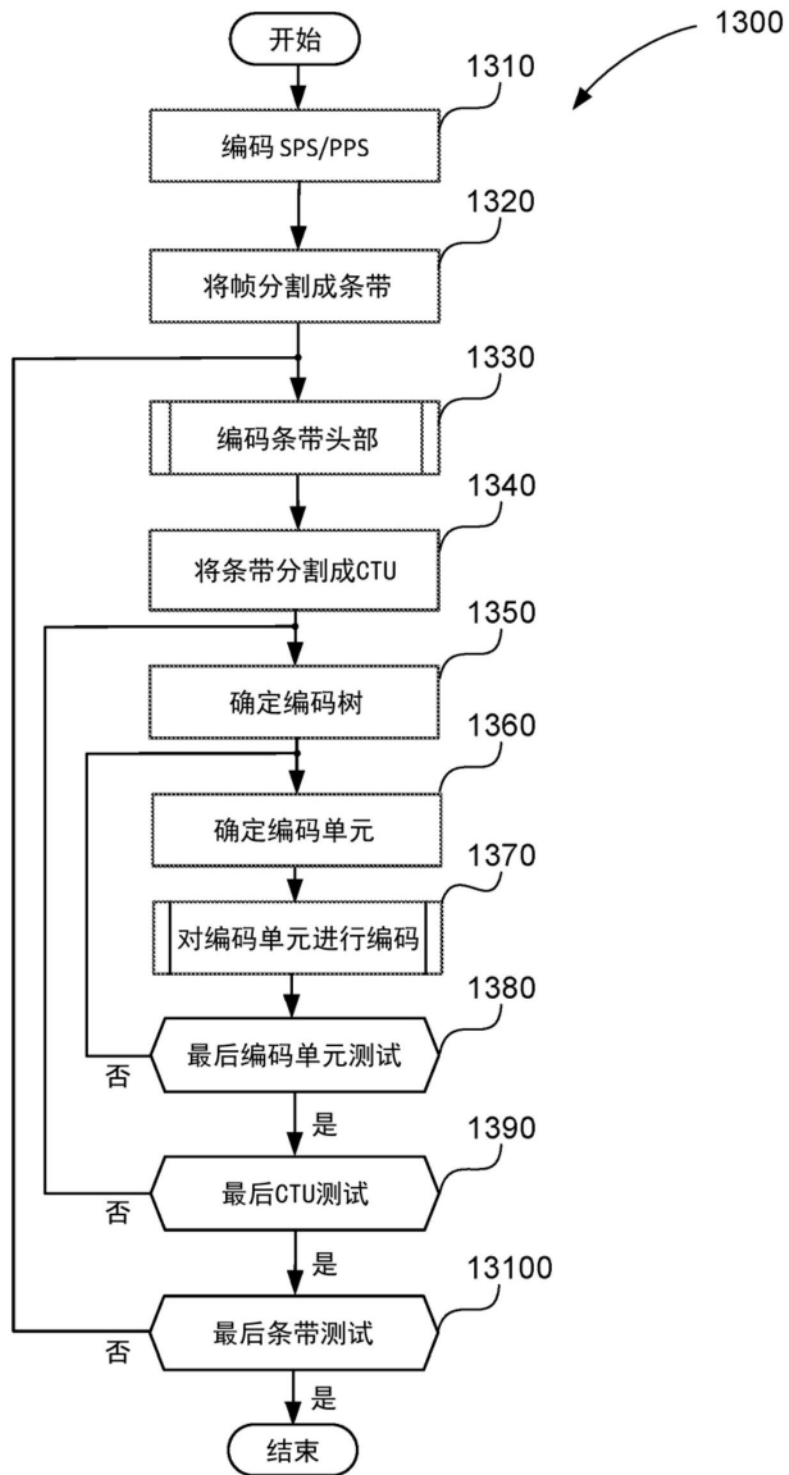


图13

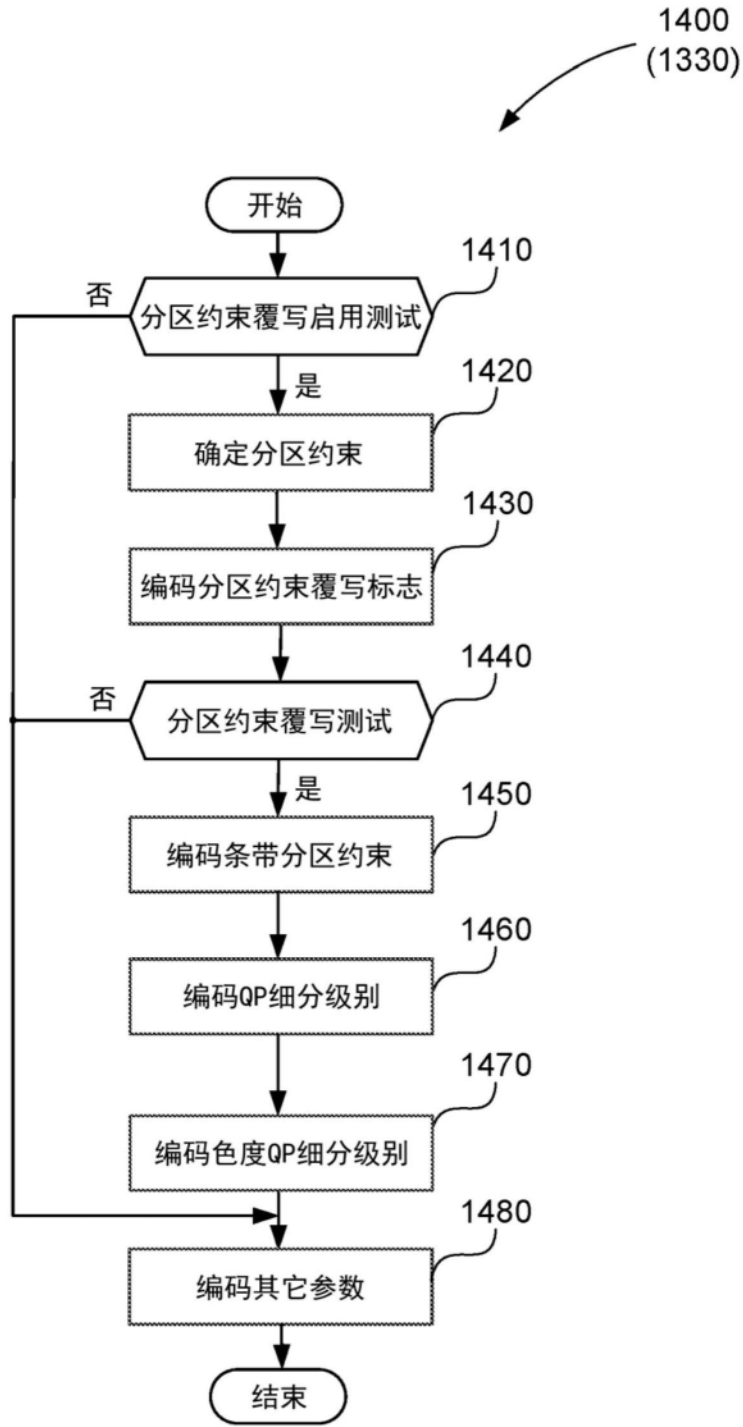


图14

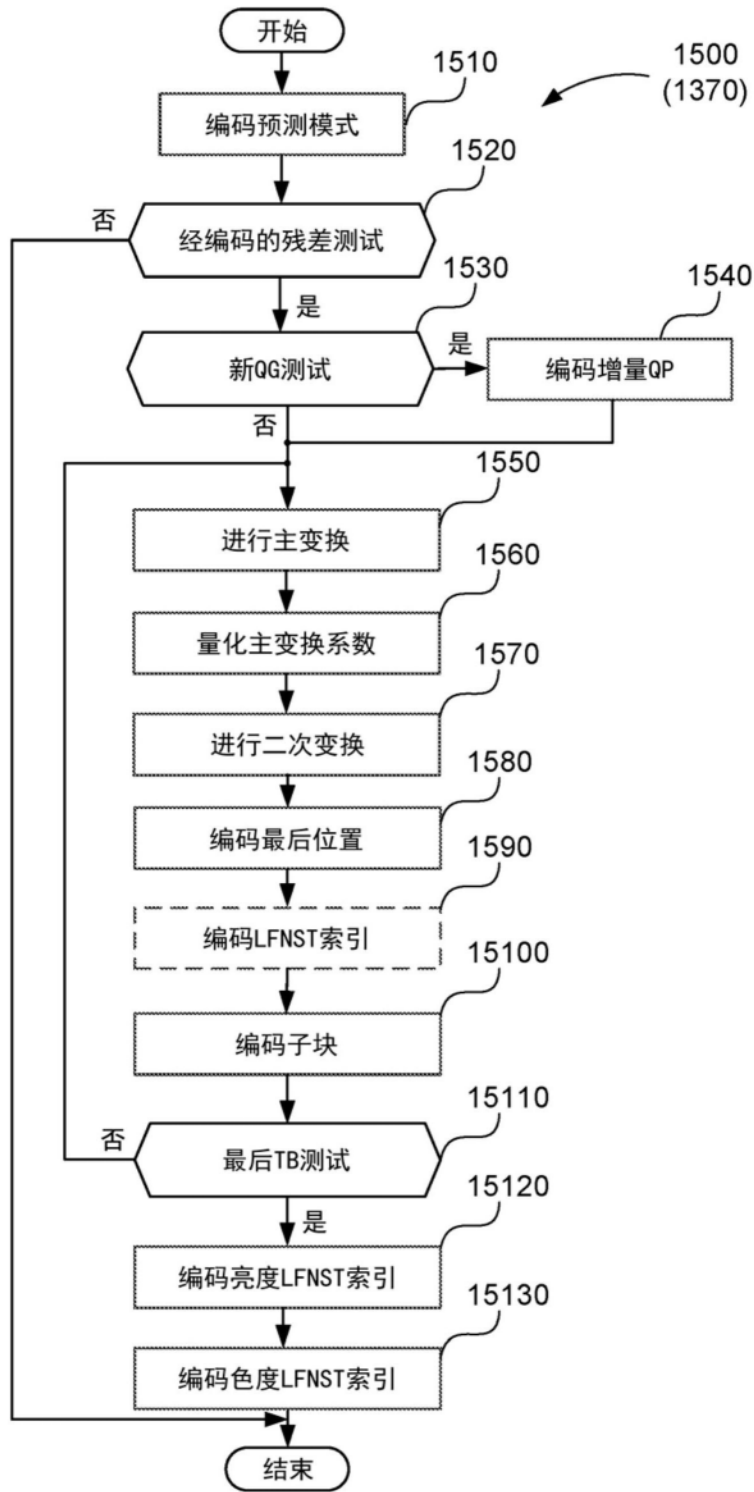


图15

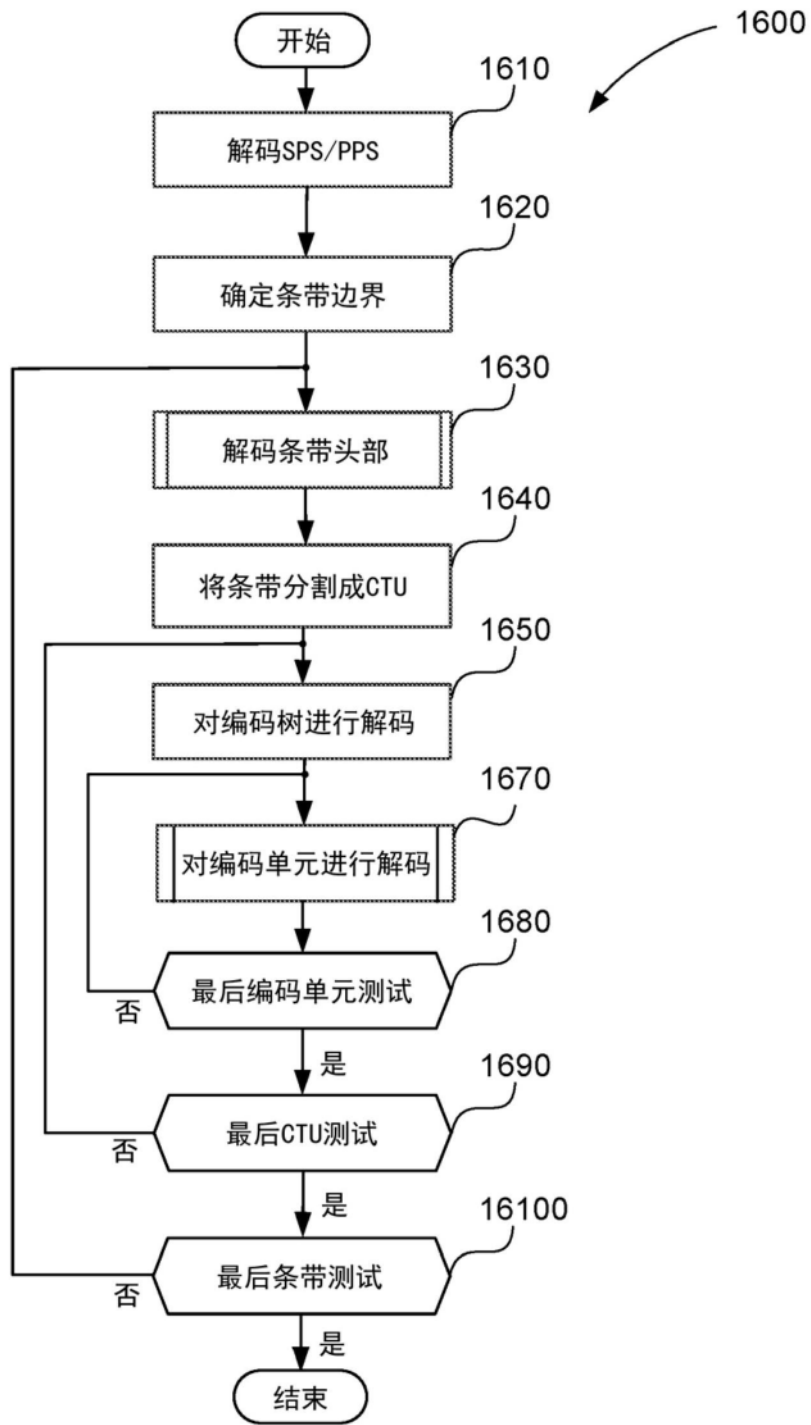


图16

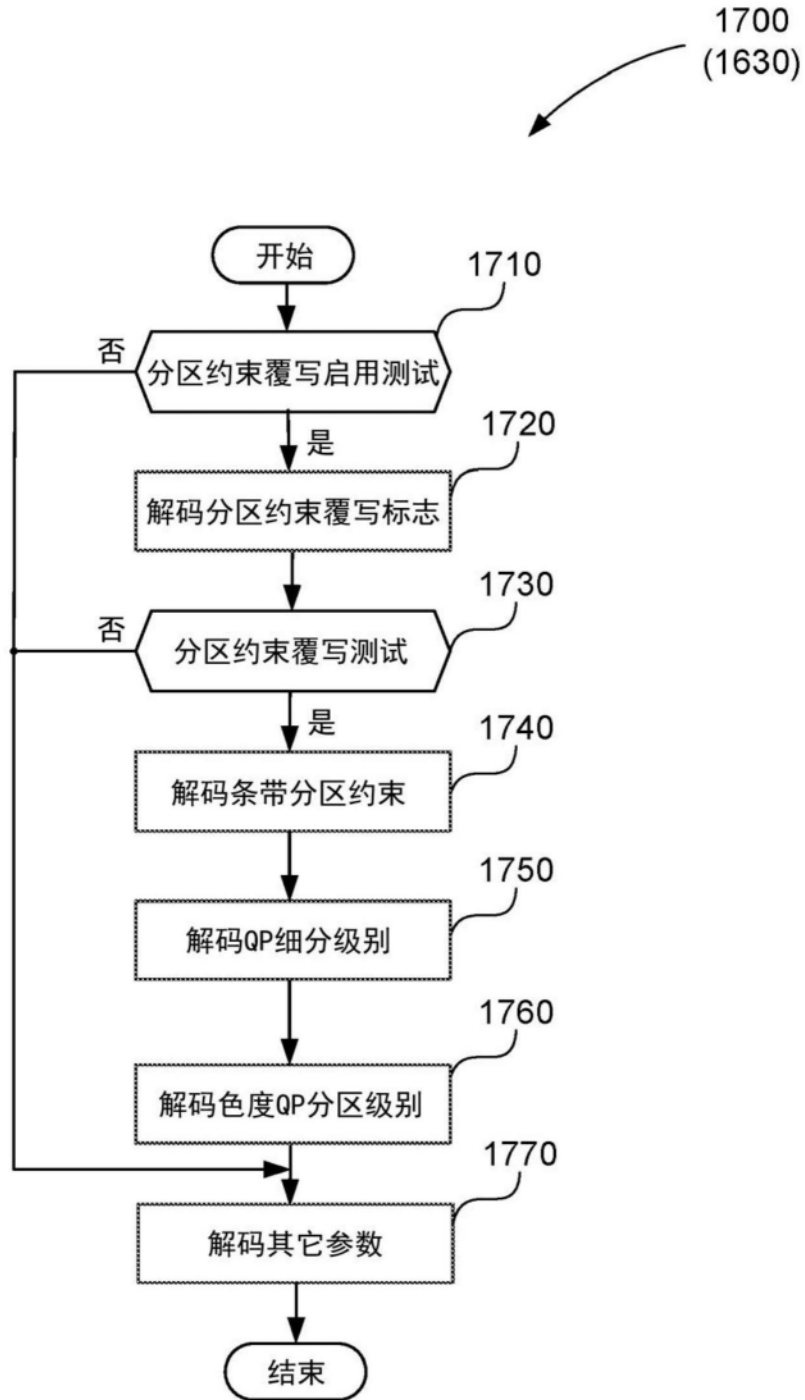


图17

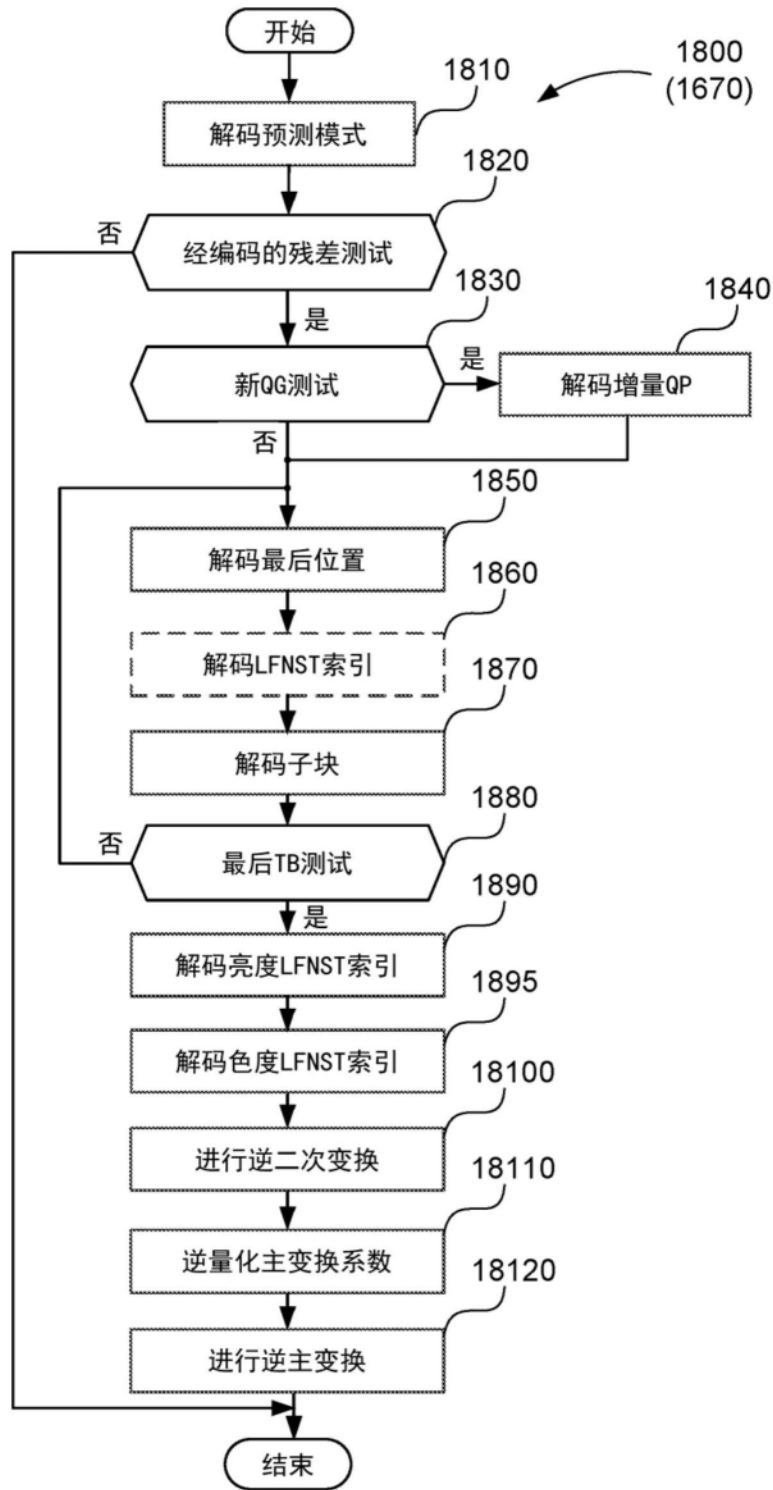


图18

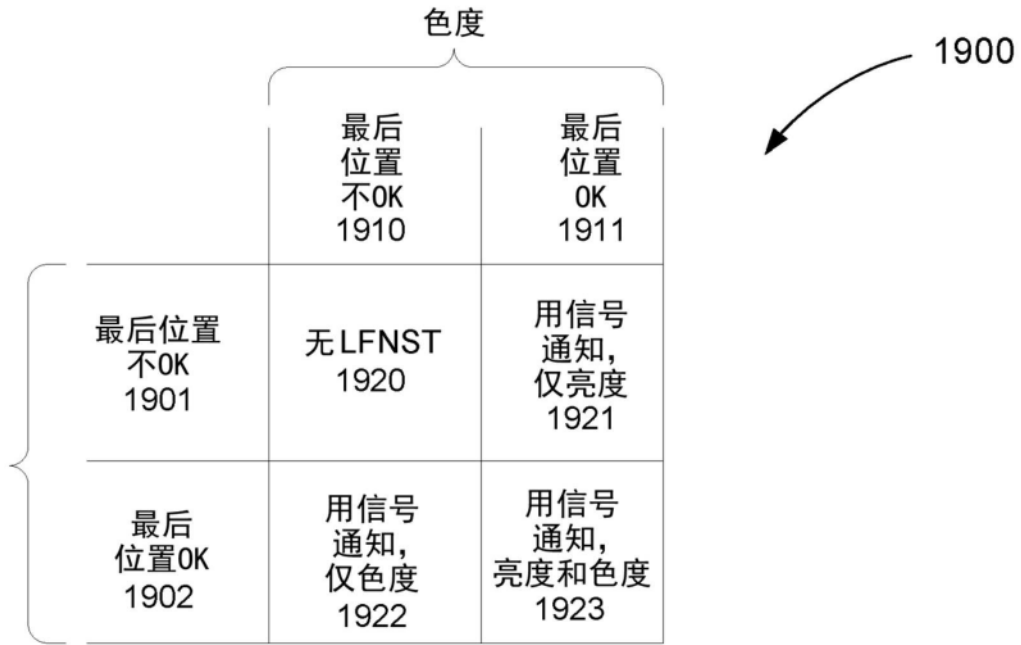


图19A

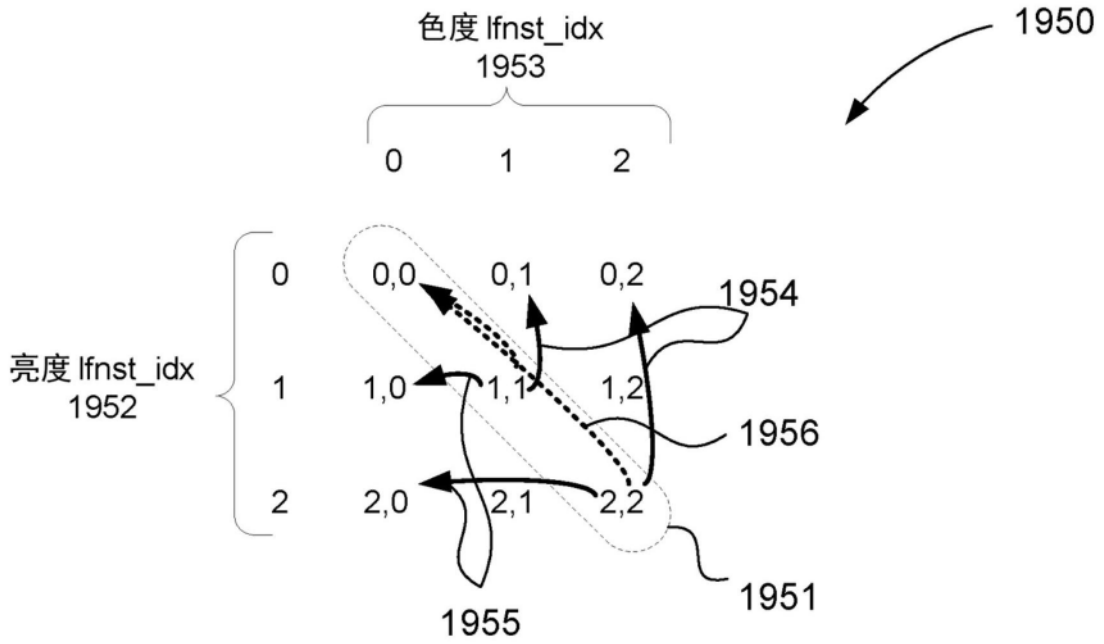


图19B