



(12) 发明专利



(10) 授权公告号 CN 110249629 B

(45) 授权公告日 2024. 06. 14

(21) 申请号 201880009344.5
(22) 申请日 2018.01.15
(65) 同一申请的已公布的文献号
申请公布号 CN 110249629 A
(43) 申请公布日 2019.09.17
(30) 优先权数据
62/452,868 2017.01.31 US
62/465,135 2017.02.28 US
62/466,976 2017.03.03 US
62/478,362 2017.03.29 US
62/491,884 2017.04.28 US
(85) PCT国际申请进入国家阶段日
2019.07.30
(86) PCT国际申请的申请数据
PCT/JP2018/000839 2018.01.15
(87) PCT国际申请的公布数据
W02018/142903 EN 2018.08.09
(73) 专利权人 夏普株式会社
地址 日本国大阪府堺市堺区匠町1番地
专利权人 鸿颖创新有限公司

(72) 发明人 基兰·穆克什·米斯拉 赵杰
克里斯托弗·安德鲁·塞格尔
(74) 专利代理机构 中科专利商标代理有限责任
公司 11021
专利代理师 王波波
(51) Int.Cl.
H04N 19/119 (2006.01)
H04N 19/70 (2006.01)
H04N 19/96 (2006.01)
(56) 对比文件
WO 2013065431 A1,2013.05.10
CHEN,Jianle.Algorithm Description of
Joint Exploration Test Model 3.Joint
Video Exploration Team (JVET) of ITU-T SG
16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 3rd
Meeting: Geneva, CH.2016,第2页第17,19-20
行.
(续)
审查员 奚惠宁

权利要求书1页 说明书39页 附图26页

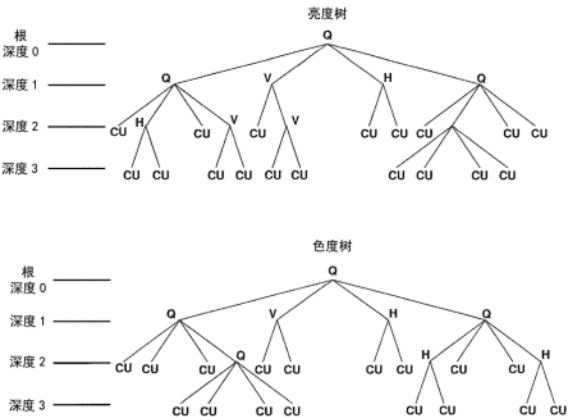
(54) 发明名称

确定对视频数据的分割以进行视频码处理的方法

(57) 摘要

一种确定对视频数据的分割以进行视频码处理的方法。确定分割树中当前节点的深度以及用于生成当前节点的分割的类型。通过以下操作确定将当前节点根据三元分裂进一步分割为三个子节点:对于当前节点,解析用于指示当前节点的分裂类型不是四元分裂的第一二进制值,对于当前节点,解析用于指示当前节点的进一步分割的第二二进制值;对于当前节点,解析用于指示当前节点的进一步分割的方向的第三二进制值;以及如果在特定深度处允许二元分裂,则对于当前节点,解析用于指示当前节点的分裂类型是二元分裂或三元分裂的第四二进制值,如果在

特定深度处不允许二元分裂时,则对于当前节点,推断出指示二叉树分裂的第四二进制值。



[接上页]

(56) 对比文件

LI, Xiang et al.. Multi-Type-Tree. Joint
Video Exploration Team (JVET) of ITU-T SG

16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 4th
Meeting: Chengdu, CN. 2016, 第2.1, 2.2部分,
图3.

1. 一种确定对视频数据的分割以进行视频码处理的方法,所述方法包括:
确定分割树中当前节点的深度以及用于生成当前节点的分割的类型;以及
通过以下操作确定将当前节点根据三元分裂进一步分割为三个子节点:
对于当前节点,解析第一二进制值,所述第一二进制值指示当前节点的分裂类型不是四元分裂,
对于当前节点,解析第二二进制值,所述第二二进制值指示当前节点的进一步分割,
对于当前节点,解析第三二进制值,所述第三二进制值指示当前节点的进一步分割的方向,
对于当前节点,如果在特定深度处允许二元分裂,解析第四二进制值,所述第四二进制值指示当前节点的分裂类型是二元分裂或三元分裂,以及
对于当前节点,如果在特定深度处不允许二元分裂,则推断出指示二叉树分裂的所述第四二进制值。
2. 根据权利要求1所述的方法,其中,所述第三二进制值指示垂直方向。
3. 根据权利要求1所述的方法,其中,所述第三二进制值指示水平方向。

确定对视频数据的分割以进行视频码处理的方法

技术领域

[0001] 本公开涉及视频编码,并且更具体地,涉及用于分割视频数据的图片的技术。

背景技术

[0002] 数字视频功能可以包含在各种设备中,包括数字电视、膝上型计算机或台式计算机、平板计算机、数字记录设备、数字媒体播放器、视频游戏设备、蜂窝电话(包括所谓的智能电话)、医疗成像设备等等。可以根据视频编码标准对数字视频进行编码。视频编码标准可以包含视频压缩技术。视频编码标准的示例包括ISO/IEC MPEG-4 Visual和ITU-T H.264(也称为[ISO/IEC MPEG-4 AVC)和高效视频编码(High-Efficiency Video Coding, HEVC)。HEVC在High Efficiency Video Coding (HEVC), Rec. ITU-T H.265 April 2015中进行了描述,其通过引用合并于此,并且在本文中称为ITU-T H.265。目前正在考虑ITU-T H.265的扩展和改进,以开发下一代视频编码标准。例如,ITU-T视频编码专家组(VCEG)和ISO/IEC(活动图片专家组(MPEG))(统称为联合视频探索组(JVET))正在研究对未来视频编码技术标准化的潜在需求,这种未来视频编码技术的压缩能力远远超过当前HEVC标准的压缩能力。联合探索模型3(JEM 3)、联合探索测试模型3(JEM 3)的算法描述、ISO/IEC JTC1/SC29/WG11文件:JVET-C1001v3, May 2016, Geneva, CH(其通过引用并入本文),将由JVET进行协调测试模型研究的编码特征描述为潜在地增强视频编码技术,使得其超过ITU-T H.265的能力。应该注意的是,JEM 3的编码特征是在Fraunhofer研究组织维护的JEM参考软件中实现的。目前,可以获得更新的JEM参考软件版本3(JEM 3.0)。如本文所使用的,术语JEM用于共同地指代JEM 3中包括的算法和JEM参考软件的实现。

[0003] 视频压缩技术使得能够减少用于存储和传输视频数据的数据需求。视频压缩技术可以通过利用视频序列中的固有冗余来减少数据需求。视频压缩技术可以将视频序列细分成连续的较小部分(即,视频序列内的帧组、帧组内的帧、帧内的片、片内的编码树单元(例如,宏块)、编码树单元内的编码块等)。帧内预测编码技术(例如,图片内(空间))和帧间预测技术(即,图片间(时间))可用于生成待编码视频数据的单元与视频数据的参考单元之间的差值。差值可以称为残差数据。可以将残差数据编码为量化变换系数。语法元素可以涉及残差数据和参考编码单元(例如,帧内预测模式索引、运动矢量和块矢量)。可以对残差数据和语法元素进行熵编码。经熵编码的残差数据和语法元素可以包括在兼容的比特流中。

发明内容

[0004] 在一个示例中,分割视频数据以进行视频码处理的方法包括:接收视频块,所述视频块包括视频数据的第一分量的样本值和视频数据的第二分量的样本值;根据第一四叉树二叉树分割结构来分割视频数据的所述第一分量的样本值;以及根据所述第一四叉树二叉树分割结构来分割视频数据的所述第二分量的样本值直至共享深度为止。

[0005] 在一个示例中,确定对视频数据的分割以进行视频码处理的方法包括:解析第一四叉树二叉树分割结构;将所述第一四叉树二叉树分割结构应用于视频数据的第一分量;

确定共享深度;以及将所述第一四叉树二叉树分割结构应用于视频数据的第二分量直至所述共享深度为止。

[0006] 在一个示例中,分割视频数据的叶节点以进行视频码处理的方法包括:确定偏移值;以及根据所述偏移值来分割所述叶节点。

[0007] 在一个示例中,分割视频数据的节点以进行视频码处理的方法包括:确定分割类型;确定与所述分割类型相对应的一个或多个偏移值;以及根据所述一个或多个偏移值来分割所述节点。

[0008] 在一个示例中,分割视频数据的节点以进行视频码处理的方法包括:根据对角分割形状来分割所述节点;以及确定从所述对角分割形状得到的每个分割的预测模式。

[0009] 在一个示例中,确定对视频数据的分割以进行视频码处理的方法包括:确定分割树中当前节点的深度以及用于生成当前节点的分割的类型;以及通过以下操作确定将当前节点根据三元分裂进一步分割为三个子节点:对于当前节点,解析第一二进制值,所述第一二进制值指示当前节点的分裂类型不是四元分裂,对于当前节点,解析第二二进制值,所述第二二进制值指示当前节点的进一步分割,对于当前节点,解析第三二进制值,所述第三二进制值指示当前节点的进一步分割的方向,对于当前节点,如果在特定深度处允许二元分裂,解析第四二进制值,所述第四二进制值指示当前节点的分裂类型是二元分裂或三元分裂,以及对于当前节点,如果在特定深度处不允许二元分裂,则推断出指示二叉树分裂的所述第四二进制值。

附图说明

[0010] 图1是示出了根据本公开的一个或多个技术的根据四叉树二叉树分割而编码的图片组的示例的概念图。

[0011] 图2是示出了根据本公开的一个或多个技术的四叉树二叉树的示例的概念图。

[0012] 图3是示出了根据本公开的一个或多个技术的视频分量四叉树二叉树分割的概念图。

[0013] 图4是示出了根据本公开的一个或多个技术的视频分量采样格式的示例的概念图。

[0014] 图5是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。

[0015] 图6A是示出了根据本公开的一个或多个技术的对视频数据块进行码处理的示例的概念图。

[0016] 图6B是示出了根据本公开的一个或多个技术的对视频数据块进行码处理的示例的概念图。

[0017] 图7是示出了根据本公开的一个或多个技术的可以被配置为对视频数据进行编码和解码的系统的示例的框图。

[0018] 图8是示出了根据本公开的一个或多个技术的可以被配置为对视频数据进行编码的视频编码器的示例的框图。

[0019] 图9是示出了根据本公开的一个或多个技术的视频分量四叉树二叉树分割的概念图。

- [0020] 图10是示出了根据本公开的一个或多个技术的视频分量四叉树二叉树分割的概念图。
- [0021] 图11是示出了根据本公开的一个或多个技术的四叉树二叉树的示例的概念图。
- [0022] 图12是示出了根据本公开的一个或多个技术的四叉树二叉树分割的概念图。
- [0023] 图13是示出了根据本公开的一个或多个技术的四叉树二叉树分割的概念图。
- [0024] 图14是示出了根据本公开的一个或多个技术的可以被配置为对视频数据进行解码的视频解码器的示例的框图。
- [0025] 图15是示出了根据本公开的一个或多个技术的分割的概念图。
- [0026] 图16是示出了根据本公开的一个或多个技术的分割的概念图。
- [0027] 图17是示出了根据本公开的一个或多个技术的分割的概念图。
- [0028] 图18是示出了根据本公开的一个或多个技术的分割的概念图。
- [0029] 图19A是示出了根据本公开的一个或多个技术的分割的概念图。
- [0030] 图19B是示出了根据本公开的一个或多个技术的分割的概念图。
- [0031] 图19C是示出了根据本公开的一个或多个技术的分割的概念图。
- [0032] 图20是示出了根据本公开的一个或多个技术的分割的概念图。
- [0033] 图21A是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。
- [0034] 图21B是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。
- [0035] 图22A是示出了根据本公开的一个或多个技术的分割的概念图。
- [0036] 图22B是示出了根据本公开的一个或多个技术的分割的概念图。
- [0037] 图22C是示出了根据本公开的一个或多个技术的分割的概念图。
- [0038] 图22D是示出了根据本公开的一个或多个技术的分割的概念图。
- [0039] 图23A是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。
- [0040] 图23B是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。
- [0041] 图23C是示出了根据本公开的一个或多个技术的用于视频数据块的可能编码结构的概念图。

具体实施方式

[0042] 总体上,本公开描述了用于对视频数据进行编码的各种技术。具体地,本公开描述了用于分割视频数据的图片的技术。应当注意,尽管关于ITU-T H.264、ITU-T H.265和JEM描述了本公开的技术,但是本公开的技术一般地适用于视频编码。例如,本文描述的编码技术可以并入视频编码系统(包括基于未来视频编码标准的视频编码系统)中,该视频编码系统包括除ITU-T H.265和JEM中所包括的之外的块结构、帧内预测技术、帧间预测技术、变换技术、滤波技术和/或熵编码技术。因此,对ITU-T H.264、ITU-T H.265和/或JEM的引用是出于描述的目的,不应被解释为限制本文描述的技术的范围。此外,应当注意,通过引用并入本文的文档是出于描述的目的,并且不应被解释为限制本文使用的术语或产生关于本文使

用的术语的歧义。例如,在并入的参考文献提供与另一并入的参考文献和/或本文中使用的术语不同的术语定义的情况下,该术语应该以宽泛地包括每个相应的定义的方式和/或替代地包括每个特定定义的方式来进行解释。

[0043] 在一个示例中,分割视频数据以进行视频码处理的设备包括一个或多个处理器,该一个或多个处理器被配置为:接收视频块,所述视频块包括视频数据的第一分量的样本值和视频数据的第二分量的样本值,根据第一四叉树二叉树分割结构来分割视频数据的第一分量的样本值,根据该第一四叉树二叉树分割结构来分割视频数据的第二分量的样本值直至共享深度为止。

[0044] 在一个示例中,非暂时性计算机可读存储介质包括存储在其上的指令,该指令在被执行时,使得设备的一个或多个处理器:接收视频块,所述视频块包括视频数据的第一分量的样本值和视频数据的第二分量的样本值,根据第一四叉树二叉树分割结构来分割视频数据的第一分量的样本值,根据该第一四叉树二叉树分割结构来分割视频数据的第二分量的样本值直至共享深度为止。

[0045] 在一个示例中,装置包括:用于接收视频块的模块,所述视频块包括视频数据的第一分量的样本值和视频数据的第二分量的样本值;根据第一四叉树二叉树分割结构来分割视频数据的第一分量的样本值的模块;以及根据该第一四叉树二叉树分割结构来分割视频数据的第二分量的样本值直至共享深度为止的模块。

[0046] 在一个示例中,重构视频数据的方法包括:解析第一四叉树二叉树分割结构,将该第一四叉树二叉树分割结构应用于视频数据的第一分量,确定共享深度,以及将该第一四叉树二叉树分割结构应用于视频数据的第二分量直至该共享深度为止。

[0047] 在一个示例中,用于重构视频数据的设备包括一个或多个处理器,该一个或多个处理器被配置为:解析第一四叉树二叉树分割结构,将该第一四叉树二叉树分割结构应用于视频数据的第一分量,确定共享深度,以及将该第一四叉树二叉树分割结构应用于视频数据的第二分量直至该共享深度为止。

[0048] 在一个示例中,非暂时性计算机可读存储介质包括存储在其上的指令,该指令在被执行时,使得设备的一个或多个处理器:解析第一四叉树二叉树分割结构,将该第一四叉树二叉树分割结构应用于视频数据的第一分量,确定共享深度,以及将该第一四叉树二叉树分割结构应用于视频数据的第二分量直至该共享深度为止。

[0049] 在一个示例中,装置包括:用于解析第一四叉树二叉树分割结构的模块;用于将该第一四叉树二叉树分割结构应用于视频数据的第一分量的模块;用于确定共享深度的模块;以及用于将该第一四叉树二叉树分割结构应用于视频数据的第二分量直至该共享深度为止的模块。

[0050] 在一个示例中,分割视频数据的叶节点以进行视频码处理的设备包括一个或多个处理器,该一个或多个处理器被配置为确定偏移值并根据该偏移值来分割叶节点。

[0051] 在一个示例中,非暂时性计算机可读存储介质包括存储在其上的指令,该指令在被执行时使得设备的一个或多个处理器确定偏移值并根据该偏移值来分割叶节点。

[0052] 在一个示例中,装置包括:用于确定偏移值的模块和用于根据该偏移值来分割叶节点的模块。

[0053] 在附图和以下描述中阐述了一个或多个示例的细节。其他特征、目的和优点将通

过说明书和附图以及权利要求而变得显而易见。

[0054] 视频内容通常包括由一系列帧(或图片)组成的视频序列。一系列帧也可以称为图片组(GOP)。每个视频帧或图片可包括多个片(slice)或图块(tile),其中片或图块包括多个视频块。如本文所使用的,术语视频块可以一般地指代图片的区域,或者可以更具体地指代可以被预测编码的最大样本值阵列、其细分部分和/或对应结构。此外,术语当前视频块可以指代正被编码或解码的图片区域。视频块可以被定义为可以预测编码的样本值阵列。应当注意,在一些情况下,像素值可以被描述为包括视频数据的各个分量的样本值,该各个分量也可以被称为颜色分量(例如,亮度(Y)和色度(Cb和Cr)分量或者红色分量、绿色分量和蓝色分量)。应注意,在一些情况下,术语像素值和样本值可互换使用。可以根据扫描模式(例如,光栅扫描)在图片内对视频块进行排序。视频编码器可对视频块及其细分部分执行预测编码。视频块及其细分部分可以称为节点。

[0055] ITU-T H.264规定了包括 16×16 个亮度样本的宏块。也就是说,在ITU-T H.264中,图片被分段为宏块。ITU-T H.265规定了类似的编码树单元(CTU)结构。在ITU-T H.265中,图片被分段为CTU。在ITU-T H.265中,对于图片,CTU大小可以被设置为包括 16×16 、 32×32 或 64×64 个亮度样本。在ITU-T H.265中,CTU由用于视频数据的每个分量(例如,亮度(Y)和色度(Cb和Cr))的相应编码树块(CTB)组成。此外,在ITU-T H.265中,可以根据四叉树(QT)分割结构来分割CTU,这导致CTU的CTB被分割为编码块(CB)。也就是说,在ITU-T H.265中,CTU可以被分割为四叉树叶节点。根据ITU-T H.265,将一个亮度CB与两个对应的色度CB和相关联的语法元素一起称为编码单元(CU)。在ITU-T H.265中,可以用信号通知CB的最小允许大小。在ITU-T H.265中,亮度CB的最小的最小允许大小是 8×8 个亮度样本。在ITU-T H.265中,使用帧内预测或帧间预测对图片区域进行编码的决定是在CU级别进行的。

[0056] 在ITU-T H.265中,CU与预测单元(PU)结构相关联,该结构的根在CU处。在ITU-T H.265中,PU结构允许分裂亮度和色度CB,以便产生相应的参考样本。也就是说,在ITU-T H.265中,亮度和色度CB可以被分裂成亮度和色度预测块(PB),其中PB包括应用相同预测的样本值的块。在ITU-T H.265中,CB可以被分割为1、2或4个PB。ITU-T H.265支持从 64×64 个样本到 4×4 个样本的PB大小。在ITU-T H.265中,支持方形PB用于帧内预测,其中CB可以形成PB或CB可以被分裂成四个方形PB(即,帧内预测PB类型包括 $M \times M$ 或 $M/2 \times M/2$,其中M是方形CB的高度和宽度)。在ITU-T H.265中,除了方形PB之外,还支持矩形PB用于帧间预测,其中CB可以垂直或水平地减半以形成PB(即,帧间预测PB类型包括 $M \times M$ 、 $M/2 \times M/2$ 、 $M/2 \times M$ 或 $M \times M/2$)。此外,应该注意的是,在ITU-T H.265中,对于帧间预测,支持四个非对称PB分割,其中CB在CB的高度(在顶部或底部)或宽度(在左侧或右侧)的四分之一处被分割为两个PB(即,非对称分割包括 $M/4 \times M$ 左部、 $M/4 \times M$ 右部、 $M \times M/4$ 顶部和 $M \times M/4$ 底部)。与PB相对应的帧内预测数据(例如,帧内预测模式语法元素)或帧间预测数据(例如,运动数据语法元素)用于产生PB的参考和/或预测样本值。

[0057] JEM规定CTU的最大大小为 256×256 个亮度样本。JEM规定四叉树加二叉树(QTBT)块结构。在JEM中,QTBT结构使得四叉树叶节点能够通过二叉树(BT)结构进一步分割。也就是说,在JEM中,二叉树结构使得四叉树叶节点能够垂直或水平地递归划分。图1示出了CTU(例如,大小为 256×256 个亮度样本的CTU)被分割成四叉树叶节点并且四叉树叶节点根据二叉树被进一步分割的示例。也就是说,在图1中,虚线指示四叉树中的附加二叉树分割。因

此, JEM中的二叉树结构启用方形和矩形叶节点, 其中每个叶节点包括CB。如图1所示, 包括在GOP中的图片可以包括片, 其中每个片包括CTU序列, 并且每个CTU可以根据QTBT结构来分割。图1示出了片中包括的一个CTU的QTBT分割的示例。图2是示出了与图1中所示的示例QTBT分割相对应的QTBT的示例的概念图。

[0058] 在JEM中, 通过用信号通知QT分裂标志和BT分裂模式语法元素来用信号通知QTBT。当QT分裂标志具有值1时, 指示QT分裂。当QT分裂标志具有值0时, 用信号通知BT分裂模式语法元素。当BT分裂模式语法元素具有值0 (即, BT分裂模式编码树=0) 时, 指示没有二元分裂(binary splitting)。当BT分裂模式语法元素具有值1 (即, BT分裂模式编码树=11) 时, 指示垂直分裂模式。当BT分裂模式语法元素具有值2 (即, BT分裂模式编码树=10) 时, 指示水平分裂模式。此外, 可以执行BT分裂直到达到最大BT深度为止。因此, 根据JEM, 可以基于表1中提供的伪语法来用信号通知图2中所示的QTBT:

[0059]

```

QT flag = 1; //深度 0 语法
    QT flag = 1; //深度 1 语法
        QT flag = 0, BT split = 0; //深度 2 语法
        QT flag = 0, BT split = 2; //深度 2 语法
            BT split = 0; //深度 3 语法
            BT split = 0; //深度 3 语法
        QT flag = 0, BT split = 0; //深度 2 语法
        QT flag = 0, BT split = 1; //深度 2 语法
            BT split = 0; //深度 3 语法
            BT split = 0; //深度 3 语法
    QT flag = 0; BT split = 1; //深度 1 语法
        BT split = 0; //深度 2 语法
        BT split = 1; //深度 2 语法
            BT split = 0; //深度 3 语法
            BT split = 0; //深度 3 语法
    QT flag = 0; BT split = 2; //深度 1 语法
        BT split = 0; //深度 2 语法
        BT split = 0; //深度 2 语法
    QT flag = 1; //深度 1 语法
        QT flag = 0, BT split = 0; //深度 2 语法
        QT flag = 1; //深度 2 语法
            QT flag = 0, BT split = 0; //深度 3 语法
            QT flag = 0, BT split = 0; //深度 3 语法
            QT flag = 0, BT split = 0; //深度 3 语法
            QT flag = 0, BT split = 0; //深度 3 语法
        QT flag = 0, BT split = 0; //深度 2 语法
        QT flag = 0, BT split = 0; //深度 2 语法

```

[0060] 表1

[0061] 在一个示例中,当达到最大QT深度时,可以跳过QT标志的信令,并且可以推断其值例如为0。在一个示例中,当当前深度小于最小QT深度时,可以跳过QT标志的信令,并且可以推断其值例如为1。在一个示例中,当达到用于分割类型的信令的最大深度时,可以不在比特流中用信号通知关联的语法元素,并且可以推断其值。在一个示例中,当尚未达到用于分割类型的信令的最小深度时,可以不在比特流中用信号通知关联的语法元素,并且可以推断其值。在一个示例中,当不允许QT分裂并且当前深度小于最小BT深度时,可以修改BT分裂的信令以不允许BT分裂等于0。

[0062] 在一个示例中,以下树遍历可以用于用信号通知分裂决策。例如:

[0063] 1、用信号通知当前节点的分裂决策

[0064] 2、针对 $i=1$ 至(步骤1中的)当前节点的子节点数,执行以下动作:

[0065] a) 确定与 i 对应的子节点 n (这可以基于查找,即基于当前节点的分裂模式)

[0066] b) 递归地调用遍历函数,遍历根在子节点 n 处的子树。

[0067] 在一个示例中,以下树遍历可以用于用信号通知分裂决策。例如:

[0068] 1、针对 $i=1$ 至(步骤1中的)当前节点的子节点数,执行以下动作:

[0069] a) 确定与 i 对应的子节点 n (这可以基于查找,即基于当前节点的分裂模式)

[0070] b) 递归地调用遍历函数,遍历根在子节点 n 处的子树

[0071] c) 用信号通知当前节点的分裂决策。

[0072] 在示例中,以下树遍历可以用于用信号通知分裂决策。例如:

[0073] 1、针对 $i=1$ 至(步骤1中的)当前节点的子节点数,执行以下动作:

[0074] a) 确定与 i 对应的子节点 n (这可以基于查找,即基于当前节点的分裂模式)

[0075] b) 递归地调用遍历函数,遍历根在子节点 n 处的子树

[0076] 2、用信号通知当前节点的分裂决策。

[0077] 在一个示例中,可以以深度增加的方式遍历树。在这种情况下,可以在前进到下一深度之前用信号通知特定深度处的节点的所有分裂决策。

[0078] 如图2和表1中所示,QT分裂标志语法元素和BT分裂模式语法元素与深度相关联,其中零深度对应于QTBT的根,而较高值深度对应于根之外的随后的深度。此外,在JEM中,亮度和色度分量可以具有单独的QTBT分割。也就是说,在JEM中,可以通过用信号通知相应的QTBT来独立地分割亮度和色度分量。图3示出了根据用于亮度分量的QTBT和用于色度分量的独立QTBT分割CTU的示例。如图3所示,当独立QTBT用于分割CTU时,亮度分量的CB无需并且不一定与色度分量的CB对齐。目前,在JEM中,对于使用帧内预测技术的片,启用独立的QTBT结构。应当注意,在一些情况下,可能需要从相关的亮度变量值导出色度变量的值。在这些情况下,色度中的样本位置和色度格式可用于确定亮度中的相应样本位置,以确定相关联的亮度变量值。

[0079] 另外,应该注意,JEM包括用于用信号通知QTBT树的以下参数:

[0080] CTU大小:四叉树的根节点大小(例如, 256×256 、 128×128 、 64×64 、 32×32 、 16×16 亮度样本);

[0081] MinQTSIZE:最小允许四叉树叶节点大小(例如, 16×16 、 8×8 个亮度样本);

[0082] MaxBTSIZE:最大允许二叉树根节点大小,即可以通过二元分裂来分割的叶四叉树节点的最大大小(例如, 64×64 个亮度样本);

[0083] MaxBTDepth:最大允许二叉树深度,即可以发生二元分裂的最低级别,其中四叉树叶节点是根(例如,3);

[0084] MinBTSIZE:最小允许二叉树叶节点大小;即二叉叶节点的最小宽度或高度(例如,4个亮度样本)。

[0085] 应当注意,在一些示例中,MinQTSIZE、MaxBTSIZE、MaxBTDepth和/或MinBTSIZE对于视频的不同分量可以是不同的。

[0086] 在JEM中,CB用于预测而无需进一步分割。也就是说,在JEM中,CB可以是应用相同

预测的样本值块。因此, JEM QTBT叶节点可以类似于ITU-T H.265中的PB。

[0087] 视频采样格式(也可以被称为色度格式)可以相对于包括在CU中的亮度样本的数量来定义包括在CU中的色度样本的数量。例如,对于4:2:0采样格式,对于水平和垂直方向两者,亮度分量的采样率是色度分量的采样率的两倍。结果,对于根据4:2:0格式进行格式化的CU,亮度分量的样本阵列的宽度和高度是色度分量的每个样本阵列的宽度和高度的两倍。图4是示出了根据4:2:0样本格式进行了格式化的编码单元的示例的概念图。图4示出了CU内的色度样本相对于亮度样本的相对位置。如上所述,通常根据水平和垂直亮度样本的数量来定义CU。因此,如图4所示,根据4:2:0样本格式进行了格式化的 16×16 CU包括亮度分量的 16×16 个样本和每个色度分量的 8×8 个样本。此外,在图4所示的示例中,示出了色度样本相对于与 16×16 CU相邻的视频块的亮度样本的相对位置。对于根据4:2:2格式进行了格式化的CU,亮度分量的样本阵列的宽度是每个色度分量的样本阵列宽度的两倍,但是亮度分量的样本阵列的高度等于每个色度分量的样本阵列的高度。此外,对于根据4:4:4格式进行了格式化的CU,亮度分量的样本阵列具有与每个色度分量的样本阵列相同的宽度和高度。

[0088] 如上所述,帧内预测数据或帧间预测数据用于产生样本值块的参考样本值。包括在当前PB或另一类型的图片区域结构中的样本值与相关联的参考样本(例如,使用预测生成的那些参考样本)之间的差可以被称为残差数据。残差数据可以包括与视频数据的每个分量相对应的各个差值阵列。残差数据可以在像素域中。诸如离散余弦变换(DCT)、离散正弦变换(DST)、整数变换、小波变换或概念上类似的变换之类的变换可以应用于差值阵列以生成变换系数。应当注意,在ITU-T H.265中,CU与其根在CU级别的变换单元(TU)结构相关联。也就是说,在ITU-T H.265中,可以对差值阵列进行细分,以生成变换系数(例如,可以将四个 8×8 变换应用于 16×16 残差值阵列)。对于视频数据的每个分量,差值的这种细分部分可以称为变换块(TB)。应当注意,在ITU-T H.265中,TB不一定与PB对齐。图5示出了可用于编码特定CB的备选PB和TB组合的示例。此外,应该注意的是,在ITU-T H.265中,TB可以具有以下大小: 4×4 、 8×8 、 16×16 和 32×32 。

[0089] 应当注意,在JEM中,使用对应于CB的残差值以生成变换系数而无需进一步分割。也就是说,在JEM中,QTBT叶节点可以类似于ITU-T H.265中的PB和TB两者。应当注意,在JEM中,可以应用核心变换和随后的二次变换(在视频编码器中)以生成变换系数。对于视频解码器,变换的顺序是相反的。此外,在JEM中,是否应用二次变换以生成变换系数可以取决于预测模式。

[0090] 可以对变换系数执行量化处理。量化对变换系数进行缩放,以便改变表示一组变换系数所需的数据量。量化可以包括将变换系数除以量化比例因子并可以包括任何相关的舍入函数(例如,舍入到最近的整数)。量化的变换系数可以称为系数级别值。逆量化(或“反量化”)可以包括将系数级别值乘以量化比例因子。应当注意,如本文所使用的,在一些情况下,术语量化处理可以指除以比例因子以生成级别值,并且在一些情况下,术语量化处理可以指乘以比例因子以恢复变换系数。也就是说,量化处理在一些情况下可以指量化并且在一些情况下可以指逆量化。此外,应当注意,尽管在下面的示例中,关于与十进制符号相关联的算术运算描述了量化处理,但是这种描述是出于说明性目的而不应被解释为限制。例如,本文描述的技术可以使用二进制运算等在设备中实现。例如,本文描述的乘法和除法运

算可以使用比特移位运算等来实现。

[0091] 图6A至图6B是示出了对视频数据块进行码处理的示例的概念图。如图6A所示,通过从当前视频数据块中减去预测值集合以生成残差、对残差执行变换、并对变换系数进行量化以生成级别值,来对当前视频数据块(例如,与视频分量相对应的CB)进行编码。如图6B所示,通过对级别值执行逆量化、执行逆变换、以及将预测值集合加到所得到的残差来对当前视频数据块进行解码。应当注意,在图6A至图6B的示例中,重构块的样本值不同于编码的当前视频块的样本值。这样,可以认为编码是有损的。然而,对于重构视频的观看者而言,可以认为样本值的差是可接受的或不可察觉的。此外,如图6A至图6B所示,使用比例因子阵列执行缩放。

[0092] 在ITU-T H.265中,通过选择缩放矩阵并将缩放矩阵中的每一项乘以量化比例因子来生成比例因子阵列。在ITU-T H.265中,基于预测模式和颜色分量选择缩放矩阵,其中定义了以下大小的缩放矩阵: 4×4 、 8×8 、 16×16 和 32×32 。因此,应该注意,ITU-T H.265没有定义除 4×4 、 8×8 、 16×16 和 32×32 以外的大小的缩放矩阵。在ITU-T H.265中,量化比例因子的值可以由量化参数QP确定。在ITU-T H.265中,QP可以取从0到51的52个值,QP的1的变化通常对应于量化比例因子值的约12%的变化。此外,在ITU-T H.265中,可以使用预测量化参数值(其可以被称为预测QP值或QP预测值)和可选地用信号通知的量化参数增量值(其可以被称为QP增量值或增量QP值)来导出变换系数集合的QP值。在ITU-T H.265中,可以针对每个CU更新量化参数,并且可以针对亮度(Y)和色度(Cb和Cr)分量中的每一个来导出量化参数。

[0093] 如图6A所示,量化的变换系数被编码成比特流。可以根据熵编码技术对量化的变换系数和语法元素(例如,指示视频块的编码结构的语法元素)进行熵编码。熵编码技术的示例包括内容自适应可变长度编码(CAVLC)、上下文自适应二进制算术编码(CABAC)、概率区间分割熵编码(PIPE)等。熵编码的量化变换系数和对应的熵编码语法元素可以形成可以用于在视频解码器处再现视频数据的兼容比特流。熵编码过程可以包括对语法元素执行二值化。二值化是指将语法值的值转换为一个或多个比特的序列的过程。这些比特可以称为“二进制数(bin)”。二值化是无损过程,并且可以包括以下编码技术中的一种或其组合:固定长度编码、一元编码、截断一元编码、截断Rice编码、Golomb编码、k阶指数Golomb编码和Golomb-Rice编码。例如,二值化可以包括使用8比特固定长度二值化技术将语法元素的整数值5表示为00000101,或者使用一元编码二值化技术将整数值5表示为11110。如本文所使用的,术语固定长度编码、一元编码、截断一元编码、截断Rice编码、Golomb编码、k阶指数Golomb编码和Golomb-Rice编码中的每个可以指这些技术的一般实现和/或这些编码技术的更具体的实现。例如,可以根据视频编码标准(例如,ITU-T H.265)具体定义Golomb-Rice编码实现。熵编码过程还包括使用无损数据压缩算法对二进制数值进行编码。在CABAC的示例中,对于特定二进制数,可以从与二进制数相关联的可用上下文模型集合中选择上下文模型。在一些示例中,可以基于先前的二进制数和/或先前语法元素的值来选择上下文模型。上下文模型可以识别具有特定值的二进制数的概率。例如,上下文模型可以指示编码0值二进制数的概率为0.7和编码1值二进制数的概率为0.3。应当注意,在一些情况下,编码0值二进制数的概率和编码1值二进制数的概率之和可能不等于1。在选择可用的上下文模型之后,CABAC熵编码器可以基于所识别的上下文模型对二进制数进行算术编码。可以基于编

码二进制数的值来更新上下文模型。可以基于与上下文一起存储的关联变量,例如自适应窗口大小、使用上下文编码的二进制数的数量,来更新上下文模型。应当注意,根据ITU-T H.265,可以实现CABAC熵编码器,使得可以使用算术编码对一些语法元素进行熵编码,而无需使用明确分配的上下文模型,这种编码可以被称为旁路编码。

[0094] 如上所述,帧内预测数据或帧间预测数据可以将图片的区域(例如,PB或CB)与对应的参考样本相关联。对于帧内预测编码,帧内预测模式可指定图片内的参考样本的位置。在ITU-T H.265中,定义的可能帧内预测模式包括平面(即,表面拟合)预测模式(predMode:0)、DC(即,平的总体平均)预测模式(predMode:1)和33个角度预测模式(predMode:2-34)。在JEM中,定义的可能帧内预测模式包括平面预测模式(predMode:0)、DC预测模式(predMode:1)和65个角度预测模式(predMode:2-66)。应当注意,平面预测模式和DC预测模式可以被称为非定向预测模式,并且角度预测模式可以被称为定向预测模式。应当注意,本文描述的技术可以是普遍适用的,而不管定义的可能预测模式的数量。

[0095] 对于帧间预测编码,运动矢量(MV)标识与要编码的视频块的图片不同的图片中的参考样本,从而利用视频中的时间冗余。例如,可以根据位于先前编码帧中的参考块来预测当前视频块,并且可以使用运动矢量来指示参考块的位置。运动矢量和相关数据可以描述例如运动矢量的水平分量、运动矢量的垂直分量、运动矢量的分辨率(例如,四分之一像素精度、二分之一像素精度、一个像素精度、双像素精度、四像素精度)、预测方向和/或参考图片索引值。此外,编码标准(例如,ITU-T H.265等)可以支持运动矢量预测。运动矢量预测使得能够使用相邻块的运动矢量来指定运动矢量。运动矢量预测的示例包括高级运动矢量预测(AMVP)、时间运动矢量预测(TMVP)、所谓的“合并”模式、以及“跳过”和“直接”运动推断。此外,JEM支持高级时间运动矢量预测(ATMVP)和空间-时间运动矢量预测(STMVP)。

[0096] 如上所述,在JEM中,QTBT叶节点(其允许任意矩形CB)可以类似于ITU-T H.265中的PB和TB两者。因此,在一些情况下,JEM在可能的PB和TB结构方面提供的灵活性可能低于ITU-T H.265中提供的灵活性。如上面进一步描述的,在ITU-T H.265中,仅允许方形TB,并且仅允许方形PB用于帧内预测。因此,ITU-T H.265中的一些过程是基于输入到过程的样本值阵列必须是正方形的假设来定义的,因此ITU-T H.265中的一些过程无法为任意矩形视频块的编码提供足够的支持。此外,JEM中定义的QTBT分割和相关信令可能不太理想。本公开描述用于使用任意矩形视频块执行视频编码的技术。

[0097] 图7是示出了根据本公开的一个或多个技术的可以被配置为对视频数据进行编解码(即,编码和/或解码)的系统的示例的框图。系统100表示根据本公开的一个或多个技术的可以使用任意矩形视频块执行视频编码的系统的示例。如图1所示,系统100包括源设备102、通信介质110和目的地设备120。在图1所示的示例中,源设备102可以包括被配置为对视频数据进行编码并将经编码视频数据传输到通信介质110的任何设备。目的地设备120可以包括被配置为经由通信介质110接收经编码视频数据并且对经编码视频数据进行解码的任何设备。源设备102和/或目的地设备120可以包括配备用于有线和/或无线通信的计算设备,并且可以包括机顶盒、数字视频记录器、电视、台式计算机、膝上型计算机或平板计算机、游戏控制台、移动设备(包括例如“智能”电话、蜂窝电话)、个人游戏设备和医疗成像设备。

[0098] 通信介质110可以包括无线和有线通信介质和/或存储设备的任何组合。通信介质

110可以包括同轴线缆、光纤线缆、双绞线、无线发射机和接收机、路由器、交换机、中继器、基站或可以有益于实现各种设备和站点之间的通信的任意其它设备。通信介质110可以包括一个或多个网络。例如,通信介质110可以包括被配置为能够访问万维网(例如,因特网)的网络。网络可以根据一个或多个电信协议的组合来操作。电信协议可以包括专有方面和/或可以包括标准化电信协议。标准化电信协议的示例包括数字视频广播(DVB)标准、高级电视系统委员会(ATSC)标准、综合业务数字广播(ISDB)标准、有线数据业务接口规范(DOCSIS)标准、全球系统移动通信(GSM)标准、码分多址(CDMA)标准、第三代合作伙伴计划(3GPP)标准、欧洲电信标准协会(ETSI)标准、互联网协议(IP)标准、无线应用协议(WAP)标准以及电气与电子工程师协会(IEEE)标准。

[0099] 存储设备可以包括能够存储数据的任何类型的设备或存储介质。存储介质可以包括有形或非暂时性计算机可读介质。计算机可读介质可以包括光盘、闪存、磁存储器或任何其他合适的数字存储介质。在一些示例中,存储器设备或其部分可以被描述为非易失性存储器,并且在其他示例中,存储器设备的部分可以被描述为易失性存储器。易失性存储器的示例可以包括随机存取存储器(RAM)、动态随机存取存储器(DRAM)和静态随机存取存储器(SRAM)。非易失性存储器的示例可以包括磁性硬盘、光盘、软盘、闪存、或电可编程存储器(EPROM)或电可擦除可编程(EEPROM)存储器的形式。存储设备可以包括存储卡(例如,安全数字(SD)存储卡)、内部/外部硬盘驱动器和/或内部/外部固态驱动器。可以根据定义的文件格式将数据存储在存储设备上。

[0100] 再次参考图1,源设备102包括视频源104、视频编码器106和接口108。视频源104可以包括被配置为捕获和/或存储视频数据的任何设备。例如,视频源104可以包括视频相机和可操作地耦合到其上的存储设备。视频编码器106可包括被配置为接收视频数据且产生表示视频数据的兼容比特流的任何设备。兼容比特流可以指代视频解码器可以接收和根据其再现视频数据的比特流。可以根据视频编码标准来定义兼容比特流的各方面。当生成兼容比特流时,视频编码器106可以压缩视频数据。压缩可以是有损的(可识别的或不可识别的)或无损的。接口108可以包括被配置为接收兼容视频比特流并且将兼容视频比特流传输和/或存储到通信介质的任何设备。接口108可以包括网络接口卡,例如以太网卡,并且可以包括光学收发器、射频收发器或可以发送和/或接收信息的任何其他类型的设备。此外,接口108可以包括计算机系统接口,该计算机系统接口可以使兼容视频比特流能够存储在存储设备上。例如,接口108可以包括支持外围组件互连(PCI)和外围组件互连快速(PCIe)总线协议、专有总线协议、通用串行总线(USB)协议、I²C的芯片组,或可用于将对等设备互相连接的任何其他逻辑和物理结构。

[0101] 再次参考图1,目的地设备120包括接口122、视频解码器124和显示器126。接口122可以包括被配置为从通信介质接收兼容视频比特流的任何设备。接口108可以包括网络接口卡,例如以太网卡,并且可以包括光学收发器、射频收发器或可以接收和/或发送信息的任何其他类型的设备。此外,接口122可以包括计算机系统接口,该计算机系统接口使得能够从存储设备取回兼容视频比特流。例如,接口122可以包括支持PCI和PCIe总线协议、专有总线协议、USB协议、I²C的芯片组,或可用于将对等设备互相连接的任何其他逻辑和物理结构。视频解码器124可包括被配置为接收兼容比特流和/或其可接受变体且从其再现视频数据的任何设备。显示器126可以包括被配置为显示视频数据的任何设备。显示器126可以包

括各种显示设备中的一种,例如液晶显示器(LCD)、等离子显示器、有机发光二极管(OLED)显示器或其他类型的显示器。显示器126可以包括高清显示器或超高清显示器。应当注意,尽管在图7所示的示例中,视频解码器124被描述为向显示器126输出数据,但是视频解码器124可以被配置为将视频数据输出到各种类型的设备和/或其子组件。例如,视频解码器124可被配置为将视频数据输出到任何通信介质,如本文中所描述。

[0102] 图8是示出了可实施本文中描述的用于编码视频数据的技术的视频编码器200的示例的框图。应当注意,尽管示例视频编码器200被示为具有不同的功能块,但是这样的图示是出于描述的目的,并且不将视频编码器200和/或其子组件限制到特定的硬件或软件架构。可以使用硬件、固件和/或软件实现的任何组合来实现视频编码器200的功能。在一个示例中,视频编码器200可被配置为根据本文中所描述的技术对视频数据进行编码。视频编码器200可执行图片区域的帧内预测编码和帧间预测编码,且因此可称为混合视频编码器。在图8所示的示例中,视频编码器200接收源视频块。在一些示例中,源视频块可以包括已经根据编码结构划分的图片区域。例如,源视频数据可以包括宏块、CTU、CB、其细分部分和/或另一等效编码单元。在一些示例中,视频编码器200可被配置为执行源视频块的附加细分。应当注意,本文描述的一些技术可以一般地适用于视频编码,而不管在编码之前和/或期间如何分割源视频数据。在图8所示的示例中,视频编码器200包括求和器202、变换系数生成器204、系数量化单元206、逆量化/变换处理单元208、求和器210、帧内预测处理单元212、帧间预测处理单元214、后置滤波器单元216和熵编码单元218。

[0103] 如图8所示,视频编码器200接收源视频块并输出比特流。如上所述,JEM包括用于用信号通知QTBT树的以下参数:CTU大小、MinQTSIZE、MaxBTSIZE、MaxBTDepth和MinBTSIZE。表2示出了针对不同CTU大小的各个QT深度处的QT叶节点的块大小(在该示例中,MinQTSIZE是8)。此外,表3示出了针对二叉树根节点大小在各个BT深度处的BT叶节点的允许块大小(即,叶四叉树节点大小)。

[0104]

		CTU大小				
		256x256	128x128	64x64	32x32	16x16
QT深度	0	256x256	128x128	64x64	32x32	16x16
	1	128x128	64x64	32x32	16x16	8x8
	2	64x64	32x32	16x16	8x8	
	3	32x32	16x16	8x8		
	4	16x16	8x8			
	5	8x8				

[0105] 表2

[0106]

		QT 叶 CB 的块大小				
		128x128	64x64	32x32	16x16	8x8
BT 深度	1	128x64	64x32	32x16	16x8	8x4
		64x128	32x64	16x32	8x16	4x8
	2	64x64	32x32	16x16	8x8	4x4
		128x32	64x16	32x8	16x4	8x2
		32x128	16x64	8x32	4x16	2x8
	3	64x32	32x16	8x16	8x4	
		32x64	16x32	16x8	4x8	
		128x16	64x8	32x4	16x2	
		16x128	8x32	4x32	2x16	

[0107] 表3

[0108] 因此,参考表2,可以基于CTU大小和QT深度来确定形成二叉树的根的四叉树节点大小。如果将四叉树进一步分裂为二叉树,则可以基于QT节点大小和BT深度来确定二叉树叶节点大小,如表3所示。MaxBTSize、MaxBTDepth和MinBTSize中的每个可用于确定最小允许二叉树叶节点大小。例如,如果CTU大小是 128×128 ,QT深度是3,MaxBTSize是 16×16 ,并且MaxBTDepth是2,则最小允许二叉树叶节点大小包括64个样本(即, 8×8 、 16×4 或 4×16)。在这种情况下,如果MaxBTDepth为1,则最小允许二叉树叶节点大小包括128个样本(即, 16×8 或 8×16)。表4示出了针对 128×128 的CTU大小在QT深度和BT深度的各种组合处BT叶节点的块大小。

[0109]

		QT 深度				
		0	1	2	3	4
BT 深度	0	128x128	64x64	32x32	16x16	8x8
	1	128x64	64x32	32x16	16x8	8x4
		64x128	32x64	16x32	8x16	4x8
	2	64x64	32x32	16x16	8x8	4x4
		128x32	64x16	32x8	16x4	8x2
		32x128	16x64	8x32	4x16	2x8
	3	64x32	32x16	8x16	8x4	
		32x64	16x32	16x8	4x8	
		128x16	64x8	32x4	16x2	
		16x128	8x32	4x32	2x16	

[0110] 表4

[0111] 如上所述,JEM中定义的QTBT分割和相关信令可能不太理想。例如,如上面参考图3所述,在JEM中,当独立QTBT用于分割CTU时,亮度分量的CB无需并且不一定与色度分量的CB对齐。也就是说,在JEM中,当独立QTBT用于分割CTU时,使用QT分裂标志和BT分裂模式语法元素的单独集合来用信号通知亮度分量分割和色度分量分割中的每个,这样的信令可能不

太理想。

[0112] 在一些示例中,根据本文中所描述的技术,视频编码器200可被配置为分割CTU,使得亮度分量和色度分量具有共同分割结构达到特定深度,且因此共享QT分裂标志和BT分裂模式语法元素的共同集合达到特定深度。应当注意,在这种情况下,深度可以对应于QTBT的绝对深度(即,由QT深度和BT深度之和形成的深度)。应当注意,在一些情况下,深度可以对应于块中的分量(例如,亮度和/或色度)的样本的数量,并且可选地可以根据最小宽度和/或最小高度来指示。例如,可以共享QTBT,直到将色度样本的阵列分割至特定大小。例如,可以共享QTBT,直到节点的高度或宽度中的一个小于指定的分量样本数量,例如8个样本。例如,可以共享QTBT,直到节点的分量(例如,亮度和/或色度)的样本数小于指定数量,例如64。在一个示例中,可以针对CTU集合预先确定深度。例如,对于视频数据的片,深度可以设置为2,或者对于视频数据的图片,深度可以设置为2。在一个示例中,可以使用语法元素(例如,shared_depth等)来用信号通知深度。在一个示例中,可以在CTU级别用信号通知共享深度语法元素。在一个示例中,可在片级别用信号通知共享深度语法元素。在一个示例中,可以在参数集级别(例如,图片参数集(PPS)或序列参数集(SPS))用信号通知共享深度语法元素。在一个示例中,可以使用较高级别标志来指示较低级别处存在共享深度语法元素。例如,在片级别包括的语法元素可指示对于包括在片中的每个CTU是否包括共享深度语法元素。应当注意,以类似的方式,CTU级别标志可用于指示共享QTBT、部分共享QTBT或用于亮度和色度分量的独立QTBT中的一个或多个。

[0113] 在一个示例中,共享深度语法元素可以是处于分裂级别的标志。例如,对于每个QT分裂标志和/或BT分裂模式,相应的标志可以指示所指示的分裂是否被共享。在一个示例中,可使用处于高级别的共享深度语法元素来设定共享深度,且可使用较低级别标志来指示超出由该语法元素指定的级别的共享。例如,可以在片级别将共享深度设置为深度1,并且片内的每个CTU可以包括标记,该标记指示特定CTU共享是否被扩展为超过深度1而达到深度2。

[0114] 图9和图10是示出了亮度分量和色度分量具有公共分割达到共享深度的示例的概念图。在图9所示的示例中,亮度分量另外被分割超出共享深度1,并且色度分量未被分割超出深度1。在图10所示的示例中,亮度分量和色度分量两者都被独立地分割超出共享深度1。如上所述,视频采样格式可以相对于CU中包括的亮度样本的数量来定义CU中包括的色度样本的数量。在一个示例中,视频编码器200可被配置为基于采样格式选择性地色度分量分割超出共享深度。例如,在根据4:2:0样本格式来格式化CTU的情况下,在一个示例中,视频编码器200可以被配置为使得色度分量可以不被进一步分割超出共享深度。此外,在根据4:4:4样本格式来格式化CTU的情况下,在一个示例中,视频编码器200可以被配置为使得色度分量可以被进一步分割超出共享深度。此外,作为采样格式的附加或替代,以下中的一个或多个可用于确定是否允许色度分量被分割超出共享深度:CTU大小、MinQTSIZE、MaxBTSIZE、MaxBTDepth和/或MinBTSIZE。

[0115] 图11是示出了与图10中所示的示例QTBT分割相对应的QTBT的示例的概念图。如图11所示,亮度的QTBT和色度的QTBT直到深度1是相同的,即,共享深度是1。此外,应该注意,出于解释的目的,图11中所示的亮度树与图2中所示的QTBT相同。这样,对于图11中所示的示例,视频编码器200可被配置为基于表1中提供的伪语法用信号通知亮度QTBT。在一个示

例中,视频编码器200可被配置为基于表5中提供的伪语法用信号通知超出共享QTBT的色度QTBT。

[0116]

//从表 1 语法得出的色度的深度 0 和深度 1

If additional partitioning condition == TRUE:

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 1; //Depth 2 syntax chroma

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 2; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 2 语法色度

QT flag = 0, BT split = 2; //深度 2 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

QT flag = 0, BT split = 0; //深度 3 语法色度

[0117] 表5

[0118] 在表5所示的示例中,附加分割条件(addition partitioning condition)可以包括基于以下中的一个或多个的条件:采样格式、CTU大小、MinQTSIZE、MaxBTSIZE、MaxBTDepth和/或MinBTSIZE,如上所述。应注意,在一个示例中,视频编码器200可被配置为通过复合表1和表5中所示的语法元素来用信号通知超出共享QTBT的色度QTBT。例如,可以在超出共享节点的和作为共享节点后代的亮度分量节点的语法元素之后,用信号通知超出共享节点的和作为共享节点后代的色度分量节点的语法元素。表6示出了伪语法的示例,其中在将共享节点终止为叶节点的、用于亮度分量的语法元素之后,用信号通知用于色度分量的语法元素。在一个示例中,可在亮度语法元素之前用信号通知色度语法元素。

[0119]

```

QT flag = 1; //深度 0 语法
    QT flag = 1; //深度 1 语法
        QT flag = 0, BT split = 0; //深度 2 语法亮度
        QT flag = 0, BT split = 2; //深度 2 语法亮度
            BT split = 0; //深度 3 语法亮度
            BT split = 0; //深度 3 语法亮度
        QT flag = 0, BT split = 0; //深度 2 语法亮度
        QT flag = 0, BT split = 1; //深度 2 语法亮度
            BT split = 0; //深度 3 语法亮度
            BT split = 0; //深度 3 语法亮度
        QT flag = 0, BT split = 0; //深度 2 语法色度
        QT flag = 0, BT split = 0; //深度 2 语法色度
        QT flag = 0, BT split = 0; //深度 2 语法色度
        QT flag = 1; //深度 2 语法色度
            QT flag = 0, BT split = 0; //深度 3 语法色度
            QT flag = 0, BT split = 0; //深度 3 语法色度
            QT flag = 0, BT split = 0; //深度 3 语法色度
            QT flag = 0, BT split = 0; //深度 3 语法色度
    ...

```

[0120] 表6

[0121] 以此方式,视频编码器200表示以下设备的示例,该设备被配置为:接收视频块,所述视频块包括视频数据的第一分量和视频数据的第二分量的样本值;根据第一四叉树二叉树分割结构对视频数据的第一分量的样本值进行分割;以及根据该第一四叉树二叉树分割结构对视频数据的第二分量的样本值进行分割,达到共享深度。

[0122] 如上所述,ITU-T H.265支持用于帧间预测的四个非对称PB分割。应该注意,ITU-T H.265中提供的非对称PB分割可能不够理想。也就是说,ITU-T H.265中提供的非对称PB分割限于使得PB具有方形CB的宽度或高度的四分之一。例如,对于ITU-T H.265中的 32×32 CB, $M/4 \times M$ 左分割将CB分割成 8×32 PB和 24×32 PB。ITU-T H.265没有提供基于任意偏移将CB分割为PB的机制。也就是说,不允许PB具有任意宽度或高度。此外,应当注意,针对JEM,已经提出了根据非对称二叉树分割来分割CU的技术。F.Le Leannec等人的“Asymmetric Coding Units in QTBT”,第4次会议,中国成都,2016年10月15-21日,文件JVET-D0064(下文中称为“Le Leannec”)描述了以下内容:除了对称的垂直和水平BT分裂模式之外,还定义了四个附加的非对称BT分裂模式。在Le Leannec中,CU的四个另外定义的BT分裂模式包括:在高度的四分之一处的水平分割(对于一个模式在顶部,或对于一个模式在底部)或在宽度的四分之一处的垂直分割(对于一个模式在左侧,或对于一个模式在右侧)。Le Leannec中的四个另外定义的BT分裂模式在图19A中示出为Hot_Up、Hor_Down、Ver_Left和Ver_Right。因此,Le Leannec中的非对称BT分割类似于ITU-T H.265中提供的非对称PB分割,因此是有

局限性的,不允许任意分割。例如,根据Le Leannec中的技术,尺寸为 32×128 的矩形节点局限于如下分割:(1)对称地水平地分成两个 32×64 块;(2)对称地垂直地分成两个 16×128 块;(3)水平不对称地分成位于顶部位置的 32×32 块和位于底部位置的 32×96 块;(4)水平不对称地分成位于顶部位置的 32×96 块和位于底部位置的 32×32 块;(5)垂直不对称地分为位于左侧位置的 8×128 块和位于右侧位置的 24×128 块;(6)垂直不对称地分为位于左侧位置的 24×128 块和位于右侧位置的 8×128 块。表7提供了Le Leannec中用于用信号通知可能分割的二进制编码树信令的概览。

[0123]

二进制编码树					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	水平对称二叉树
0	1	0	1	0	顶部块尺寸的水平 1/4 (Hor_Up)
0	1	0	1	1	底部块尺寸的水平 1/4 (Hor_Down)
0	1	1	0	N/A	垂直对称二叉树
0	1	1	1	0	左侧块尺寸的垂直 1/4 (Ver_Left)
0	1	1	1	1	右侧块尺寸的垂直 1/4 (Ver_Right)

[0124] 表7

[0125] 在一些情况下,根据任意偏移对CTB进行分割可能是有用的。例如,在上面的示例中,对于 32×32 CB,在一些情况下,基于图像的属性,将CB分割为 10×32 PB和 22×32 PB可能是有用的。此外,参考上面的表3,在一些情况下,根据任意偏移进一步分割二叉叶节点可能是有用的。也就是说,在JEM中,可能的叶节点大小限于表3中所示的那些。例如,在二叉叶节点是 32×128 的情况下,将二叉叶节点进一步分割为 32×28 CB和 32×100 CB可能是有用的。应当注意,根据本文描述的技术的、根据任意偏移对视频数据块进行的分割可以至少应用于以下一种或多种情况:(1)在CU(或CB)形成PU(或PB)的根的情况下,可以将任意偏移分割应用于将CTU(或CTB)分割为CU(或CB);(2)在CU(或CB)不形成PU(或PB)的根的情况下,即在预测在CB级别确定的情况下,可以将任意偏移分割应用于将CTU(或CTB)分割为CU(或CB);(3)可以将任意偏移分割应用于PU(或PB)的分割;(4)可以将任意偏移分割应用于分割与编码树的节点相对应的样本块。应当注意,在一些情况下,可以选择性地启用任意偏移分割以用于CTU分割和/或PU分割。

[0126] 图12示出了根据偏移水平地进一步分割二叉叶节点的示例。应当注意,尽管图12中所示的示例包括根据任意偏移分割来分割二叉叶节点,但是这样的示例不应被解释为进行限制,并且如本文所述,任意偏移分割可适用于分割视频数据的各种场景。在图12所示的示例中,CTB可以对应于大小为 256×256 的亮度CTB。在这种情况下,右上角的二叉叶节点的大小为 32×128 。如上所述,将 32×128 二叉叶节点进一步分割为 32×28 CB和 32×100 CB可能是有用的。在图12所示的示例分割中,偏移的值为28。在一个示例中,视频编码器200可被配

置为根据偏移来分割QTBT的叶节点。在一个示例中,视频编码器200可被配置为使得可允许任何数量的非对称偏移分割结构。也就是说,在一些示例中,对于垂直偏移,偏移可以在2到块高度减去2的范围内,对于水平偏移,偏移可以在2到块宽度减去2的范围内。在一些示例中,对于垂直偏移,偏移可以在1到块高度减去1的范围内,对于水平偏移,偏移可以在1到块宽度减去1的范围内。在一些示例中,可以基于与CTU相关联的属性和/或预测模式来限制允许的非对称偏移分割。例如,可以基于CU是根据帧内预测还是帧间预测编码的来限制非对称偏移分割。此外,在一些示例中,可以基于CU或CB的大小来限制非对称偏移分割。在一个示例中,偏移的值可以被限制为整数倍数集合。在一个示例中,偏移的值可以被限制为整数倍数集合和一些另外的整数值(例如,2)。在一些示例中,整数倍数集合可以基于正在应用偏移的叶节点的大小。例如,如上所述关于水平分割 32×128 叶节点的情况。在一个示例中,偏移的值可以被限制为4的倍数(即,允许的偏移值包括4、8、12、16、...、120、124)。在一个示例中,可以使用索引的偏移值集合来指定偏移的值。例如,如上所述关于水平分割 32×128 叶节点的情况,在一个示例中,偏移的值可以限于下面的偏移值集合28、42、84和100。在一些示例中,可以选择索引的偏移值集合,以便避免可以使用QTBT信令或其近似变体来用信号通知的分割。例如,在水平分割 32×128 叶节点的情况下,在一些情况下(例如,取决于MaxBTDpth的值),BT结构可以允许 32×128 叶节点被分裂成两个 32×64 分割。在这种情况下,可以选择索引的偏移值集合,使得偏移不在指定范围64内。此外,在一些示例中,索引的偏移值集合可以基于MaxBTDpth的值。

[0127] 应当注意,在一些示例中,允许的非对称偏移分割可以包括水平或垂直分割。例如,在一个示例中,关于 32×128 二叉叶,视频编码器200可以被配置为进一步将 32×128 二叉叶节点分割为 8×128 CB和 24×128 CB。以这种方式,偏移可以指示相对于锚点的偏移值。例如,锚点可以包括用于垂直分割的左边缘和用于水平分割的顶边缘。应注意,在一些示例中,锚(anchor)可以是距边缘的设定数量的样本。例如,锚可以设置在距边缘4个样本处。以这种方式,偏移值零将指示距边缘4个样本的分割。在一个示例中,偏移可以包括固定长度二值化。在一个示例中,偏移可以包括截断的一元二值化。

[0128] 如上所述,在一个示例中,可以使用索引的偏移值集合来指定偏移的值。在一个示例中,索引的偏移值集合可以对应于分数分割。表8和表9提供了与分数分割相对应的索引的偏移值集合的示例。关于表8和表9,应当注意,在一些示例中,分数分割可以舍入到最近的样本值。例如,关于如上所述水平分割 32×128 叶节点的情况,在一个示例中,距边缘值 $1/3$ 的偏移可以舍入到43。关于表8和表9,应当注意,在示例中,可以将分数分割舍入到最近的整数倍样本值。例如,关于如上所述水平分割 32×128 叶节点的情况,在一个示例中,距边缘值 $1/3$ 的偏移可以舍入到44,44是最近的4倍样本。关于表8和表9,应当注意,在示例中,可以将分数分割向下舍入到最近的整数倍样本值。例如,如上所述关于水平分割 32×128 叶节点的情况,在一个示例中,距边缘值 $1/3$ 的偏移可以舍入到40,44是最近的4倍样本。

[0129]	距边缘的偏移	偏移的二进制表示
	所考虑的块尺寸的 $1/4$	01
	所考虑的块尺寸的 $1/2$	1
	所考虑的块尺寸的 $3/4$	00

[0130] 表8

[0131]

距边缘的偏移	偏移的二进制表示
所考虑的块尺寸的1/3	01
所考虑的块尺寸的1/2	1
所考虑的块尺寸的2/3	00

[0132] 表9

[0133] 如上所述,视频编码器200可被配置为用信号通知QTBT。在一个示例中,视频编码器200可被配置为通过在QTBT的信令内并入偏移信令来指示偏移值。例如,图12中所示的示例包括与图1中所示的示例相同的QTBT结构。这样,偏移信令可以基于表1中示出的示例伪语法,其中,在一个示例中,在指示叶节点的语法之后包括偏移信令。表10示出了与以下情况相对应的示例伪语法:针对 256×256 CTB,右上角的大小为 32×128 的二叉叶节点被进一步分割为 32×28 CB和 32×100 CB。

[0134]

```
...
    QT flag = 0; BT split = 1; //深度 1 语法
    BT split = 0; //深度 2 语法
    Offset = FALSE //偏移标志
    BT split = 1; //深度 2 语法
    BT split = 0; //深度 3 语法
    Offset = FALSE //偏移标志
    BT split = 0; //深度 3 语法
    Offset = TRUE; //偏移标志
    Offset_type = Horizontal; //偏移类型标志
    Offset value = 28; //偏移值
    QT flag = 0; BT split = 2; //深度 1 语法
...
```

[0135] 表10

[0136] 因此,根据表10中所示的示例,视频编码器200可被配置为用信号通知指示将偏移分割应用于QTBT叶节点的标志,用信号通知指示偏移分割是垂直分割还是水平分割的标志,且用信号通知指示偏移值的值。应注意,在其它示例中,视频编码器200可被配置为使用其他信令技术指示偏移值。例如,视频编码器200可被配置为在CB级别用信号通知偏移值。应当注意,在一些示例中,可以将偏移用信号通知为当前BT分裂模式信令的扩展。也就是说,例如,在JEM中,BT分裂模式语法元素导致节点变为原来的一半。在一个示例中,根据本文描述的技术,BT分裂模式信令可以包括用信号通知分裂类型和偏移对。例如,参考图12所示的示例,在一个示例中,可以如下用信号通知偏移:(BT分裂=2,偏移值=28)。

[0137] 此外,在一个示例中,CTB的每个CB可以根据定义的扫描顺序被索引,并且视频编码器200可以被配置为通过用信号通知CB的索引值来用信号通知偏移值。例如,参考图13,右上角的二叉叶节点被示为索引为CB₈。因此,在一个示例中,视频编码器200可被配置为使用此索引值来指示对此叶节点执行偏移分割。以此方式,视频编码器200表示以下设备的示

例,该设备被配置为确定偏移值并且根据该偏移值来分割叶节点。

[0138] 在一个示例中,按预定顺序的一组分裂决策(任意偏移分割和/或QT分割)可以应用于样本块,并使用单个指示符在比特流中指示。

[0139] 如上所述,可以将偏移用信号通知为当前BT分裂模式信令的扩展,并且可以包括用信号通知分裂类型和偏移对。在一些情况下,根据多个偏移对叶节点进行分割可能是有用的。例如,关于如上所述的水平分割 32×128 叶节点的情况,在一些情况下,将 32×128 叶节点从顶到底分割为 32×16 、 32×96 和 32×16 可能是有用的。图15示出了根据多个任意偏移对叶节点进行分割的一般情况。在图15中的顶右叶节点为 32×128 并且使用 32×16 、 32×96 和 32×16 的从顶到底分割的情况下,可以用信号通知 Offset_0 等于16并且可以用信号通知 Offset_1 等于112。应当注意,尽管在关于图15所示的示例中, Offset_0 是距顶边缘16个样本而 Offset_1 是距底边缘16个样本,但是在一些示例中,根据本文描述的技术, Offset_0 和 Offset_1 的值可以是任意的。

[0140] 如上所述,例如关于表8和表9,偏移值的信令可以包括使用与分数分割对应的索引的偏移值集合。以类似的方式,当根据多个偏移分割叶节点时,可以使用与分数分割对应的索引的偏移值集合。例如,在图15中的顶右节点为 32×128 并且使用 32×16 、 32×96 和 32×16 的从顶到底分割的情况下,可以通过使用对应于 $1/8$ 的索引值来用信号通知 Offset_0 和 Offset_1 ,假设用信号通知指示距顶部边缘的偏移和距底部边缘的偏移的垂直分割模式。应当注意,在一些情况下,关于一个方向将节点分割为三个块可以被称为三叉树(TT)分割。因此,分裂类型可以包括水平和垂直二叉分裂以及水平和垂直TT分裂。

[0141] Li等人的“Multi-Type-Tree”,第4次会议,中国成都,2016年10月15-21日,文件JVET-D0117r1(下文中称为“Li”)描述了一个示例,其中除了对称的垂直和水平BT分裂模式之外,还定义了两个附加的TT分裂模式。在Li中,节点的两个另外定义的TT分裂模式包括:(1)在距节点的顶边缘和底边缘四分之一高度处的水平TT分割;(2)在距节点的左边缘和右边缘四分之一宽度处的垂直TT分割。Li中的两个另外定义的TT分裂模式在图19A中示为垂直TT和水平TT。表11提供了在Li中用于用信号通知可能分割的二进制编码树信令的概览。

二进制编码树				分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	
1	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	叶节点
0	1	0	0	水平对称二叉树
0	1	0	1	块尺寸的 1/4 处的水平三叉树 (水平 TT)
0	1	1	0	垂直对称二叉树
0	1	1	1	块尺寸的 1/4 处的垂直三叉树 (垂直 TT)

[0142] 表11

[0143] 参考表11,Li中定义的分割类型是有限的。例如,在 32×128 节点的情况下,Li无法定义启用任意非对称BT分割(例如, 32×28 和 32×100 分割)的信令,并且还无法定义启用任意非对称TT分割(32×16 、 32×96 和 32×16 的从顶到底分割)的信令。因此,Li中描述的技术

可能不太理想。

[0145] 根据本文描述的技术,水平TT分割和垂直TT分割可以利用除了表11中所示的四分之一偏移之外的预定义偏移。例如,在一个示例中,水平TT分割和垂直TT分割可以利用三分之一偏移。在一个示例中,类似于上面关于表8和表9描述的技术,在TT分割中使用的分数分割可以舍入到最近的样本值或舍入到作为特定整数倍(例如,4)的样本值。例如,如果在距边缘1/4和3/4处发生的分数偏移被舍入到最近的4的整数倍,则在块尺寸为28的情况下,在这种情况下偏移可以在8和20处发生。在一些示例中,特定整数倍(对于一个或多个分量)可以取决于图片类型(例如,帧内或帧间)和/或视频采样格式。例如,在一些情况下,亮度分量和色度分量可以共享分割结构(例如,对于帧间图片),并且对于4:2:0色度格式,对于亮度分量,整数倍可以是8,对于色度分量,整数倍可以是4,并且对于4:4:4色度格式,对于亮度分量,整数倍可以是8,对于色度分量,整数倍可以是8。应当注意,在其他示例中,可以使用一个或多个数学运算的任何其他组合(例如,下取整到预定的整数倍,上取整到预定的整数倍,向上舍入到预定的整数倍,向下舍入到预定的整数倍)以导出偏移的预定整数倍。在一个示例中,可以仅将第一偏移导出为预定整数倍,并且可以基于第一偏移和/或被分割的尺寸(例如,高度、宽度)导出剩余偏移。例如,当被分割的块尺寸是28时,对于一个偏移,在距边缘1/4处发生的分数偏移可以舍入到最近的4的整数倍(即,在8处发生),并且可以将另一个偏移确定为距相对边缘相同数量(即,8)的样本(即,在20处发生)。

[0146] 再次参考表11,应该注意 Bin_2 指示分裂是垂直还是水平的,并且 Bin_3 指示分裂类型是BT还是TT,根据本文描述的技术, Bin_2 和 Bin_3 可以互换,例如, Bin_2 指示分裂类型是BT还是TT,而 Bin_3 指示分裂是垂直的还是水平的。此外,应当注意,在一些示例中,根据本文描述的技术,在一些示例中,可以推断出分割。例如,根据表11中提供的信令,在一个示例中,如果满足以下条件,则可以针对子节点将分割推断为TT(而不是BT):(1)如果在父节点上发生BT分裂以生成子节点,(2)子节点是第二个BT叶(即,右或底BT叶),并且(3)对第一个BT叶(即,左或顶BT叶)执行了与在父节点上发生的分裂垂直的BT分裂。也就是说,不允许在第二个BT上进行BT分裂以得到QT分裂,这是因为这种分裂可以使用父节点上的QT分裂来用信号通知。这样,在一些示例中,表11中的 Bin_3 可以推断为等于1。在示例中,当在解码器处推断出信息的信令时,该信息不被解码器明确接收。在示例中,当在解码器处推断出信息的信令时,该信息不被明确地包括在比特流中。

[0147] 如上所述,根据本文描述的技术,可以修改表11,使得 Bin_2 指示分裂类型是BT还是TT并且 Bin_3 指示分裂是垂直的还是水平的。在这种情况下,类似于上面描述的,可以推断出是垂直还是水平分裂。在一个示例中,如果满足以下条件,则可以针对子节点推断分割是垂直的还是水平的:(1)子节点是第二个BT叶(即,右或底BT叶),并且(2)对第一个BT叶(即,左或顶BT叶)执行了与在父节点上发生的分裂垂直的BT分裂。在这种情况下,针对该子节点,可以将分割推断为垂直于在第一个BT叶上发生的分裂。也就是说,不允许在第二个BT上进行BT分裂以得到QT分裂,这是因为这种分裂可以使用父节点上的QT分裂来用信号通知。

[0148] 根据本文描述的技术,视频编码器可以配置为根据QT分割、偏移BT分割和偏移TT分割来分割视频块。表12示出了根据本文描述的技术的用于用信号通知QT分割、偏移BT分割和偏移TT分割的二进制编码树信令的示例。

[0149]

二进制编码树				偏移信令		分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Offset ₀	Offset ₁	
1	N/A	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	N/A	叶节点
0	1	0	0	Offset ₀	N/A	水平偏移二叉树
0	1	0	1	Offset ₀	Offset ₁	水平偏移三叉树
0	1	1	0	Offset ₀	N/A	垂直偏移二叉树
0	1	1	1	Offset ₀	Offset ₁	垂直偏移三叉树

[0150] 表12

[0151] 如表12所示, Bin₀指示是否根据QT分裂对块进行了分割。在一些示例中, 当达到最大QT深度时, Bin₀不包括在比特流中, 并且其值可以由视频解码器推断为0。在一些示例中, 当一种类型的分割已被用于父节点 (例如, BT或TT) 时, Bin₀不包括在比特流中, 并且其值可以由视频解码器推断为0。Bin₁指示该块是否是叶节点。Bin₂指示是根据水平还是垂直分割模式对该块进行了分割。Bin₃指示是根据BT还是TT分割模式对该块进行了分割。如表12中进一步所示, 对于BT分割模式, 用信号通知偏移Offset₀。应当注意, 可以使用本文描述的偏移信令技术的任何组合来用信号通知Offset₀。例如, 可以使用与分数分割对应的索引的偏移值集合、使用距锚的样本数量、使用距锚的分数样本数、和/或使用标志并有条件地用信号通知偏移值, 从而用信号通知Offset₀。例如, 如果使用标志, 则标志可以指示分割是否是对称的, 或者可以指示分割是否是任意的并且用信号通知偏移值。如表12中进一步所示, 对于TT分割模式, 用信号通知偏移Offset₀和Offset₁。应当注意, 可以使用本文描述的偏移信令技术的任何组合来用信号通知Offset₀和Offset₁。参考表12, 应当注意, 在一些示例中, Bin₂和Bin₃的顺序可以互换。也就是说, 可以在指示BT或TT分割模式之一之后, 指示垂直或水平分割模式信令。

[0152] 除了BT和TT分裂类型之外, 还可以定义T形分裂类型。图17示出了QT分割、偏移BT分割、偏移TT分割和偏移T形分割的示例。如图17所示, T形分割包括首先根据BT分割对块进行分割, 并根据具有垂直取向的BT分割对所得块之一进一步分割。如图所示, T形分裂产生三个块。图16示出了根据具有任意偏移的T形分裂来分割叶节点的一般情况。如图17中进一步所示, 可以基于在第一次分割之后得到的哪个块被进一步分割 (例如, 对于垂直T形, 左还是右) 来定义T形分裂。应当注意, 如本文所使用的, 2X T形分割可以指使用两个对称BT分裂生成T形分割的情况。根据本文描述的技术, 视频编码器可被配置为根据QT分割、偏移BT分割、偏移TT分割和偏移T形分割来分割视频块。表13示出了根据本文描述的技术的用于用信号通知QT分割、偏移BT分割、偏移TT分割和偏移T形分割的二进制编码树信令的示例。

[0153]

二进制编码树						偏移信令		分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	Bin ₅	Offset ₀	Offset ₁	
1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	四叉树
0	0	N/A	N/A	N/A	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	N/A	Offset ₀	N/A	水平偏移二叉树
0	1	0	1	0	N/A	Offset ₀	Offset ₁	水平偏移三叉树
0	1	0	1	1	0	Offset ₀	Offset ₁	水平偏移 T 形树 (底部)
0	1	0	1	1	1	Offset ₀	Offset ₁	水平偏移 T 形树 (顶部)
0	1	1	0	N/A	N/A	Offset ₀	N/A	垂直偏移二叉树
0	1	1	1	0	N/A	Offset ₀	Offset ₁	垂直偏移三叉树
0	1	1	1	1	0	Offset ₀	Offset ₁	垂直偏移 T 形树 (右)
0	1	1	1	1	1	Offset ₀	Offset ₁	垂直偏移 T 形树 (左)

[0154] 表13

[0155] 如表13所示, Bin₀指示是否根据QT分裂对块进行了分割。在一些示例中, 当一种类型的分割已被用于父节点 (例如, BT、TT或T形) 时, Bin₀不包括在比特流中, 并且其值可以由视频解码器推断为0。此外, 在一些示例中, 当T形分割用于父节点时, 则不允许对当前节点进行BT分割。更一般地, 在一些示例中, 父节点的分割类型可以确定分割树中的当前节点的允许分割类型。此外, 应当注意关于表11至表13, 根据本文描述的技术, 可用于进一步分割节点的分割类型 (并且因此是否可以推导出二进制值) 可以取决于节点的块宽度、节点的块高度、节点的深度、用于生成节点的分割的数量和/或类型、用于生成节点的偏移、和/或同级节点的形状中的一个或多个。参考表13, Bin₁指示该块是否是叶节点。Bin₂指示是根据水平还是垂直分割模式对该块进行了分割。Bin₃指示是否根据BT分割模式对该块进行了分割。如表13中进一步所示, 对于BT分割模式, 用信号通知偏移Offset₀。应当注意, 可以使用本文描述的偏移信令技术的任何组合来用信号通知Offset₀。Bin₄指示是否根据TT分割模式对该块进行了分割。Bin₅指示T形分割的取向。如表12中进一步所示, 对于TT和T形分割模式, 用信号通知偏移Offset₀和Offset₁。应当注意, 可以使用本文描述的偏移信令技术的任何组合来用信号通知Offset₀和Offset₁。应当注意, 在一些示例中, Offset₀和Offset₁可以将偏移位置用信号通知为 X^{Offset} 。例如, X可以等于2, 并且因此Offset₀和Offset₁将偏移位置指示为2的幂。以类似的方式, Offset₀和Offset₁可以将偏移位置用信号通知为 $X \times \text{Offset}$ 。

在这些情况的每一种中, $Offset_0$ 和 $Offset_1$ 可以包括正整数, 并且在一些示例中可以大于特定值 (例如, 0 或 1)。此外, 参考表 13, 应当注意, 在一些示例中, 一些二进制数的顺序可以互换。此外, 应当注意, 在一些示例中, 由 Bin_5 指示的取向信息可以替代地由 $Offset$ 信令提供。

[0156] 应当注意, 可能存在有若干种方式来用信号通知 CU 的特定分割。在这种情况下, 用信号通知特定分割结构的一些方式可能被认为是低效和/或冗余的。例如, 在一些情况下, BT 和 TT 分裂可以产生可以通过简单地使用 QT 分裂来实现的方形块。可以不允许冗余和/或低效信令。应当注意, 当不允许信令时, 在一些情况下, 可以推断其他类型的信令, 例如, 如果在特定深度级别不允许某种类型的分裂 (例如, BT 分裂), 则可以在后续深度推断其他类型的分裂 (例如, TT 分裂)。

[0157] 此外, 应当注意, 尽管表 13 的示例中所示的信令使得根节点能够根据分裂模式和用信号通知的偏移的各种组合被分割成任意所得叶节点的集合, 但是在一些情况下, 与可以以更高效的方式用信号通知的分裂模式和偏移的特定组合相比, 生成所得叶节点的集合的分裂模式和偏移的特定组合在编码改进方面可能提供很少的益处。也就是说, 在用于用信号通知分割的开销 (例如, 用于用信号通知分割的平均比特数) 与所得叶节点的编码性能之间存在折衷。图 18 示出了示例, 其中左侧的 CTB 被示为根据具有预定义形状的 QT、BT、TT 和 T 形分割模式 (例如, 对称 BT 分裂、1/4-1/2-1/4 分裂和 2XT 形分裂) 被分割, 并且右侧的 CTB 被示为根据具有任意偏移的相应的 QT、BT、TT 和 T 形分割模式被分割。在图 18 所示的示例中, 对于特定图片, 右侧的 CTB 可以更紧密地与局部图片特性对齐, 然而, 可以用信号通知左侧的 CTB, 而不会产生与用信号通知任意偏移相关联的信令开销。根据本文描述的技术, 可以用信号通知具有预定义形状的一组分割模式, 以便优化分割灵活性和信令开销。

[0158] 图 19A 至图 19C 示出了根据本文描述的技术的可以用信号通知的预定义形状的示例。表 14 示出了根据本文描述的技术的用于用信号通知 QT 分割、对称 BT 分割、1/4 处 TT 分割和 2X T 形分割的二进制编码信令的示例。

[0159]

二进制编码					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	0	水平对称二叉树
0	1	0	1	0	块尺寸的 1/4 处的水平三叉树 (水平 TT)
0	1	1	0	0	水平 2XT 形树 (底部)
0	1	1	0	1	水平 2XT 形树 (顶部)
0	1	0	0	1	垂直对称二叉树
0	1	0	1	1	块尺寸的 1/4 处的垂直三叉树 (垂直 TT)
0	1	1	1	0	垂直 2XT 形树 (右)
0	1	1	1	1	垂直 2XT 形树 (左)

[0160] 表14

[0161] 应当注意,在其他示例中,可以使用表14的示例中所示的二进制编码来用信号通知其他类型的分割形状 (例如,图19A至图19C中所示的这些分割形状的子集)。表15示出了根据本文描述的技术的用于用信号通知QT分割、对称BT分割、和1/4T分割的二进制编码信号的示例。

[0162]

二进制编码					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	0	水平对称二叉树
0	1	0	1	0	顶-左 1/4T
0	1	1	0	0	左-底 1/4T
0	1	1	0	1	底-右 1/4T
0	1	0	0	1	垂直对称二叉树
0	1	0	1	1	顶-右 1/4T
0	1	1	1	0	右-顶 1/4T
0	1	1	1	1	左-底 1/4T

[0163] 表15

[0164] 应当注意, 在一些示例中, 可以使用较高级别的信令 (例如, 片级别或CU级别标志) 来指示表14中所示的预定义形状或表15中所示的预定义形状是否用于分割。

[0165] 应当注意, 在其他示例中, 可以使用表11的示例中所示的二进制编码来用信号通知其他类型的分割形状 (例如, 图19A至图19C中所示的这些分割形状的子集)。表16A至表16B示出了根据本文描述的技术的用于用信号通知QT分割、对称BT分割、和T形分割的二进制编码信令的示例。

[0166]

二进制编码树				分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	
1	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	叶节点
0	1	0	0	水平对称二叉树
0	1	0	1	水平 2XT 形 (底部)
0	1	1	0	垂直对称二叉树
0	1	1	1	垂直 2XT 形 (右)

[0167] 表16A

[0168]

二进制编码树				分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	
1	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	叶节点
0	1	0	0	水平对称二叉树
0	1	0	1	顶-左 1/4 T
0	1	1	0	垂直对称二叉树
0	1	1	1	右-底 1/4 T

[0169] 表16B

[0170] 应当注意,在一些示例中,可以使用较高级别的信令(例如,片级别或CU级别标志)来指示表11中所示的预定义形状、表16A中所示的预定义形状和/或表16B中所示的预定义形状是否用于分割。

[0171] 应当注意,在其他示例中,可以使用表7的示例中所示的二进制编码来用信号通知其他类型的分割形状(例如,图19A至图19C中所示的这些分割形状的子集)。表17A至表17B示出了根据本文描述的技术的用于用信号通知QT分割、对称BT分割、和T形分割的二进制编码信令的示例。

[0172]

二进制编码树					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	水平对称二叉树
0	1	0	1	0	水平 2XT 形(底部)
0	1	0	1	1	水平 2XT 形(顶部)
0	1	1	0	N/A	垂直对称二叉树
0	1	1	1	0	垂直 2XT 形(左)
0	1	1	1	1	水平 2XT 形(右)

[0173] 表17A

[0174]

二进制编码树					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	水平对称二叉树
0	1	0	1	0	顶-左 1/4 T
0	1	0	1	1	底-右 1/4 T
0	1	1	0	N/A	垂直对称二叉树
0	1	1	1	0	左-顶 1/4 T
0	1	1	1	1	左-底 1/4 T

[0175] 表17B

[0176] 应当注意,在一些示例中,可以使用较高级别的信令(例如,片级别或CU级别标志)来指示表7中所示的预定义形状、表17A中所示的预定义形状和/或表17B中所示的预定义形状是否用于分割。

[0177] 应当注意,在一些示例中,用于编码分割的二进制数的数量可以取决于块宽度、块高度、已经发生的分割的数量、在分割树的当前节点处允许的分割的偏移和/或在分割树的当前节点处允许的分割形状中的一个或多个的组合。例如,参考表17A至表17B,在一些示例中,对于其中 Bin_0 至 Bin_3 具有值0101或0111之一的二进制串, Bin_4 的信令可以取决于块大小是否具有最小样本数量。在这些情况下,当没有用信号通知 Bin_4 时,可以将新类型的分割分配给包括 Bin_0 至 Bin_3 的二进制串(例如,如果没有用信号通知 Bin_4 ,则表17B中的0111可以对应于垂直TT分割模式等)。

[0178] 在一个示例中,以下参数中的一个或多个可以确定是推断还是显式地用信号通知语法元素值:块高度、块宽度、样本数量和/或用于所允许分割的预定数量最小样本数。例如,如果当前块中的样本数小于用于所允许分割的集合的预定数量最小样本数,则推断出当前块不再被分割。

[0179] 参考图19B和图19C,应该注意,T形可以被描述为具有取向。例如,垂直2X T形(左)可以被描述为水平2X T形(底部)的顺时针旋转。因此,在示例中,包括预定义偏移的T形和具有任意偏移的T形可以用信号通知为具有取向(例如,一比特标记可以指示T形是否顺时针旋转)。再次参考图17,在一些示例中, Offset_0 和/或 Offset_1 的值可取决于块的宽度、块的高度和/或取向值中的一个或多个。在一些示例中,可以用信号通知的取向可以限于值的子集。在一些示例中,偏移可以限于值的子集,并且信令针对编码效率适当地改变。例如,参考图19B中的垂直2X T形(左)和水平2X T形(底部),因为可以使用两个连续的BT分割来用信号通知这两种配置,所以在一些情况下,使用取向值和偏移来用信号通知这些配置可能是不利的。此外,应当注意,在一些情况下,基于同级节点被分割的方式,可以不允许子节点进行特定的T形分割。例如,对于根据QT分割的父节点,四个子节点中的每个的可用分割可以彼此依赖。也就是说,可以不允许低效/冗余分割。在示例中,对于使用了BT分割的节点的分割树中的所有后代或后代的子集,可以不允许QT分割。在示例中,对于使用了TT分割的节点的分割树中的所有后代或后代的子集,可以不允许QT分割。在示例中,对于使用了T形分割的节点的分割树中的所有后代或后代的子集,可以不允许QT分割。在示例中,对于使用了T形分割的节点的分割树中的所有后代或后代的子集,可以不允许BT分割。

[0180] 此外,在一些示例中,可以不允许通过组合N个或更多个连续的特定分割类型而构造的分割。例如,对于方形块,可以通过组合2X T树和BT分裂来构造QT。信令可以被配置为不允许通过连续用信号通知2X T树和BT分裂组合来生成QT。

[0181] 在示例中,如果对于第一种情况,有类型为(水平, Offset_0)和(垂直, Offset_1)的两个连续分割,对于第二种情况,有类型为(垂直, Offset_1)和(水平, Offset_0)的两个连续分割,则不允许其中一种情况。可以适当地修改分割的信令,以考虑不允许的情况。在示例中, Offset_1 可以对应于(宽度- Offset_0)。

[0182] 在一个示例中,可以仅允许左-顶1/4T和顶-左1/4T分割中的一个。在一个示例中,可以仅允许左-底1/4T和底-左1/4T分割中的一个。在一个示例中,可以仅允许右-顶1/4T和顶-右1/4T分割中的一个。在一个示例中,可以仅允许右-底1/4T和底-右1/4T分割中的一

个。可以适当地修改分割的信令,以考虑不允许的分割。

[0183] 在一个示例中,允许(左-顶1/4T、左-底1/4T、右-顶1/4T、右-底1/4T)分割,但不允许(顶-左1/4T、底-左1/4T、顶-右1/4T、底-右1/4T),并且例如如表18所示修改分割信令。

[0184]

二进制编码树					分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	
1	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	水平对称二叉树
0	1	0	1	0	左-顶 1/4T
0	1	0	1	1	左-底 1/4T
0	1	1	0	N/A	垂直对称二叉树
0	1	1	1	0	右-顶 1/4T
0	1	1	1	1	右-底 1/4T

[0185] 表18

[0186] 如上所述,可以使用二进制编码来用信号通知其他类型的分割形状(例如,图19A至图19C中所示的这些分割形状的子集)。表19A至表19B示出了根据本文描述的技术的用于用信号通知QT分割、对称BT分割、TT分割和非对称1/4BT分割的二进制编码信令的示例。应当注意,在表19A至表19B所示的示例中,TT分割在一些示例中可以包括1/4TT分割且在其他示例中可以包括1/3TT分割。

[0187]

二进制编码树						分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	Bin ₅	
1	N/A	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	N/A	水平 TT
0	1	0	1	0	N/A	水平对称二叉树
0	1	0	1	1	0	顶部块尺寸的水平 1/4 (Hor_Top)
0	1	0	1	1	1	右侧块尺寸的水平 1/4 (Hor_Bottom)
0	1	1	0	N/A	N/A	垂直 TT
0	1	1	1	0	N/A	垂直对称二叉树
0	1	1	1	1	0	左侧块尺寸的垂直 1/4 (Ver_Left)
0	1	1	1	1	1	右侧块尺寸的垂直 1/4 (Ver_Right)

[0188] 表19A

[0189]

二进制编码树						分割类型
Bin ₀	Bin ₁	Bin ₂	Bin ₃	Bin ₄	Bin ₅	
1	N/A	N/A	N/A	N/A	N/A	四叉树分裂
0	0	N/A	N/A	N/A	N/A	叶节点
0	1	0	0	N/A	N/A	水平对称二叉树
0	1	0	1	0	N/A	水平 TT
0	1	0	1	1	0	顶部块尺寸的水平 1/4 (Hor_Top)
0	1	0	1	1	1	右侧块尺寸的水平 1/4 (Hor_Bottom)
0	1	1	0	N/A	N/A	垂直对称二叉树
0	1	1	1	0	N/A	垂直 TT
0	1	1	1	1	0	左侧块尺寸的垂直 1/4 (Ver_Left)
0	1	1	1	1	1	右侧块尺寸的垂直 1/4 (Ver_Right)

[0190] 表19B

[0191] 如上所述,在一些情况下,可以不允许冗余和/或低效信令。如上所述,在一些示例中,当不允许信令组合时,二进制值可以不包括在比特流中,并且解码器可以推断分割类型。例如,当在舍入之后所有BT偏移得出相同的分割时,则不必用信号通知偏移并且解码器可以推断出正确的偏移。此外,在另一示例中,在 $12 \times N$ 块可以被分割为 $(4 \times N, 8 \times N)$ 或 $(8 \times N, 4 \times N)$ BT分割,并且不能被分割为 $(3 \times N, 6 \times N, 3 \times N)$ TT分割(例如,因为3和6不是4的倍数)的情况下,指示BT或TT分割之一的信令是不需要的,解码器可以推断正确的分割。例如,参考表19A,在不允许TT分割的情况下,可以将Bin₃的值推断为1。

[0192] 在一个示例中,对于包括例如表19A至表19B中所示的TT分割和BT分割在内的TT分割和BT分割(对称或非对称),是否允许TT分割和/或BT分割可以基于以下中的一个或者多个:块大小、帧和/或片预测类型(即,帧内或帧间)、预定阈值(其可以与块大小或块尺寸进行比较(例如,块高度或宽度大于预定阈值X))、和/或帧的最小CU大小。应注意,在一些示例中,可以基于视频数据的属性来确定阈值。

[0193] 此外,在一个示例中,如果minPartSize表示帧和/或片的最小允许CU大小,则可以将以下规则中的一个或多个应用于表19A至表19B中所示的示例的TT分割和BT分割:(1) 如果CU大小 $\leq 2 * \text{minPartSize}$,则不允许对称BT或非对称1/4BT;(2) 如果CU大小 $\leq 2 * \text{minPartSize}$ 并且CU包括在帧内预测帧中,则不允许TT;(3) 如果CU大小 $< 2 * \text{minPartSize}$ 和CU包括在帧间预测帧中,则不允许TT。

[0194] 如上所述,在一些示例中,亮度和色度分量的分割可以是独立的,或者可以直至特定深度为止是不独立的。在一个示例中,帧间预测片可将公共分割树用于所有块(例如,PU、CU、TU、CTU)的亮度和色度分量两者。在一个示例中,帧间预测片中的视频块可基于预测模式将独立分割树用于亮度和色度分量(例如,帧内预测块可使用独立分割树)。在一个示例中,帧间预测片中的块可以基于预测模式使用部分独立的分割树(例如,基于预测模式从根开始直至某个深度共享分割树)。

[0195] 此外,在一些示例中,视频编码器200可被配置为根据其它预定义形状来分割视频块。图20示出了根据本文描述的技术的可以用信号通知的预定义形状的示例。图20中的形状可以称为对角分割形状。在图20所示的示例中,对角线边界上的样本可以包括在分割之一中。参考图20,关于对角TT分割形状,与上文中描述的偏移类似,偏移($\text{Offset}_{\text{WT}}$ 、 $\text{Offset}_{\text{WB}}$ 、 $\text{Offset}_{\text{HR}}$ 和 $\text{Offset}_{\text{HB}}$)在一些示例中可以用信号通知,在一些示例中可以预先确定。在一个示例中, $\text{Offset}_{\text{WT}}$ 和 $\text{Offset}_{\text{WB}}$ 可以相等,和/或 $\text{Offset}_{\text{HL}}$ 和 $\text{Offset}_{\text{HR}}$ 可以相等,如果用信号通知,则可以作为单个值用信号通知。此外,在一个示例中, $\text{Offset}_{\text{WT}}$ 和 $\text{Offset}_{\text{WB}}$ 可以等于 $w/\text{sqrt}(2)$,和/或 $\text{Offset}_{\text{HL}}$ 和 $\text{Offset}_{\text{HR}}$ 可以等于 $h/\text{sqrt}(2)$,其中 $\text{sqrt}(2)$ 返回2的平方根,并且得到的偏移被舍入为整数。在一些示例中,舍入可以包括舍入到最近的倍数(例如, $64/\text{sqrt}(2)$ 可以舍入到最近的4的倍数,即44)。在一些示例中,本文描述的任何分割偏移的舍入可以包括舍入到2的幂。应当注意,在一些示例中,对于对角TT分割形状,可以确定偏移使得中心分割的面积等于角部分割的面积之和。在其他示例中,可以定义分割面积之间的其他关系。应当注意,图20中所示的对角分割结构可以以与上面针对QT、BT和TT分割提供的类似方式用信号通知,例如,根据二进制编码树用信号通知。

[0196] 在一个示例中,对于对角分割,可以在被分割的矩形和/或方形块上而不是分割后的对角分割上进行变换。例如,对角分割可以用于PU而不是TU。在一个示例中,变换分割树可以独立于使用对角分割的分割树,并且可以包括矩形和/或方形块。在一个示例中,对角分割可以总是得到叶节点(即,不进一步分裂)。在这种情况下,不需要在比特流中用信号通知附加二进制数。如上所述,在一些示例中,对角分割可以用于PU,在一个示例中,对于帧内模式预测,由对角分割产生的每个节点可以是PB,并且因此可以具有单独的帧内模式。然而,应注意,在一些示例中,用于PU的参考样本集合可以是相同的。图21A至图21B示出了示例,其中对角分割得到PB。

[0197] 如上所述,对于对角分割,对角线边界上的样本可以包括在分割之一中。关于对角顶-左BT和对角顶-右BT,在一个示例中,可以定义BT分割,使得最左边的样本列包括在分割之一中。如上所述,待分割的节点可以包括矩形节点。图22A至图22D示出了根据对角顶-左BT和对角顶-右BT形状分割矩形节点(即, 16×4 节点)的示例。如图22A至图22D的示例所示,分割被称为 Part_0 和 Part_1 。 Part_0 和 Part_1 通常可以指代由对角BT分割产生的每个分割。应当注意,在图22A至图22D所示的每个示例中, Part_0 包括最左边的样本列。此外,图22A示出了 Part_0 包括底-右样本的示例,图22B示出了 Part_1 包括底-右样本的示例。以类似的方式,图22C示出了 Part_0 包括顶-右样本的示例,图22D示出了 Part_1 包括顶-右样本的示例。因此,在不同的示例中, Part_0 和 Part_1 可以包括父节点的不同样本。此外,应当注意,关于图22A至图22D中所示的示例, Part_0 和 Part_1 不包括相同数量的样本,例如,在图22A中, Part_0 包括34个样本, Part_1 包括30个样本,在图22B中, Part_0 包括33个样本, Part_1 包括31个样本。应当注意,在其他示例中,可以定义对角顶-左BT和对角顶-右BT形状,使得 Part_0 和 Part_1 包括相同数量的样本(即,对于 16×4 根节点为32)或与图22A至图22D的示例中所示的样本不同的样本。此外,应当注意,在一些示例中,可以定义对角顶-左BT和对角顶-右BT形状,使得 Part_0 包括最底行(例如,对于对角顶-左BT)或最顶行(例如,对于对角顶-右BT)。

[0198] 如上面参考图21A至图21B所述,在一些示例中,对于帧内预测,可以为对角分割产生的每个节点分配单独的预测模式(例如,由对角分割产生的每个节点可以是PB)。针对由

对角分割产生的PB,可用于选择的帧内预测模式和/或所选择的特定预测模式可以基于本文描述的一个或多个技术。此外,针对与由对角分割产生的PB相邻的块,可用于选择的帧内预测模式和/或所选择的特定预测模式可以基于本文描述的一个或多个技术。

[0199] 例如,关于对角顶-左BT和对角顶-右BT形状,在一些示例中,根据本文描述的技术,视频编码器200可被配置为使得不允许相同帧内预测模式用于两个部分(例如,用于Part₁的角度预测模式不能等同于用于Part₀的角度预测模式)。在一些示例中,根据本文中所述的技术,视频编码器200可被配置为使得仅可允许Part₀且Part₁的帧内预测模式的特定组合。表20示出了可以允许Part₀和Part₁的帧内预测模式的组合的示例。关于表20,应当注意,在一些示例中,“任意角度”条目可以用可能的角度预测模式的子集替换,或者可以包括除针对其他分割指定的平面或DC预测模式之外的所有可用预测模式(或其子集)。

[0200]

	Part ₀ 帧内预测模式	Part ₁ 帧内预测模式
组合1	平面	任何角度
组合2	任何角度	平面
组合3	DC	任何角度
组合4	任何角度	DC
组合5	平面	DC
组合6	DC	平面

[0201] 表20

[0202] 在一个示例中,表20中示出的子集或组合可用于选择。例如,在一个示例中,可以选择组合1、组合2、组合3和组合4中的一个用于Part₀和Part₁的帧内预测。在一个示例中,可以选择组合5和组合6中的一个用于Part₀和Part₁的帧内预测。在一个示例中,可用于选择的组合可以基于对角BT类型(顶-左或顶-右)、根/父节点的大小和/或根/父节点的形状中的一个或多个。

[0203] 在一个示例中,可以根据权重掩模来组合由针对Part₀选择的预测模式和针对Part₁选择的预测模式得到的相应预测。根据权重掩模组合预测可以促进沿着重构视频中的对角分割边界的平滑过渡。在一个示例中,加权可以取决于样本位置、到参考样本的距离、和/或针对每个分割选择的预测模式。在一个示例中,加权可以取决于样本位置和/或距分割边界的距离。此外,在一个示例中,可以使用两个帧内预测模式来形成组合预测。在一个示例中,组合预测可以是以下形式的线性组合:

[0204]
$$\text{Prediction} = \text{weight}_0 * \text{prediction}_{\text{Part}_0} + (1 - \text{weight}_0) * \text{prediction}_{\text{Part}_1}$$

[0205] 其中weight₀在0和1之间;

[0206] prediction_{Part₀}是针对Part₀产生的预测;并且

[0207] prediction_{Part₁}是针对Part₁产生的预测。

[0208] 在一个示例中,可以基于被预测的样本距预测中使用的参考样本的距离来确定weight₀。在一个示例中,可以基于距离集合S来确定weight₀。在一个示例中,集合S可以对应于被预测的样本距预测中使用的参考样本的距离(例如,欧几里德距离)的子集。

[0209] 在ITU-T H.265中,对于当前预测块,可以通过使用来自相邻帧内预测的预测单元的帧内预测模式来导出35种可能的帧内预测模式之一。在ITU-T H.265中,可以通过生成最可能模式(Most Probable Mode,MPM)的列表并且用信号通知与列表中的条目相对应的索

引值,从而从相邻帧内预测的预测单元导出帧内预测模式。在一个示例中,根据本文描述的技术,对于对角分割产生的PB和/或其相邻块,可以通过从相邻块推断帧内预测模式来导出帧内模式预测。

[0210] 参考图23A,在图23A所示的示例中,块C是当前编码块,并且块C继承用于其左相邻块(块L)中包括的PB之一的帧内预测模式(在这种情况下,用于PB₁的帧内预测模式)。关于图23A,在一个示例中,用于PB₀的帧内预测模式和用于PB₁的帧内预测模式可以包括在用于块C的MPM列表中,并且视频编码器200可以选择MPM中的一个并用信号通知相应的索引值。此外,在一个示例中,针对块C,可以推断用于PB₀的帧内预测模式和用于PB₁的帧内预测模式之一,因此不需要由视频编码器200用信号向相应的视频解码器通知索引值。在一个示例中,用于PB₀的帧内预测模式或用于PB₁的帧内预测模式之一的推断可以基于用于块L的对角分割形状(例如,可以针对对角顶-右BT推断出PB₁的模式,并且可以针对对角顶-左BT推断出PB₀的模式)。以类似的方式,MPM列表和分配给候选预测模式的对应的索引值可以基于用于块L的对角分割形状。

[0211] 在其他示例中,块C的预测模式可以是用于由相邻块中的对角分割产生的PB的帧内预测模式的函数。例如,如图23B所示,块C的预测模式可以是用于块L中的PB₀的帧内预测模式和用于块L中的PB₁的帧内预测模式的平均。在一个示例中,相邻块是从相邻块中的PB继承预测模式,还是使用相邻块中使用的预测模式的函数来确定预测模式,和/或还是使用特定函数(例如,使用平均函数或另一函数),可以基于用于相邻块的对角分割形状。例如,参考图23A和图23B中所示的示例,在使用对角顶-右BT来分割块L的情况下,块C可以从PB₁继承预测模式(如图23A所示),并且在使用对角顶-左BT来分割块L的情况下,块C的预测模式可以是PB₁和PB₀的预测模式的平均。

[0212] 此外,在一个示例中,相邻块是从若干可能的相邻块之一中的PB继承预测模式,还是使用一个或多个相邻块中使用的预测模式的函数来确定预测模式,还是使用特定函数(例如,使用平均函数或另一函数),和/或还是生成特定MPM列表,可以基于用于一个或多个相邻块的对角分割形状。例如,参考图23C,作为相邻块的块A和块L与块C相邻。在一个示例中,视频编码器200可以被配置为使得块C从块A中的PB₀或块L中的PB₁之一继承预测模式。此外,在一个示例中,视频编码器200可被配置为基于用于一个或多个相邻块的对角分割形状来产生MPM列表。在一个示例中,当根据对角分割形状分割一个或多个相邻矩形块(例如,图23C中的块A和块L)时,可以根据以下技术中的一个或多个生成MPM列表:(1)可以将每个对角分割的预测模式添加到MPM列表中(例如,在图23C中,块A中的PB₀和PB₁的预测模式和块L中的PB₀和PB₁的预测模式可以被添加到MPM列表中);和/或(2)可以将每个相邻块的相应预测模式添加到MPM列表。相邻块的相应预测模式可以包括来自矩形块中的对角PB之一的所选预测模式(例如,用于块L中的PB₀或PB₁之一的预测模式可以被添加到MPM列表中),或者可以包括矩形块中的对角PB的预测模式的函数(例如,块L中的PB₀和PB₁的预测模式可以进行平均,并且可以将平均值包括在MPM列表中)。在一个示例中,从矩形块中的对角PB之一选择预测模式可以包括选择与被编码的块最近(例如,就平均采样距离而言)的PB的帧内预测模式。应当注意,这样的选择将使得选择取决于分割的形状和相邻块相对于当前块的位置。

[0213] 如上所述,针对由对角分割产生的PB,可用于选择的帧内预测模式和/或所选择的特定预测模式可以得到约束。在一个示例中,视频编码器200可被配置为使得特定预测模式

不包括在MPM列表中。例如,在一个示例中,视频编码器200可被配置为使得不允许Part₁使用用于Part₀的帧内预测模式。在这种情况下,如果预测模式不可用于Part₁,则Part₁的可能预测模式的信令可以被简化为基于可能预测模式的减小的集合。例如,为Part₁生成MPM列表和非MPM列表的过程可以考虑以下约束:用于Part₀的帧内模式不是可能的预测模式。在一个示例中,可以修改这种情况下非MPM列表索引的二值化,以使所考虑的可能的预测模式减少一个。此外,应当注意,在一些示例中,视频编码器200可以被配置为使得不允许Part₁使用与用于Part₀的角度预测模式相邻的角度帧内预测模式。例如,如果Part₀使用垂直角度预测(例如,ITU-T H.265中的模式索引26),则可以不允许Part₁使用在垂直角度预测的指定角度内的角度预测模式(例如,不允许使用ITU-T H.265中的模式索引22-26)。通过以这种方式减少允许的预测模式,可以进一步提高用信号通知预测模式索引时的二值化效率。

[0214] 如上所述,在一些示例中,变换分割树可以独立于使用对角分割的预测分割树,并且可包括矩形和/或方形块。此外,可能存在对矩形和/或方形块进行操作的其他编码过程。在一些情况下,对矩形和/或方形块操作的过程可以是帧内预测模式的函数和/或取决于帧内预测模式。例如,是否执行特定变换(或特定变换类型)可取决于帧内预测模式。因此,在一些情况下,为矩形和/或方形块定义有效帧内预测模式可能是有用的。例如,参考图23A,在一些情况下,确定块L的有效预测模式可能是有用的。在一个示例中,有效帧内预测模式可以被预先确定为块的特定对角分割的预测模式(例如,对于对角BT,预测模式Part₀或Part₁之一)。在一个示例中,可以基于块的哪个对角分割包括最多样本来确定有效帧内预测模式。在一个示例中,可以基于块的一个或多个对角分割的帧内预测模式的函数来确定有效帧内预测模式。根据本文描述的技术,有效帧内预测模式可以用于以下编码过程中的至少一个或多个:从一组变换中选择二次变换(NSST);从一组扫描模式中选择系数扫描索引;以及在所谓的直接模式中,从与亮度块相对应的有效预测模式导出色度分量的帧内预测模式。

[0215] 此外,在其他示例中,以下编码处理中的一个或多个可以基于有效帧内预测模式:如果有效帧内模式是直接模式,则可以执行交叉分量预测;旋转变换(ROT)索引的信令可以取决于当前块的有效帧内模式;当有效帧内模式是水平帧内预测模式并且当前块使用残余差分脉冲编码调制(RDPCM)时,可以避免系数符号数据隐藏;当有效帧内模式是垂直帧内预测模式并且当前块使用RDPCM时,可以避免系数符号数据隐藏;如果块的所有分割(例如,对角分割块的两个(Part₀和Part₁)或三个或更多个部分)具有等于直接模式的帧内预测模式,则可以执行交叉分量预测;和/或当使用对角分割并且当前块使用RDPCM时,可以避免系数符号数据隐藏。

[0216] 在一个示例中,当对包括在帧内预测片中的对角分割的帧内预测编码块进行编码时,不同的分量(例如,亮度和色度)可以共享帧内预测模式信令。此外,在一个示例中,当对包括在帧内预测片中的对角分割的帧内预测编码块进行编码时,不同的分量不能共享帧内预测模式信令。

[0217] 再次参考图8,视频编码器200可以通过从源视频块中减去预测视频块来生成残差数据。求和器202表示被配置为执行该减法运算的组件。在一个示例中,视频块的减法发生在像素域中。变换系数生成器204将诸如离散余弦变换(DCT)、离散正弦变换(DST)或概念上类似的变换之类的变换应用于残差块或其细分部分(例如,四个8×8变换可以应用于16×

16的残差值阵列)以产生残差变换系数的集合。变换系数生成器204可以被配置为执行包括在离散三角变换族中的变换的任何和所有组合。如上所述,在ITU-T H.265中,TB被限制为以下大小: 4×4 、 8×8 、 16×16 和 32×32 。在一个示例中,变换系数生成器204可以被配置为根据具有 4×4 、 8×8 、 16×16 和 32×32 大小的阵列来执行变换。在一个示例中,变换系数生成器204还可以被配置为根据具有其他尺寸的阵列来执行变换。特别地,在一些情况下,对差值的矩形阵列执行变换可能是有用的。在一个示例中,变换系数生成器204可以被配置为根据以下大小的阵列执行变换: 2×2 、 $2 \times 4N$ 、 $4M \times 2$ 和/或 $4M \times 4N$ 。在一个示例中,2维(2D) $M \times N$ 逆变换可以被实现为1维(1D) M 点逆变换以及之后的1D N 点逆变换。在一个示例中,2D逆变换可以实现为1D N 点垂直变换以及之后的1D N 点水平变换。在一个示例中,2D逆变换可以实现为1D N 点水平变换以及之后的1D N 点垂直变换。变换系数生成器204可将变换系数输出到系数量化单元206。

[0218] 系数量化单元206可被配置为执行变换系数的量化。如上所述,可以通过调整量化参数来修改量化程度。系数量化单元206还可以被配置为确定量化参数和输出QP数据(例如,用于确定量化组大小和/或增量QP值的数据),其可以在视频解码期间由视频解码器用于重构量化参数以执行逆量化。应当注意,在其他示例中,可以使用一个或多个附加或替代参数来确定量化水平(例如,比例因子)。本文描述的技术一般可应用于:基于与视频数据的另一分量相对应的变换系数的量化水平,来确定与视频数据的一个分量相对应的变换系数的量化水平。

[0219] 如图8所示,量化的变换系数被输出到逆量化/变换处理单元208。逆量化/变换处理单元208可以被配置为应用逆量化和逆变换以生成重构的残差数据。如图8所示,在求和器210处,可以将重构的残差数据加到预测视频块。以此方式,可以重构经编码的视频块,并且可以使用所得到的重构视频块来评估给定预测、变换和/或量化的编码质量。视频编码器200可被配置为执行多遍编码(例如,改变预测、变换参数和量化参数中的一个或多个并执行编码)。可以基于对重构视频块的评估,来优化比特流的速率失真或其他系统参数。此外,可以存储重构的视频块并将其用作预测后续块的参考。

[0220] 如上所述,可以使用帧内预测来对视频块进行编码。帧内预测处理单元212可以被配置为针对待编码的视频块选择帧内预测模式。帧内预测处理单元212可被配置为评估帧和/或其区域,且确定用于对当前块进行编码的帧内预测模式。如图8中所示,帧内预测处理单元212将帧内预测数据(例如,语法元素)输出到熵编码单元218和变换系数生成器204。如上所述,对残差数据执行的变换可以是模式相关的。如上所述,可能的帧内预测模式可以包括平面预测模式、DC预测模式和角度预测模式。此外,在一些示例中,可以根据用于亮度预测模式的帧内预测来推断出用于色度分量的预测。帧间预测处理单元214可以被配置为对当前视频块执行帧间预测编码。帧间预测处理单元214可被配置为接收源视频块并且计算视频块的PU的运动矢量。运动矢量可以指示当前视频帧内的视频块的PU(或类似编码结构)相对于参考帧内的预测块的位移。帧间预测编码可以使用一个或多个参考图片。此外,运动预测可以是单预测(使用一个运动矢量)或双预测(使用两个运动矢量)。帧间预测处理单元214可以被配置为通过计算由例如绝对差之和(SAD)、平方差之和(SSD)或其他差异度量确定的像素差来选择预测块。如上所述,可以根据运动矢量预测来确定和指定运动矢量。帧间预测处理单元214可以被配置为执行运动矢量预测,如上所述。帧间预测处理单元214可以

被配置为使用运动预测数据生成预测块。例如,帧间预测处理单元214可以在帧缓冲器(图8中未示出)内定位预测视频块。应注意,帧间预测处理单元214还可以被配置为将一个或多个内插滤波器应用于经重构的残差块,以计算用于运动估计的子整像素值。帧间预测处理单元214可将针对所计算的运动矢量的运动预测数据输出到熵编码单元218。如图8中所示,帧间预测处理单元214可经由后置滤波器单元216接收经重构的视频块。后置滤波器单元216可以被配置为执行解块滤波和/或采样自适应偏移(SAO)滤波。解块指的是平滑重构视频块的边界的过程(例如,使得边界对于观看者来说更不易察觉)。SAO滤波是非线性幅度映射,其可以用于通过向重构视频数据添加偏移来改善重构。

[0221] 再次参考图8,熵编码单元218接收量化的变换系数和预测语法数据(即,帧内预测数据、运动预测数据、QP数据等)。应注意,在一些示例中,系数量化单元206可在将系数输出到熵编码单元218之前执行对包括量化变换系数的矩阵的扫描。在其他示例中,熵编码单元218可以执行扫描。熵编码单元218可以被配置为根据本文描述的一个或多个技术执行熵编码。熵编码单元218可以被配置为输出兼容比特流,即视频解码器可以从其接收和再现视频数据的比特流。

[0222] 图14是示出了根据本公开的一个或多个技术的可以被配置为对视频数据进行解码的视频解码器的示例的框图。在一个示例中,视频解码器300可被配置为基于上文所描述的技术中的一个或多个来重构视频数据。也就是说,视频解码器300可以以与上述视频编码器200相反的方式操作。视频解码器300可被配置为执行帧内预测解码和帧间预测解码,且因此可称为混合解码器。在图14所示的示例中,视频解码器300包括熵解码单元302、逆量化单元304、逆变换处理单元306、帧内预测处理单元308、帧间预测处理单元310、求和器312、后置滤波器单元314和参考缓冲器316。视频解码器300可被配置为以与视频编码系统一致的方式解码视频数据,所述视频编码系统可以实施视频编码标准的一个或多个方面。应当注意,尽管示例视频解码器300被示为具有不同的功能块,但是这样的图示是出于描述的目的,并且不将视频解码器300和/或其子组件限制到特定的硬件或软件架构。可以使用硬件、固件和/或软件实现方式的任何组合来实现视频解码器300的功能。

[0223] 如图14中所示,熵解码单元302接收熵编码的比特流。熵解码单元302可被配置为根据与熵编码过程互逆的过程来解码来自比特流的经量化语法元素和经量化系数。熵解码单元302可被配置为根据上文所描述的任何熵编码技术来执行熵解码。熵解码单元302可以以与视频编码标准一致的方式解析编码的比特流。视频解码器300可被配置为解析经编码比特流,该经编码比特流是基于上述技术产生的。也就是说,例如,视频解码器300可以被配置为确定基于上面描述的一个或多个技术生成的和/或用信号通知的QTBT分割结构,以用于重构视频数据。例如,视频解码器300可被配置为解析语法元素和/或评估视频数据的属性,以便确定QTBT的共享深度。此外,视频解码器300可被配置为确定偏移值并根据偏移值来分割视频数据块。

[0224] 再次参见图14,逆量化单元304接收来自熵解码单元302的经量化变换系数(即,水平值)和量化参数数据。量化参数数据可以包括上面描述的增量QP值和/或量化组大小值等的任何和所有组合。视频解码器300和/或逆量化单元304可被配置为:基于由视频编码器用信号通知的值和/或通过视频属性和/或编码参数,来确定用于逆量化的QP值。也就是说,逆量化单元304可以以与上述系数量化单元206互逆的方式操作。例如,逆量化单元304可被配

置为根据上文所描述的技术推断预定值(例如,基于编码参数确定QT深度和BT深度的总和)、允许的量化组大小等。逆量化单元304可以被配置为应用逆量化。逆变换处理单元306可以被配置为执行逆变换以生成重构的残差数据。由逆量化单元304和逆变换处理单元306分别执行的技术可以类似于由上述逆量化/变换处理单元208执行的技术。逆变换处理单元306可被配置为将逆DCT、逆DST、逆整数变换、不可分离二次变换(NSST)或概念上类似的逆变换过程应用于变换系数,以便产生像素域中的残差块。此外,如上所述,是否执行特定变换(或特定变换类型)可取决于帧内预测模式。如图14所示,可以将重构的残差数据提供给求和器312。求和器312可以将重构的残差数据加到预测视频块并生成重构的视频数据。可以根据预测视频技术(即,帧内预测和帧间预测)来确定预测视频块。在一个示例中,视频解码器300和后置滤波器单元314可被配置为确定QP值且将其用于后置滤波(例如,解块)。在一个示例中,视频解码器300的使用QP的其它功能块可基于所接收的信令确定QP并将其用于解码。

[0225] 帧内预测处理单元308可被配置为接收帧内预测语法元素且从参考缓冲器316取回预测视频块。参考缓冲器316可以包括被配置为存储一帧或多帧视频数据的存储器设备。帧内预测语法元素可以标识帧内预测模式,例如上文所描述的帧内预测模式。在一个示例中,帧内预测处理单元308可根据本文中所描述的帧内预测编码技术中的一个或多个来重构视频块。帧间预测处理单元310可接收帧间预测语法元素,并且生成运动矢量以标识存储在参考缓冲器316中的一个或多个参考帧中的预测块。帧间预测处理单元310可产生运动补偿块,其可能基于内插滤波器执行内插。用于子像素精度的运动估计的内插滤波器的标识符可以包括在语法元素中。帧间预测处理单元310可使用内插滤波器来计算参考块的子整像素的内插值。后置滤波器单元314可以被配置为对重构的视频数据执行滤波。例如,后置滤波器单元314可被配置为执行解块滤波和/或SAO滤波,如上文关于后置滤波器单元216所描述的。此外,应当注意,在一些示例中,后置滤波器单元314可以被配置为执行专有自主滤波(例如,视觉增强)。如图14中所示,重构的视频块可以由视频解码器300输出。以此方式,视频解码器300可被配置为根据本文中所描述的一种或多种技术产生经重构视频数据。以这种方式,视频解码器300可以被配置为解析第一四叉树二叉树分割结构,将该第一四叉树二叉树分割结构应用于视频数据的第一分量,确定共享深度,以及将该第一四叉树二叉树分割结构应用于视频数据的第二分量直至共享深度。以此方式,视频解码器300表示以下设备的示例,即,该设备被配置为确定偏移值并且根据该偏移值来分割叶节点。

[0226] 在一个或多个示例中,可以用硬件、软件、固件或它们的任意组合来实现所描述的功能。如果用软件实现,则功能可以作为一个或多个指令或代码存储在计算机可读介质上或通过计算机可读介质传输,并由基于硬件的处理单元来执行。计算机可读介质可以包括计算机可读存储介质,其对应于诸如数据存储介质之类的有形介质,或者可以包括通信介质,其包括有助于例如根据通信协议将计算机程序从一个地方传送到另一个地方的任何介质。以这种方式,计算机可读介质通常可以对应于(1)非暂时性的有形计算机可读存储介质或(2)诸如信号或载波之类的通信介质。数据存储介质可以是任何可用介质,能够由一个或多个计算机或一个或多个处理器访问该可用介质以取回用于实现本公开中描述的技术的指令、代码和/或数据结构。计算机程序产品可包括计算机可读介质。

[0227] 通过示例而非限制的方式,这样的计算机可读存储介质可以包括RAM、ROM、

EEPROM、CD-ROM或其它光盘存储器、磁盘存储器或其它磁存储设备、闪存或者可用于存储以指令或数据结构形式的期望程序代码并可由计算机访问的任何其它介质。此外,可以将任意连接适当地命名为计算机可读介质。例如,如果使用同轴电缆、光缆、双绞线、数字用户线(DSL)或无线技术(例如红外线、无线电和微波)从网站、服务器或其他远程源发送指令,则同轴电缆、光缆、双绞线、DSL或无线技术(例如红外线、无线电和微波)包括在介质的定义中。然而,应该理解,计算机可读存储介质和数据存储介质不包括连接、载波、信号或其他暂时性介质,而是涉及非暂时性有形存储介质。如本文中所使用的磁盘和光盘包括紧凑盘(CD)、激光盘、光盘、数字多功能盘(DVD)、软盘和蓝光盘,其中,磁盘通常以磁的方式再现数据,而光盘用激光以光的方式再现数据。以上项的组合也应当包括在计算机可读介质的范围内。

[0228] 指令可以由一个或多个处理器执行,例如一个或多个数字信号处理器(DSP)、通用微处理器、专用集成电路(ASIC)、现场可编程逻辑阵列(FPGA)或其他等效的集成或离散逻辑电路。因此,本文使用的术语“处理器”可以指任何前述结构或适于实现本文所述技术的任何其他结构。另外,在一些方面,可以在配置用于编码和解码的专用硬件和/或软件模块内提供本文描述的功能,或者将本文描述的功能包含在组合的编解码器中。而且,这些技术可以在一个或多个电路或逻辑元件中完全实现。

[0229] 本公开的技术可以在各种各样的设备或装置中实现,包括无线电话、集成电路(IC)或IC组(例如,芯片组)。在本公开中描述了各种组件、模块或单元以强调被配置为执行所公开的技术的设备的功能方面,但不一定需要由不同的硬件单元实现。而是,如上所述,各个单元可以组合在编解码器硬件单元中,或者由一组互操作硬件单元提供,其包括如上所述的一个或多个处理器以及合适的软件和/或固件。

[0230] 此外,在上述每个实施例中使用的基站设备和终端设备的每个功能块或各种特征可以由电路来实现或执行,该电路通常是一个集成电路或多个集成电路。设计为执行本说明书中描述的功能的电路可以包括通用处理器、数字信号处理器(DSP)、专用或通用集成电路(ASIC)、现场可编程门阵列(FPGA)或其它可编程逻辑器件、分立门或晶体管逻辑、或分立硬件组件、或它们的组合。通用处理器可以是微处理器,或者备选地,处理器可以是常规处理器、控制器、微控制器或状态机。通用处理器或上述每个电路可以由数字电路构成或者可以由模拟电路构成。此外,当由于半导体技术的进步而出现制成取代当前集成电路的集成电路的技术时,还能够使用通过该技术制成的集成电路。

[0231] 已经描述了各种示例。这些示例和其他示例在所附权利要求的范围内。

[0232] <交叉引用>

[0233] 此非临时申请根据35U.S.C.§119要求2017年1月31日的临时申请No.62/452,868、2017年2月28日的No.62/465,135、2017年3月3日的No.62/466,976、2017年3月29日的No.62/478,362、2017年4月28日的No.62/491,884的优先权,这些申请的全部内容通过引用并入本文。

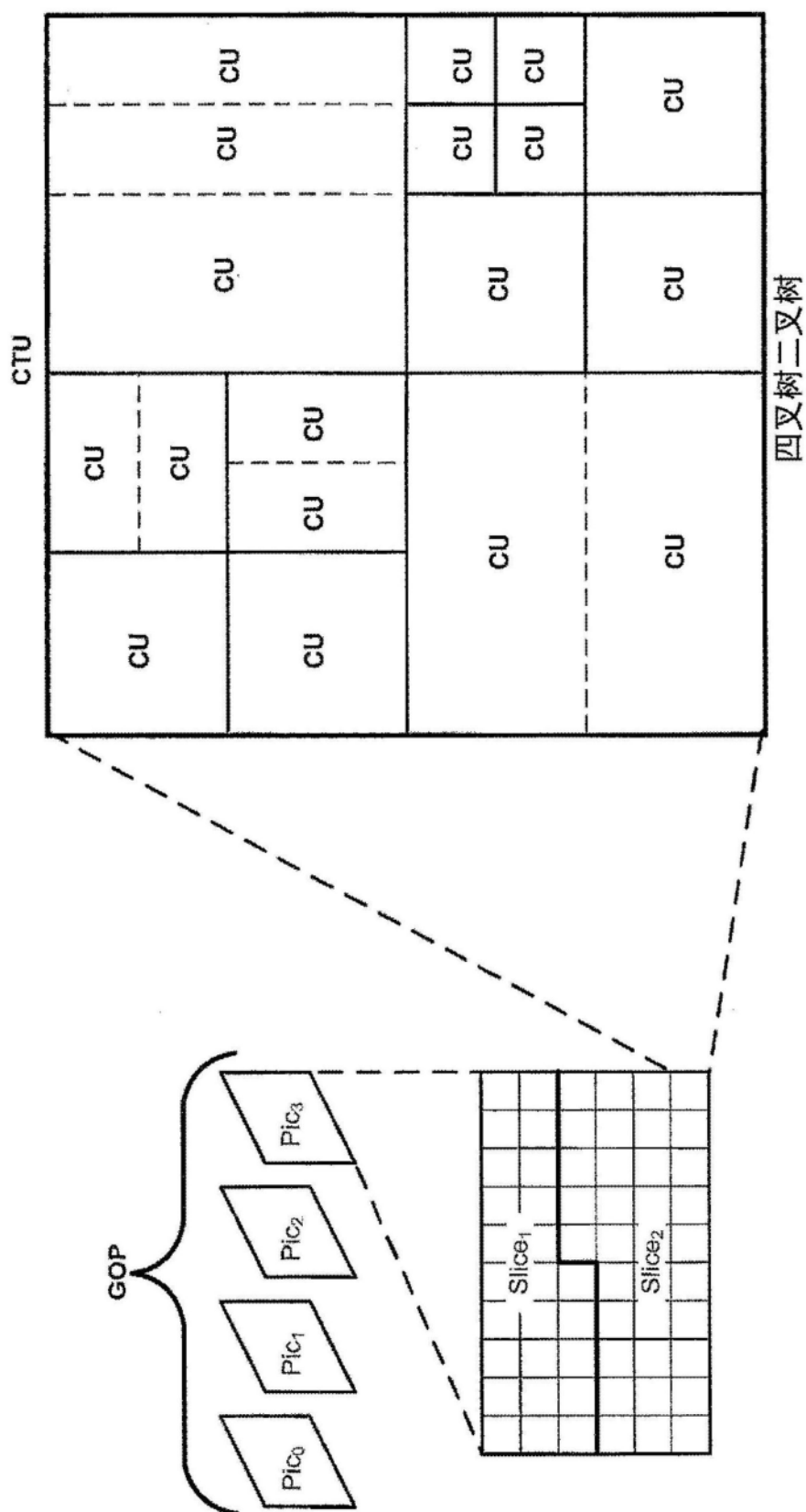


图1

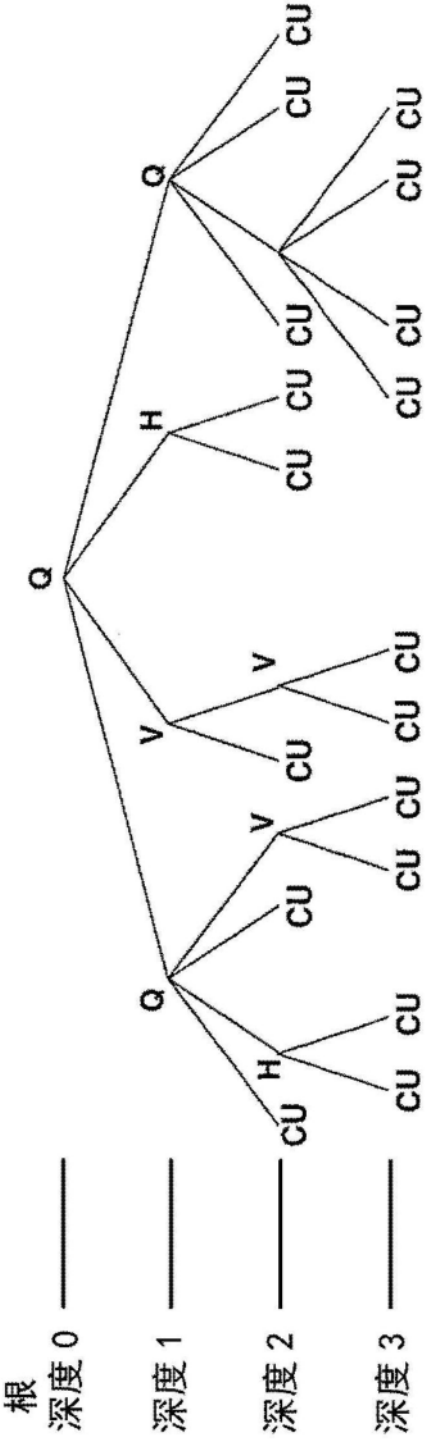


图2

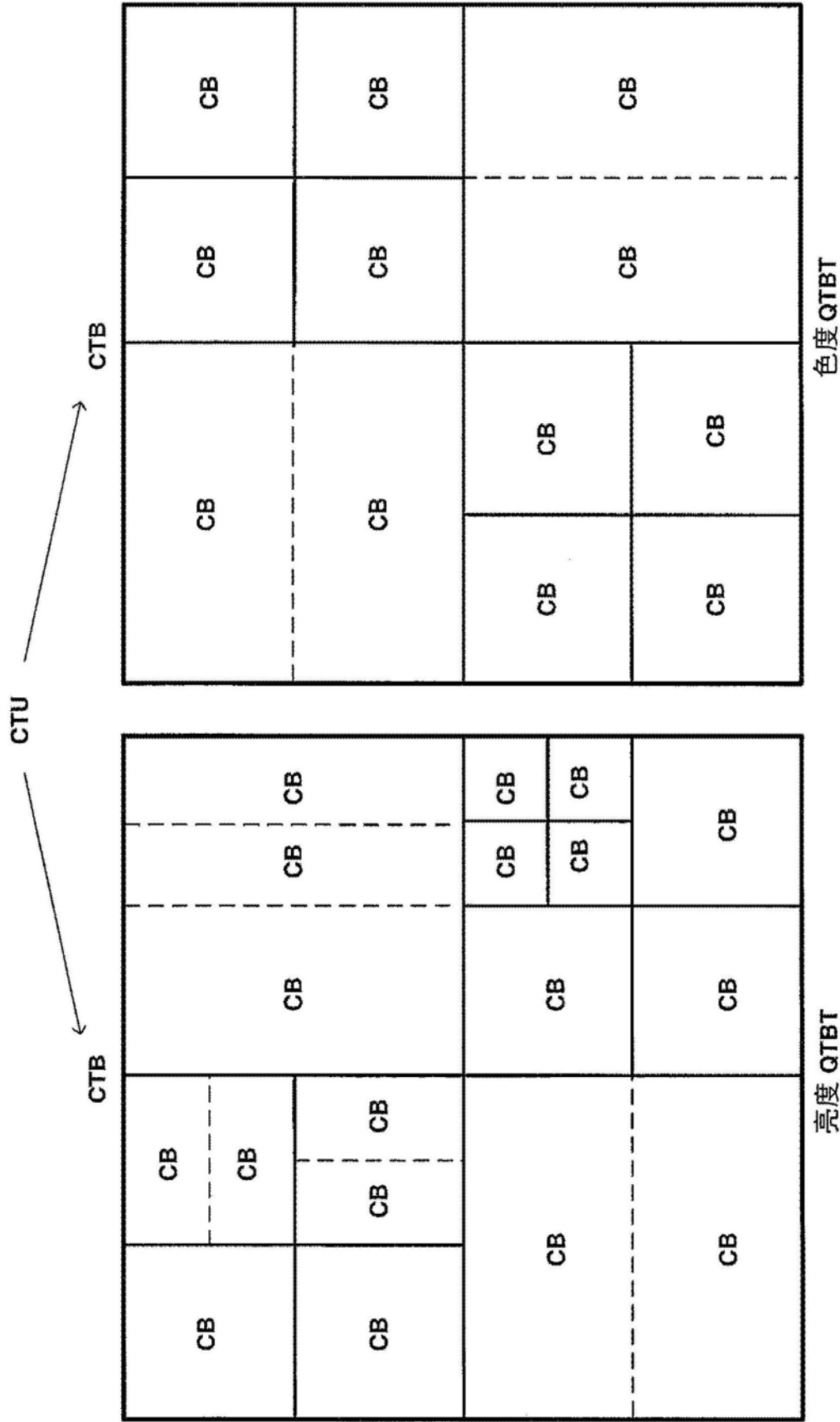


图3

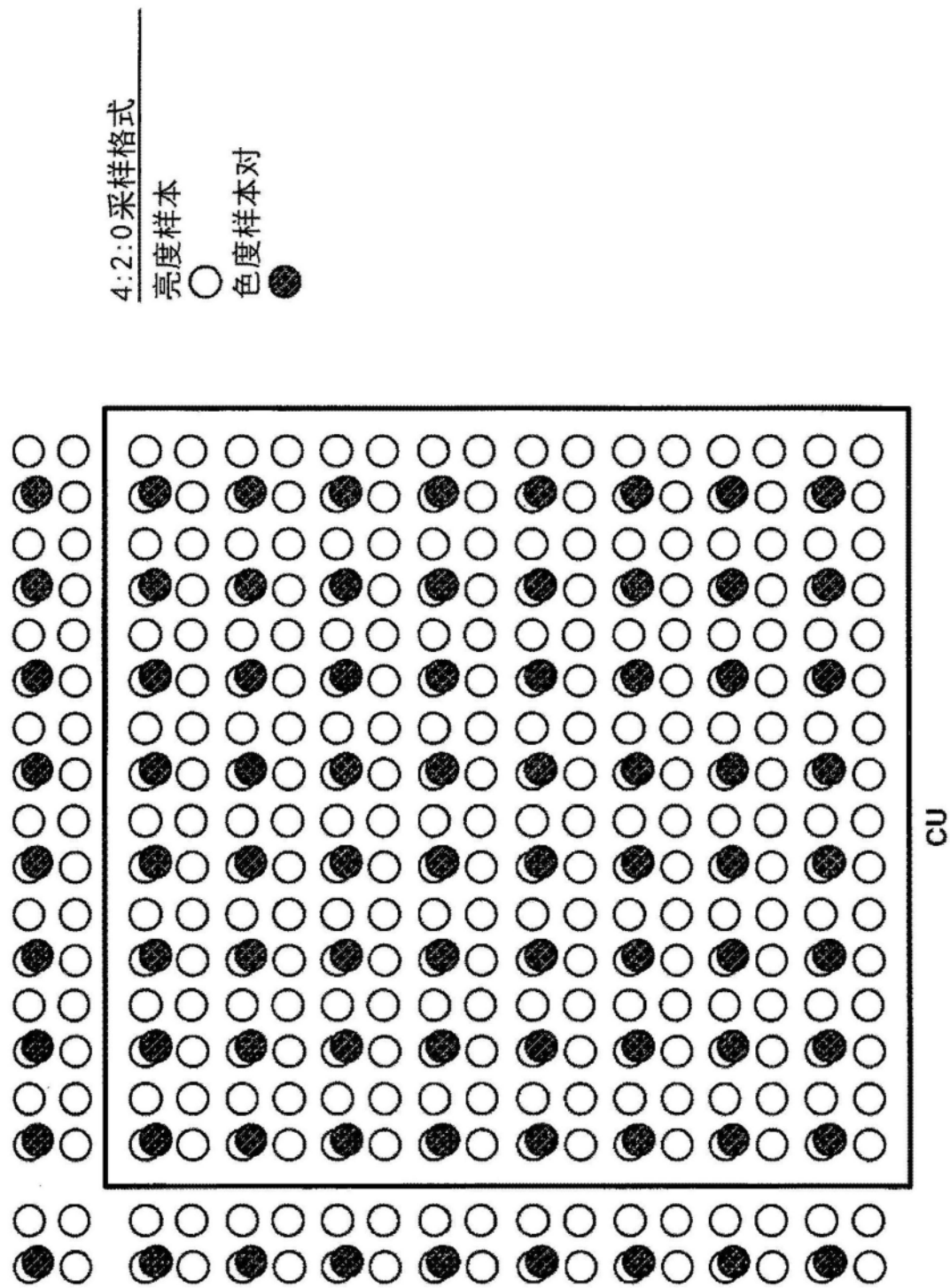


图4

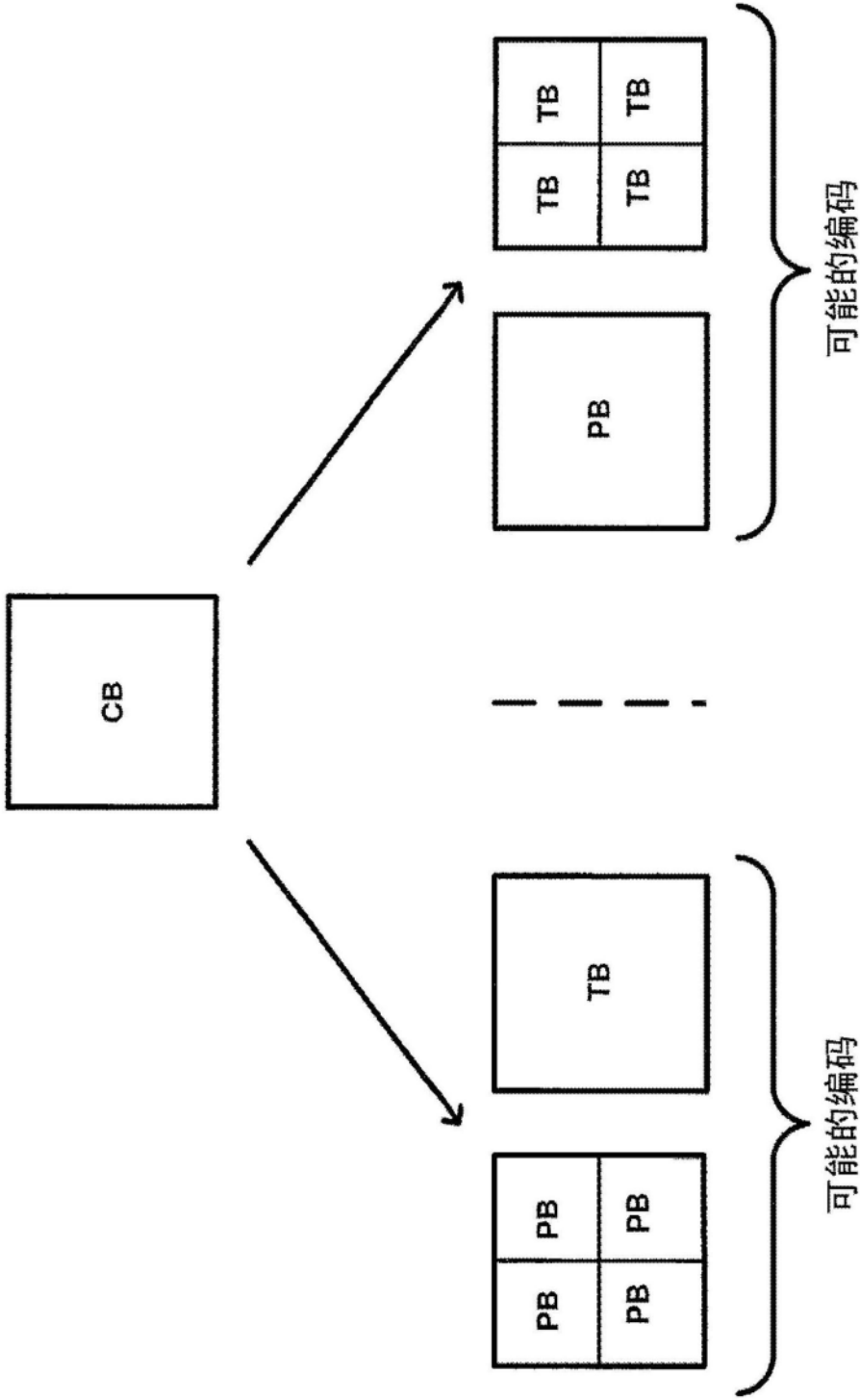


图5

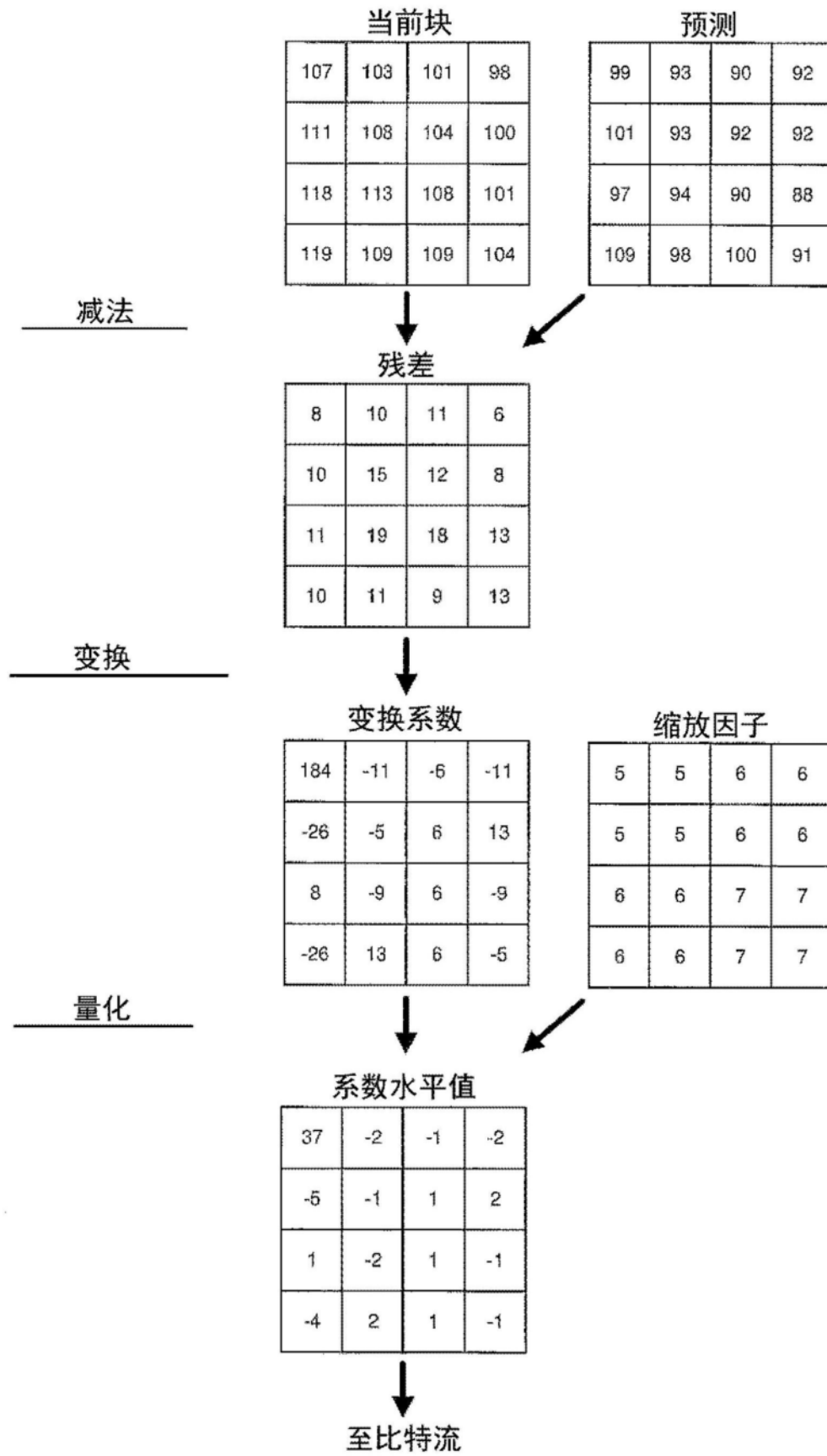


图6A

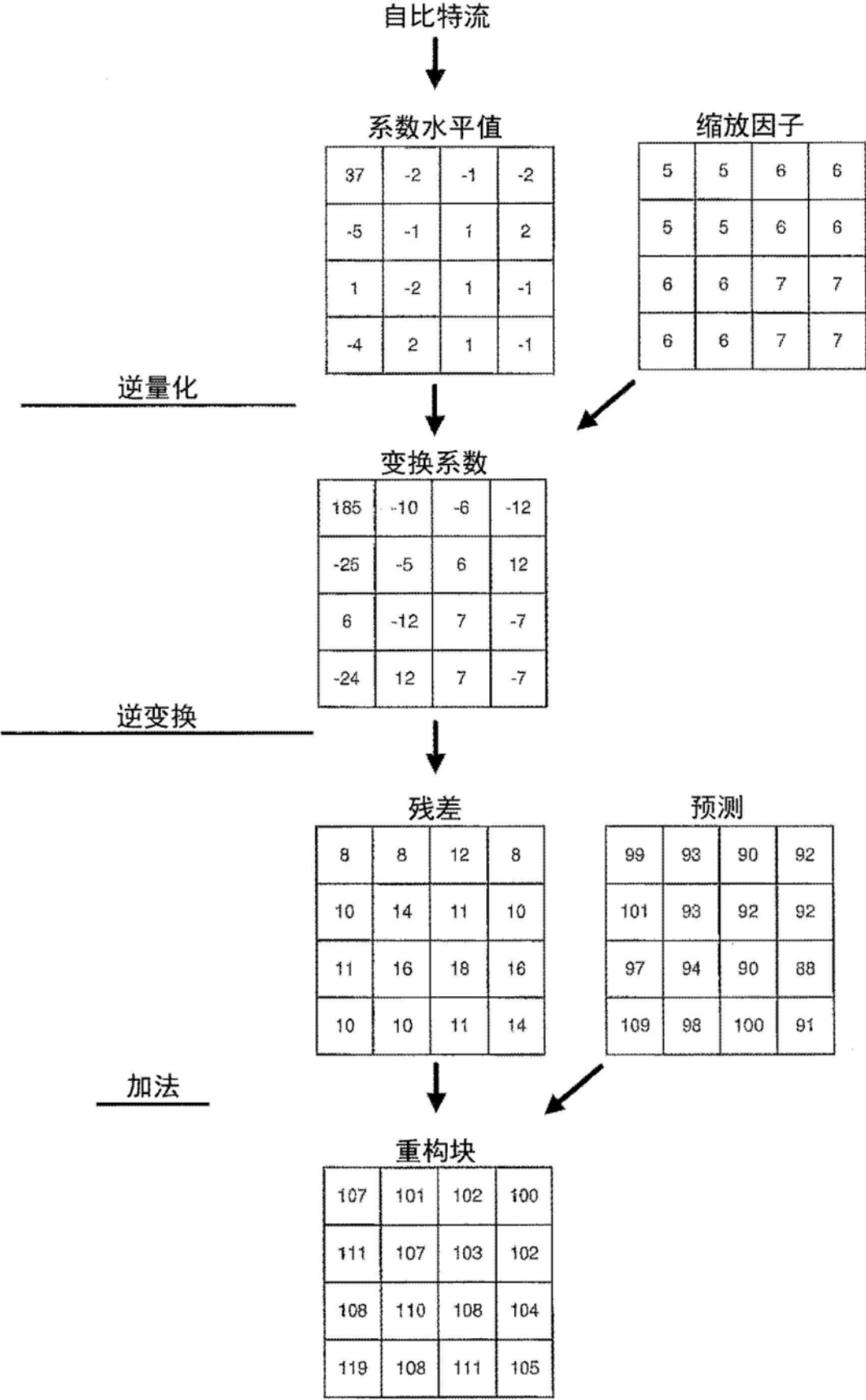


图6B

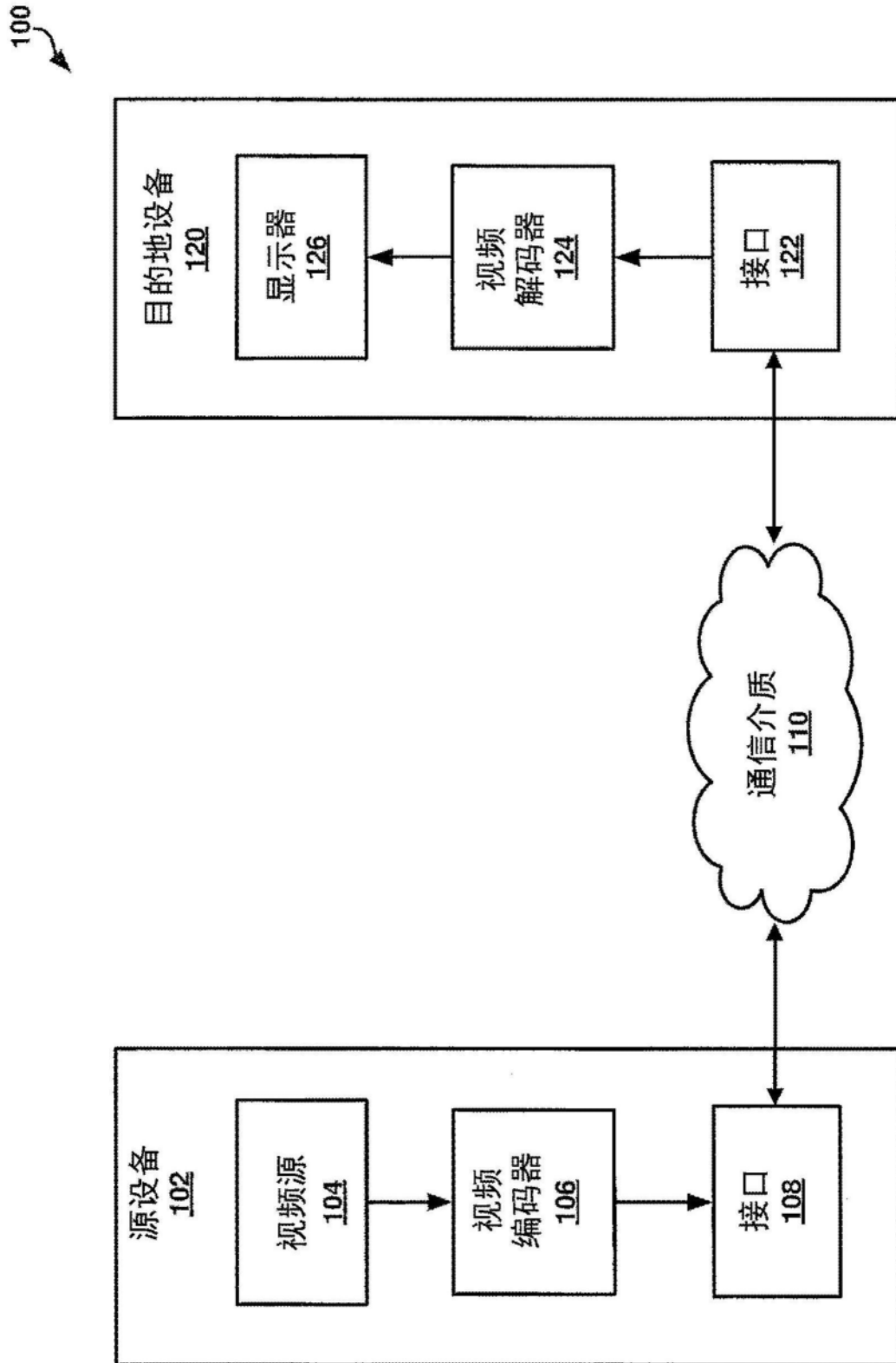


图7

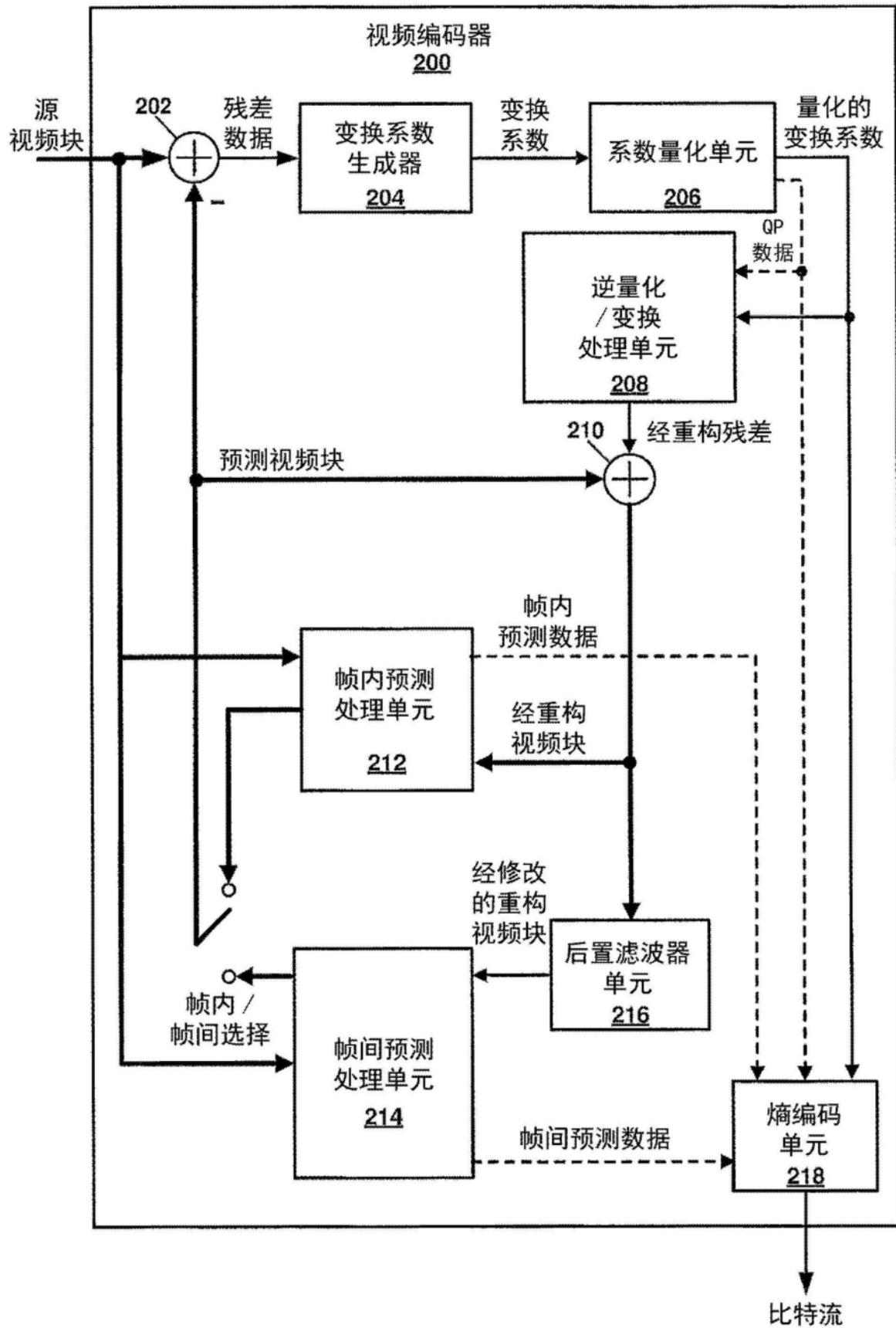


图8

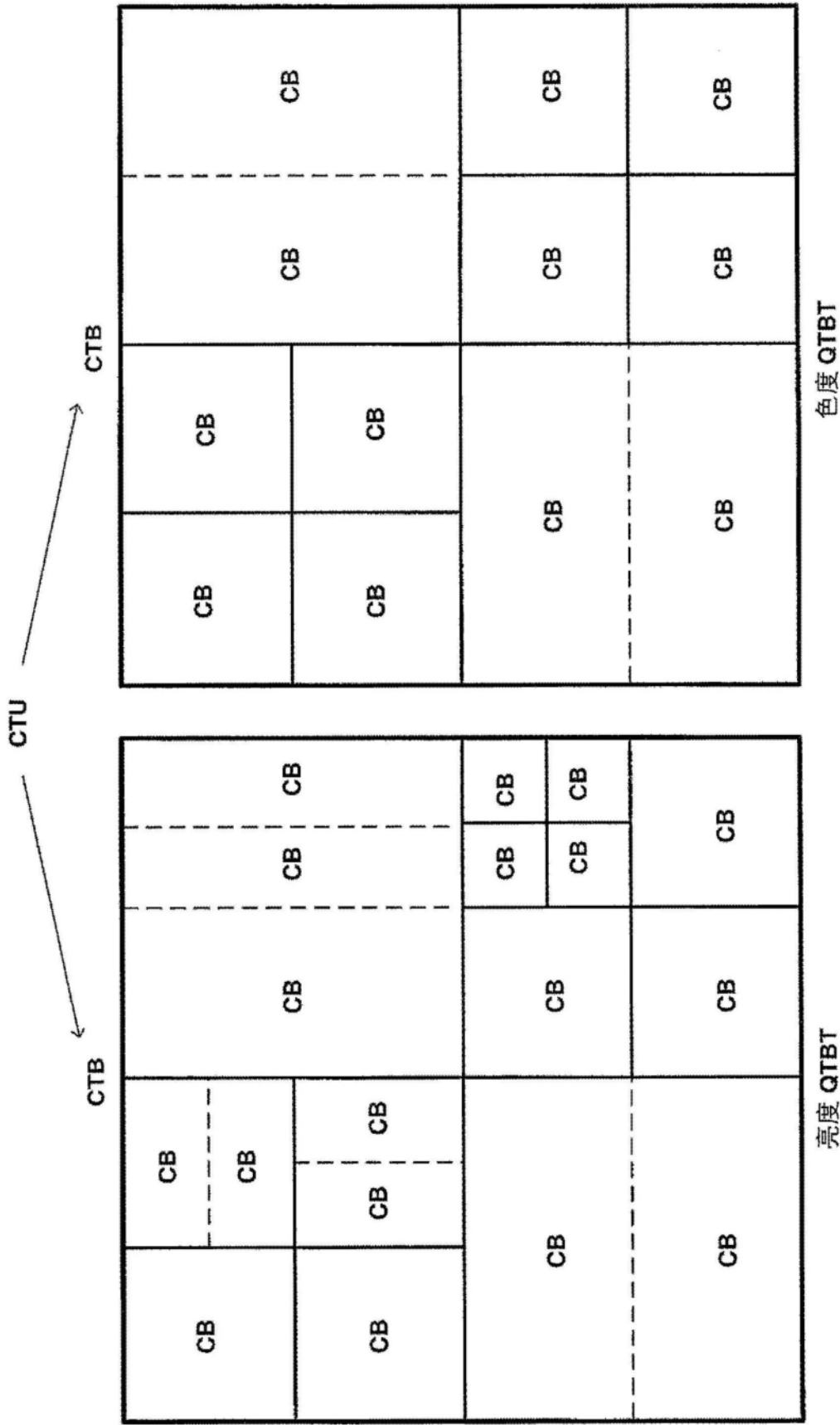


图9

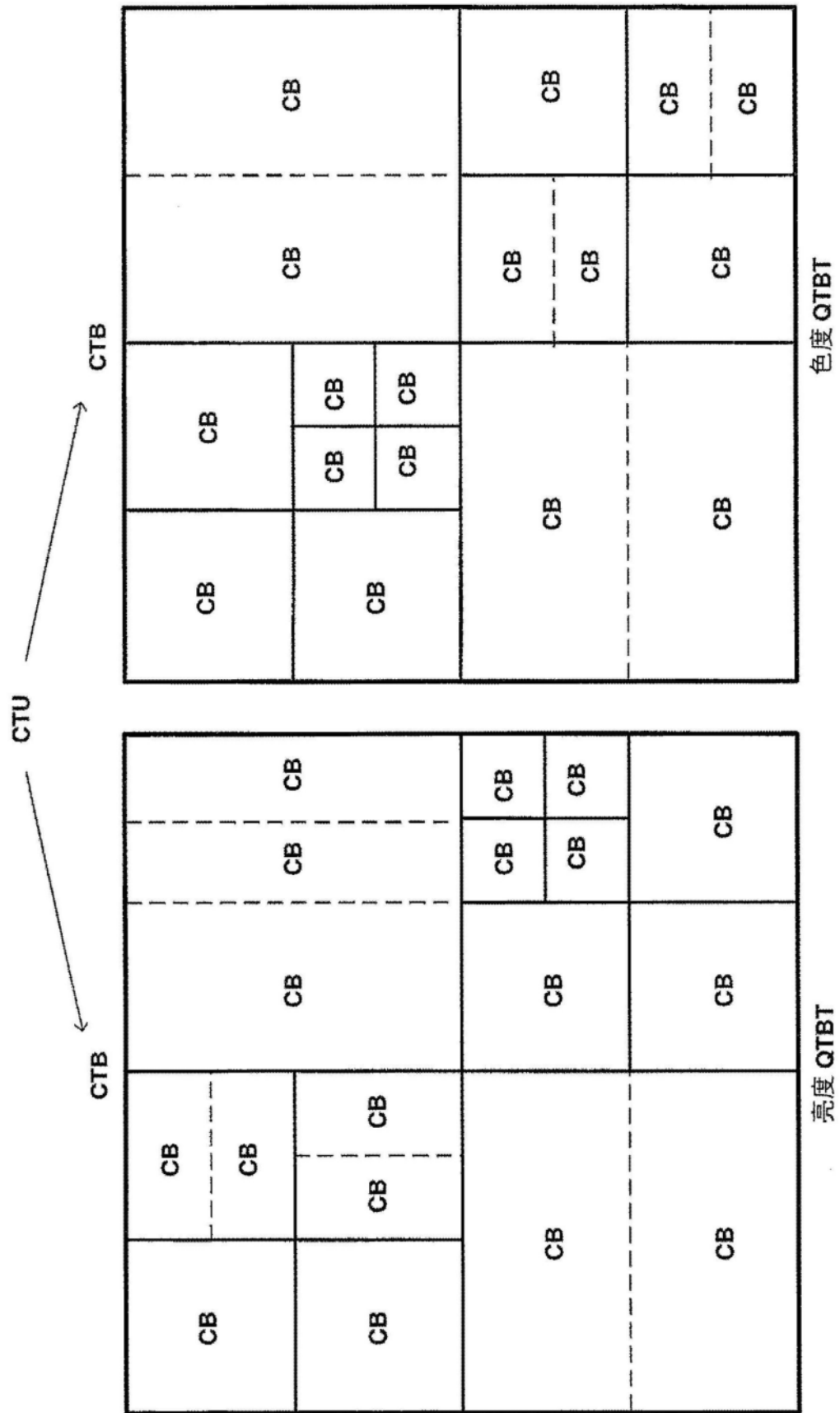


图10

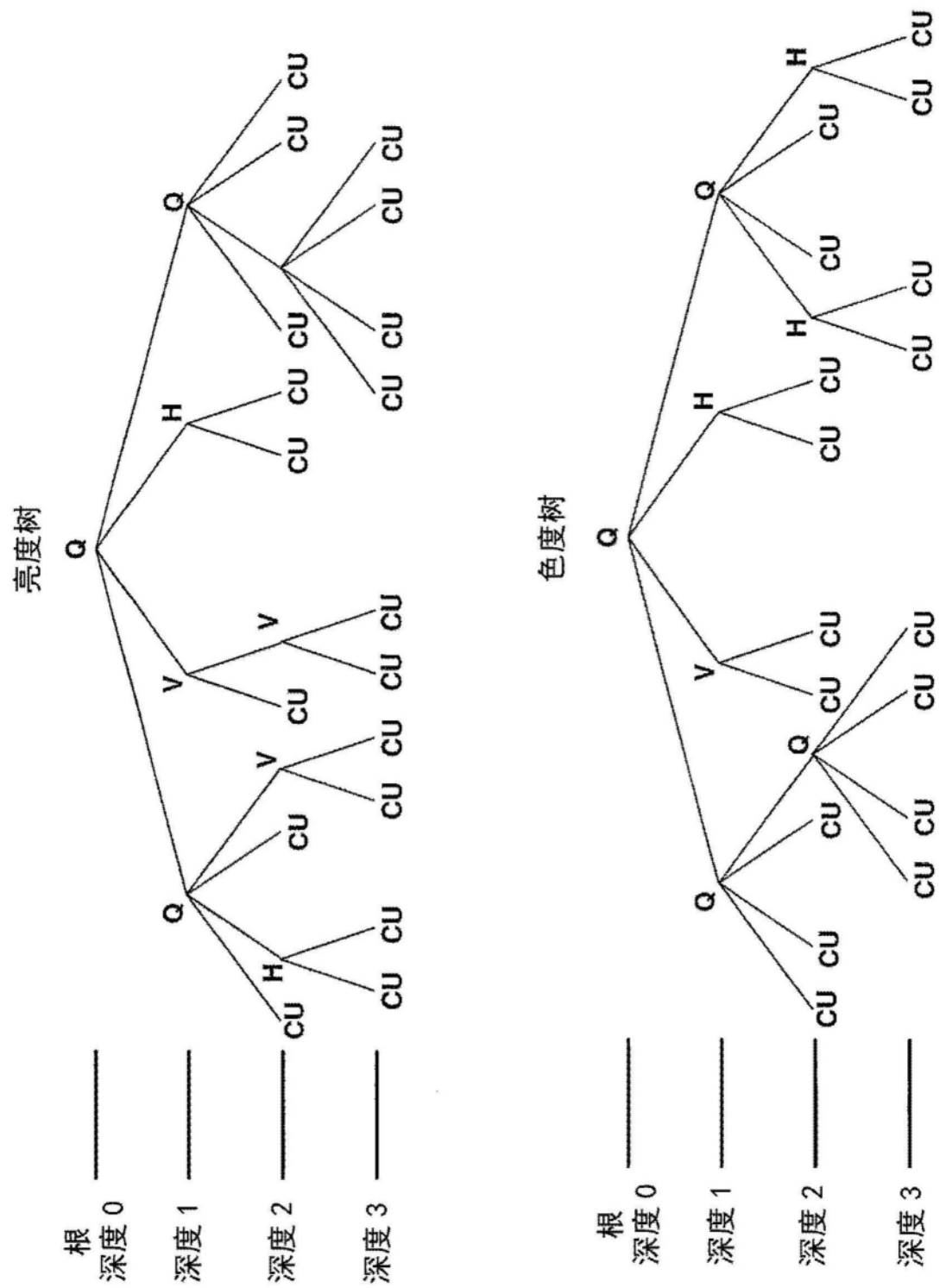


图11

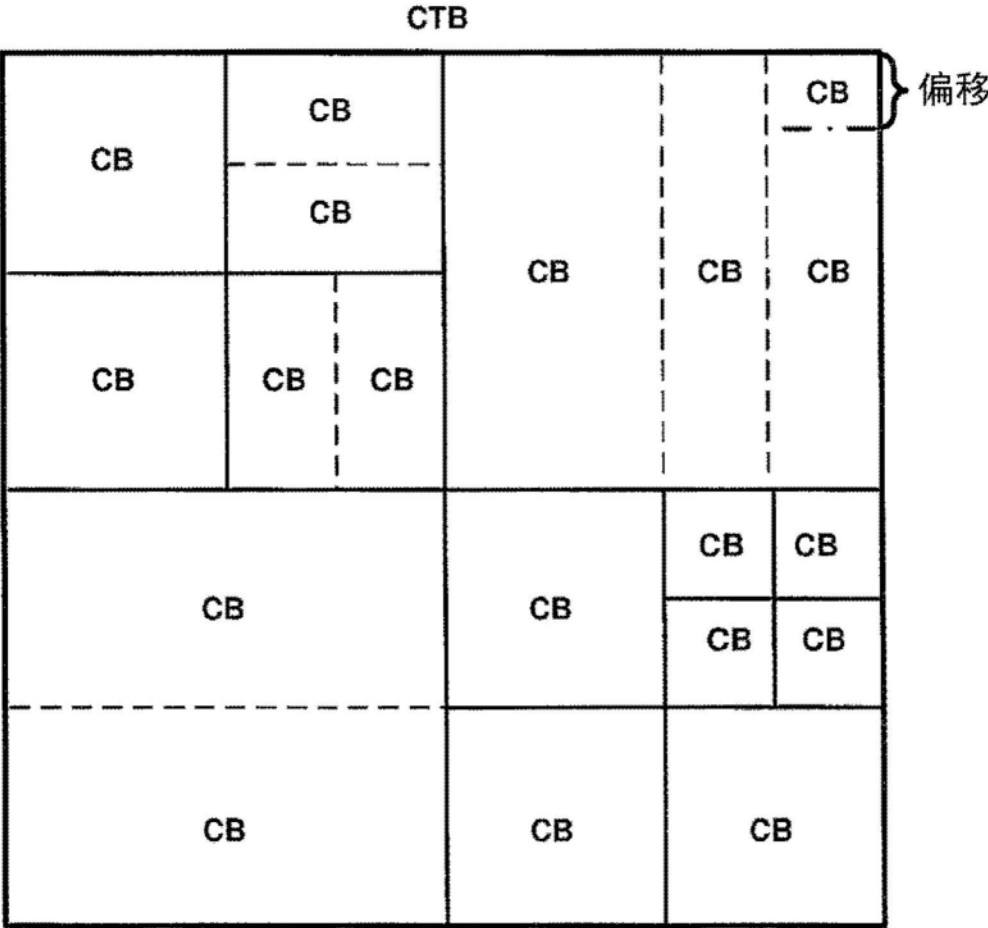


图12

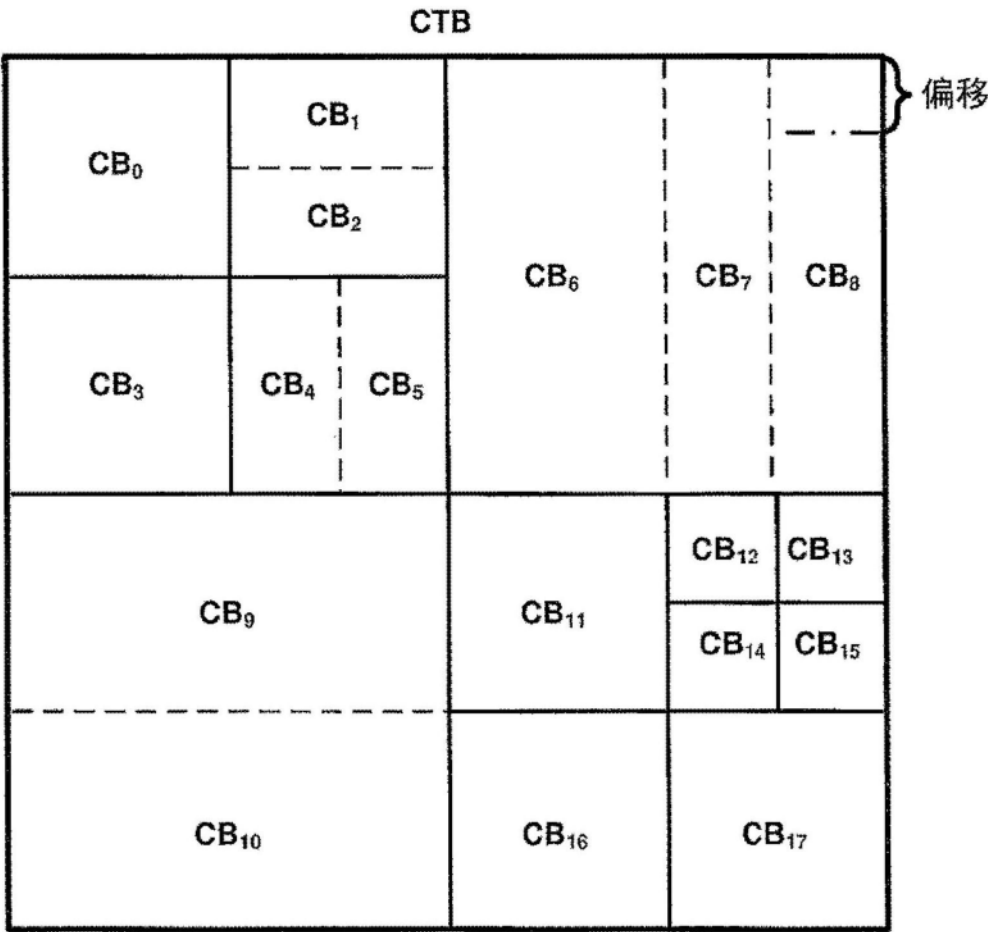


图13

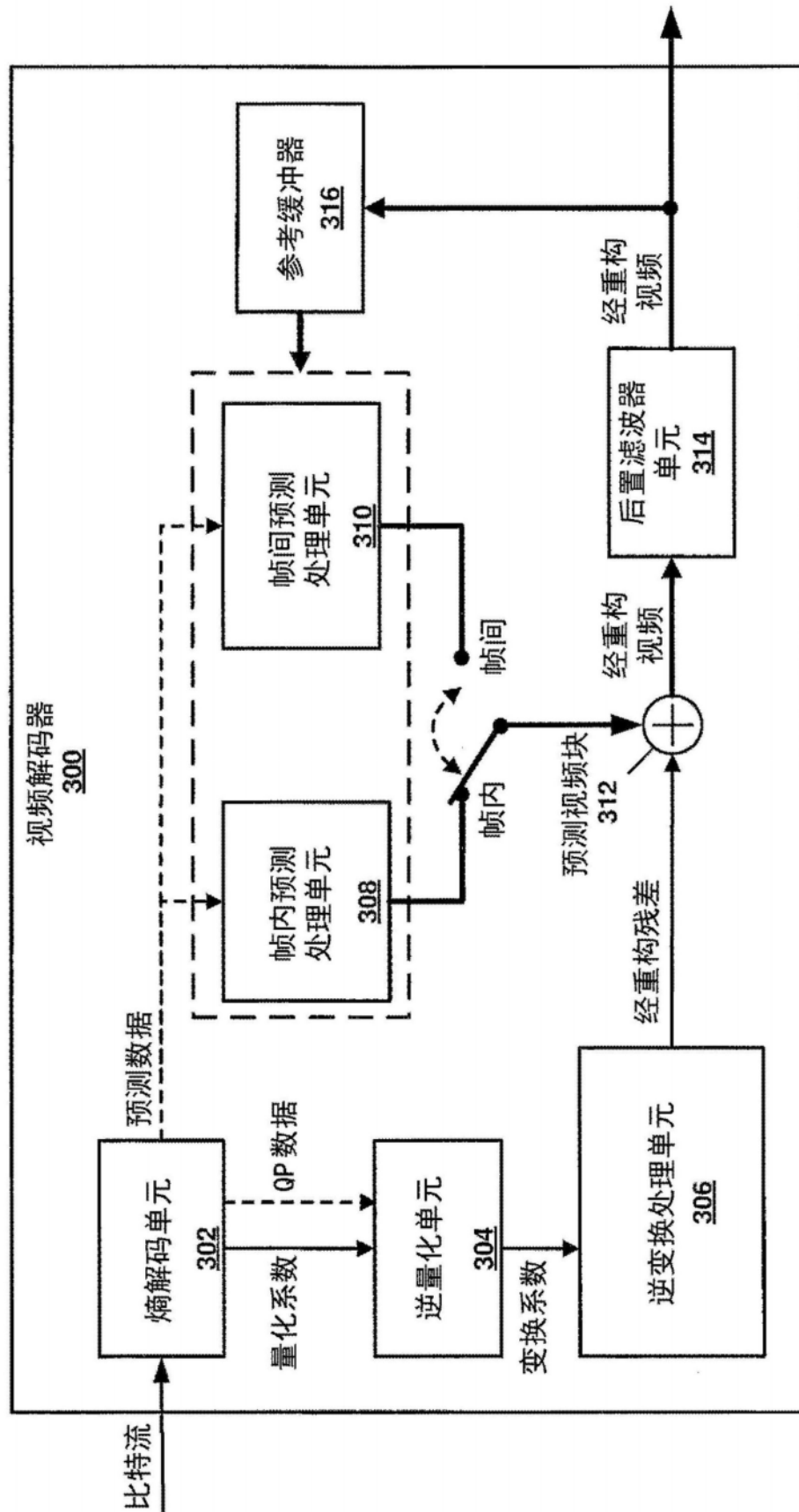


图14

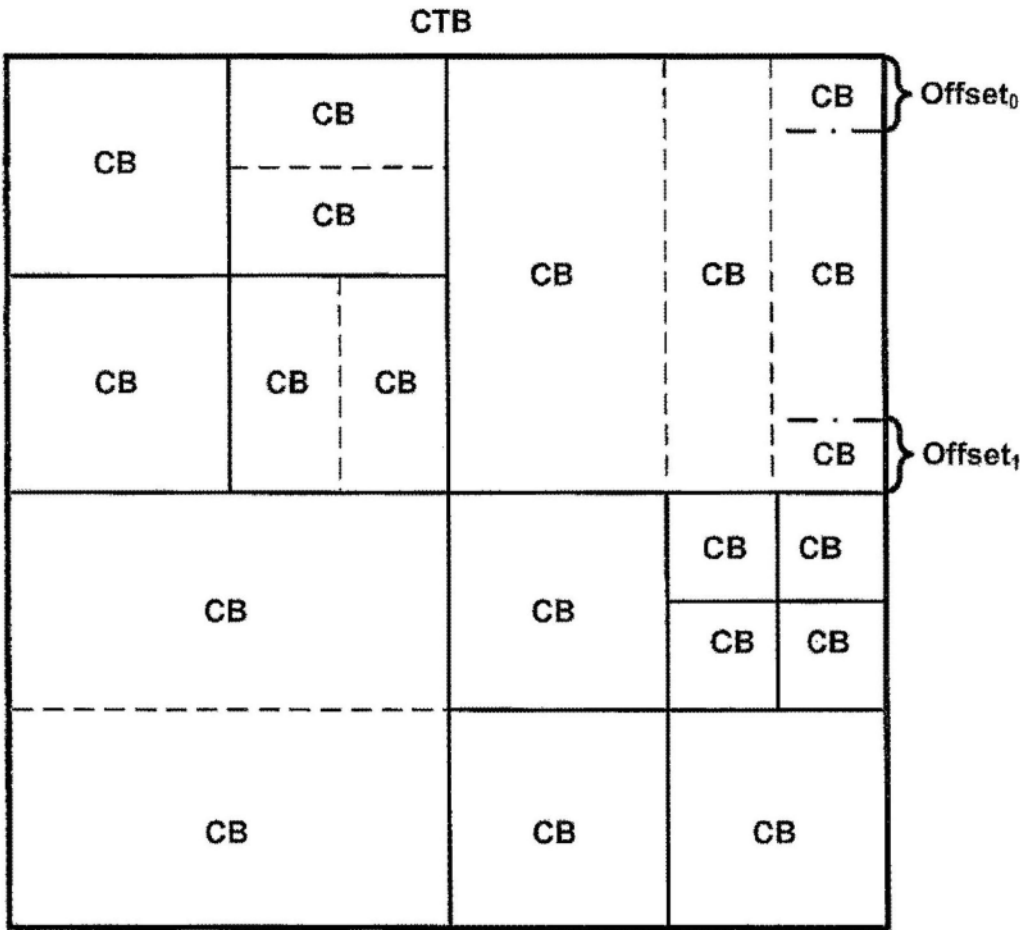


图15

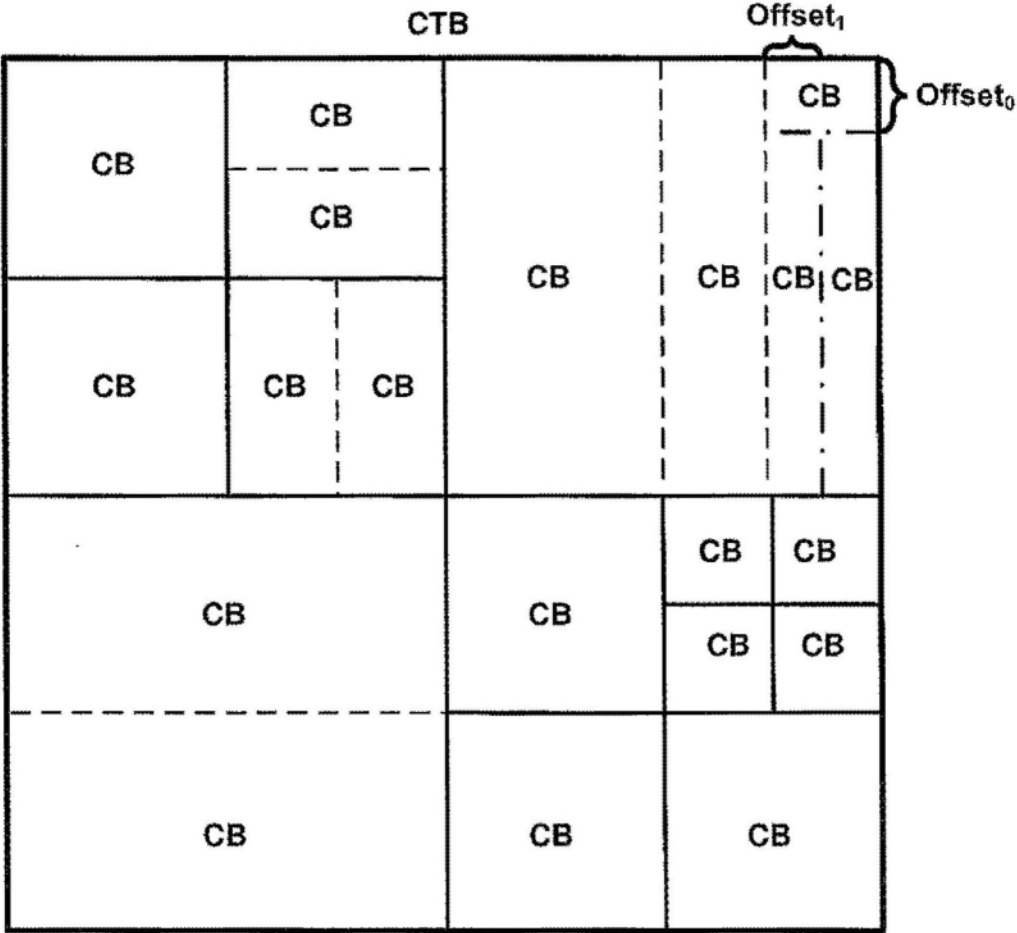


图16

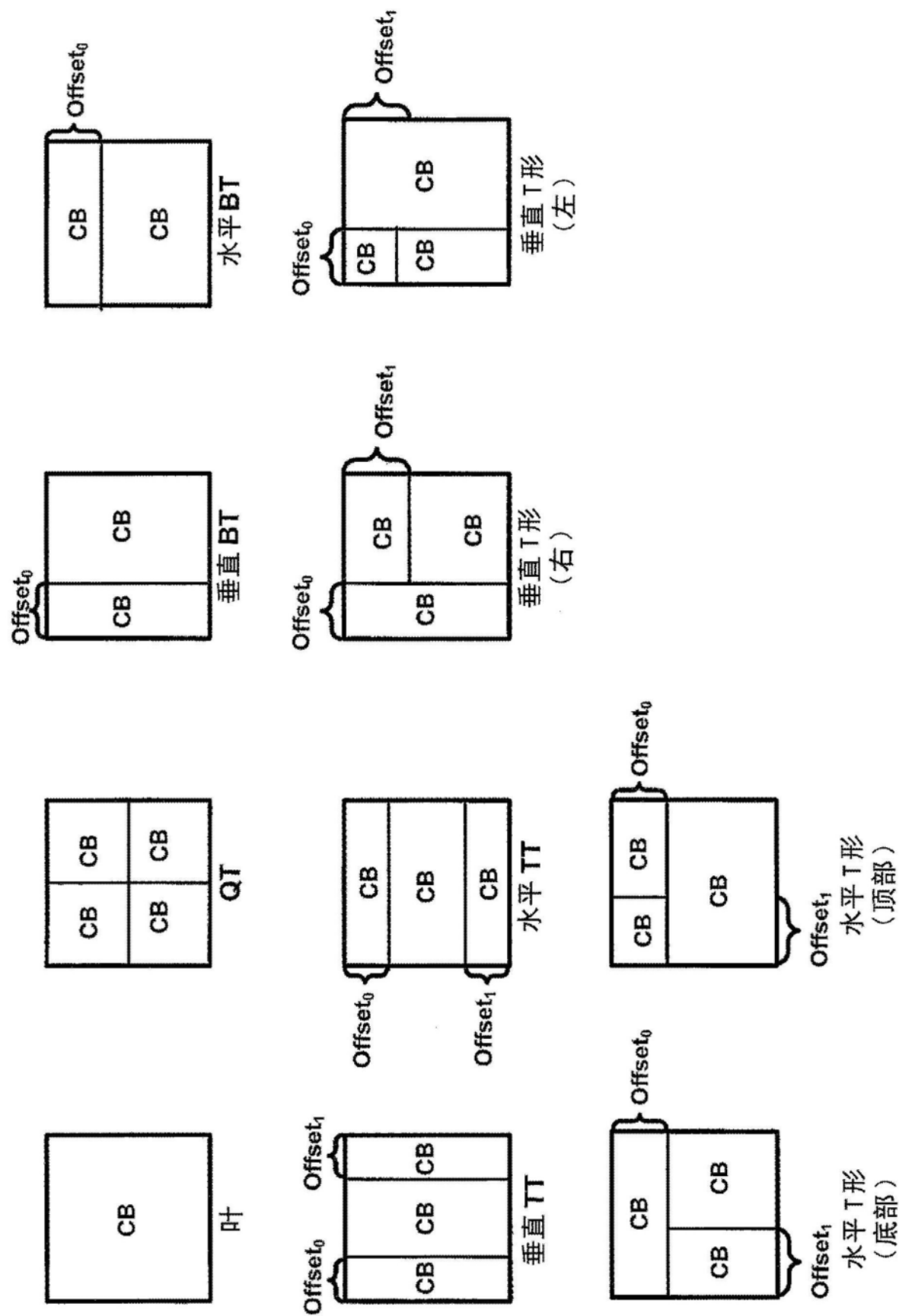


图17

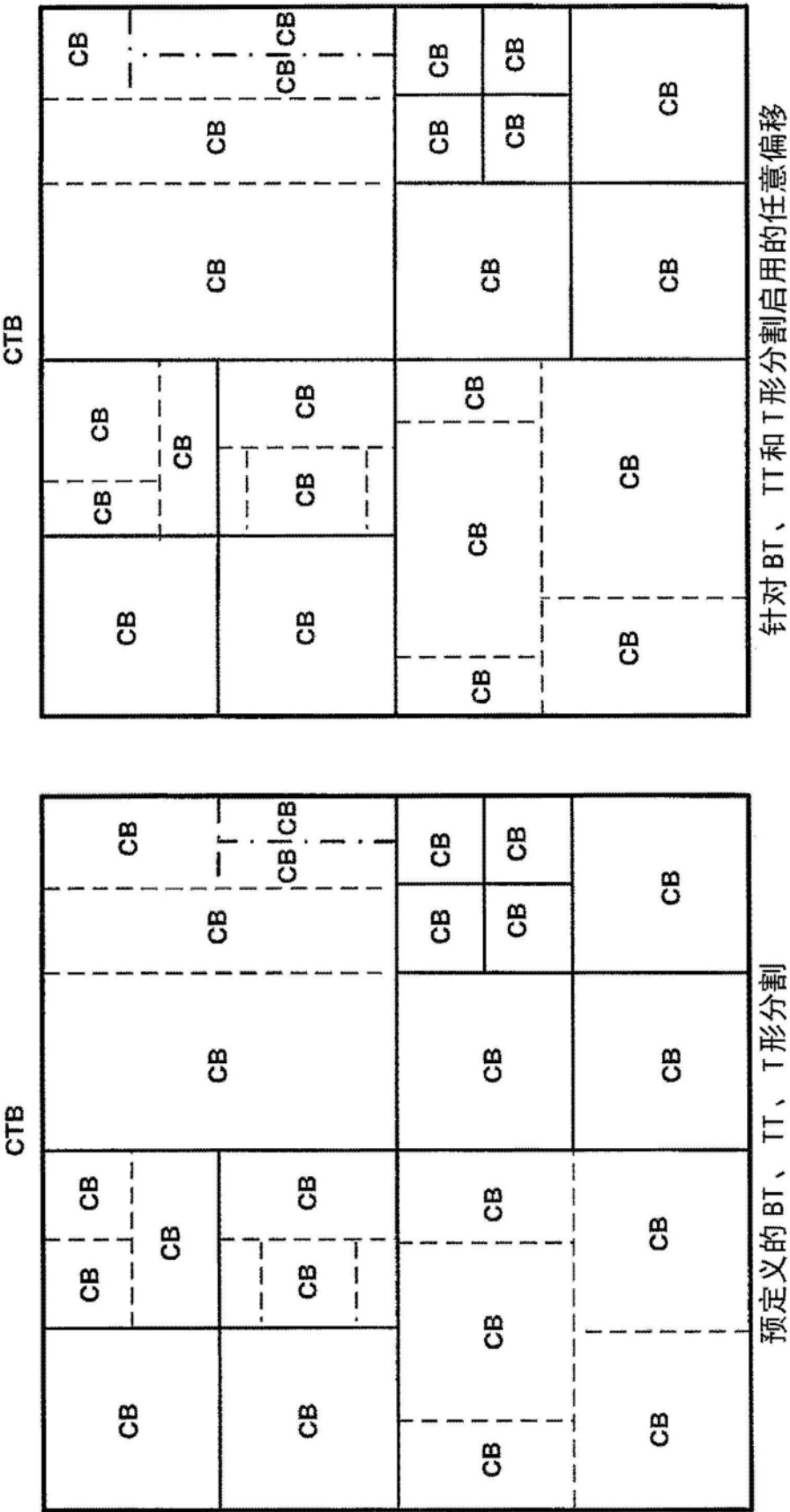


图18

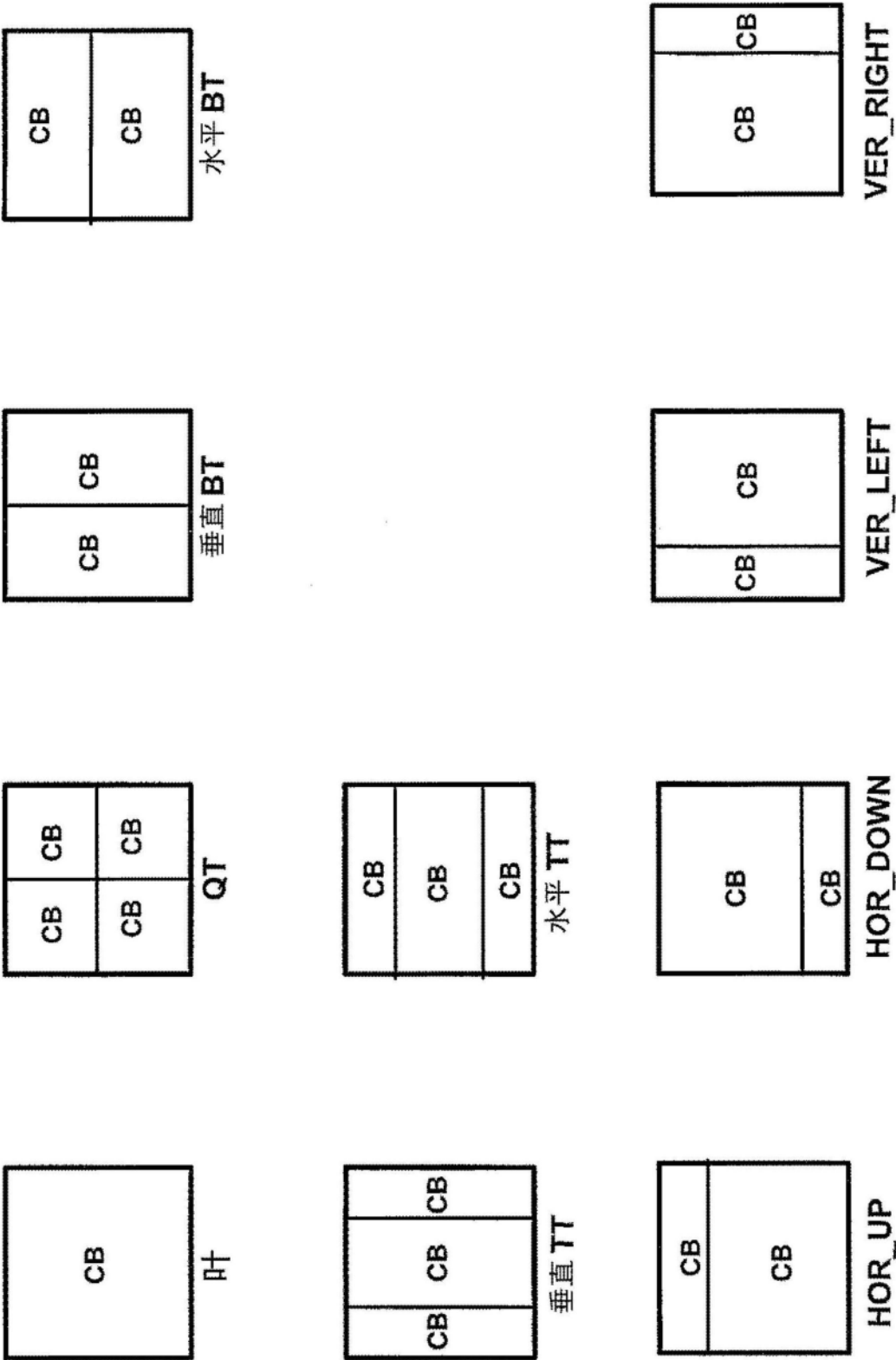


图19A

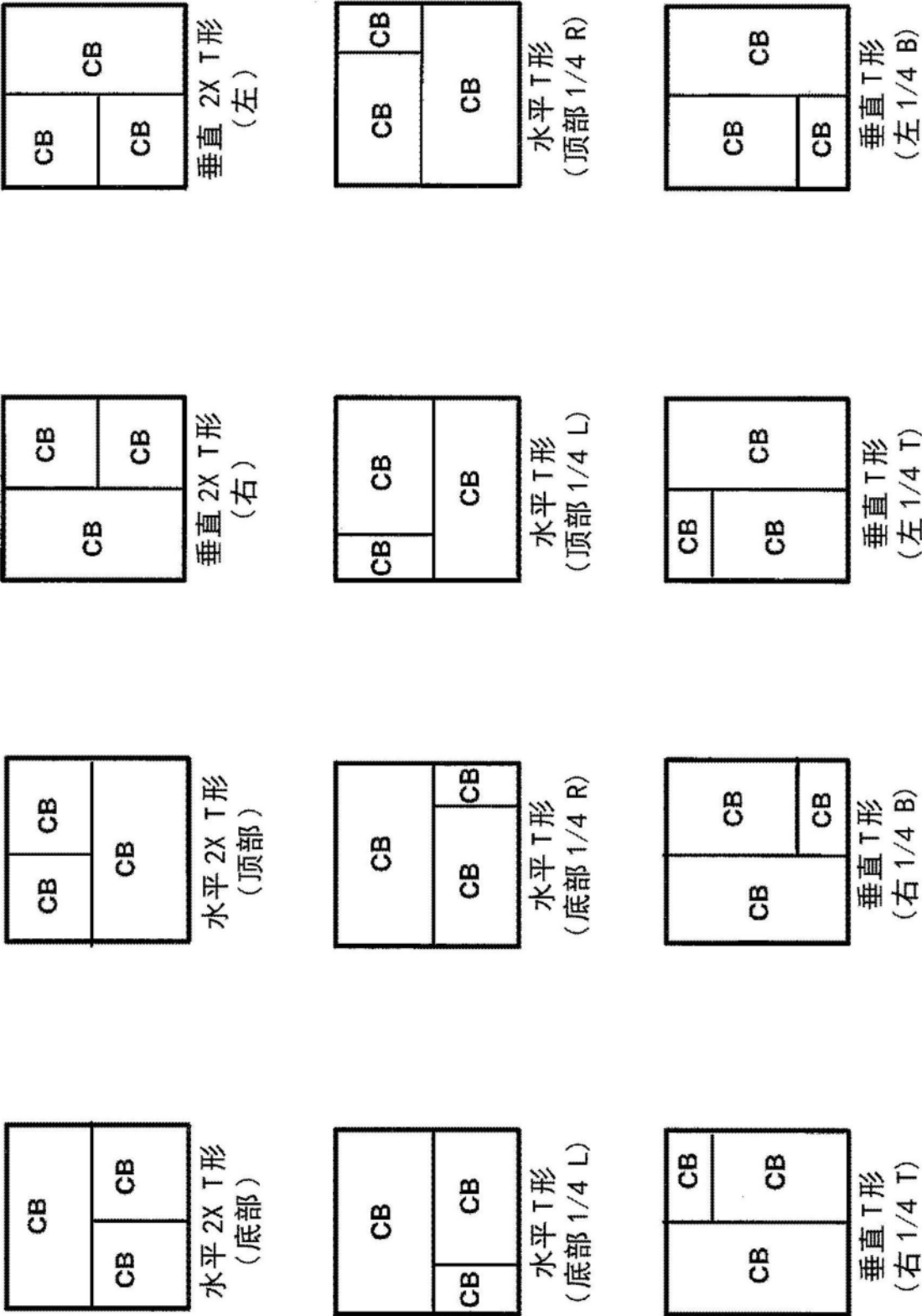


图19B

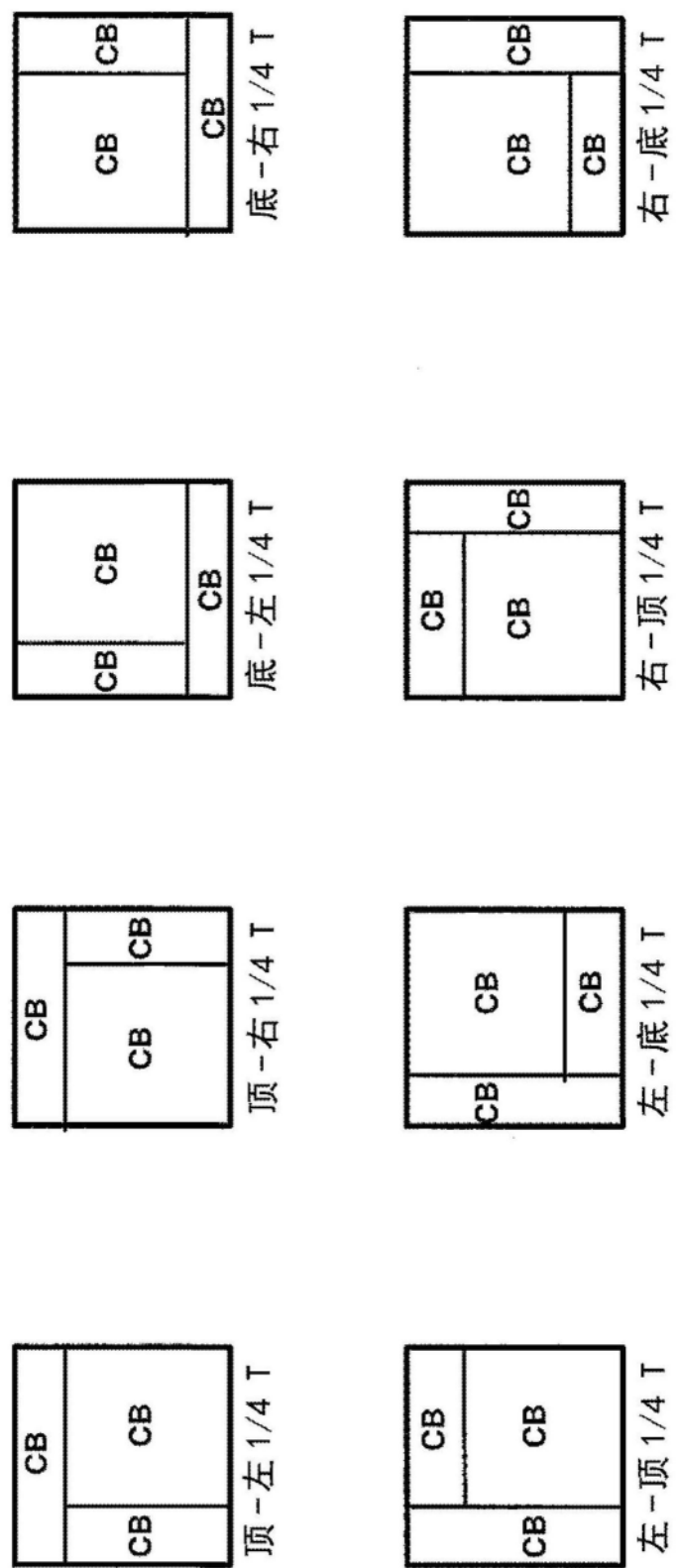
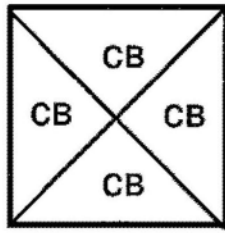
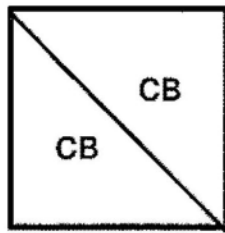


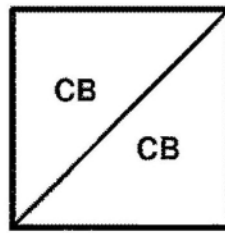
图19C



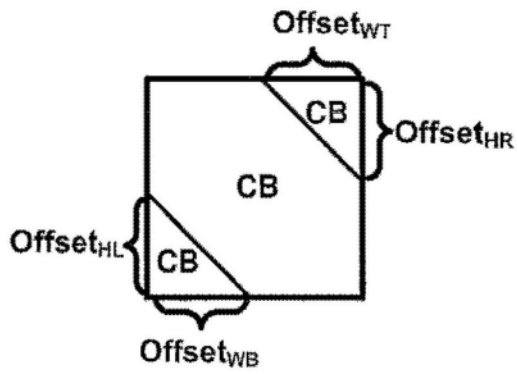
对角顶 - 左 QT



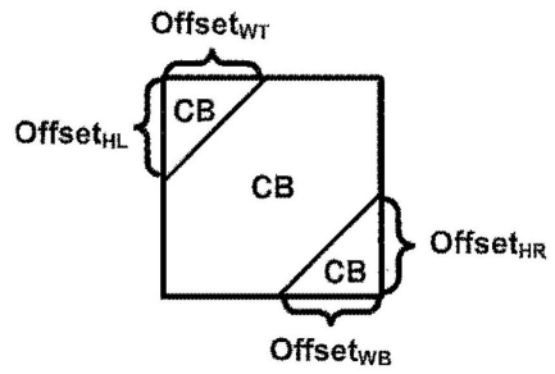
对角顶 - 左 BT



对角顶 - 右 BT



对角左 TT



对角右 TT

图20

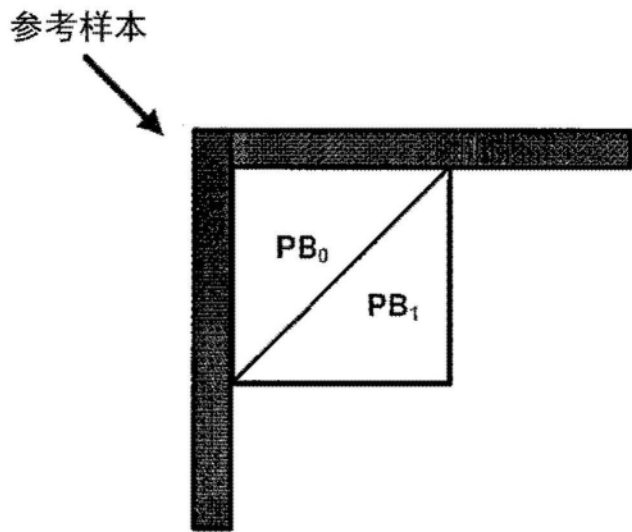


图21A

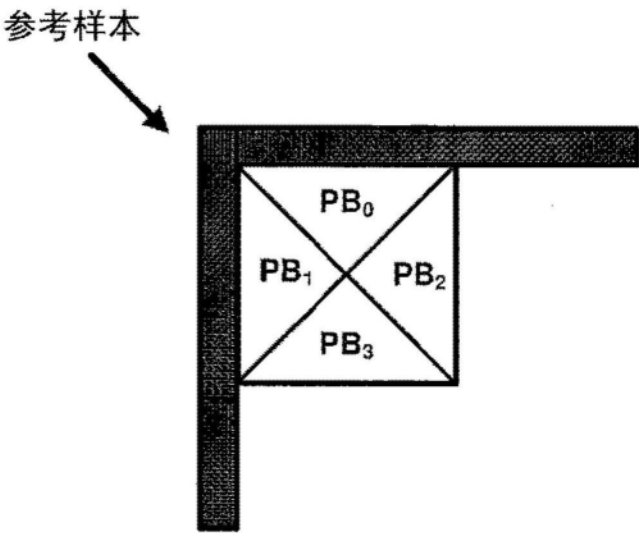


图21B

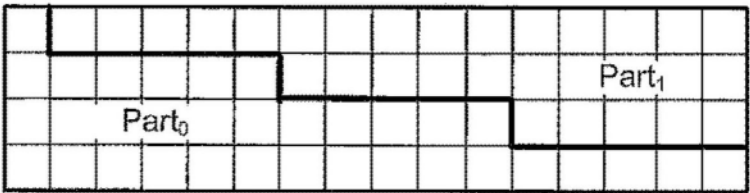


图22A

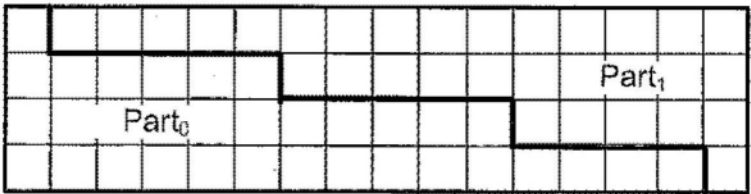


图22B

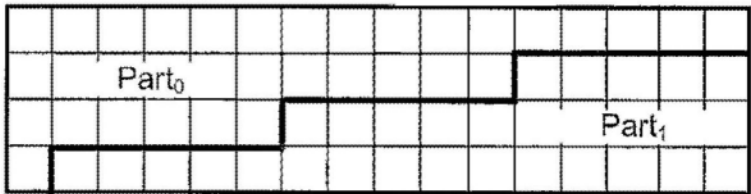


图22C

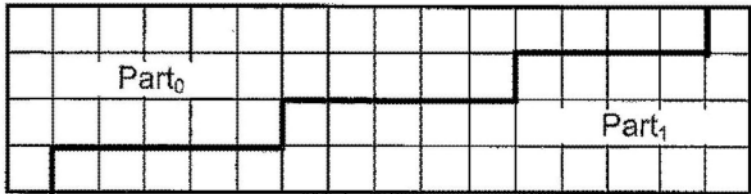


图22D

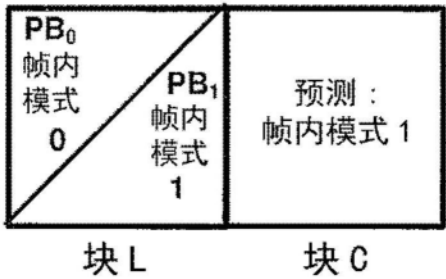


图23A

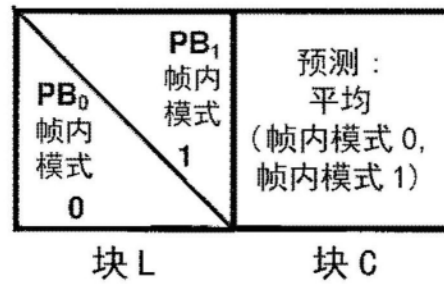


图23B

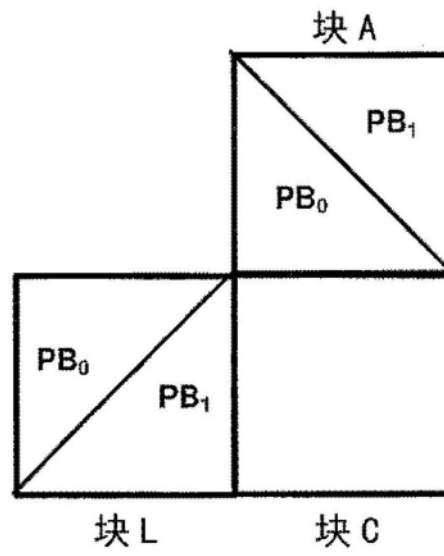


图23C