(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 111989928 B
(45)授权公告日 2022.12.27

(21)申请号 201980024703.9
(22)申请日 2019.03.25
(65)同一申请的已公布的文献号
申请公布号 CN 111989928 A
(43)申请公布日 2020.11.24
(30)优先权数据
1805886.7 2018.04.09 GB
(85)PCT国际申请进入国家阶段日
2020.10.09
(86)PCT国际申请的申请数据
PCT/EP2019/057432 2019.03.25
(87)PCT国际申请的公布数据
WO2019/197145 EN 2019.10.17

(73)专利权人 佳能株式会社
地址 日本东京都大田区下丸子3丁目30番2号
(72)发明人 乔纳森·泰奎特
内尔·奥德拉奥果
弗兰克·德诺奥
弗雷德里克·梅兹
(74)专利代理机构 北京魏启学律师事务所 11398
专利代理师 魏启学

(51)Int.Cl.
H04N 19/70(2006.01)
H04N 19/174(2006.01)
审查员 高宇腾

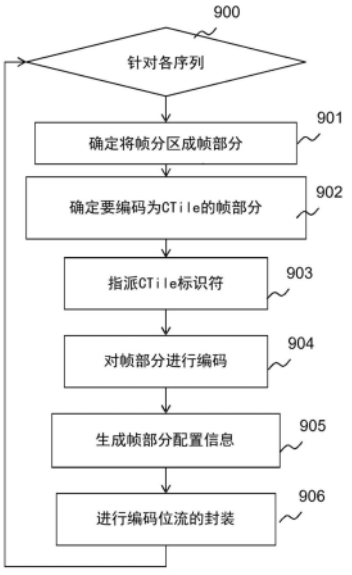
权利要求书2页 说明书24页 附图23页

(54)发明名称

对具有帧部分的视频数据进行编码或解码的方法和装置

(57)摘要

本发明涉及一种用于将包括帧的视频数据编码到位流中的方法,所述帧在空间上被划分为帧部分,所述方法包括:将至少一个帧部分编码到一个或多个第一编码单位中;其中,在所述方法中,还包括:在所述第一编码单位中表示至少一个帧部分标识符,所述帧部分标识符标识一个编码帧部分;以及提供帧部分配置信息,所述帧部分配置信息包括所述帧部分标识符和与所述帧部分有关的空间信息。



CN 111989928 B

1.一种用于将帧编码到位流中的方法,所述帧在空间上被划分为帧部分,所述方法包括:

-将所述帧中的帧部分编码到所述位流中;

-在所述位流中表示所述帧中的所述帧部分中的每一个帧部分的标识符;以及

-对与帧部分在所述帧内的位置有关的空间信息进行表示,其中所述标识符和所述空间信息是在所述位流中的一个参数集中被表示的,并且用于代表所述标识符的位数信息进一步在所述位流中被表示。

2.根据权利要求1所述的方法,其中,所述帧部分是被独立编码的。

3.根据权利要求2所述的方法,还包括:提供指示所述帧部分已被独立编码的标志。

4.根据权利要求1所述的方法,其中,所述参数集专用于与一个或多个帧有关的信息。

5.根据权利要求1所述的方法,其中,所述空间信息包括由编码树单位地址给出的所述帧部分的位置。

6.根据权利要求1所述的方法,其中,所述空间信息包括所述帧部分的大小。

7.根据权利要求1所述的方法,其中,所述帧部分的位置是关于所述帧而给出的。

8.根据权利要求1所述的方法,其中,所述参数集包括指示给定的后滤波算法是否能够用于所述帧部分的标志。

9.一种用于从位流解码帧的方法,所述帧在空间上被划分成帧部分,所述方法包括:

-从所述位流中的一个参数集获得所述帧中的所述帧部分中的每一个帧部分的标识符和与所述帧部分在所述帧内的位置有关的空间信息,其中要获得的所述标识符和所述空间信息是在所述位流中的所述一个参数集中被表示的,并且用于代表所述标识符的位数信息进一步在所述位流中被表示;

-基于所述空间信息来确定所述帧部分在所述帧内的位置;以及

-从所述位流解码所述帧部分。

10.根据权利要求9所述的方法,其中,所述帧部分是被独立编码的。

11.根据权利要求9所述的方法,还包括:获得指示所述帧部分已被独立编码的标志。

12.根据权利要求9所述的方法,其中,所述参数集专用于与一个或多个帧有关的信息。

13.根据权利要求9所述的方法,其中,所述空间信息包括由编码树单位地址给出的所述帧部分的位置。

14.根据权利要求9所述的方法,其中,所述空间信息包括所述帧部分的大小。

15.根据权利要求9所述的方法,其中,所述帧部分的位置是关于所述帧而给出的。

16.根据权利要求9所述的方法,其中,所述参数集包括指示给定的后滤波算法是否能够用于所述帧部分的标志。

17.一种用于将帧编码到位流中的装置,所述帧在空间上被划分成帧部分,所述装置包括被配置为进行以下操作的电路:

-将所述帧中的帧部分编码到所述位流中;

其中,所述电路进一步被配置为进行以下操作:

-在所述位流中表示所述帧中的所述帧部分中的每一个帧部分的标识符;以及

-对与帧部分在所述帧内的位置有关的空间信息进行表示,其中所述标识符和所述空间信息是在所述位流中的一个参数集中被表示的,并且用于代表所述标识符的位数信息进

一步在所述位流中被表示。

18.一种用于从位流中解码帧的装置,所述帧在空间上被划分成帧部分,所述装置包括被配置为进行以下操作的电路:

-从所述位流中的一个参数集获得所述帧中的所述帧部分中的每一个帧部分的标识符和与所述帧部分在所述帧内的位置有关的空间信息,其中要获得的所述标识符和所述空间信息是在所述位流中的所述一个参数集中被表示的,并且用于代表所述标识符的位数信息进一步在所述位流中被表示;

-基于所述空间信息来确定所述帧部分在所述帧内的位置;以及

-从所述位流解码所述帧部分。

19.一种存储有计算机程序的指令的计算机可读存储介质,所述指令用于使计算机实现根据权利要求1至16中任一项所述的方法。

对具有帧部分的视频数据进行编码或解码的方法和装置

技术领域

[0001] 本发明涉及用于对包括空间部分的视频数据进行编码或解码的方法和装置。

背景技术

[0002] 引入并设计了HEVC区块以进行并行编码。然而,对于大小大的视频内容,存在以不同的方式使用区块的数个使用情况。特别地,开发了用于流传输个体区块或区块集合的需求。一些应用还开发了组合来自同一序列或来自不同序列的不同区块以构成新视频序列的需求。

[0003] HEVC中的当前机制是在未考虑到这种方案的情况下设计的。利用当前HEVC机制实现这些方案意味着在区块上添加编码约束,并且解码时的区块的任意复合都涉及数据的重写处理。特别地,通常需要包括条带分段头部的操纵的数据的重写。

发明内容

[0004] 设计了本发明以解决上述问题中的一个或多个。本发明涉及帧部分的定义以及这些帧部分在位流中的表示(signaling)。本发明的目的是便于在解码时进行这些帧部分的提取和重新组合、同时限制在进行这些操作时所需的重写处理。

[0005] 根据本发明的第一方面,提供一种用于将包括帧的视频数据编码到位流中的方法,所述帧在空间上被划分为帧部分,所述方法包括:

[0006] -将至少一个帧部分编码到一个或多个第一编码单位中;

[0007] 其中,所述方法还包括:

[0008] -在所述第一编码单位中表示至少一个帧部分标识符,所述帧部分标识符标识一个编码帧部分;以及

[0009] -提供帧部分配置信息,所述帧部分配置信息包括所述帧部分标识符和与所述帧部分有关的空间信息。

[0010] 本发明的第一方面具有提供了更大的灵活性和更简单的操纵、同时使得能够与诸如HEVC区块等的已知区块化设计相比改善压缩的优点。

[0011] 在实施例中,将所述帧部分配置信息提供到一个第二编码单位中。

[0012] 在实施例中,所述至少一个帧部分是被独立编码的。

[0013] 在实施例中,所述方法还包括:提供指示所述帧部分已被独立编码的标志。

[0014] 在实施例中,所述一个或多个第一编码单位包括针对各帧部分指示该帧部分已被独立编码的标志。

[0015] 在实施例中,所述一个或多个第一编码单位包括指示所述至少一个帧部分已被独立编码的标志。

[0016] 在实施例中,所述一个或多个第一编码单位包括指示用于对所述帧部分进行编码的编码约束的级别的标志。

[0017] 在实施例中,所述帧部分是条带,并且所述第一编码单位是包括数据部分的条带

单位，所述标志包括在所述条带单位的所述数据部分的条带分段头部中。

[0018] 在实施例中，所述帧部分是条带，并且所述第一编码单位是包括数据部分的条带单位，所述帧部分标识符包括在所述条带单位的所述数据部分的条带分段头部中。

[0019] 在实施例中，所述第一编码单位包括头部部分和数据部分，所述数据部分包括所述编码帧部分，所述帧部分标识符包括在所述头部部分中。

[0020] 在实施例中，在所有的帧部分编码单位中表示帧部分标识符，并且预定义的帧部分标识符值指示所述帧部分未被独立编码。

[0021] 在实施例中，所述第二编码单位是专用于与一个或多个帧有关的信息的参数集。

[0022] 在实施例中，所述第二编码单位是专用于帧部分信息的参数集。

[0023] 在实施例中，第一编码单位具有指示所述帧部分已被独立编码的特定类型。

[0024] 在实施例中，所述帧部分标识符是使用固定的预定数量的位来编码的。

[0025] 在实施例中，所述帧部分标识符是使用所表示的数量的位来编码的。

[0026] 在实施例中，所述空间信息包括由编码树单位地址给出的所述帧部分的位置。

[0027] 在实施例中，所述空间信息包括由样本地址给出的所述帧部分的位置。

[0028] 在实施例中，所述空间信息包括所述帧部分的大小。

[0029] 在实施例中，所述帧部分的位置是关于所述帧而给出的。

[0030] 在实施例中，在针对同一帧部分包括不同的帧部分配置的所述位流中表示多个参数数据单位。

[0031] 在实施例中，所述第二编码单位包括指示给定的后滤波算法是否能够用于所述帧部分的标志。

[0032] 在实施例中，能够使用相同的帧部分标识符来标识定义帧部分集合的多个帧部分。

[0033] 在实施例中，所述头部部分包括层标识符，并且所述层标识符用于表示所述帧部分标识符。

[0034] 根据本发明的第二方面，提供一种用于从至少一个位流解码包括帧的视频数据的方法，所述帧在空间上被划分成帧部分，所述方法包括：

[0035] -从所述位流获得帧部分配置信息，所述帧部分配置信息包括帧部分标识符和与所述帧部分有关的空间信息；

[0036] -从所述位流中的一个或多个第一编码单位中至少提取帧部分，所述帧部分包括所述帧部分标识符；

[0037] -基于所述空间信息来确定所述帧部分在所述帧内的位置；以及

[0038] -根据所确定的位置来对所述帧部分进行解码，以将所述帧部分渲染成帧。

[0039] 在实施例中，将所述帧部分配置信息提供到一个第二编码单位中。

[0040] 在实施例中，所述至少一个帧部分是被独立编码的。

[0041] 在实施例中，所述方法还包括：获得指示所述帧部分已被独立编码的标志。

[0042] 在实施例中，所述一个或多个第一编码单位包括针对各帧部分指示该帧部分已被独立编码的标志。

[0043] 在实施例中，所述一个或多个第一编码单位包括指示所述至少一个帧部分已被独立编码的标志。

[0044] 在实施例中，所述一个或多个第一编码单位包括指示用于对所述帧部分进行编码的编码约束的级别的标志。

[0045] 在实施例中，所述帧部分是条带，并且所述第一编码单位是包括数据部分的条带单位，所述标志包括在所述条带单位的所述数据部分的条带分段头部中。

[0046] 在实施例中，所述帧部分是条带，并且所述第一编码单位是包括数据部分的条带单位，所述帧部分标识符包括在所述条带单位的所述数据部分的条带分段头部中。

[0047] 在实施例中，所述第一编码单位包括头部部分和数据部分，所述数据部分包括所述编码帧部分，所述帧部分标识符包括在所述头部部分中。

[0048] 在实施例中，在所有的帧部分编码单位中表示帧部分标识符，并且预定义的帧部分标识符值指示所述帧部分未被独立编码。

[0049] 在实施例中，所述第二编码单位是专用于与一个或多个帧有关的信息的参数集。

[0050] 在实施例中，所述第二编码单位是专用于帧部分信息的参数集。

[0051] 在实施例中，第一编码单位具有指示所述帧部分已被独立编码的特定类型。

[0052] 在实施例中，所述帧部分标识符是使用固定的预定数量的位来编码的。

[0053] 在实施例中，所述帧部分标识符是使用所表示的数量的位来编码的。

[0054] 在实施例中，所述空间信息包括由编码树单位地址给出的所述帧部分的位置。

[0055] 在实施例中，所述空间信息包括由样本地址给出的所述帧部分的位置。

[0056] 在实施例中，所述空间信息包括所述帧部分的大小。

[0057] 在实施例中，所述帧部分的位置是关于所述帧而给出的。

[0058] 在实施例中，从针对同一帧部分包括不同的帧部分配置的所述位流中获得多个参数数据单位。

[0059] 在实施例中，所述第二编码单位包括指示给定的后滤波算法是否能够用于所述帧部分的标志。

[0060] 在实施例中，能够使用相同的帧部分标识符来标识定义帧部分集合的多个帧部分。

[0061] 在实施例中，所述头部部分包括层标识符，并且所述层标识符用于表示所述帧部分标识符。

[0062] 根据本发明的第三方面，提供一种用于生成新位流的方法，所述新位流包括包含帧的视频数据，所述帧在空间上被划分成帧部分，所述方法包括：

[0063] -确定要从多个位流中提取并合并到新位流中的多个帧部分，所述多个位流是根据权利要求1至25中任一项进行编码的；

[0064] -确定要提取的帧部分的帧部分标识符；

[0065] -针对所述新位流生成帧部分配置信息；

[0066] -从所述多个位流中提取要提取的多个帧部分；以及

[0067] -将所述多个帧部分和所生成的帧部分配置信息嵌入到所述新位流中。

[0068] 在实施例中，所述方法还包括：

[0069] -针对所提取的帧部分确定新的帧部分标识符；以及

[0070] -用所提取的帧部分的所述新的帧部分标识符替代所述帧部分标识符。

[0071] 在实施例中，提取所述多个帧部分包括：

[0072] -解析所述多个位流;以及

[0073] -提取帧部分编码数据单位,所述帧部分编码数据单位包括所确定的帧部分标识符其中之一。

[0074] 根据本发明的第四方面,提供一种用于将包括帧的视频数据编码到位流中的装置,所述帧在空间上被划分成帧部分,所述装置包括被配置为进行以下操作的电路:

[0075] -将至少一个帧部分编码到一个或多个第一编码单位中;

[0076] 其中,所述方法还包括:

[0077] -在所述第一编码单位中表示至少一个帧部分标识符,所述帧部分标识符标识一个编码帧部分;以及

[0078] -提供帧部分配置信息,所述帧部分配置信息包括所述帧部分标识符和与所述帧部分有关的空间信息。

[0079] 根据本发明的第五方面,提供一种用于从至少一个位流中解码包括帧的视频数据的装置,所述帧在空间上被划分成帧部分,所述装置包括被配置为进行以下操作的电路:

[0080] -从所述位流获得帧部分配置信息,所述帧部分配置信息包括帧部分标识符和与所述帧部分有关的空间信息;

[0081] -从所述位流中的一个或多个第一编码单位中至少提取帧部分,所述帧部分包括所述帧部分标识符;

[0082] -基于所述空间信息来确定所述帧部分在所述帧内的位置;以及

[0083] -根据所确定的位置来对所述帧部分进行解码,以将所述帧部分渲染成帧。

[0084] 根据本发明的第六方面,提供一种用于生成新位流的装置,所述新位流包括包含帧的视频数据,所述帧在空间上被划分成帧部分,所述装置包括被配置为进行以下操作的电路:

[0085] -确定要从多个位流中提取并合并到新位流中的多个帧部分,所述多个位流是根据权利要求1至25中任一项进行编码的;

[0086] -确定要提取的帧部分的帧部分标识符;

[0087] -针对所述新位流生成帧部分配置信息;

[0088] -从所述多个位流中提取要提取的多个帧部分;以及

[0089] -将所述多个帧部分和所生成的帧部分配置信息嵌入到所述新位流中。

[0090] 根据本发明的第七方面,提供一种可编程设备所用的计算机程序产品,所述计算机程序产品包括指令序列,所述指令序列用于在被载入所述可编程设备并由所述可编程设备执行的情况下,实现根据本发明的方法。

[0091] 根据本发明的第八方面,提供一种存储有计算机程序的指令的计算机可读存储介质,所述指令用于实现根据本发明的方法。

[0092] 根据本发明的第九方面,提供一种计算机程序,其在执行时,使得进行本发明的方法。

[0093] 根据本发明的方法的至少一部分可以通过计算机来实现。因此,本发明可以采用以下形式:完全以硬件实现的实施例;完全以(包括固件、常驻软件、微代码等的)软件实现的实施例;或者组合这里通常可全部称为“电路”、“模块”或“系统”的软件和硬件方面的实施例。此外,本发明可以采用以介质中嵌入有计算机可用程序代码的表现的任何有形介质

中体现的计算机程序产品的形式。

[0094] 由于本发明可以以软件来实现,因此本发明可以体现为计算机可读代码以提供至可编程设备的任何适当载体介质上。有形的非暂时性载体介质可以包括诸如软盘、CD-ROM、硬盘驱动器、磁带装置或固态存储器装置等的存储介质。暂时性载体介质可以包括诸如电气信号、电子信号、光学信号、声学信号、磁信号或者例如微波或RF信号的电磁信号等的信号。

附图说明

[0095] 现在将仅通过示例的方式并且参考以下附图来说明本发明的实施例,其中:

[0096] 图1示出本发明可以集成于的系统;

[0097] 图2示出基于块的视频编码器的图像编码结构(例如,HEVC);

[0098] 图3示出根据两种分区(在HEVC中被称为条带分段和区块)的图像的划分;

[0099] 图4示出HEVC中用于跨越图像的边界(boundary)的CTU的四叉树推理机制;

[0100] 图5示出例如在HEVC中使用的边框(border)扩展机制;

[0101] 图6示出HEVC位流组织的示例;

[0102] 图7示出用于关注区域(ROI)流传输的HEVC分区的示例;

[0103] 图8a和8b示出用于关注区域的组合的两个不同使用方案示例;

[0104] 图9示出本发明集成于的视频编码器的典型编码处理;

[0105] 图10示出本发明集成于的视频解码器的典型解码处理;

[0106] 图11描述本发明的示例性使用;

[0107] 图12提供与封装步骤有关的详情;

[0108] 图13a、13b和13c示出通过编码处理所进行的帧部分配置的表示;

[0109] 图14示出基于非网格的分区的示例;

[0110] 图15示出用以表示CTile标识符的替代实施例;

[0111] 图16a示出包括各CTile的依赖性列表的XPS;

[0112] 图16b示出CTile依赖性的第一示例;

[0113] 图16c示出CTile依赖性的第二示例;

[0114] 图17(17a和17b)提供CTile可以在连续编码帧之间改变位置或大小的实施例的示例;以及

[0115] 图18是用于实现本发明的一个或多个实施例的计算装置的示意框图。

具体实施方式

[0116] 将视频序列的帧编码到空间帧部分中例如在与所谓的360度视频的流传输有关的方案中特别有用,其中这些360度视频实际上是将360度全景视频或球面视频投影到经典的2D视频表示上的结果。

[0117] 360度视频(或仅仅是360视频)是可以具有非常高的分辨率以提供良好的用户体验的视频。当在头戴式显示器的内部(或在屏幕上)显示时,仅将360视频内容的空间子部分呈现给用户。

[0118] 因而,令人关注的是,例如利用如经由HTTP的动态自适应流传输(DASH)那样的流

传输协议来仅针对用户正查看的区域请求具有高质量的空间帧部分。对于未看到的区域(即,用户未正查看的区域),可以简单地跳过空间帧部分。

[0119] 本发明的应用涉及使流传输适应于用户的观看方向的流传输方法。换句话说,该应用是指视口依赖流传输。对于这样的方法,在存储成本、计算成本和用户体验之间的一个良好折衷是将序列编码到具有各种质量的独立空间帧部分中。然后,根据需求和带宽约束,可以随机地访问、提取帧部分、以及/或者将这些帧部分与其它帧部分序列组合。该应用不要求任何附加的编码或转码。参考图8a来说明这样的方案的示例。

[0120] 应用涉及视频监控系统,在该视频监控系统中,将多个不同视频的空间帧部分重新组织在新视频中以匹配从系统操作员请求的结构。例如,操作员可能仅想要原始视频的部分。特别地,在图8b中示出该应用。

[0121] 最后,在另一应用中,仅包括从完整视频序列中提取的单一帧部分的新“视频”可以涉及在新视频中的空间帧部分的新位置不同的情况下重写编码参数。

[0122] 当使用HEVC时,空间帧部分的编码是基于HEVC区块的。然而,HEVC区块、并且更一般地HEVC型区块未被设计为解决上述应用。

[0123] 图1描述本发明可以集成于的系统(例如,交互式流传输视频系统)的实施例。

[0124] 将视频位流从服务器或代理服务器100通过网络101发送至客户端102。服务器100使用由视频编码器103生成的、符合基于块的视频编解码器(例如,HEVC视频编解码器)的规范的视频流(或视频文件)。

[0125] 如以下所述,编码器在根据本发明提供对一些空间帧部分的空间随机访问的同时,对具有不同的率/失真权衡的视频序列的集合进行压缩。

[0126] 服务器100通过通信网络101接收对可用于交互式流传输的视频流的描述的请求。通信网络101是基于互联网协议标准的。为了在IP网络101上发送媒体呈现所采用的标准协议优选是MPEG DASH:经由HTTP的动态自适应流传输。然而,本发明也可用于任何其它流传输协议。

[0127] 图2示出根据两种分区(条带分段和空间帧部分)的对图像的划分。图像206被划分为三个条带分段。条带分段是图像的一部分或整个图像。各条带分段包含整数个编码块(其可以对应于HEVC的编码单位)。编码块由样本构成。

[0128] 两种条带分段为独立条带分段207和依赖条带分段208。各条带分段嵌入在一个NAL单元中,该NAL单元是具有供用在面向分组和面向位流的传输系统中的通用格式的结构。两个类型的条带分段之间的区别在于以下事实:在独立条带分段头部中指定的数据定义了对条带分段的编码块进行解码所需的所有参数。另一方面,依赖条带分段的头部减小,并且依赖条带分段依赖于先前的第一个独立条带分段来推断在其头部中不可用的参数。条带中的第一编码单位的地址是在独立条带分段头部中指定的。

[0129] 图3示出向空间帧部分(SPF)的另一分区允许如帧305所示将各帧拆分成独立编码的矩形区域。

[0130] 如HEVC型区块那样,空间帧部分包含整数个编码块。与条带边界类似,SPF边界310破坏了所有的帧内预测机制。

[0131] 如HEVC型区块那样,SPF是在用于初始化解码处理的特定NAL单元内所包括的图片参数集中定义的。PPS NAL单元包括可以指定图片中的区块行的数量和区块列的数量及其

关联大小的句法元素。其它参数集NAL单元(例如,视频参数集或VPS、序列参数集或SPS)传送用于描述位流的编码结构的参数。在本发明中,将这些参数集中的任何参数集称为XPS(X用作通配符)。一个条带分段中的SPF位置(例如,以位为单位的偏移)是利用在条带分段头部的末尾处可用的句法元素来标识的。

[0132] SPF和条带分段可以联合使用,但存在一些限制。必须验证以下条件中的一个或两个:

[0133] -一个条带(或条带分段)的所有编码块都属于同一SPF;或者

[0134] -一个SPF的所有编码块都属于同一条带(或条带分段)。

[0135] 这意味着一个条带(或条带分段)可以包含多个完整SPF、或者仅仅是单个区块的子部分。其次,SPF可以包含多个完整条带(或条带分段)、或者仅仅是单个条带(或条带分段)的子部分。

[0136] 图4示意性示出在HEVC中用于跨越图像的边界的编码单位的四叉树推理机制,仅供例示目的用。在HEVC中,图像不限于编码单位大小的宽度和高度的倍数。然后,帧的最右编码单位可以跨越图像的右边界401,并且帧的最下编码单位可以跨越图像的下边界402。在这些情况下,HEVC定义了针对跨越边界的编码单位的四叉树推理机制。该机制在于递归地拆分正跨越图像边界的编码单位的任何CU,直到不再存在跨越边界的CU为止、或者直到针对这些编码单位达到最大四叉树深度为止。例如,编码单位403不是自动拆分的,而编码单位404、405和406是自动拆分的。不存在所推断出的四叉树的表示:解码器必须在图像边界上推断相同的四叉树。然而,例如,如407那样,针对在帧的内部的编码单位,可以通过(在未达到最大四叉树深度的情况下)针对这些编码单位表示拆分信息来进一步细化自动获得的四叉树。

[0137] 图6示出从服务器发送至客户端的典型视频位流600。该位流符合HEVC或基于块的位流。

[0138] 位流600被组织为一系列网络抽象层(NAL)单元。存在多个种类(类型)的NAL单元。参数集NAL单元(例如,HEVC所用的VPS、SPS和PPS)描述用于对序列进行编码的编码工具的结构。这些参数集NAL单元还描述与图像的特性(分辨率、帧频等)有关的一些信息。

[0139] 第一NAL单元601是提供整个位流的信息的视频参数集(VPS)。特别地,第一NAL单元601指示位流中的可分级层的数量。

[0140] 下一NAL单元602是序列参数集(SPS)。该NAL单元602提供序列级参数。之后是提供图片级参数的图片参数集(PPS)NAL单元603。然后可以提供条带分段604。通常针对各帧具有一个条带分段。条带分段604可以包含到具有各种NAL单元类型(CRA、IDR、BLA、RASL、RADL、STSA、TSA或TRAIL...)的NAL单元中。包含条带分段的NAL单元由NAL头部605(将通过图10的说明来提供NAL头部的进一步描述)和原始字节序列有效载荷(RBSP)606组成。NAL头部605包含包括NAL单元类型的信息。RBSP(即,NAL单元数据)包含特定于NAL单元类型的信息。在条带分段的情况下,RBSP包含条带分段头部607以及之后的条带分段数据608。条带分段数据是条带分段的光栅扫描顺序的编码树单位609的一系列编码数据。

[0141] 在(这里未示出的)实施例中,可以在位流中在条带分段NAL单元之前插入被称为TPS(其是区块化参数集的缩写)的参数集。相应的参数是有效的,直到找到新的TPS为止。TPS描述帧的分区结构。

[0142] 在另一实施例中，如果在位流中不存在TPS，则假定在位流中存在仅一个空间帧部分。所述空间帧部分具有与视频帧相同的尺寸，并且位于其原点处。

[0143] 位流可以包含独立帧部分或关注区域(ROI)。图7示意性示出在这里被视为帧内的矩形区域的关注区域的示例。ROI在HEVC和基于块的编解码器中是众所周知的。

[0144] 流传输ROI或独立帧部分意味着分区策略。这对编码效率有影响，因为引入区块边界会破坏一些HEVC预测机制。

[0145] 在图7中，帧700是按4×4SPF网格分区的。为了访问预定义的ROI 701，选择嵌入有SPF 6、7、10和11的相应条带分段的NAL单元并将这些NAL单元发送至客户端。

[0146] 优选地，在本发明中，在SPF中嵌入有一个独立条带分段以及零个或更多个依赖条带分段。优点是以独立于包括ROI的帧的其它部分的方式保证对该ROI的访问。

[0147] 实际上，需要提醒的是，对于HEVC和更一般地对于基于块的编解码器，HEVC区块或类似区块破坏了这些区块的边界处的所有帧内预测机制(循环滤波处理除外)。因此，不允许空间预测机制。然而，多个预测机制依赖于视频序列的帧之间的数据的时间冗余以改善压缩性能。例如，可以从部分或全部在当前HEVC区块边界的外部的预测结果块来预测HEVC区块中的一个块。此外，预测结果块也可以部分或全部在帧边界的外部，因为HEVC提供了众所周知的边框扩展机制来扩展图像的边框，以允许预测结果块部分或全部在参考图像的外部。

[0148] 最后，预测结果块可以位于子像素位置处。这意味着，参考块像素值是子像素插值滤波器的结果，该子像素插值滤波器从在位于与预测结果块相对应的完整像素坐标处的像素块的外部的多达四个像素的范围来生成子像素值。结果，时间预测可以在HEVC区块内的块与位于HEVC区块边界的外部的一组像素数据之间引入编码依赖性。

[0149] 时间预测中涉及的第二HEVC机制在于使用运动矢量预测结果对运动矢量进行预测编码。

[0150] 最后，HEVC提供了一组环路滤波器，其引入连续区块的像素之间的依赖性。这些环路滤波器是消除特别是通过残差块的量化所引入的一些伪影的去块滤波器和SAO滤波器。HEVC在图片参数集中提供标志，以指示这些环路滤波器在区块和/或条带边界处是否被禁用。当被禁用时，这些压缩工具不会引入区块之间的编码依赖性。

[0151] 为了保证对关注区域进行解码(这意味着独立地对关注区域进行解码)，解决方案是禁用前述的预测机制的一部分或全部。

[0152] 这会导致由此得到的位流的压缩不够高效且其位率较高。可以根据关注区域使用方案来调整预测机制的激活/去激活，以便优化如此得到的位流的位率。

[0153] 图8a和8b示出针对以上已经说明的关注区域的组合的两个不同的应用示例。

[0154] 例如，在第一示例中，图8a表示来自由四个关注区域构成的两个不同视频流的两个帧800和801。第一视频流800具有高质量编码参数，并且第二视频流801是低质量的，因而是低位率版本。客户端高效地将高质量版本的关注区域#3与区域1、2和4的低质量关注区域组合。这许可强调关注区域#3的质量，同时针对其它不太重要的区域将位率维持相对较低。

[0155] 在第二示例中，在图8b中示出四个视频流(803、804、805和806)的集合。在该使用方案中，客户端形成各视频流的不同关注区域的新拼接视频。该客户端将各视频流的关注区域重排或组合到如此得到的视频流中的新位置。

[0156] 根据本发明的实施例，提出了定义在这里被称为约束区块(在以下说明中其快捷方式为CTile)的空间帧部分。CTile是指属于帧序列的空间帧部分，该帧序列被划分成可被随机访问且在无解码误差的情况下进行完全解码的空间帧部分。CTile的解码可以独立于其空间位置和/或其邻域而进行。换言之，CTile是独立编码的、或者以解码器始终能够无任何误差地进行解码的方式编码的。

[0157] 对与形成CTile的样本相对应的编码数据进行独立编码。例如，数据被编码到形成条带(或具有类似特征的帧的任何其它部分)的编码单位或NAL单元中，使得一个解析器可以提取与CTile相对应的样本。结果，两个CTile被编码到两个不同集合的编码单位中。为了在任何空间位置处进行解码，CTile不能是条带中的包含其它CTile的部分。因而，与CTile相对应的条带的编码数据独立于来自其它条带的编码数据。

[0158] 与条带相对应的编码数据可被进一步划分为多个条带分段编码单位。

[0159] 在实施例中，CTile是严格独立可解码的，这意味着对形成CTile的编码数据进行解析所需的所有数据都包含在所述CTile中。此外，预测机制使用根据同一CTile的编码数据计算出的预测信息。对于INTER(帧间)预测，参考块是从另一帧中的相同CTile检索到的。

[0160] 在其它实施例中，可以解除编码限制。

[0161] 在第一其它实施例中，将边框扩展机制用于图像边界。将在图5中示意性示出的该机制用于CTile边界，以允许不受限制和更高效的运动补偿。

[0162] 图5以简化方式示出例如在HEVC中使用的边框扩展机制。该机制允许以在帧的外部的样本值为基准来进行INTER预测(众所周知的预测模式，其允许使用在当前帧的外部的数据)中的运动补偿参。

[0163] 当在对帧502进行编码的同时预测块501时，允许从跨越参考帧504的边界的块503进行预测是有用的。这例如允许从部分在先前帧的视野外部的相同内容来预测运动内容。优选地，定义样本填充方法，以允许访问参考图片的帧边界周围的空白内的样本。

[0164] 在第二其它实施例中，对于CTile，可以以完全不依赖于任何相邻区块信息和区块化结构的方式授权运动矢量预测结果(或任何其它预测结果)的导出机制。

[0165] 图9示出根据本发明的在视频编码器中实现的编码处理的示例。

[0166] 首先，对于所考虑的各输入视频序列900，在步骤901中，编码器确定帧向帧部分的分区(即，帧部分配置)。在一些实施例中，帧部分的大小是预先确定的，使得一个帧部分覆盖单个关注区域或关注区域的一部分。例如，帧部分可以具有512×512个像素的大小。

[0167] 然后，在步骤902中，编码器确定必须将哪些帧部分编码为CTile。例如，这样的帧部分可以对应于客户端可能想要单独解码或者客户端可能想要利用一个或多个其它关注区域构成的关注区域(ROI)。

[0168] 然后，在步骤903中，编码器确定标识符并将标识符指派至每个CTile。在变形例中，编码器可以确定标识符并将标识符指派至仅CTile的选择。在变形例中，可以推断出CTile的标识符。

[0169] 当将相同的标识符指派至多个编码帧中的CTile时，这意味着这些CTile属于同一CTile序列。CTile的序列(或CTile序列)可以独立于其它帧部分进行解码。仅需要来自CTile序列的数据来对所述CTile序列进行解码。换言之，来自CTile序列的CTile可以同时具有时间依赖性。

[0170] 在确定了CTile和CTile标识符之后，编码器根据编码结构对帧部分904进行压缩和编码。帧部分的编码确保了任何解码器均可以如前面所述对这些帧部分进行解码。

[0171] 在步骤905中，编码器生成帧部分配置信息。帧部分配置信息在于确定帧划分为帧部分的描述参数。帧部分配置信息还在于通过将CTile标识符与CTile标识符在帧或帧序列中的位置相关联来确定CTile的描述参数。如参考图13a、图13b和图13c(第一替代方案)所述或者在图15(第二替代方案)中所述，提出了不同的表示替代方案。步骤905包括生成参数集(XPS)中的表示区块化参数。在变形例中，步骤905是在编码步骤904之前而不是在编码步骤904之后进行的。

[0172] 步骤906包括将XPS的NAL单元以及压缩后的CTile帧部分可选地封装到位流中。

[0173] 例如，基于流传输协议，该步骤还可以包括对例如ISO BMF文件格式那样的更高级别的视频描述格式内部的位流进行封装。该步骤例如还可以允许将视频数据与音频数据复用。

[0174] 步骤901、902和903可以通过使用提供预先确定的帧部分位置的一个或多个配置文件来实现，从而提供关于帧部分是否是CTile以及必须将哪些标识符用于CTile的信息。在替代实施例中，例如使用深度神经网络或更简单的分割算法，可以通过视频内容的分析来自动确定帧部分和CTile。

[0175] 如在一些实施例中所述，步骤901可用于确定在整个视频序列内(或至少对于视频分段内的多个连续帧)恒定的分区。这意味着，CTile的位置和大小在至少在包含CTile序列的一部分的多个帧内的整个CTile序列内是恒定的。

[0176] 可替代地，所确定的帧部分可以在帧之间具有可变的大小和位置。

[0177] 图10示出在视频解码器中实现的解码处理的示例。解码处理涉及使用如上文定义的CTile。

[0178] 首先，视频解码器提取包含参数集(XPS)的NAL单元。在步骤1000中，从参数集获得帧部分配置信息。

[0179] 对于所考虑的各帧部分1001，解码器在步骤1002中根据帧部分配置信息中判断帧部分是否是CTile。

[0180] 如果帧部分被表示为CTile(在测试1003之后为分支“是”)，则解码器从帧部分配置信息提取(或推断)CTile标识符，并且在步骤1005中，由于CTile标识符和与该标识符相关联的CTile位置信息而确定CTile的解码位置。否则，当帧部分不是CTile时(在测试1003之后为分支“否”)，解码器在步骤1005从在帧部分编码数据中描述的定位信息以及从XPS信息来确定帧部分的解码位置。

[0181] 最后，解码器考虑到帧部分是否是CTile来对帧部分编码数据1006进行解码，并且将解码后的样本值放入渲染图片缓冲器的内部。

[0182] 图11描述利用图9的编码处理生成的两个位流的合并处理的示例(参见图8a和图8b中的应用)。合并处理意味着将所提取的CTile组合成新的视频位流以发送到客户端。

[0183] 合并处理从以下开始：在步骤1100中，确定要从一个或多个视频位流中提取并合并到新位流中的一组CTile。例如，图形用户界面允许用户选择该组CTile并将该该组CTile在帧中重排。在另一示例中，基于位流的内容来自动进行选择。应用可以选择包含移动内容的一组CTile。

[0184] 该处理在步骤1101中确定CTile在合并到新视频位流中时的新位置。

[0185] 一旦已知要提取的CTile，则在步骤1102中，通过获得所确定的要提取的CTile中的每一个的当前CTile标识符来确定这些CTile的新标识符。根据本发明的实施例，在帧配置信息中表示这些标识符。如前面所述，在替代实施例中，可以采用用于对输入位流进行封装的文件格式来描述帧配置信息。帧配置信息可以采用XPS和文件格式的形式存在。

[0186] 在标识符冲突、这意味着多于一个的CTile具有相同标识符的情况下，步骤1101还包括：确定新的CTile标识符以解决这些冲突。

[0187] 然后，该处理生成1103新视频位流的合并视频序列的帧部分配置信息。该处理包括：生成XPS其中之一中的将合并位流中的CTile的新位置与这些CTile的新CTile标识符相关联的参数。

[0188] 在步骤1104中，提取或获得在步骤1100中确定的一组CTile的编码帧部分数据。该步骤包括：检索包含CTile的编码帧部分数据的NAL单元。这可以通过解析输入位流中的所有NAL单元来进行，以便提取具有在步骤1102中确定的CTile标识符的NAL单元。当输入位流符合文件格式规范时，将与一个帧部分相对应的所有NAL单元都封装在一个容器(例如，ISOBMFF的视频轨)中。然后，步骤1104包括检索与所选择的帧部分的轨相对应的数据。

[0189] 最后，在可选步骤1105中，通过将XPS的NAL单元和包含所提取的CTile编码帧部分数据的NAL单元嵌入到新的位流中、以及可能地将该位流封装到更高级别描述格式中，来生成新位流。

[0190] 对于由于CTile标识符冲突因而在步骤1101中确定了新CTile标识符的CTile，步骤1105还包括修改NAL单元中所包含的、包括原始CTile标识符的头部。修改这些头部，使得用在步骤1102中确定的CTile标识符来替换原始CTile标识符。

[0191] 在一个示例中，图11的合并处理在于：从同一位流中提取CTile的子集。在这种情况下，无需处理标识符冲突。

[0192] 图13a、13b和13c示出根据本发明的多个实施例的、通过有关的编码处理所进行的帧部分配置的表示的示例。

[0193] 图13a示出根据本发明实施例的位流中的CTile的标识。

[0194] 在帧部分编码数据中指示在这里命名为ctile_unique_identifier 1301的CTile标识符。优选地，在属于帧部分编码数据的各数据序列(即，条带分段头部)中指示该标识符。因而，该标识符允许：

[0195] -容易地识别位流的哪些部分属于CTile，以及

[0196] -快速访问或提取这些部分。

[0197] 更准确地，在图13a所示的实施例中，在与具有标识符1301的CTile相对应的各条带分段的条带分段头部(slice_segment_header)1302中表示CTile标识符1301。

[0198] 如前面所述，解码器基于帧部分配置信息来解析CTile标识符以确定CTile的关联位置。根据实施例，如后面将参考图13b或图13c所述，在参数集(例如，TPS)中提供帧部分配置信息。

[0199] 为了简单起见，在下文，除非明确提到或不适用，否则将不区分CTile和CTile序列。此外，还可以将CTile标识符视为CTile序列标识符。

[0200] 在实施例中，为了将不一定需要标识符的HEVC型区块与CTile区分开，可以在属于

帧部分编码数据的数据序列中(例如,在条带分段头部中)使用例如ctile_flag 1303那样的信息。如果ctile_flag处于不活动状态(例如设置为“假”),则提供HEVC型区块的参数1304。这些参数可以包括例如first_slice_in_pic_flag或CTU地址(slice_segment_adress)那样的区块定位信息、或者如slice_pic_parameter_set_id那样的对其它位流元素的参考。这些句法元素取决于帧分区,并且可能会因视频序列而有所不同。

[0201] 当ctile_flag处于活动状态时,省略这些参数,而作为代替将提供包含CTile唯一标识符1301的CTile特定信息。为了允许在CTile中具有多个条带的可能性,一个解决方案是提供在这里命名为ctb_addr_offset_inside_tile 1305的信息。该信息1305还用于相对于具有所考虑的帧的CTile位置指定开始条带分段的解码的位置。例如,该位置相对于CTile的开始及其(以CTB为单位)宽度用编码块(例如,作为HEVC标准编码树块的CTB)的原始扫描顺序编号来表示,因此ctb_addr_offset_inside_tile信息独立于CTile编码/解码位置。

[0202] 在另一实施例中,不使用标志ctile_flag。例如,在所有的区块(CTile和其它区块(HEVC型区块))中都存在CTile标识符。可以使用预定值(例如值零)来标识HEVC型区块。

[0203] 在实施例中,提供信息以标识空间帧部分是否是CTile。

[0204] 在另一实施例中,假定仅CTile是唯一使用的帧部分,则不提供用以标识空间帧部分是否是CTile的信息。

[0205] 优选地,在给定帧中,不存在多于一个的具有给定标识符的CTile。在时间上依赖的所有CTile中(例如,在CTile序列中)使用相同的CTile标识符。因此,如果提取出连续编码图片中的具有相同CTile标识符的CTile,将正确地解码这些CTile。

[0206] 换句话说,CTile标识符是标识编码视频序列内的CTile的唯一标识符。在实施例中,将CTile标识符插入CTile中所包含的条带分段的条带头部中。这意味着,在位流中,与CTile相对应的NAL单元(条带分段)包含CTile标识符。因此,可以基于该CTile标识符来从位流中容易地解析和提取任何CTile。

[0207] 在位流中表示CTile配置信息是有利的。例如,CTile配置信息由CTile的数量、关联的CTile标识符和CTile在帧中的位置来定义。

[0208] 图13b示出根据本发明实施例的CTile配置。

[0209] 在第一实施例中,编码器指定与要解码的图片中的CTile的帧部分配置信息有关的附加表示信息。该表示信息是在参数集(XPS)中、优选在区块化参数集(TPS)中提供的。优选地,附加表示信息包括在这里被称为num_ctiles的图片中的CTile的编号1311。针对各CTile,该附加表示信息关联CTile的唯一标识符1312以及在这里被称为tile_ctb_addr的CTile位置1313(其意味着图片的内部的解码位置)。CTile位置在图片中被提供为解码位置。CTile位置可以用(例如,相对于光栅扫描顺序的)CTB索引编号来表示。

[0210] 在另一实施例中,在图13a所示的条带分段头部中的由1304指定的部分中也提到的、在这里被命名为slice_pic_parameter_set_id的参数是指代表TPS的唯一标识符。在变形例中,唯一标识符表示PPS。在该另一实施例中:

[0211] -各TPS包括用于标识TPS的tile_parameter_set_id(为了简单起见而未示出)参数(例如,每当在图片中CTile配置改变时,这意味着编码器可以生成新TPS),推荐生成具有相同TPS(或PPS)唯一标识符的TPS,以避免重写帧的各条带头部;

[0212] -条带分段头部的slice_pic_parameter_set_id 1314等于应用于条带的TPS的tile_parameter_set_id。在这种情况下,条带分段头部的slice_pic_parameter_set_id的命名可被重命名为slice_tile_parameter_set_id。

[0213] 在一个替代方案中,TPS标识符在条带数据中是未指定的:解码器推断出在条带NAL单元之前的最后一个TPS NAL单元包含当前CTile的帧部分配置。

[0214] 图13c示出根据本发明的另一实施例的CTile配置。根据本实施例,TPS1320包含指示区块数量减去1的参数值,例如“num_tiles_minus1”1321。可替代地,TPS包含直接地或间接地提供帧中的区块的数量的例如被命名为“num_tiles”的参数值。

[0215] 在实施例中,如果TPS指示存在仅一个帧部分,则假定该帧部分是具有与视频帧相同的尺寸且位于其原点处的CTile。否则(TPS描述多个帧部分),如先前实施例那样描述帧部分位置。

[0216] 在另一实施例中,如果不存在TPS,则假定存在一个具有与视频帧相同的尺寸的CTile。所述一个CTile位于其原点。

[0217] 根据另一实施例,TPS可以利用类似于HEVC网格的句法来描述空间帧部分网格:例如,指定“num_tile_rows_minus1”、“num_tile_cols_minus1”和“uniform_spacing_flag”。如果未设置“uniform_spacing_flag”,则还指定各列的宽度和各行的高度(可以推断出的最后一行和最后一列的大小除外)。如果设置了“uniform_spacing_flag”,则例如在HEVC规范中那样,根据图片宽度和高度来计算CTile的宽度和高度。在这样的实施例中,由于网格索引允许定位相应的CTile,因此可以(例如使用区块的光栅扫描顺序)用与空间帧部分网格索引相对应的CTile编号来表示CTile位置。

[0218] 根据替代实施例,“ctile_flag”被可以取多个值的“ctile_level”替代,各值指示应用于CTile的编码约束的不同级别。例如,ctile_level等于零,这表示CTile不受约束(如HEVC型CTile那样)。ctile_level等于“1”,这表示CTile受到约束使得可以(在无该CTile的原始邻域的情况下)单独提取且适当地解码该CTile,或者可以利用该CTile的原始邻域对该CTile进行解码,但在CTile与其它CTile混洗(shuffle)的情况下,可能无法适当地对该CTile进行解码。ctile_level等于“2”,这表示CTile受到约束使得可以在任何地方对CTile进行解码,并且CTile可以与任何邻域(相当于先前实施例中的ctile_flag等于1)混洗。

[0219] 在另一实施例中,“ctile_level”仅提供编码器做出其编码决定以满足约束级别的信息。因而,具有任何约束级别的CTile的解码处理可以通过与HEVC型区块相同的解码处理来实现(例如,不对CTile边界进行边框扩展)。

[0220] 在另一实施例中,编码和解码处理对于所有级别的约束不都是相同的。例如,ctile_level等于“1”的CTile使用与HEVC型CTile相同的解码处理(在编码器处使用的一些限制不影响解码器),而ctile_level等于“2”的CTile必须使用CTile边界上的边框扩展、并且使用运动矢量预测结果的列表的特定导出处理来进行解码。

[0221] 根据另一实施例,即使HEVC型区块也可能需要具有标识符(例如,以关联XPS中的这些区块的参数),然后以与CTile标识符类似的方式在条带分段头部中指定该标识符。在给定帧中,HEVC型区块不具有与CTile或另一HEVC型区块相同的标识符。

[0222] 根据一个实施例,编码器通过在参数集其中之一(例如,PPS或TPS)中表示各空间帧部分的“ctile_flag”的值来指示空间帧部分是CTile。例如,编码器针对帧的各区块生成

唯一标识符。当描述帧部分配置时，编码器将标志(例如，ctile_flag)与各区块唯一标识符相关联。当相应区块(即，标识符等于关联的唯一标识符的区块)的编码受到约束以确保独立解码时，该标志为真。相反，当区块的编码未受到足够约束以保证独立解码时，该标志为假。

[0223] 根据第二实施例，编码器生成包括另一标志(例如，all_ctile_flag)的帧部分配置信息。如果该标志设置为“1”，则这意味着在帧部分配置中描述的所有区块都是CTile。表示空间帧部分是否是CTile的标志(例如，各ctile_flag)被省略且被推断为等于真。如果该标志设置为零，则使用先前实施例其中之一来明确地描述CTile。如果参数特定于HEVC型区块，则代替在条带分段头部中而是在XPS中(例如在TPS中)表示这些参数。例如，对于另一实施例而在1304中并入参考的slice_segment_address是特定于HEVC型区块的。在实施例中，在TPS指示TPS是否也指示出空间帧部分不是CTile。本实施例允许简化条带分段头部的句法和解析。

[0224] 根据另一实施例，编码器代替使用条带分段头部中的“ctile_flag”，针对与CTile相对应的分段数据来定义新的NAL单元类型。例如，编码器针对来自在CTile内部的瞬时解码刷新(IDR)帧的条带NAL单元来定义CTILE_IDR NAL单元。编码器定义与编码格式针对常规分段数据指定的NAL单元类型一样多的新NAL单元类型。例如，HEVC针对清洁随机存取(CRA)图片的条带分段来定义以下的NAL单元类型CRA_NUT；针对随机存取可解码前导(RADL)IDR图片的条带分段来定义IDR_W_RADL；针对在位流中不存在关联的前导图片的IDR图像的条带分段来定义IDR_N_LP；针对断链访问(BLA)图片的条带分段来定义BLA_W_LP、BLA_W_RADL、BLA_N_LP；针对随机访问跳过前导(RASL)图片的条带分段来定义RASL_N、RASL_R；针对RADL图片的条带分段来定义RADL_N、RADL_R；针对逐步时间子层访问(STSA)图片的条带分段来定义STSA_N、STSA_R；针对时间子层访问(TSA)图片的条带分段来定义TSA_N、TSA_R；以及针对非TSA、非STSA拖尾图片的条带分段了定义TRAIL_N、TRAIL_R。

[0225] W_LP：可以具有关联的RASL或RADL图片；W_RADL：无关联的RASL图片；N_LP：无关联的前导图片；*_N：图片是子层非参考(SLNR)图片(否则图片是子层参考图片)；*_R：图片仅是子层参考图片。

[0226] 可以利用具有用于受约束的区块数据的相同目的的新的相应NAL单元类型CTILE_BLA_*、CTILE_CRA_*、CTILE_IDR_*、CTILE_RASL_*、CTILE_RADL_*、CTILE_STSA_*、CTILE_TSA_*、CTILE_TRAIL_*，来扩展这些HEVC NAL单元类型。使用这些新NAL单元类型其中之一来指示NAL单元属于CTile。

[0227] 由于编码器仅必须解析各NAL单元的开头位以判断条带数据是否在CTile内，因此该替代方案简化了解码处理。

[0228] 优选地，通过在编码时针对给定序列(这意味着在给定位流中)的构造来保证CTile标识符的唯一性。然而，当对来自不同序列(这意味着来自不同位流)的CTile进行混洗时，不能保证唯一性。根据实施例，为了便于空间帧部分与可能来自各种序列的CTile混洗，在有限数量的位上，CTile标识符是唯一的。该唯一值可以是随机值，例如哈希值或不一定代表其位置的任何其它值。因而，降低了在采取不同位流的CTile时具有标识符冲突的概率。

[0229] 在实施例中，当进行来自多个序列的CTile的混洗时，在两个CTile标识符之间发

生冲突的情况下，替换冲突的CTile标识符就足够了。为了通过不必重新生成所有的条带分段头部来高效地实现此，在优选实施例中，使用固定的预定数量的位来对CTile标识符进行编码。例如，在图13a和13b中，将CTile标识符编码在8位上。

[0230] 在替代实施例中，将序列或图片的所有标识符都编码在相同数量的位上。该数量的位是在一个参数集(诸如SPS、PPS或RPS等)的参数中指定的，例如“uid_num_bits”。在条带分段头部中，优选(当需要不是8的倍数的位数时)在CTile标识符之后具有字节对齐机制。可替代地，位数可以用字节数(8位)来表示：例如“uid_num_bytes”。当将来自各种序列的CTile混洗在一起时，可能需要在CTile标识符不是全部具有相同的位数时改变CTile标识符。这将要求改变多个条带分段头部，但与更新条带分段头部相比更容易，因为将仅需添加/删除或替换字节。

[0231] 在又一替代实施例中，可以利用可变数量的位来对各CTile标识符进行编码。该数量的位是在条带分段头部中指定的。可替换地，可以根据CTile标识符所使用的代码来自动确定位数：使用可变字节长度代码，例如指数哥伦布编码(或等同的可变长度代码，之后是字节对齐位)。

[0232] 根据实施例，在依赖条带分段头部中不表示CTile标识符，以减小表示大小。然后，依赖条带分段头部的CTile标识符是从先前的独立条带分段头部推断出的。根据替代实施例，在依赖条带头部中表示CTile标识符，以便于解析和提取包含CTile的子位流。

[0233] 作为利用“tile_ctb_addr[i]”1313或“slice_segment_address”1306编码单位地址来表示CTile位置的替代，引入了具有更细粒度定位的细粒度CTile。该粒度可被细化到亮度样本位置，但在另一实施例中，与“2”的幂相对应的多个亮度样本的粒度(小于CTU大小)就足够了。在一些实施例中，粒度可以是预先确定的。在替代实施例中，粒度例如是在VPS、SPS或PPS中表示的。当使用细粒度CTile时，CTile的尺寸不一定是CTU大小的倍数。

[0234] 当CTile的大小不是CTU大小的倍数时，如利用图4所示，CTile的右方和底边的编码单位正使用与图片的右方和底边的HEVC CTU所使用的自动拆分机制相同的自动拆分机制。

[0235] 根据替代实施例，即使编码单位不完整，句法也将描述完整编码，从而为分解树(例如，四叉树或QTBT)的率失真优化留出一些空间，并且允许最终填充适合于改善压缩的信息。

[0236] 在HEVC型区块中，尺寸是使用网格指定的。因此，所有的HEVC型区块都按行和列对齐，并且给定行的所有HEVC型区块都具有相同的高度，且给定列的所有HEVC型区块都具有相同的宽度。在XPS中指定各列的宽度和各行的高度。利用细粒度的HEVC型区块，允许不太严格的配置(例如，允许多个ROI的更高效编码)可能是便利的。

[0237] 根据实施例，可以在CTile的条带分段的条带分段头部中指定CTile的尺寸。为了减小如此得到的位流的大小，仅在第一条带分段中指定CTile的尺寸。之后的条带分段重复使用相同的CTile尺寸。作为替代，将所有的CTile的尺寸例如连同CTile位置一起都设置在XPS中。

[0238] 作为另一替代，在CTile的第一条带分段头部以及XPS这两者中都提供尺寸。

[0239] 作为另一替代，CTile的尺寸不是提供的，而是从用于在XPT中提供区块信息(例如，位置或“ctile_flag”)的排序以及从CTile位置推断出的：例如，在XPS中声明CTile的位

置,并且对CTile位置进行排序,使得CTile的各相应右下角是(例如)按光栅扫描顺序增大的方式来排序。在下文,图14提供这样的排序的示例。

[0240] 根据实施例,在图13a的示例中使用的dependent_slice_segment_enabled_flag具有与HEVC中相同的含义:其用于指示是否允许依赖条带分段。在HEVC中,dependent_slice_segment_enabled_flag是在PPS中表示的。根据优选实施例,针对各CTile,在tiling_parameter_set(TPS)中表示dependent_slice_segment_enabled_flag(以允许将利用依赖条带分段编码的CTile以及未利用依赖条带分段编码的CTile用在同一位流中)。为了减少所有的CTile都是在利用依赖条带分段或不利用依赖条带分段的情况下编码的常见用例的TPS中的句法,在TPS结构的根部使用另一标志:dependent_slice_segment_enabled_flag_for_all_ctiles。当该标志设置为1时,针对各CTile不表示dependent_slice_degment_enabled_flag。作为代替,ctile_dependent_slice_segment_enabled_flag也是在TPS结构的根部表示的,并且提供针对各CTile的dependent_slice_degment_enabled_flag所要推断的值。对于HEVC型区块,dependent_slice_segment_enabled_flag仍可以在PPS中表示,但在优选实施例中,dependent_slice_segment_enabled_flag是在TPS中表示的。

[0241] 根据替代实施例,根本不会表示dependent_slice_segments_enabled_flag,并且始终将dependent_slice_segments_enabled_flag推断为真,以简化句法。

[0242] 图14示出基于非网格的分区的示例。帧1401被拆分成编号为#1～#15的15个CTile。该编号提供了在XPS中声明区块位置的顺序,使得各区块的右下角按光栅扫描顺序排序。使用该排序,可以推导出各CTile的大小。例如,采用最后一个区块(CTile#15),可以推导出该区块的尺寸,因为作为最后一个区块,该区块的右下角按光栅顺序是最后一个,因此是帧的右下角。然后,条带#15的尺寸是帧的尺寸-条带的位置:h#15＝h_frame-y#15;w#15＝w_frame-x#15。区块#14必须具有CTile#15之前的最后一个右下角作为自身的右下角,因此右下角位置是最下方(帧的下方)和最右方(恰好在前一区块的左方)。然后,CTile#14尺寸是h#14＝h_frame-y#14、w#14＝x#15-x#14。针对区块#13和#12重复相同的操作。然后,对于区块#11,由于填充最下方位置,因此新的最下方位置是y#14。然后,h#11＝y#14-x#11,等等,直到CTile#1为止。

[0243] 根据替代实施例,代替在XPS中指定CTile位置,仅指定CTile尺寸,并且CTile位置是使用根据左上位置(例如,按光栅扫描顺序增大的方式)排序后的CTile根据CTile尺寸所计算出的。用以根据尺寸计算位置的算法可以容易地从前述的用以根据位置计算尺寸的算法导出。

[0244] 根据实施例,在XPS中描述的CTile参数可以针对非现有的CTile提供CTile位置(和/或CTile尺寸):针对这些CTile将不存在条带分段。在仅提供了位置或尺寸并且推断出尺寸或位置的实施例中,需要该CTile描述以允许适当推断。

[0245] 对于视频渲染,使用默认样本值或填充方法来填充非现有的CTile,或者可替代地,在XPS参数中设置填充方法的值或索引。这例如可以通过在解码处理的开头(例如在图9中的步骤900之前)添加预备步骤来实现,该预备步骤在于:

[0246] -利用适当的默认样本值来初始化渲染缓冲器中的帧的内容,以及/或者

[0247] -在解码了所有的帧部分之后添加新步骤,该步骤在于例如通过使用图像修复方

法来填充未被任何区块或CTile覆盖的所有区域。

[0248] 根据实施例，可以处理相同空间位置处的多个CTile或者重叠的CTile。对于各CTile标识符，存在关联的解码CTile缓冲器(相当于HEVC中的解码图片缓冲器(DPB)，但在这里仅包含解码CTile数据)。对于给定帧，使用在关联的解码CTile缓冲器中可利用的时间数据来对各CTile进行解码。然后，根据第一替代方案，CTile的渲染顺序是与位流中的CTile顺序相同的顺序。在另一替代方案中，CTile与根据XPS数据可以确定的渲染顺序相关联。对于这两个替代方案，将各CTile的解码结果的样本按CTile的渲染顺序放入渲染帧缓冲器的帧中(然后，可能地擦除/掩蔽由先前CTile顺次先前放入的样本)。

[0249] 根据实施例，CTile样本还包括阿尔法通道，该阿尔法通道指示当在渲染帧缓冲器的帧中渲染CTile时应当应用的透明度水平。可替代地，样本还包括二进制掩码值，该二进制掩码值指示在渲染帧缓冲器的帧中必须渲染CTile的哪些样本。

[0250] 根据实施例，在可以处理相同位置处的多个CTile或重叠的CTile的情况下，必须在XPS中指定CTile位置和CTile大小这两者，因为在这种情况下不能从这两者其中之一推导出另一个。

[0251] 根据实施例，对于任何给定的后滤波算法(例如，去块滤波器、样本自适应偏移或自适应环路滤波器)，可以在XPS中指定CTile边界后滤波标志，以指示后滤波算法是否可用于CTile。CTile边界后滤波标志(例如，“usable_for_post_filtering_flag”)指示出给定的后滤波算法可以应用于渲染帧缓冲器(而不是解码图片缓冲器，因为解码图片缓冲器可能修改时间解码)的渲染帧中的CTile边界。有利地，后滤波算法的目的例如是提高视觉质量。可以针对整个帧级别和/或针对每一个CTile来指定标志。该标志可用于防止已知在被后滤波时容易引入伪影的一些边缘的滤波。例如，对于在自适应质量流传输的情况下的CTile混洗，标志将为真，但在CTile边界在360°内容的立方体投影的两个面之间、并且这些面在该边缘上不邻接的情况下，标志将为假。后滤波后的CTile边框是边缘的两侧正指定可以应用后滤波、或者当边缘在HEVC型区块和授权后滤波的CTile之间时的CTile边框。

[0252] 根据替代实施例，在该实施例中，可以在解码图片缓冲器(DPB)中对CTile边界进行后滤波，以确保解码在任何解码结构中均是正确的，并且当使用INTER预测时，预测所使用的样本是未经后滤波的样本。因而，使用边界信息将边框扩展机制应用于后滤波后的样本之前的边框上的最后样本，这意味着对未经滤波的样本进行边框扩展。

[0253] 根据替代实施例，多于一个的CTile可以具有相同的CTile标识符。在本实施例中，CTile标识符成为CTile集合标识符。形成CTile集合的一组CTile必须保持在一起且具有相同的相对定位，以被适当地解码。

[0254] 在这些实施例中，从XPS推断出CTile集合的位置和大小。CTile集合的位置和大小对应于属于CTile集合的一组CTile的包围框的位置和大小。因而，在XPS中，CTile集合标识符与一个或多个位置和大小相关联(针对CTile集合中的各CTile存在一个位置和大小)。

[0255] 在这些实施例中，条带分段头部“ctb_addr_offset_inside_tile”1305信息可以被“ctb_addr_offset_inside_tile_set”信息代替。“ctb_addr_offset_inside_tile_set”允许推导出哪个CTile属于条带分段，因此推导出在对条带分段进行解码时要使用的几何形状。

[0256] 在这样的实施例其中之一中，可以使用一组CTile的任何样本来进行时间运动补

偿。如果运动补偿使用在CTile集合的外部的样本值,则将该样本值设置为CTile集合中的任一个CTile的在空间上最接近的样本的值(相当于应用边框扩展,但仅适用于不由两个CTile共享的CTile边界)。如果CTile的外部的任何样本具有多于一个的最接近CTile样本,则使用简单规则来确定使用哪个样本(例如,光栅扫描顺序最小的样本)。

[0257] 图15示出图13a、13b和13c所示的用以表示CTile标识符的实施例的替代实施例。在当前的基于块的编解码器(典型HEVC)中,NAL单元头1501包含以下字段:

[0258] -设置为0的一位:(False(假));

[0259] -包含NAL单元类型的六位:(Type(类型));

[0260] -包含层标识符的六位:(LayerID),其在HEVC中始终等于零,但对应于可分级HEVC(SHVC)中的可分级层索引、或者例如多视图HEVC(MV-HEVC)中的视图索引;以及

[0261] -表示时间层标识符的三位:(TID),其对应于HEVC中的时间可分级性的时间层索引。

[0262] 在实施例中,基于NAL单元头部1501,编码器将视频序列拆分成帧部分。编码器针对各帧部分使用一个编码或可分级层。这可被视为基于空间区域的层编码。编码器可以将各空间区域层独立于其它区域进行编码。在这种情况下,各空间区域层对应于一个CTile。在该特定情况下,编码时的空间区域层的所有条带的ctile_flag都被设置为真。主要区别在于,各空间区域层可被进一步划分为HEVC型区块。

[0263] 编码器用LayerId表示不同的空间区域层。编码器将LayerId的值设置成等于CTile的标识符。结果,在条带分段头部中不需要CTile标识符。由于CTile标识符具有固定的位长度,因此当将视频流的帧部分混洗时,对CTile标识符的处理保持简单。

[0264] 编码器在参数集其中之一(例如,VPS)中表示帧部分配置。VPS通过利用可以与在先前实施例中所述的句法相对应的句法将空间区域层的唯一标识符与解码位置相关联来指示各空间区域层的解码位置。

[0265] 编码器还描述视频流的不同层之间的依赖性。然后,解码器通过分析在参数集NAL单元中描述的层之间的依赖性来确定独立于其它层而编码的空间区域层。

[0266] 编码器将空间区域层的子集以独立于其它空间区域层的方式压缩为CTile。当一个空间区域层依赖于先前帧(即,具有相同的CTile标识符的帧)中的另一空间区域层(该层中的条带的ctile_flag被设置为假)时,编码器将来自该依赖层的参考帧添加到当前层的解码图片缓冲器中。当两个层的大小不同时,应用上采样或下采样滤波器,使得参考帧的大小等于当前层的大小。

[0267] 根据实施例,还可以使用LayerId来推断ctile_flag:当LayerId为零时,NAL单元属于HEVC型区块。当LayerId不为零时,NAL单元属于CTile标识符等于LayerId的CTile。可选地,保留一位的LayerId以表示ctile_flag。使用LayerId来传输CTile标识符的优点是大大降低了解析位流以提取CTile标识符的复杂性。

[0268] 在另一实施例中,与HEVC中的时间可分级性类似地定义空间区域可分级性,即不同的层标识符标识来自其它可分级层(例如SNR、分辨率、多视图)的时间、空间区域。实际上,该方法的优点是可以使用空间区域可分级层以及SNR或分辨率可分级层这两者。

[0269] 利用现在指示帧部分标识符的随机访问标识符(RAID)来扩展NAL单元头部1502。LayerId语义保持与HEVC的LayerId语义相同,即LayerId语义指示多视图、SNR或分辨率可

分级层。

[0270] 编码器通过在参数集其中之一(例如,VPS)中将其RAID值与解码位置相关联来指定各空间区域层的位置。对空间区域进行编码的各NAL单元(包括SPS、PPS和VCL NAL单元)的RAID等于与该空间区域相对应的帧部分标识符(CTile(集合)标识符)。

[0271] 结果,上述合并处理(其在于从一组视频位流中提取CTile并将这些CTile组合到新的视频位流中)从与要组合的视频流有关的帧部分配置中提取要合并的空间区域层的CTile标识符。然后,该合并处理提取RAID值等于所提取的标识符的集合的所有NAL单元。

[0272] 为了限制在组合两个视频序列时标识符冲突的风险,编码器将RAID值设置为随机值。这包括视频序列包含单个帧部分的情况。

[0273] 根据实施例,RAID指定空间区域是否是CTile(RAID代替条带分段头部中的ctile_flag的表示):当RAID为零时,NAL单元属于HEVC型区块。当RAID不为零时,NAL单元属于标识符等于RAID的CTile。可替代地,保留一位的RAID以表示ctile_flag。在替代实施例中,RAID标识符是16位或24位,以允许更多的CTile。

[0274] 根据实施例,CTile序列被认为是独立位流。例如,在一些实施例中,具有相同标识符的CTile的序列顺序可以不同于具有另一标识符的CTile的序列顺序不同(即,GOP结构在两个CTile之间可以不同)。因此,同一帧中的两个CTile可以具有不同的NAL单元Type或TID。

[0275] 在另一实施例中,XPS包括描述CTile之间的一些依赖性的附加信息,使得解码器能够无任何误差地处理这些CTile。已经看到,可以在一组CTile的级别而不是在各CTile的级别考虑CTile的独立性。在该结构中,CTile集合内的一些CTile可能具有某些依赖性。

[0276] 例如,图16a示出包括各CTile的依赖性列表的XPS。依赖性列表提供依赖于具有标识符1600的给定CTile的CTile标识符1601。当给定CTile被指示为具有对另一CTile的依赖性时,这意味着在无该另一CTile的情况下无法提取给定CTile。

[0277] 图16b示出CTile依赖性的第一示例。当前帧1602中的CTile#1在根据先前编码帧1603进行运动补偿时使用来自CTile#2的样本值;并且CTile#2使用来自CTile#1的样本值。在这样的示例中,XPS指示具有标识符#1的CTile具有对具有标识符#2的CTile的依赖性,并且具有标识符#2的CTile具有对具有标识符#1的CTile的依赖性。这些相互依赖性的表示是用以表示CTile集合的替代方案。

[0278] 图16c示出CTile依赖性的第二示例。在这样的示例中,存在于帧1605和1607中的CTile#3具有对帧1604和1607中的CTile#1和CTile#2的依赖性。在这样的示例中,不能在也没有提取CTile#1和CTile#2的情况下提取CTile#3。但CTile#1和CTile#2不具有依赖性,并且可以单独提取。根据实施例,可以应用该方案以便于例如在帧1604～1606按时间排序的情况下,以各种帧频提取CTile。可替换地,该方案可用于例如在帧1605是帧1604的细化层、并且帧1607是帧1606的细化层的情况下进行可分级编码,以便于针对不同质量的层提取CTile。

[0279] 根据一些实施例,CTile可以在连续帧之间改变空间位置或大小。

[0280] 根据实施例,还介绍了“区块参数集”即TilePS。TPS允许更新CTile的子集(仅移动和/或改变大小的子集)的CTile参数,TPS例如包含“num_updated_tiles”值,然后TPS将区块标识符与“num_updated_tiles”修改后的CTile的新属性相关联。

[0281] 传统上,运动矢量给出参考图像中的预测结果块相对于与要编码的块同位置的块的位置。对于要编码的给定块,第一步骤是在参考图像中识别同位置块。同位置块被定义为参考图像中的具有相同位置(这意味着与要编码的块具有相同的原点(左上位置)和相同的大小)的块。然后,将运动矢量应用于同位置块的原点,以确定预测结果块的原点。

[0282] 当考虑CTile时,同位置块的确定适合于考虑在CTile内具有相同位置但在帧内不再具有相同位置的块。当对CTile进行了混洗时,这意味着CTile在帧中的位置在解码时的帧中相对于CTile在编码时的帧内的位置已被修改。然而,考虑到预测被限制在CTile内以确保独立解码,仍可以通过将运动矢量应用于与CTile内的编码块同位置的块来确定正确的预测结果块。这是适用的,只要CTile在帧之间保持其大小和位置即可。当CTile在帧中改变其位置和/或在连续两个帧之间改变其大小时,出现困难。在这种情况下,编码器和解码器商定用于确定同位置块在运动矢量要应用于的参考帧中的位置以正确地确定预测结果块的方式。

[0283] 根据CTile可以在连续编码帧之间改变位置或大小的实施例,两个连续帧中的CTile的相对位置在两个不同的位流中可能不相同。图17提供这样的实施例的示例,其中第一位流包含视频监控的帧1700。在第一帧1701中,存在包括针对具有给定ctile_id的移动关注区域的CTile 1702的多个空间帧部分。在另一帧1703中,具有给定的ctile_id的CTile 1704已移动且已改变了大小。第二位流包含通过将从1700中提取的CTile与所生成的包含均匀颜色(例如,黑色)的CTile组装所生成的视频1705。在第一帧1706中,从第一位流中提取了具有给定的ctile_id的CTile 1702,并将该CTile放入帧1707的中心。在另一帧1708中,从第一位流中提取了具有给定的ctile_id的CTile 1704,并将该CTile放入帧1709的中心。在第一位流中,CTile 1704使用具有对CTile 1701的时间参考的INTER预测。因此,在所生成的视频1705中,CTile 1709使用具有对CTile 1707的时间参考的INTER预测。CTile 1702和1704之间的相对空间位置与在1707和1709之间不相同。因此,为了不论解码相对位置如何都适当地进行解码,当使用INTER预测模式时,编码运动矢量不考虑CTile位置变化(即,连续帧之间的相对空间位置)。

[0284] 根据第一替代方案,计算运动矢量,仿佛两个连续编码帧中的CTile的预定参考点在相同的空间位置(例如,左上、右上、左下、右下、中上、中下、左中、右中或中心)处。因此,编码运动矢量也将对应于如下:帧参考中的针对块的运动矢量减去帧参考中的在CTile的参考点之间的运动矢量:因而获得了相对于CTile的参考中的运动矢量。结果表明,CTile随后可独立于后续空间改变而解码。

[0285] 图17b示出:对于编码在帧1711中的CTile 1710,使用参考帧1713中的具有相同的ctile_id的参考CTile 1712。使用与帧1716中的运动矢量和预定参考点1717间的运动矢量之间的差相对应的运动矢量1715来对块1714进行编码(在该示例中,预定参考点是CTile的左上角)。图17b还示出:即使CTile的相对时间解码位置与编码位置不相同,当通过将块1718添加到解码帧1719中的预定参考点之间的运动矢量以获得解码帧1720中的运动矢量来对块1718进行解码时,编码矢量1715仍有效。

[0286] 根据第二替代方案,计算运动矢量,仿佛CTile的给定点在两个连续编码帧中的相同空间位置处一样。该给定点在CTile编码数据中被表示为预定点(例如,左上、右上、左下、右下、中上、中下、左中、右中或中心)的列表中的索引。

[0287] 根据第三替代方案，考虑固定点(或可替代地表示点)，并且将运动矢量编码在CTile编码数据中。当考虑到两个连续编码帧中的CTile的固定(或表示)点在相同的空间位置处时，编码在CTile编码数据中的运动矢量提供添加至与CTile中的时间预测相关联的INTER运动矢量中的每一个INTER运动矢量的运动矢量。这允许编码器降低运动矢量的编码成本。例如，编码器可以选择CTile的运动补偿块的平均运动矢量。例如，如图17b所示，该平均运动矢量将被减至矢量1715。相减的结果被视为要编码的运动矢量。在结果方面等同的替代方案是代替提供固定点或固定点索引以及运动矢量(或者除了提供固定点或固定点索引以及运动矢量之外)，提供编码CTile中的参考点的(子)像素位置。

[0288] 根据第四替代方案，考虑固定点(或可替代地，表示点)。将运动矢量场的参数编码在CTile编码数据中。运动字段在考虑到两个连续编码帧中的CTile的固定(或表示)点在相同的空间位置处时，允许确定添加至与CTile中的时间预测相关联的INTER运动矢量中的每一个INTER运动矢量的运动矢量。例如，编码器可以估计CTile中的块的运动矢量，并且可以估计使这些块的预测最小化的运动矢量场，从而最小化这些块的残差并由此降低这些块的编码成本。然后，INTER编码块的各运动补偿矢量是将从运动矢量场参数计算出的运动矢量减至运动补偿矢量(例如，1715)的结果。

[0289] 根据实施例，INTER预测模式可以参考多于一个的先前编码的参考帧。在该实施例中，可以扩展前述实施例以考虑CTile的固定(或表示)点在编码帧和每一个参考帧中都对齐。

[0290] 在表示运动矢量或运动矢量场的实施例中，可以以两个替代方式进行向多个帧的扩展：

[0291] -通过表示与参考帧的数量一样多的运动矢量(或运动矢量场参数)，或者

[0292] -通过表示仅一个运动矢量(或运动矢量场)“x”，其用于根据该运动矢量与编码帧的时间差来针对各参考帧导出一个运动矢量(或运动矢量场)。

[0293] 例如，使用线性缩放：如果参考帧的时间位置是“t-s”(其中：“s”是帧之间的恒定时间采样时间段，并且“t”是时间)、并且编码帧的时间位置是“t”，则所使用的缩放因子是“(t)/s-(t-s)/s＝1”，但如果参考帧的时间位置是“t+2s”，则所使用的缩放因子是“(t)/s-(t+2s)/s＝-2”。应用缩放因子以针对各参考帧计算运动矢量。例如，如图17b所示，在参考帧1713是“t+2s”的情况下，将运动矢量1715减去-2*“x”。相减的结果“y”是编码后的运动矢量的值(例如，如果编码模式是HEVC的帧间预测模式，则“y”是使用运动矢量预测结果索引所预测的运动矢量)。换言之，在解码器侧，针对运动补偿块对运动矢量“y”进行解码，然后将运动矢量“y”加上-2*“x”以获得矢量1715。进一步加上运动矢量1717以获得帧级别的运动矢量。

[0294] 图12提供在如前面的描述所述正将位流封装成更高级别的描述格式时的与编码器侧的封装步骤906或1105有关的详情。

[0295] 在优选实施例中，根据ISO基媒体文件格式(ISOBMFF、ISO/IEC14496-12和14496-15)来对具有CTile的视频位流进行封装。在与图12有关的以下说明中，如针对ISOBMFF所定义的，术语“样本”对应于“帧”、即来自与编码图片相对应的视频位流的NAL单元的集合。

[0296] 封装由ISOBMFF或mp4写入器处理。该写入器包含NAL单元头部的解析器。该写入器能够提取NALU类型、标识符和相应的压缩数据。通常，将所提取的NALU数据放置在封装文件

的媒体数据容器即“mdat”框中。将NAL单元的描述所用的元数据放置在主“moov”框下方的框的结构化层级结构中。将一个视频位流封装在由具有子框的“trak”框所描述的视频轨中。

[0297] 对于分区视频帧，根据视频的预期使用，存在不同的可能封装。例如在初始化步骤1200中，该使用可被硬编码在mp4写入器应用中，或者可以由一个用户或另一程序提供作为输入参数。在实施例中，将一个帧部分或给定的一组帧部分封装到一个视频轨中，这可能是方便的，从而得到多轨封装。

[0298] 一旦进行了ISOBMFF写入器的初始化，编码器在步骤1201中通过读取NALU类型、特别是与参数集(XPS)相对应的NALU类型来解析视频位流。如以上已经说明的，参数集是提供与视频位流等的编码结构有关的高级别和一般信息的特定NAL单元。

[0299] 从这些参数集的解析，mp4写入器可以在测试1202中判断视频位流是否包含帧部分(例如：在参数集其中之一中是否存在TPS或特定分区结构)。如果存在帧部分，则mp4写入器在相同的测试1202中判断这些帧部分是否是“约束区块”、即CTile。如果位流不包含帧部分或不包含CTile，则测试1202为假，并且在步骤1203中将视频位流封装为一个视频轨。

[0300] TPS(区块化参数集)被视为参数集信息的一个NALU，并且可被嵌入在元数据中，该元数据提供解码器结构、或者如在专用于样本描述等的框其中之一(例如，“stsd”框)(通常在一些编解码器特定的样本条目)中可以找到的DecoderConfigurationRecord(解码器结构记录)框那样的设置信息。

[0301] 可替代地，根据本发明的实施例，TPS可被处理为视频数据的NAL单元(VCL NALU)，并且可被作为一个样本数据存储在“mdat”框中。TPS也可以同时存在于样本条目和样本数据中。如果帧部分结构沿着视频序列改变，则将TPS存储在样本级别(样本数据)而不是样本描述级别(样本条目)是更方便的。

[0302] 当帧部分分区结构的变化需要在接收侧重置解码器时，ISOBMFF写入器优选将来自视频位流的TPS和CTile相关信息存储在样本条目中。该解码器重置允许接收或消耗文件的装置考虑新分区结构。新分区结构例如可以包含要支持的编码工具(即，配置文件)或要处理的数据量(即，级别)的指示。根据分区结构的配置文件和级别值或其它参数，该装置可以支持或不支持新分区结构。当不支持时，装置可以调整传输，或者可以在可用时选择视频的替代版本。当支持新结构时，装置继续文件的解码和渲染。

[0303] 当需要对ROI的空间访问时(在测试1204之后分支为“是”)，ISOBMFF写入器可以根据使用情况而具有不同的封装策略。空间访问例如是指基于ROI或分区的显示(即，仅提取和解码与ROI或部分或帧部分或帧部分的集合相对应的数据的一部分)或者基于ROI或部分的流传输(即，仅传输与ROI或部分或帧部分的集合相对应的数据的该一部分和元数据)。如果预期使用情形是用于本地显示的存储、这对应于测试1205为真(分支“是”)，则将分区视频位流存储在一个轨中、但包括向需要空间访问的ROI或帧部分或帧部分的集合的NALU映射可以是方便的。在步骤1206中生成NALU映射。对于ISOBMFF写入器，NALU映射针对视频位流中的各NALU在于列出与给定CTile有关的NAL单元或者具有相同的随机访问标识符(图15中的RAID参考1502)(即，对应于可选择和可解码的帧部分或空间区域)。为了能够进行该列出，ISOBMFF写入器的NALU解析器模块根据位流生成的实施例，检查在步骤903中指派至需要空间访问的ROI或帧部分或帧部分的集合的标识符的值。

[0304] 如果位流不包含在针对需要空间访问的ROI或帧部分或帧部分的集合的NALU头部特定标识符中，则ISOBMFF写入器需要条带头部解析器来获得在903中指派的CTile的标识符的值、例如ctile_unique_identifier(在图13a中标记为1301)。

[0305] 然后，在步骤1206中，创建NALUMapEntry(NALU映射条目)结构“nalm”作为“trak”框层级结构下方的框，以存储NALU的列表以及它们到帧部分或帧部分集合的映射。对于各帧部分或帧部分的集合，类型“trif”的SampleGroupDescriptionBox(样本组描述框)针对各帧部分或帧部分的集合提供该帧部分或帧部分的集合的描述，例如提供来自ISO/IEC 14496-15的TileRegionGroupEntry(区块区域组条目)的参数。将帧部分描述符“trif”的group_ID值设置为要封装的CTile的标识符的值。

[0306] 然后，在步骤1203中，将所有帧部分或帧部分的集合的数据封装为单个轨。当使用情形是流传输时、这对应于测试1208为真(分支“是”)时，将与视频中的空间访问级别相对应的各帧部分或帧部分的集合拆分成专用轨，这可以是方便的，并且在测试1208为真时，进行单个轨封装。

[0307] 对于流传输使用情况，针对各帧部分或帧部分的集合，在步骤1209中关于NALU映射生成帧部分描述。可以通过解析TPS来确定帧部分或帧部分的集合的数量。使用“trif”样本组，甚至可以使用默认样本组，因为针对在步骤1210中生成的各帧部分轨(用于封装与一个帧部分或帧部分的集合有关的数据的轨)存在一个帧部分或一组帧部分。然后，根据ISO/IEC 14496-15，所有样本映射到作为帧部分描述符“trif”的相同样本组描述。将帧部分描述符“trif”的group_ID值设置为要封装的CTile(或RAID(在存在的情况下)的标识符的值。

[0308] 然后，在步骤1210中，将各帧部分或帧部分的集合插入自身轨即帧部分轨中。帧部分轨包括特定样本条目，这些特定样本条目指示样本实际是视频的空间部分，并且当没有剩余更多的要封装的帧部分或帧部分集合时(测试1211)，参考在步骤1212中创建的帧部分基轨。该帧部分基轨包含与包括定时参数集(TPS)的参数集相对应的特定NAL单元。帧部分基轨利用特定轨参考类型顺次参考各帧部分轨，以允许隐式重建对帧部分或帧部分的集合的任意选择。步骤1212可以被复合轨替代，在该复合轨中，被称为提取器的NAL单元提供根据一个或多个帧部分轨的显式重建。

[0309] 然后，提取器针对复合轨的各样本，通过简单地使提取器指向帧部分或帧部分的集合的给定标识符(通常参考相应CTile的标识符(或者在存在的情况下为RAID))，来允许帧部分或帧部分的集合的任何配置(即使不同于原始配置)。

[0310] 当在步骤1212中使用复合轨时，步骤1210中的帧部分轨实际上可以是可解码的帧部分轨，这意味着各自包含(在1209中生成的)帧部分描述以及参数集。由于提取器可以以不同的方式重新组合，因此在各帧部分轨中存在TPS是可选的。然后，样本描述可以指示符合使用中的编解码器的样本条目：例如，如果HEVC在使用中，则为“hvc1”或“hvc2”；如果AVC(高级视频编码)在使用中，则为“avc1”或“avc2”；或者明确地标识使用中的视频编码器的任何保留的四字符代码。

[0311] 图18是用于实现本发明的一个或多个实施例的计算装置1800的示意框图。计算装置1800可以是诸如微计算机、工作站或轻型便携式装置等的装置。计算装置1800包括通信总线，其中该通信总线连接至以下组件：

[0312] -诸如微处理器等的表示为CPU的中央处理单元1801；

[0313] -表示为RAM的随机存取存储器1802,用于存储本发明实施例的方法的可执行代码以及如下寄存器,其中这些寄存器被适配为记录实现根据本发明实施例的方法所需的变量和参数,其中RAM 1802的存储器容量例如可以利用连接至扩展端口的可选RAM来扩展;

[0314] -表示为ROM的只读存储器1803,用于存储用于实现本发明实施例的计算机程序;

[0315] -网络接口1804,其通常连接至进行要处理的数字数据的发送或接收所经由的通信网络。网络接口1804可以是单个网络接口、或者包括不同的网络接口的集合(例如,有线接口和无线接口或者不同种类的有线接口或无线接口)。在CPU 1801中所运行的软件应用的控制下,将数据包写入网络接口以供发送或者从网络接口读取数据包以供接收;

[0316] -用户接口1805,其可用于从用户接收输入或向用户显示信息;

[0317] -表示为HD的硬盘1806,其可被设置为大容量存储装置;

[0318] -I/O模块1807,其可用于相对于诸如视频源或显示器等的外部装置进行数据的发送/接收。

[0319] 可执行代码可以存储在只读存储器1803中、硬盘1806上或者例如盘等的可移除数字介质上。根据变形例,程序的可执行代码可以在执行之前利用通信网络经由网络接口1804来接收,从而存储在通信装置1800的诸如硬盘1806等的存储部件其中之一内。

[0320] 中央处理单元1801被适配为控制和引导根据本发明实施例的一个或多个程序的指令或软件代码的一部分的执行,其中这些指令存储在上述存储部件其中之一内。在通电后,CPU 1801例如能够在从程序ROM 1803或硬盘(HD)1806加载了来自主RAM存储器1802的与软件应用有关的指令之后,执行这些指令。这种软件应用在由CPU 1801执行的情况下,使得进行本发明的流程图的各步骤。

[0321] 本发明的算法的任何步骤均可以通过利用可编程计算机(诸如PC(“个人计算机”)、DSP(“数字信号处理器”)或微控制器等)执行一组指令或程序来以软件实现;或者利用机器或专用组件(诸如FPGA(“现场可编程门阵列”)或ASIC(“专用集成电路”)等)来以硬件实现。

[0322] 尽管以上参考具体实施例对本发明进行了描述,但本发明不限于这些具体实施例,并且对本领域技术人员而言,在本发明的范围内的修改将是显而易见的。

[0323] 许多其它修改和改变在参考仅以示例方式给出的并且并不意图限制本发明的范围的前述例示实施例时向精通本技术的人员表明这些修改和改变仅是由所附权利要求书来确定的。特别地,在适当情况下,可以互换来自不同实施例的不同特征。

[0324] 上述本发明的各个实施例可以单独实现或作为多个实施例的组合来实现。此外,在需要的情况下或在将各个实施例中的元件或特征组合成一个实施例有益的情况下,可以组合来自不同实施例的特征。

[0325] 除非另外明确说明,否则本说明书中所公开的各特征(包括任何所附权利要求书、摘要和附图)均可被服务相同、等同或类似目的的替代特征所取代。因而,除非另外明确说明,否则所公开的各特征仅仅是通用的一系列等同或类似特征的一个示例。

[0326] 在权利要求书中,词语“包括”没有排除其它元件或步骤,并且不定冠词“a”或“an”没有排除多个。在相互不同的从属权利要求中陈述不同的特征的仅有事实并不表明不能有利地使用这些特征的组合。

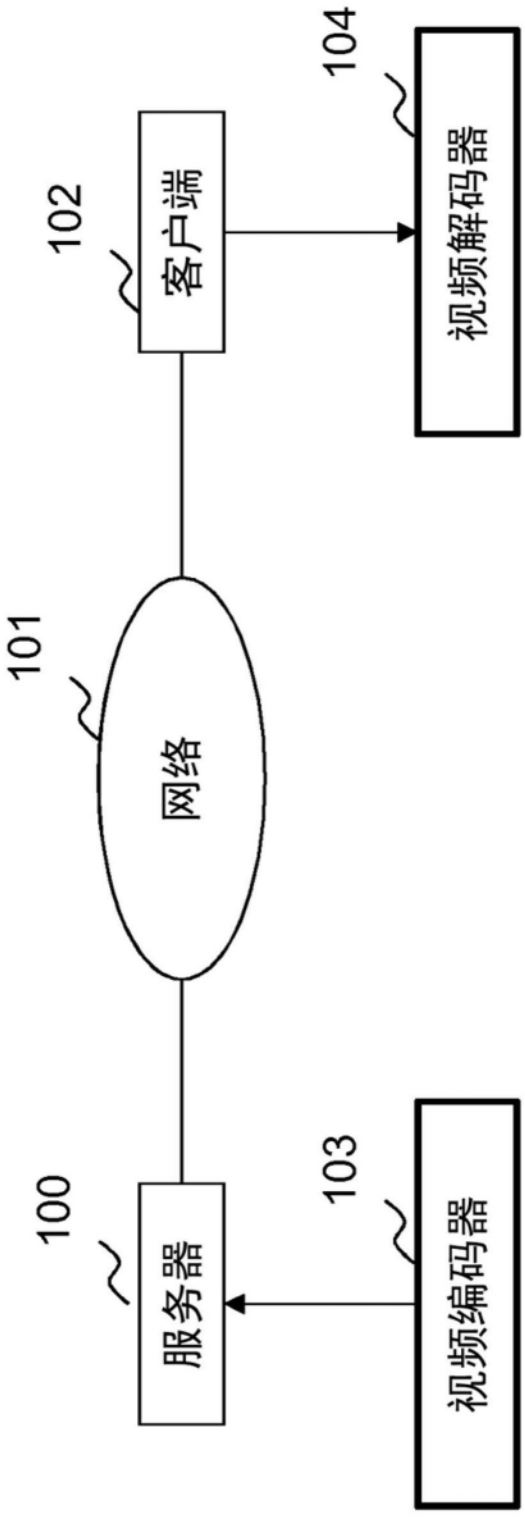


图1

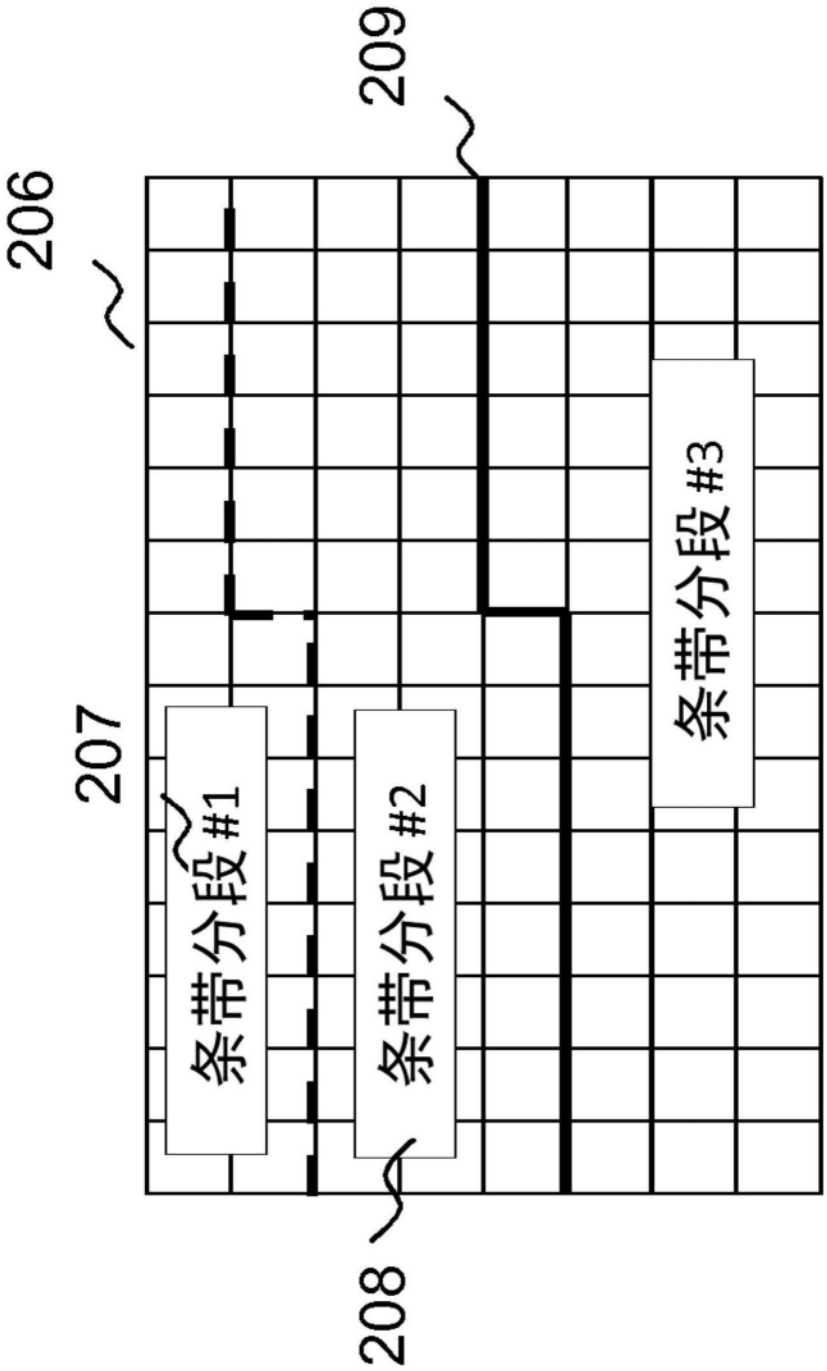


图2

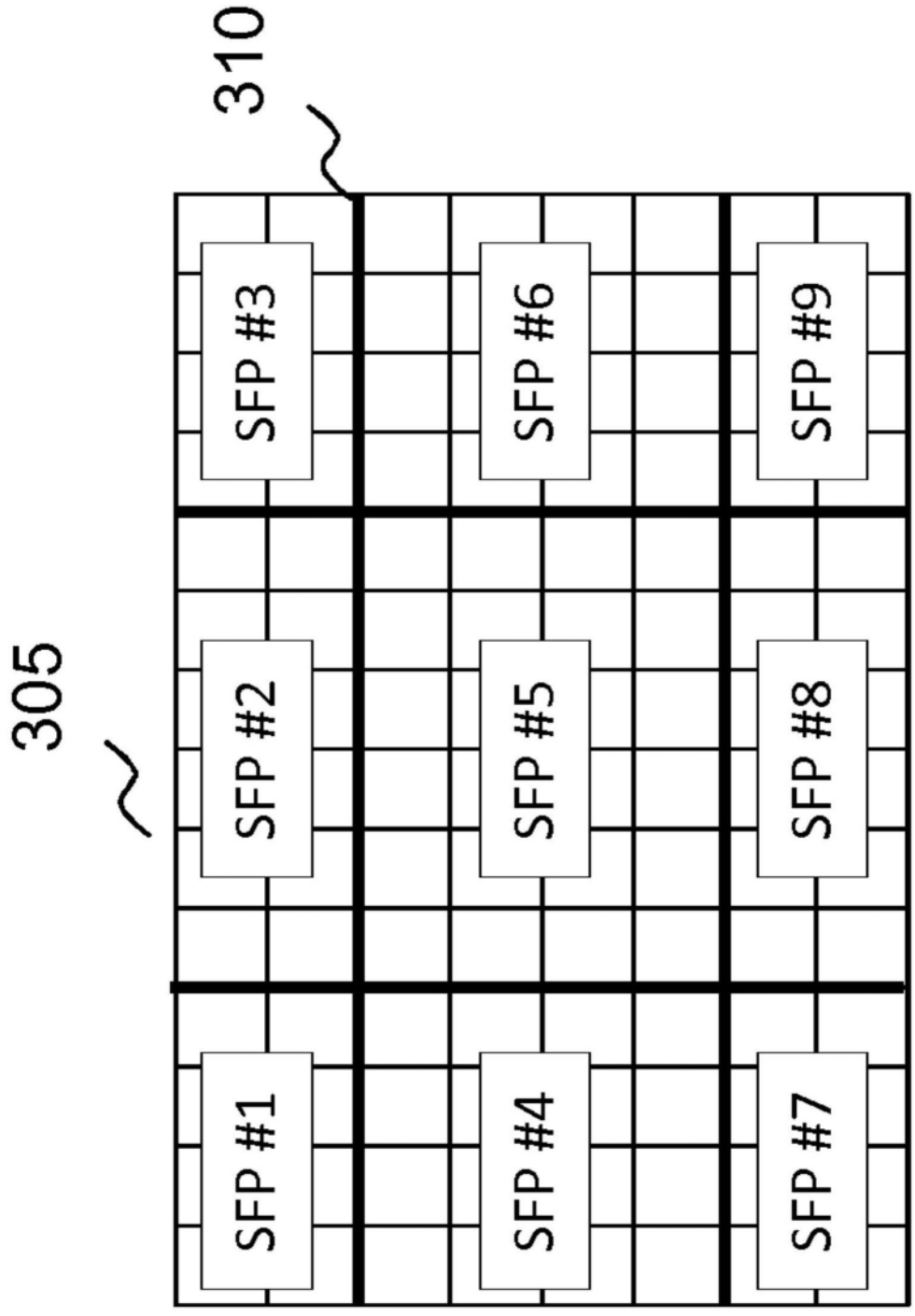


图3

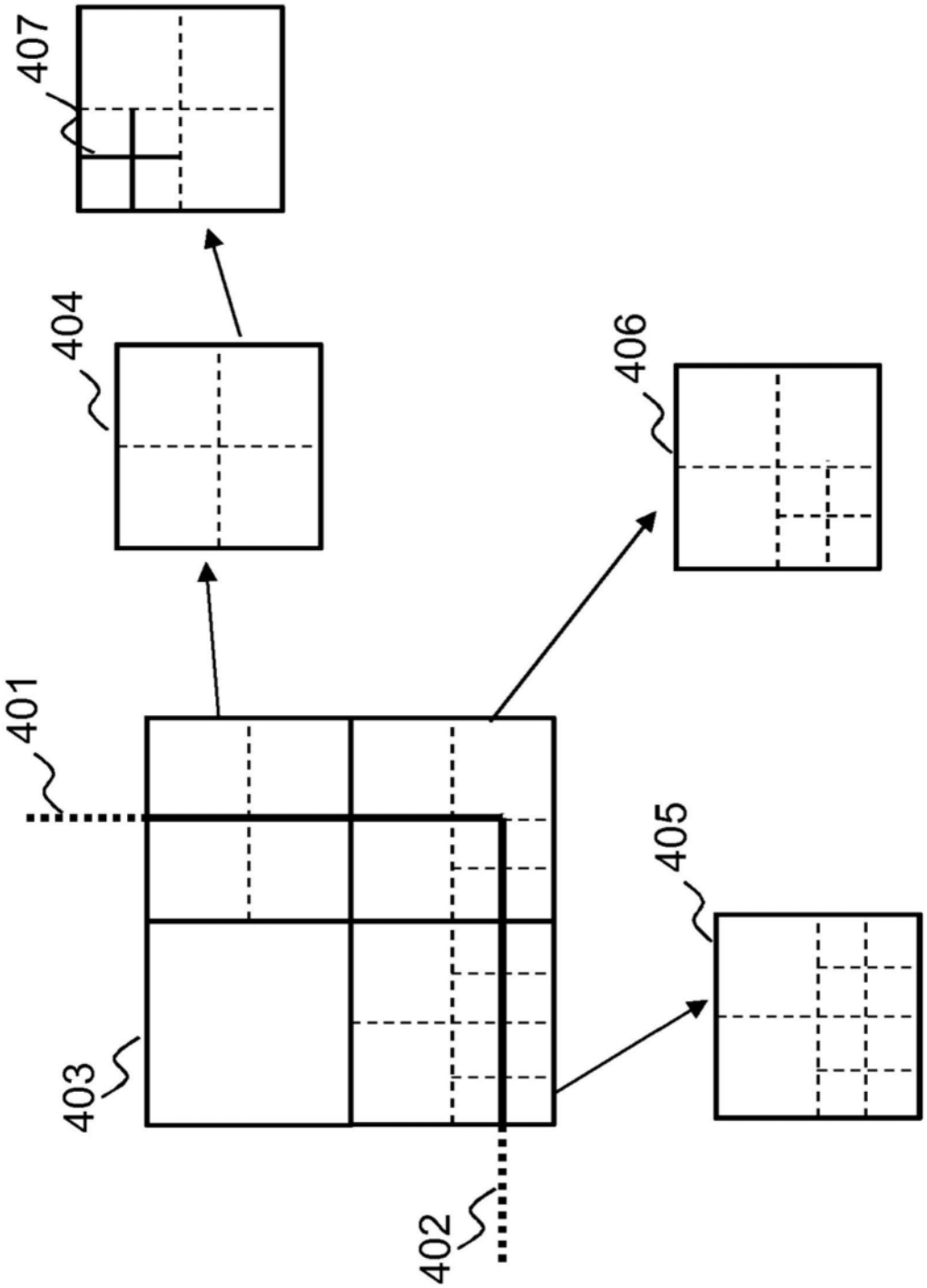


图4

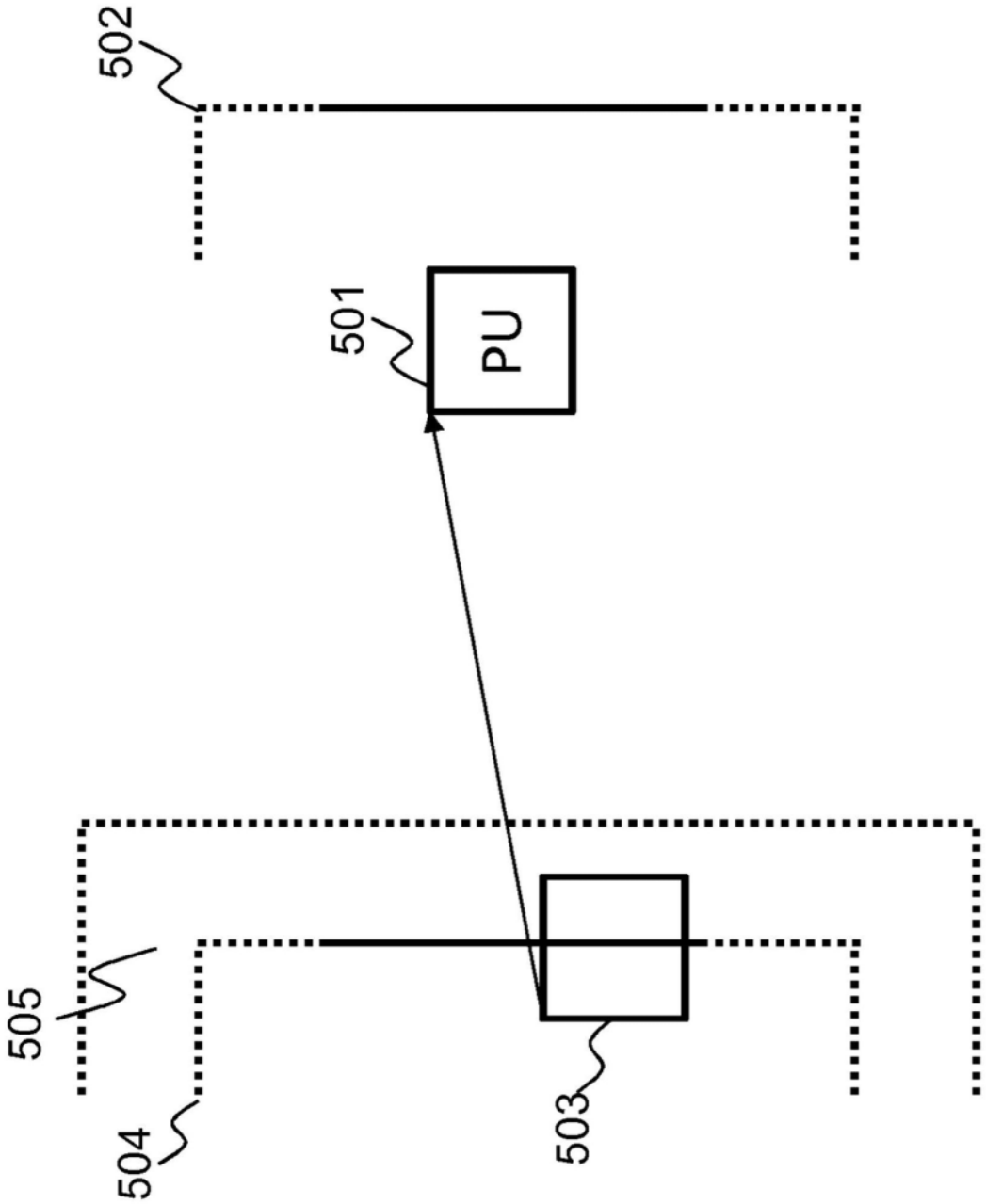


图5

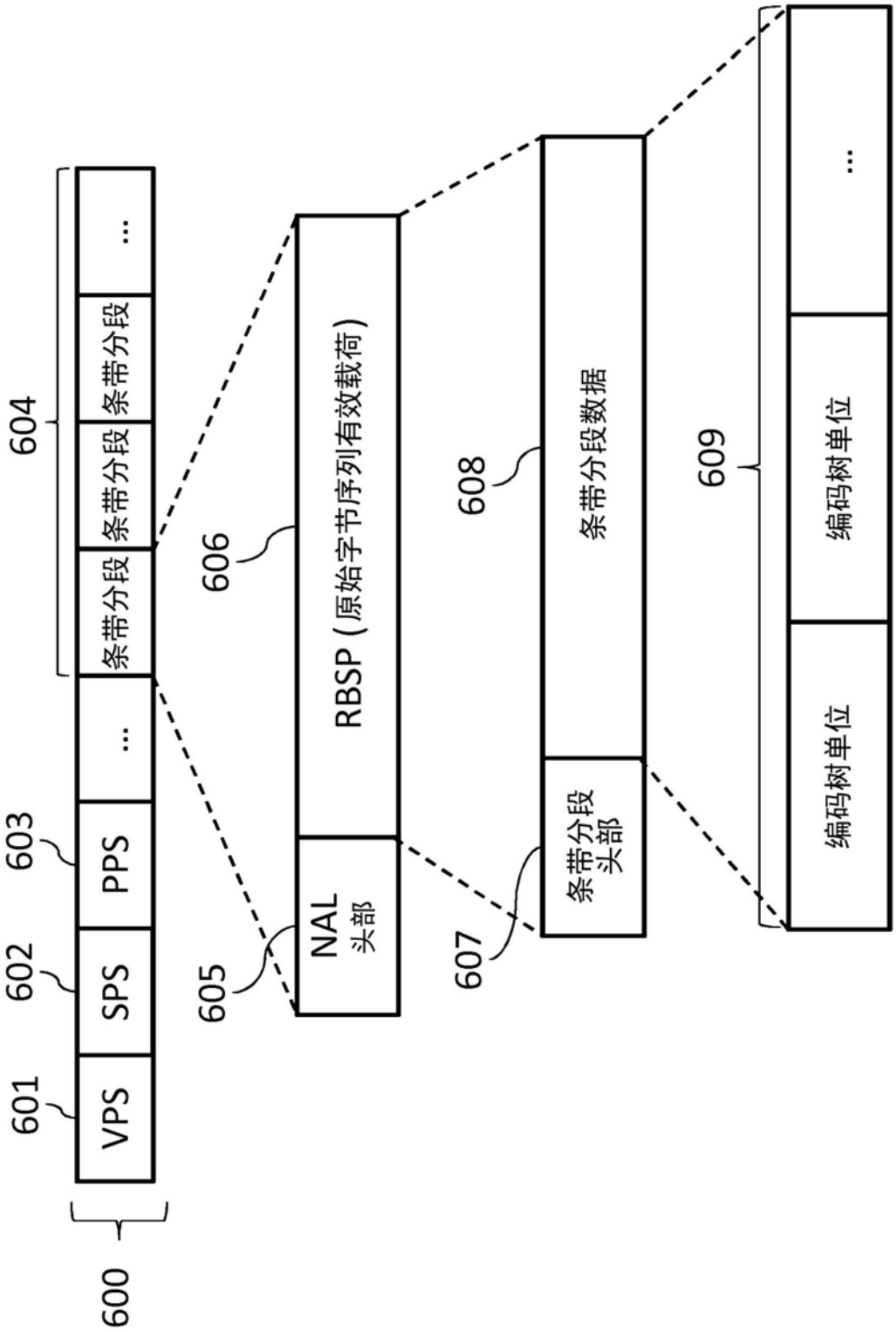

600
601
VPS
602
SPS
603
PPS
...
604
条带分段
条带分段
条带分段
...
605
NAL
头部
606
RBSP (原始字节序列有效载荷)
607
条带分段
头部
608
条带分段数据
609
编码树单位
编码树单位
...

图6

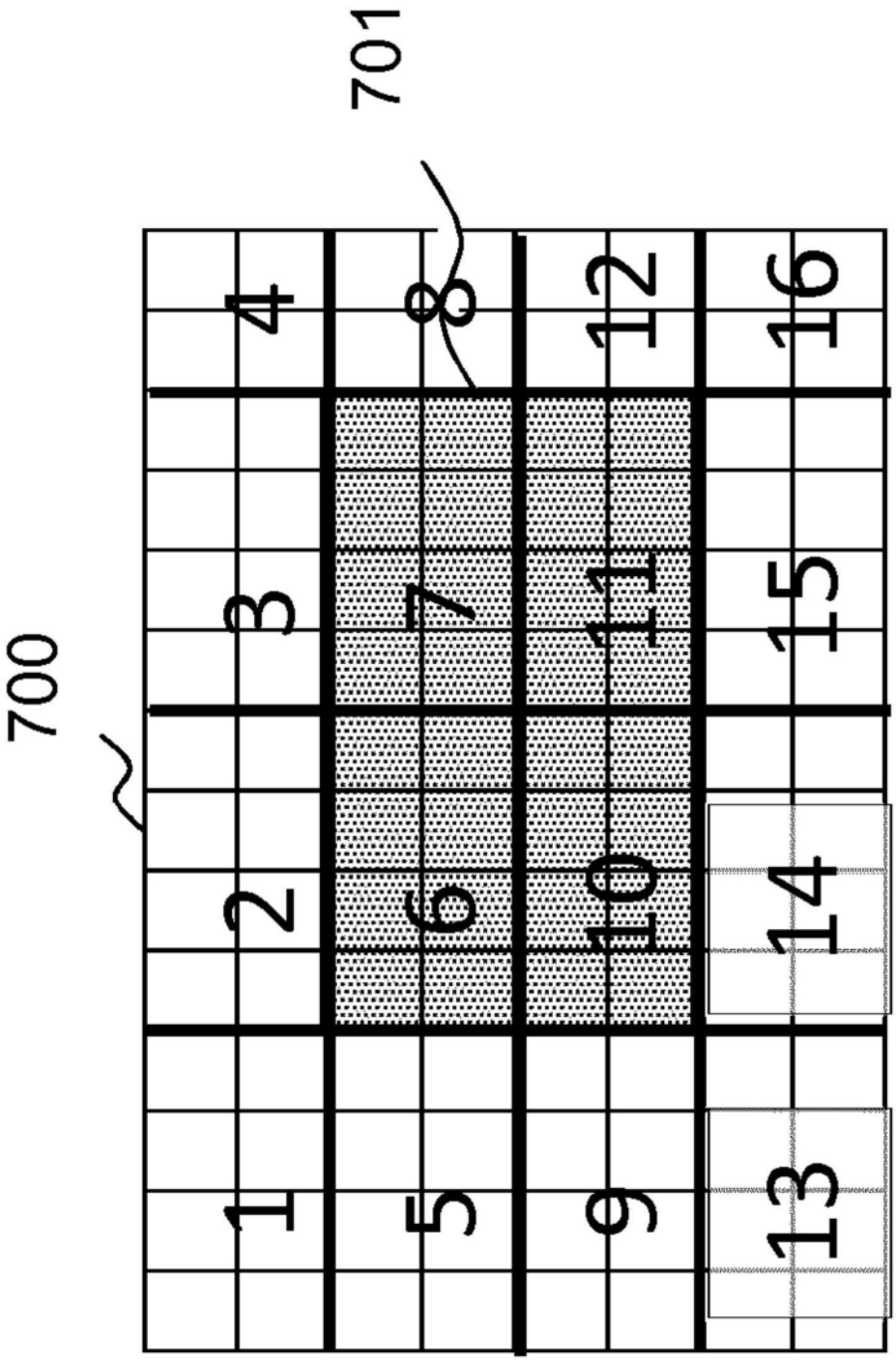


图7

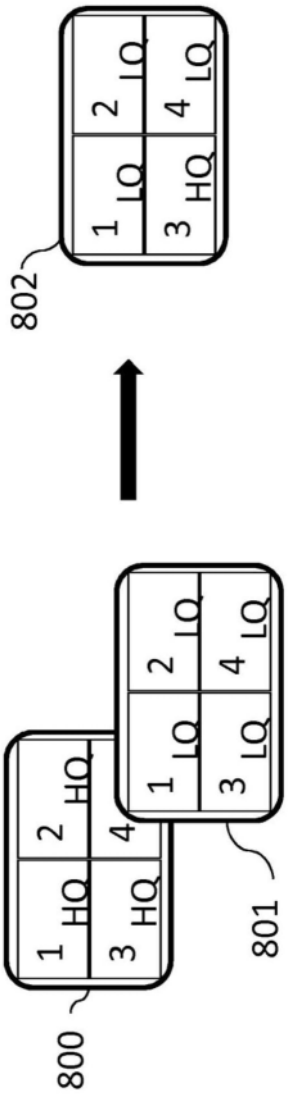


图8a

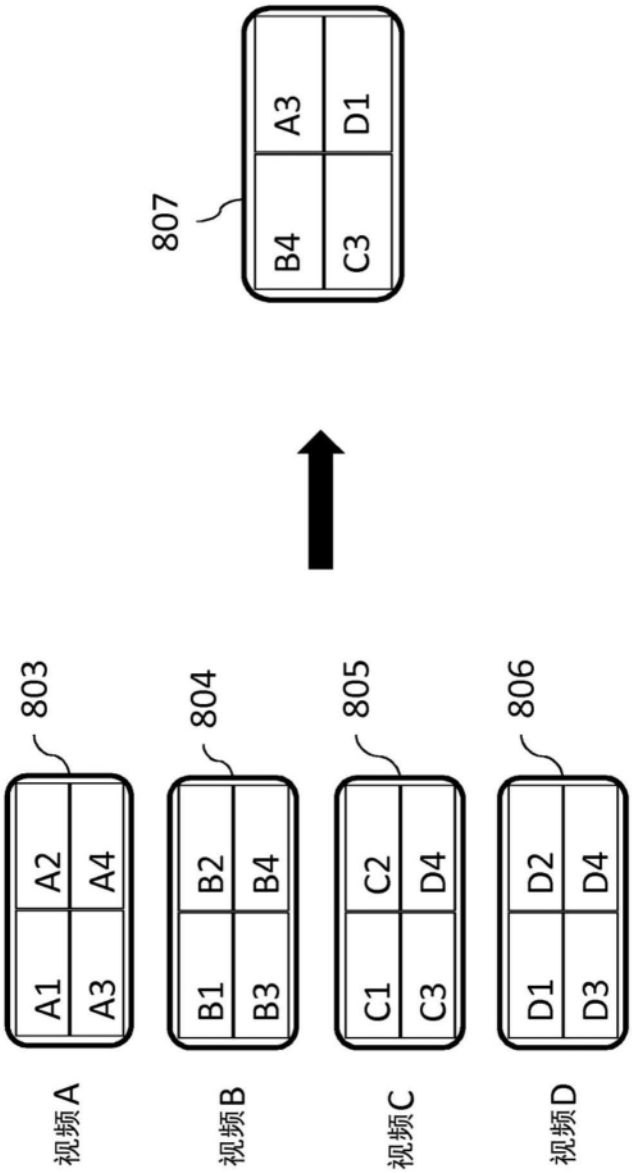

807
A3
D1
B4
C3
803
804
805
806
A2
A4
A1
A3
B2
B4
B1
B3
C2
D4
C1
C3
D2
D4
D1
D3
视频A
视频B
视频C
视频D

图8b

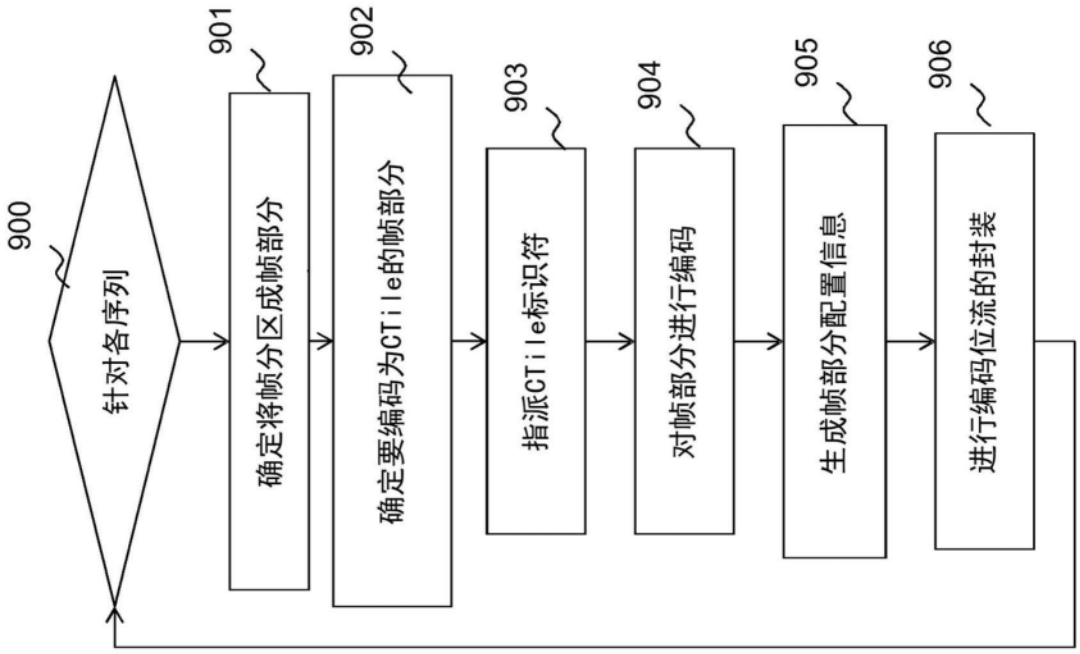

900
针对各序列
901
确定将帧分区成帧部分
902
确定要编码为CTile的帧部分
903
指派CTile标识符
904
对帧部分进行编码
905
生成帧部分配置信息
906
进行编码位流的封装

图9

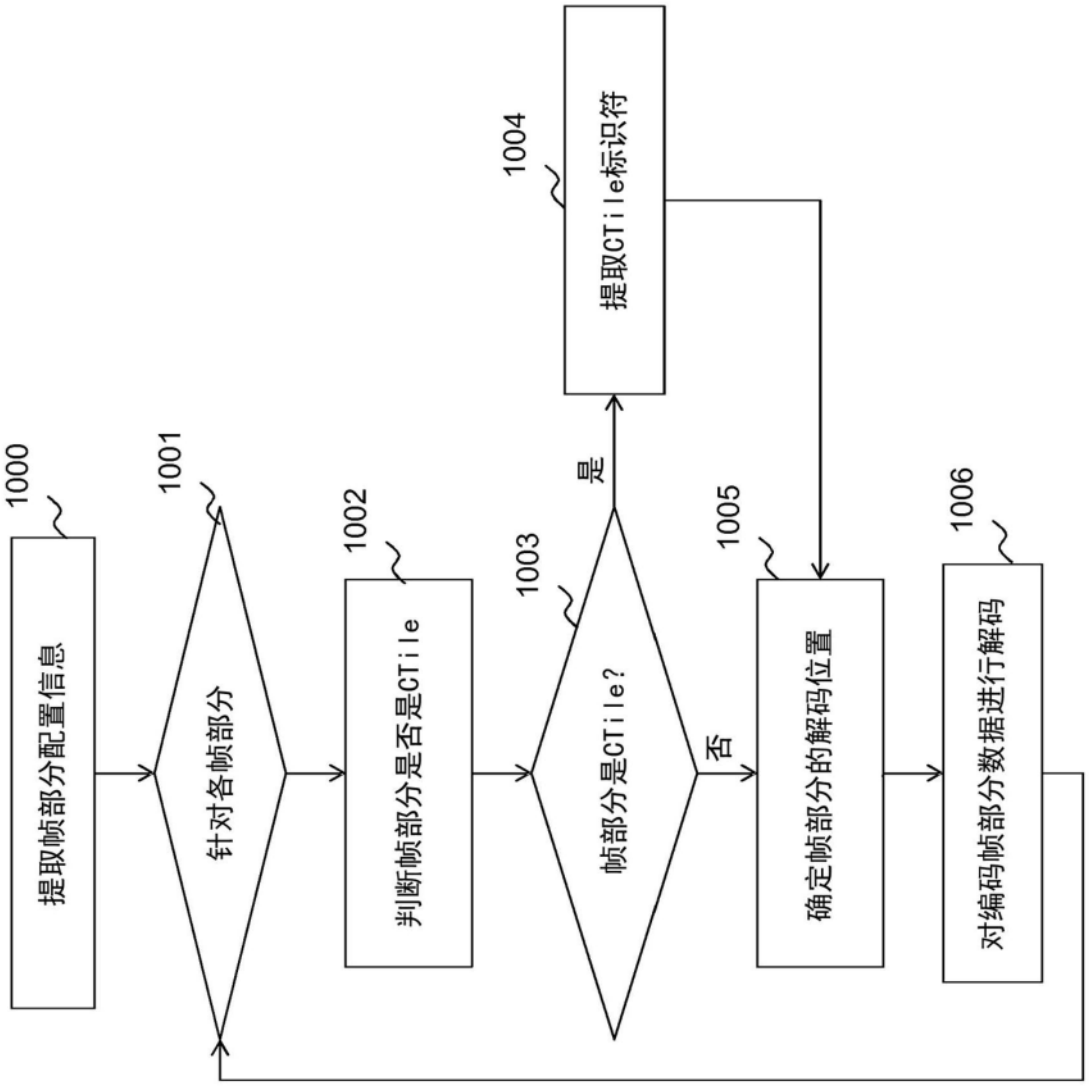

1000
提取帧部分配置信息
1001
针对各帧部分
1002
判断帧部分是否是CTile
1003
帧部分是CTile?
是
1004
提取CTile标识符
否
1005
确定帧部分的解码位置
1006
对编码帧部分数据进行解码

图10

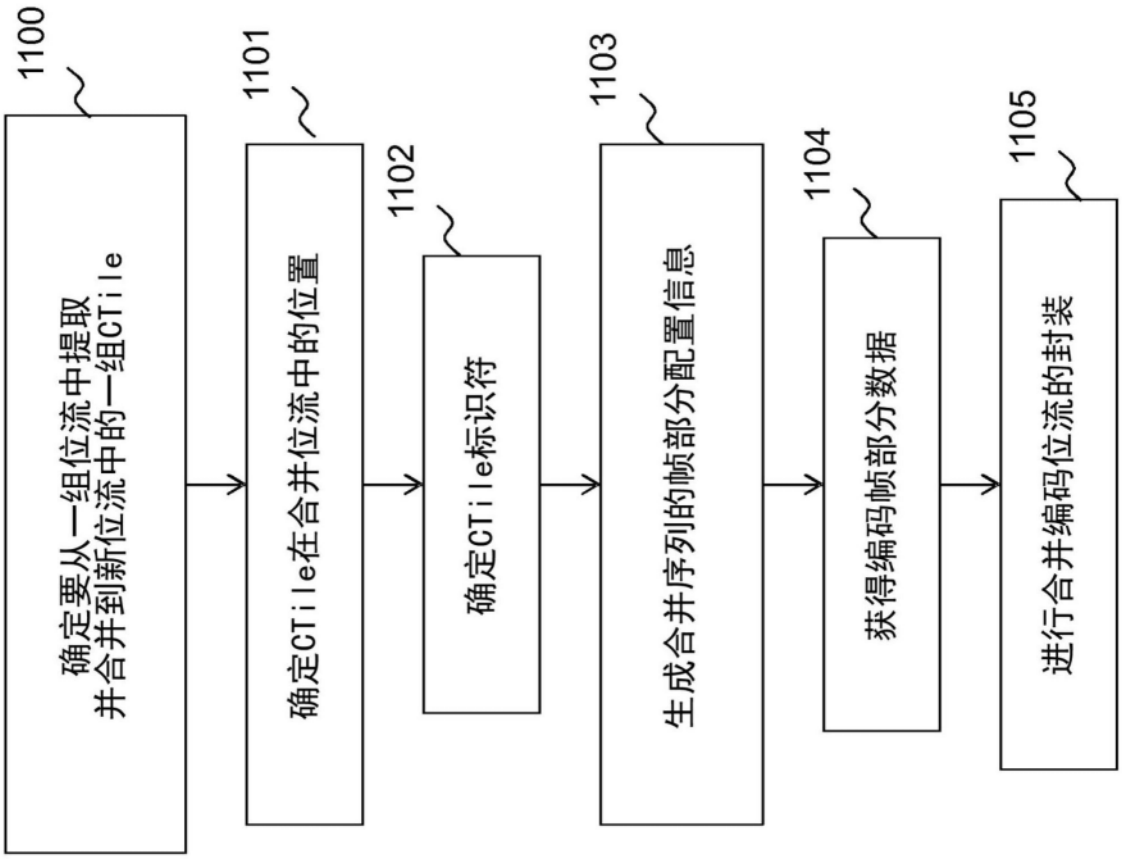

1100
确定要从一组位流中提取并合并到新位流中的一组CTile
1101
确定CTile在合并位流中的位置
1102
确定CTile标识符
1103
生成合并序列的帧部分配置信息
1104
获得编码帧部分数据
1105
进行合并编码位流的封装

图11

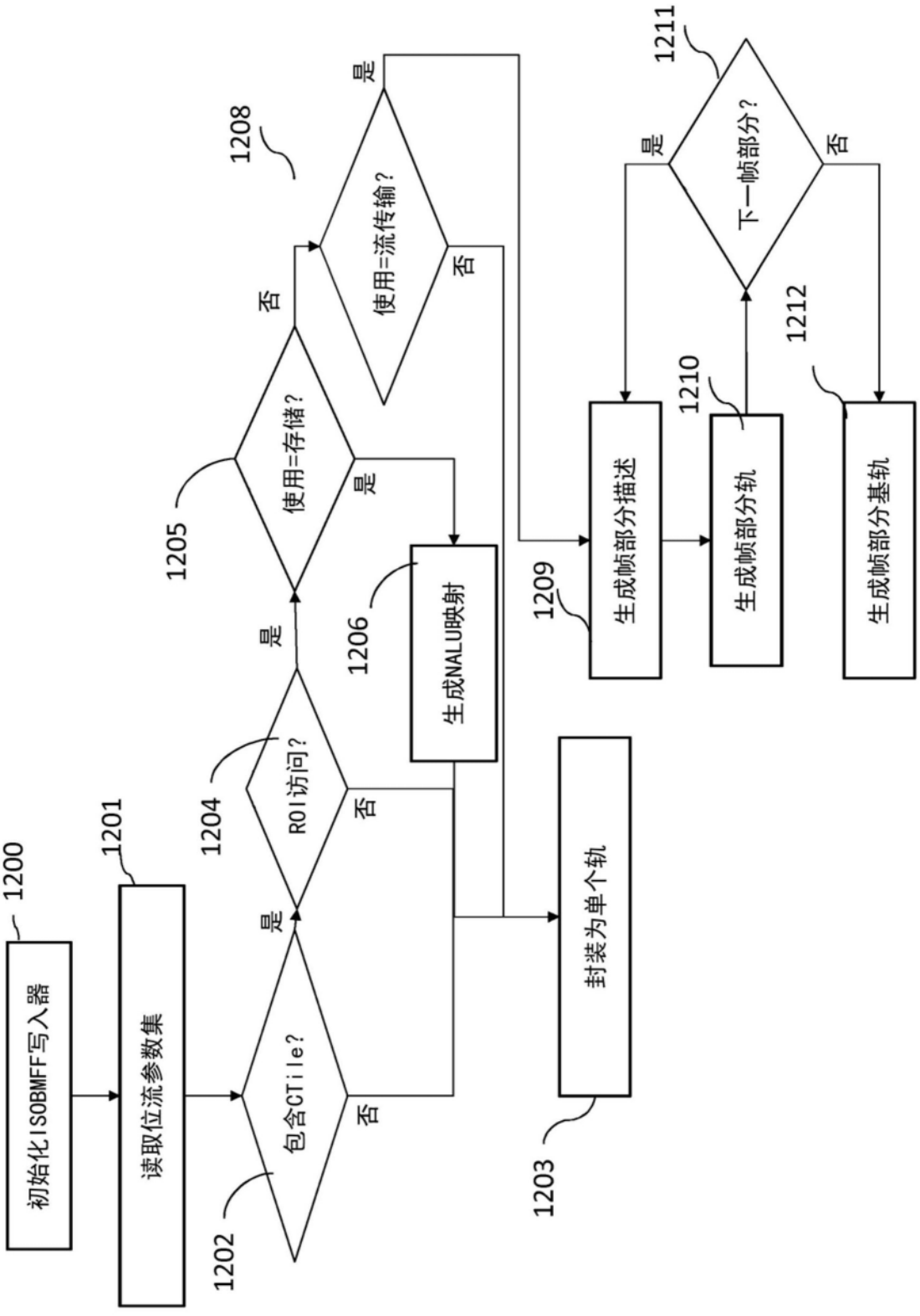

1200
初始化ISOBMFF写入器
1201
读取位流参数集
1202
包含CTile?
1203
封装为单个轨
1204
ROI访问?
1205
使用=存储?
1206
生成NALU映射
1208
使用=流传输?
1209
生成帧部分描述
1210
生成帧部分轨
1211
下一帧部分?
1212
生成帧部分基轨
是
否

图12

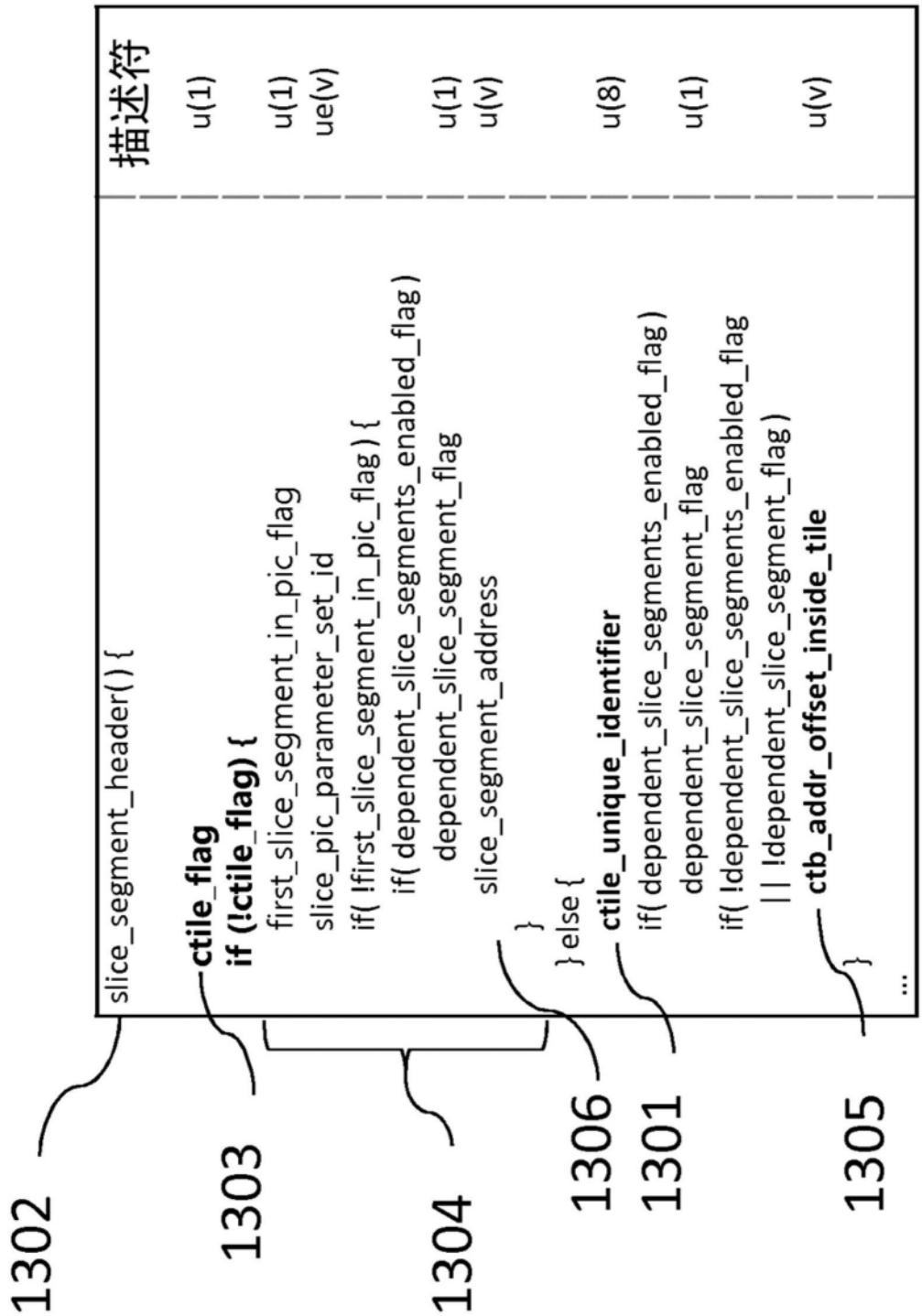

描述符
slice_segment_header( ) {
ctile_flag u(1)
if (!ctile_flag) {
first_slice_segment_in_pic_flag u(1)
slice_pic_parameter_set_id ue(v)
if(!first_slice_segment_in_pic_flag) {
if(dependent_slice_segments_enabled_flag)
dependent_slice_segment_flag u(1)
slice_segment_address u(v)
}
} else {
ctile_unique_identifier u(8)
if(dependent_slice_segments_enabled_flag)
dependent_slice_segment_flag u(1)
if(!dependent_slice_segments_enabled_flag
|| !dependent_slice_segment_flag)
ctb_addr_offset_inside_tile u(v)
}
...
1302
1303
1304
1306
1301
1305

图13a

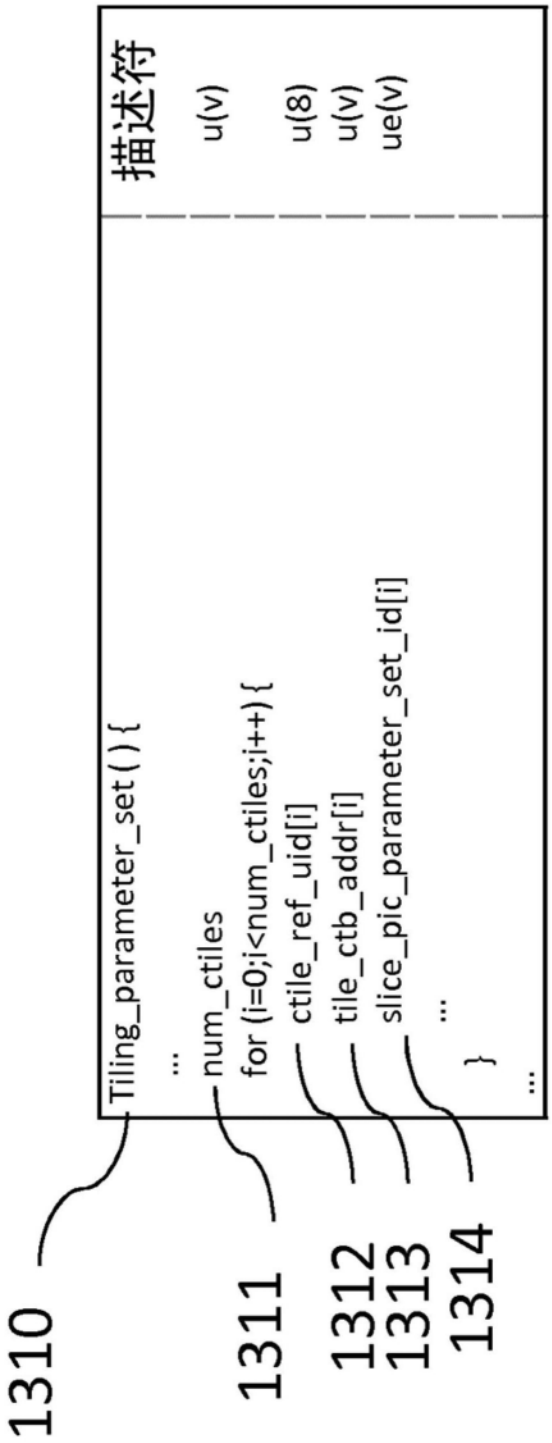

描述符
Tiling_parameter_set ( ) {
...
num_ctiles
for (i=0;i<num_ctiles;i++) {
ctile_ref_uid[i]
tile_ctb_addr[i]
slice_pic_parameter_set_id[i]
...
}
...
u(v)
u(8)
u(v)
ue(v)
1310
1311
1312
1313
1314

图13b

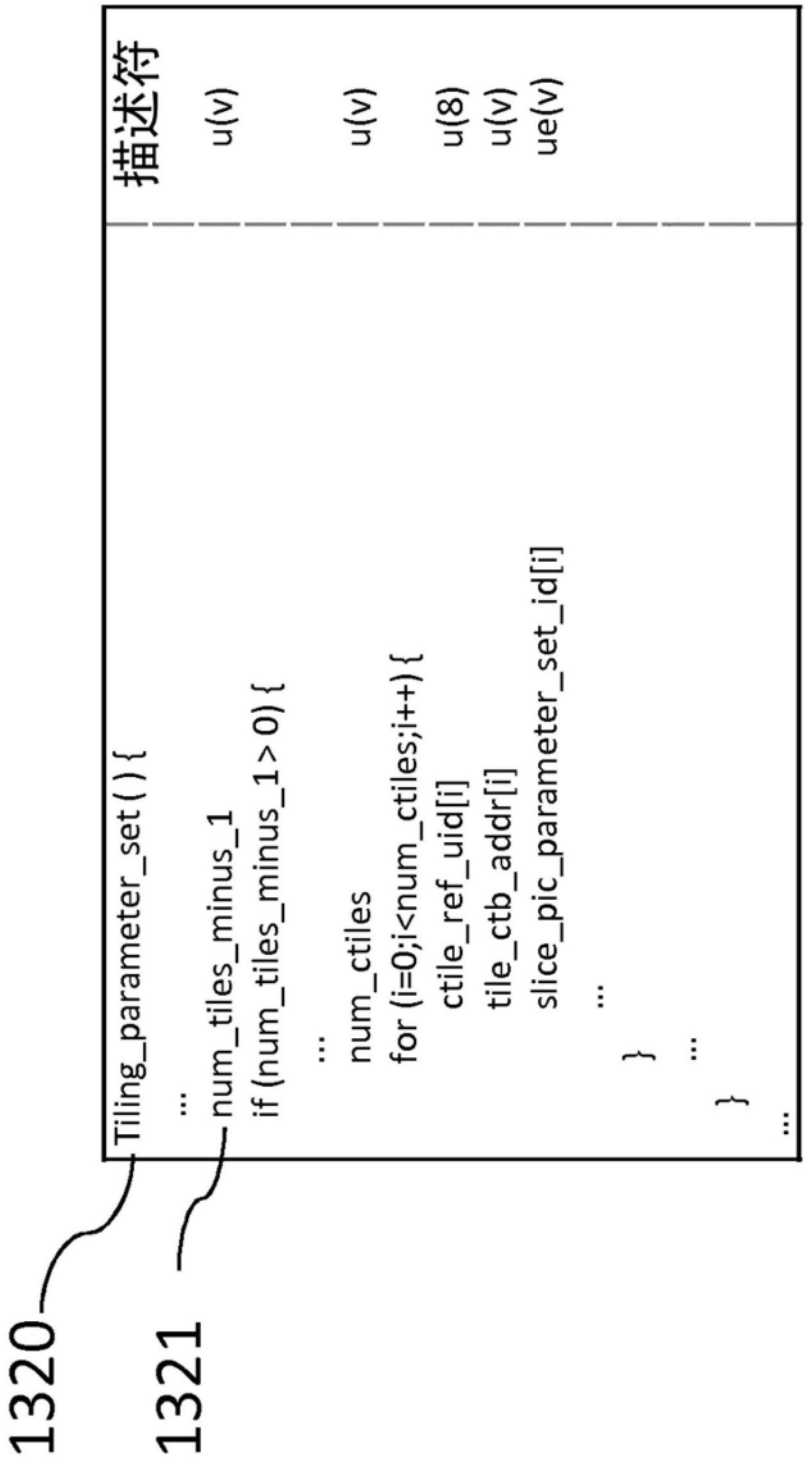

描述符
Tiling_parameter_set ( ) {
...
num_tiles_minus_1
if (num_tiles_minus_1 > 0) {
...
num_ctiles
for (i=0;i<num_ctiles;i++) {
ctile_ref_uid[i]
tile_ctb_addr[i]
slice_pic_parameter_set_id[i]
...
}
...
}
...
u(v)
u(v)
u(8)
u(v)
ue(v)
1320
1321

图13c

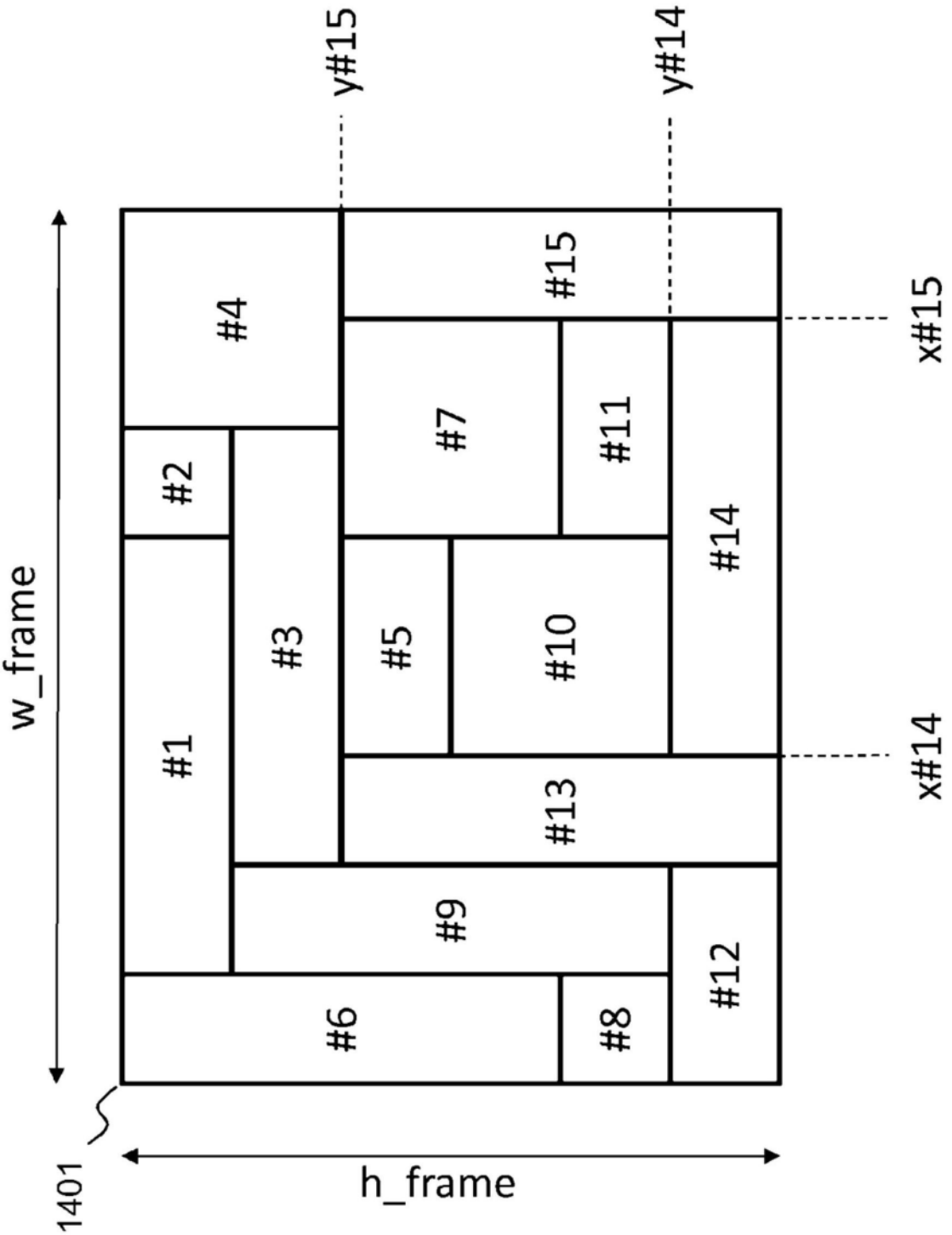


图14

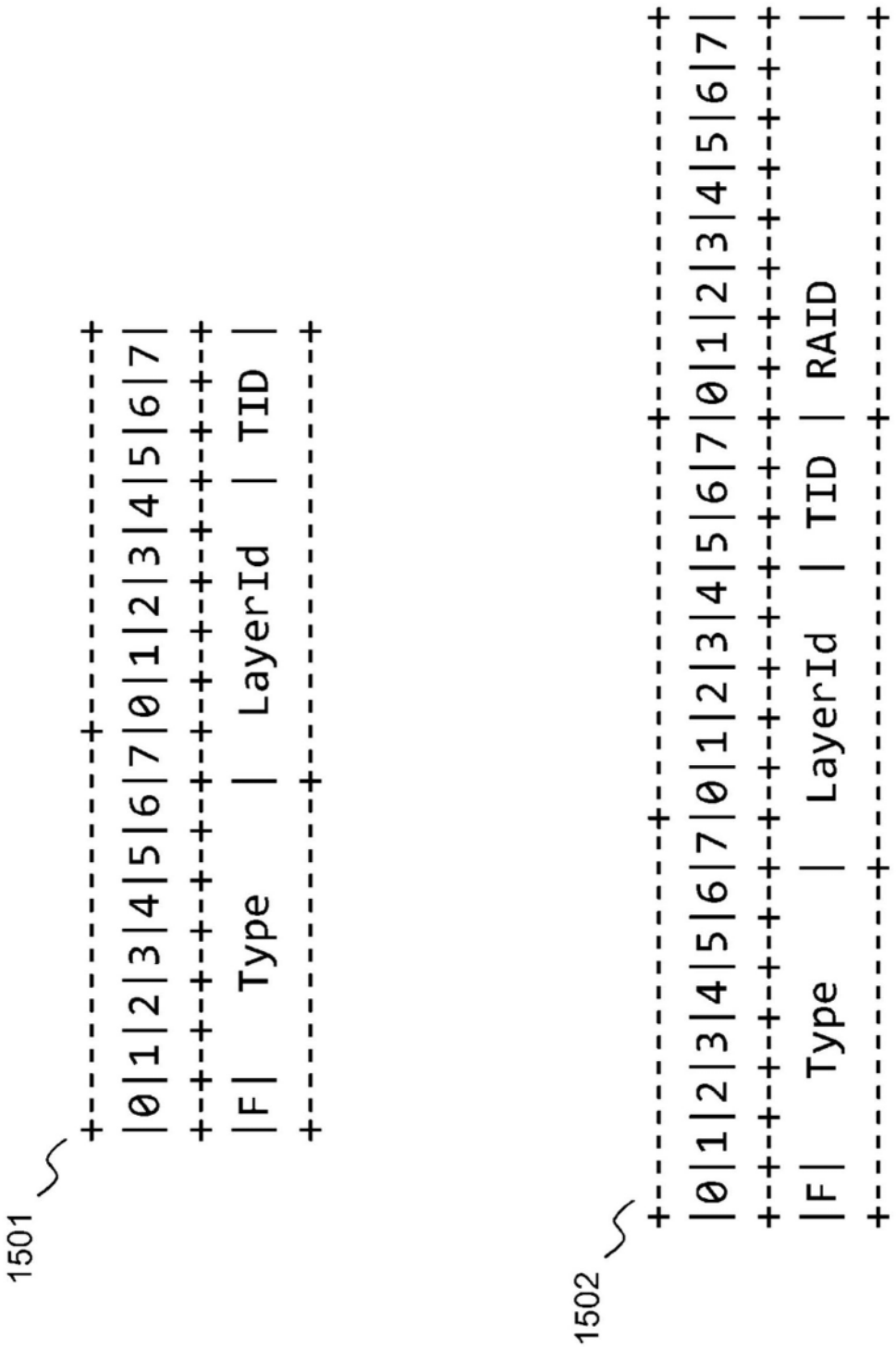

1501
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|0|1|2|3|4|5|6|7|0|1|2|3|4|5|6|7|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|F| Type | LayerId | TID |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
1502
+---+
|0|1|2|3|4|5|6|7|0|1|2|3|4|5|6|7|0|1|2|3|4|5|6|7|
+---+
|F| Type | LayerId | TID | RAID |
+---+

图15

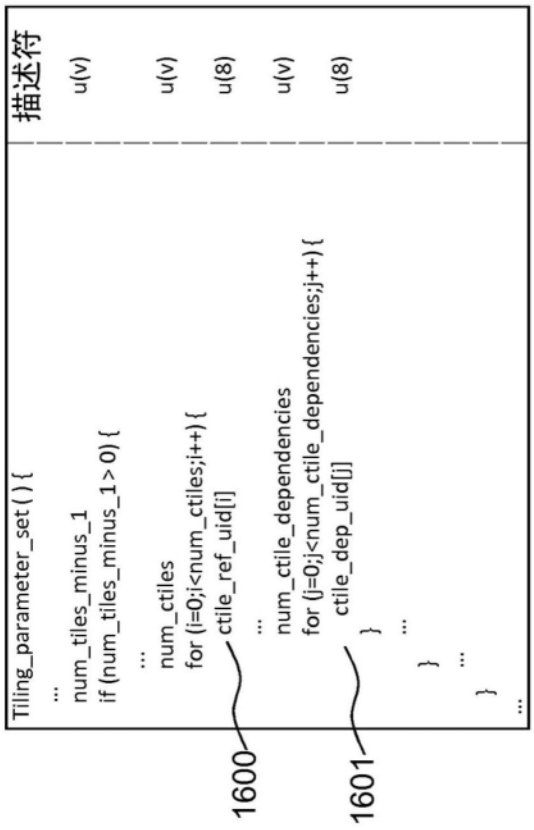

描述符
Tiling_parameter_set( ) {
...
num_tiles_minus_1 u(v)
if (num_tiles_minus_1 > 0) {
...
num_ctiles u(v)
for (i=0;i<num_ctiles;i++) {
ctile_ref_uid[i] u(8)
...
num_ctile_dependencies u(v)
for (j=0;j<num_ctile_dependencies;j++) {
ctile_dep_uid[j] u(8)
}
...
}
...
}
...
1600
1601

图16a

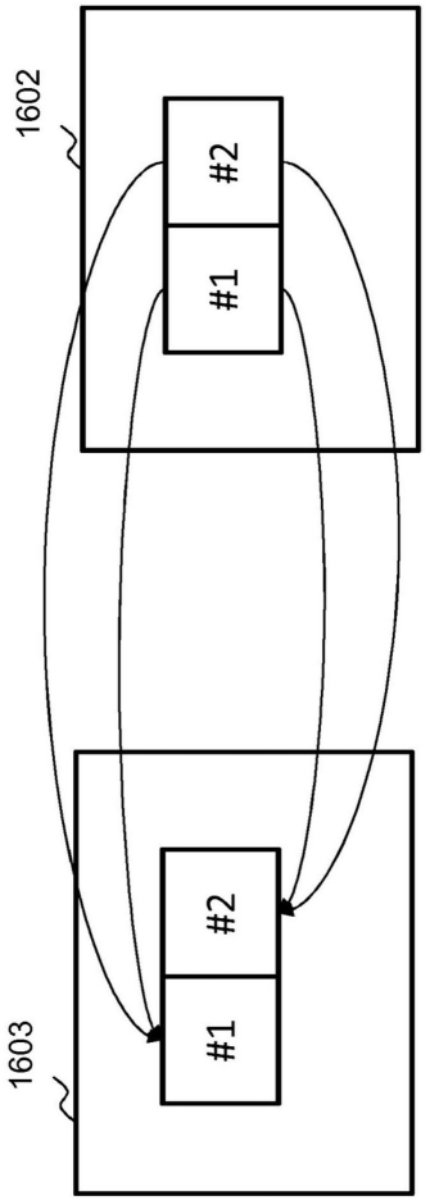


图16b

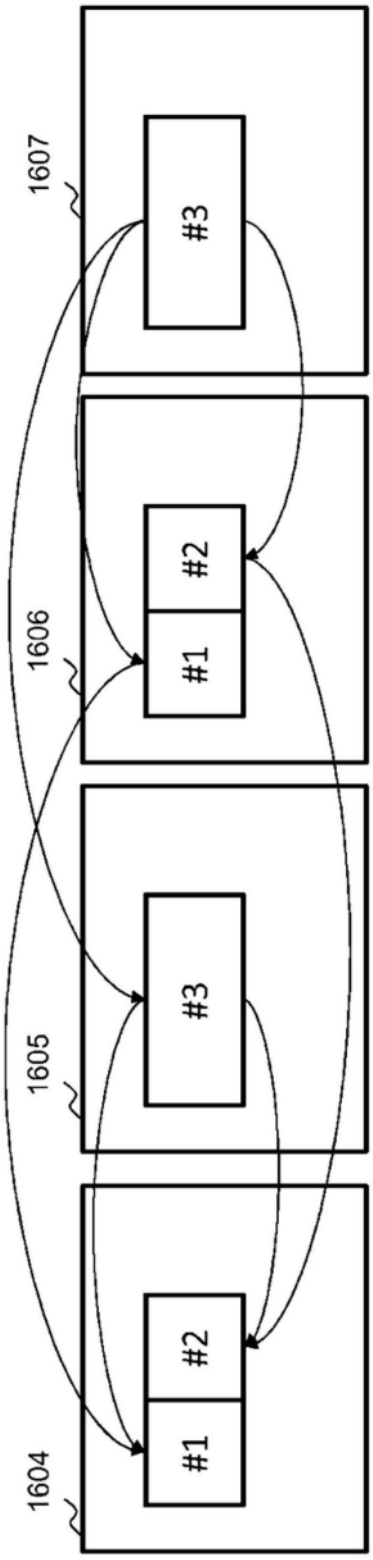


图16c

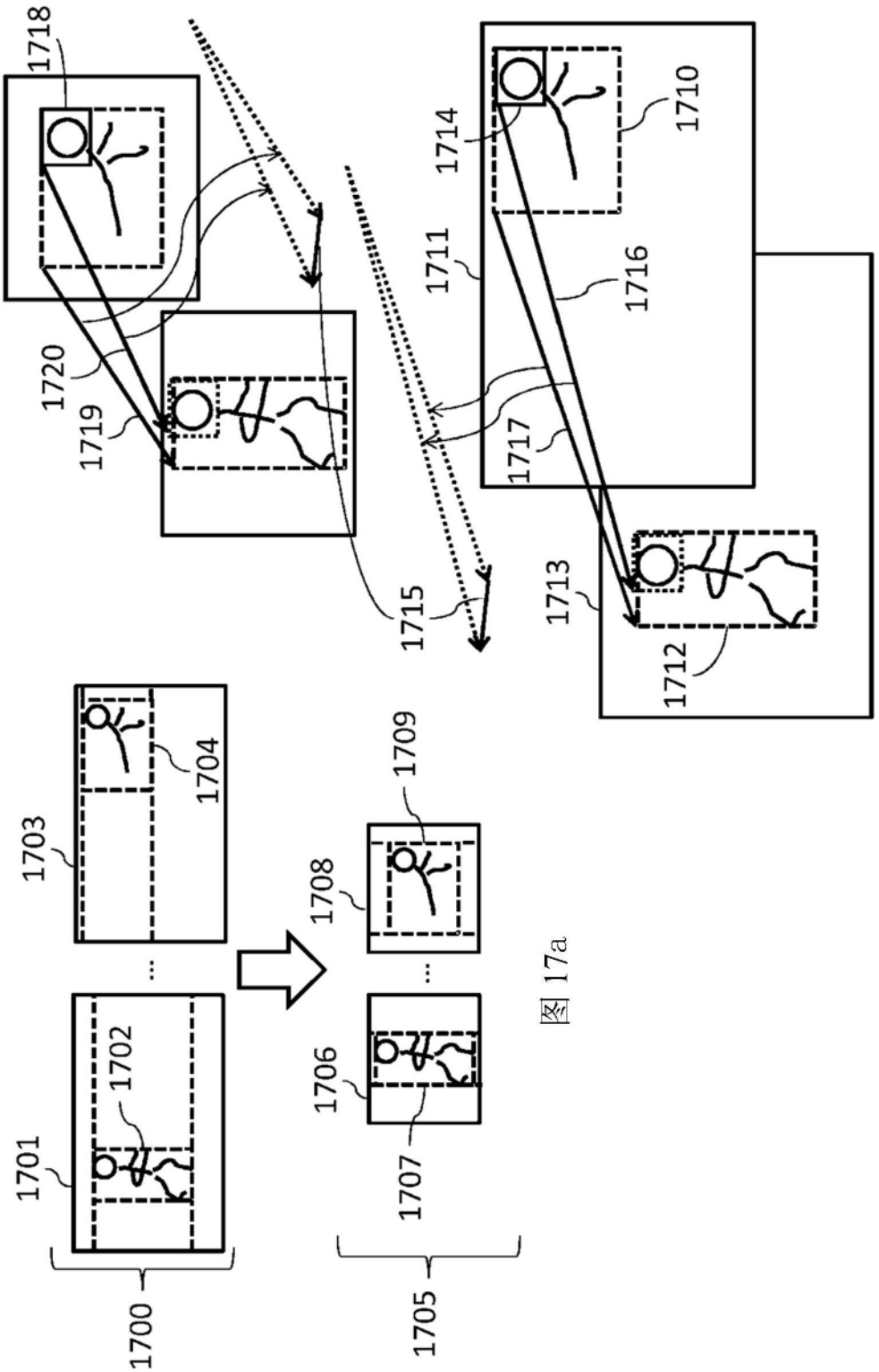

1718
1720
1719
1715
1714
1710
1711
1716
1717
1713
1712
1703
1704
1708
1709
1701
1702
1706
1707
1700
1705

图 17a

图 17b

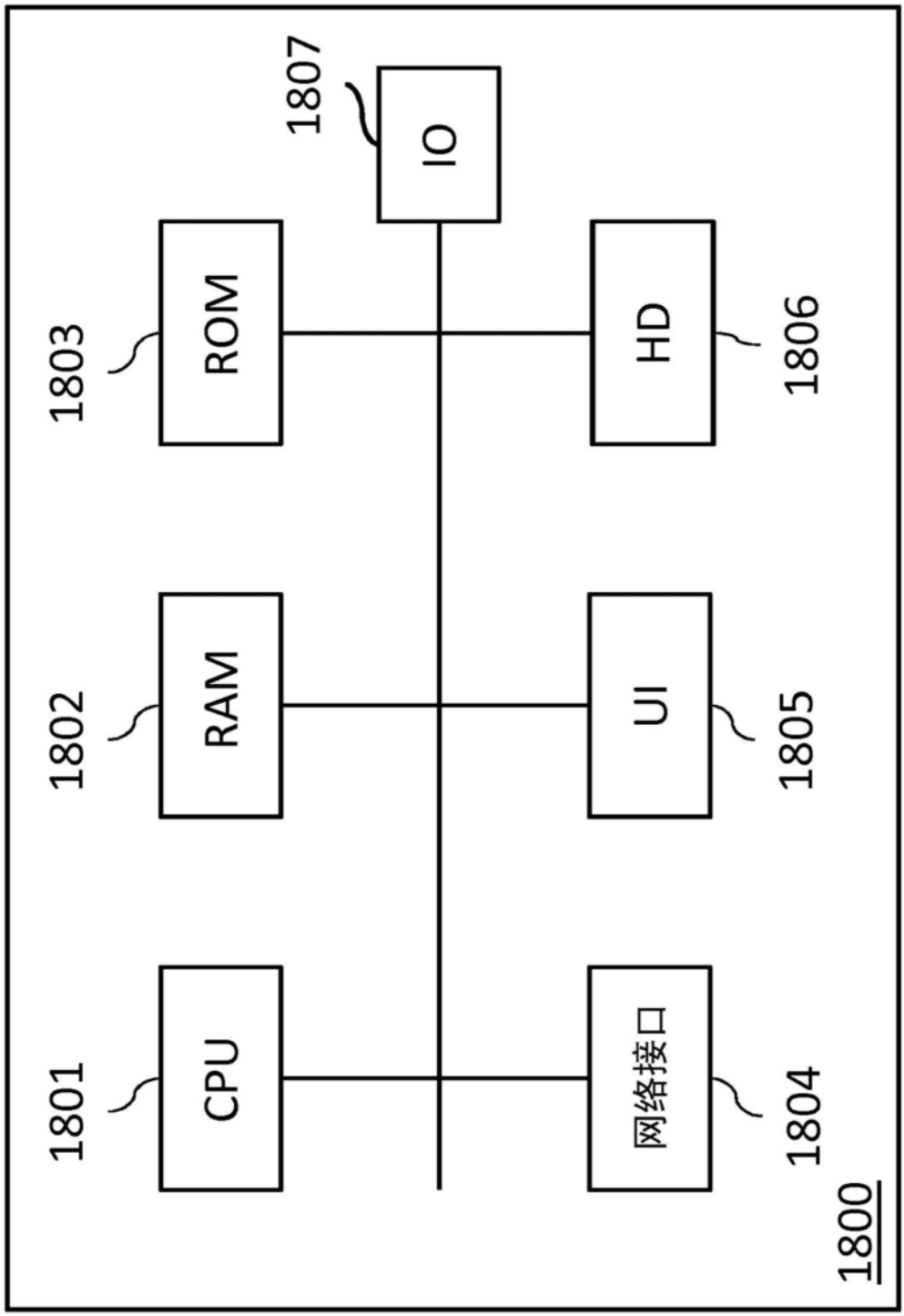

1807
IO
1803
ROM
HD
1806
1802
RAM
UI
1805
1801
CPU
网络接口
1804
1800

图18