



(12) 发明专利申请

(10) 申请公布号 CN 115486074 A

(43) 申请公布日 2022. 12. 16

(21) 申请号 202180031987.1

(22) 申请日 2021.05.21

(30) 优先权数据

63/028,111 2020.05.21 US

(85) PCT国际申请进入国家阶段日

2022.11.02

(86) PCT国际申请的申请数据

PCT/US2021/033608 2021.05.21

(87) PCT国际申请的公布数据

WO2021/237063 EN 2021.11.25

(71) 申请人 阿里巴巴集团控股有限公司

地址 英属开曼群岛大开曼资本大厦一座四
层847号邮箱

(72) 发明人 陈杰 叶琰 廖如伶

(74) 专利代理机构 北京同钧律师事务所 16037

专利代理师 柴海平 许怀远

(51) Int.Cl.

H04N 19/40 (2006.01)

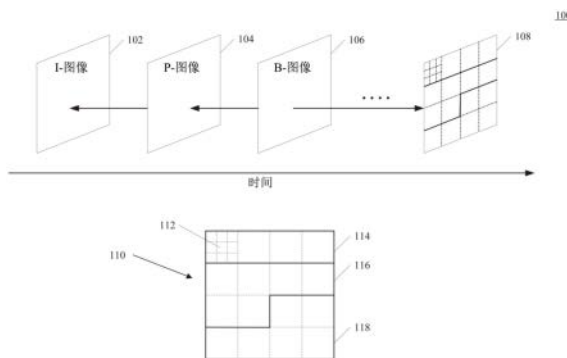
权利要求书3页 说明书38页 附图34页

(54) 发明名称

视频处理中的砖块和条带分割

(57) 摘要

本公开提供了一种用于编码或解码视频的计算机实现方法。该方法包括在与编码层视频序列 (CLVS) 的图像相关联的多个图像参数集 (PPS) 中, 编码或解码对应的第一 PPS 标志, 该对应的第一 PPS 标志指示图像是否被允许分割为多个砖块或条带。在第一 PPS 中, 具有第一值的对应的第一 PPS 标志指示 CLVS 的第一图像未分割, 并且在第二 PPS 中, 具有与第一值不同的第二值的另一对对应的第一 PPS 标志指示 CLVS 的第二图像被允许分割。



1. 一种用于编码或解码视频的方法,包括:

在与编码层视频序列CLVS的图像相关联的多个图像参数集PPS中,编码或解码对应的第一PPS标志,所述对应的第一PPS标志指示图像是否被允许分割为多个砖块或条带,其中:

在第一PPS中,具有第一值的对应的第一PPS标志指示所述CLVS的第一图像未分割,并且在第二PPS中,具有与所述第一值不同的第二值的另一对应的第一PPS标志指示所述CLVS的第二图像被允许分割。

2. 根据权利要求1所述的方法,还包括:

确定分割图像的砖块的数量是否大于阈值,所述阈值大于1;以及

响应于所述分割图像的砖块的数量大于所述阈值,编码或解码第二PPS标志,所述第二PPS标志与应用于所述分割图像的条带模式相关联。

3. 根据权利要求2所述的方法,其中,所述编码或解码所述第二PPS标志包括:

当光栅扫描条带模式应用于参考所述PPS的所述分割图像时,编码或解码具有第三值的所述第二PPS标志;或者当矩形条带模式应用于参考所述PPS的所述分割图像时,编码或解码具有不同于所述第三值的第四值的所述第二PPS标志。

4. 根据权利要求3所述的方法,还包括:

当所述矩形条带模式应用于参考所述PPS的所述分割图像时,编码或解码所述PPS中指定条带信息的一个或多个句法元素。

5. 一种装置,包括:

存储器,被配置为存储指令;以及

一个或多个处理器,被配置为执行所述指令以使所述装置:

确定图像是否被分割为多个砖块或条带;以及

响应于确定所述图像被允许分割为多个砖块或条带,在图像参数集PPS中编码或解码与应用于参考所述PPS的所述图像的条带模式相关联的第一标志,其中:

当光栅扫描条带模式应用于分割所述图像时,编码或解码具有第一值的第一标志;或者当矩形条带模式应用于分割所述图像时,编码或解码具有与所述第一值不同的第二值的所述第一标志。

6. 根据权利要求5所述的装置,其中:

在所述光栅扫描条带模式中,所述图像被分割为多个光栅扫描条带,并且所述光栅扫描条带中的任一个都包括所述图像的砖块光栅扫描中的一个或多个完整砖块的序列;并且

在所述矩形条带模式中,所述图像被分割为多个矩形条带,其中,所述矩形条带中的一个包括一个或多个覆盖所述图像的矩形区域的砖块,或者所述矩形条带中的一个覆盖砖块的编码树单元的一个或多个连续行。

7. 根据权利要求5所述的装置,其中,所述一个或多个处理器被配置为执行所述指令,以进一步使所述装置:

编码或解码第一图像参数,所述第一图像参数指定为对应砖块中的条带提供的条带高度的数量减去1,如果所述对应砖块包含单个条带,则所述第一图像参数等于0。

8. 根据权利要求7所述的装置,其中,所述第一图像参数的值在从0到所述对应砖块的高度减去2的包含范围内。

9. 根据权利要求5所述的装置,其中,所述一个或多个处理器被配置为执行所述指令,

以进一步使所述装置：

编码或解码第三图像参数，所述第三图像参数指定为对应砖块中的条带提供的砖块高度的数量，如果所述对应砖块包含单个条带，则所述第三图像参数等于0。

10. 根据权利要求9所述的装置，其中，所述一个或多个处理器被配置为执行所述指令，以进一步使所述装置：

编码或解码第二图像参数，所述第二图像参数指定所述对应砖块中对应矩形条带的高度减去1。

11. 根据权利要求10所述的装置，其中，所述第二图像参数的上限值是所述对应砖块的高度减去2。

12. 根据权利要求10所述的装置，其中，所述第二图像参数的上限值是所述对应砖块的高度减去第四图像参数，其中，所述第四图像参数指示为所述对应砖块中的条带提供的条带高度的数量。

13. 根据权利要求10所述的装置，其中，所述第二图像参数的上限值是所述对应砖块的高度减去第五图像参数，再减去1，

其中，所述第五图像参数指示为对应砖块中的条带提供的条带高度的数量减去1。

14. 根据权利要求10所述的装置，其中，所述第二图像参数的上限值是所述对应砖块的高度减去2或所述对应砖块的高度减去所述对应砖块中提供的条带高度的数量中的较小值。

15. 一种非暂时性计算机可读存储介质，存储一组指令，所述一组指令可由设备的一个或多个处理器执行，以使所述设备执行包括以下的操作：

在与编码层视频序列CLVS的至少一个图像相关联的图像参数集PPS中，编码或解码第一PPS标志，所述第一PPS标志指示关联图像是否被允许分割为多个砖块或条带，其中，第一PPS标志等于第一值指示所述关联图像未分割，或者第一PPS标志等于与所述第一值不同的第二值指示所述关联图像被允许分割；以及

当所述第一PPS标志等于第一值时，在所述PPS中跳过编码或解码第二PPS标志，所述第二PPS标志指示是否所述关联图像的每个子图像包括单个矩形条带，并确定所述第二PPS标志的值等于第三值，其中，等于所述第三值的所述第二PPS标志指示所述关联图像的每个子图像包括单个矩形条带。

16. 根据权利要求15所述的非暂时性计算机可读存储介质，其中，可由所述设备的所述一个或多个处理器执行的所述一组指令还使所述设备执行包括以下的操作：

对于索引*i*，在与编码层视频序列CLVS的至少一个图像相关联的图像参数集PPS中，编码或解码与第*i*个矩形条带的宽度相关联的第一图像参数；以及

当所述第一图像参数不存在所述PPS中时，确定所述第一图像参数为零，

其中，所述索引*i*在从0到等于所述图像中的矩形条带的数量减去2的上限值的范围内。

17. 根据权利要求16所述的非暂时性计算机可读存储介质，其中，可由所述设备的所述一个或多个处理器执行的所述一组指令还使所述设备执行包括以下的操作：

对于所述索引*i*，编码或解码与所述第*i*个矩形条带的高度相关联的第二图像参数；以及

当对于所述索引*i*，所述PPS中不存在所述第二图像参数时，确定所述第二图像参数为

零或等于与第(i-1)个矩形条带的高度相关联的第三图像参数的值。

18. 根据权利要求15所述的非暂时性计算机可读存储介质,其中可由所述设备的所述一个或多个处理器执行的所述一组指令还使所述设备执行包括以下的操作:

在比特流的序列参数集SPS中,编码或解码第一SPS标志,所述第一SPS标志指示参考所述SPS的所述CLVS中的一个或多个图像是否被分割为多个砖块或条带;以及

在与所述SPS相关联的所述PPS中,编码或解码等于所述第一SPS标志的第一PPS标志,所述第一PPS标志用于指示参考所述PPS的所述图像是否被分割。

19. 根据权利要求18所述的非暂时性计算机可读存储介质,其中,可由所述设备的所述一个或多个处理器执行的所述一组指令还使所述设备执行包括以下的操作:

响应于指示所述一个或多个图像未分割的所述第一SPS标志,跳过编码或解码所述SPS中的第二SPS标志,所述第二SPS标志指示对于参考所述SPS的所述CLVS,子图像信息是否存在。

20. 根据权利要求18所述的非暂时性计算机可读存储介质,其中,可由所述设备的所述一个或多个处理器执行的所述一组指令还使所述设备执行包括以下的操作:

响应于指示所述一个或多个图像未分割的所述第一SPS标志,跳过编码或解码与参考所述SPS的所述CLVS中的每个图像中的子图像的数量相关联的第一SPS序列参数。

视频处理中的砖块和条带分割

[0001] 相关申请的交叉引用

[0002] 本公开要求于2020年5月21日提交的申请号为63/028,111的美国临时申请的优先权,其全部内容通过引用并入本文。

技术领域

[0003] 本公开通常涉及视频数据处理,并且更具体地,涉及图像分割。

背景技术

[0004] 视频是一组捕捉视觉信息的静态图像(或“帧”)。为了减少存储内存和传输带宽,可以在存储或传输之前对视频进行压缩,然后在显示之前进行解压缩。压缩过程通常称为编码,解压缩过程通常称为解码。有各种使用标准化视频编码技术的视频编码格式,最常见的是基于预测、变换、量化、熵编码和环路滤波。视频编码标准,诸如高效视频编码(HEVC/H.265)标准、通用视频编码(VVC/H.266)标准和AVS标准等,规定了具体的视频编码格式,由标准化组织制定。随着越来越多的先进视频编码技术在视频标准中被采用,新的视频编码标准的编码效率也越来越高。

发明内容

[0005] 本公开的实施例提供了一种用于编码或解码视频的方法。该方法包括在与编码层视频序列(CLVS)的图像相关联的多个图像参数集(PPS)中,编码或解码对应的第一PPS标志,该第一PPS标志指示图像是否被允许分割为多个砖块或条带。在第一PPS中,具有第一值的对应的第一PPS标志指示CLVS的第一图像未分割,并且在第二PPS中,具有与第一值不同的第二值的另一对应的第一PPS标志指示CLVS的第二图像被允许分割。

[0006] 在一些实施例中,一种用于编码或解码视频的方法包括:确定图像是否被分割为多个砖块或条带;以及响应于确定图像被允许分割为多个砖块或条带,在图像参数集(PPS)中编码或解码与应用于参考该PPS的图像的条带模式相关联的第一标志,其中:当光栅扫描条带模式应用于分割该图像时,编码或解码具有第一值的第一标志;或者,当矩形条带模式应用于分割所述图像时,编码或解码具有与第一值不同的第二值的第一标志。

[0007] 在一些实施例中,解码视频的方法包括:在与编码层视频序列(CLVS)的至少一个图像相关联的图像参数集(PPS)中,编码或解码第一PPS标志,该第一PPS标志指示关联图像是否被允许分割为多个砖块或条带,其中,第一PPS标志等于第一值指示关联图像未分割,或者第一PPS标志等于与第一值不同的第二值指示关联图像被允许分割;以及,当第一PPS标志等于第一值时,在PPS中跳过编码或解码第二PPS标志,该第二PPS标志指示是否关联图像的每个子图像包括单个矩形条带,并确定第二PPS标志的值等于第三值,其中,等于第三值的第二PPS标志指示关联图像的每个子图像包括单个矩形条带。

[0008] 本公开的实施例提供了一种装置,包括存储器,被配置为存储指令;以及,一个或多个处理器,被配置为执行该指令以使该装置执行上述任何方法。

[0009] 本公开的实施例提供了一种非暂时性计算机可读存储介质。该非暂时性计算机可读存储介质存储可由设备的一个或多个处理器执行的一组指令,使得该设备执行上述任何方法的操作。

附图说明

[0010] 以下具体实施方式和附图中说明了本公开的实施例和各个方面。图中所示的各种特征不是按比例绘制的。

[0011] 图1是示出与本公开的一些实施例一致的示例视频序列的结构示意图。

[0012] 图2A是示出与本公开实施例一致的混合视频编码系统的示例性编码过程的示意图。

[0013] 图2B是示出与本公开实施例一致的混合视频编码系统的另一示例性编码过程的示意图。

[0014] 图3A是示出与本公开实施例一致的混合视频编码系统的示例性解码过程的示意图。

[0015] 图3B是示出与本公开实施例一致的混合视频编码系统的另一示例性解码过程的示意图。

[0016] 图4是示出与本公开的一些实施例一致的用于对视频进行编码或解码的示例性装置的框图。

[0017] 图5是与本公开的一些实施例一致的示例性比特流的示意图。

[0018] 图6是示出与本公开的一些实施例一致的分割为块的图像的结构示意图。

[0019] 图7是示出与本公开的一些实施例一致的以光栅扫描条带模式分割的图像的结构示意图。

[0020] 图8是示出与本公开的一些实施例一致的以矩形条带模式分割的图像的结构示意图。

[0021] 图9是示出与本公开的一些实施例一致的以矩形条带模式分割的图像的结构示意图。

[0022] 图10是示出与本公开的一些实施例一致的以矩形条带模式分割的图像的结构示意图。

[0023] 图11示出了与本公开的一些实施例一致的部分SPS句法结构的示例性编码句法表。

[0024] 图12示出了与本公开的一些实施例一致的部分PPS句法结构的示例性编码句法表。

[0025] 图13-23示出了与本公开的一些实施例一致的示例性伪代码。

[0026] 图24示出了与本公开的一些实施例一致的部分SPS句法结构的示例性编码句法表。

[0027] 图25示出了与本公开的一些实施例一致的部分SPS句法结构的另一示例性编码句法表。

[0028] 图26示出了与本公开的一些实施例一致的部分SPS句法结构的另一示例性编码句法表。

[0029] 图27A-27C示出了与本公开的一些实施例一致的示例性视频编码或解码方法的流程图。

[0030] 图28示出了与本公开的一些实施例一致的部分PPS句法结构的示例性的修改的编码句法表。

[0031] 图29示出了与本公开的一些实施例一致的示例性视频编码或解码方法的流程图。

[0032] 图30示出了与本公开的一些实施例一致的部分PPS句法结构的示例性的修改的编码句法表。

[0033] 图31A和图31B示出了与本公开的一些实施例一致的图29的方法中的步骤的示例性详细操作。

[0034] 图32示出了与本公开的一些实施例一致的示例性视频编码或解码方法的流程图。

[0035] 图33和图34示出了与本公开的一些实施例一致的示例性伪代码。

具体实施方式

[0036] 现在将详细参考示例性实施例,其示例在附图中示出。以下描述参照附图,除非另有说明,否则不同附图中的相同数字表示相同或相似的元件。在示例性实施例的以下描述中阐述的实施方式并不代表与本公开一致的所有实施方式。相反,它们仅仅是与如所附权利要求中所述的与本公开相关的方面一致的装置和方法的示例。下面更详细地描述本公开的特定方面。如果与通过引用并入的术语和/或定义相冲突,则以本文提供的术语和定义为准。

[0037] ITU-T视频编码专家组 (ITU-T VCEG) 和ISO/IEC运动图像专家组 (ISO/IEC MPEG) 的联合视频专家组 (JVET) 目前正在制定通用视频编码 (VVC/H.266) 标准。VVC标准旨在将其前身高效视频编码 (HEVC/H.265) 标准的压缩效率提高一倍。换句话说,VVC的目标是使用一半的带宽实现与HEVC/H.265相同的主观质量。

[0038] 为了使用一半的带宽实现与HEVC/H.265相同的主观质量,JVET一直在使用联合探索模型 (JEM) 参考软件开发HEVC之外的技术。随着编码技术被纳入JEM,JEM实现了比HEVC更高的编码性能。

[0039] VVC标准最近已经制定,并继续包括更多提供更好压缩性能的编码技术。VVC基于与现代视频压缩标准 (如HEVC、H.264/AVC、MPEG2、H.263等) 中使用的相同的混合视频编码系统。

[0040] 视频是一组按时间顺序排列以存储视觉信息的静态图像 (或“帧”)。视频捕获设备 (例如,相机) 可用于按时间顺序捕获和存储这些图像,并且视频播放设备 (例如,电视、计算机、智能手机、平板电脑、视频播放器或具有显示功能的任何最终用户终端) 可用于按时间顺序显示此类图像。此外,在一些应用中,视频捕获设备可以实时地将捕获的视频传输到视频播放设备 (例如,具有监视器的计算机),诸如用于监控、会议或直播等。

[0041] 为了减少此类应用所需的存储空间和传输带宽,可以在存储和传输之前对视频进行压缩,并在显示之前进行解压缩。压缩和解压缩可以由一个或多个处理器 (例如,通用计算机的一个或多个处理器) 或专用硬件执行的软件实现。用于压缩的模块通常称为“编码器”,用于解压缩的模块通常称为“解码器”,编码器和解码器可以统称为“编解码器”。编码器和解码器可以实现为各种合适的硬件、软件或其组合中的任何一种。例如,编码器和解码

器的硬件实现可以包括电路,例如一个或多个微处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、离散逻辑或其任何组合。编码器和解码器的软件实现可以包括程序代码、计算机可执行指令、固件或固定在计算机可读介质中的任何合适的计算机实现的算法或过程。视频压缩和解压缩可以通过各种算法或标准来实现,例如MPEG-1、MPEG-2、MPEG-4、H.26x系列等。在一些应用中,编解码器可以从第一编码标准解压缩视频,并使用第二编码标准重新压缩解压缩的视频,在这种情况下,编解码器可以称为“转码器”。

[0042] 视频编码过程可以识别并保留可用于重构图像的有用信息,并忽略对于重构不重要的信息。如果无法完全重构被忽略的、不重要的信息,则这种编码过程可以称为“有损”。否则,它可以被称为“无损”。大多数编码过程都是有损的,这是为了减少所需的存储空间和传输带宽而做出的权衡。

[0043] 正在被编码的图像(称为“当前图像”)的有用信息包括相对于参考图像(例如,先前编码和重构的图像)的变化。这种变化可以包括像素的位置变化、亮度变化或颜色变化,其中最关注的是位置变化。代表对象的一组像素的位置变化可以反映对象在参考图像和当前图像之间的运动。

[0044] 在不参考另一个图像的情况下编码的图像(即,它是自己的参考图像)被称为“I-图像”。如果图像中的一些或所有块(例如,通常指视频图像部分的块)是采用使用一个参考图像的帧内预测或帧间预测(例如,单预测)进行预测的,则该图像被称为“P-图像”。如果图像中的至少一个块是用两个参考图像进行预测的(例如,双向预测),则该图像被称为“B-图像”。

[0045] 在本公开中,可以修改与砖块/条带分割相关的SPS和PPS句法元素,以移除不必要的约束或确定被有条件地指示的句法元素的值,从而实现更高的编码性能。通过采用这些修改,可以提高视频流的编码和解码过程的一致性和效率。

[0046] 图1示出了根据本公开的一些实施例的示例视频序列100的结构。视频序列100可以是实况视频或已捕获并存档的视频。视频序列100可以是真实视频、计算机生成的视频(例如,计算机游戏视频)或其组合(例如,具有增强现实效果的真实视频)。视频序列100可以从视频捕获设备(例如,相机)、包含先前捕获的视频的视频存档(例如,存储在存储设备中的视频文件)或接收来自视频内容提供商的视频的视频馈送接口(例如,视频广播收发器)输入。

[0047] 如图1所示,视频序列100可以包括沿时间轴在时间上排列的一系列图像,包括图像102、104、106和108。图像102-106是连续的,并且在图像106和108之间有更多的图像。在图1中,图像102是I-图像,其参考图像是图像102本身。图像104是P-图像,其参考图像是图像102,如箭头所示。图像106是B-图像,其参考图像是图像104和108,如箭头所示。在一些实施例中,图像(例如,图像104)的参考图像可以不紧接在图像之前或之后。例如,图像104的参考图像可以是图像102之前的图像。应当注意,图像102-106的参考图像只是示例,本公开并不将参考图像的实施例限制为图1中所示的示例。

[0048] 通常,由于编解码任务的计算复杂性,视频编解码器不会一次对整个图像进行编码或解码。相反,他们可以将图像分割为基本段,并逐段对图像进行编码或解码。这样的基本段在本公开中被称为基本处理单元(BPU)。例如,图1中的结构110示出了视频序列100的

图像(例如,图像102-108中的任何一个)的示例结构。在结构110中,图像被划分为 4×4 个基本处理单元,其边界显示为虚线。在一些实施例中,基本处理单元可以在一些视频编码标准(例如,MPEG系列、H.261、H.263或H.264/AVC)中称为“宏块”,或者在一些其他视频编码标准(例如,H.265/HEVC或H.266/VVC)中称为“编码树单元”(CTU)。图像中的基本处理单元可以具有可变大小,例如 128×128 、 64×64 、 32×32 、 16×16 、 4×8 、 16×32 或像素的任何任意形状和大小。可以基于编码效率和要在基本处理单元中保持的细节水平的平衡来为图像选择基本处理单元的大小和形状。

[0049] 基本处理单元可以是逻辑单元,其可以包括存储在计算机存储器中(例如,在视频帧缓存区中)的一组不同类型的视频数据。例如,彩色图像的基本处理单元可以包括表示色差亮度信息的亮度分量(Y)、表示颜色信息的一个或多个色度分量(例如,Cb和Cr)以及相关句法元素,其中亮度分量和色度分量可以具有相同大小的基本处理单元大小。在一些视频编码标准(例如H.265/HEVC或H.266/VVC)中,亮度和色度分量可以称为“编码树块”(CTB)。对基本处理单元执行的任何操作都可以对其亮度和色度分量中的每一个重复执行。

[0050] 视频编码具有多个操作阶段,其示例在图2A-2B和图3A-3B中示出。对于每个阶段,基本处理单元的大小仍然可能太大而无法进行处理,因此可以进一步划分为本公开中称为“基本处理子单元”的段。在一些实施例中,基本处理子单元可以在一些视频编码标准(例如,MPEG系列、H.261、H.263或H.264/AVC)中称为“块”,或者在一些其他视频编码标准(例如,H.265/HEVC或H.266/VVC)中称为“编码单元”(CU)。基本处理子单元的大小可以与基本处理单元相同或更小。与基本处理单元类似,基本处理子单元也是逻辑单元,其可以包括存储在计算机存储器(例如,在视频帧缓存区中)中的一组不同类型的视频数据(例如,Y、Cb、Cr和相关句法元素)。对基本处理子单元执行的任何操作都可以对其亮度和色度分量中的每一个重复执行。应当注意,这种划分可以根据处理需要进一步进行。还应当注意,不同的阶段可以使用不同的方案划分基本处理单元。

[0051] 例如,在模式决定阶段(其示例在图2B中示出),编码器可以决定针对基本处理单元使用什么预测模式(例如,图像内预测或图像间预测),但基本处理单元可能太大而无法做出这样的决定。编码器可以将基本处理单元分割为多个基本处理子单元(例如,如H.265/HEVC或H.266/VVC中的CU),并为每个单独的基本处理子单元确定预测类型。

[0052] 对于另一个示例,在预测阶段(其示例在图2A-2B中示出),编码器可以在基本处理子单元(例如,CU)级别执行预测操作。然而,在某些情况下,基本处理子单元可能仍然太大而无法处理。编码器可以进一步将基本处理子单元分割为更小的段(例如,在H.265/HEVC或H.266/VVC中称为“预测块”或“PB”),在该段的级别上可以执行预测操作。

[0053] 对于另一个示例,在变换阶段(其示例在图2A-2B中示出),编码器可以对残差基本处理子单元(例如,CU)执行变换操作。然而,在某些情况下,基本处理子单元可能仍然太大而无法处理。编码器可以进一步将基本处理子单元分割为更小的段(例如,在H.265/HEVC或H.266/VVC中称为“变换块”或“TB”),在该段的级别上可以执行变换操作。应当注意,相同基本处理子单元的划分方案在预测阶段和变换阶段可以不同。例如,在H.265/HEVC或H.266/VVC中,相同CU的预测块和变换块可以具有不同的大小和数量。

[0054] 在图1的结构110中,基本处理单元112进一步划分为 3×3 个基本处理子单元,其边

界如虚线所示。同一图像的不同基本处理单元可以划分为不同方案的基本处理子单元。

[0055] 在一些实现方式中,为了提供对视频编码和解码的并行处理能力和容错能力,可以将图像划分为多个区域进行处理,使得对于图像的一个区域,编码或解码过程可以不依赖于来自图像任何其他区域的信息。换句话说,可以独立处理图像的每个区域。通过这样做,编解码器可以并行处理图像的不同区域,从而提高编码效率。此外,当一个区域的数据在处理中损坏或在网络传输中丢失时,编解码器可以正确地对同一图像的其他区域进行编码或解码,而不依赖于损坏或丢失的数据,从而提供容错能力。在一些视频编码标准中,可以将图像划分为不同类型的区域。例如,H.265/HEVC和H.266/VVC提供了两种类型的区域:“条带(slice)”和“砖块(tile)”。还应注意,视频序列100的不同图像可以具有用于将图像划分为多个区域的不同划分方案。

[0056] 例如,在图1中,结构110划分为三个区域114、116和118,其边界在结构110内显示为实线。区域114包括四个基本处理单元。区域116和118中的每一个均包括六个基本处理单元。应当注意,图1中的结构110的基本处理单元、基本处理子单元和区域只是示例,本公开不限制其实施例。

[0057] 图2A示出了与本公开实施例一致的示例编码过程200A的示意图。例如,编码过程200A可以由编码器执行。如图2A所示,编码器可以根据过程200A将视频序列202编码为视频比特流228。与图1中的视频序列100类似,视频序列202可以包括按时间顺序排列的一组图像(称为“原始图像”)。类似于图1中的结构110,视频序列202的每个原始图像可以由编码器划分为基本处理单元、基本处理子单元或区域进行处理。在一些实施例中,编码器可以在基本处理单元级别对视频序列202的每个原始图像执行过程200A。例如,编码器可以以迭代方式执行过程200A,其中编码器可以在过程200A的一次迭代中编码基本处理单元。在一些实施例中,编码器可以对视频序列202的每个原始图像的区域(例如,区域114-118)并行执行过程200A。

[0058] 在图2A中,编码器可以将视频序列202的原始图像的基本处理单元(称为“原始BPU”)馈送到预测阶段204,以生成预测数据206和预测的BPU208。编码器可以从原始BPU中减去预测的BPU 208以生成残差BPU210。编码器可以将残差BPU210馈送到变换阶段212和量化阶段214以生成量化的变换系数216。编码器可以将预测数据206和量化的变换系数216馈送到二进制编码阶段226以生成视频比特流228。组件202、204、206、208、210、212、214、216、226和228可以称为“前向路径”。在过程200A期间,在量化阶段214之后,编码器可以将量化的变换系数216馈送到逆量化阶段218和逆变换阶段220以生成重构的残差BPU222。编码器可以将重构的残差BPU222添加到预测的BPU208以生成预测参考224,该预测参考224用于过程200A的下一迭代迭代的预测阶段204中。过程200A的组件218、220、222和224可以称为“重构路径”。重构路径可用于确保编码器和解码器使用相同的参考数据进行预测。

[0059] 编码器可以迭代地执行过程200A以对原始图像的每个原始BPU进行编码(在前向路径中),并生成用于编码原始图像的下一个原始BPU的预测参考224(在重构路径中)。在对原始图像的所有原始BPU进行编码之后,编码器可以继续对视频序列202中的下一个图像进行编码。

[0060] 参考过程200A,编码器可以接收由视频捕获设备(例如,相机)生成的视频序列202。本文中使用的术语“接收”可指接收、输入、获取、检索、获得、读取、访问或以任何方式

输入数据的任何动作。

[0061] 在预测阶段204,在当前迭代中,编码器可以接收原始BPU和预测参考224,并执行预测操作以生成预测数据206和预测的BPU208。可以从过程200A的先前迭代的重构路径生成预测参考224。预测阶段204的目的是通过提取预测数据206来减少信息冗余,预测数据206可用于从预测数据206和预测参考224中将原始BPU重构为预测的BPU 208。

[0062] 理想情况下,预测的BPU208可以与原始BPU相同。然而,由于非理想的预测和重构操作,预测的BPU208通常与原始BPU略有不同。为了记录这种差异,在生成预测的BPU208之后,编码器可以将其从原始BPU中减去以生成残差BPU210。例如,编码器可以将预测的BPU208的像素的值(例如,灰度值或RGB值)从原始BPU的对应像素的值中减去。残差BPU210的每个像素可以具有残差值,该残差值是原始BPU和预测的BPU208的对应像素之间的这种减法的结果。与原始BPU相比,预测数据206和残差BPU 210可以具有更少的比特,但它们可以用于重构原始BPU,而不会出现显著的质量劣化。因此,原始BPU被压缩。

[0063] 为了进一步压缩残差BPU210,在变换阶段212,编码器可以通过将残差BPU210分解为一组二维“基本模式”来减少其空间冗余,每个基本模式与“变换系数”相关联。基本模式可以具有相同的大小(例如,残差BPU210的大小)。每个基本模式可以表示残差BPU210的变化频率(例如,亮度变化的频率)分量。任何基本模式都不能从任何其他基本模式的任何组合(例如,线性组合)再现。换言之,分解可以将残差BPU210的变化分解到频域中。这种分解类似于函数的离散傅里叶变换,其中基本模式类似于离散傅里叶变换的基本函数(例如,三角函数),变换系数类似于与基本函数相关的系数。

[0064] 不同的变换算法可以使用不同的基本模式。可以在变换阶段212使用各种变换算法,例如,离散余弦变换、离散正弦变换等。在变换阶段212的变换是可逆的。也就是说,编码器可以通过变换的逆运算(称为“逆变换”)来恢复残差BPU210。例如,为了恢复残差BPU210的像素,逆变换可以是将基本模式的对应像素的值乘以相应的相关系数,并将乘积相加以产生加权和。对于视频编码标准,编码器和解码器都可以使用相同的变换算法(因此使用相同的基本模式)。因此,编码器可能只记录变换系数,解码器可以从这些系数重构残差BPU210,而无需从编码器接收基本模式。与残差BPU210相比,变换系数可以具有更少的比特,但它们可以用于重构残差BPU210,而不会显著降低质量。因此,残差BPU210被进一步压缩。

[0065] 编码器可以在量化阶段214进一步压缩变换系数。在变换过程中,不同的基本模式可以表示不同的变化频率(例如,亮度变化频率)。由于人眼通常更善于识别低频变化,因此编码器可以忽略高频变化的信息,而不会导致解码质量显著下降。例如,在量化阶段214,编码器可以通过将每个变换系数除以整数值(称为“量化比例因子”)并将商舍入到其最接近的整数来生成量化的变换系数216。在这样的操作之后,高频基本模式的一些变换系数可以转换为零,低频基本模式的变换系数可以转换为较小的整数。编码器可以忽略零值的量化的变换系数216,通过该操作进一步压缩变换系数。量化过程也是可逆的,其中量化的变换系数216可以在量化的逆运算(称为“逆量化”)中重构为变换系数。

[0066] 因为编码器在舍入操作中忽略了这种除法的余数,所以量化阶段214可能是有损的。通常,量化阶段214可以在过程200A中造成最大的信息损失。信息损失越大,量化的变换系数216可能需要的比特越少。为了获得不同程度的信息丢失,编码器可以使用量化参数的

不同值或量化过程的任何其他参数。

[0067] 在二进制编码阶段226,编码器可以使用二进制编码技术(例如,熵编码、可变长度编码、算术编码、霍夫曼编码、上下文自适应二进制算术编码或任何其他无损或有损压缩算法)对预测数据206和量化的变换系数216进行编码。在一些实施例中,除了预测数据206和量化的变换系数216之外,编码器可以在二进制编码阶段226编码其他信息,例如,在预测阶段204使用的预测模式、预测操作的参数、在变换阶段212的变换类型,量化过程的参数(例如,量化参数)、编码器控制参数(例如,比特率控制参数)等。编码器可以使用二进制编码阶段226的输出数据来生成视频比特流228。在一些实施例中,视频比特流228可以进一步打包用于网络传输。

[0068] 参考过程200A的重构路径,在逆量化阶段218,编码器可以对量化的变换系数216执行逆量化以生成重构的变换系数。在逆变换阶段220,编码器可以基于重构的变换系数生成重构的残差BPU222。编码器可以将重构的残差BPU222添加到预测的BPU208,以生成将在过程200A的下一迭代中使用的预测参考224。

[0069] 应当注意,可以使用过程200A的其他变体对视频序列202进行编码。在一些实施例中,过程200A的各个阶段可以由编码器以不同的顺序执行。在一些实施例中,过程200A的一个或多个阶段可以组合成单个阶段。在一些实施例中,过程200A的单个阶段可分为多个阶段。例如,变换阶段212和量化阶段214可以组合成单个阶段。在一些实施例中,过程200A可以包括附加阶段。在一些实施例中,过程200A可以省略图2A中的一个或多个阶段。

[0070] 图2B示出了与本公开实施例一致的另一示例编码过程200B的示意图。过程200B可以修改自过程200A。例如,过程200B可由符合混合视频编码标准(例如,H.26x系列)的编码器使用。与过程200A相比,过程200B的前向路径另外包括模式决定阶段230,并将预测阶段204划分为空间预测阶段2042和时间预测阶段2044。过程200B的重构路径另外包括环路滤波器阶段232和缓存器234。

[0071] 通常,预测技术可以分为两类:空间预测和时间预测。空间预测(例如,图像内预测或“帧内预测”)可以使用来自同一图像中一个或多个已编码的相邻BPU的像素来预测当前BPU。即,空间预测中的预测参考224可以包括相邻的BPU。空间预测可以减少图像固有的空间冗余。时间预测(例如,图像间预测或“帧间预测”)可以使用来自一个或多个已编码的图像的区域来预测当前BPU。即,时间预测中的预测参考224可以包括已编码的图像。时间预测可以减少图像的固有时间冗余。

[0072] 参考过程200B,在前向路径中,编码器在空间预测阶段2042和时间预测阶段2044执行预测操作。例如,在空间预测阶段2042,编码器可以执行帧内预测。对于正在被编码的图像的原始BPU,预测参考224可以包括在同一图像中已被编码(在前向路径中)和重构(在重构路径中)的一个或多个相邻BPU。编码器可以通过外推相邻BPU来生成预测的BPU 208。外推技术可以包括例如线性外推或内插、多项式外推或内插等。在一些实施例中,编码器可以在像素级执行外推,诸如通过为预测的BPU208的每个像素对对应像素的值进行外推。用于外推的相邻BPU可以相对于原始BPU位于各种方向,例如在垂直方向(例如,在原始BPU的顶部)、水平方向(例如,在原始BPU的左侧)、对角方向(例如,在原始BPU的左下、右下、左上或右上),或在所使用的视频编码标准中定义的任何方向。对于帧内预测,预测数据206可以包括例如所使用的相邻BPU的位置(例如,坐标)、所使用的相邻BPU的大小、外推参数、所使

用的相邻BPU相对于原始BPU的方向等。

[0073] 对于另一示例,在时间预测阶段2044,编码器可以执行帧间预测。对于当前图像的原始BPU,预测参考224可以包括已经编码(在前向路径中)和重构(在重构路径中)的一个或多个图像(称为“参考图像”)。在一些实施例中,可以逐个BPU地对参考图像进行编码和重构。例如,编码器可以将重构的残差BPU 222添加到预测的BPU208以生成重构的BPU。当生成同一图像的所有重构BPU时,编码器可以生成重构图像作为参考图像。编码器可以执行“运动估计”操作,以在参考图像的范围(称为“搜索窗口”)中搜索匹配区域。参考图像中搜索窗口的位置可以基于原始BPU在当前图像中的位置来确定。例如,搜索窗口可以以参考图像中与当前图像中的原始BPU具有相同坐标的位置为中心,并且可以向外扩展预定距离。当编码器(例如,通过使用像素递归算法、块匹配算法等)识别出搜索窗口中与原始BPU类似的区域时,编码器可以确定此类区域作为匹配区域。匹配区域可以具有与原始BPU不同的尺寸(例如,小于、等于、大于或形状不同)。由于参考图像和当前图像在时间轴上是时间分离的(例如,如图1所示),因此可以认为匹配区域随着时间的推移“移动”到原始BPU的位置。编码器可以将这种运动的方向和距离记录为“运动矢量”。当使用多个参考图像(例如,如图1中的图像106)时,编码器可以针对每个参考图像搜索匹配区域并确定其相关运动矢量。在一些实施例中,编码器可以将权重分配给各个匹配参考图像的匹配区域的像素值。

[0074] 运动估计可用于识别各种类型的运动,例如平移、旋转、缩放等。对于帧间预测,预测数据206可以包括例如匹配区域的位置(例如,坐标)、与匹配区域相关联的运动矢量、参考图像的数量、与参考图像相关联的权重等。

[0075] 为了生成预测的BPU208,编码器可以执行“运动补偿”的操作。运动补偿可用于基于预测数据206(例如,运动矢量)和预测参考224重构预测的BPU208。例如,编码器可以根据运动矢量移动参考图像的匹配区域,其中编码器可以预测当前图像的原始BPU。当使用多个参考图像时(例如,如图1中的图像106),编码器可以根据匹配区域的各自的运动矢量和平均像素值来移动参考图像的匹配区域。在一些实施例中,如果编码器已将权重分配给各个匹配参考图像的匹配区域的像素值,则编码器可以将移动的匹配区域的像素值的加权和相加。

[0076] 在一些实施例中,帧间预测可以是单向的或双向的。单向帧间预测可以使用在相对于当前图像相同的时间方向上的一个或多个参考图像。例如,图1中的图像104是单向帧间预测图像,其中参考图像(例如,图像102)在图像104之前。双向帧间预测可以在相对于当前图像的两个时间方向上使用一个或多个参考图像。例如,图1中的图像106是双向帧间预测图像,其中参考图像(例如,图像104和108)位于相对于图像104的两个时间方向。

[0077] 仍然参考过程200B的前向路径,在空间预测2042和时间预测阶段2044之后,在模式决定阶段230,编码器可以为过程200B的当前迭代选择预测模式(例如,帧内预测或帧间预测之一)。例如,编码器可以执行率失真优化技术,其中编码器可以根据候选预测模式的比特率和候选预测模式下重构的参考图像的失真来选择预测模式,以最小化代价函数的值。根据选择的预测模式,编码器可以生成相应的预测的BPU208和预测数据206。

[0078] 在过程200B的重构路径中,如果在前向路径中选择了帧内预测模式,则在生成预测参考224(例如,当前图像中已编码和重构的当前BPU)之后,编码器可以直接将预测参考224馈送到空间预测阶段2042以供以后使用(例如,用于外推当前图像的下一个BPU)。编码

器可以将预测参考224馈送到环路滤波器阶段232,在该阶段,编码器可以将环路滤波器应用于预测参考224,以减少或消除在预测参考224的编码期间引入的失真(例如,块伪影)。编码器可以在环路滤波器阶段232应用各种环路滤波器技术,例如去块、采样自适应偏移(SAO)、自适应环路滤波器(ALF)等。环路滤波的参考图像可以存储在缓存区234(或“解码图像缓存区(DPB)”)中以供以后使用(例如,用作视频序列202的未来图像的帧间预测参考图像)。编码器可以将一个或多个参考图像存储在缓存区234中,以在时间预测阶段2044使用。在一些实施例中,编码器可以在二进制编码阶段226对环路滤波器的参数(例如,环路滤波器强度)以及量化的变换系数216、预测数据206和其他信息进行编码。

[0079] 图3A示出了与本公开实施例一致的示例解码过程300A的示意图。过程300A可以是对应于图2A中的压缩过程200A的解压缩过程。在一些实施例中,过程300A可以类似于过程200A的重构路径。解码器可以根据过程300A将视频比特流228解码为视频流304。视频流304可以与视频序列202非常类似。然而,由于压缩和解压缩过程中的信息丢失(例如,图2A-2B中的量化阶段214),通常,视频流304与视频序列202不同。类似于图中的过程200A和200B,解码器可以在基本处理单元(BPU)级对视频比特流228中编码的每个图像执行过程300A。例如,解码器可以以迭代方式执行过程300A,其中解码器可以在过程300A的一次迭代中解码基本处理单元。在一些实施例中,解码器可以针对视频比特流228中编码的每个图像的区域(例如,区域114-118)并行执行过程300A。

[0080] 在图3A中,解码器可以将与编码图像的基本处理单元(称为“编码BPU”)相关联的视频比特流228的一部分馈送到二进制解码阶段302。在二进制解码阶段302,解码器可以将该部分解码为预测数据206和量化的变换系数216。解码器可以将量化的变换系数216馈送到逆量化阶段218和逆变换阶段220以生成重构的残差BPU222。解码器可以将预测数据206馈送到预测阶段204以生成预测的BPU208。解码器可以将重构的残差BPU222添加到预测的BPU208以生成预测参考224。在一些实施例中,预测参考224可以存储在缓存区中(例如,计算机存储器中的解码图像缓存区)。解码器可以将预测参考224馈送到预测阶段204,以在过程300A的下一迭代中执行预测操作。

[0081] 解码器可以迭代地执行过程300A以对编码图像的每个编码BPU进行解码,并生成用于对编码图像的下一个编码BPU进行编码的预测参考224。在对编码图像的所有编码BPU进行解码之后,解码器可以将图像输出到视频流304以进行显示,并继续解码视频比特流228中的下一个编码图像。

[0082] 在二进制解码阶段302,解码器可以执行编码器使用的二进制编码技术(例如,熵编码、可变长度编码、算术编码、哈夫曼编码、上下文自适应二进制算术编码或任何其他无损压缩算法)的逆运算。在一些实施例中,除了预测数据206和量化的变换系数216之外,解码器可以在二进制解码阶段302解码其他信息,例如,预测模式、预测操作的参数、变换类型、量化过程的参数(例如,量化参数)、编码器控制参数(例如,比特率控制参数)等。在一些实施例中,如果视频比特流228通过网络以包的形式传输,则解码器可以在将视频比特流228馈送到二进制解码阶段302之前对其进行解包。

[0083] 图3B示出了与本公开实施例一致的另一示例解码过程300B的示意图。过程300B可以修改自过程300A。例如,过程300B可由符合混合视频编码标准(例如,H.26x系列)的解码器使用。与过程300A相比,过程300B另外将预测阶段204分为空间预测阶段2042和时间预测

阶段2044,并且另外包括环路滤波器阶段232和缓存区234。

[0084] 在过程300B中,对于正在被解码的编码图像(称为“当前图像”)的编码基本处理单元(称为“当前BPU”),由解码器从二进制解码阶段302解码的预测数据206可以包括各种类型的数据,这取决于编码器使用什么预测模式对当前BPU进行编码。例如,如果编码器使用帧内预测对当前BPU进行编码,则预测数据206可以包括指示帧内预测的预测模式指示符(例如,标志值)、帧内预测操作的参数等。帧内预测操作的参数可以包括,例如,用作参考的一个或多个相邻BPU的位置(例如,坐标)、相邻BPU的大小、外推参数、相邻BPU相对于原始BPU的方向等。对于另一示例,如果编码器使用帧间预测对当前BPU进行编码,则预测数据206可以包括指示帧间预测的预测模式指示符(例如,标志值)、帧间预测操作的参数等。帧间预测操作的参数可以包括,例如,与当前BPU相关联的参考图像的数量、分别与参考图像相关联的权重、各个参考图像中一个或多个匹配区域的位置(例如,坐标),分别与匹配区域相关联的一个或多个运动矢量等。

[0085] 基于预测模式指示符,解码器可以决定是在空间预测阶段2042执行空间预测(例如,帧内预测),还是在时间预测阶段2044执行时间预测(例如,帧间预测)。图2B中描述了执行这种空间预测或时间预测的细节,下文不再重复。在执行这种空间预测或时间预测之后,解码器可以生成预测的BPU208。如图3A所述,解码器可以将预测的BPU 208和重构的残差BPU222相加以生成预测参考224。

[0086] 在过程300B中,解码器可以将预测参考224馈送到空间预测阶段2042或时间预测阶段2044,以在过程300B的下一代迭代中执行预测操作。例如,如果在空间预测阶段2042使用帧内预测对当前BPU进行解码,则在生成预测参考224(例如,解码的当前BPU)之后,解码器可以直接将预测参考224馈送到空间预测阶段2042以供以后使用(例如,用于外推当前图像的下一个BPU)。如果在时间预测阶段2044使用帧间预测对当前BPU进行解码,则在生成预测参考224(例如,其中所有BPU已被解码的参考图像)之后,解码器可以将预测参考224馈送到环路滤波器阶段232以减少或消除失真(例如,块伪影)。解码器可以如图2B所述的方式将环路滤波器应用于预测参考224。经环路滤波的参考图像可以存储在缓存区234中(例如,计算机存储器中的解码图像缓存区(DPB))以供以后使用(例如,用作视频比特流228的未来编码图像的帧间预测参考图像)。解码器可以将一个或多个参考图像存储在缓存区234中,以在时间预测阶段2044使用。在一些实施例中,预测数据还可以包括环路滤波器的参数(例如,环路滤波器强度)。在一些实施例中,当预测数据206的预测模式指示符指示帧间预测被用于编码当前BPU时,预测数据包括环路滤波器的参数。来自缓冲区234的重构图像也可以发送到显示器,诸如电视、PC、智能手机或平板电脑,以供最终用户观看。

[0087] 图4是与本公开实施例一致的用于编码或解码视频的示例装置400的框图。如图4所示,装置400可以包括处理器402。当处理器402执行本文所述的指令时,装置400可以成为用于视频编码或解码的专用机器。处理器402可以是能够操纵或处理信息的任何类型的电路。例如,处理器402可以包括任意数量的中央处理单元(或“CPU”)、图形处理单元(或“GPU”)、神经处理单元(“NPU”)、微控制器单元(“MCU”)、光学处理器、可编程逻辑控制器、微控制器、微处理器、数字信号处理器、知识产权(IP)核、可编程逻辑阵列(PLA)、可编程阵列逻辑(PAL)、通用阵列逻辑(GAL)、复杂可编程逻辑器件(CPLD)、现场可编程门阵列(FPGA)、片上系统(SoC)、专用集成电路(ASIC)等的任意组合。在一些实施例中,处理器402还可以是

分组为单个逻辑组件的一组处理器。例如,如图4所示,处理器402可以包括多个处理器,包括处理器402a、处理器402b和处理器402n。

[0088] 装置400还可以包括被配置为存储数据(例如,一组指令、计算机代码、中间数据等)的存储器404。例如,如图4所示,存储的数据可以包括程序指令(例如,用于实现过程200A、200B、300A或300B中的阶段的程序指令)和用于处理的数据(例如,视频序列202、视频比特流228或视频流304)。处理器402可以访问程序指令和用于处理的数据(例如,通过总线410),并执行程序指令以对用于处理的数据执行操作或控制。存储器404可以包括高速随机存取存储设备或非易失性存储设备。在一些实施例中,存储器404可以包括任意数量的随机存取存储器(RAM)、只读存储器(ROM)、光盘、磁盘、硬盘驱动器、固态驱动器、闪存驱动器、安全数字(SD)卡、记忆棒、紧凑闪存(CF)卡等的任意组合。存储器404也可以是分组为单个逻辑组件的一组存储器(图4中未示出)。

[0089] 总线410可以是在装置400内的组件之间传输数据的通信设备,诸如内部总线(例如,CPU-存储器总线)、外部总线(例如,通用串行总线端口、外围组件互连快速端口)等。

[0090] 为了便于解释而不引起歧义,处理器402和其他数据处理电路在本公开中统称为“数据处理电路”。数据处理电路可以完全实现为硬件,或软件、硬件或固件的组合。此外,数据处理电路可以是单个独立模块,或者可以完全或部分组合到装置400的任何其他组件中。

[0091] 装置400还可以包括网络接口406,以提供与网络(例如,互联网、内联网、局域网、移动通信网络等)的有线或无线通信。在一些实施例中,网络接口406可以包括任意数量的网络接口控制器(NIC)、射频(RF)模块、转发器、收发器、调制解调器、路由器、网关、有线网络适配器、无线网络适配器、蓝牙适配器、红外适配器、近场通信(“NFC”)适配器、蜂窝网络芯片等的任意组合。

[0092] 在一些实施例中,可选地,装置400还可以包括外围接口408,以提供到一个或多个外围设备的连接。如图4所示,外围设备可以包括但不限于光标控制设备(例如,鼠标、触摸板或触摸屏)、键盘、显示器(例如,阴极射线管显示器、液晶显示器或发光二极管显示器)、视频输入设备(例如,相机或耦合到视频档案的输入接口)等。

[0093] 应当注意,视频编解码器(例如,执行过程200A、200B、300A或300B的编解码器)可以实现为装置400中任何软件或硬件模块的任何组合。例如,过程200A、200B、300A或300B的一些或所有阶段可以被实现为装置400的一个或多个软件模块,诸如可以加载到存储器404中的程序指令。再例如,过程200A、200B、300A或300B的一些或所有阶段可以实现为装置400的一个或多个硬件模块,例如专用数据处理电路(诸如,FPGA、ASIC、NPU等)。

[0094] 图5是与本公开的一些实施例一致的由编码器编码的比特流500的示例的示意图。在一些实施例中,比特流500的结构可以应用于图2A-2B和图3A-3B中所示的视频比特流228。在图5中,比特流500包括视频参数集(VPS)510、序列参数集(SPS)520、图像参数集(PPS)530、图像头540、条带550-570,这些由同步标记M1-M7分隔。条带550-570每个都包括相应的头块(例如,头552)和数据块(例如,数据554),每个数据块包括一个或多个CTU(例如,数据554中的CTU1-CTUn)。

[0095] 根据一些实施例,比特流500是网络抽象层(NAL)单元或字节流形式的比特序列,比特流500形成一个或多个编码视频序列(CVS)。CVS包括一个或多个编码层视频序列(CLVS)。在一些实施例中,CLVS是图像单元(PU)序列,每个PU包含一个编码图像。特别地,PU

包括零个或一个包含图像头句法结构作为有效负载的图像头NAL单元(例如,图像头540),一个包括一个或多个视频编码层(VCL)NAL单元的编码图像,以及,可选地,一个或多个其他非VCL NAL单元。VCL NAL单元是编码条带NAL单元(例如,条带550-570)和具有在一些实施例中分类为VCL NAL单元的NAL单元类型的保留值的NAL单元子集的统称。编码条带NAL单元包含条带头和条带数据块(例如,头552和数据554)。

[0096] 换句话说,在本公开的一些实施例中,层可以是一组具有特定NAL层ID值的视频编码层(VCL)NAL单元和相关联的非VCL NAL单元。在这些层中,可以在不同层之间应用层间预测以实现高压缩性能。

[0097] 如上所述,在通用视频编码(例如VVC/H.266)标准中,可以将图像划分为一组CTU,多个CTU可以形成一个砖块、一个条带或子图像。当图像包括用于存储三个颜色分量(例如,一个亮度分量和两个色度分量)的三个样本阵列时,CTU可以包括 $N \times N$ (N 为整数)个亮度样本块,每个亮度样本块与两个色度样本块相关联。在一些实施例中,可以指定输出层集(OLS)以支持对一些但不是所有层进行解码。OLS是包括一组指定的层的一组层,其中该组层中的一个或多个层被指定为输出层。因此,OLS可以包含一个或多个输出层以及解码输出层所需的用于层间预测的其他层。

[0098] 举例来说,图6是示出与本公开的一些实施例一致的分割为块的图像600的结构示意图。在图6中,每个正方形代表一个CTU 610,并且图像600被分割为 8×6 个CTU 610。在一些实施例中,CTU中亮度块的最大允许大小为 128×128 ,而亮度变换块的最大允许大小为 64×64 。在一些实施例中,CTU中亮度块的最小允许大小为 32×32 。应当注意,亮度块的最大允许大小、亮度变换块的最大允许大小及亮度块的最小允许大小可以在各种视频编码标准中指定为不同的值和不同的形状,并且本公开并不将其限制于上述示例。

[0099] 与本公开的一些实施例一致,图像可以被分割为一个或多个砖块行和一个或多个砖块列。本公开中的“砖块”可以指覆盖图像的矩形区域的CTU序列。本公开中的“条带”可以包括整数个完整的砖块,或者包括图像的砖块内的整数个连续的完整CTU行。

[0100] 在一些实施例中,可以用两种方式将图像分割为条带,即“光栅扫描条带模式”和“矩形条带模式”。在光栅扫描条带模式中,图像的条带可以包括按照图像的光栅扫描顺序的完整砖块的序列。在矩形条带模式中,图像的条带可以包括共同形成图像的矩形区域的数个完整砖块,或者包括砖块的数个连续的完整CTU行,其中这些CTU行共同形成图像的矩形区域。矩形条带内的砖块可以在与矩形条带相对应的所形成的矩形区域内以光栅扫描顺序进行扫描。

[0101] 作为示例,图7是示出与本公开的一些实施例一致的以光栅扫描条带模式分割的图像700的结构示意图。在图7中,每个虚线正方形表示一个CTU,并且将图像700分割为 16×14 个CTU。图像700的CTU形成十二个砖块(例如,砖块712-716、722-726、732-736和742-746),包括四个砖块行和三个砖块列,其边界由与虚线重叠的较细实线表示。此外,图像700被划分为3个以不同阴影表示的光栅扫描条带,其边界由与虚线或较细实线重叠的较粗实线表示。如图7所示,第一条带包括砖块712和714。第二条带包括砖块716、722-726和732-734。第三条带包括砖块736和742-746。图像700的三个条带按光栅扫描顺序进行分割,并且三个条带中的每一个都包括整数个完整的砖块。

[0102] 作为示例,图8是示出与本公开的一些实施例一致的以矩形条带模式分割的图像

800的结构示意图。在图8中,每个虚线正方形表示一个CTU,并且将图像800分割为 16×14 个CTU。图800的CTU形成二十个砖块,包括四个砖块行和五个砖块列,其边界由与虚线重叠的较细实线表示。此外,图像800被划分为9个以不同阴影表示的矩形条带,其边界由与虚线或较细实线重叠的较粗实线表示。如图8所示,以形成九个矩形区域的矩形方式分割图像800的九个条带,并且九个条带中的每一个都包括整数个完整的砖块。例如,第一条带包括砖块812和814。第二条带包括砖块816和818。第三条带包括砖块819。第四条带包括砖块822、824、832和834。第五条带包括砖块826、828、836和838。第六条带包括砖块829和839。第七条带包括砖块842和844。第八条带包括砖块846和848。第九条带包括砖块849。

[0103] 作为示例,图9是示出与本公开的一些实施例一致的以矩形条带模式分割的图像900的结构示意图。在图9中,每个虚线正方形表示一个CTU,图900被分割为 16×14 个CTU。图900的CTU形成四个砖块910、920、930和940,包括两个砖块行和两个砖块列,其边界由虚线表示。例如,第一砖块910可以位于左上,大小为 7×10 个CTU。第二砖块920可以位于左下,大小为 7×4 个CTU。第三砖块930可以位于右上,大小为 9×10 个CTU。第四砖块940可以位于右下,大小为 9×4 个CTU。此外,图像900被划分为4个以不同阴影表示的矩形条带,其边界由与虚线或较细实线重叠的较粗实线表示。如图9所示,图像900的四个条带以形成四个矩形区域的矩形方式进行分割,并且四个条带中的每一个都包括整数个完整砖块或整数个连续完整的CTU行,这些行位于图像800的砖块内。例如,第一条带(以白色表示)可以包括两个完整的砖块910和920,大小为 7×14 个CTU。第二条带(以灰色表示)可以包括砖块930的一部分(例如,部分932),大小为 9×4 个CTU。第三条带(以白色表示)可以包括砖块930的另一部分(例如,部分934),其大小为 9×6 个CTU。第四条带(以灰色表示)可以包括完整的砖块940,大小为 9×4 个CTU。

[0104] 作为示例,图10是示出与本公开的一些实施例一致的以矩形条带模式分割的图像1000的结构示意图。在图10中,每个虚线正方形表示一个CTU,图像1000被分割为 16×16 个CTU。图像1000的CTU形成20个砖块1012-1019、1022-1029、1032-1039和1042-1049,包括四个砖块行和五个砖块列,其边界由与虚线重叠的较粗实线表示。如图10所示,左手侧的12块砖块(例如,砖块1012-1016、1022-1026、1032-1036和1042-1046),每个覆盖 4×4 个CTU的一个条带。右手侧的8块砖块(例如1018、1019、1028、1029、1038、1039、1048和1049),每个覆盖2个垂直堆叠的 2×2 个CTU的条带,得到28个以不同的阴影表示的条带,其中每个条带是一个子图像。例如,砖块1018覆盖垂直堆叠的条带1018a和1018b,砖块1028覆盖垂直堆叠的条带1028a和1028b,依此类推。条带/子图像的边界由较粗的虚线表示。

[0105] 在一些实施例中,子图像布局或子图像分割可以在序列参数集(SPS)中进行指示。图11示出了与本公开的一些实施例一致的用于指示子图像布局的部分SPS句法结构1100的示例性编码句法表。图11中所示的伪代码可以是VVC标准的一部分。

[0106] 在图11中,当SPS标志1110(“sps_subpic_info_present_flag(sps_子图像_信息_存在_标志)”)等于1时,可以指定,对于CLVS,子图像信息存在,并且CLVS的每个图像中可能有一个或多于一个子图像。与上述公开一致,CLVS是属于同一层的一组图像,从随机接入点开始,并且随后是可能相互依赖并依赖随机接入点图像的图像。当SPS标志1110等于0时,对于CLVS,子图像信息不存在,并且CLVS的每个图像中只有一个子图像。在一些实施例中,SPS标志“sps_res_change_in_clvs_allowed_flag(sps_res_更改_in_clvs_允许_标志)”等于

1指定SPS标志1110的值等于0。当比特流是子比特流提取过程的结果并且仅包含到子比特流提取过程的输入比特流的子图像的子集时,可能需要在序列参数集(“SPS”)的原始字节序列有效载荷(“RBSP”)中将SPS标志1110的值设置为1。

[0107] 在图11中,SPS句法元素“sps_num_subpics_minus1(sps_num_子图像_减1)”(例如,图11中的句法元素1112)加1指定CLVS中每个图像中的子图像的数量。句法元素1112(“sps_num_subpics_minus1”)的值在从0到 $\text{ceil}(\text{sps_pic_width_max_in_luma_samples} \div \text{CtbSizeY}) \times \text{ceil}(\text{sps_pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ 的包含范围内。当不存在时,句法元素1112(“sps_num_subpics_minus1”)的值被确定为等于0。

[0108] 在图11中,当SPS标志1120(“sps_independent_subpics_flag(sps_独立_子图像_标志)”)等于1时,可以指定CLVS中的所有子图像边界被视为图像边界,并且没有跨越子图像边界的环路滤波。SPS标志1120等于0不会施加此类约束。当不存在时,SPS标志1120的值被确定为等于1。

[0109] 在图11中,SPS句法元素“sps_subpic_ctu_top_left_x[i](sps_子图像_ctu_上_左_x[i])”(例如,图11中的句法元素1122)以CtbSizeY(Ctb大小Y)为单位指定第i个子图像的左上CTU的水平位置。该句法元素1122的长度为 $\text{ceil}(\log_2((\text{sps_pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}))$ 比特。当不存在时,SPS句法元素“sps_subpic_ctu_top_left_x[i]”的值被确定为等于0。

[0110] 类似地,SPS句法元素“sps_subpic_ctu_top_left_y[i](sps_子图像_ctu_上_左_y[i])”(例如,图11中的句法元素1124)以CtbSizeY为单位指定第i个子图像左上CTU的垂直位置。

[0111] 该句法元素1124的长度为 $\text{ceil}(\log_2((\text{sps_pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}))$ 比特。当不存在时,SPS句法元素“sps_subpic_ctu_top_left_y[i]”的值被确定为等于0。

[0112] 在图11中,SPS句法元素“sps_subpic_width_minus1[i](sps_子图像_宽度_减1[i])”(例如,图11中的句法元素1126)加1以CtbSizeY为单位指定第i个子图像的宽度。该句法元素1126的长度为 $\text{ceil}(\log_2((\text{sps_pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}))$ 比特。当不存在时,SPS句法元素“sps_subpic_width_minus1[i]”的值被确定为等于 $((\text{sps_pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) - \text{sps_subpic_ctu_top_left_x}[i] - 1$ 。

[0113] 类似地,SPS句法元素“sps_subpic_height_minus1[i](sps_子图像_高度_减1[i])”(例如,图11中的句法元素1128)加1以CtbSizeY为单位指定第i个子图像的高度。该句法元素1128的长度为 $\text{ceil}(\log_2((\text{sps_pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}))$ 比特。当不存在时,SPS句法元素“sps_subpic_height_minus1[i]”的值被确定为等于 $((\text{sps_pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) \gg \text{CtbLog2SizeY}) - \text{sps_subpic_ctu_top_left_y}[i] - 1$ 。

[0114] 在一些实施例中,为了满足比特流一致性,子图像的形状使得每个子图像在解码时具有其整个左边界和整个上边界,包括图像边界或包括先前解码的子图像的边界。

[0115] 在一些实施例中,对于子图像索引i在从0到句法元素1112的值的包含范围内的每个子图像,为了满足比特流一致性,以下条件为真。首先,(sps_subpic_ctu_top_left_x[i]

$\times \text{CtbSizeY}$) 的值小于 $(\text{sps_pic_width_max_in_luma_samples} - \text{sps_conf_win_right_offset} \times \text{SubWidthC})$ 。第二, $((\text{sps_subpic_ctu_top_left_x}[i] + \text{sps_subpic_width_minus1}[i] + 1) \times \text{CtbSizeY})$ 的值大于 $(\text{sps_conf_win_left_offset} \times \text{SubWidthC})$ 。第三, $(\text{sps_subpic_ctu_top_left_y}[i] \times \text{CtbSizeY})$ 的值小于 $(\text{sps_pic_height_max_in_luma_samples} - \text{sps_conf_win_bottom_offset} \times \text{SubHeightC})$ 。第四, $((\text{sps_subpic_ctu_top_left_y}[i] + \text{sps_subpic_height_minus1}[i] + 1) \times \text{CtbSizeY})$ 的值大于 $(\text{sps_conf_win_top_offset} \times \text{SubHeightC})$ 。

[0116] 在图11中,当SPS标志1130 (“sps_subpic_treated_as_pic_flag[i] (sps_子图像_对待_作为_图像_标志[i])”) 等于1时,可以指定CLVS中每个编码图像的第i个子图像在除环路内滤波操作之外的解码过程中被视为图像。SPS标志1130等于0指定CLVS中每个编码图像的第i个子图像在除环路内滤波操作之外的解码过程中不被视为图像。当不存在时,SPS标志1130的值被确定为等于1。

[0117] 当SPS句法元素“sps_num_subpics_minus1” (例如,图11中的句法元素1112) 的值大于0且SPS标志1130等于1时,对于参考SPS的当前层的每个CLVS,AU的目标集 (“targetAuSet”) 是指从按解码顺序包括CLVS的第一个图像的AU开始,到按解码顺序包括CLVS的最后一个图像(含)的AU的所有访问单元 (“AU”)。在一些实施例中,为了满足比特流一致性,对于目标层集 (“targetLayerSet”),以下条件为真,其中目标层集包括当前层和使用当前层作为参考层的层。首先,对于targetAuSet中的每个AU,targetLayerSet中各层的所有图像具有相同的pps_pic_width_in_luma_samples (pps_图像_宽度_in_亮度_样本) 值和相同的pps_pic_height_in_luma_samples (pps_pic_高度_in_亮度_样本) 值。第二, targetLayerSet中的层所参考的所有SPS具有相同的句法元素1112的值,并且,对于从0到句法元素1112的包含范围内的每个j值,分别具有相同的sps_subpic_ctu_top_left_x[j]、sps_subpic_ctu_top_left_y[j]、sps_subpic_width_minus1[j]、sps_subpic_height_minus1[j]和sps_subpic_treated_as_pic_flag[j]的值。第三,对于targetAuSet中的每个AU,targetLayerSet中各层的所有图像对于从0到句法元素1112的包含范围内的每个j值具有相同的SubpicIdVal[j] (子图像IdVal[j]) 值。

[0118] 在图11中,当等于1时,标志1140 (“sps_loop_filter_across_subpic_enabled_flag[i] (sps_环路_滤波器_跨_子图像_启用_标志[i])”) 指定跨子图像边界的环路滤波操作被启用,并且可以跨CLVS中每个编码图像中的第i个子图像的边界执行。当等于0时,标志1140指定跨子图像边界的环路滤波操作被禁用,并且不跨CLVS中每个编码图像中的第i个子图像的边界执行。当不存在时,将标志1140的值确定为等于0。

[0119] SPS句法元素“sps_subpic_id_len_minus1 (sps_子图像_标识_长度_减1)” (例如,图11中的句法元素1142) 加1指定用于表示SPS句法元素“sps_subpic_id[i] (sps_子图像_id[i])”、PPS句法元素“pps_subpic_id[i] (pps_子图像_id[i])” (当存在时),和句法元素“sh_subpic_id (sh_子图像_id)” (当存在时) 的比特数。在一些实施例中,SPS句法元素“SPS_subpic_id_len_minus1” 的值在从0到15的包含范围内。 $1 \ll (\text{sps_subpic_id_len_minus1} + 1)$ 的值大于或等于sps_num_subpics_minus1+1。

[0120] 在图11中,当等于1时,标志1150 (“sps_subpic_id_mapping_explicitly_signalled_flag (sps_子图像_id_映射_显式_指示_标志)”) 指定子图像ID映射在SPS中或

在由CLVS的编码图像参考的PPS中显式指示。当等于0时,标志1150指定对于CLVS,子图像ID映射没有显式指示。当不存在时,将标志1150的值确定为等于0。

[0121] 在图11中,当等于1时,标志1160(“sps_subpic_id_mapping_present_flag(sps_子图像_id_映射_存在_标志)”)指定当标志1150等于1时,子图像ID映射在SPS中进行指示。当等于0时,标志1160指定当标志1150等于1时,子图像ID映射不在由CLVS的编码图像参考的PPS中进行指示。

[0122] SPS句法元素“sps_subpic_id[i]”(例如,图11中的句法元素1162)指定第i个子图像的子图像ID。在一些实施例中,SPS句法元素“sps_subpic_id[i]”的长度是“SPS_subpic_id_len_minus1”的值加1比特。

[0123] 在一些实施例中,用于砖块和条带分割的砖块映射信息可以在图像参数集(PPS)中进行指示。图12示出了与本公开的一些实施例一致的用于指示砖块映射和砖块映射中的条带的部分PPS句法结构1200的示例性编码句法表。图12中所示的伪代码可以是VVC标准的一部分。

[0124] 在图12中,PPS标志1210(“pps_no_pic_partition_flag(pps_无_图像_分割_标志)”)等于1时,可以指定不将图像分割应用于参考(包括PPS标志1210的)PPS的任何图像,并且当其等于0时,可以指定参考PPS的每个图像可以分割为多于一个砖块或条带。在一些实施例中,比特流一致性要求是,对于编码层视频序列(CLVS)内的编码图像所参考的所有PPS,PPS标志1210的值相同。在一些实施例中,当句法元素1112(“sps_num_subpics_minus1”)的值大于0或当句法元素“pps_mixed_nalu_types_in_pic_flag(pps_混合_nalu_类型_in_图像_标志)”的值等于1时,PPS标志1210的值等于0也是比特流一致性要求。

[0125] 在图12中,标志1220(“pps_subpic_id_mapping_present_flag(pps_子图像_id_映射_存在_标志)”)的值等于1指定子图像ID映射在PPS中进行指示。标志1220等于0指定子图像ID映射不在PPS中进行指示。在一些实施例中,如果SPS标志“sps_subpic_id_mapping_explicitly_signalled_flag”(例如,标志1150)等于0或SPS标志“sps_subpic_id_mapping_present_flag”(例如,标志1160)等于1,则标志1220的值等于0。如果SPS标志1150等于1,并且SPS标志1160等于0,则标志1220的值等于1。

[0126] 在图12中,句法元素“pps_num_subpics_minus1(pps_数量_子图像_减1)”(例如,图12中的句法元素1222)的值等于SPS句法元素“sps_num_subpics_minus1(sps_数量_子图像_减1)”(例如,图11中的句法元素1112)的值。当PPS标志1210等于1时,句法元素1222的值被确定为等于0。

[0127] 在图12中,句法元素“pps_subpic_id_len_minus1(pps_子图像_id_长度_减1)”(例如,图12中的句法元素1224)的值等于SPS句法元素“sps_subpic_id_len_minus1(sps_子图像_id_长度_减1)”(例如,图11中的句法元素1142)的值。

[0128] 在图12中,句法元素“pps_subpic_id[i]”(例如,图12中的句法元素1226)的值指定第i个子图像的子图像ID。在一些实施例中,句法元素1226的长度是pps_subpic_id_len_minus1+1比特。

[0129] 图13示出了根据本公开的一些实施例的用于推导变量SubpicIdVal值的示例性伪代码。如图13所示,对于从0到句法元素1112(“sps_num_subpics_minus1”)的包含范围内的索引i的每个值,可以推导出变量SubpicIdVal的值。在一些实施例中,为了满足比特流一致

性,应用以下两个约束。首先,对于从0到句法元素1112(“sps_num_subpics_minus1”)的包含范围内的索引i和索引j的任意两个不同值,SubpicIdVal[i]不等于SubpicIdVal[j]。其次,对于从0到句法元素1112(“sps_num_subpics_minus1”)的包含范围内的索引i的每个值,当nuh_layer_id(nuh_层_id)等于特定值layerId(层Id)的当前图像的SubpicIdVal[i]的值不等于nuh_layer_id等于layerId的参考图像的SubpicIdVal[i]的值时,当前图像的第i个子图像中的编码条带的参考图像列表(“RPL”)的激活条目不包括该参考图像。

[0130] 在图12中,句法元素“pps_log2_ctu_size_minus5(pps_log2_ctu_大小_减5)”(例如,图12中的句法元素1228)的值加5指定每个CTU的亮度编码树块大小。在一些实施例中,句法元素1228等于在SPS中指示的句法元素“sps_log2_ctu_size_minus5(sps_log2_ctu_大小_减5)”。

[0131] 在图12中,句法元素“pps_num_exp_tile_columns_minus1(pps_数量_显式_砖块_列_减1)”(例如,图12中的句法元素1232)的值加1指定显式提供的砖块列宽度的数量。在一些实施例中,句法元素1232的值在从0到“PicWidthInCtbsY-1”的包含范围内。当PPS标志1210等于1时,句法元素1232的值被确定为0。

[0132] 在图12中,句法元素“pps_num_exp_tile_rows_minus1(pps_num_exp_砖块_行_减1)”(例如,图12中的句法元素1234)的值加1指定显式提供的砖块行高度的数量。在一些实施例中,句法元素1234的值在从0到“PicHeightInCtbsY-1”的包含范围内。当PPS标志1210等于1时,句法元素1234的值被确定为0。

[0133] 在图12中,句法元素“pps_tile_column_width_minus1[i](pps_砖块_列_宽度_减1[i])”(例如,图12中的句法元素1236)的值加1指定了以CTB为单位的第i个砖块列的宽度,其中索引i在从0到句法元素1232的包含范围内。在一些实施例中,句法元素“pps_tile_column_width_minus1[num_exp_tile_columns_minus1]”可用于推导索引大于本文规定的句法元素1232的砖块列的宽度。句法元素1236的值在从0到“PicWidthInCtbsY-1”的包含范围内。当在PPS中不存在时,句法元素“pps_tile_column_width_minus1[0](pps_砖块_列_宽度_减1[0])”的值被确定为等于“PicWidthInCtbsY-1”。

[0134] 在图12中,句法元素“pps_tile_row_height_minus1[i](pps_砖块_行_高度_减1[i])”(例如,图12中的句法元素1238)的值加1指定了以CTB为单位的第i个砖块行的高度,其中索引i在从0到句法元素1234的包含范围内。在一些实施例中,句法元素“pps_tile_row_height_minus1[pps_num_exp_tile_rows_minus1]”用于推导索引大于本文规定的句法元素1234的砖块行的高度。1238的值在从0到“PicHeightInCtbsY-1”的包含范围内。当在PPS中不存在时,句法元素“pps_tile_row_height_minus1[0]”的值被确定为等于“PicHeightInCtbsY-1”。

[0135] 在图12中,当等于1时,标志1230(“pps_loop_filter_across_tiles_enabled_flag(pps_环路_滤波器_跨_砖块_启用_标志)”)指定跨砖块边界的环路内滤波操作被启用,并且可以跨参考PPS的图像中的砖块边界执行。当等于0时,标志1230指定跨砖块边界的环路内滤波操作被禁用,并且不跨参考PPS的图像中的砖块边界执行。环路内滤波操作包括去块滤波器、采样自适应偏移滤波器和自适应环路滤波器操作。当不存在时,将标志1230的值被确定为等于1。

[0136] 在图12中,当等于0时,标志1240(“pps_rect_slice_flag(pps_矩形_条带_标

志)”)指定每个条带内的砖块是按照光栅扫描顺序,并且条带信息在PPS中不进行指示。当等于1时,标志1240指定每个条带内的砖块覆盖图像的矩形区域,并且条带信息在PPS中进行指示。当在PPS中不存在时,标志1240可以被确定为等于1。在一些实施例中,当SPS标志1110(“sps_subpic_info_present_flag”)等于1或PPS标志“pps_mixed_nalu_types_in_pic_flag”等于1时,标志1240的值等于1。

[0137] 在图12中,当等于0时,标志1250(“pps_single_slice_per_subpic_flag(pps_单个_条带_每个_子图像_标志)”)指定每个子图像可以包括一个或多个矩形条带。当等于1时,标志1250指定每个子图像包括一个矩形条带。当不存在时,标志1250被确定为等于1。

[0138] 在图12中,句法元素“pps_num_slices_in_pic_minus1(pps_num_条带_in_图像_减1)”(例如,图12中的句法元素1252)加1指定参考PPS的每个图像中的矩形条带的数量。在一些实施例中,句法元素1252的值(“pps_num_slices_in_pic_minus1”)在从0到句法元素“MaxSlicesPerPicture(最大条带每幅图像)”的值减1的包含范围内。句法元素“MaxSlicesPerPicture”指示图像中允许的最大条带数量的级别限制。当PPS标志1210等于1时,句法元素1252的值(“pps_num_slices_in_pic_minus1”)被确定为等于0。即,没有将图像分割应用于参考PPS的任何图像,因此该图像包括单个条带。

[0139] 当PPS标志1250等于1时,句法元素1252(“pps_num_slices_in_pic_minus1”)的值被确定为等于SPS中句法元素1112(“sps_num_subpics_minus1”)的值。即,每个子图像包括单个矩形条带,因此每个图像中矩形条带的数量等于每个图像中子图像的数量。

[0140] 在图12中,当等于0时,标志1260(“pps_tile_idx_delta_present_flag(pps_砖块_索引_delta_存在_标志)”)指定在PPS中不存在pps_tile_idx_delta_val[i](pps_砖块_索引_delta_val[i])句法元素,并且以光栅扫描顺序将参考PPS的图像分割为矩形条带行和矩形条带列。当等于1时,标志1260指定pps_tile_idx_delta_val[i]句法元素可以存在于PPS中,并且参考PPS的图像中的矩形条带按照pps_tile_idx_delta_val[i]句法元素的值所指示的顺序(以索引i的值增加的方式)指定。当不存在时,标志1260的值被确定为等于0。

[0141] 在图12中,句法元素“pps_slice_width_in_tiles_minus1[i](pps_条带_宽度_in_砖块_减1[i])”(例如,图12中的句法元素1262)加1指定以砖块列为单位的第i个矩形条带的宽度。在一些实施例中,句法元素1262的值(“pps_slice_width_in_tiles_minus1[i]”)在从0到“NumTileColumns-1”的包含范围内。当在PPS中不存在时,句法元素1262的值(“pps_slice_width_in_tiles_minus1[i]”)被确定为等于0。

[0142] 在图12中,句法元素“pps_slice_height_in_tiles_minus1[i](pps_条带_高度_in_砖块_减1[i])”(例如,图12中的句法元素1264)加1指定以砖块行为单位的第i个矩形条带的高度。在一些实施例中,句法元素1264(“pps_slice_height_in_tiles_minus1[i]”)的值在从0到“NumTileRows-1”的包含范围内。当PPS中不存在句法元素1264(“pps_slice_height_in_tiles_minus1[i]”)时,如果“SliceTopLeftTileIdx[i]/NumTileColumns(条带上左砖块索引[i]/数量砖块列)”的值等于“NumTileRows-1”的值,句法元素1264的值被确定为等于0;否则,句法元素1264的值被确定为等于“pps_slice_height_in_tiles_minus1[i-1](pps_条带_高度_in_砖块_减1[i-1])”的值。

[0143] 在图12中,句法元素“pps_num_exp_slices_in_tile[i](pps_num_exp_条带_in_

砖块[i]” (例如,图12中的句法元素1266) 的值指定为当前砖块 (例如,包含第i个条带的砖块) 中的条带显式提供的条带高度的数量。在一些实施例中,句法元素1266 (“pps_num_exp_slices_in_tile[i]”) 的值在从0到“RowHeight[SliceTopLeftTileIdx[i]/NumTileColumns]-1”的包含范围内。当不存在时,句法元素1266的值被确定为等于0。当句法元素1266等于0时,变量“NumSlicesInTile[i]”的值被推导为等于1,指示包含第i个条带的砖块未被拆分为多个条带。当句法元素1266大于0时,包含第i个条带的砖块可以被拆分成多个条带。

[0144] 在图12中,句法元素“pps_exp_slice_height_in_ctus_minus1[i][j] (pps_显式_条带_高度_in_ctus_减1[i][j])” (例如,图12中的句法元素1268) 的值加1指定了以CTU行为单位的包含第i个条带的砖块中第j个矩形条带的高度。当句法元素1266 (“pps_num_exp_slices_in_tile[i]”) 大于0时 (即,该砖块可以分割为多个砖块),索引j在从0到“pps_num_exp_slices_in_tile[i]-1”的值的包含范围内。

[0145] 句法元素“pps_exp_slice_height_in_ctus_minus1[i][pps_num_exp_slices_in_tile[i]”的值也用于推导包含第i个条带的砖块中索引大于如本文规定的pps_num_exp_slices_in_tile[i]-1的矩形条带的高度。句法元素1268 (“pps_exp_slice_height_in_ctus_minus1[i][j] (pps_显式_条带_高度_in_ctus_减1[i][j])”) 的值应在从0到RowHeight[SliceTopLeftTileIdx[i]/NumTileColumns]-1的包含范围内。即,砖块中任何矩形条带的高度的最大可能值应为当前砖块的高度。

[0146] 在图12中,句法元素“pps_tile_idx_delta_val[i]” (例如,图12中的句法元素1272) 指定包含第(i+1)个矩形条带中第一个CTU的砖块的砖块索引与包含第i个矩形条带中第一个CTU的砖块的砖块索引之差。句法元素1272 (“pps_tile_idx_delta_val[i]”) 的值应在:“-NumTilesInPic+1”至“NumTilesInPic-1”的包含范围内。当不存在时,句法元素1272的值被确定为等于0。当存在时,句法元素1272的值不等于0。

[0147] 在一些实施例中,当标志1240等于1时,比特流一致性要求是,对于属于同一图像和不同子图像的具有图像级别条带索引idxA和idxB的任何两个条带,当SubpicIdxForSlice[idxA]小于SubpicIdxForSlice[idxB]时,idxA的值应小于idxB。

[0148] 在图12中,当等于1时,标志1270 (“pps_loop_filter_across_slices_enabled_flag (pps_环路_滤波器_跨_条带_启用_标志)”) 指定跨条带边界的环路内滤波操作被启用,并且可以跨参考PPS的图像中条带边界被执行。当等于0时,标志1270指定跨条带边界的环路内滤波操作被禁用,并且不跨参考PPS的图像中的条带边界被执行。环路内滤波操作包括去块滤波器、采样自适应偏移滤波器和自适应环路滤波器操作。当不存在时,将标志1270的值被确定为等于0。

[0149] 指定砖块列的数量的变量NumTileColumns,以及指定以CTB为单位的第i个砖块列的宽度的列表colWidth[i] (列宽度[i]) 可以被推导,i的范围从0到NumTileColumns-1 (含)。图14示出了根据本公开的一些实施例的包括变量NumTileColumns的推导的示例性伪代码。

[0150] 指定砖块行的数量的变量NumTileRows,以及指定以CTB为单位的第j个砖块行的高度的列表RowHeight[j] (行高度[j]) 可以被推导,j的范围从0到NumTileRows-1 (含)。图15示出了根据本公开的一些实施例的包括变量NumTileRows的推导的示例性伪代码。变量

NumTilesInPic设置为等于NumTileColumns \times NumTileRows。

[0151] 以CTB为单位指定第*i*个砖块列边界的位置的列表tileColBd[*i*] (砖块列边界[*i*]) 可以被推导,*i*的范围从0到NumTileColumns (含)。图16示出了根据本公开的一些实施例的包括变量tileColBd[*i*]的推导的示例性伪代码。如图16所示,阵列tileColBd[]的大小比实际的砖块列的数量大一。

[0152] 以CTB为单位指定第*j*个砖块行边界的位置的列表tileRowBd[*j*] (砖块行边界[*j*]) 可以被推导,*j*的范围从0到NumTileRows (含)。图17示出了根据本公开的一些实施例的包括变量tileRowBd[*j*]的推导的示例性伪代码。如图17所示,阵列tileRowBd[]的大小比实际的砖块行的数量大一。

[0153] 分别指定从水平CTB地址到以CTB为单位的左侧砖块列边界和砖块列索引的转换的列表CtbToTileColBd[ctbAddrX]和ctbToTileColIdx[ctbAddrX]可以分别被推导, ctbAddrX的范围从0到PicWidthInCtbsY (含)。图18示出了根据本公开的一些实施例的包括变量CtbToTileColBd和ctbToTileColIdx的推导的示例性伪代码。如图18所示,阵列CtbToTileColBd[]和ctbToTileColIdx[]的大小比CTB中的实际图像宽度大一。

[0154] 分别指定从垂直CTB地址到以CTB为单位的顶部砖块列边界和到砖块行索引的转换的列表CtbToTileRowBd[ctbAddrY]和ctbToTileRowIdx[ctbAddrY]可以分别被推导, ctbAddrY的范围为从0到PicHeightInCtbsY (含)。图19示出了根据本公开的一些实施例的包括变量CtbToTileRowBd和ctbToTileRowIdx的推导的示例性伪代码。如图19所示,阵列CtbToTileRowBd[]和ctbToTileRowIdx[]的大小比CTB中的实际图像高度大一。

[0155] 分别指定第*i*个子图像的宽度和高度的列表SubpicWidthInTiles[*i*] (子图像宽度In砖块[*i*]) 和SubpicHeightInTiles[*i*] (子图像高度In砖块[*i*]) 可以分别被推导,*i*的范围为从0到句法元素1112 (“sps_num_subpics_minus1”) (含), 并且, 指定第*i*个子图像的高度是否小于一个砖块行的列表subpicHeightLessThanOneTileFlag[*i*] (子图像高度小于一个砖块标志[*i*]) 可以被推导,*i*的范围为从0到句法元素1112 (“sps_num_subpics_minus1”) (含)。图20示出了根据本公开的一些实施例的包括变量SubpicWidthInTitles和SubpicHeightInTiles的推导的示例性伪代码。如图20所示, 当一个砖块被分割为多个矩形条带并且第*i*个子图像中仅包括该砖块的矩形条带的子集时, 该砖块在SubpicHeightInTiles[*i*]的值中被计为一个砖块。

[0156] 当标志1240 (“pps_rect_slice_flag”) 等于1时, 以下可以被推导: 指定第*i*个条带中CTU的数量的列表NumCtusInSlice[*i*] (数量CtusIn条带[*i*]), *i*的范围为从0到句法元素1252 (“pps_num_slices_in_pic_minus1”) (含), 指定包含条带中第一个CTU的砖块的砖块索引的列表SliceTopLeftTileIdx[*i*] (条带上左砖块索引[*i*]), *i*的范围为从0到句法元素1252 (“pps_num_slices_in_pic_minus1”) (含), 以及指定第*i*个条带内第*j*个CTB的图像光栅扫描地址的矩阵CtbAddrInSlice[*i*][*j*] (Ctb地址In条带[*i*][*j*]), *i*的范围为从0到句法元素1252 (“pps_num_slices_in_pic_minus1”) (含) 且*j*的范围从0到NumCtusInSlice[*i*]-1 (含), 以及指定包含第*i*个条带的砖块中条带的数量的变量NumSlicesInTile[*i*]。图21和图22示出了根据本公开的一些实施例的包括变量NumCtusInSlice、SliceTopLeftTileIdx、CtbAddrInSlice和NumSlicesInTile的推导的示例性伪代码。

[0157] 为了满足比特流一致性, 对于*i*的范围为从0到句法元素1252 (“pps_num_slices_

in_pic_minus1”) (含), 列表NumCtusInSlice[i]的值都大于0。此外, 为了满足比特流一致性, 对于i的范围为从0到句法元素1252 (“pps_num_slices_in_pic_minus1”) (含), j的范围从0到NumCtusInSlice[i]-1 (含), 矩阵CtbAddrInSlice[i][j]包括0到PicSizeInCtbsY-1 (含) 范围内的所有CTB地址, 一次且仅一次。

[0158] 分别指定第i个子图像中条带的数量、具有图像级别条带索引j的条带的子图像级别条带索引、以及具有图像级别条带索引j的条带的子图像索引的列表NumSlicesInSubpic[i] (数量条带In子图像[i])、SubpicLevelSliceIdx[j] (子图像级别条带索引[j]), 以及SubpicIdxForSlice[j] (子图像索引For条带[j]) 可以被推导。图23示出了根据本公开的一些实施例的包括变量NumSlicesInSubpic、SubpicLevelSliceIdx和SubpicIdxForSlice的推导的示例性伪代码。

[0159] 本公开的实施例提供在SPS中指示标志, 以指示当前CLVS中的图像是否被分割为多个砖块或条带, 并且在PPS中用与在SPS中指示的标志相同的值来指示标志。图24示出了与本公开的一些实施例一致的、在SPS中指示的部分SPS句法结构2400的示例性编码句法表, 以粗体强调。基于图11的句法结构1100修改句法结构2400。如图24所示, 以突出显示方式显示来自图11所示的先前句法的更改。

[0160] 与图11中所示的SPS句法相比, 在一些实施例中, 在图24所示的SPS句法中, SPS标志2410 (“sps_no_pic_partition_flag (sps_no_pic_分割_标志)”) 在SPS标志1110之前进行指示。等于1的SPS标志2410 (“sps_no_pic_partition_flag”) 指定不将图片分割应用于参考该SPS的每个图像。等于0的SPS标志2410指定参考该SPS的每个图像可以分割为多个砖块或条带。

[0161] 在一些实施例中, 可以引入与图24中所示句法相关联的一个或多个附加约束。例如, 附加约束可以是, 在PPS中指示的标志1210 (“pps_no_pic_partition_flag”) 的值等于相关联SPS中SPS标志2410 (“sps_no_pic_partition_flag”) 的值。

[0162] 此外, 可以添加另一个约束, 以确保当SPS标志2410的值等于1时, 句法元素1112 (“sps_num_subpics_minus1”) 的值为0。因为子图像是一组条带, 当没有条带的分割 (例如, SPS句法元素标志为1) 时, 就不会有子图像的分割 (例如, SPS句法元素1112为0)。

[0163] 图25示出了与本公开的一些实施例一致的、在SPS中指示的部分SPS句法结构2500的示例性编码句法表, 以粗体强调。句法结构2500也基于图11的句法结构1100进行了修改。如图25所示, 以突出显示的方式显示了来自图11所示的先前句法的更改。

[0164] 如上所述, 在一些实施例中, 当SPS标志2410 (“sps_no_pic_partition_flag”) 等于1时, 子图像的数量只能为1。因此, SPS标志1110 (“sps_subpic_info_present_flag”) 或SPS句法元素1112 (“sps_num_subpics_minus1”) 可以基于SPS标志2410 (“sps_no_pic_partition_flag”) 的值有条件地进行指示。例如, 如图25所示, 当SPS标志2410 (“sps_no_pic_partition_flag”) 等于0时, 指示SPS标志1110 (“sps_subpic_info_present_flag”), 当SPS标志2410 (“sps_no_pic_partition_flag”) 的值等于1时, SPS标志1110 (“sps_subpic_info_present_flag”) 的指示可以被跳过。当跳过SPS标志1110的指示时, SPS标志1110 (“sps_subpic_info_present_flag”) 的值可以被确定为0。换言之, 当SPS不允许对图像进行分割时, 子图像信息不存在。

[0165] 在图25的句法中, 等于1的SPS标志2410 (“sps_no_pic_partition_flag”) 指定对

参考SPS的每个图像都不应用图像分割。等于0的SPS标志2410指定参考SPS的每个图像可以被分割为多个砖块或条带。在图25的句法中,等于1的SPS标志1110(“sps_subpic_info_present_flag”)指定对于CLVS,子图像信息存在,并且CLVS的每个图像中可能存在一个或多个子图像。等于0的SPS标志1110(“sps_subpic_info_present_flag”)指定对于CLVS,子图像信息不存在,并且CLVS的每个图像中只有一个子图像。当不存在时,SPS标志1110(“sps_subpic_info_present_flag”)的值可以被确定为等于0。

[0166] 图26示出了与本公开的一些实施例一致的、在SPS中指示的部分SPS句法结构2600的另一示例性编码句法表,以粗体强调。句法结构2600也是基于图11的句法结构1100进行了修改。如图26所示,以在突出显示的方式显示了来自图11所示的先前句法的更改。

[0167] 如图26所示,在一些实施例中,当标志1110等于1且SPS标志2410等于0时,指示SPS标志1112(“sps_num_subpics_minus1”)。或者说,当SPS标志2410(“sps_no_pic_partitoin_flag”)等于1时,SPS标志1112(“sps_num_subpics_minus1”)的指示被跳过。如上所讨论的,当SPS标志2410的值为1时,SPS标志1112(“sps_num_subpics_minus1”)的值被确定为0,指示。换言之,当SPS不允许对图像进行分割时,子图像的数量只能是1。

[0168] 在图26的句法中,等于1的SPS标志2410(“sps_no_pic_partitoin_flag”)指定不对参考SPS的每个图像应用图像分割。SPS标志2410等于0指定参考SPS的每个图像可以分割为多个砖块或条带。在图26的句法中,SPS标志1112(“sps_num_subpics_minus1”)加1指定CLVS中每个图像中的子图像的数量。SPS标志1112(“sps_num_subpics_minus1”)的值在从0到Ceil(sps_pic_width_max_in_luma_samples÷CtbSizeY)×Ceil(sps_pic_height_max_in_luma_sample s÷CtbSizeY)-1(含)的范围内。当不存在时,SPS标志1112(“sps_num_subpics_minus1”)的值可以被确定为等于0。

[0169] 在一些实施例中,PPS标志1210(“pps_no_pic_partition_flag”)的值对于CLVS内编码图像所参考的所有PPS可以相同的约束可以被移除。换言之,对于同一CLVS内的图像,不同的PPS可以独立地指示具有不同值的PPS标志1210(“pps_no_pic_partition_flag”)。结果,在一个CLVS内,允许PPS标志1210(“pps_no_pic_partition_flag”)等于1的一些图像不被分割为多个条带或砖块,而PPS标志1210(“pps_no_pic_partition_flag”)等于0的其他一些图像被分割为多个条带或砖块。因此,PPS标志1210(“pps_no_pic_partition_flag”)的值对于CLVS内编码图像所参考的所有PPS应相同的约束可以被移除。

[0170] 图27A-图27C分别示出了与本公开的一些实施例一致的示例性视频编码或解码方法2700A、2700B和2700C的流程图。通过应用视频编码或解码方法2700A、2700B和2700C中的任何一种,可以基于其他SPS句法元素或标志的值有条件地指示一个或多个SPS句法元素或标志,从而减少输出比特数并实现更高的编码性能。当SPS句法元素或标志的值未在SPS中指示时,可以相应地被确定或分配。在一些实施例中,方法2700A-2700C可以由编码器(例如,执行图2A中的过程200A或图2B中的过程200B的编码器)或解码器(例如,执行图3A中的解码过程300A或图3B中的解码过程300B的解码器)执行,以生成或解码图5中所示的比特流500。例如,编码器或解码器可以实现为用于编码或转码视频序列(例如,图2A或图2B中的视频序列202)以生成视频序列的比特流(例如,图2A或图2B中的视频比特流228),或解码比特流(例如,图3A或图3B中的视频比特流228)以重构比特流的视频流(例如,图3A或图3B中的视频流304)的装置(例如,图4中的装置400)的一个或多个软件或硬件组件。例如,一个或多

个处理器(例如,图4中的处理器402)可以执行方法2700A-2700C。

[0171] 图24中所示的句法结构2400可以应用于方法2700A中。在步骤2710,装置被配置为在比特流的序列参数集 (SPS) 中编码或解码SPS标志2410 (“sps_no_pic_partition_flag”)。SPS标志2410指示参考SPS的编码层视频序列 (CLVS) 中的一个或多个图像是否被允许分割为砖块或条带。

[0172] 在步骤2720,装置被配置为在SPS中编码或解码SPS标志1110 (“sps_subpic_info_present_flag”)。SPS标志1110指示对于参考SPS的CLVS,子图像信息是否存在。

[0173] 在步骤2730,装置被配置为确定SPS标志1110是否等于1。响应于SPS标志1110为零(步骤2730-否),装置绕过步骤2740并继续编码或解码过程,而不指示SPS句法元素1112 (“sps_num_subpics_minus1”)。

[0174] 响应于SPS标志1110等于1(步骤2730-是),装置执行步骤2740,并被配置为在SPS中编码或解码SPS句法元素1112 (“sps_num_subpics_minus1”)。SPS句法元素1112 (“sps_num_subpics_minus1”)是一个序列参数,与参考SPS的CLVS中每个图像中的子图像的数量相关。特别是,SPS句法元素1112加1指定CLVS中每个图像中的子图像的数量。应当理解,虽然该示例流程图将SPS标志1110表示为等于1导致步骤2740被执行,应当理解,0和1的指示是一种设计选择,并且可以反转SPS标志1110和其他句法元素的结果(例如,SPS标志1110等于0导致执行步骤2740)。

[0175] 在步骤2750,装置被配置为在SPS之后的图像参数集 (PPS) 中编码或解码PPS标志1210 (“pps_no_pic_partition_flag”),PPS标志1210等于SPS标志1110,用于指示参考PPS的图像是否被允许分割。

[0176] 图25中所示的句法结构2500可以应用于方法2700B中。与方法2700A相比,在方法2700B中,在步骤2170之后,装置执行步骤2715,并被配置为确定SPS标志2410是否等于零(指示与SPS相关联的图像被允许分割)。响应于SPS标志2410为一(步骤2715-否),指示与SPS相关联的图像未分割,该装置绕过步骤2720并跳过编码或解码SPS标志1110。在一些实施例中,当跳过SPS标志1110的指示时,SPS标志1110的值可以被确定为0。

[0177] 图26中所示的句法结构2600可以应用于方法2700C中。与方法2700A和2700B相比,在方法2700C中,响应于SPS标志1110等于1(步骤2730-是),装置执行步骤2715,并配置为确定SPS标志2410是否等于零。响应于SPS标志2410为一(步骤2715-否),指示与SPS相关联的图像未分割,该装置绕过步骤2740并跳过编码或解码与参考SPS的CLVS中每个图像中的子图像的数量相关联的SPS句法元素1112。在一些实施例中,当SPS句法元素1112的指示被跳过时,SPS句法元素1112的值可以被确定为0。

[0178] 通过方法2700B或2700C,SPS标志1110 (“sps_subpic_info_present_flag”)或SPS句法元素1112 (“sps_num_subpics_minus1”)可以基于SPS标志2410 (“sps_no_pic_partition_flag”)的值有条件地进行指示。

[0179] 本公开的实施例提供了一种更新的方法来指示标志1240 (“pps_rect_slice_flag”)。图28示出了根据本公开的一些实施例的包括指示矩形条带模式或光栅扫描条带模式的更新的标志的部分PPS句法结构2800的示例性的修改后的编码句法表。如图28中所示,以突出显示方式显示了来自图12所示的先前句法的更改。

[0180] 在VVC(例如VVC草稿9)中,标志1240 (“pps_rect_slice_flag”)是指示使用矩形条

带模式还是光栅扫描条带模式的标志。如图12所示,当砖块的数量大于1时,标志1240 (“pps_rect_slice_flag”)可以被指示。当图像仅包含一个砖块时,仅包含一个砖块的光栅扫描条带也可以通过矩形条带模式指示。因此,当砖块的数量等于1时,不需要使用光栅扫描条带模式或对标志1240 (“pps_rect_slice_flag”)进行指示。因此,标志1240 (“pps_rect_slice_flag”)的指示被跳过,且标志1240的值被确定为1,指示应用矩形条带模式。

[0181] 在某些情况下,当砖块的数量小于预定值时,标志1240的指示是不必要的和冗余的。当图像中的砖块的数量小于4时,图像中最多有一个砖块列或一个砖块行,因此光栅扫描条带也可以由矩形条带表示。因此,在一些实施例中,只要砖块的数量小于4,光栅扫描条带的条带分割由矩形条带模式表示。或者,当砖块的数量为2或3时,分割可以通过矩形条带模式表示。通过采用这种修改,可以提高视频流的编码和解码过程的一致性和效率。

[0182] 因此,如图28所示,当砖块的数量小于4时,标志1240 (“pps_rect_slice_flag”)的指示可以被跳过,且标志1240的值被确定为1。当砖块的数量大于3时,标志1240 (“pps_rect_slice_flag”)被指示。在图28所示的句法中,等于0的标志1240 (“pps_rect_slice_flag”)指定光栅扫描条带模式用于参考PPS的每个图像(即,每个条带内的砖块按光栅扫描顺序),并且条带信息在PPS中不进行指示。等于1的标志1240 (“pps_rect_slice_flag”)指定矩形条带模式用于参考PPS的每个图像(即,每个条带内的砖块覆盖图像的矩形区域),条带信息在PPS中进行指示。当不存在时,标志1240 (“pps_rect_slice_flag”)被确定为等于1。当SPS标志1110 (“sps_subpic_info_present_flag”)等于1或PPS标志“pps_mixed_nalu_types_in_pic_flag”等于1时,标志1240 (“pps_rect_slice_flag”)的值等于1。

[0183] 图29示出了与本公开的一些实施例一致的示例性视频编码或解码方法2900的流程图。与方法2700A-2700C类似,方法2900可以由编码器或解码器执行,其可以被实现为用于编码或转码视频序列或解码比特流以重构视频流的装置(例如,图4中的装置400)的一个或多个软件或硬件组件。例如,一个或多个处理器(例如,图4中的处理器402)可以执行方法2900。

[0184] 图28所示的句法结构2800可以应用于方法2900中。在步骤2910,装置被配置为在SPS之后的PPS中编码或解码用于指示参考PPS的图像是否被分割的PPS标志1210 (“pps_no_pic_partition_flag”)。如上所述,在一些实施例中,对于同一CLVS内的图像,不同的PPS可以独立地指示具有不同值的PPS标志1210。例如,在步骤2910,该装置可以在第一PPS中编码或解码指示参考第一PPS的第一图像是否被分割为砖块或条带的第一标志,并在第二PPS中编码或解码指示参考第二PPS的第二图像是否被分割的第二标志,其中第一标志和第二标志具有不同的值。例如,具有第一值的第一标志可以指示CLVS的第一图像未分割,而具有不同于第一值的第二值的第二标志指示第二图像被允许分割。也就是说,CLVS的第二图像可能是分割的,但第二图像也可能是未分割的。

[0185] 在步骤2920,该装置被配置为确定图像的砖块的数量(例如,变量NumTilesInPic)是否大于阈值(例如,3)。在一些实施例中,阈值是大于1的整数。

[0186] 响应于图像的砖块的数量大于阈值(步骤2920-是),该装置执行步骤2930来编码或解码指示条带模式应用于图像的标志1240 (“pps_rect_slice_flag”)。例如,当光栅扫描条带模式应用于参考PPS的图像时,用第一值(例如,0)编码或解码标志1240 (“pps_rect_slice_flag”),并且当矩形条带模式应用于参考PPS的图像时,用与不同于第一值的第二值

(例如,1) 编码或解码标志1240 (“pps_rect_slice_flag”)。

[0187] 响应于图像的砖块的数量不大于阈值(步骤2920-否),跳过步骤2930。在一些实施例中,当标志1240 (“pps_rect_slice_flag”)的指示被跳过时,标志1240的值被确定为第二值(例如,1),指示应用了矩形条带模式。

[0188] 在步骤2940,装置被配置为根据标志1240的值确定是否应用矩形条带模式。当矩形条带模式应用于参考PPS的图像时(步骤2940-是),该装置执行步骤2950以编码或解码PPS中的条带信息。当光栅扫描条带模式应用于参考PPS的图像时(步骤2940-否),跳过步骤2950。

[0189] 通过方法2900,约束被移除,使得与同一CLVS内的图像相关联的不同PPS可以指示具有不同的值的PPS标志1210。此外,标志1240的指示可以被简化,并根据砖块的数量有条件地执行。此外,还可以根据用于分割的条带模式在PPS中有条件地指示条带信息。

[0190] 在VVC(例如VVC草案9)中,根据矩形条带模式的定义,在VVC中支持矩形条带的两种情况。在第一种情况下,矩形条带包括数个完整的砖块,这些砖块共同构成图像的矩形区域。在第二种情况下,矩形条带包括一个砖块的数个连续完整CTU行,这些行共同构成图像的矩形区域。但是,当pps_slice_flag等于1时,pps_slice_flag的语义仅指定第一种情况。因此,VVC中之前的语义可能不准确。

[0191] 本公开的实施例提供了标志1240 (“pps_rect_slice_flag”)的更新语义。在一些实施例中,矩形条带包含一个或多个完整砖块的情况(例如,图9中第一个条带包括完整砖块910和920)和一个砖块包含一个或多个条带的情况(例如,图9中砖块930包括与第二个条带和第三个条带相对应的部分932和934)均被指定在等于1的标志1240 (“pps_rect_slice_flag”)的语义中。或者说,标志1240的语义被修改以反映等于1的标志1240 (“pps_rect_slice_flag”)指定每个条带内的砖块覆盖图像的矩形区域,“或砖块内的每个条带覆盖图像的矩形区域”,并且条带信息在PPS中进行指示。

[0192] 或者,标志1240的语义也可以被修改,以反映等于1的标志1240 (“pps_rect_slice_flag”)指定每个条带内的砖块覆盖图像的矩形区域,“或砖块内的每个条带覆盖砖块的一个或多个连续完整CTU行”,并且条带信息在PPS中进行指示。

[0193] 在一些其他实施例中,标志1240 (“pps_rect_slice_flag”)的语义可以直接指光栅扫描条带模式和矩形条带模式。例如,还可以修改标志1240的语义,以反映等于0的标志1240 (“pps_rect_slice_flag”)指定“光栅扫描条带模式用于参考PPS的每个图像”,并且条带信息不在PPS中进行指示,并且等于1的标志1240 (“pps_rect_slice_flag”)指定“矩形条带模式用于参考PPS的每个图像”,并且条带信息在PPS中进行指示。

[0194] 在VVC(例如,VVC草稿9)中,标志1250 (“pps_single_slice_per_subpic_flag”)指定每个子图像是否包括一个或多个矩形条带。因此,当应用矩形条带模式时,标志1250是相关的。当标志1250不存在时,标志1250的值可以被确定为等于1。然而,如图12所示,在两种不同的场景下,标志1250 (“pps_single_slice_per_subpic_flag”)的指示可能被跳过。在第一种场景中,PPS标志1210 (“pps_no_pic_partition_flag”)等于1,与条带分割相关的标志和句法元素被跳过。在第二种场景中,标志1240 (“pps_rect_slice_flag”)等于0,指示应用了光栅扫描条带模式,而标志1250 (“pps_single_slice_per_subpic_flag”)不相关。因此,在第二种场景下,确定标志1250的值 (“pps_single_slice_per_subpic_flag”)是不合

理的。

[0195] 因此,在一些实施例中,当不存在时,标志1250的值在应用矩形条带模式时确定,而在应用估价者扫描条带模式时不确定。因此,标志1250当不存在于PPS中时,可以在矩形条带模式下被正确地确定,并且在无关的估价者扫描条带模式下不会被确定。因此,可以提高编码性能和一致性。例如,可以修改标志1250的语义,以反映等于1的标志1250(“pps_single_slice_per_subpic_flag”)指定每个子图像包含且仅包含一个矩形条带。等于0的标志1250(“pps_single_slice_per_subpic_flag”)指定每个子图像可以包含一个或多个矩形条带。当PPS标志1210(“pps_no_pic_partition_flag”)等于1时,标志1250的值被确定为等于1。

[0196] 在VVC(例如,VVC草稿9)中,当句法元素1262(“pps_slice_width_in_tiles_minus1[i]”)和句法元素1264(“pps_slice_height_in_tiles_minus1[i]”)不存在时,确定句法元素1262、1264等于0。然而,在某些语义中,索引i的范围没有清楚地指定。此外,在一些实施例中,句法元素1262和1264指示第i个条带的宽度和高度,因此索引i的范围是从0到句法元素1252(例如,条带的数量减1)。然而,对于等于句法元素1252(“pps_num_slices_in_pic_minus1”)的索引i,句法元素1262(“pps_slice_width_in_tiles_minus1[i]”)和句法元素1264(“pps_slice_height_in_tiles_minus1[i]”)的值不被确定为等于0。因此,希望清楚地指定其句法元素1262和1264被确定为等于0的索引i的范围。

[0197] 本公开的实施例提供了句法元素1262和1264的更新语义。在一些实施例中,可以修改句法元素1262和1264的语义,以反映句法元素1262加1指定以砖块列为单位的第i个矩形条带的宽度,并且句法元素1262的值应在从0到NumTileColumns-1的包含范围内。当对于在从0到句法元素1252减1的范围内的索引i不存在时,句法元素1262的值被确定为等于0。

[0198] 类似地,当句法元素1266等于0时,句法元素1264加1指定以砖块行为单位的第i个矩形条带的高度。句法元素1264的值应在从0到NumTileRows-1的包含范围内。当对于在从0到句法元素1252减1的范围内的索引i不存在时,当SliceTopLeftTileIdx[i]/NumTileColumns等于NumTileRows-1时,句法元素1264的值可以被确定为等于0。否则,索引i的句法元素1264的值被确定为等于索引(i-1)的句法元素1264(“pps_slice_height_in_tiles_minus1[i-1]”)的值。

[0199] 如图12所示,在一些实施例中,句法元素1266(“pps_num_exp_slices_in_tile[i]”)指定了为包含第i个条带的砖块中的条带显式提供的条带高度的数量。如果句法元素1266(“pps_num_exp_slices_in_tile[i]”)等于0,则包含第i个条带的砖块不会拆分为多个条带。如果句法元素1266(“pps_num_exp_slices_in_tile[i]”)不等于0(例如,句法元素1266(“pps_num_exp_slices_in_tile[i]”)大于0),则包含第i个条带的砖块可能或可能不拆分为多个条带。此外,当句法元素1266(“pps_num_exp_slices_in_tile[i]”)大于0时,句法元素1268(“pps_exp_slice_height_in_ctus_minus1[i][j]”)加1指定以CTU行为单位的包含第i个条带的砖块中第j个矩形条带的高度,索引j在从0到句法元素1266(“pps_num_exp_slices_in_tile[i]”)减1的包含范围内。句法元素1268的值(“pps_exp_slice_height_in_ctus_minus1[i][j]”)在从0到RowHeight[SliceTopLeftTileIdx[i]/NumTileColumns]-1的包含范围内。

[0200] 在砖块仅包括一个条带的场景中,可以应用两种不同的指示方法。在第一种方法

中,句法元素1266 (“pps_num_exp_slices_in_tile[i]”)等于0,并且句法元素1268 (“pps_exp_slice_height_in_ctus_minus1[i][j]”)的指示被跳过。在第二种方法中,句法元素1266 (“pps_num_exp_slices_in_tile[i]”)等于1,而且pps_exp_slice_height_in_ctus_minus1[i][0]等于RowHeight[SliceTopLeftTileIdx[i]/NumTileColumns]-1(例如,砖块的高度减1)。这两种指示方法都指示该砖块仅包括一个条带。

[0201] 在一些实施例中,可以通过修改句法元素及其语义来消除上述指示方法中的冗余。本公开的实施例提供了PPS句法的更新语义。图30示出了与本公开的一些实施例一致的部分PPS句法结构3000的示例性的修改的编码句法表。如图30所示,来自图12所示的先前句法的更改以突出模式显示,建议删除的句法进一步以删除线显示。

[0202] 与图12所示的SPS句法相比,在一些实施例中,如图30所示,将句法元素1266 (“pps_num_exp_slices_in_tile[i]”)替换为句法元素3066 (“pps_num_exp_slices_in_tile_minus1[i]”) (例如,pps_num_exp_slices_in_tile[i]减1)。因此,第一种指示方法被移除。当一个砖块包括一个条带时,编码器指示句法元素3066 (“pps_num_exp_slices_in_tile_minus1[i]”)等于0,然后指示句法元素1268(例如,“pps_exp_slice_height_in_ctus_minus1[i][0]”)等于砖块高度减1。

[0203] 在图30的句法中,句法元素3066加1指定了为包含第i个条带的砖块(即,砖块索引等于SliceTopLeftTileIdx[i]的砖块)中的条带显式提供的条带高度的数量。句法元素3066的值应在从0到对应砖块高度(RowHeight[SliceTopLeftTileIdx[i]/NumTileColumns])减去2的包含范围内。当不存在时,句法元素3066的值被确定为等于0。

[0204] 在图30的句法中,句法元素1268加1指定以CTU行为单位的包含第i个条带的砖块中第j个矩形条带的高度,索引j在0到句法元素3066 (“pps_num_exp_slices_in_tile_minus1[i]”)的包含范围内。句法元素“pps_exp_slice_height_in_ctus_minus1[i][pps_num_exp_slices_in_tile_minus1[i]]”也用于推导包含第i个条带的砖块中索引大于本文规定的句法元素3066的矩形条带的高度。

[0205] 图31A和图31B分别示出了与本公开的一些实施例一致的在图29中的方法2900的步骤2950中编码或解码PPS中的条带信息的示例性详细操作。当应用图12的句法时,如图31A所示,步骤2950包括步骤2952和2954。在步骤2952,装置编码或解码句法元素1266 (“pps_num_exp_slices_in_tile[i]”),该句法元素指定了为包含第i个条带的砖块中的条带显式提供的条带高度的数量。在步骤2954,装置对句法元素1268 (“pps_exp_slice_height_in_ctus_minus1[i][j]”)进行编码或解码。句法元素1268加1指定包含第i个条带的砖块中第j个矩形条带的高度。

[0206] 当应用图30的修改的句法时,如图31B所示,步骤2950包括步骤2956和2954。在步骤2956,装置编码或解码句法元素3066 (“pps_num_exp_slices_in_tile_minus1[i]”)。句法元素3066加1指定为包含第i个条带的砖块中的条带显式提供的条带高度的数量。

[0207] 图32示出了与本公开的一些实施例一致的示例性视频编码或解码方法3200的流程图。通过应用视频编码或解码方法3200,一个或多个PPS句法元素或标志可以基于其他句法元素或标志的值有条件地指示,这减少了输出比特的数量并实现了更高的编码性能。当不在PPS中进行指示时,可以相应地确定或分配PPS句法元素或标志的值。与方法2700A-2700C和2900类似,方法3200可以由编码器或解码器执行,编码器或解码器可以实现为用于

编码或转码视频序列,或解码比特流以重构视频流的装置(例如,图4中的装置400)的一个或多个软件或硬件组件。例如,一个或多个处理器(例如,图4中的处理器402)可以执行方法3200。

[0208] 在步骤3205,装置被配置为在PPS中编码或解码PPS标志1210(“pps_no_pic_partition_flag”),PPS标志1210用于指示参考PPS的图像是否被分割。如上所述,在一些实施例中,对于同一CLVS内的图像,不同的PPS可以独立地指示具有不同值的PPS标志1210。例如,该装置可以在第一PPS中编码或解码指示参考第一PPS的第一图像是否被分割为砖块或条带的第一标志,并在第二PPS中编码或解码指示参考第二PPS的第二图像是否被分割的第二标志,其中第一标志和第二标志具有不同的值。

[0209] 在步骤3210,装置基于PPS标志1210的值确定参考PPS的图像是否未分割。当PPS标志1210等于1(步骤3210-是)时,参考PPS的图像未分割,并且装置执行步骤3220和3230以跳过编码或解码标志1250,并确定标志1250的值以指示每个子图像包括单个矩形条带。例如,在步骤3230,可以将标志1250确定为1。当PPS标志1210等于零时(步骤3210-否),装置执行步骤3225以编码或解码标志1250。

[0210] 在步骤3230或步骤3225中确定或编码/解码标志1250的值后,在步骤3240,对于在从0到等于图像中矩形条带的数量减去2的上限值的范围内的索引*i*,当PPS中缺少句法元素1262(“pps_slice_width_in_tiles_minus1[*i*]”)时,该装置确定句法元素1262为零。对于索引*i*,句法元素1262是与第*i*个矩形条带的宽度相关联的图像参数。在一些实施例中,句法元素1262的值在从0到对应砖块的列数(NumTileColumns)减去1的包含范围内。

[0211] 作为步骤3250,对于在从0到等于图像中矩形条带的数量减去2的上限值的范围内的索引*i*,当PPS中缺少句法元素1264(“pps_slice_height_in_tiles_minus1[*i*]”)时,该装置确定句法元素1264的值。对于索引*i*,句法元素1264是与第*i*个矩形条带的高度相关联的图像参数。在一些实施例中,句法元素1264的值在从0到对应砖块的行数(例如,NumTileRows)减去1的包含范围内。如在上述实施例中所讨论的,当不存在时,第*i*个矩形条带的句法元素1264可以被确定为零或等于第(*i*-1)个矩形条带的句法元素1264的值。

[0212] 在一些其他实施例中,还可以修改句法元素1268的语义,以反映句法元素1268的值在从0到RowHeight[SliceTopLeftTileIdx[*i*]/NumTileColumns]减去2的包含范围内。或者说,句法元素1268是使用对应砖块高度的上限值减去2来编码或解码的图像参数。

[0213] 由于句法元素1268的最大值被减小到砖块的高度减去2,第二种指示方法被移除。因此,当对应砖块包含单个条带时,句法元素1266被编码或解码为等于0。当对应砖块包含两个或多个条带时,句法元素1266被编码或解码为等于或大于1。

[0214] 如上所讨论的,在一些实施例中,对于一个砖块内的矩形条带,句法元素1266指定了为包含第*i*个条带的砖块中的条带显式提供的条带高度的数量。当句法元素1266大于0时,句法元素1268加1指定以CTU行为单位的包含第*i*个条带的砖块中第*j*个矩形条带的高度,索引*j*在从0到句法元素1266减去1的包含范围内。句法元素1268的值在从0到RowHeight[SliceTopLeftTileIdx[*i*]/NumTileColumns]减去1的包含范围内。

[0215] 应当注意,对于砖块中的每个单独的条带,每个被显式指示的条带高度在从0到砖块高度的范围内。另外,考虑到一个砖块内所有条带的高度之和应等于砖块的高度,被显式指示的条带高度之和小于或等于砖块的高度。由于条带的最小高度等于1,每个被显式指示

的条带高度减去1应小于或等于砖块高度减去砖块中的条带的总数。

[0216] 本公开的实施例提供了句法元素1268的更新语义。因此,可以将每个被显式指示的条带高度的范围减小到更精确的值。特别是,句法元素1268(“pps_exp_slice_height_in_ctus_minus1[i][j]”)的最大值可以更改为 $\text{RowHeight}[\text{SliceTopLeftTileIdx}[i]/\text{NumTileColumns}]$ 减去为当前砖块中的条带显式提供的条带高度的数量。

[0217] 当与图12所示的实施例相结合时,可以修改句法元素1268的语义,以反映句法元素1268的值应在从0到 $\text{RowHeight}[\text{SliceTopLeftTileIdx}[i]/\text{NumTileColumns}]$ 减去句法元素1266的值的包含范围内。或者说,句法元素1268的上限值是对应砖块的高度减去为对应砖块中的条带显式提供的条带高度的数量。

[0218] 当与图30所示的用句法元素3066替换句法元素1266的实施例相结合时,可以修改句法元素1268的语义,以反映句法元素1268的值应在从0到 $\text{RowHeight}[\text{SliceTopLeftTileIdx}[i]/\text{NumTileColumns}]$ 减去句法元素3066的值再减去1的包含范围内。

[0219] 当与将句法元素1268的最大值减少到砖块的高度减去2的实施例相结合时,可以使用函数 $\max(\text{pps_num_exp_slices_in_tile}[i], 2)$,该函数选择句法元素1266(“pps_num_exp_slices_in_tile[i]”)和2中的较大者,句法元素1268的语义可以进一步修改,以反映句法元素1268的值应在从0到 $\text{RowHeight}[\text{SliceTopLeftTileIdx}[i]/\text{NumTileColumns}]$ 减去 $\max(\text{pps_num_exp_slices_in_tile}[i], 2)$ 的包含范围内。或者说,句法元素1268的上限值是相应砖块的高度(例如, $\text{RowHeight}[\text{SliceTopLeftTileIdx}[i]/\text{NumTileColumns}]$)减去2,或者相应砖块的高度减去相应砖块中显式提供的砖块高度的数量($\text{pps_num_exp_slices_in_tile}[i]$)中的较小值。

[0220] 图33示出了根据本公开的一些实施例的包括变量 NumCtusInSlice , $\text{SliceTopLeftTileIdx}$, CtbAddrInSlice , 和 NumSlicesInTile 的更新推导的示例性伪代码。如图33所示,与先前VVC的更改以突出模式显示,建议删除的句法进一步以删除线显示。

[0221] 图34示出了根据本公开的一些实施例的包括变量 NumCtusInSlice 和 CtbAddrInSlice 的更新推导的示例性伪代码。如图34所示,与先前VVC的更改以斜体显示,建议删除的句法进一步以删除线显示。在图34所示的伪代码中,纠正了先前VVC伪代码中 $\text{NumCtusInSlice}[i]$ 和 $\text{CtbAddrInSlice}[i][j]$ 的推导中的错误和一个冗余行。

[0222] 综上所述,如本公开的各种实施例中所提出的,通过修改与砖块/条带分割相关的SPS和PPS句法元素,编码/解码方法可以是一致和高效的。另外,通过在句法元素未进行指示时正确确定句法元素的值,在某些情况下与砖块/条带分割相关的一些句法元素的指示可以被跳过,这减少了输出比特的数量,因此提高了编码效率。

[0223] 虽然本公开的各种实施例是关于当前的VVC标准描述的,但可以理解的是,各种实施例适用于其他视频编码技术。

[0224] 可以使用以下条款进一步描述实施例:

[0225] 1. 一种用于编码或解码视频的方法,包括:

[0226] 在与编码层视频序列(CLVS)的图像相关联的多个图像参数集(PPS)中,编码或解码对应的第一PPS标志,第一PPS标志指示图像是否被允许分割为多个砖块或条带,其中:

[0227] 在第一PPS中,具有第一值的对应第一PPS标志指示CLVS的第一图像未分割,并且

在第二PPS中,具有与第一值不同的第二值的另一对应第一PPS标志指示CLVS的第二图像被允许分割。

[0228] 2.一种用于编码或解码视频的方法,包括:

[0229] 确定分割图像的砖块的数量是否大于阈值,阈值大于1;以及

[0230] 响应于该分割图像的砖块的数量是否大于阈值,编码或解码与应用于该分割图像的条带模式相关联的第二PPS标志。

[0231] 3.根据第1条或第2条的方法,其中,编码或解码第二PPS标志包括:

[0232] 当光栅扫描条带模式应用于参考PPS的该分割图像时,编码或解码具有第三值的第二PPS标志;或者,当矩形条带模式应用于参考PPS的该分割图像时,编码或解码具有不同于第三值的第四值的第二PPS标志。

[0233] 4.根据第3条的方法,还包括:

[0234] 当矩形条带模式应用于参考该PPS的该分割图像时,编码或解码该PPS中指定条带信息的一个或多个句法元素。

[0235] 5.一种用于编码或解码视频的方法,包括:

[0236] 确定图像是否被分割为多个砖块或条带;以及

[0237] 响应于确定图像被允许分割为多个砖块或条带,在图像参数集(PPS)中编码或解码与应用于参考该PPS的图像的条带模式相关联的第一标志,其中:

[0238] 当光栅扫描条带模式应用于分割该图像时,编码或解码具有第一值的第一标志;或者当矩形条带模式应用于分割该图像时,编码或解码具有与第一值不同的第二值的第一标志。

[0239] 6.根据第3-5条中任一条的方法,其中:

[0240] 在光栅扫描条带模式中,该图像被分割为多个光栅扫描条带,并且光栅扫描条带中的任一个包括该图像的砖块光栅扫描中的一个或多个完整砖块的序列;并且

[0241] 在矩形条带模式中,该图像被分割为多个矩形条带,其中矩形条带中的一个包括一个或多个覆盖该图像的矩形区域的砖块,或者矩形条带中的一个覆盖砖块的编码树单元的一个或多个连续行。

[0242] 7.根据第5条或第6条的方法,还包括:

[0243] 编码或解码第一图像参数,其指定了为对应砖块中的条带提供的条带高度的数量减去1,如果对应砖块包含单个条带,则第一图像参数等于0。

[0244] 8.根据第7条的方法,其中,第一图像参数的值在从0到对应砖块高度减去2的包含范围内。

[0245] 9.根据第5-8条中任一条的方法,还包括:

[0246] 编码或解码第三图像参数,其指定了为对应砖块中的条带提供的砖块高度的数量,如果对应砖块包含单个条带,则第三图像参数等于0。

[0247] 10.根据第9条的方法,还包括:

[0248] 编码或解码第二图像参数,其指定对应砖块中对应矩形条带的高度减去1。

[0249] 11.根据第10条的方法,其中第二图像参数的上限值是对应砖块的高度减去2。

[0250] 12.根据第10条的方法,其中第二图像参数的上限值是对应砖块的高度减去第四图像参数,其中第四图像参数指示为对应砖块中的条带提供的条带高度的数量。

[0251] 13. 根据第10条的方法, 其中, 第二图像参数的上限值是对应砖块的高度减去第五图像参数, 再减去1,

[0252] 其中, 第五图像参数指示为对应砖块中的条带提供的条带高度的数量减去1。

[0253] 14. 根据第10条的方法, 其中, 第二图像参数的上限值是对应砖块的高度减去2或对应砖块的高度减去对应砖块中提供的条带高度的数量中的较小值。

[0254] 15. 一种用于编码或解码视频的方法, 包括:

[0255] 在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中, 编码或解码第一PPS标志, 第一PPS标志指示关联图像是否被允许分割为多个砖块或条带, 其中, 第一PPS标志等于第一值指示关联图像未分割, 或者第一PPS标志等于与第一值不同的第二值指示关联图像被允许分割; 以及

[0256] 当第一PPS标志等于第一值时, 在PPS中跳过编码或解码第二PPS标志, 第二PPS标志指示是否关联图像的每个子图像包括单个矩形条带, 并确定第二PPS标志的值等于第三值, 其中, 等于第三值的第二PPS标志指示关联图像的每个子图像包括单个矩形条带。

[0257] 16. 一种用于编码或解码视频的方法, 包括:

[0258] 对于索引 i , 在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中, 编码或解码与第 i 个矩形条带的宽度相关联的第一图像参数; 以及

[0259] 当第一图像参数不存在于PPS中时, 确定第一图像参数为零,

[0260] 其中, 索引 i 在从0到等于图像中的矩形条带的数量减去2的上限值的范围内。

[0261] 17. 根据第16条的方法, 还包括:

[0262] 对于索引 i , 编码或解码与第 i 个矩形条带的高度相关联的第二图像参数; 以及

[0263] 当对于索引 i , 第二图像参数不存在于PPS中时, 确定第二图像参数为零或等于与第 $(i-1)$ 个矩形条带的高度相关联的第三图像参数的值。

[0264] 18. 一种用于编码或解码视频的方法, 包括:

[0265] 在比特流的序列参数集 (SPS) 中, 编码或解码第一SPS标志, 第一SPS标志指示参考SPS的编码层视频序列 (CLVS) 中的一个或多个图像是否被分割为多个砖块或条带; 以及

[0266] 在与SPS相关联的图像参数集 (PPS) 中, 编码或解码等于第一SPS标志的第一PPS标志, 第一PPS标志用于指示参考PPS的图像是否被分割。

[0267] 19. 根据第18条的方法, 还包括:

[0268] 响应于指示该一个或多个图像未分割的第一SPS标志, 跳过编码或解码SPS中的第二SPS标志, 第二SPS标志指示对于参考SPS的CLVS, 子图像信息是否存在。

[0269] 20. 根据第18条或第19条的方法, 还包括:

[0270] 响应于指示该一个或多个图像未分割的第一SPS标志, 跳过编码或解码与参考SPS的CLVS中的每个图像中的子图像的数量相关联的第一SPS序列参数。

[0271] 21. 一种装置, 包括:

[0272] 存储器, 被配置为存储指令; 以及

[0273] 一个或多个处理器, 被配置为执行指令以使装置:

[0274] 在与编码层视频序列 (CLVS) 的图像相关联的多个图像参数集 (PPS) 中, 编码或解码对应的第一PPS标志, 对应的第一PPS标志指示图像是否被允许分割为多个砖块或条带, 其中:

[0275] 在第一PPS中,具有第一值的对应的第一PPS标志指示CLVS的第一图像未分割,并且在第二PPS中,具有与第一值不同的第二值的另一对应的第一PPS标志指示CLVS的第二图像被允许分割。

[0276] 22.一种装置,包括:

[0277] 存储器,被配置为存储指令;以及

[0278] 一个或多个处理器,被配置为执行指令以使装置:

[0279] 确定分割图像的砖块的数量是否大于阈值,阈值大于1;以及

[0280] 响应于分割图像的砖块的数量大于该阈值,编码或解码与应用于分割图像的条带模式相关联的第二PPS标志。

[0281] 23.根据第21条或第22条的装置,其中,该一个或多个处理器被配置为执行指令,以使装置通过以下方式编码或解码第二PPS标志:

[0282] 当光栅扫描条带模式应用于参考PPS的分割图像时,编码或解码具有第三值的第二PPS标志;或者,当矩形条带模式应用于参考PPS的分割图像时,编码或解码具有不同于第三值的第四值的该第二PPS标志。

[0283] 24.根据第23条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0284] 当矩形条带模式应用于参考PPS的分割图像时,编码或解码PPS中指定条带信息的一个或多个句法元素。

[0285] 25.一种装置,包括:

[0286] 存储器,被配置为存储指令;以及

[0287] 一个或多个处理器,被配置为执行指令以使装置:

[0288] 确定图像是否被分割为多个砖块或条带;以及

[0289] 响应于确定图像被允许分割为多个砖块或条带,在图像参数集(PPS)中编码或解码与应用于参考PPS的图像的条带模式相关联的第一标志,其中:

[0290] 当光栅扫描条带模式应用于分割该图像时,编码或解码具有第一值的第一标志;或者,当矩形条带模式应用于分割该图像时,编码或解码具有与第一值不同的第二值的第一标志。

[0291] 26.根据第25条的装置,其中:

[0292] 在光栅扫描条带模式中,该图像被分割为多个光栅扫描条带,并且光栅扫描条带中的任一个包括该图像的砖块光栅扫描中的一个或多个完整砖块的序列;以及

[0293] 在矩形条带模式中,该图像被分割为多个矩形条带,其中,矩形条带中的一个包括一个或多个覆盖该图像的矩形区域的砖块,或者矩形条带中的一个覆盖砖块的编码树单元的一个或多个连续行。

[0294] 27.根据第25条或第26条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0295] 编码或解码第一图像参数,其指定了为对应砖块中的条带提供的条带高度的数量减去1,如果对应砖块包含单个条带,则第一图像参数等于0。

[0296] 28.根据第27条的装置,其中,第一图像参数的值在从0到对应砖块高度减去2的包含范围内。

[0297] 29. 根据第25-28条中任一条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0298] 编码或解码第三图像参数,其指定了为对应砖块中的条带提供的砖块高度的数量,如果对应砖块包含单个条带,则第三图像参数等于0。

[0299] 30. 根据第29条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0300] 编码或解码第二图像参数,其指定对应砖块中对应矩形条带的高度减去1。

[0301] 31. 根据第30条的装置,其中,第二图像参数的上限值是对应砖块的高度减去2。

[0302] 32. 根据第30条的装置,其中,第二图像参数的上限值是对应砖块的高度减去第四图像参数,其中,第四图像参数指示为对应砖块中的条带提供的条带高度的数量。

[0303] 33. 根据第30条的装置,其中,第二图像参数的上限值是对应砖块的高度减去第五图像参数,再减去1,

[0304] 其中,第五图像参数指示为对应砖块中的条带提供的条带高度的数量减去1。

[0305] 34. 根据第30条的装置,其中,第二图像参数的上限值是对应砖块的高度减去2或对应砖块的高度减去对应砖块中提供的条带高度的数量中的较小值。

[0306] 35. 一种装置,包括:

[0307] 存储器,被配置为存储指令;以及

[0308] 一个或多个处理器,被配置为执行指令以使该装置:

[0309] 在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中,编码或解码第一PPS标志,第一PPS标志指示关联图像是否被允许分割为多个砖块或条带,其中,第一PPS标志等于第一值指示关联图像未分割,或者第一PPS标志等于与第一值不同的第二值指示关联图像被允许分割;以及

[0310] 当第一PPS标志等于第一值时,在PPS中跳过编码或解码第二PPS标志,第二PPS标志指示是否关联图像的每个子图像包括单个矩形条带,并确定第二PPS标志的值等于第三值,其中,等于第三值的第二PPS标志指示关联图像的每个子图像包括单个矩形条带。

[0311] 36. 一种装置,包括:

[0312] 存储器,被配置为存储指令;以及

[0313] 一个或多个处理器,被配置为执行指令以使该装置:

[0314] 对于索引*i*,在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中,编码或解码与第*i*个矩形的宽度相关联的第一图像参数;以及

[0315] 当第一图像参数不存在于PPS中时,确定第一图像参数为零,

[0316] 其中,索引*i*在从0到等于图像中的矩形条带的数量减去2的上限值的范围内。

[0317] 37. 根据第36条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0318] 对于索引*i*,编码或解码与第*i*个矩形条带的高度相关联的第二图像参数;以及

[0319] 当对于索引*i*,第二图像参数不存在于PPS中时,确定第二图像参数为零或等于与第(*i*-1)个矩形条带的高度相关联的第三图像参数的值。

[0320] 38. 一种装置,包括:

[0321] 存储器,被配置为存储指令;以及

[0322] 一个或多个处理器,被配置为执行指令以使该装置:

[0323] 在比特流的序列参数集 (SPS) 中,编码或解码第一SPS标志,第一SPS标志指示参考SPS的编码层视频序列 (CLVS) 中的一个或多个图像是否被分割为多个砖块或条带;以及

[0324] 在与SPS相关联的图像参数集 (PPS) 中,编码或解码等于第一SPS标志的第一PPS标志,该、第一PPS标志指示参考PPS的图像是否被分割。

[0325] 39. 根据第38条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0326] 响应于指示该一个或多个图像未分割的第一SPS标志,跳过编码或解码SPS中的第二SPS标志,第二SPS标志指示对于参考SPS的CLVS,子图像信息是否存在。

[0327] 40. 根据第38条或第39条的装置,其中,该一个或多个处理器被配置为执行指令,以进一步使该装置:

[0328] 响应于指示该一个或多个图像未分割的第一SPS标志,跳过编码或解码与参考SPS的CLVS中的每个图像中的子图像的数量相关联的第一SPS序列参数。

[0329] 41. 一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由设备的一个或多个处理器执行,以使该设备执行包括如下的操作:

[0330] 在与编码层视频序列 (CLVS) 的图像相关联的多个图像参数集 (PPS) 中,编码或解码对应的第一PPS标志,对应的第一PPS标志指示图像是否被允许分割为多个砖块或条带,其中:

[0331] 在第一PPS中,具有第一值的对应的第一PPS标志指示CLVS的第一图像未分割,并且在第二PPS中,具有与第一值不同的第二值的另一对应的第一PPS标志指示CLVS的第二图像被允许分割。

[0332] 42. 一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由设备的一个或多个处理器执行,以使该设备执行包括以下的操作:

[0333] 确定分割图像的砖块的数量是否大于阈值,阈值大于1;以及

[0334] 响应于分割图像的砖块的数量大于阈值,编码或解码与应用于分割图像的条带模式相关联的第二PPS标志。

[0335] 43. 根据第41条或第42条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备通过以下方式编码或解码第二PPS标志:

[0336] 当光栅扫描条带模式应用于参考PPS的分割图像时,编码或解码具有第三值的第二PPS标志;或者,当矩形条带模式应用于参考PPS的分割图像时,编码或解码具有不同于第三值的第四值的第二PPS标志。

[0337] 44. 根据第43条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0338] 当矩形条带模式应用于参考PPS的分割图像时,编码或解码PPS中指定条带信息的一个或多个句法元素。

[0339] 45. 一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由该设备的一个或多个处理器执行,以使该设备执行包括以下的操作:

[0340] 确定图像是否被分割为多个砖块或条带;以及

[0341] 响应于确定图像被允许分割为多个砖块或条带,在图像参数集 (PPS) 中编码或解

码与应用于参考PPS的图像的条带模式相关联的第一标志,其中:

[0342] 当光栅扫描条带模式应用于分割该图像时,编码或解码具有第一值的第一标志;或者,当矩形条带模式应用于分割该图像时,编码或解码具有与该第一值不同的第二值的第一标志。

[0343] 46. 根据第45条的非暂时性计算机可读存储介质,其中:

[0344] 在光栅扫描条带模式中,该图像被分割为多个光栅扫描条带,并且光栅扫描条带中的任一个包括该图像的砖块光栅扫描中的一个或多个完整砖块的序列;并且

[0345] 在矩形条带模式中,该图像被分割为多个矩形条带,其中,矩形条带中的一个包括一个或多个覆盖该图像的矩形区域的砖块,或者该矩形条带中的一个覆盖砖块的编码树单元的一个或多个连续行。

[0346] 47. 根据第45条或第46条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0347] 编码或解码第一图像参数,第一图像参数指定为对应砖块中的条带提供的条带高度的数量减去1,如果对应砖块包含单个条带,则第一图像参数等于0。

[0348] 48. 根据第47条的非暂时性计算机可读存储介质,其中,该第一图像参数的值在从0到对应砖块的高度减去2的包含范围内。

[0349] 49. 根据第45-48条中任一条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0350] 编码或解码第三图像参数,第三图像参数指定为对应砖块中的条带提供的砖块高度的数量,如果对应砖块包含单个条带,则第三图像参数等于0。

[0351] 50. 根据第49条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0352] 编码或解码第二图像参数,第二图像参数指定为对应砖块中对应矩形条带的高度减去1。

[0353] 51. 根据第50条的非暂时性计算机可读存储介质,其中,第二图像参数的上限值是对应砖块的高度减去2。

[0354] 52. 根据第50条的非暂时性计算机可读存储介质,其中,第二图像参数的上限值是该对应砖块的高度减去第四图像参数,其中,第四图像参数指示为对应砖块中的条带提供的条带高度的数量。

[0355] 53. 根据第50条所述的非暂时性计算机可读存储介质,其中,第二图像参数的上限值是对应砖块的高度减去第五图像参数,再减去1,

[0356] 其中,第五图像参数指示为对应砖块中的条带提供的条带高度的数量减去1。

[0357] 54. 根据第50条的非暂时性计算机可读存储介质,其中,第二图像参数的上限值是对应砖块的高度减去2或对应砖块的高度减去对应砖块中提供的条带高度的数量中的较小值。

[0358] 55. 一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由设备的一个或多个处理器执行,以使该设备执行包括以下的操作:

[0359] 在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中,编码或解码第一PPS标志,第一PPS标志指示关联图像是否被允许分割为多个砖块或条带,其中,

第一PPS标志等于第一值指示关联图像未分割,或者第一PPS标志等于与第一值不同的第二值指示关联图像被允许分割;以及

[0360] 当第一PPS标志等于第一值时,在PPS中跳过编码或解码第二PPS标志,第二PPS标志指示是否关联图像的每个子图像包括单个矩形条带,并确定第二PPS标志的值等于第三值,其中,等于第三值的第二PPS标志指示关联图像的每个子图像包括单个矩形条带。

[0361] 56.一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由设备的一个或多个处理器执行,以使该设备执行包括以下的操作:

[0362] 对于索引*i*,在与编码层视频序列 (CLVS) 的至少一个图像相关联的图像参数集 (PPS) 中,编码或解码与第*i*个矩形条带的宽度相关联的第一图像参数;以及

[0363] 当第一图像参数不存在于PPS中时,确定第一图像参数为零,

[0364] 其中,索引*i*在从0到等于图像中的矩形条带的数量减去2的上限值的范围内。

[0365] 57.根据第56条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0366] 对于索引*i*,编码或解码与第*i*个矩形条带的高度相关联的第二图像参数;以及

[0367] 当对于索引*i*,第二图像参数不存在于PPS中时,确定第二图像参数为零或等于与第(*i*-1)个矩形条带的高度相关联的第三图像参数的值。

[0368] 58.一种非暂时性计算机可读存储介质,存储一组指令,该一组指令可由设备的一个或多个处理器执行,以使该设备执行包括以下的操作:

[0369] 在比特流的序列参数集 (SPS) 中,编码或解码第一SPS标志,第一SPS标志指示参考SPS的编码层视频序列 (CLVS) 中的一个或多个图像是否被分割为多个砖块或条带;以及

[0370] 在与SPS相关联的图像参数集 (PPS) 中,编码或解码等于第一SPS标志的第一PPS标志,第一PPS标志指示参考PPS的图像是否被分割。

[0371] 59.根据第58条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0372] 响应于指示一个或多个图像未分割的第一SPS标志,跳过编码或解码SPS中的第二SPS标志,第二SPS标志指示对于参考SPS的CLVS,子图像信息是否存在。

[0373] 60.根据第58条或第59条的非暂时性计算机可读存储介质,其中,可由该设备的一个或多个处理器执行的该一组指令还使该设备执行包括以下的操作:

[0374] 响应于指示该一个或多个图像未分割的第一SPS标志,跳过编码或解码与参考SPS的CLVS中的每个图像中的子图像的数量相关联的第一SPS序列参数。

[0375] 在一些实施例中,也提供了包括指令的非暂时性计算机可读存储介质,并且该指令可以由设备(例如所公开的编码器和解码器)执行,以执行上述方法。非暂时性介质的常见形式包括,例如软盘、可折叠磁盘、硬盘、固态驱动器、磁带或任何其他磁性数据存储介质、CD-ROM、任何其他光学数据存储介质、任何具有孔图案的物理介质、RAM、PROM和EPROM、FLASH-EPROM或任何其他闪存、NVRAM、高速缓存、寄存器、任何其他存储芯片或盒及其网络版本。该设备可以包括一个或多个处理器(CPU)、输入/输出接口、网络接口和/或存储器。

[0376] 应当注意,本文中的关系术语,诸如“第一”和“第二”等,仅用于区分一个实体或操作与另一实体或操作,并不要求或暗示这些实体或操作之间的任何实际关系或顺序。此外,词语“包括”、“具有”、“包含”和“包括”以及其他类似形式旨在具有相同的含义并且是开放

式的,因为在这些词语中任何一个之后的一个或多个项目并不意味着是这一个或多个项目的详尽列表,或仅限于所列的一个或多个项目。

[0377] 如本文所用,除非另有特别说明,否则术语“或”包括所有可能的组合,除非不可行。例如,如果声明数据库可能包括A或B,则除非另有明确说明或不可行,否则数据库可能包括A、或B,或A和B。作为第二个示例,如果声明数据库可能包括A、B或C,则除非另有明确说明或不可行,否则数据库可能包括A、或B或C,或A和B,或A和C,或B和C,或A和B和C。

[0378] 可以理解,上述实施例可以通过硬件或软件(程序代码)或硬件和软件的组合来实现。如果通过软件实现,则可以将其存储在上述计算机可读介质中。该软件当由处理器执行时,可以执行所公开的方法。本公开中描述的计算单元和其他功能单元可以通过硬件、软件或硬件和软件的组合来实现。本领域的普通技术人员还将理解,上述模块/单元中的多个可以组合为一个模块/单元,并且上述每个模块/单元可以进一步划分为多个子模块/子单元。

[0379] 在前述说明书中,已经参考可以随实施方式而变化的许多具体细节对实施例进行了描述。可以对所描述的实施例进行某些调整和修改。考虑到本文公开的本公开的说明书和实践,其他实施例对于本领域技术人员而言可以是显而易见的。本说明书和示例仅被视为示例性的,本公开的真正范围和精神由以下权利要求指示。图中所示的步骤顺序也仅用于说明目的,并不旨在限于任何特定的步骤顺序。因此,本领域技术人员可以理解,在实现相同方法的同时,可以以不同的顺序执行这些步骤。

[0380] 在附图和说明书中,已经公开了示例性实施例。然而,可以对这些实施例进行许多变化和修改。因此,尽管使用了特定术语,但它们仅在一般和描述性意义上使用,而不是出于限制的目的。

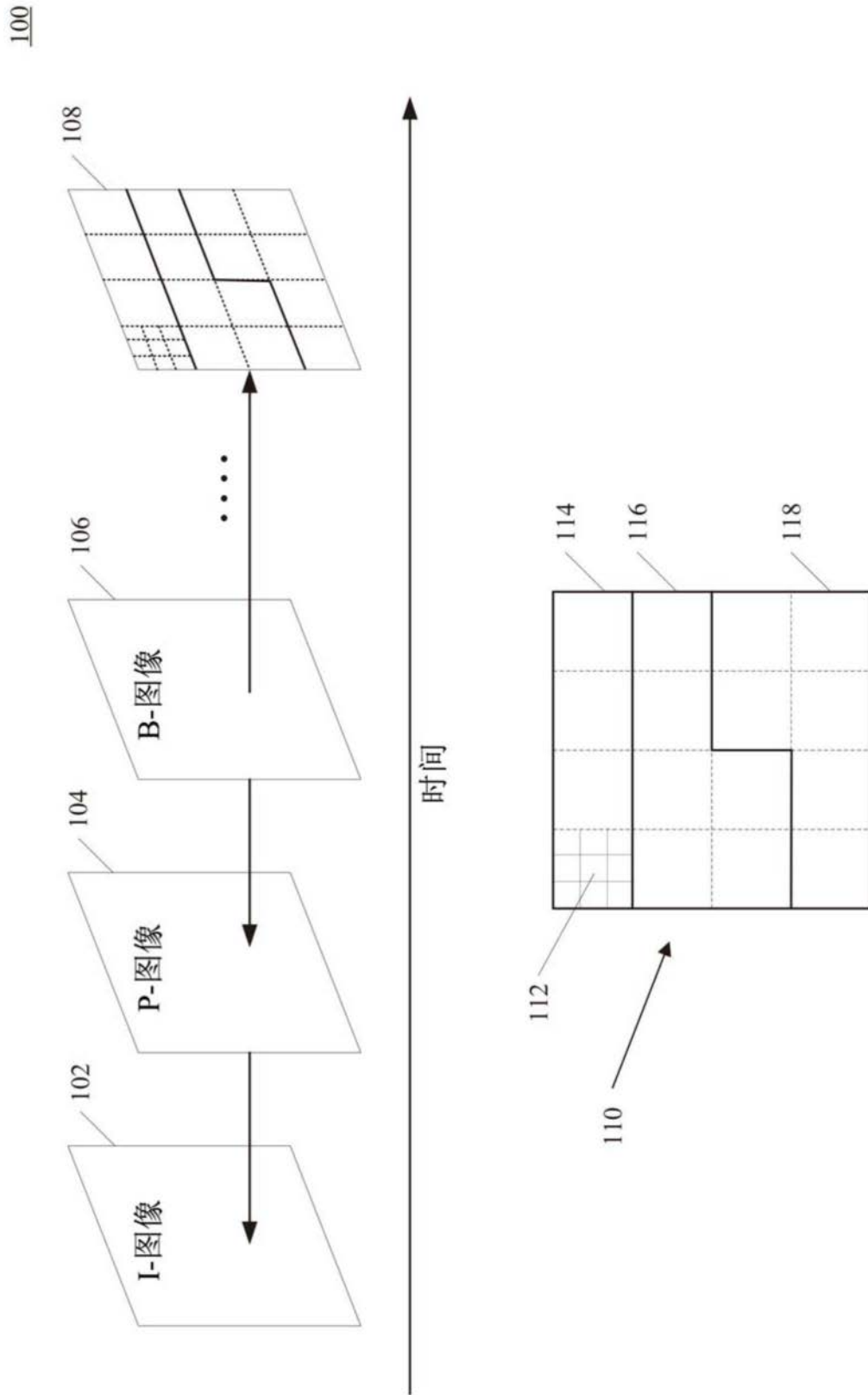


图1

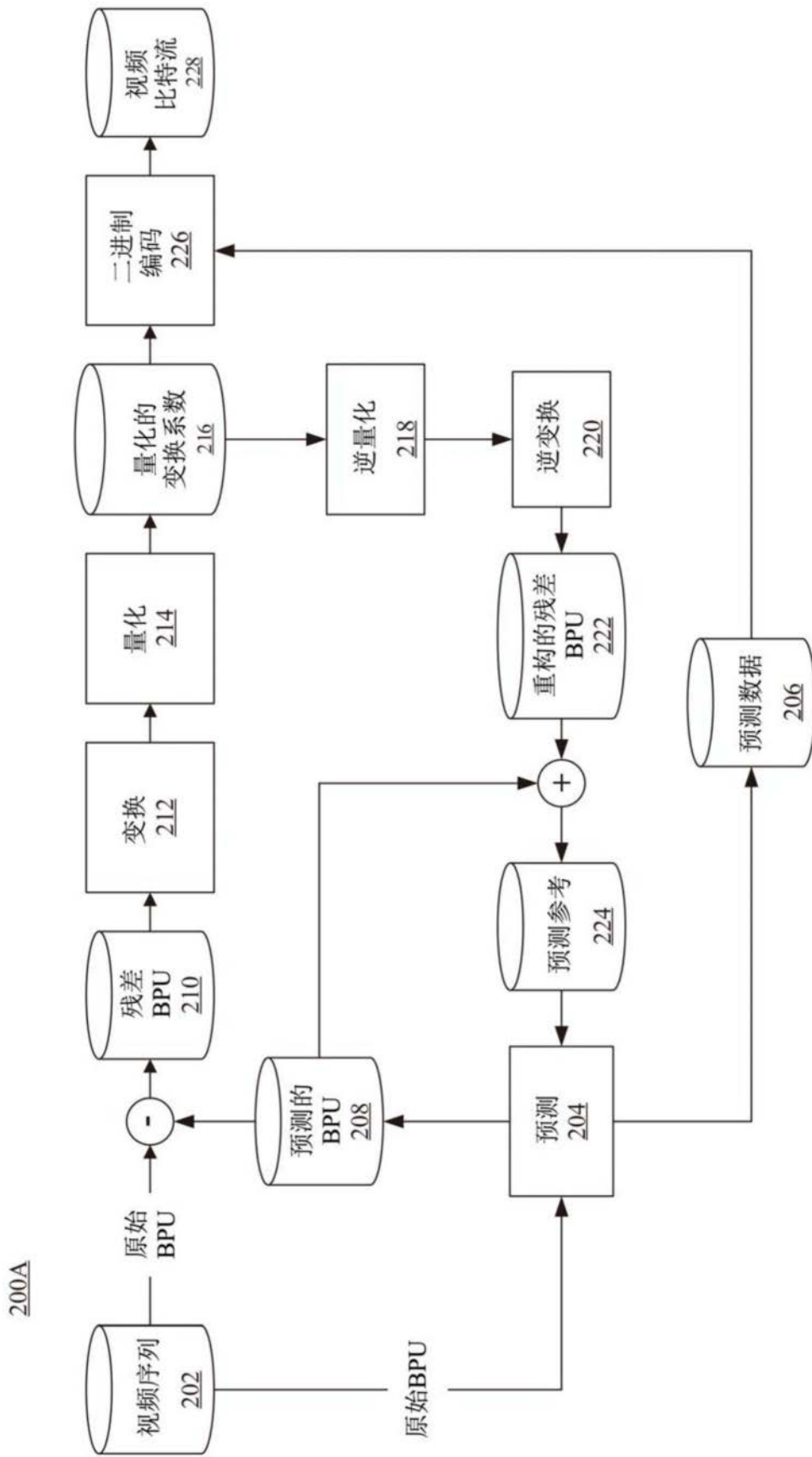


图2A

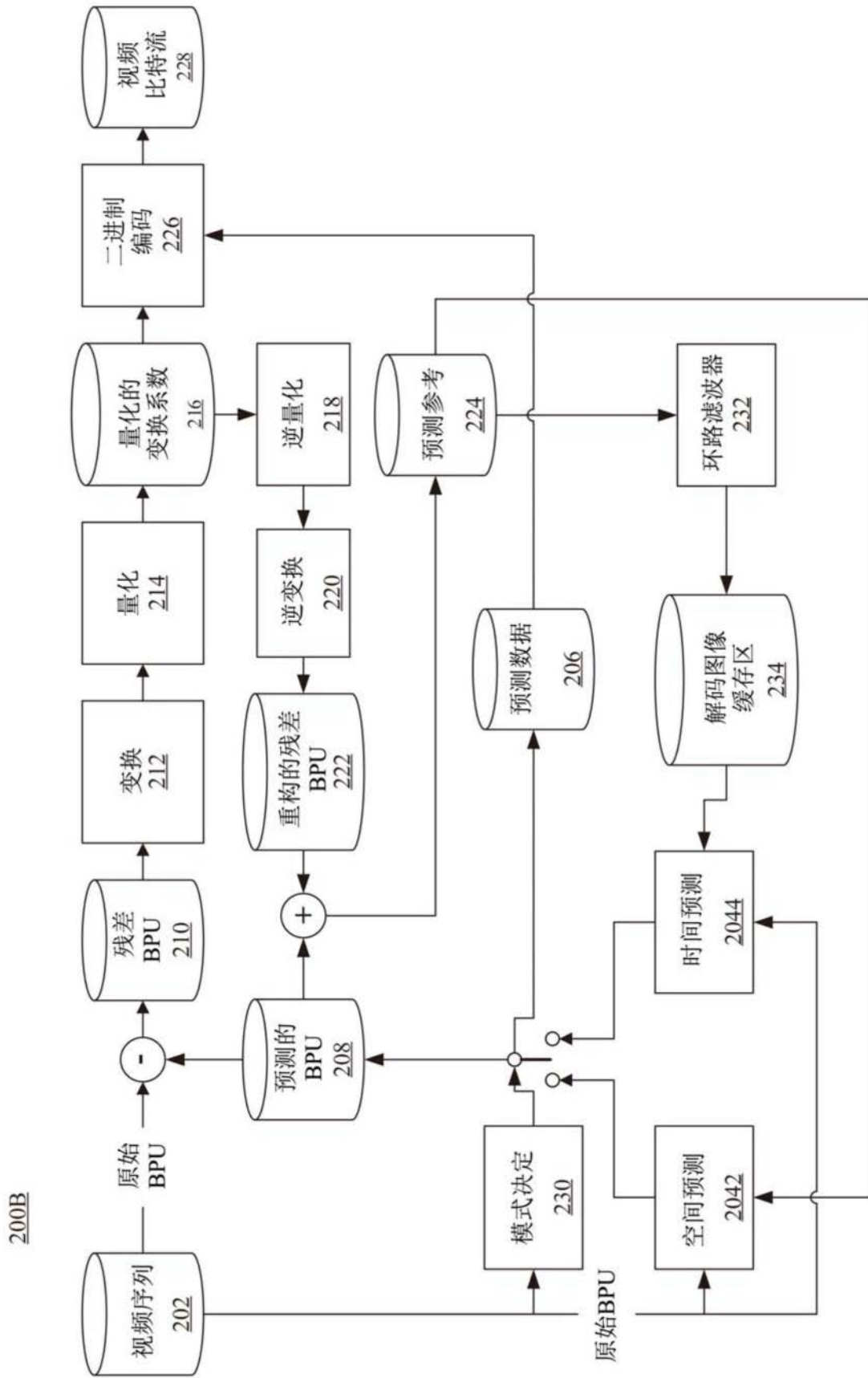


图2B

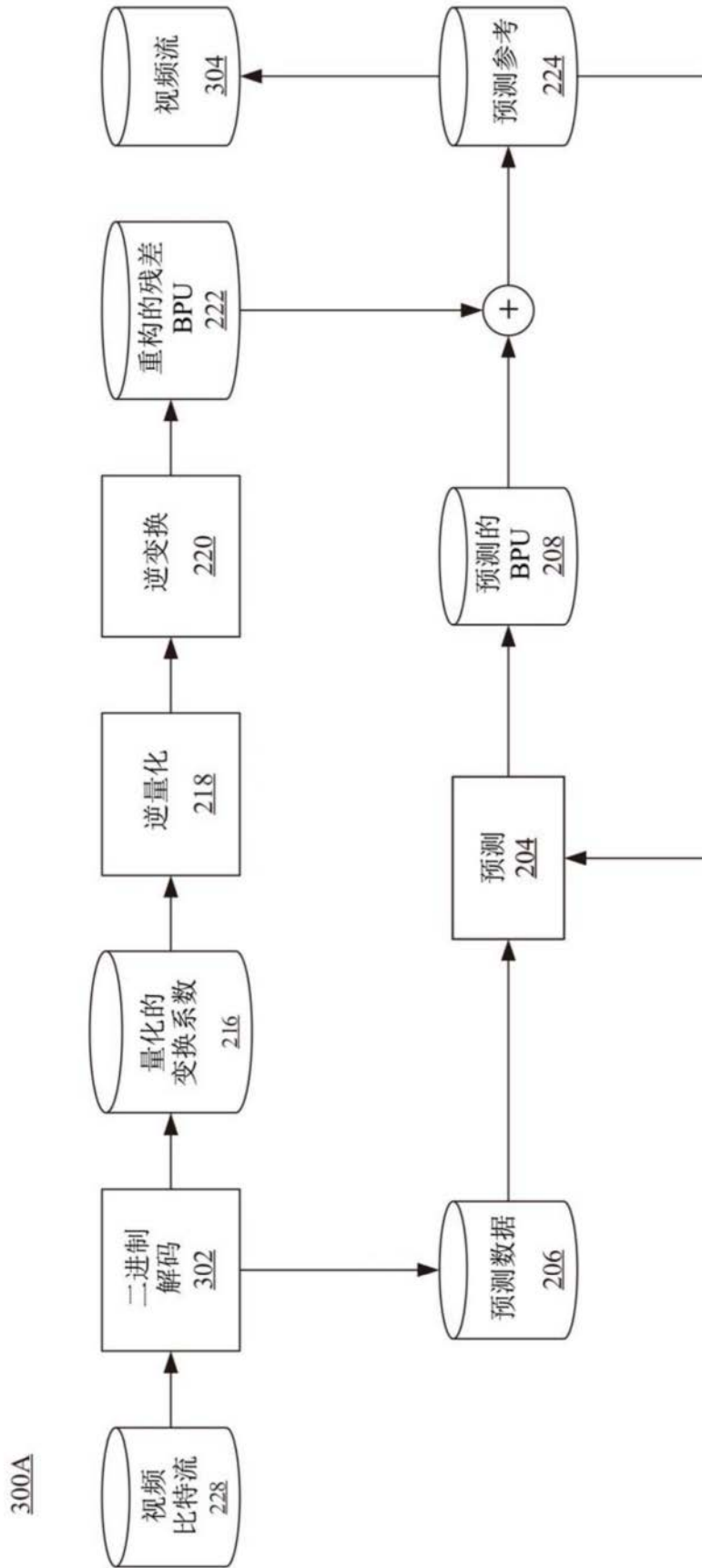


图3A

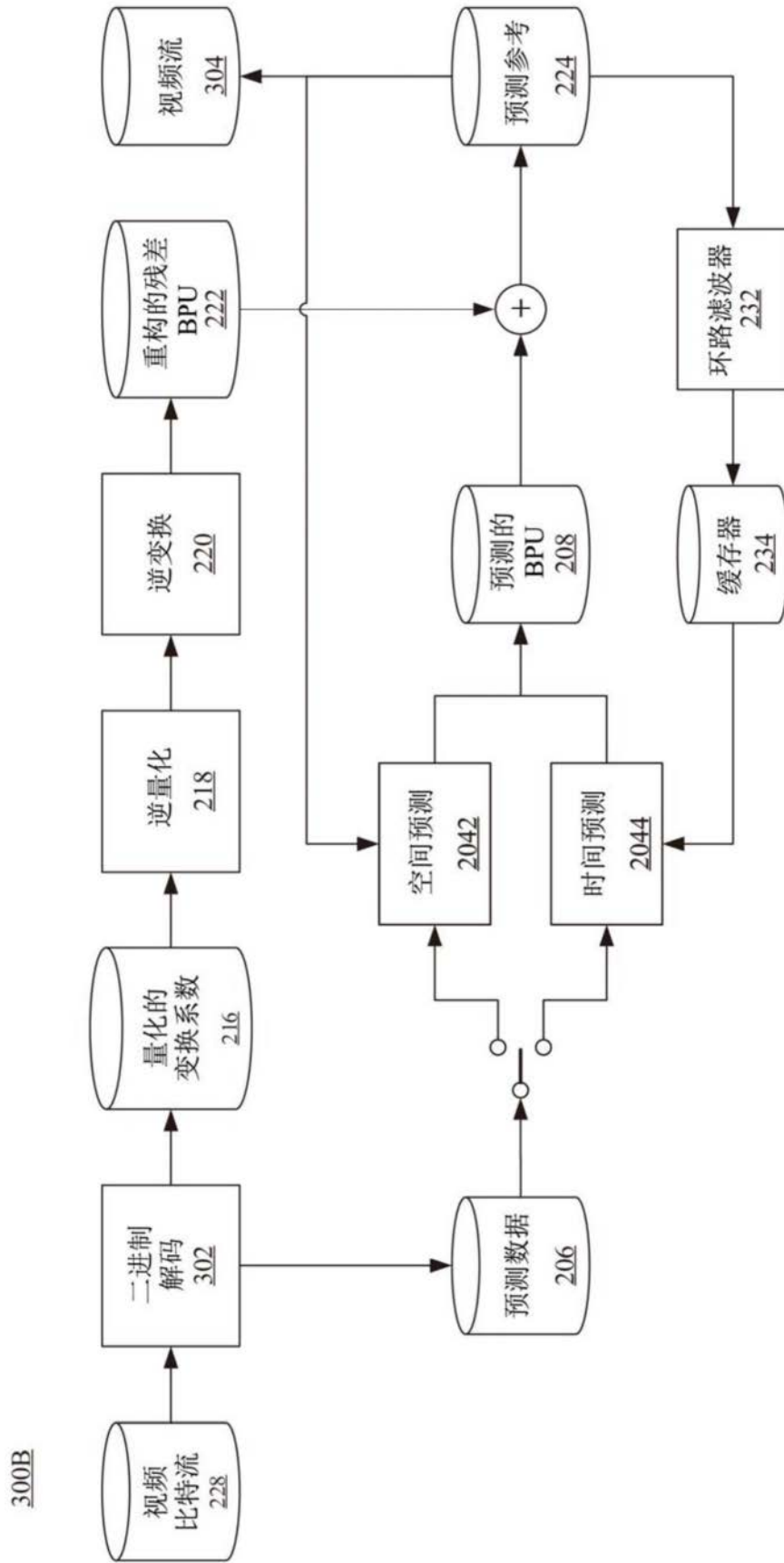


图3B

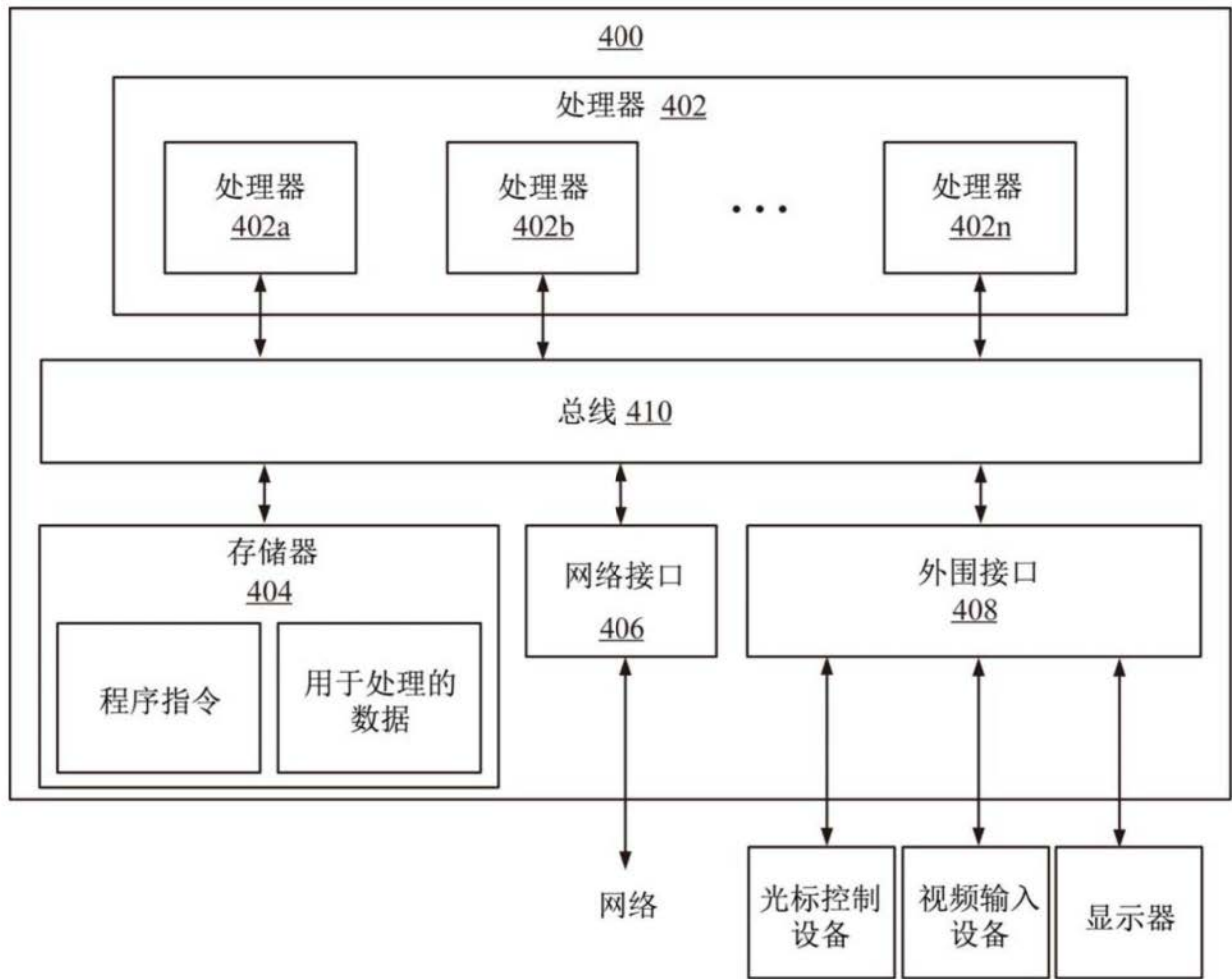


图4

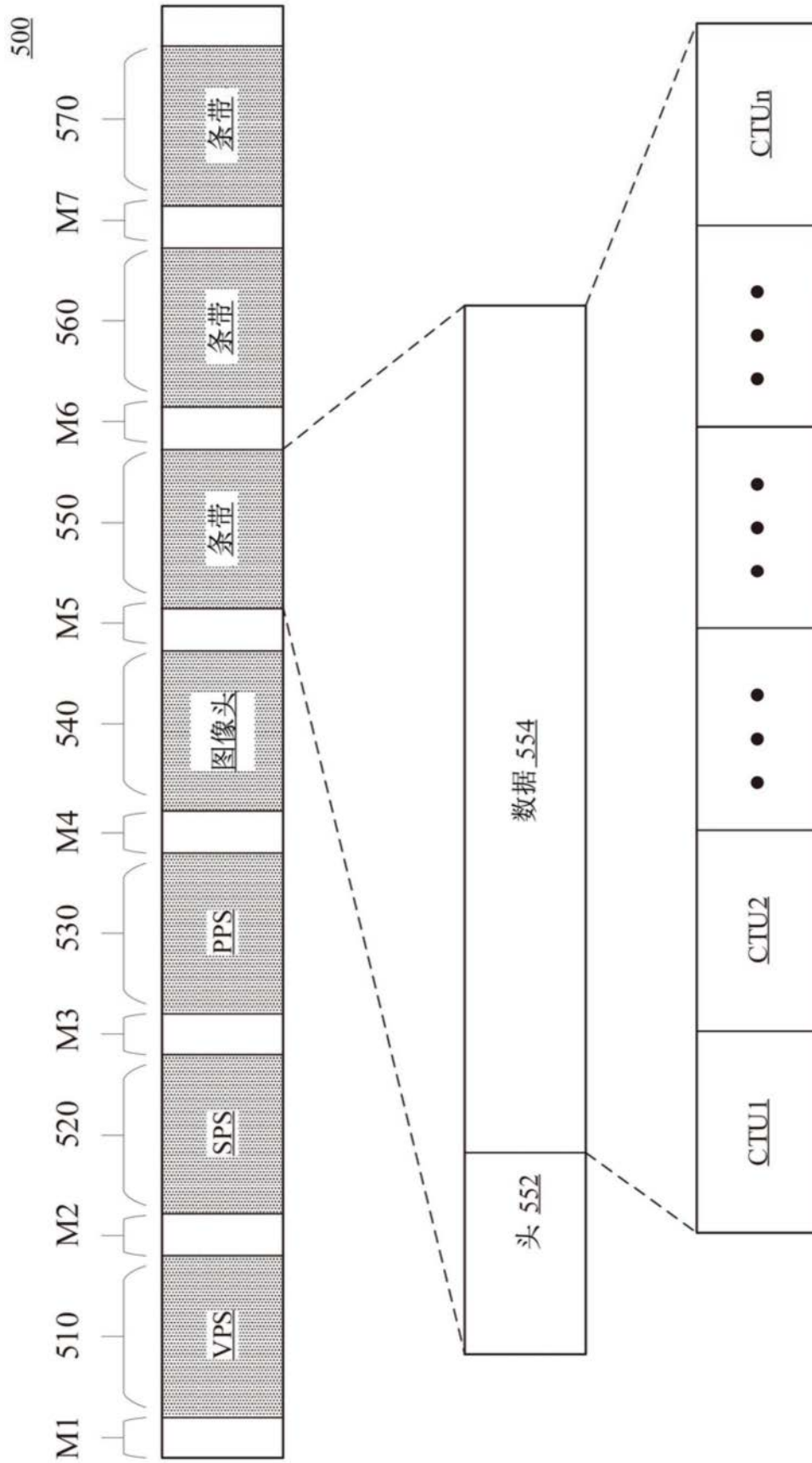


图5

600

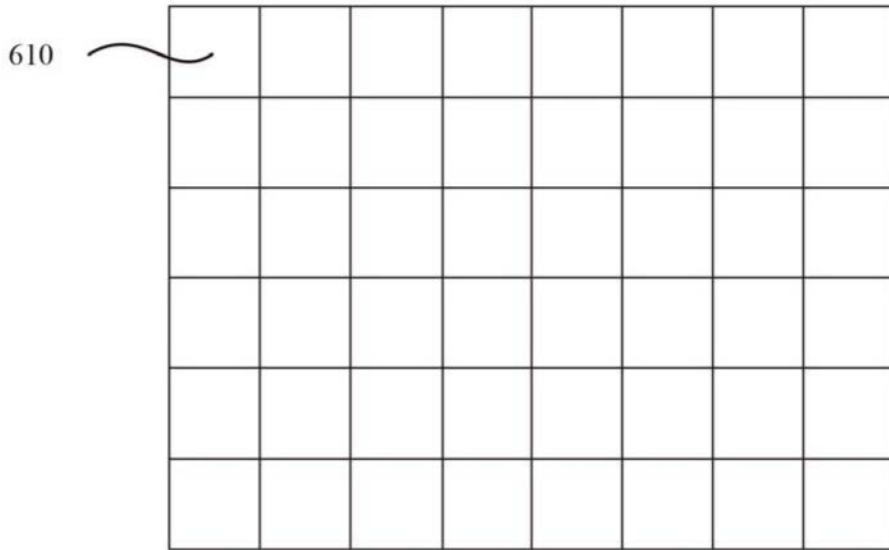


图6

700

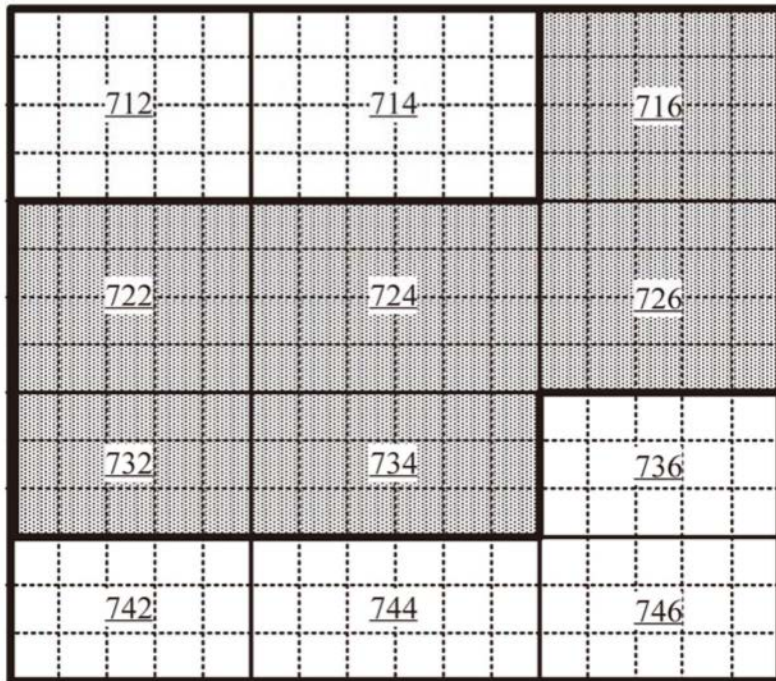


图7

800

<u>812</u>	<u>814</u>	<u>816</u>	<u>818</u>	<u>819</u>
<u>822</u>	<u>824</u>	<u>826</u>	<u>828</u>	<u>829</u>
<u>832</u>	<u>834</u>	<u>836</u>	<u>838</u>	<u>839</u>
<u>842</u>	<u>844</u>	<u>846</u>	<u>848</u>	<u>849</u>

图8

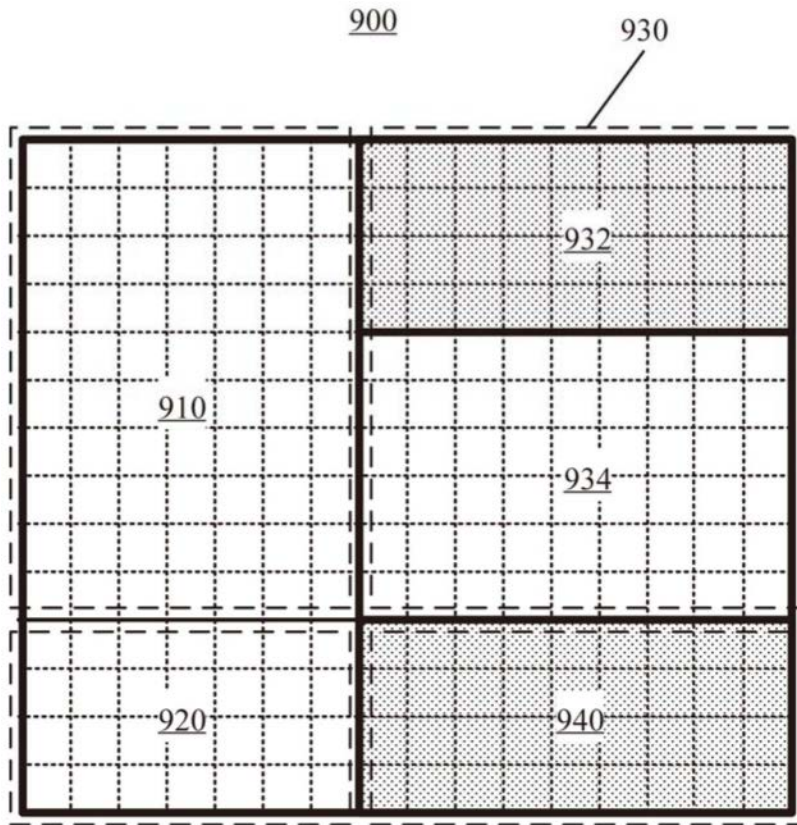


图9

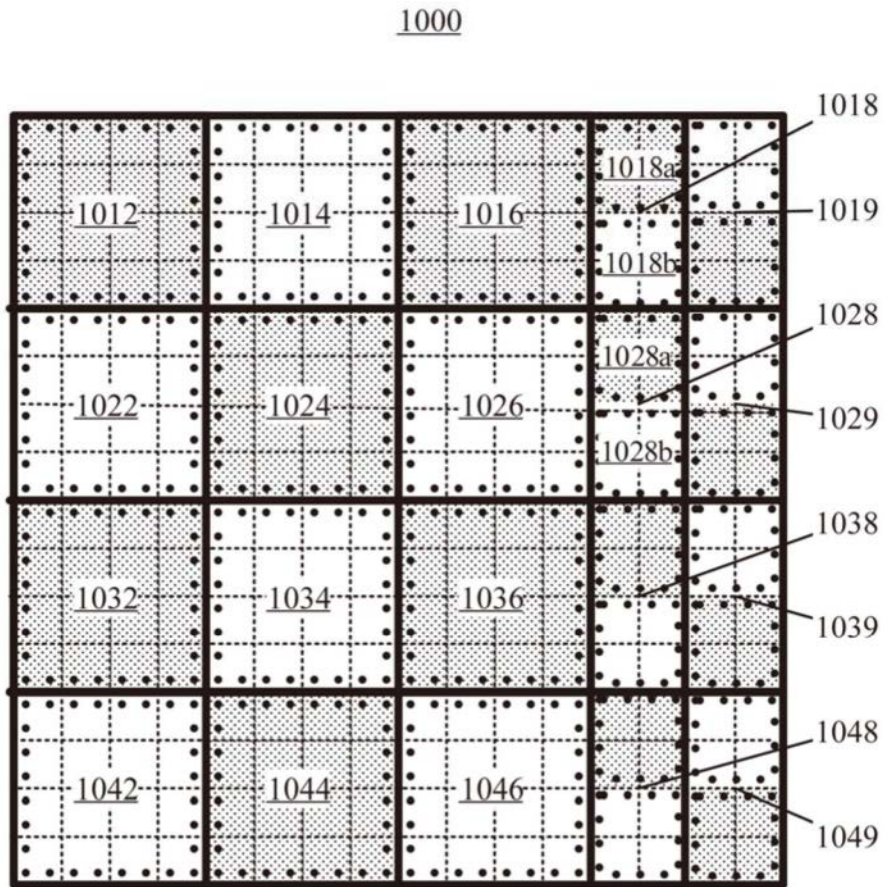


图10

	seq_parameter_set_rbsp() {	描述符
	...	u(2)
1110	sps_subpic_info_present_flag	u(1)
	if(sps_subpic_info_present_flag) {	
1112	sps_num_subpics_minus1	ue(v)
	if(sps_num_subpics_minus1 > 0)	
1120	sps_independent_subpics_flag	u(1)
	for(i = 0; sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++) {	
	if(i > 0 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1122	sps_subpic_ctu_top_left_x[i]	u(v)
	if(i > 0 && sps_pic_height_max_in_luma_samples > CtbSizeY) {	
1124	sps_subpic_ctu_top_left_y[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1126	sps_subpic_width_minus1[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_height_max_in_luma_samples > CtbSizeY)	
1128	sps_subpic_height_minus1[i]	u(v)
	if(!sps_independent_subpics_flag) {	
1130	sps_subpic_treated_as_pic_flag[i]	u(1)
1140	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	
	}	
1142	sps_subpic_id_len_minus1	ue(v)
1150	sps_subpic_id_mapping_explicitly_signalled_flag	u(1)
	if(sps_subpic_id_mapping_explicitly_signalled_flag) {	
1160	sps_subpic_id_mapping_present_flag	u(1)
	if(sps_subpic_id_mapping_present_flag)	
	for(i = 0; i <= sps_num_subpics_minus1; i++)	
1162	sps_subpic_id[i]	u(v)
	}	
	}	
	}	
	...	ue(v)

图11

1200

	pic_parameter_set_rbsp() {	描述符
	...	u(1)
1210	pps_no_pic_partition_flag	u(1)
1220	pps_subpic_id_mapping_present_flag	u(1)
	if(pps_subpic_id_mapping_present_flag) {	
	if(!pps_no_pic_partition_flag)	
1222	pps_num_subpics_minus1	ue(v)
1224	pps_subpic_id_len_minus1	ue(v)
	for(i = 0; i <= pps_num_subpic_minus1; i++)	
1226	pps_subpic_id[i]	u(v)
	}	
	if(!pps_no_pic_partition_flag) {	
1228	pps_log2_ctu_size_minus5	u(2)
1232	pps_num_exp_tile_columns_minus1	ue(v)
1234	pps_num_exp_tile_rows_minus1	ue(v)
	for(i = 0; i <= pps_num_exp_tile_columns_minus1; i++)	
1236	pps_tile_column_width_minus1[i]	ue(v)
	for(i = 0; i <= pps_num_exp_tile_rows_minus1; i++)	
1238	pps_tile_row_height_minus1[i]	ue(v)
	if(NumTilesInPic > 1) {	
1230	pps_loop_filter_across_tiles_enabled_flag	u(1)
1240	pps_rect_slice_flag	u(1)
	}	
	if(pps_rect_slice_flag)	
1250	pps_single_slice_per_subpic_flag	u(1)
	if(pps_rect_slice_flag && !pps_single_slice_per_subpic_flag) {	
1252	pps_num_slices_in_pic_minus1	ue(v)
	if(pps_num_slices_in_pic_minus1 > 1)	
1260	pps_tile_idx_delta_present_flag	u(1)
	for(i = 0; i < pps_num_slices_in_pic_minus1; i++) {	
	if(SliceTopLeftTileIdx[i] % NumTileColumns != NumTileColumns - 1)	
1262	pps_slice_width_in_tiles_minus1[i]	ue(v)

图12

	if(SliceTopLeftTileIdx[i] / Num TileColumns != NumTileRows - 1 && (pps_tile_idx_delta_present_flag SliceTopLeftTileIdx[i] % Num TileColumns == 0))	
1264	pps_slice_height_in_tiles_minus1[i]	ue(v)
	if(pps_slice_width_in_tiles_minus1[i] == 0 && pps_slice_height_in_tiles_minus1[i] == 0 && RowHeight[SliceTopLeftTileIdx[i] / Num TileColumns] > 1) {	
1266	pps_num_exp_slices_in_tile[i]	ue(v)
	for(j = 0; j < pps_num_exp_slices_in_tile[i]; j++)	
1268	pps_exp_slice_height_in_ctus_minus1[i][j]	ue(v)
	i += NumSlicesInTile[i] - 1	
	}	
	if(pps_tile_idx_delta_present_flag && i < pps_num_slices_in_pic_minus1)	
1272	pps_tile_idx_delta_val[i]	se(v)
	}	
	}	
	if(!pps_rect_slice_flag pps_single_slice_per_subpic_flag pps_num_slices_in_pic_minus1 > 0)	
1270	pps_loop_filter_across_slices_enabled_flag	u(1)
	}	
	...	u(1)

图12(续)

```

for( i = 0; i <= sps_num_subpics_minus1; i++ )
    if( sps_subpic_id_mapping_explicitly_signalled_flag )
        SubpicIdVal[ i ] = pps_subpic_id_mapping_present_flag ? pps_subpic_id[ i ] : sps_subpic_id[ i ]
    (81)
    else
        SubpicIdVal[ i ] = i
    
```

图13


```

remainingWidthInCtbsY = PicWidthInCtbsY
for( i = 0; i <= pps_num_exp_tile_columns_minus1; i++ ) {
    colWidth[ i ] = pps_tile_column_width_minus1[ i ] + 1
    remainingWidthInCtbsY -= colWidth[ i ]
}
uniformTileColWidth = pps_tile_column_width_minus1[ pps_num_exp_tile_columns_minus1 ] + 1
while( remainingWidthInCtbsY >= uniformTileColWidth ) {
    colWidth[ i++ ] = uniformTileColWidth
    remainingWidthInCtbsY -= uniformTileColWidth
}
if( remainingWidthInCtbsY > 0 )
    colWidth[ i++ ] = remainingWidthInCtbsY
NumTileColumns = i

```

图14

```

remainingHeightInCtbsY = PicHeightInCtbsY
for( j = 0; j <= pps_num_exp_tile_rows_minus1; j++ ) {
    RowHeight[ j ] = pps_tile_row_height_minus1[ j ] + 1
    remainingHeightInCtbsY -= RowHeight[ j ]
}
uniformTileRowHeight = pps_tile_row_height_minus1[ pps_num_exp_tile_rows_minus1 ] + 1
while( remainingHeightInCtbsY >= uniformTileRowHeight ) {
    RowHeight[ j++ ] = uniformTileRowHeight
    remainingHeightInCtbsY -= uniformTileRowHeight
}
if( remainingHeightInCtbsY > 0 )
    RowHeight[ j++ ] = remainingHeightInCtbsY
NumTileRows = j

```

图15

```

for( tileColBd[ 0 ] = 0, i = 0; i < NumTileColumns; i++ )
    tileColBd[ i + 1 ] = tileColBd[ i ] + colWidth[ i ]

```

图16

```

for( tileRowBd[ 0 ] = 0, j = 0; j < NumTileRows; j++ )
    tileRowBd[ j + 1 ] = tileRowBd[ j ] + RowHeight[ j ]

```

图17

```

tileX = 0
for( ctbAddrX = 0; ctbAddrX <= PicWidthInCtbsY; ctbAddrX++) {
    if( ctbAddrX == tileColBd[ tileX + 1 ] )
        tileX++
    CtbToTileColBd[ ctbAddrX ] = tileColBd[ tileX ]
    ctbToTileColIdx[ ctbAddrX ] = tileX
}

```

图18

```

tileY = 0
for( ctbAddrY = 0; ctbAddrY <= PicHeightInCtbsY; ctbAddrY++) {
    if( ctbAddrY == tileRowBd[ tileY + 1 ] )
        tileY++
    CtbToTileRowBd[ ctbAddrY ] = tileRowBd[ tileY ]
    ctbToTileRowIdx[ ctbAddrY ] = tileY
}

```

图19

```

for( i = 0; i <= sps_num_subpics_minus1; i++) {
    leftX = sps_subpic_ctu_top_left_x[ i ]
    rightX = leftX + sps_subpic_width_minus1[ i ]
    SubpicWidthInTiles[ i ] = ctbToTileColIdx[ rightX ] + 1 - ctbToTileColIdx[ leftX ]
    topY = sps_subpic_ctu_top_left_y[ i ]
    bottomY = topY + sps_subpic_height_minus1[ i ]
    SubpicHeightInTiles[ i ] = ctbToTileRowIdx[ bottomY ] + 1 - ctbToTileRowIdx[ topY ]
    if( SubpicHeightInTiles[ i ] == 1 &&
        sps_subpic_height_minus1[ i ] + 1 < RowHeight[ ctbToTileRowIdx[ topY ] ])
        subpicHeightLessThanOneTileFlag[ i ] = 1
    else
        subpicHeightLessThanOneTileFlag[ i ] = 0
}

```

图20

```

if( pps_single_slice_per_subpic_flag ) {
    if( !sps_subpic_info_present_flag ) /* 图像中没有子图像信息，且只有一个条带。*/
        for( j = 0; j < NumTileRows; j++ )
            for( i = 0; i < NumTileColumns; i++ )
                AddCtbsToSlice( 0, tileColBd[ i ], tileColBd[ i + 1 ], tileRowBd[ j ],
tileRowBd[ j + 1 ] )
    else {
        for( i = 0; i <= sps_num_subpics_minus1; i++ ) {
            NumCtbsInSlice[ i ] = 0
            if( subpicHeightLessThanOneTileFlag[ i ] ) /* 条带由砖块中的一组CTU行组成。*/

                AddCtbsToSlice( i, sps_subpic_ctu_top_left_x[ i ],
                    sps_subpic_ctu_top_left_x[ i ] + sps_subpic_width_minus1[ i ] + 1,
                    sps_subpic_ctu_top_left_y[ i ],
                    sps_subpic_ctu_top_left_y[ i ] + sps_subpic_height_minus1[ i ] + 1 )
            else /* 条带由数个覆盖矩形区域的完整砖块组成。*/

                tileX = CtbToTileColBd[ sps_subpic_ctu_top_left_x[ i ] ]
                tileY = CtbToTileRowBd[ sps_subpic_ctu_top_left_y[ i ] ]
                for( j = 0; j < SubpicHeightInTiles[ i ]; j++ )
                    for( k = 0; k < SubpicWidthInTiles[ i ]; k++ )
                        AddCtbsToSlice( i, tileColBd[ tileX + k ],
tileColBd[ tileX + k + 1 ],
                                tileRowBd[ tileY + j ],
tileRowBd[ tileY + j + 1 ] )
        }
    }
} else {
    tileIdx = 0
    for( i = 0; i <= pps_num_slices_in_pic_minus1; i++ )
        NumCtbsInSlice[ i ] = 0
    for( i = 0; i <= pps_num_slices_in_pic_minus1; i++ ) {
        SliceTopLeftTileIdx[ i ] = tileIdx
        tileX = tileIdx % NumTileColumns
        tileY = tileIdx / NumTileColumns
        if( i < pps_num_slices_in_pic_minus1 ) {
            sliceWidthInTiles[ i ] = pps_slice_width_in_tiles_minus1[ i ] + 1
            sliceHeightInTiles[ i ] = pps_slice_height_in_tiles_minus1[ i ] + 1
        } else {
            sliceWidthInTiles[ i ] = NumTileColumns - tileX
            sliceHeightInTiles[ i ] = NumTileRows - tileY
            NumSlicesInTile[ i ] = 1
        }
    }
}

```

图21

```

        for(j = 0; j < pps_num_exp_slices_in_tile[i]; j++) {
            sliceHeightInCtus[i + j] = pps_exp_slice_height_in_ctus_minus1[i][j] + 1
            remainingHeightInCtbsY -= sliceHeightInCtus[i + j]
        }
        uniformSliceHeight = sliceHeightInCtus[i + j - 1]
        while(remainingHeightInCtbsY >= uniformSliceHeight) {
            sliceHeightInCtus[i + j] = uniformSliceHeight
            remainingHeightInCtbsY -= uniformSliceHeight
            j++
        }
        if(remainingHeightInCtbsY > 0) {
            sliceHeightInCtus[i + j] = remainingHeightInCtbsY
            j++
        }
        NumSlicesInTile[i] = j
    }
    ctbY = tileRowBd[ tileY ]
    for(j = 0; j < NumSlicesInTile[i]; j++) {
        AddCtbsToSlice(i + j, tileColBd[ tileX ], tileColBd[ tileX + 1 ],
            ctbY, ctbY + sliceHeightInCtus[i + j])
        ctbY += sliceHeightInCtus[i + j]
    }
    i += NumSlicesInTile[i] - 1
} else
for(j = 0; j < sliceHeightInTiles[i]; j++)
for(k = 0; k < sliceWidthInTiles[i]; k++)
    AddCtbsToSlice(i, tileColBd[ tileX + k ], tileColBd[ tileX + k + 1 ],
        tileRowBd[ tileY + j ], tileRowBd[ tileY + j + 1 ])
if(i < pps_num_slices_in_pic_minus1) {
    if(pps_tile_idx_delta_present_flag)
        tileIdx += pps_tile_idx_delta_val[i]
    else {
        tileIdx += sliceWidthInTiles[i]
        if(tileIdx % NumTileColumns == 0)
            tileIdx += (sliceHeightInTiles[i] - 1) * NumTileColumns
    }
}
}
}
}
}

```

图21 (续)

```

for( ctbY = startY; ctbY < stopY; ctbY++)
    for( ctbX = startX; ctbX < stopX; ctbX++) {
        CtbAddrInSlice[ sliceIdx ][ NumCtusInSlice[ sliceIdx ] ] = ctbY * PicWidthInCtbsY + ctbX
        NumCtusInSlice[ sliceIdx ]++
    }
}

```

图22

```
for( i = 0; i <= sps_num_subpics_minus1; i++) {  
    NumSlicesInSubpic[ i ] = 0  
    for( j = 0; j <= pps_num_slices_in_pic_minus1; j++) {  
        posX = CtbAddrInSlice[ j ][ 0 ] % PicWidthInCtbsY  
        posY = CtbAddrInSlice[ j ][ 0 ] / PicWidthInCtbsY  
        if( ( posX >= sps_subpic_ctu_top_left_x[ i ] ) &&  
            ( posX < sps_subpic_ctu_top_left_x[ i ] + sps_subpic_width_minus1[ i ] + 1 ) &&  
            ( posY >= sps_subpic_ctu_top_left_y[ i ] ) &&  
            ( posY < sps_subpic_ctu_top_left_y[ i ] + sps_subpic_height_minus1[ i ] + 1 ) ) {  
            SubpicIdxForSlice[ j ] = i  
            SubpicLevelSliceIdx[ j ] = NumSlicesInSubpic[ i ]  
            NumSlicesInSubpic[ i ]++  
        }  
    }  
}
```

图23

	seq_parameter_set_rbsp() {	描述符
	...	u(2)
2410	sps_no_pic_partition_flag	u(1)
1110	sps_subpic_info_present_flag	u(1)
	if(sps_subpic_info_present_flag) {	
1112	sps_num_subpics_minus1	ue(v)
	if(sps_num_subpics_minus1 > 0)	
1120	sps_independent_subpics_flag	u(1)
	for(i = 0; sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++) {	
	if(i > 0 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1122	sps_subpic_ctu_top_left_x[i]	u(v)
	if(i > 0 && sps_pic_height_max_in_luma_samples > CtbSizeY) {	
1124	sps_subpic_ctu_top_left_y[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1126	sps_subpic_width_minus1[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_height_max_in_luma_samples > CtbSizeY)	
1128	sps_subpic_height_minus1[i]	u(v)
	if(!sps_independent_subpics_flag) {	
1130	sps_subpic_treated_as_pic_flag[i]	u(1)
1140	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	
	}	
	}	
	...	ue(v)

图24

2500

	seq_parameter_set_rbsp() {	描述符
	...	u(2)
2410	sps_no_pic_partition_flag	u(1)
	if(!sps_no_pic_partition_flag)	
1110	sps_subpic_info_present_flag	u(1)
	if(sps_subpic_info_present_flag) {	
1112	sps_num_subpics_minus1	ue(v)
	if(sps_num_subpics_minus1 > 0)	
1120	sps_independent_subpics_flag	u(1)
	for(i = 0; sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++) {	
	if(i > 0 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1122	sps_subpic_ctu_top_left_x[i]	u(v)
	if(i > 0 && sps_pic_height_max_in_luma_samples > CtbSizeY) {	
1124	sps_subpic_ctu_top_left_y[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1126	sps_subpic_width_minus1[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_height_max_in_luma_samples > CtbSizeY)	
1128	sps_subpic_height_minus1[i]	u(v)
	if(!sps_independent_subpics_flag) {	
1130	sps_subpic_treated_as_pic_flag[i]	u(1)
1140	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	
	...	ue(v)

图25

	seq_parameter_set_rbsp() {	描述符
	...	u(2)
2410	sps_no_pic_partition_flag	u(1)
1110	sps_subpic_info_present_flag	u(1)
	if(sps_subpic_info_present_flag) {	
	if(!sps_no_pic_partition_flag)	
1112	sps_num_subpics_minus1	ue(v)
	if(sps_num_subpics_minus1 > 0)	
1120	sps_independent_subpics_flag	u(1)
	for(i = 0; sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++) {	
	if(i > 0 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1122	sps_subpic_ctu_top_left_x[i]	u(v)
	if(i > 0 && sps_pic_height_max_in_luma_samples > CtbSizeY) {	
1124	sps_subpic_ctu_top_left_y[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
1126	sps_subpic_width_minus1[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_height_max_in_luma_samples > CtbSizeY)	
1128	sps_subpic_height_minus1[i]	u(v)
	if(!sps_independent_subpics_flag) {	
1130	sps_subpic_treated_as_pic_flag[i]	u(1)
1140	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	
	...	ue(v)

图26

2700A

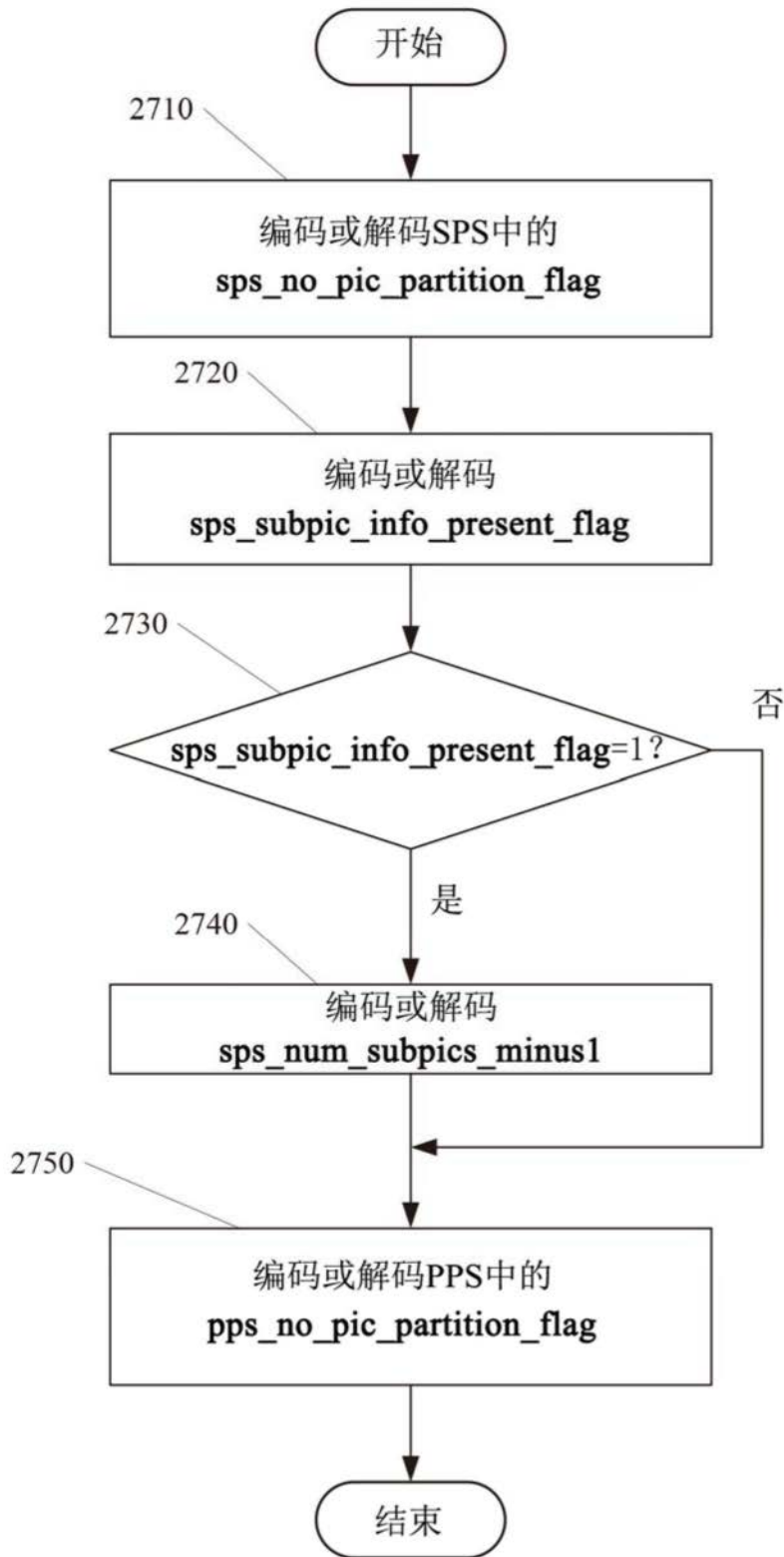


图27A

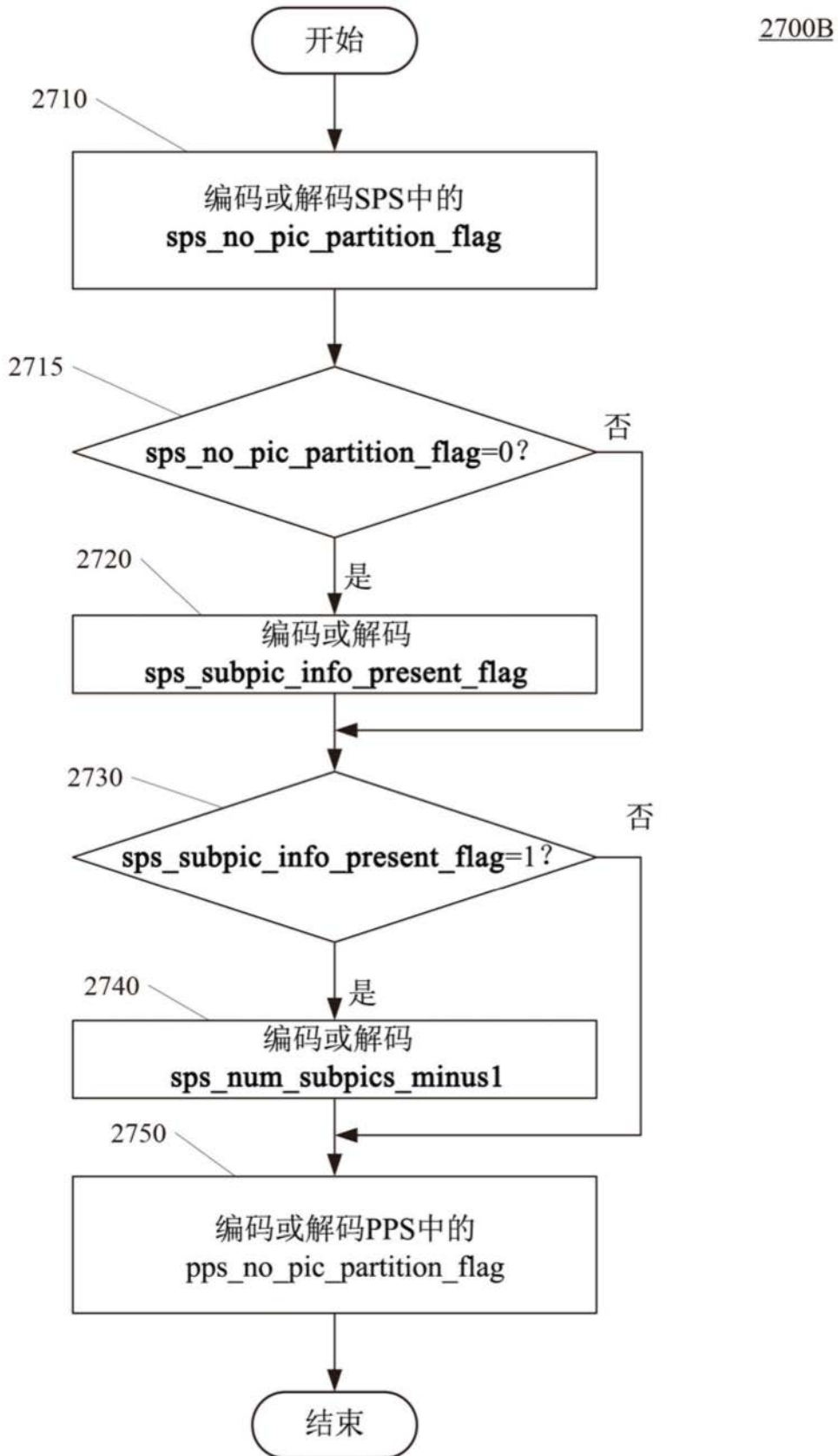


图27B

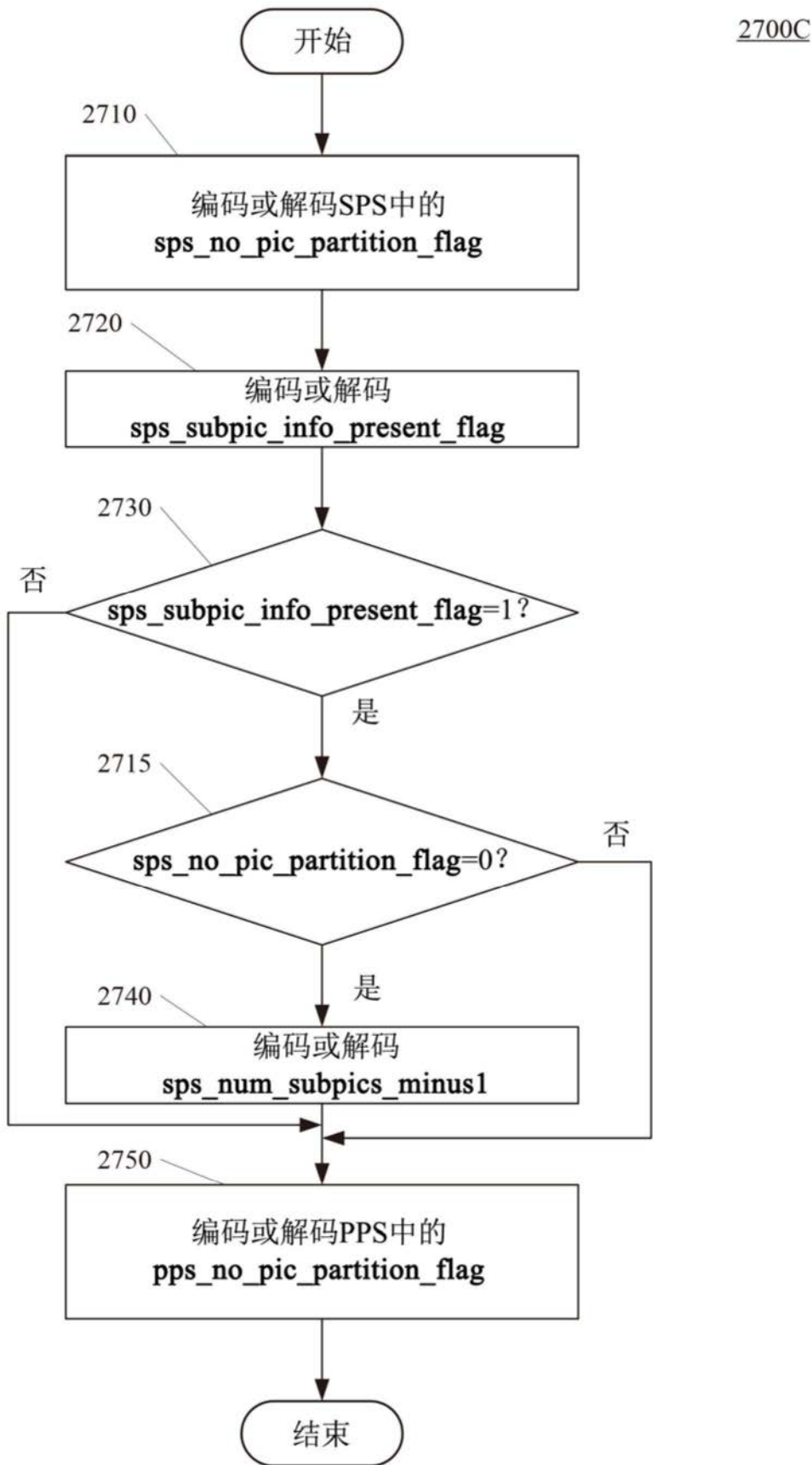


图27C

	pic_parameter_set_rbsp() {	描述符
	...	u(1)
1210	pps_no_pic_partition_flag	u(1)
	...	u(1)
	if(!pps_no_pic_partition_flag) {	
1228	pps_log2_ctu_size_minus5	u(2)
1232	pps_num_exp_tile_columns_minus1	ue(v)
1234	pps_num_exp_tile_rows_minus1	ue(v)
	for(i = 0; i <= pps_num_exp_tile_columns_minus1; i++)	
1236	pps_tile_column_width_minus1[i]	ue(v)
	for(i = 0; i <= pps_num_exp_tile_rows_minus1; i++)	
1238	pps_tile_row_height_minus1[i]	ue(v)
	if(NumTilesInPic > 1) {	
1230	pps_loop_filter_across_tiles_enabled_flag	u(1)
	}	
	if(NumTilesInPic > 3){	
1240	pps_rect_slice_flag	u(1)
	}	
	...	
	...	u(1)

图28

2900

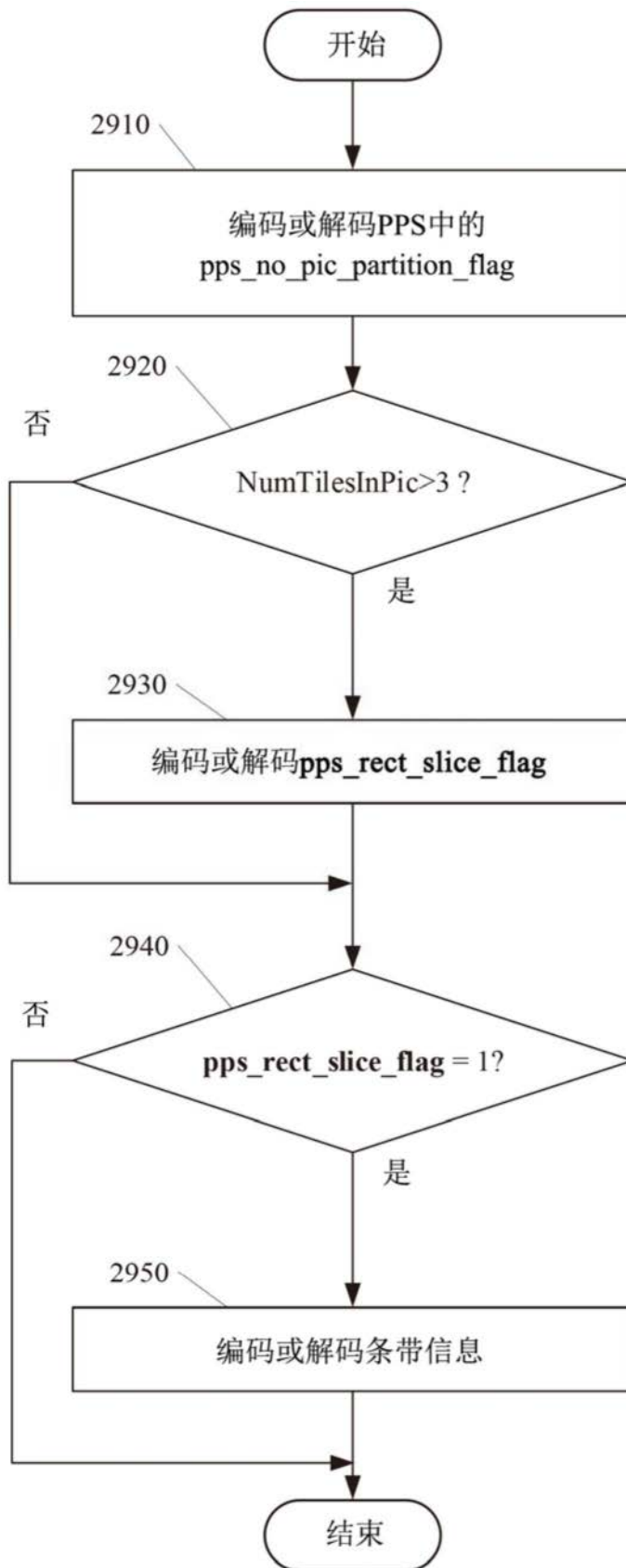


图29

3000

	pic_parameter_set_rbsp() {	描述符
	...	u(1)
	if(pps_rect_slice_flag)	
1250	pps_single_slice_per_subpic_flag	u(1)
	if(pps_rect_slice_flag && !pps_single_slice_per_subpic_flag) {	
1252	pps_num_slices_in_pic_minus1	ue(v)
	if(pps_num_slices_in_pic_minus1 > 1)	
1260	pps_tile_idx_delta_present_flag	u(1)
	for(i = 0; i < pps_num_slices_in_pic_minus1; i++) {	
	if(SliceTopLeftTileIdx[i] % NumTileColumns != NumTileColumns - 1)	
1262	pps_slice_width_in_tiles_minus1[i]	ue(v)
	if(SliceTopLeftTileIdx[i] / NumTileColumns != NumTileRows - 1 && (pps_tile_idx_delta_present_flag SliceTopLeftTileIdx[i] % NumTileColumns == 0))	
1264	pps_slice_height_in_tiles_minus1[i]	ue(v)
	if(pps_slice_width_in_tiles_minus1[i] == 0 && pps_slice_height_in_tiles_minus1[i] == 0 && RowHeight[SliceTopLeftTileIdx[i] / NumTileColumns] > 1) {	
3066	pps_num_exp_slices_in_tile[i] pps_num_exp_slices_in_tile_minus1[i]	ue(v)
	for(j = 0; j < pps_num_exp_slices_in_tile[i] <=pps_num_exp_slices_in_tile_minus1[i]; j++)	
1268	pps_exp_slice_height_in_ctus_minus1[i][j]	ue(v)
	i += NumSlicesInTile[i] - 1	
	}	
	if(pps_tile_idx_delta_present_flag && i < pps_num_slices_in_pic_minus1)	
1272	pps_tile_idx_delta_val[i]	se(v)
	}	
	}	
	if(!pps_rect_slice_flag pps_single_slice_per_subpic_flag pps_num_slices_in_pic_minus1 > 0)	
	pps_loop_filter_across_slices_enabled_flag	u(1)
1270	}	
	...	u(1)

图30

2950

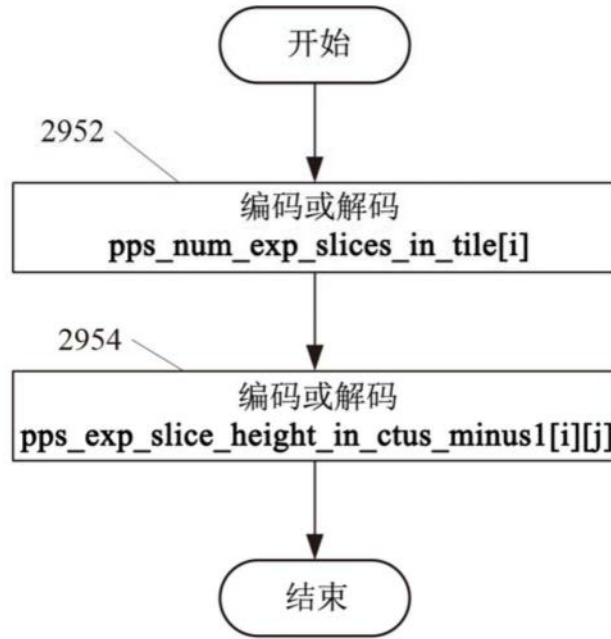


图31A

2950

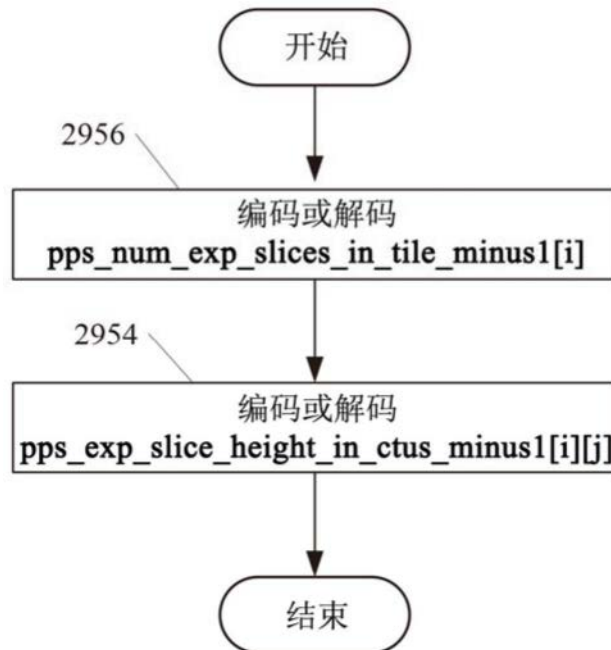


图31B

3200

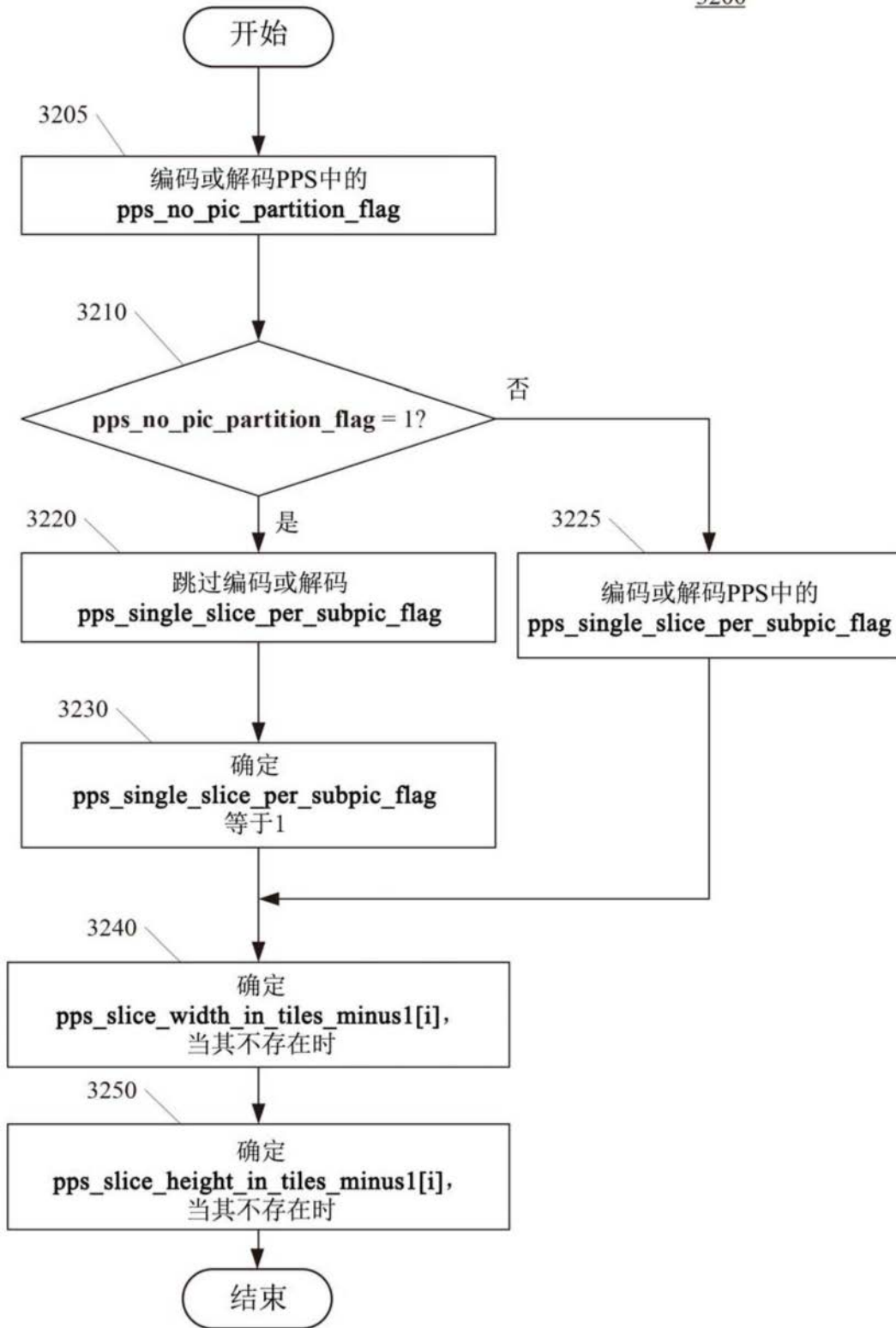


图32


```
if( sliceWidthInTiles[ i ] == 1 && sliceHeightInTiles[ i ] == 1 ) {
    if( pps_num_exp_slices_in_tile[ i ] == 0 ) {
        NumSlicesInTile[ i ] = 1
        sliceHeightInCtus[ i ] = RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
    } else {
        remainingHeightInCtbsY = RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
        for( j = 0; j <= pps_num_exp_slices_in_tile_minus1[ i ]; j++ ) {
            sliceHeightInCtus[ i + j ] = pps_exp_slice_height_in_ctus_minus1[ i ][ j ] + 1
            remainingHeightInCtbsY -= sliceHeightInCtus[ i + j ]
        }
        uniformSliceHeight = sliceHeightInCtus[ i + j - 1 ]
        while( remainingHeightInCtbsY >= uniformSliceHeight ) {
            sliceHeightInCtus[ i + j ] = uniformSliceHeight
            remainingHeightInCtbsY -= uniformSliceHeight
            j++
        }
        if( remainingHeightInCtbsY > 0 ) {
            sliceHeightInCtus[ i + j ] = remainingHeightInCtbsY
            j++
        }
        NumSlicesInTile[ i ] = j
    }
}
```

图33

```

if( pps_single_slice_per_subpic_flag ) {
    if( !sps_subpic_info_present_flag ) /* 图像中没有子图像信息，且只有一个条带。*/
        for( j = 0; j < NumTileRows; j++ )
            for( i = 0; i < NumTileColumns; i++ )
                AddCtbsToSlice( 0, tileColBd[ i ], tileColBd[ i + 1 ], tileRowBd[ j ],
tileRowBd[ j + 1 ] )
    else {
        for( i = 0; i <= sps_num_subpics_minus1; i++ ) {
            NumCtbsInSlice[ i ] = 0
            if( subpicHeightLessThanOneTileFlag[ i ] ) /* 条带由砖块中的一组CTU
行组成。*/
                AddCtbsToSlice( i, sps_subpic_ctu_top_left_x[ i ],

                sps_subpic_ctu_top_left_x[ i ] + sps_subpic_width_minus1[ i ] + 1,
                sps_subpic_ctu_top_left_y[ i ],

                sps_subpic_ctu_top_left_y[ i ] + sps_subpic_height_minus1[ i ] + 1 )
            else { /* 条带由数个覆盖矩形区域的完整砖块组成。*/

                tileX =
CtbsToTileColBdCtbsToTileColBdIdx[ sps_subpic_ctu_top_left_x[ i ] ]
                tileY = CtbsToTileRowBdCtbsToTileRowBdIdx
[ sps_subpic_ctu_top_left_y[ i ] ]
                for( j = 0; j < SubpicHeightInTiles[ i ]; j++ )
                    for( k = 0; k < SubpicWidthInTiles[ i ]; k++ )
                        AddCtbsToSlice( i, tileColBd[ tileX + k ],
tileColBd[ tileX + k + 1 ],
                                tileRowBd[ tileY + j ], tileRowBd[ tileY + j + 1 ] )
                    }
                }
            }
        } else {
            tileIdx = 0
            for( i = 0; i <= pps_num_slices_in_pic_minus1; i++ )
                NumCtbsInSlice[ i ] = 0
            for( i = 0; i <= pps_num_slices_in_pic_minus1; i++ ) {
                SliceTopLeftTileIdx[ i ] = tileIdx
                tileX = tileIdx % NumTileColumns
                tileY = tileIdx / NumTileColumns
            }
        }
    }
}

```

图34

```

if( i < pps_num_slices_in_pic_minus1 ) {
    sliceWidthInTiles[ i ] = pps_slice_width_in_tiles_minus1[ i ] + 1
    sliceHeightInTiles[ i ] = pps_slice_height_in_tiles_minus1[ i ] + 1
} else {
    sliceWidthInTiles[ i ] = NumTileColumns - tileX
    sliceHeightInTiles[ i ] = NumTileRows - tileY
    NumSlicesInTile[ i ] = 1
}
if( sliceWidthInTiles[ i ] == 1 && sliceHeightInTiles[ i ] == 1 ) {
    if( pps_num_exp_slices_in_tile[ i ] == 0 ) {
        NumSlicesInTile[ i ] = 1
        sliceHeightInCtus[ i ] =
RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
    } else {
        remainingHeightInCtbsY =
RowHeight[ SliceTopLeftTileIdx[ i ] / NumTileColumns ]
        for( j = 0; j < pps_num_exp_slices_in_tile[ i ]; j++ ) {
            sliceHeightInCtus[ i + j ] = pps_exp_slice_height_in_ctus_minus1[ i ][ j ] +
1
            remainingHeightInCtbsY -= sliceHeightInCtus[ i + j ]
        }
        uniformSliceHeight = sliceHeightInCtus[ i + j - 1 ]
        while( remainingHeightInCtbsY >= uniformSliceHeight ) {
            sliceHeightInCtus[ i + j ] = uniformSliceHeight
            remainingHeightInCtbsY -= uniformSliceHeight
            j++
        }
        if( remainingHeightInCtbsY > 0 ) {
            sliceHeightInCtus[ i + j ] = remainingHeightInCtbsY
            j++
        }
        NumSlicesInTile[ i ] = j
    }
}
ctbY = tileRowBd[ tileY ]
for( j = 0; j < NumSlicesInTile[ i ]; j++ ) {

```

图34(续)

```
        AddCtbsToSlice( i + j, tileColBd[ tileX ], tileColBd[ tileX + 1 ],
                       ctbY, ctbY + sliceHeightInCtus[ i + j ] )
        ctbY += sliceHeightInCtus[ i + j ]
    }
    i += NumSlicesInTile[ i ] - 1
} else
    for( j = 0; j < sliceHeightInTiles[ i ]; j++ )
        for( k = 0; k < sliceWidthInTiles[ i ]; k++ )
            AddCtbsToSlice( i, tileColBd[ tileX + k ],
                           tileColBd[ tileX + k + 1 ],
                           tileRowBd[ tileY + j ], tileRowBd[ tileY + j + 1 ] )
if( i < pps_num_slices_in_pic_minus1 ) {
    if( pps_tile_idx_delta_present_flag )
        tileIdx += pps_tile_idx_delta_val[ i ]
    else {
        tileIdx += sliceWidthInTiles[ i ]
        if( tileIdx % NumTileColumns == 0 )
            tileIdx += ( sliceHeightInTiles[ i ] - 1 ) * NumTileColumns
    }
}
}
}
```

图34 (续)