(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 105637876 B
(45)授权公告日 2019.04.02

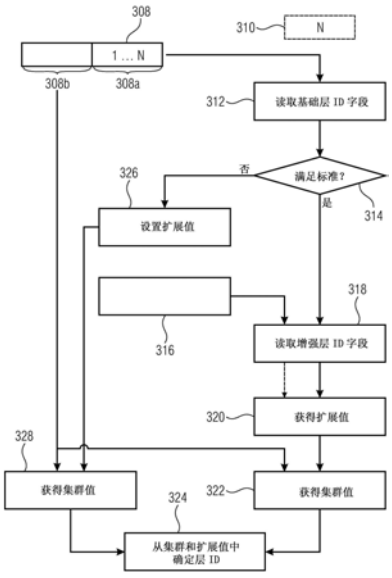
(21)申请号 201480050666.6
(22)申请日 2014.07.15
(65)同一申请的已公布的文献号
申请公布号 CN 105637876 A
(43)申请公布日 2016.06.01
(30)优先权数据
61/846,479 2013.07.15 US
(85)PCT国际申请进入国家阶段日
2016.03.14
(86)PCT国际申请的申请数据
PCT/EP2014/065181 2014.07.15
(87)PCT国际申请的公布数据
WO2015/007750 EN 2015.01.22
(73)专利权人 GE视频压缩有限责任公司
地址 美国纽约
(72)发明人 卡斯滕·聚林 托马斯·席尔 德特勒夫·马佩 罗伯特·斯库平 亚戈·桑切斯 德拉富恩特 格哈德·特克
(74)专利代理机构 北京康信知识产权代理有限责任公司 11240
代理人 梁丽超 陈鹏
(51)Int.Cl.
H04N 19/65(2006.01)
H04N 19/70(2006.01)
(56)对比文件
US 2009125636 A1,2009.05.14,
CN 101317460 A,2008.12.03,
Karsten Suehring.MV-HEVC/SHVC HLS: Extending the supported number of layers.《Joint Collaborative Team on 3D Video Coding Extensions of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 4th Meeting: Incheon, KR》.2013,
审查员 刘冰洁
权利要求书3页 说明书22页 附图19页

(54)发明名称

处理多层数据流的装置和方法、编码器和方法及存储介质

(57)摘要

本申请涉及处理多层数据流的装置和方法、编码器和方法及存储介质。描述了在多层数据流的层之间的层间依赖性的信号作用。通过表示在基础层ID能表示的几对不同值之间的相互依赖性的第一相互依赖性语法结构以及表示在扩展层ID能表示的几对不同值之间的相互依赖性的第二相互依赖性语法结构的方式描述层间依赖性,发现在一方面的层间依赖性的潜在多样性的太密集限制与另一方面的层间依赖性的太复杂信令之间的良好折衷,基础层ID和扩展层ID索引与多层数据流的部分相关联的层。根据该概念,可在一方面的可信令层间依赖性的增加的多样性与另一方面的用于信令层间依赖性的减小的边信息开销之间转移重点。



CN 105637876 B

1.一种被配置为处理多层数据流(10)的装置,使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入所述多层数据流,所述多层数据流包括多个数据包(12),每个数据包与所述不同层中的一层相关联,每层由基础层ID(20)、或基础层ID(20)和扩展层ID(22)索引,所述装置配置为:

从所述多层数据流中读取

第一相互依赖性语法结构(24),表示在所述基础层ID(20)能表示的几对不同值之间的相互依赖性,以及

第二相互依赖性语法结构(26),表示在所述扩展层ID(22)能表示的几对不同值之间的相互依赖性,并且

基于所述第一相互依赖性语法结构和第二相互依赖性语法结构,构造显示在所述不同层之间的可能层间预测依赖性的层间依赖性描述(14),

其中,所述第一相互依赖性语法结构(24)限定在不同层组之间的相互依赖性,属于一个组的所有的层具有相同的基础层ID,并且所述第二相互依赖性语法结构(26)限定在表示为由第一相互依赖性语法结构互连几对组中的层之间的相互依赖性。

2.根据权利要求1所述的装置,其中,所述装置配置为:

通过有条件地根据表示在对(j,k)之间存在或不存在相互依赖性的所述第一相互依赖性语法结构,从用于对(j,k)的多层数据流中,读取或者抑制读取所述第二相互依赖性结构(26),以

从所述多层数据流中读取所述第二相互依赖性结构(26)的多个实例化(34),即用于所述基础层ID能表示的值$0<k\leq j\leq n<N$的不同几对(j,k),N是所述基础层ID能表示的值的数量,

其中,为对j=k的(j,k)读取的所述第二相互依赖性结构(26)的实例化(34)表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID j的层之中的其他层的层间预测,并且为对$k<j$的(j,k)读取的所述第二相互依赖性结构(26)的实例化(34)表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID k的层之中的层的层间预测。

3.根据权利要求1所述的装置,其中,所述装置配置为:

从所述多层数据流中读取所述第二相互依赖性结构(26)的一个实例化,N是所述基础层ID能表示的值的数量,所述第二相互依赖性结构(26)的所述一个实例化分别表示

通常对于所述基础层ID能表示的每个值$0<i\leq n<N$,在具有基础层ID i的层之中的哪些层是依赖于在具有基础层ID i的层之中的其他层的层间预测,或者

通常对于$0<k<j\leq n<N$的一组对(j,k)中的每个,在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID k的层之中的层的层间预测。

4.根据权利要求1所述的装置,其中,所述装置配置为:

从所述多层数据流中读取所述第二相互依赖性结构(26)的实例化,以及对于所述第二相互依赖性结构(26)的每个实例化,读取使所述第二相互依赖性结构(26)的各实例化与所述基础层ID能表示的值$0<k\leq j\leq n<N$的对(j,k)的各子集相关联的关联语法结构,N是所述基础层ID能表示的值的数量。

5.根据权利要求4所述的装置,其中,所述关联语法结构包括用于每对(j,k)的标志,并且所述装置配置为:

通过跳过值0<k≤j≤n<N的几对(j,k)，解析所述关联语法结构，其中，所述第一相互依赖性语法结构表示在对(j,k)之间不存在相互依赖性。

6.根据权利要求5所述的装置，其中，所述扩展层ID(22)能表示的值的数量是M，并且所述第二相互依赖性语法结构包括

用于扩展值为0<q<p≤m<M的每对(p,q)的一个二进制值，以便表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID j的层之中的其他层的层间预测，以及

用于扩展值为0<p,q≤m<M的每对(p,q)的一个二进制值，以便表示具有扩展值p并且具有基础层ID j的哪个层是依赖于具有扩展值q并且具有基础层ID k<j的层的层间预测。

7.根据权利要求6所述的装置，其中，所述装置配置为在数据流中从在所述基础层ID能表示的N值之中的最大假定值的明确信令中获得n，和/或在数据流中从在所述扩展层ID能表示的M值之中的最大假定值的明确信令中获得m。

8.根据权利要求1所述的装置，其中，所述装置配置为基于所述第一相互依赖性语法结构和数据包的基础层ID，丢弃以层组为单位的数据包，每个数据包具有共同的基础层ID。

9.根据权利要求8所述的装置，配置为基于所述第一相互依赖性语法结构和第二相互依赖性语法结构和数据包的基础层ID以及数据包的扩展层ID，丢弃进一步的单独层。

10.根据权利要求1所述的装置，其中，所述装置是

视频解码器，配置为使用所述层间预测和所述层间依赖性矩阵，解码所述多层数据流，或者

网络元件，配置为基于所述层间依赖性矩阵，丢弃所述多层数据流的数据包。

11.根据权利要求1所述的装置，其中，所述多层数据流具有使用层间预测在不同层上编码入所述多层数据流的视频材料，以便从另一层中层间预测的任何层增加以下中的一个或多个：

进一步视图；

深度信息；

阿尔法混合信息；

颜色分量信息；

空间分辨率细化；以及

SNR分辨率细化。

12.一种用于处理多层数据流(10)的方法，使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入所述多层数据流，所述多层数据流包括多个数据包(12)，每个所述数据包与所述不同层中的一层相关联，每层由基础层ID(20)、或基础层ID(20)和扩展层ID(22)索引，所述方法包括：

从所述多层数据流中读取

第一相互依赖性语法结构(24)，表示在所述基础层ID(20)能表示的几对不同值之间的相互依赖性，以及

第二相互依赖性语法结构(26)，表示在所述扩展层ID(22)能表示的几对不同值之间的相互依赖性，并且

基于所述第一相互依赖性语法结构和第二相互依赖性语法结构，构造显示在所述不同层之间的层间预测依赖性的层间依赖性描述(14)，

其中，所述第一相互依赖性语法结构(24)限定在不同层组之间的相互依赖性，属于一个组的所有的层具有相同的基础层ID，并且所述第二相互依赖性语法结构(26)限定在表示为由第一相互依赖性语法结构互连几对组中的层之间的相互依赖性。

13.一种编码器，配置为使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入多层数据流(10)，所述多层数据流包括多个数据包(12)，每个数据包与所述不同层中的一层相关联，每层由基础层ID(20)、或基础层ID(20)和扩展层ID(22)索引，所述编码器配置为：

在所述多层数据流内插入

第一相互依赖性语法结构(24)，表示在所述基础层ID(20)能表示的几对不同值之间的相互依赖性，以及

第二相互依赖性语法结构(26)，表示在所述扩展层ID(22)能表示的几对不同值之间的相互依赖性，

设置所述第一相互依赖性语法结构和第二相互依赖性语法结构，以便基于所述第一相互依赖性语法结构和第二相互依赖性语法结构，能构造显示在所述不同层之间的层间预测依赖性的层间依赖性描述(14)，

其中，所述第一相互依赖性语法结构(24)限定在不同层组之间的相互依赖性，属于一个组的所有的层具有相同的基础层ID，并且所述第二相互依赖性语法结构(26)限定在表示为由第一相互依赖性语法结构互连几对组中的层之间的相互依赖性。

14.一种用于使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入多层数据流(10)的方法，所述多层数据流包括多个数据包(12)，每个数据包与所述不同层中的一层相关联，每层由基础层ID(20)、或基础层ID(20)和扩展层ID(22)索引，所述方法包括：

在所述多层数据流内插入

第一相互依赖性语法结构(24)，表示在所述基础层ID(20)能表示的几对不同值之间的相互依赖性，以及

第二相互依赖性语法结构(26)，表示在所述扩展层ID(22)能表示的几对不同值之间的相互依赖性，

设置所述第一相互依赖性语法结构和第二相互依赖性语法结构，以便基于所述第一相互依赖性语法结构和第二相互依赖性语法结构，能构造显示在所述不同层之间的层间预测依赖性的层间依赖性描述(14)，

其中，所述第一相互依赖性语法结构(24)限定在不同层组之间的相互依赖性，属于一个组的所有的层具有相同的基础层ID，并且所述第二相互依赖性语法结构(26)限定在表示为由第一相互依赖性语法结构互连几对组中的层之间的相互依赖性。

15.一种计算机可读存储介质，存储有具有程序代码的计算机程序，配置为在计算机上运行时，执行根据权利要求12或14所述的方法。

处理多层数据流的装置和方法、编码器和方法及存储介质

技术领域

[0001] 本申请涉及基于集群的依赖性信令，例如，在多层视频编码的领域中。

背景技术

[0002] 视频编解码器规范(例如，HEVC基础规范【1】及其扩展，例如，多视图扩展【2】【3】)通常获得机构，用于在分层编码场景中信令视频数据的每个编码块所属的层。信令机构需要随时可用，比较在图1中的表格中提供的HEVC中的den NAL单元报头(nuh_layer_id)内的层标识符。

[0003] 由于其在位方面的尺寸有限，所以相应HEVC语法元素允许64个值用于识别可扩展层、视图和/或深度。对于使用当前HEVC NAL单元报头语法的多视图视频，例如，这意味着64个视图或32个视图加上深度的组合限制为最大。虽然这对于很多共同的立体视图场景和其他应用充分，但是对于应用(例如，光场成像【4】、【5】或全息显示【6】)具有不止100个视图的大型摄像头阵列需要可扩展机构，用于信令层标识符的更大空间。

[0004] 而且，通过其相对于彼此的编码依赖性或空间关系将某些层或视图分组，构成层标识符的空间，可具有优点。进一步，由于在编码的视频位流内从已经可用的信息中获得这种信息，需要无法容忍的计算资源，所以在构成的层标识符空间内提供编码依赖性的紧凑表示，也具有优点。

发明内容

[0005] 因此，本申请的目标在于，提供用于与多层编码相结合基于集群的依赖性信令的信令的概念，这在(例如)更小的边信息开销和/或改进的向后兼容性方面更有效。由即将发生的独立权利要求的主题实现这个目标。

[0006] 本申请的一个方面涉及在多层数据流的层之间的层间依赖性的信号作用。根据该方面，通过表示在基础层ID能表示的几对不同值之间的相互依赖性的第一相互依赖性语法结构以及表示在扩展层ID能表示的几对不同值之间的相互依赖性的第二相互依赖性语法结构的方式描述层间依赖性，发现在一方面的层间依赖性的潜在多样性的太密集限制与另一方面的层间依赖性的太复杂信令之间的良好折衷，所述基础层ID和扩展层ID索引与所述多层数据流的部分相关联的层。根据该概念，可在一方面的可信令层间依赖性的增加的多样性与另一方面的用于信令所述层间依赖性的减小的边信息开销之间转移重点：例如，分别调用具有共同的基础层ID的层的组，即，“集群”，所述相同的第二相互依赖性语法结构可用于单独在所有集群内并且在通过所述第一相互依赖性语法结构彼此相关的所有集群之间调节所述相互依赖性。可替换地，所述第二相互依赖性语法结构的两个实例化可用于描述一方面在所述集群内并且另一方面在不同集群的层之间的层的相互依赖性。无论是将重点放在增加的多样性还是减小的边信息开销上，所述相互依赖性信令理念的结果是保持低信令开销。

[0007] 本申请的一个方面涉及与多层视频信号的每个数据包相关联的层ID的信令。具体

而言，这方面实现信令该层关联的有效方法，然而，与编码解码器保持向后兼容性，根据该向后兼容性，基础层ID字段的某个值限制为不可扩展，例如，在基础层ID字段内的基础层ID值0。并非特别相对于该不可扩展的基础层ID值避开该限制，通过将基础层ID字段细分成第一子字段和第二子字段，通过可扩展的方式信令多层数据流的部分的层ID：每当基础层ID字段的第一子字段满足预定标准，提供扩展层ID字段，并且如果基础层ID字段的第一子字段不满足预定标准，那么省略扩展层ID字段。上述不可扩展基础层ID值“隐藏”在这组基础层ID值内，其中，基础层ID字段的第一子字段不满足预定标准，因此，这个不可扩展基础层ID值不单独处理，而是渲染前一个组的一部分。确切地说，如果基础层ID字段的第一子字段满足预定标准，那么从在多层数据流内信令的扩展层ID字段中获得扩展值，以便这同样位于扩展值的域的第一子集内，并且如果基础层ID字段的第一子字段不满足预定标准，那么该扩展值设置为与扩展值的域的第一子集不相交的值。然后，使用扩展值以及从基础层ID字段的第二子字段中获得的集群值，索引与相应部分相关联的层。总而言之，尽管保持向后兼容性，也未丧失信令效率。

[0008] 本申请的一个方面涉及用于多层视频信号的层的至少一个特征的信令(例如，对于每层，通过层间预测与相应层直接相关的相关层的指示)或者上述第二相互依赖性语法结构的信令。根据该方面，在多层视频信号内信令最大语法元素，以指示多层视频信号的数据包的扩展层ID字段的最大使用值，例如，最大语法元素的范围是(例如)延伸穿过多层视频信号的几个部分的多层视频信号的预定部分。因此，根据该方面，对于接收多层视频信号的装置(例如，解码器或网络元件)，可行的是为多层视频信号的较大预定部分获得关于扩展层ID字段可信令的可能值的可能域的实际消耗部分的知识，并且不需要为基础层ID字段值和扩展层ID字段值的每个组合信令至少一个特征，更确切地说，足以为基于最大假定值确定的最大层数信令所述至少一个特征。因此，不需要为具有层ID的层传输/信令所述至少一个特征，所述层的扩展层ID不在多层视频信号的预定部分内发生。除此以外，根据进一步实施方式，最大假定值的知识可用于减少用于信令每个部分的层ID的边信息开销，即，用于减少在多层视频信号的数据包内信令扩展层ID字段所需要的位。

附图说明

[0009] 上述方面可组合或者单独使用，并且上述方面的优选实现方式是从属权利要求的主题并且稍后参照附图讨论，其中：

[0010] 图1示出了在HEVC内的NAL单元报头的语法；

[0011] 图2示意性示出了包括多层视频编码器、网络元件以及多层视频解码器的环境，其中，多层视频编码器、网络元件以及多层视频解码器可由在以下附图中描述的概念的任何概念或任何组合改进；

[0012] 图3示意性示出了网络装置以及通过逐个集群的方式在到达网络装置的多层数据流内信令层相互依赖性的概念；

[0013] 图4示意性示出了根据图3的概念信令层间依赖性的方式；

[0014] 图5示意性示出了在使用图3的信令理念的情况下获得层间依赖性的更具体的实例；

[0015] 图6通过示出通过指示层ID扩展的语法元素扩展的一部分薄片段报头来示出薄片

段报头的HEVC语法的可能扩展；

[0016] 图7示出了示例性扩展为实现图3的信令理念的VPS语法的实例；

[0017] 图8到图10示出了图7的实例的替换物；

[0018] 图11示意性示出了用于设置本申请的进一步概念的装置，根据该概念，在数据流内信令扩展层ID字段的最大假定值；

[0019] 图12示意性示出了图11的概念；

[0020] 图13示出了在扩展为包括扩展层ID字段的HEVC的薄片段报头之中的部分的实例；

[0021] 图14示出了扩展为包括语法元素的VPS语法的实例，以便表示图13的扩展层ID字段的长度；

[0022] 图15示出了扩展为包括表示扩展层ID字段的最大值的语法元素的VPS语法的实例；

[0023] 图16a到16c示出了扩展为包括扩展层ID字段的在SEI扩展、SPS语法或PPS语法的语法结构之中的部分的实例；

[0024] 图17示出了关于在视频数据流内的层的VPS语法信令信息的实例，根据相对于图11和12解释的最大假定值，循环穿过其语法部分多次；

[0025] 图18示出了根据特定实例例证在扩展层ID字段、基础层ID字段与层ID之间的关系的示意图，具体而言，使用layer_id_ext表示扩展层ID字段，使用nuh_layer_id表示基础层ID字段，并且使用layerID表示层ID，并且扩展层ID字段的能表示值的数量由max_layer_id_ext(加上1)表示，通过将扩展长度设置为等于2，将扩展层ID字段的长度选择为2，该数量在此处例证地设置为3；

[0026] 图19示出了扩展为包括扩展层ID字段的在薄片段报头之中的部分的实例；

[0027] 图20a示出了扩展为包括表示基础层ID字段的子字段的长度的语法元素的VPS语法的实例；

[0028] 图20b示出了修改为有条件地明确信令扩展层ID字段的在薄片段报头之中的部分的实例；以及

[0029] 图21示出了说明根据图18的概念使用实现的关系获得层ID的网络装置的功能的流程图。

具体实施方式

[0030] 在描述本申请的某些实施方式之前，参照图2，描述示例性多层视频编解码器环境，其中，可有利地使用下面概述的实施方式和方面。虽然示例性多层视频编解码器环境的这个概述不理解为限制下面进一步概述的实施方式，但是相对于以下附图示例性提供的细节和功能被视为描述下面进一步概述的实施方式的大量可能扩展，以便产生更详细的实施方式。在任何情况下，在后文中提出的说明性多层视频编解码器环境使得更容易理解下面概述的实施方式的优点。

[0031] 图2说明性示出了在其输入处接收视频材料202的多层视频编码器200。仅仅为了说明的目的，视频材料202说明为多视图视频信号，传输多个视图的每个视图的视频数据，例如，视频的每个图片的纹理和深度。一般而言，视频材料202能表示一个场景的多维取样。一个维度可能是时间t。即，视频材料202在时间上给该场景取样，从而为每个时刻产生一个

图片。而且，可从不同的视角捕捉该场景，从而在每个时刻为每个视图产生一个视频或者为每个视图产生一个图片。进一步，除了场景的纹理的空间取样，即，场景的颜色的空间取样，视频材料202还为每个视图或者视图的每个子集提供场景的深度图，用于相应的视图。例如，图2在使用阴影线说明的深度图与没有阴影线绘制的纹理图或图像之间区分。然而，应注意的是，应注意的是，下面进一步概述的实施方式都不限于这种多视图材料。确切地说，提供在图2中的说明，仅仅为了减轻下面提出的描述。

[0032] 多层视频编码器200配置为将视频材料202编码入多层数据流或者视频信号204。具体而言，多层视频编码器200使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入多层数据流204。这表示以下内容。多层视频编码器200生成多层数据流，以便多层数据流包括多个数据包206，每个数据包与不同层中的一层相关联。例如，多层视频编码器200将视频材料202的信息量的某个基础等级编码入某个基础层的数据包206，例如，仅仅一个视图的纹理等。例如，从在每个数据包206内的层识别语法元素结构208中可获得数据包206与各种层中的任何层的关联，其中，例如，层识别语法元素结构208可包含在相应数据包206的报头210内。例如，为了说明的目的，数字1、2以及3显示为刻入数据包206内，从而分别使用层ID 1、2以及3说明层。在这些层之中，例如，具有层ID 1的层可形成最低或者基础层，提供信息量的基础等级。其他层的数据包206可仅仅在一个或多个或各种信息类型的方面提高在基础层的数据包206上可用的信息量。例如，除了在基础或参考层的数据包206内已经编码的视图，某个层的数据包206还可具有在其内编码的进一步视图。可替换地，某个层的数据包206可具有关于视频材料202的场景的深度信息，例如，示图的深度信息，该信息的纹理已经编码入可能是任何更低层的基础或者(更具体而言)参考层的数据包。同样，表面反射率信息可编码入进一步层的数据包，即，相对于在该场景内的对象的表面反射率的场景的空间取样，与照明环境无关。并且甚至进一步地，阿尔法(alpha)混合信息可编码入进一步层的数据包，即，对应于单独图片样本的场景点的透明度的空间取样。某个层的数据包206还可在某个颜色分量上增加信息，或者可增大空间分辨率，即，提供空间分辨率细化。同样，某个层的数据包206可仅仅提供SNR分辨率细化，即，提高编码的视频材料的信噪比。为了就各种层而言，在多层数据流204内避免冗余，多层视频编码器200使用层间预测：即，某个层的数据包206相对于从一个或多个其他层(相对于前一层称为参考层)中获得的层间预测在其内编码预测残差。层间预测可应用于不同的实体中，例如，描述视频材料202的语法元素和参数。例如，描述由某个层的数据包206传输的预测残差可校正逐个像素地在空间域内从所述一个或多个参考层中获得的层间预测。可替换地，例如，多层视频编码器200可使用变换残差编码，并且例如，可在变换域(例如，DCT域)中逐个变换系数地发生层间预测校正。可替换地或者此外，多层视频编码器200可能是使用空间和/或时间预测的混合视频编解码器型，并且例如，此外或者可替换地，层间预测可属于运动向量的细化。甚至进一步地，多层视频编码器200可使用分层多树细分，以便将视频材料202的图片细分成：编码块，以所述编码块为单位，应用不同的预测模式，例如，空间和时间预测；变换块，以所述变换块为单位，在预测残差上执行上述变换；和/或其他块体，以所述其他块体为单位，设置某些编码参数，用于编码视频材料202，而非重新为进一步层信令细分信息，可从所述一个或多个基础层中的任一个中完全采用所述细分，或者可信令相对于该细分信息的细化。

[0033] 通过在层之中的层间依赖性，多层视频信号204的层通过树212的分支彼此相关，

所述树的节点由上述层形成。为了传输关于在层之中的层间依赖性的信息，即，使关于层间依赖性树212的信息在解码侧上可用，多层视频信号204可具有编码入该信号的关于该信号的信息。

[0034] 在图2中，例如，说明多层视频编码器200在视频数据流204内点缀包括或者传输信息214的高级语法数据包216。信息214可包括层间依赖性语法结构，以便描述在树212内的层间依赖性。

[0035] 然而，关于树212的信息可在编码器和解码器上可替换地默认已知或者可重构。因此，此外或者可替换地，层相关的信息214可包括与某个层ID扩展机构相关的信息。具体而言，虽然对于很多应用，中等数量的层足以形成多层视频信号204，但是一些其他应用可经受层识别语法元素结构208可信令的太少数量的层。换言之，构成层识别语法元素结构208，以便对需要这种大量层的那些应用，这也会容纳大量层，在考虑应用的整体多重性时，会不利地造成高边信息开销，并且解释层识别语法元素结构208，以便这同样仅仅容纳中等数量的层，这会拒绝其他应用由位于数据流204之下的多层视频编解码器充分支持。因此，可使用某个扩展机构，以便能够在层识别语法元素结构208仅仅容纳较低的中等数量的层的数据流204的模式与层识别语法元素结构甚至容纳大量层的模式之间切换，并且信息214可另外或者可替换地参与在这些模式之间切换。

[0036] 虽然高级语法数据包216的效力范围可包含整个多层视频信号204，但是数据包216所属的预定部分也可能更小，例如，预定部分可与多层视频信号204在时间上分成的块体相关，每个块体是图片序列。例如，使用DASH或另一个合适的流协议，多层视频编码器200可能以刚刚提及预定部分为单位，改变编码入位流204的层数、具有树结构212的形式的层间依赖性和/或相对于层识别语法元素结构208在扩展或非扩展之间的切换。在任何情况下，定期传输高级语法数据包216使接收者能够在随机访问的基础上具有多个随机访问时间实例，以开始在其间解码多层视频信号214。

[0037] 图2还示例性示出了用于接收多层数据流的装置。例如，可选的网络元件218接收并且处理多层视频数据流204，以便将该数据流转发给多层视频数据流204的接收者。因此，图2还示出了配置为解码多层视频数据流204的多层视频解码器220。这两个实体(即，网络元件218和多层视频解码器220)形成用于处理多层数据流204的装置的实例。

[0038] 网络元件218不需要能够使用在其各自层之间的层间预测解码多层数据流204。尽管如此，都需要通知网络元件218和多层视频解码器220(即，接收者)与所述各种层以及由树21限定的在其间的层间依赖性相关联的数据包206。例如，网络元件218丢弃与层相关联的数据流204的数据包206，其中，例如，由于在链路内的比特率缺点，即，接收者不能再现额外信息量等，所以分别在网络元件218与多层视频解码器220之间的进一步链路内初步不需要所提供的额外信息量。同样，多层视频解码器220还可决定丢弃响应于外部参数(例如，当前可用的计算功率)、再现装置的显示性能(例如，空间分辨率)、最大数量的输入视图等的某些层的一些数据包206。即，装置218/220能够读取数据包206的层识别语法元素结构208，并且如果存在的话，那么能够从信息214中获得在层之间的层间依赖性，和/或能够相对于响应于信息214的层识别语法元素结构208，在扩展模式或非扩展模式之间切换，和/或能够从信息214中相对于层读取其他特征。

[0039] 除此以外，多层视频解码器220能够通过收集和使用与该层以及通过帧间预测依

赖性与该层相关的所有层相关联的数据包206的信息，从入站数据流204中重构视频材料202，直到某个等级(参照树结构212)。即，如上所述，多层视频解码器220可使用某个层的数据包206，作为相对于多层视频解码器220从该层的一个或多个参考层的数据包206中获得的层间预测的预测残差。在这方面，如上所述，多层视频解码器220还可能是使用变换残差解码、混合视频解码、分层多树细分和/或其他编码概念的解码器。

[0040] 仅仅为了完整性，图2示出了可示例性提供多层视频编码器200和数据流204，以便数据包206通过这种方式包含在数据流204内，以便属于不同时间实例或图片的数据包206不彼此交错，以便在每个时刻形成连续访问单元222，每个访问单元收集属于该访问单元222的相应时刻的数据包206，以便在该访问单元222之前或之后，属于任何其他时刻的数据包206在数据流204内。然而，应注意的是，选择该限制，仅仅用于说明的目的，并且可替换地选择在数据流204内的数据包206的更松弛的设置。

[0041] 在下文中，更详细地描述根据树结构212信令层间依赖性的可能性。如上所述，信息214可包括该信令。根据该可能性，所述层分成集群。在数据流内，单独信令在一方面是集群并且另一方面是在集群内的层之中的层间依赖性。在这范围内，下面提出的描述表示实现图2的信息214的可能性，以便信令层间依赖性212，但是如上所述，在后文中相对于图3描述的细节不应限于在图2中陈述的细节。确切地说，图2应被视为相对于图3的描述的可能实现储层。

[0042] 图3示出了网络装置18，该装置可以是图2的网络元件218或多层视频解码器220，并且多层数据流10同样配置为处理(例如)图2的数据流204。

[0043] 图3示出了多层数据流10，在由层ID 1ID识别的不同层上将视频材料编码入多层数据流。每层对应于信息量的不同等级。已经相对于2讨论了在这方面的可能性，但是为了更容易理解，人们可考虑每层给数据流10增加某个视图。然而，这并非表示限于图3的描述。可替换地，每层可对应于信息量的不同测度的组合，例如，多个视图和空间分辨率等。

[0044] 因此，多层数据流10由多个数据包12构成，所述数据包可示例性对应于图2的数据包206。这种数据包12(或206)可能是子流，例如，所述子流允许波前并行处理编码入视频信号10的单独图片，所述子流可反过来由更小的单元构成，例如，NAL单元、薄片等。然而，同时主张图2和图3的数据包12或数据包206也可能是NAL单元、薄片或其他单元。

[0045] 每个数据包12与不同层中的一个相关联，并且为了减少多层数据流10的位消耗，使用层间预测，以便每个数据包12仅仅将“残差”加入数据流10的更低层的某个子集的数据包中。如在14中所示，对于每层，由小圆点说明“更低层”。

[0046] 根据在图3中示出的实施方式，如下面更详细地所描述的并且使用连续线路16在图中所说明的，可限制层间预测依赖性。具体而言，如在下面所描述的，由两个值的串联形成层ID。具体而言，接收多层数据流10的网络装置对每个数据包12读取使用在图2中的参考符号208表示的层ID语法结构，但是在此处由基础层ID字段20以及有条件地(例如，有条件地根据基础层ID字段或高级语法元素打开和关闭扩展机构)扩展层ID字段22构成。示例性地，在图中仅仅示出了具有这两个字段的数据包12。但是如上所述，基础层ID字段20的一个或多个值可信令相应数据包12的字段22的不存在。例如，整个扩展功能可以通过在数据流内的扩展标志在数据流内打开/关闭，并且需要或者不需要扩展层ID字段的可能条件可能在于，如上所述，基础层ID字段是0，是或者具有设置或未设置的某个位，或者采用在字段20

或者具有大于或小于某个值的值的基础层ID字段的子部分内的某个值。下面陈述进一步实例。

[0047] 基于层ID语法结构20和22,网络装置18获得识别与相应数据包12相关联的层的层ID,即,lID。下面描述不同的可能性。网络装置18也从多层数据流10中读取包含(例如)上述direct_dependency_flag的第一相互依赖性语法结构以及包含(例如)下面表示的direct_ext_dependency_flag和/或general_direct_ext_dependency_flag的第二相互依赖性语法结构。第一相互依赖性语法结构通过二进制的方式表示在基础层ID字段20能表示的几对不同值之间的相互依赖性,而第二相互依赖性语法结构通过二进制的方式表示在扩展层ID字段能表示的几对不同值之间的相互依赖性。基于这两个相互依赖性语法结构,然后,网络装置18形成显示在不同层之间的层间预测依赖性的层间依赖性矩阵,例如,在图中的14中描述的矩阵。下面使用伪码说明可进行导出的方式,所述伪码使用涉及第一和第二相互依赖性语法结构的接下来的循环,并且还相对于以下示图解释该方式。然而,应注意的是,所述信令的层间预测依赖性不需要实际上用于数据流内。确切地说,可能的层间预测依赖性的信令用于指导解码器或其他网络装置采用必要的步骤,相关的数据包按照合适的顺序可用,即,在参考数据包之前其他数据包根据层间预测依赖性参考的数据包。

[0048] 从下面的描述中显而易见,可执行层间依赖性矩阵14的构造,以便第二相互依赖性语法结构应用于所有实例中,其中,层间依赖性矩阵14与在从在基础层ID字段20内的相同值中获得的并且从而与该值相关联的层ID的层之间的层间预测依赖性相关。根据下面也更详细地描述的另一个实施方式,第二相互依赖性语法结构从位流10中读取并且在位流内传输几次,例如,单独用于基础层ID字段20的每个可能值或者用于基础层ID字段20的可能值的子集,同时使用索引,例如,在下面提出的实例中使用included_nuh_layer_id,使第二相互依赖性语法结构的各种实例化与基础层ID字段20的可能值相关联。在甚至其他实施方式中,该结构概括为允许所有可能的层间预测依赖性,并且通过第二相互依赖性语法结构可描述,即,通过为基础层ID字段的每对不同值传输该结构,其中,第一相互依赖性语法结构表示相互依赖性存在。

[0049] 换言之,并且如进一步在图4中所述,图3提出了用于信令层间依赖性的实例,根据该实例,单独的相互依赖性语法结构24和26用于通过分层的方式描述层间依赖性,在这两个相互依赖性语法结构24和26的范围之间的界面与由在数据流的每个数据包内的字段20和22提供的基础层ID和扩展层ID的范围之间的界面一致。由字段20和22提供的基础层ID和扩展层ID独特地限定包括字段20和22的相应数据包的层ID。这组扩展层ID和基础层ID的组合能表示的所有能表示的层ID在圆圈28内由圆点30表示。即,每个圆点30对应于不同的一对基础层ID和扩展层ID。例如,层ID可能是基础层ID和扩展层ID的串联。仅仅使用字段20提供的基础层ID,整组28层ID 30细分成层ID的不相交的组32,在下面称为集群32,其中,属于某个集群32的所有层ID具有相同的基础层ID。如上面相对于图2所述,由于在图4中使用在圆点30之间的虚线说明具有这些层间依赖性的层间预测,所以与层ID 30相关联的层通过树状方式彼此链接。为了容易理解,仅仅在图4中说明实际的层间依赖性的子集。

[0050] 在任何情况下,通过第一相互依赖性语法结构24表示通过在这对第一集群32的层与第二集群32的层之间的层间依赖性在两个集群32之间的链接。换言之,第一相互依赖性语法结构24粗糙地或者按照集群的方式描述在层之间的相互依赖性。在图4中,使用在集群

32之间的连续线路说明这些相互依赖性。第一和第二集群32的所有对(其中,第一集群的至少一层与在第二集群内的一个集群链接)互连,因此,在第一相互依赖性语法结构24中表示。然后,第二相互依赖性语法结构26阐明表示为由第一相互依赖性语法结构24互连的几对集群32中的哪些层实际上通过层间预测彼此相关。即,第二相互依赖性语法结构26阐明细粒度相互依赖性。然而,第二相互依赖性语法结构26也单独限定在每个集群32内的层之间的相互依赖性,即,在层之间的集群内依赖性。在图4中,例如,示出了6个集群32,从而产生第二相互依赖性语法结构26调节在不同集群32的层之间的细粒度相互依赖性的可能15对集群32加上第二相互依赖性语法结构26可在内部调节相互依赖性的6个集群。如上所述并且如下面进一步所述,因此,最大可具有第二相互依赖性语法结构的1到21个实例化34,即,每个集群单独具有1个并且互连的每对集群32具有一个。显然,不需要为信令为通过第一相互依赖性语法结构24不链接的几对集群32,信令第二相互依赖性语法结构26,从而保存有价值的边信息比特率。应注意的是,在使用第二相互依赖性语法结构26,以便描述不同的几对集群32的层的层间依赖性的情况下,对于所有集群32,每个集群32的层数应相等。如果使用基础层ID和扩展层ID的串联描述层ID,那么这是这种情况。然而,应提及的是,在理论上,集群32的层数可以在集群32之间变化。在这种情况下,例如,第二相互依赖性语法结构26的单独实例化34会在数据流内为互连的每对集群32信令,并且会为每个集群尺寸传输至少一个实例化34。

[0051] 例如,图3示出了通过将基础层ID用作最高有效数位并且将扩展层ID用作不太有效数位来从基础层ID和扩展层ID中获得层ID的情况。图3还示出了一个实例化34用于描述在一个集群内的层的层间依赖性并且另一个实例化34用于描述在不同集群的层之间的依赖性的示例性情况。为了完整性,要注意的是,图3的矩阵14具有与层ID一样多的行和列。仅仅填充在对角线之下的下半部分,这是因为任何层可仅仅通过层间预测取决于任何先前的(即,分层次更低的)层。在图3的实例中,列数对应于层的层ID,通过使用层间预测,所述层ID取决于进一步层,即,基础层,这些基础层由(例如)二进制层表示,而二进制零表示不参与层间预测相应层的层,所述层的层ID对应于当前列。在该范围内,在图3的实施方式中,第二相互依赖性语法结构26几乎描述矩阵14的子矩阵。

[0052] 如上所述,第一和第二相互依赖性语法结构24和26可由在高级数据包216内的信息214包括(比较图2)。图5示出了一个实例,其中,第一相互依赖性语法结构24显示在36中描述的层集群之间的相互依赖性。例如,具有基础层ID 2的集群取决于具有基础层ID 2和1的集群。

[0053] 第二相互依赖性语法结构的第一实例化34也存在于数据流内,并且通过子矩阵38的形式调节在图5中描述的层之间的集群内依赖性。进一步,根据图5的实例,数据流还包括调节不同集群的层的逐层相互依赖性的第二相互依赖性语法结构26的实例化34。具体而言,第二实例化可通过对于被参考集群的每个增强层ID具有一行并且对于参考集群的每个增强层ID具有一列的子矩阵40描述在不同集群的层之间的依赖性。

[0054] 在图5的实例中,子矩阵38放在矩阵36表示在集群之间的相互依赖性的每个位置,即,其中,定位1,并且所述集群具有相同的基础层ID,即,位于矩阵36的对角线上的集群,并且子矩阵40放在矩阵36通过“1”表示在不同的基础层ID的集群之间的相互依赖性的位置。在42中示出结果。

[0055] 应注意的是，通过矩阵(例如，矩阵42)描述层间依赖性，仅仅是用于描述层间依赖性的一个实例。还可使用其他描述。矩阵36到40由第一和第二相互依赖性语法结构编码的方式可如下：第一相互依赖性语法结构24可信令在对角线之下并且包括对角线的矩阵36的每个系数的二进制值。表示矩阵38的第二相互依赖性语法结构26的实例化34可信令在对角线之下并且不包括对角线的矩阵38的每个系数的二进制值。表示矩阵40的第二相互依赖性语法结构26的实例化34可信令矩阵40的所有系数的二进制值。

[0056] 在描述关于如何信令层间依赖性的可能性之后，在后文中示例性提出更详细的实现方式，作为HEVC标准的扩展。

[0057] 具体而言，层标识符集群和集群依赖性信令可如下嵌入现有编解码器内。

[0058] 两个语法元素nuh_layer_id和layer_id_ext可用于基于性能(例如，彼此的空间关系、编码依赖性或其他性能)将在视频位流内的编码层分组成所谓的集群。将层标识符构造成集群，可允许在单独集群内具有相等构造的编码依赖性的集群，即，在所有限定的集群内或者限定的集群的子集内的编码依赖性相同。如在图6中所述，可在集群内作为额外一组依赖性标志信令依赖性并且组合所述依赖性和现有依赖性信令(比较在HEVC扩展的VPS扩展内的direct_dependency_flag)，以确定特定层的参考层。

[0059] 等于0的direct_ext_dependency_flag[i][j]规定具有扩展索引j的层并非在相同层集群内具有扩展索引i的层的直接参考层。等于1的direct_dependency_flag[i][j]规定具有扩展索引j的层可能是在相同集群内具有扩展索引i的层的直接参考层。在对于在0到(1<<layer_id_ext_len)-1的范围内的i和j，不存在direct_ext_dependency_flag[i][j]时，推断出等于0。

[0060] 替换的语法可限制在具有max_layer_id_ext的direct_ext_depencency_flag语法元素之上的循环，以避免将编码依赖性信令给未使用的层标识符。

[0061] 可如下允许在集群内并且在集群之间的编码依赖性的更灵活的信令：

[0062] a)逐个集群地信令在集群内的编码依赖性；

[0063] b)限定多个集群依赖性组并且识别应用限定的集群依赖性的集群(例如，通过nuh_layer_id)；

[0064] c)通过信令在集群之间的第一依赖性并且信令在两个相关集群内的层的第二依赖性的方式，分层信令依赖性；

[0065] d)组合b)和c)。因此，信令(signal)以下数据：

[0066] 1、在集群之间的依赖性(例如，几对参考和相关集群)；

[0067] 2、限定在参考(被参考)集群和相关(参考)集群内的层之间的依赖性的多个集群依赖性组；

[0068] 3、对于每个集群依赖性组，其适应哪些集群依赖性的指示(如在1中信令的)。

[0069] 要注意的是，在集群内的层之间具有依赖性时，对于c)和d)，集群可取决于自身。

[0070] 在不均匀地使用用于构造的性能空间(例如，摄像头的空间位置)时，a)到d)允许构造层标识符的空间。一个实例是二维摄像头阵列，该空间密度不恒定或者该阵列并非矩形。下面提供语法实例，用于根据以上内容通过多种方式信令限定集群依赖性组。

[0071] a)图7提供了在逐个集群地限定在集群内的编码依赖性的视频参数组扩展语法方面的示例性实施方式。

[0072] 等于0的direct_dependency_flag[i][j]规定具有索引j的层或集群并非具有索引i的层或集群的直接参考层或参考集群。等于1的direct_dependency_flag[i][j]规定具有索引j的层或集群可能是具有索引i的层或参考集群的直接参考层或参考集群。在对于在0到vps_max_layers_minus1的范围内的i和j,不存在direct_dependency_flag[i][j]时,推断出等于0。

[0073] 等于0的general_direct_ext_dependency_flag[i][j]规定具有扩展索引j的层并非在相同层集群内具有扩展索引i的层的直接参考层。等于1的general_direct_dependency_flag[i][j]规定具有扩展索引j的层可能是在相同集群内具有扩展索引i的层的直接参考层。在对于在0到max_layer_id_ext的范围内的i和j,不存在general_direct_ext_dependency_flag[i][j]时,推断出等于0。

[0074] 等于0的direct_ext_dependency_flag[i][j][k]规定具有扩展索引k的层并非在第i层集群内具有扩展索引j的层的直接参考层。等于1的direct_ext_dependency_flag[i][j][k]规定具有扩展索引k的层可能是在第i集群内具有扩展索引j的层的直接参考层。在对于在0到max_layer_id_ext的范围内的i、j以及k,不存在direct_ext_dependency_flag[i][j][k]时,推断出等于0。

[0075] b)图8提供了在视频参数组扩展中的语法方面的示例性实施方式,该扩展限定通过多个集群依赖性组(number_of_cluster_dependency_sets)的信令在集群内的编码依赖性、在每个集群依赖性组内的编码依赖性(direct_ext_dependency_flag)以及应用集群(由规定的语法元素included_nuh_layer_id通过其相应nuh_layer_id的积极标志示例性识别)。

[0076] c)图9提供了通过信令在集群之间的第一依赖性并且仅仅信令在两个相关集群内的层的第二依赖性的方式分层信令依赖性的示例性实施方式。

[0077] 等于0的direct_dependency_flag[i][j]规定具有索引j的层或集群并非具有索引i的层或集群的直接参考层或参考集群。等于1的direct_dependency_flag[i][j]规定具有索引j的层或集群可能是具有索引i的层或参考集群的直接参考层或参考集群。在对于在0到vps_max_layers_minus1的范围内的i和j,不存在direct_dependency_flag[i][j]时,推断出等于0。

[0078] 注意-如果layer_id_ext_len大于0,那么direct_dependency_flag[i][j]信令在集群之间的依赖性,否则,(layer_id_ext_len等于0),direct_dependency_flag[i][j]信令层的依赖性。

[0079] 等于0的direct_ext_dependency_flag[i][j][k][l]规定在第j集群内的第l层并非在第i集群内的第k层的直接参考层。等于1的direct_ext_dependency_flag[i][j][k][l]规定在第j集群内的第l层可能是在第i集群内的第k层的直接参考层。在不存在时,direct_ext_dependency_flag[i][j][k][l]推断出等于0。

[0080] d)图10提供了b)和c)的组合的示例性实施方式。

[0081] 等于0的direct_dependency_flag[i][j]规定具有索引j的层或集群并非具有索引i的层或集群的直接参考层或参考集群。等于1的direct_dependency_flag[i][j]规定具有索引j的层或集群可能是具有索引i的层或参考集群的直接参考层或参考集群。在对于在0到vps_max_layers_minus1的范围内的i和j,不存在direct_dependency_flag[i][j]时,

推断出等于0。

[0082] 注意-如果layer_id_ext_len大于0,那么direct_dependency_flag[i][j]信令在集群之间的依赖性,否则,(layer_id_ext_len等于0),direct_dependency_flag[i][j]信令层的依赖性。

[0083] 在dependency_set_applies_flag[n][i][j]等于1时,等于0的direct_ext_dependency_flag[n][k][l]规定在第j集群内的第l层并非在第i集群内的第k层的直接参考层。在dependency_set_applies_flag[n][i][j]等于1时,等于1的direct_ext_dependency_flag[n][k][l]规定在第j集群内的第l层可能是在第i集群内的第k层的直接参考层。

[0084] 等于0的dependency_set_applies_flag[n][i][j]规定在第i集群内的层与在第j集群内的层之间的依赖性不由direct_ext_dependency_flag[n][k][l]规定。等于1的dependency_set_applies_flag[n][i][j]规定在第i集群内的层与在第j集群内的层之间的依赖性由direct_ext_dependency_flag[n][k][l]规定。在不存在时,dependency_set_applies_flag[n][i][j]推断出等于0。在对于在0到(number_of_cluster_dependency_sets-1)的范围内(包括0和number_of_cluster_dependency_sets-1)的i和j以及所有n的特定组合,dependency_set_applies_flag[n][i][j]等于0时,在第j集群内的层并非在第i集群内的任何层的直接参考层。

[0085] 因此,在实现到HEVC扩展内的图2到图5的及与集群的相互依赖性信令的以上实例中,图2的数据包206是薄片或薄片段,并且例如,数据包216是参数组,例如,视频、图片或序列参数组。根据图7的语法实例,对于每对集群i、j(i≤j),使用语法元素direct_dependency_flag,信令第一相互依赖性语法结构24,其中,vps_max_layers_minus1是基础层ID的最大数量减去1。因此,例如,语法元素显示图5的粗糙矩阵36。称为unique_cluster_dependencies_flag的语法元素44在两个选择之间区分:第二相互依赖性语法结构26的一个实例化提交给并且应用于所有基础层ID,或者为每个基础层ID i,传输一个实例化。在第一种情况下,标志general_direct_ext_dependency_flag根据每个基础层ID的38共同建立子矩阵,并且在第二种情况下,标志direct_ext_dependency_flag根据每个基础层ID的38单独建立子矩阵。unique_cluster_dependencies_flag在两个选择之间切换。因此,在图7的情况下,如果unique_cluster_dependencies_flag等于0,那么所产生的子矩阵38对应于相等的参考和被参考集群ID的每个位置放在矩阵36内,其中,1由direct_dependency_flag表示。可默认使用预定的子矩阵,即,所有参与装置(例如,编码器、网络元件以及多层视频解码器)已知的子矩阵,填充剩余位置,其中,第一相互依赖性语法结构24表示在不同集群ID的集群之间的相互依赖性。在unique_cluster_dependencies_flag等于1的情况下,对于第一相互依赖性语法结构24在相等的被参考和参考集群ID的位置表示1的每个位置,在矩阵42内的相应位置信令并且定位正好这个集群ID的单独子矩阵。因此,max_layer_id_ext对应于在每个vps_max_layers_minus1集群内的层数。

[0086] 在图8的情况下,第一相互依赖性语法结构24通过与在图7的实例中相同的方式传送:对于每个集群ID,每个集群ID的一个标志等于或小于相应集群ID。标志cluster_dependency_sets_present_flag 46在如上所述使用语法元素general_direct_ext_dependency_flag的图7的第一选择与不同模式之间切换,根据所述模式,使用称为集群依

赖性组的数量的语法元素，明确信令使用语法元素direct_ext_dependency_flag在数据流内提交的第二相互依赖性语法结构26的实例化的数量。然后，通过使用语法元素included_nuh_layer_id 50明确列出每个实例化适用的集群ID，这些实例化与这些实例化适用的集群ID相关联。通过该措施，不为每个集群ID提交第二相互依赖性语法结构26的实例化34。确切地说，在cluster_dependency_sets_present_flag等于1的情况下，仅仅提交第二相互依赖性语法结构26的更少量的实例化，使每个实例化与和该实例化相关联的集群ID相关联。而且，就相互依赖的几对不同集群而言，默认规则可用于在这种位置填充矩阵42(在集群对角线之下)。

[0087] 根据图9的实例，相对于第一相互依赖性语法结构，通过交错的方式示例性信令第二相互依赖性语法结构的信令，但是根据替换的实施方式，可省略该交错。在任何情况下，在图9的实施方式中，为每对(相等的或不相等的)集群ID提交子矩阵38或40，即，第二相互依赖性语法结构26的实例化34，其中，第一相互依赖性语法结构24表示集群间依赖性。从语法实例中可推论出，子矩阵40的传输消耗比子矩阵38更多的位direct_ext_dependency_flag。这分别从条件条款52和条件条款54中可推论。换言之，在传输用于集群内部相互依赖性调节的第二相互依赖性语法结构26的实例化时，其中，由计数i表示的参考集群ID等于由j表示的被参考集群ID，仅仅传输direct_ext_dependency_flag，用于参考和被参考层的增强层ID的组合，其中，被参考层的增强层ID(在此处是l)小于参考层的增强层ID(在此处是k)。即，在子矩阵38之中，仅仅为在子矩阵38的对角线之下的位置传输direct_ext_dependency_flag。在其他情况下，即，对于相互依赖的集群对$i \neq j$的第二相互依赖性语法结构26的实例化，传输整个子矩阵，即，用于子矩阵40的每个位置的标志direct_ext_dependency_flag。

[0088] 图10的实施方式组合图8和图9的实施方式的特殊特征：整个子矩阵，即，提交第二相互依赖性语法结构26的实例化，即，在数量上的number_of_cluster_dependency_sets，然后，这些实例化中的每个与矩阵36的网格站点相关联，其中，例如，通过1应用第二相互依赖性语法结构26的各实例化(在第一语法结构24表示集群依赖性的存在的实例化之中(比较56))。

[0089] 甚至换言之，如相对于图9所述，第二相互依赖性语法结构26的多个实例化34可从多层数据流中读取，即，用于值$0<k \leq j \leq n<N$的几对(j,k)，N是所述基础层ID能表示的值的数量。n可设置为等于N，但是也如在本文中所述，可使用实际上使用的集群基数的明确信令，以便限制传输开销。穿过所述对(j,k)(在图9中的24中见在i和j之上的循环)，并且有条件地根据表示在该对之间存在或不存在相互依赖性的所述第一相互依赖性语法结构，为所述对(j,k)执行或者抑制读取第二相互依赖性结构(在图9中见“如果direct_dependency_flag[i,j]”，其中，i、j分别对应于j和k)，其中，为对j＝k的(j,k)读取的所述第二相互依赖性结构26的实例化34表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID j的层之中的其他层的层间预测，并且为一对不同值$k<j$的(j,k)读取的所述第二相互依赖性结构26的实例化34表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID k的层之中的层的层间预测。与图7相似，然而，第二相互依赖性结构26的一个实例化也可能从多层数据流中读取，第二相互依赖性结构26的所述一个实例化分别对于每个集群值(j,j)对，共同表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID j的层之中

的其他层的层间预测，与图7中的情况一样(如果unique_cluster_dependencies_flag等于0)，或者共同对于对(j,k)(j不等于k)(其中，第一相互依赖性语法结构24表示在相应对之间的相互依赖性的存在)，表示在具有基础层ID j的层之中的哪些层是依赖于在具有基础层ID k的层之中的层的层间预测。扩展层ID 22能表示的值的数量表示为M，第二相互依赖性语法结构可包括用于扩展值为0<q<p≤m<M的每对(p,q)的一个二进制值，以便表示在具有基础层ID i的层之中的哪些层是依赖于在具有基础层ID i的层之中的其他层的层间预测，以及用于每对扩展值0<p,q≤m<M的一个二进制值，以便表示具有扩展值p并且具有基础层ID j的哪个层是依赖于具有扩展值q并且具有基础层ID k的层的层间预测。而且，如在说明书中的别处所述，m可设置为M，但是也如在本文中所述，可使用实际上使用的扩展基数的明确信令，以便限制传输开销。进一步，如在图8和图10中所示，可使用索引语法结构50，以便使第二相互依赖性结构的实例化与基础层ID值的单独对(j,k)相关联。在此处，图10显示了关联语法结构可包括用于每对(j,k)的标志，并且所述关联语法结构的解析可涉及跳过值0<k≤j≤n<N的对(j,k)，其中，所述第一相互依赖性语法结构表示在对(j,k)之间不存在相互依赖性(独立性)。

[0090] 即，在以上部分中描述的内容能表示为“层标识符集群和集群依赖性信令”。在这方面，装置可能是视频解码器。因此，在操作时，视频解码器能够选择由(扩展的)layerID识别的位流的数据包，用于解码。然而，可替换地，该装置可以是能够基于外部因素(例如，网络利用率、关于解码器的知识等)丢弃由(扩展的)layerID识别的位流的数据包的网络元件。

[0091] 自然地，上述概念也影响编解码器的现有编码器，以由该概念改进：编码器(例如，图2的编码器)配置为使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入多层数据流10，所述多层数据流包括多个数据包12，每个数据包与所述不同层中的一层相关联，每层由基础层ID 20或基础层ID 20和扩展层ID 22索引，并且在所述多层数据流内插入表示在所述基础层ID 20能表示的几对不同值之间的相互依赖性的第一相互依赖性语法结构24以及表示在所述扩展层ID 22能表示的几对不同值之间的相互依赖性的第二相互依赖性语法结构26，设置所述第一相互依赖性语法结构和第二相互依赖性语法结构，以便基于所述第一相互依赖性语法结构和第二相互依赖性语法结构，能构造显示在所述不同层之间的可能层间预测依赖性的层间依赖性描述14。

[0092] 接下来描述的细节涉及通过有效的方式信令层相关的信息的可能性，其中，例如，这种层相关的信息可包括涉及如相对于图2到图10概述的层间依赖性的信息。作为起始点，使用上面通常提出的情况，根据该情况，每个数据包206(比较图2)包括的层识别语法元素结构208由基础层ID字段108和扩展层ID字段112构成，如在图11中所述。图11示出了用于处理多层数据流(例如，网络元件218)或多层视频解码器202的装置100，如在图2中所示。装置100显示为包括接收器102，所述接收器配置为接收由一系列数据包构成的多层视频信号，每个数据包包括层识别语法元素。

[0093] 由接收器102接收的多层视频信号104(对应于在图2中的204)由一系列数据包106(对应于在图2中的206)构成，每个数据包包括层识别语法元素结构(对应于在图2中的208)，该结构包括至少基础层ID字段108并且无条件地或者有条件地取决于高级语法元素或基础层ID字段108、扩展层ID字段112的值。

[0094] 例如，数据包106可以是多层视频信号104细分成的NAL单元、薄片、瓷砖、图片或其他数据包。对于包括多个数据包106的多层视频信号的预定部分，例如，部分113，装置100也包括的层识别扩展器110从多层视频信号中读取最大语法元素，表示多层视频信号的预定部分113包括的数据包108的扩展层ID字段112的最大假定值。例如，预定部分113可能是编码的视频序列、块体、图片组等。最大语法元素114可包含在部分113的特殊数据包内，例如，VPS NAL单元。对于在预定部分113内的每个数据包，扩展器110基于由108并且有条件地112构成的层ID语法结构，确定相应数据包的层ID。例如，这两个语法元素108和112的两个值的串联可造成层ID。

[0095] 扩展器110可使用上面说明的最大语法元素，获得扩展层ID字段的位长度。可替换地，为此，可使用额外语法元素。甚至可替换地，可默认固定用于表示扩展层ID字段112的位数。

[0096] 然后，扩展器110基于最大假定值，确定在部分113内的最大层数。例如，扩展器110也从数据流104中读取语法元素，对于部分103，表示在部分103内的数据包106的基础层ID字段108的最大假定值，并且组合这两个最大假定值，用于确定最大层数。

[0097] 通过等于最大层数的多次反复解析多层视频信号的层表征语法部分，扩展器110确定最大层数的每层的至少一个特征。有利地，数据流104不需要信令每个可能值的至少一个特征，即，并非用于扩展层ID字段112的整个基数，而是仅仅用于在该基数之中的实际上使用的子部分。此外，可甚至使用最大语法元素114，以便如上所述，获得在数据流104内的扩展层ID字段112的表示位的数量。例如，“特征”可能是对其他层等的层内预测。

[0098] 换言之，相对于图11，描述了可以在数据流204/104内传输关于在考虑在兴趣预定部分113内的所有数据包106/206时实际上采用的扩展层ID字段的能表示状态的数量的明确信号作用。例如，假设扩展层ID字段112是N位的字段，那么2^N扩展层ID 120由扩展层ID字段112能表示。字段112的长度可通过在图12中使用虚线122表示的某个高级语法元素固定或者可信令并且设置在数据流内。然而，由于某种原因，并非所有可用扩展层ID字段可用于兴趣预定部分113内。在图12中，例如，说明了仅仅使用7个能表示的扩展层ID，即，通过阴影线。

[0099] 根据图11，最大语法元素114表示扩展层ID字段112的实际使用/假定值的这个数量。由于该语法元素114，所以可行的是，仅仅相对于扩展层ID的实际假定数量，而非用于扩展层ID字段112的能表示值的整个数量，将层特定的信息插入数据流内。因此，如在图12中的122中所示，例如，解析多层视频信号的装置可解析多层视频信号的某个层表征语法部分124，解析的次数仅仅等于由语法元素114表示的最大层数，而非扩展层ID字段的实际可用/能表示值的整个数量。因此，通过数据流消耗的边信息量可保持更低。例如，在图6到图10的实例中，可行的是，将max_layer_id_ext设置为等于由语法元素114表示的扩展状态的最大假定数量，而非将该值设置为等于扩展层ID字段112的能表示值的整个数量。因此，“表征语法部分”124可由先前提出的标志direct_ext_dependency_flag体现。

[0100] 如上所述，字段112的位长度的存在/信令可选。在任何情况下，与由语法元素122设置的扩展层ID字段112的可用/能表示值的数量的更粗糙的设置相比，语法元素114表示实际使用的/活动的扩展层ID的数量的粒度/保真度增大。优选地，语法元素114表示扩展层ID字段112的实际假定值的数量的保真度或粒度足够精细，以便表示实际假定值的精确数

量。可替换地，由语法元素122提供的在1与字段112的可用/能表示值的最大数量的信号作用的保真度/粒度之间的某个地方的保真度/粒度也可行。换言之，最大语法元素114能表示扩展层ID字段108的最大假定值，以小于$(n-1)^2$为单位，n是扩展层ID字段的位长度，或者可能甚至是1。

[0101] 因此，根据图11和12，描述了装置(例如，网络元件或多层视频解码器)可包括接收器102和扩展器110，其中，接收器102配置为接收由一系列数据包106构成的多层视频信号104，每个数据包包括层识别语法元素结构，该结构由基础层ID字段并且可能仅仅有条件地扩展层ID字段112构成，其中，对于包括在这系列数据包106之中的多个数据包106的多层视频信号的预定部分113，层识别扩展器110配置为从多层视频信号104中读取最大语法元素114，最大语法元素114表示在预定部分113包括的数据包106之中的扩展层ID字段112的最大假定值。对于在预定部分113内的每个数据包106，扩展器110基于层识别语法元素结构，确定相应数据包106的层ID，例如，通过如上所述，串联字段108和112的基础层ID和扩展层ID。扩展器110还基于最大假定值，确定在预定部分113内的最大层数，并且通过等于最大层数的多次反复解析层表征语法部分104，确定最大层数的每层的至少一个特征。在组合图11和12的实施方式和上面相对于图2到图10概述的实施方式时，“最大层数”能表示每个集群的最大层数，在这种情况下，例如，“最大层数”的确定直接将最大假定值用作“最大层数”，并且“至少一个特征”可能是表示在第二相互依赖性语法结构内的层间依赖性的标志的数量。然而，特定的另一个实例也可能是在单独层ID的相应编码参数组的数据流内的信号作用。在这种情况下，“最大层数”可确定为等于所使用的或者能表示的基础层ID的数量乘以最大假定值，即，实际上使用的/假定的扩展层ID的数量。其他实例也可行。在任何情况下，通过语法元素114传输实际假定值，能够保存有价值的边信息比特率。

[0102] 如上所述，图11和12的概念可与上面相对于图2到图10概述的概念组合，或者可单独使用，而不使用前面描述的概念。

[0103] 与上面相对于图2到图10提出的描述相似，然而，以下描述显示了如何将图11和12的概念嵌入现有视频编解码器(例如，HEVC的扩展)内的可能性。

[0104] 在HEVC中，可从在视频和元数据包内明确信令的层标识符中获得称为LayerID的层标识符变量，例如，在视频或元数据包的每个块体内信令的nuh_layer_id和额外信息，形成特定数据包的新变量LayerId，以用于其识别。

[0105] 根据图11和12信令的额外信息可涉及额外语法元素，比较layer_id_ext，例如，在视频或元数据包的报头或有效载荷的某个部分内，考虑特定的视频或元数据包属于增强层，从而与仅仅基础层传统装置保持兼容。在视频或元数据包属于基础层，但是由增强层处理能力装置处理时，可假设默认值用于layer_id_ext。

[0106] 图13提供信令额外信息的薄片段报头的实例，作为在包含在HEVC编码视频的增强层(比较nuh_layer_id>0)的视频数据包内的薄片报头中的语法元素layer_id_ext。

[0107] 使用数学运算，以便将nuh_layer_id和layer_id_ext组合成LayerId，这允许更大的标识符空间并且对于nuh_layer_id和layer_id_ext的两个值的独特组合是独特的。例如，数学运算可将nuh_layer_id的位用作LayerId变量的MSB，并且将layer_id_ext用作LSB，反之亦然，例如，如下。

[0108] 如下确定变量LayerId，其中，LengthOfExtension是在位方面的layer_id_ext语

法元素的长度。

[0109] LayerId＝(nuh_layer_id<<LengthOfExtension)+layer_id_ext

[0110] 图11和12的概念可包括明确信令用于在视频或元数据包的报头或有效载荷的某个部分内传输额外信息(比较上面的LengthOfExtension)的位的量,以便有效利用传输的数据量。图14提供在元数据包(例如,HEVC视频参数组扩展语法)中信令的LengthOfExtension值的示例性语法和语义。

[0111] layer_id_ext_len表示用于扩展LayerId范围的位数。

[0112] 图11和12的概念包括明确信令用于编码的视频位流内的layer_id_ext语法元素的最大值,例如,max_layer_id_ext。在此基础上,可获得在视频或元数据包的报头或有效载荷的某个部分内传输额外信息(比较上面的LengthOfExtension)所需要的最低可能的位的量,以便有效利用传输的数据量。图15提供在元数据包(例如,HEVC视频参数组扩展语法)中信令的layer_id_ext语法元素值的最大值的示例性语法和语义。

[0113] max_layer_id_ext表示在编码的视频序列内的任何薄片报头内的layer_id_ext语法元素的最大值。

[0114] 下面提供根据本发明的以上扩展的layer_id_ext语法元素的语义的两个示例性变体。

[0115] layer_id_ext与nuh_layer_id相结合使用,以识别层。使用layer_id_ext_len位,编码语法元素layer_id_ext。在不存在时,其值推断为0。

[0116] layer_id_ext与nuh_layer_id相结合使用,以识别层。使用编码的Ceil(Log2(max_layer_id_ext))位,编码语法元素layer_id_ext。在不存在时,其值推断为0。

[0117] 在图16a-16c中可扩展提供HEVC视频编解码器标准的参数组或SEI消息的层标识符,而不解析依赖性。

[0118] 在分层编码场景中,调整与层标识符或其最大数量相关的信令,以覆盖扩展的层标识符空间(比较LayerId的值范围),而非规则的明确层标识符空间(比较nuh_layer_id的值范围),如在图17中的视频参数组扩展的语法表格示例性提供的,其中,如下示例性提供变量VpsMaxNumLayers。

[0119] 规定在扩展层范围内的最大层数的变量VpsMaxNumLayer设置为等于(vps_max_layers_minus1)*$2^{layer_id_ext_len}$.+1,

[0120] 或者

[0121] 规定在扩展层范围内的最大层数的变量VpsMaxNumLayer设置为等于(vps_max_layers_minus1)*max_layer_id_ext.+1。

[0122] layer_id_in_nalu[i]规定与第i层的VCL NAL单元相关联的LayerId值的值。对于在从0到VpsMaxNumLayers-1(包括0和VpsMaxNumLayers-1)的范围内的i,在不存在时,layer_id_in_nalu[i]的值推断为等于i。

[0123] 在i大于0时,layer_id_in_nalu[i]大于layer_id_in_nalu[i-1]。

[0124] 对于在从0到vps_max_layers_minus1(包括0和vps_max_layers_minus1)的范围内的i,变量LayerIdInVps[layer_id_in_nalu[i]]设置为等于i。

[0125] dimension_id[i][j]规定第i层的第j个存在的可扩展性维度类型的标识符。用于表示dimension_id[i][j]的位数是dimension_id_len_minus1[j]+1位。对于在从0到

NumScalabilityTypes-1(包括0和NumScalabilityTypes-1)的范围内的i,在不存在时,dimension_id[i][j]推断为等于((layer_id_in_nalu[i]&((1<<dimBitOffset[j+1])-1))>>dimBitOffset[j])。

[0126] 如下获得规定第i层的第smIdx个可扩展性维度类型的标识符的变量ScalabilityId[i][smIdx]、规定第i层的视图标识符的变量ViewId[layer_id_in_nuh[i]]、以及规定第i层是否是视图可扩展性扩展层的变量ViewScalExtLayerFlag:

[0127]
```
for (i = 1; i <= VpsMaxNumLayers; i++) {
    lId = layer_id_in_nalu[i]
    for(smIdx= 0, j =0; smIdx < 16; smIdx ++)
        if((i ! = 0) && scalability_mask[smIdx])
            ScalabilityId[i][smIdx] = dimension_id[i][j++]
        else
            ScalabilityId[i][smIdx] = 0
    ViewId[lId] = ScalabilityId[i][0]
    ViewScalExtLayerFlag[lId] = (ViewId[lId] != ViewId[0] )
}
```

[0128] 自然地,上述概念也影响编解码器的现有编码器,以由该概念改进:编码器(例如,图2的编码器)配置为将视频材料编码入由一系列数据包106构成的多层数据信号104,每个数据包包括由基础层ID字段108和扩展层ID字段112构成的层识别语法元素结构,并且对于包括多个数据包的多层视频信号的预定部分113,将最大语法元素114插入多层视频信号内,表示多层视频信号的预定部分113包括的数据包的扩展层ID字段112构的最大假定值,基于最大假定值,确定在多层视频信号的预定部分内的最大层数,并且通过等于最大层数的次数,反复写入多层视频信号的层表征语法部分124,来信令在多层数据信号104内的最大层数中的每层的至少一个特征。例如,编码器延迟流104的输出,直到完成部分113的编码。

[0129] 以下附图论述信令包含在多层数据流内的数据包的层ID的概念以及在具有(由于任何原因)请求从在其中附加扩展层ID的可能性中排除基础层ID的某个值(例如,0)的必要性的情况下,实现使用扩展机构有效信令该层ID的可能性,而不危及向后兼容性。后面的必要性造成以下不对称:对于基础层ID的几乎所有可能值,附加扩展层ID是可行的。这造成以下不对称:对于基础层ID的几乎所有值,附加扩展层ID是可行的,从而造成基础层ID的这些几乎所有值的数量乘以扩展层ID能表示层ID值的能表示值的数量。然而,限制的基础层ID值不包括附加任何扩展层ID的可能性。因此,接受这个不对称,或者在使用扩展机构的情况下,不使用基础层ID的禁止值。下面陈述的解决方案通过更有效的方式解决了这个问题。

[0130] 此时,相对于HEVC标准的可能扩展,直接概述该下一个概念的描述,其中,正好应用刚刚概述的限制。然而,下面概述的概念也可传递给其他编解码器,例如,所述编解码器在任何层指示扩展机构上施加相似的限制。

[0131] 如上所述,HEVC的限制在于,不能扩展基础层。因此,为了符合HEVC规范,不能信令

层id扩展值layer_id_ext，用于具有等于0的nuh_layer_id的层，因此，与等于0的nuh_layer_id相关联的集群可以仅仅包括基础层本身。在不同集群内的这种不相等数量的层可能不可取。

[0132] 该问题可以由下述概念解决，该概念通过与上述方式不同的方式，获得集群以及在集群内的层id扩展值。由于等于0的nuh_layer_id不能具有多层，所以没有多层的限制扩展为不等于0的其他nuh_layer_id值。然后，属于这些限制的nuh_layer_id值的层与未限制的nuh_layer_id值的层组合，以形成具有相等层数的集群。

[0133] 基本理念在于，nuh_layer_id分成两个值，例如，通过位屏蔽。一个值用于确定集群。这样做，不同的nuh_layer_id值可以属于相同的集群。另一个值用于确定layer_id_ext值的存在(因此，用于确定多层是否可以属于nuh_layer_id值)并且确定在LayerId推导中与layer_id_ext_value的额外偏移。

[0134] 在下面提供并且在图18中说明基本理念的实施方式的实例。因此，图18提供在layer_id_ext、nuh_layer_id以及LayerId之间的关系的实例，用于max_layer_id_ext等于3的示例性情况(LengthOfExtension等于2)。

[0135] 仅仅在实例中，6位nuh_layer_id置的5个最高有效位用于确定集群，因此，具有不同最低有效位的两个nuh_layer_id值属于一个集群。

[0136] 如果最低有效位等于0，那么不为nuh_layer_id信令layer_id_ext值，并且仅仅一个层属于nuh_layer_id。按照定义，该层是在由5个最高有效位规定的集群内具有等于0的layer_id_ext的层。否则，如果最低有效位等于1，那么多层属于nuh_layer_id值。对于每层，明确信令layer_id_ext，并且每层属于由5个最高有效位规定的集群。

[0137] 图19的语法表格可能用于该实施方式。示出了数据包206的报头，即，薄片段报头，或者更确切地说，其片段。

[0138] 在nuh_layer_id的最低有效位等于0时，不存在Layer_id_ext。在不存在时，推断出等于0。如下确定变量LayerId，其中，LengthOfExtension是在位方面的layer_id_ext语法元素的长度。

[0139] LayerId＝((nuh_layer_id>>1)<<LengthOfExtension)+layer_id_ext

[0140] 在另一个实施方式中，在位流内，例如，在VPS内，信令变量add_layer_id_ext_len。(请注意，上述实施方式是add_layer_id_ext_len等于1的该实施方式的特殊情况)。6位nuh_layer_id语法元素的6-add_layer_id_ext_len)最高有效位用于确定集群。最低有效位add_layer_id_ext_len规定是否推断layer_id_ext，而且，固定添加到layer_id_ext值中的偏移。图20a和20b的语法表格用于演示该实施方式：

[0141] 在该实施方式中，在nuh_layer_id的add_layer_id_ext_len最低有效位等于1时，仅仅存在layer_id_ext。在不存在layer_id_ext时，推断出等于0。如下确定变量LayerId，其中，LengthOfExtension是在位方面的layer_id_ext语法元素的长度。

[0142] LayerId＝((nuh_layer_id>>add_layer_id_ext_len)

[0143] <<LengthOfExtension)+layer_id_ext+layerIdExtOffset

[0144] 因此，根据上面相对于图18到图20b概述的概念，装置(例如，多层视频解码器的网络元件(例如，在图2中描述的解码器))配置为处理多层数据流，使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入所述多层数据流，该装置进行以下操作，对

于所述多层数据流包括的每个数据包，每个数据包与不同层中的一层相关联，如上所述。

[0145] 具体而言，如图21中所示，装置从多层数据流中(或者更确切地说，从当前数据包中)读取基础层ID字段308。基础层ID字段包括第一子字段308a和第二子字段308b。例如，第一子字段可由字段308的一个或多个最低有效位构成，而第二子字段308b由字段308的其他最高有效位构成。如在图20b的实例中所指示的，称为add_layer_id_ext_len 310的更高级语法元素可限定子字段308a的长度。

[0146] 在步骤312中读取基础层ID字段308之后，装置在步骤314中检查子字段308a是否满足预定标准。例如，如果子字段是1或者是“全部1序列”，那么满足该标准。在满足预定标准的情况下，装置在步骤318中从数据流中(或者更确切地说，从当前数据包中)读取扩展层ID字段316，并且在步骤320中使用扩展层ID字段获得扩展值，以便扩展值位于扩展值的域的第一子集内。在步骤320中的推导可对应于在字段316内的位表示的单纯复制。数据流的要求可能在于，在扩展层ID字段316内的值不采用某个值，例如，0。为基础层ID字段308子字段308a不满足下面进一步概述的标准的情况，预留后一个值。

[0147] 在步骤320之后，通过在集群值的域上从第二子字段308b的可能值的域中映射，装置从第二子字段308b中获得集群值。然后，在步骤324中，使用在步骤320和322中获得的集群和扩展值，索引与当前数据包相关联的层(即，包括字段308和316的层)。例如，集群和扩展值彼此仅仅串联，例如，使用最高有效位的集群值。例如，基于子字段308b的集群值322的推导还可由子字段308b的位的直接使用构成，作为集群值的二进制表示。

[0148] 然而，如果在步骤314中，检查造成未满足预定标准，那么装置抑制从用于当前数据包的多层数据流中读取扩展层ID字段。确切地说，在步骤326中，该装置在与扩展值的域的第一子集不相交的值上设置扩展值，例如，通过将扩展值设置为0，并且通过在集群值的域上从第二子字段308b的可能值的域中映射，从第二子字段308b中获得集群值。使用相同的映射，可执行在步骤322和328中的推导。然后，从在步骤326和328中设置和推导的集群和扩展值中，确定当前数据包的层ID，其中，再次通过相同的方式执行层ID的确定，犹如在步骤320和322中获得集群和扩展值。

[0149] 使图21的描述与图18的实施方式相关，因此，第一子字段308a可能是1位长，并且具体而言，可能是字段308的LSB。由于在图18的情况下，layer_id_ext(即，字段316)是2位长，所以如果赢得该LSB 308a，并且扩展层ID字段316跨过的扩展值的上述域是{0,1,2,3}，那么可满足预定标准。因此，扩展值的域的第一子集(即，在步骤320中的推导的上域)是{1,2,3}。具体而言，在图18的实例中，在步骤320中的推导通过应用于扩展层ID字段中的单式矩阵体现。可替换地，可使用某个其他计算，例如，将第一子字段308a的值加入单式矩阵应用的结果中。因此，在步骤326中，扩展值设置为与{1,2,3}不相交的值，即，0。在推导322和328中使用的映射可相同，但是可替换地不同。具体而言，映射是双射的。

[0150] 在324中的索引的确定可使用集群值来设置更高有效数位，并且使用扩展值来设置更低有效数位，如在图18中所述。因此，如在图18中所述，在nuh_layer_id的LSB(即，基础层ID字段308)是1的情况下，仅仅存在layer_id_ext字段(即，扩展层ID字段316)。否则，layer_id_ext字段推断为0。因此，层ID到layer_id_ext和nuh_layer_id的组合的映射完全对称，并且没有未使用的位表示。通过传输layer_id_ext，layer_id_ext的推断值可用于数据流中，以便用于传输layer_id_ext的在数据流内跨过的位减少，以便解释layer_id_ext

实际上仅仅采用3个不同值{1,2,3}这一事实。例如，二进制一元码可用于在二进制串上映射3个元素{1,2,3}，例如，然后，所述二进制串编码入数据流，并且使用熵编码/解码，从中解码。

[0151] 因此，尤其地，相对于后一个概念描述了一种装置，该装置配置为处理多层数据流，使用层间预测在对应于信息量的不同等级的不同层上将视频材料编码入所述多层数据流，所述多层数据流包括多个数据包，每个数据包与所述不同层中的一层相关联，所述装置配置为对于多层数据流的每个数据包，从所述多层数据流中读取基础层ID字段，所述基础层ID字段包括第一子字段和第二子字段；检查基础层ID字段的第一子字段是否满足预定标准，并且

[0152] -如果基础层ID字段的第一子字段满足预定标准，那么从所述多层数据流中读取扩展层ID字段，使用扩展层ID字段，获得扩展值，以便扩展值位于扩展值的域的第一子集内，并且通过在集群值的域上从第二子字段的可能值的域中映射，从第二子字段中获得集群值；并且

[0153] -如果基础层ID字段的第一子字段不满足预定标准，那么抑制从所述多层数据流中读取扩展层ID字段，将扩展值设置为与扩展值的域的第一子集不相交的值，并且通过在集群值的域上从第二子字段的可能值的域中映射，从第二子字段中获得集群值，

[0154] 其中，装置使用集群和扩展值索引与相应数据包相关联的层。应注意的是，可从所述多层数据流中读取预定标准。如上所述，“第一子字段”可能是1位长，并且具体而言，LSB；“预定标准”可能是“如果一个”，“扩展值的域”可能是(例如){0,1,2,3}，“第一子字段”可能是{1,2,3}，“使用扩展层ID字段”可能是应用于扩展层ID字段中的单式矩阵，可替换地：将第一子字段的值加入单式矩阵应用的结果中，“将扩展值设置为不相交的值”可能是“到0”的设置；“在集群值的域上从第一子字段的可能值的域中映射”可能对于这两种情况相同，但是可替换地不同；映射可能是双射的，并且“使用集群和扩展值索引与相应数据包相关联的层”可涉及使用(例如)用于设置更高有效数位的集群值以及用于设置更低有效数位的扩展值。

[0155] 后一种概念直接转化成编码器，该编码器配置为使用层间预测在对应于信息量的不同等级的不同层上将视频材料202编码入多层数据流204，所述多层数据流包括多个数据包206，每个数据包与所述不同层中的一层相关联，其中，与相应数据包相关联的层由集群和扩展值独特地相关联，编码器配置为对于多层数据流的每个数据包206，如果扩展值在扩展值的域的第一子集内，那么在所述多层数据流内插入基础层ID字段308，所述基础层ID字段包括第一子字段308a和第二子字段308b，使用集群值设置第二子字段，并且设置第一子字段，以便满足预定条件；并且在所述多层数据流内插入扩展层ID字段，使用扩展值设置扩展层ID；并且如果扩展值等于与扩展值的域的第一子集不相交的值，那么抑制将扩展层ID字段插入并且将基础层ID字段308插入312多层数据流内，基础层ID字段包括第一子字段308a和第二子字段308b，使用集群值设置第二子字段，并且设置第一子字段，以便不满足预定条件。即，编码器生成数据流，以便解码器通过上述方式获得正确的层ID，并且为此，使用逆映射，与相对于解码或接收侧的上述映射相比较。

[0156] 概述以上描述，该描述提供了扩展机构，用于增强层数据的层标识符，将层标识符分成集群，并且有效地信令在集群内的编码依赖性。

[0157] 上面陈述的细节在分层编码场景中可适用于增强层的数据，而基础层的数据保持不变。可在基础和增强层数据包的共同报头语法中，通过层标识符(比较nuh_layer_id)识别分层编码场景的基础层。

[0158] 在以上实例中，在接收侧上的相应装置可能是视频解码器。因此，在操作时，视频解码器能够选择由(扩展的)layerID识别的位流的数据包，用于解码。然而，可替换地，该装置可以是能够基于外部因素(例如，网络利用率、关于解码器的知识等)丢弃由(扩展的)layerID识别的位流的数据包的网络元件。

[0159] 在所有以上实施方式中，所述多层视频信号可具有使用层间预测在不同层上编码入所述多层视频信号的视频材料，以便从另一层中层间预测的任何层增加以下中的一个或多个：进一步视图；深度信息；表面反射率信息；颜色分量信息；空间分辨率细化；以及SNR分辨率细化。

[0160] 虽然在设备的背景下描述了一些方面，但是显然，这些方面也表示相应方法的描述，其中，块体或装置对应于方法步骤或者方法步骤的特征。同样，在方法步骤的背景下描述的方面也表示相应设备的相应块体或物品或特征的描述。一些或所有方法步骤可由(或使用)硬件设备执行，例如，微处理器、可编程计算机或电子电路。在一些实施方式中，某一个或多个最重要的方法步骤可由这种设备执行。

[0161] 上面提及的本发明的编码信号可以储存在数字储存介质上或者可以通过传输介质传输，例如，无线传输介质或有线传输介质，例如，互联网。

[0162] 根据某些实现方式要求，本发明的实施方式可以在硬件内或者在软件内实现。可以使用数字存储器介质，执行实施方式，例如，软盘、DVD、蓝光光盘、CD、ROM、PROM、EPROM、EEPROM或FLASH存储器，电子可读控制信号储存在该数字存储器介质其上，这些信号与可编程计算机系统配合(或者能够与其配合)，以便执行相应方法。因此，数字存储器介质内可能是计算机可读的。

[0163] 根据本发明的一些实施方式包括数据载体，具有电子可读控制信号，这些信号能够与可编程的计算机系统配合，以便执行在本文中描述的一种方法。

[0164] 通常，可以实现本发明的实施方式，作为具有程序代码的计算机程序产品，计算机程序产品在计算机上运行时，程序代码有效地用于执行一种方法。例如，程序代码可储存在机器可读载体上。

[0165] 其他实施方式包括用于执行在本文中描述的一种方法的计算机程序，所述计算机程序储存在机器可读载体上。

[0166] 换言之，因此，本发明方法的一个实施方式是具有程序代码的计算机程序，计算机程序在计算机上运行时，用于执行在本文中描述的一种方法。

[0167] 因此，本发明方法的进一步实施方式是数据载体(或数字存储器介质或计算机可读介质)，在该数据载体上记录用于执行在本文中描述的一种方法的计算机程序。数据载体、数字存储器介质或记录介质通常是有形和/或非过渡的。

[0168] 因此，本发明方法的进一步实施方式是数据流或一系列信号，表示用于执行在本文中描述的一种方法的计算机程序。例如，数据流或这系列信号可配置为通过数据通信连接来传输，例如，通过互联网。

[0169] 进一步实施方式包括处理装置，例如，计算机或可编程逻辑装置，配置为或者适合

于执行在本文中描述的一种方法。

[0170] 进一步实施方式包括计算机，在该计算机上安装用于执行在本文中描述的一种方法的计算机程序。

[0171] 根据本发明的进一步实施方式包括设备或系统，配置为将用于执行在本文中描述的一种方法的计算机程序传输(例如，电子地或光学地)给接收器。例如，接收器可能是计算机、移动装置、存储器装置等。例如，该设备或系统可包括文件服务器，用于将计算机程序传输给接收器。

[0172] 在一些实施方式中，可编程逻辑装置(例如，现场可编程门阵列)可用于执行在本文中描述的方法的一些或所有功能。在一些实施方式中，现场可编程门阵列可与微处理器配合，以便执行在本文中描述的一种方法。通常，这些方法优选地由任何硬件设备执行。

[0173] 上述实施方式仅仅说明本发明的原理。要理解的是，对于本领域的技术人员，在本文中描述的设置和细节的修改和变化显而易见。因此，其目的在于，仅仅由即将发生的专利权利要求的范围限制限制，而不由通过在本文中的实施方式的描述和解释提出的具体细节限制。

[0174] 参考文献

[0175] [1]B.Bross,W.-J.Han,J.-R.Ohm,G.J.Sullivan,T.Wiegand(Eds.),“High Efficiency Video Coding(HEVC)text specification draft 10”,JCTVC-L1003,Geneva,CH,Jan.2013

[0176] [2]G.Tech,K.Wegner,Y.Chen,M.Hannuksela,J.Boyce(Eds.),“MV-HEVC Draft Text 3(ISO/IEC 23008-2PDAM2)”,JCT3V-C1004,Geneva,CH,Jan.2013

[0177] [3]G.Tech,K.Wegner,Y.Chen,S.Yea(Eds.),“3D-HEVC Test Model Description,draft specification”,JCT3V-C1005,Geneva,CH,Jan.2013

[0178] [4]WILBURN,Bennett,et al.High performance imaging using large camera arrays.ACM Transactions on Graphics,2005,24.Jg.,Nr.3,S.765-776.

[0179] [5]WILBURN,Bennett S.,et al.Light field video camera.In:Electronic Imaging 2002.International Society for Optics and Photonics,2001.S.29-36.

[0180] [6]HORIMAI,Hideyoshi,et al.Full-color 3D display system with 360degree horizontal viewing angle.In:Proc.Int.Symposium of 3D and Contents.2010.S.7-10.

nal_unit_header( ) {	描述符
forbidden_zero_bit	f(1)
nal_unit_type	u(6)
nuh_layer_id	u(6)
nuh_temporal_id_plus1	u(3)
}	

图1

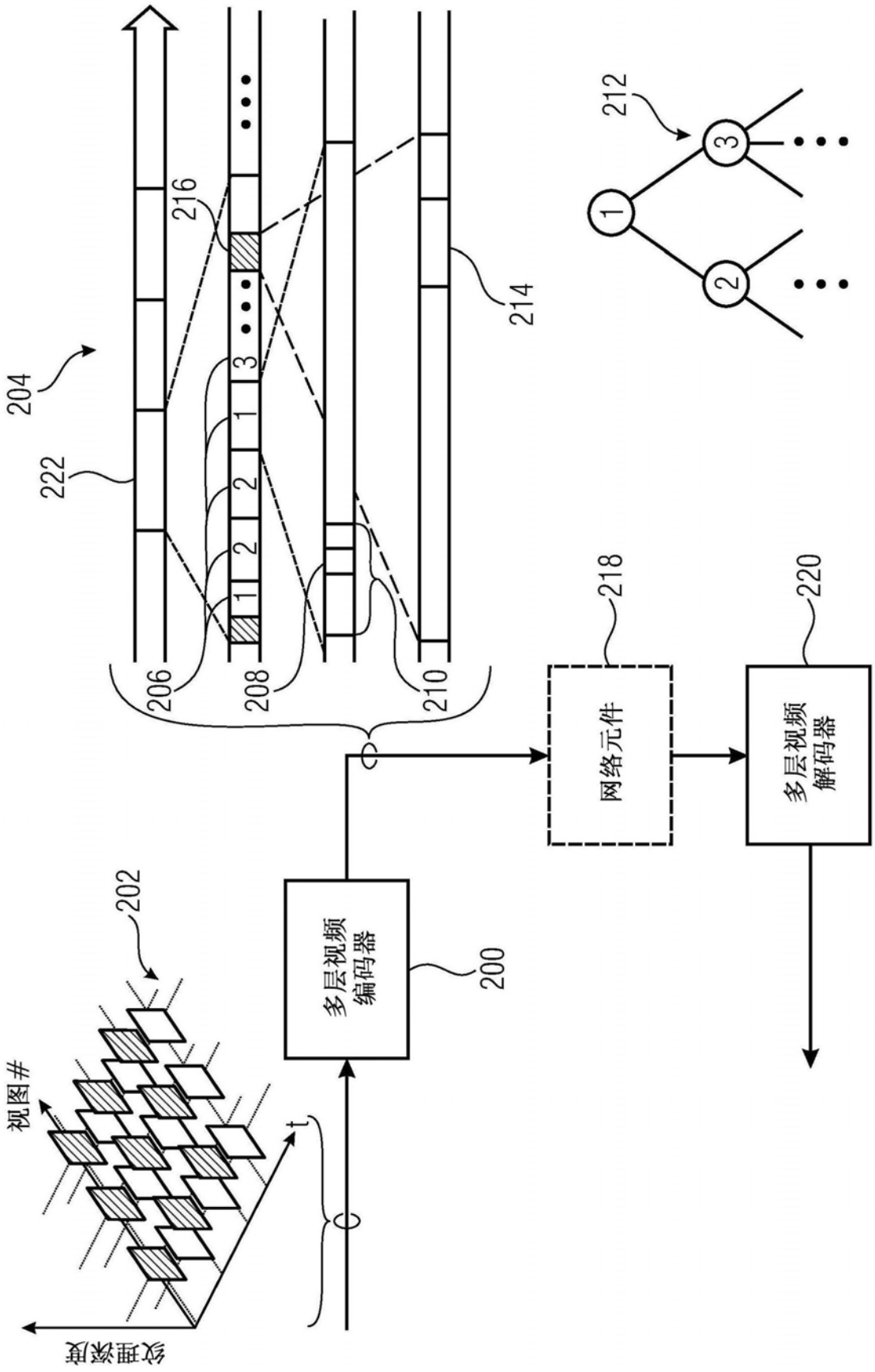

纹理深度
视图#
t
202
多层视频编码器
200
网络元件
218
多层视频解码器
220
204
222
206
208
210
214
216
212
1
2
3

图2

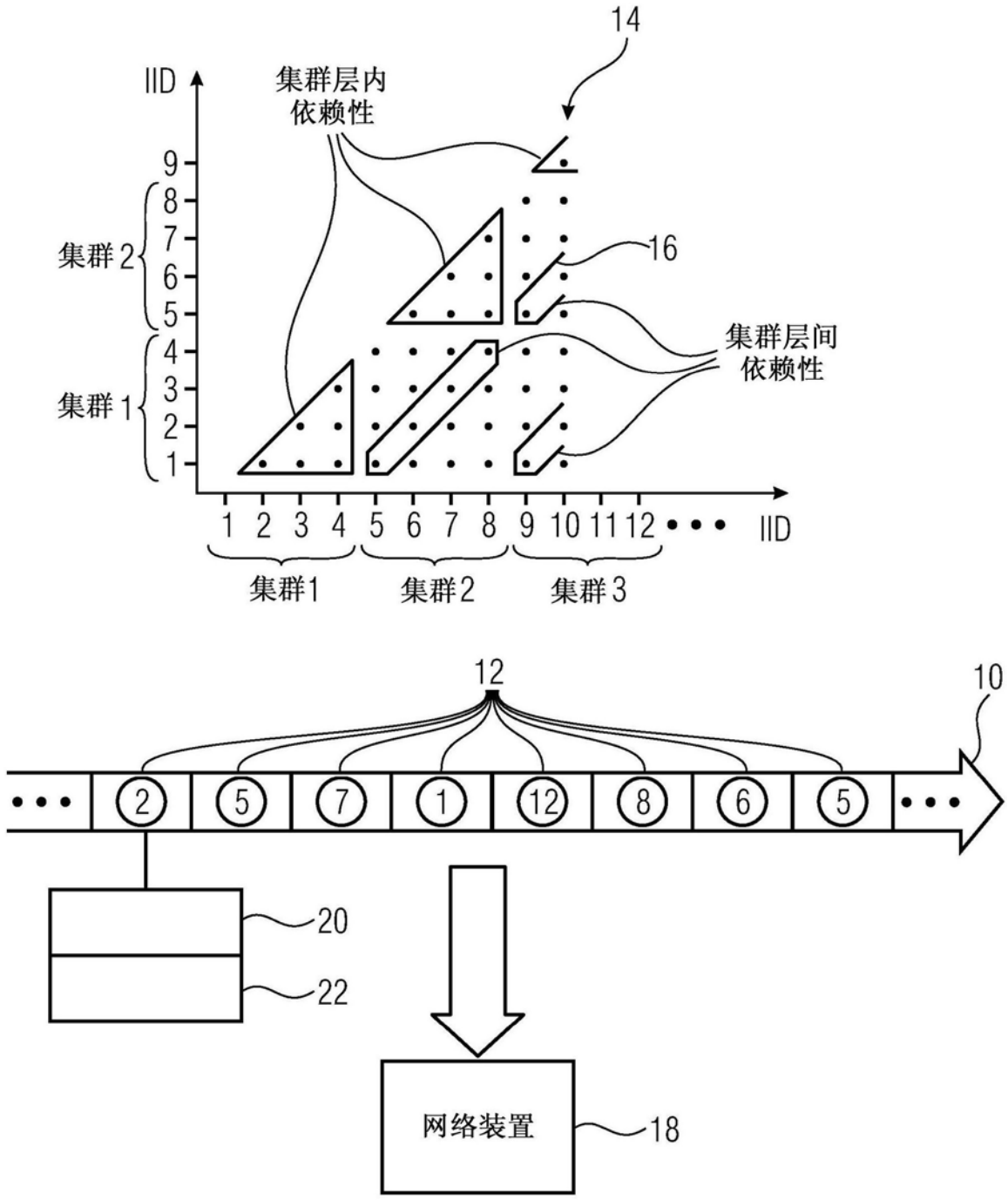

14
IID
集群层内
依赖性
9
8
7
6
5
集群2
4
3
2
1
集群1
16
集群层间
依赖性
1 2 3 4 5 6 7 8 9 10 11 12
IID
集群1
集群2
集群3
12
10
2
5
7
1
12
8
6
5
20
22
网络装置
18

图3

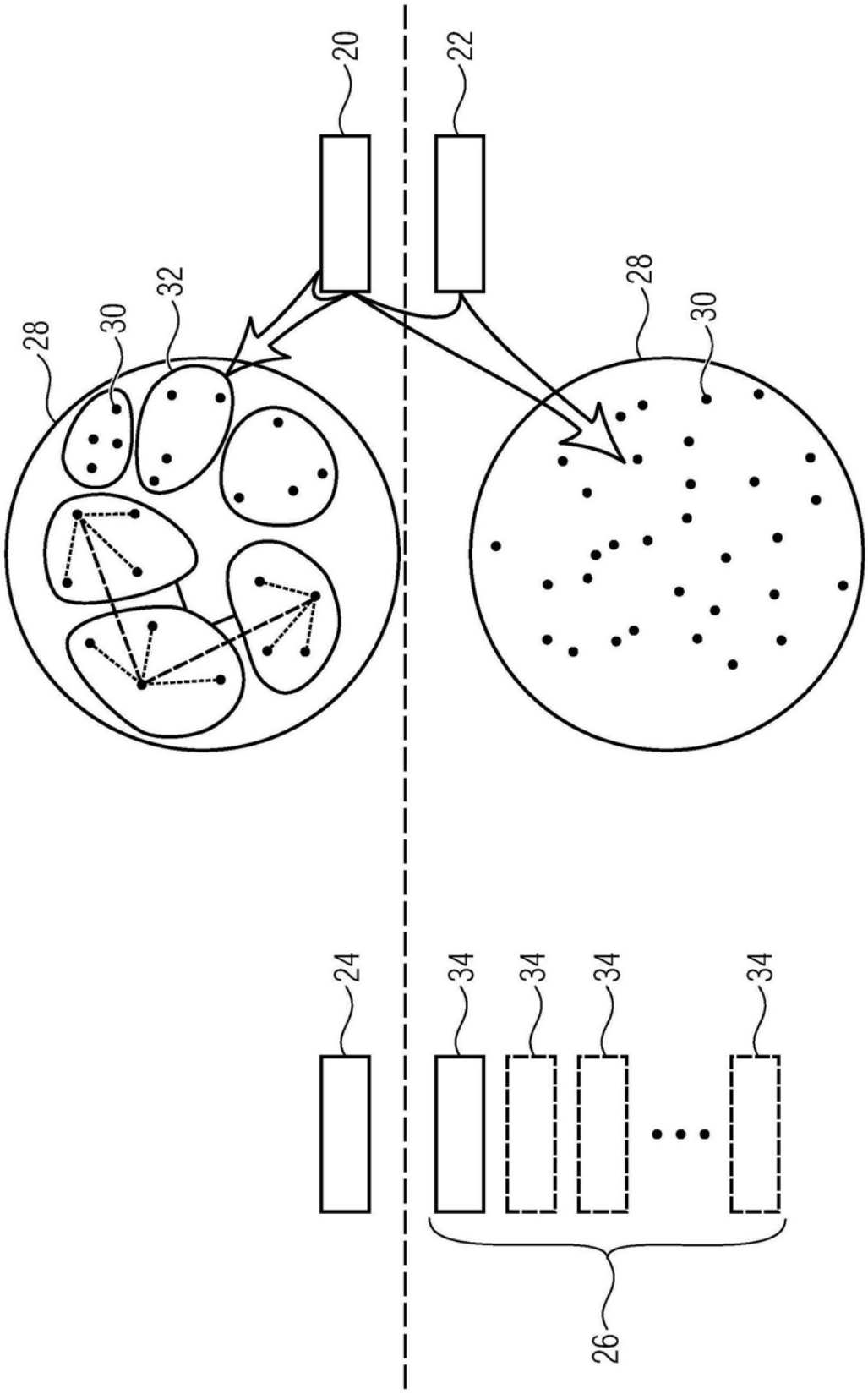


图4

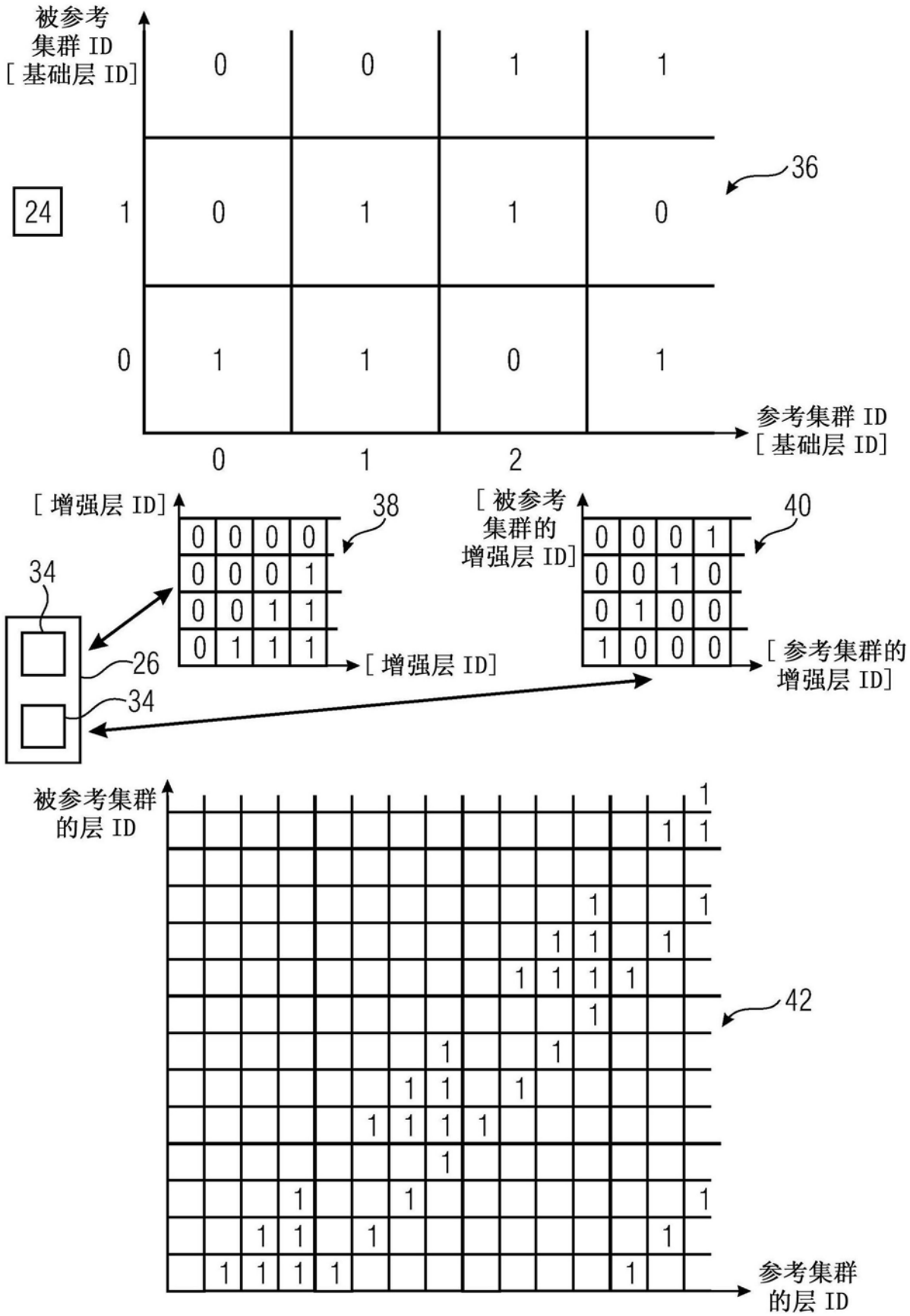
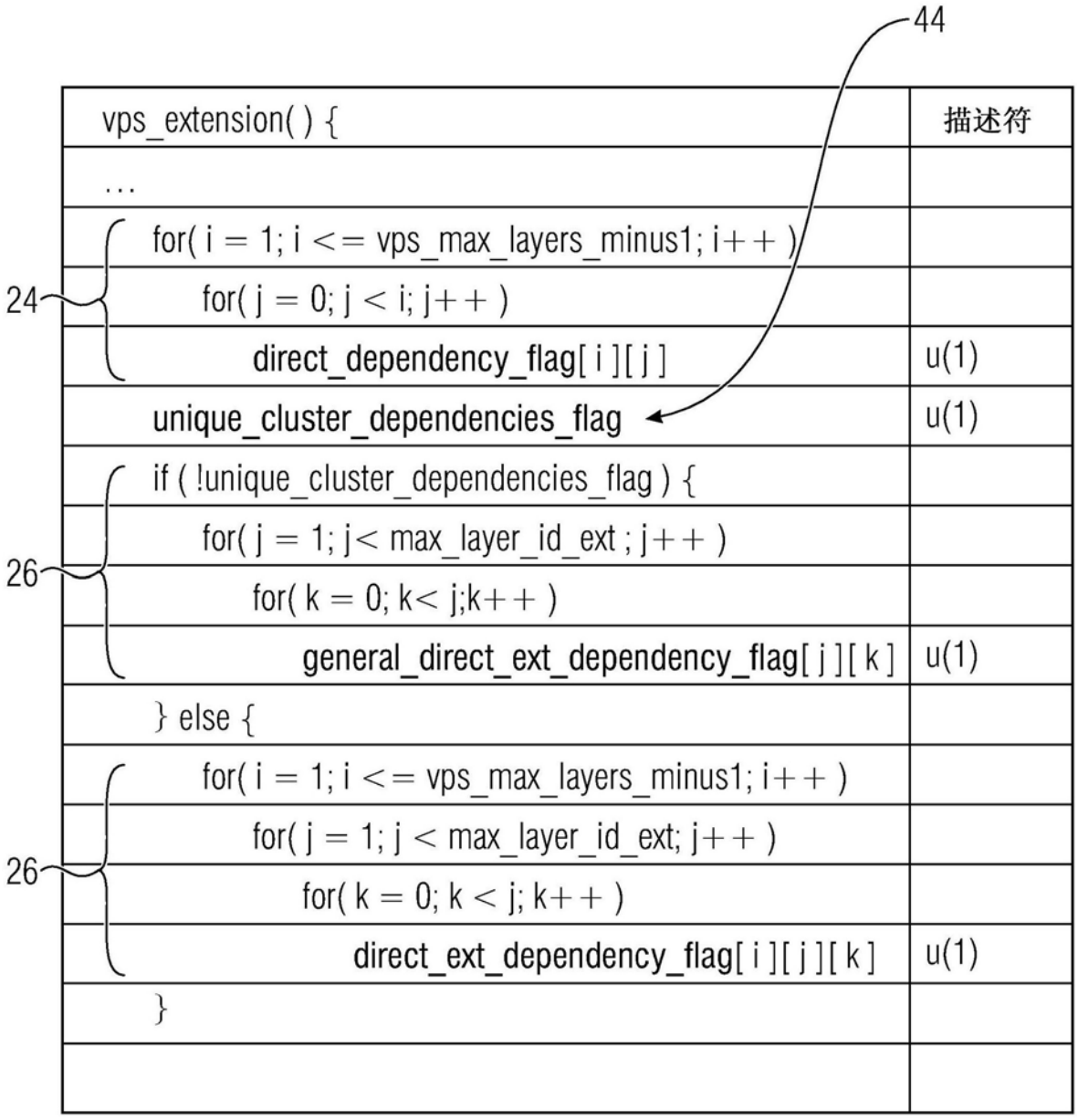

被参考
集群 ID
[基础层 ID]
0 0 1 1
1 0 1 1 0
0 1 1 0 1
0 1 2
参考集群 ID
[基础层 ID]
24
36
[增强层 ID]
38
0 0 0 0
0 0 0 1
0 0 1 1
0 1 1 1
[增强层 ID]
34
26
34
[被参考
集群的
增强层 ID]
40
0 0 0 1
0 0 1 0
0 1 0 0
1 0 0 0
[参考集群的
增强层 ID]
被参考集群
的层 ID
42
参考集群
的层 ID

图5

vps_extension( ) {	描述符
...	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
for(i = 1; i < (1 << layer_id_ext_len) ; i++)	
for(j = 0; j < i; j++)	
direct_ext_dependency_flag[i][j]	u(1)

图6



vps_extension( ) {	描述符
...	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
unique_cluster_dependencies_flag	u(1)
if (!unique_cluster_dependencies_flag) {	
for(j = 1; j< max_layer_id_ext ; j++)	
for(k = 0; k< j;k++)	
general_direct_ext_dependency_flag[j][k]	u(1)
} else {	
for(i = 1; i <= vps_max_layers_minus1; i++)	
for(j = 1; j < max_layer_id_ext; j++)	
for(k = 0; k < j; k++)	
direct_ext_dependency_flag[i][j][k]	u(1)
}	

图7

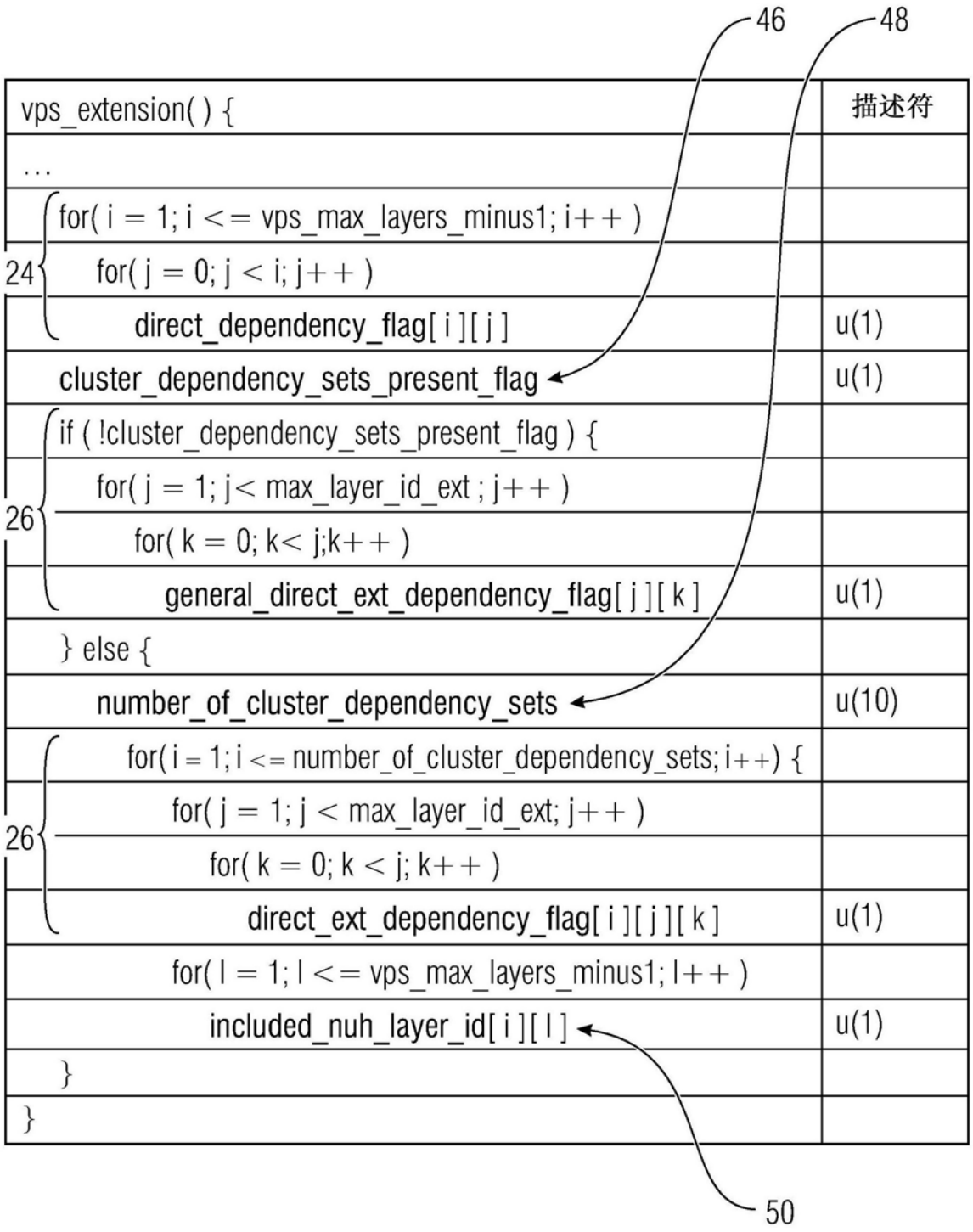

46
48
vps_extension( ) {
描述符
...
for(i = 1; i <= vps_max_layers_minus1; i++)
24
for(j = 0; j < i; j++)
direct_dependency_flag[i][j]
u(1)
cluster_dependency_sets_present_flag
u(1)
if (!cluster_dependency_sets_present_flag) {
for(j = 1; j< max_layer_id_ext ; j++)
26
for(k = 0; k< j;k++)
general_direct_ext_dependency_flag[j][k]
u(1)
} else {
number_of_cluster_dependency_sets
u(10)
for(i = 1; i <= number_of_cluster_dependency_sets; i++) {
for(j = 1; j < max_layer_id_ext; j++)
26
for(k = 0; k < j; k++)
direct_ext_dependency_flag[i][j][k]
u(1)
for(l = 1; l <= vps_max_layers_minus1; l++)
included_nuh_layer_id[i][l]
u(1)
}
}
50

图8

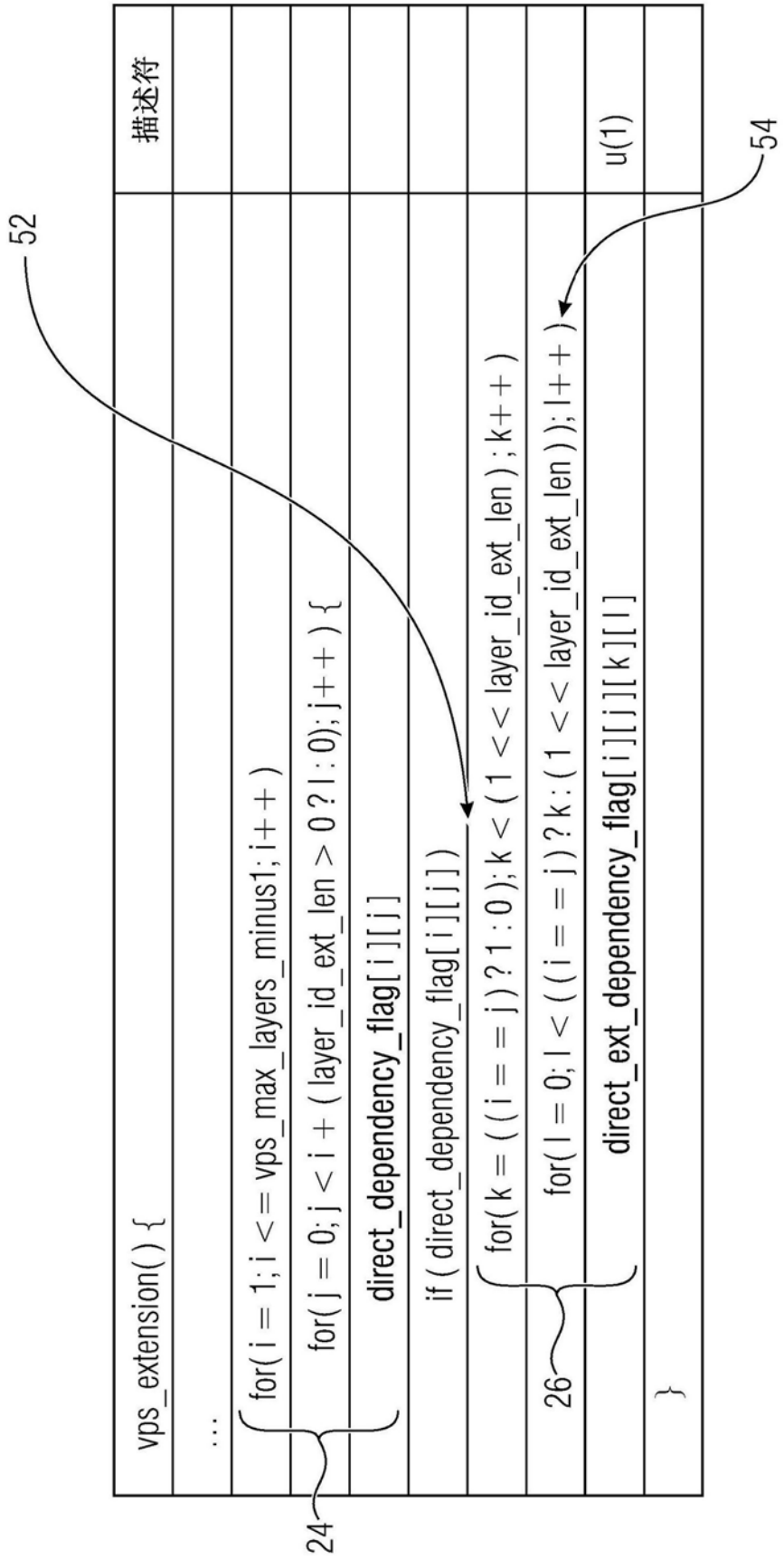

描述符
vps_extension( ) {
...
for(i = 1; i <= vps_max_layers_minus1; i++)
for(j = 0; j < i + (layer_id_ext_len > 0 ? 1 : 0); j++) {
direct_dependency_flag[i][j]
if (direct_dependency_flag[i][j])
for(k = ((i == j) ? 1 : 0); k < (1 << layer_id_ext_len) ; k++)
for(l = 0; l < ((i == j) ? k : (1 << layer_id_ext_len)); l++)
direct_ext_dependency_flag[i][j][k][l]
u(1)
}
52
54
24
26

图9

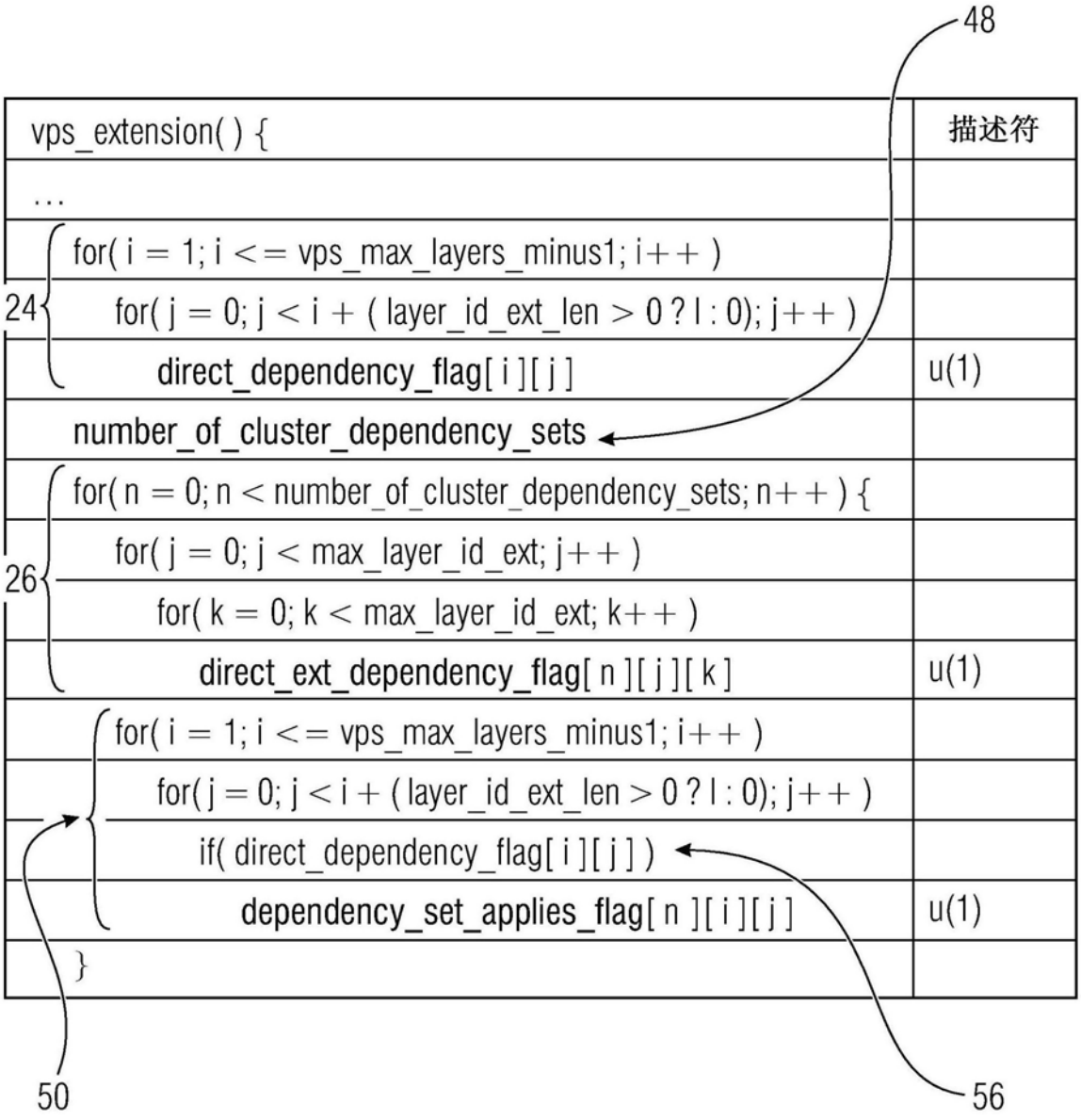
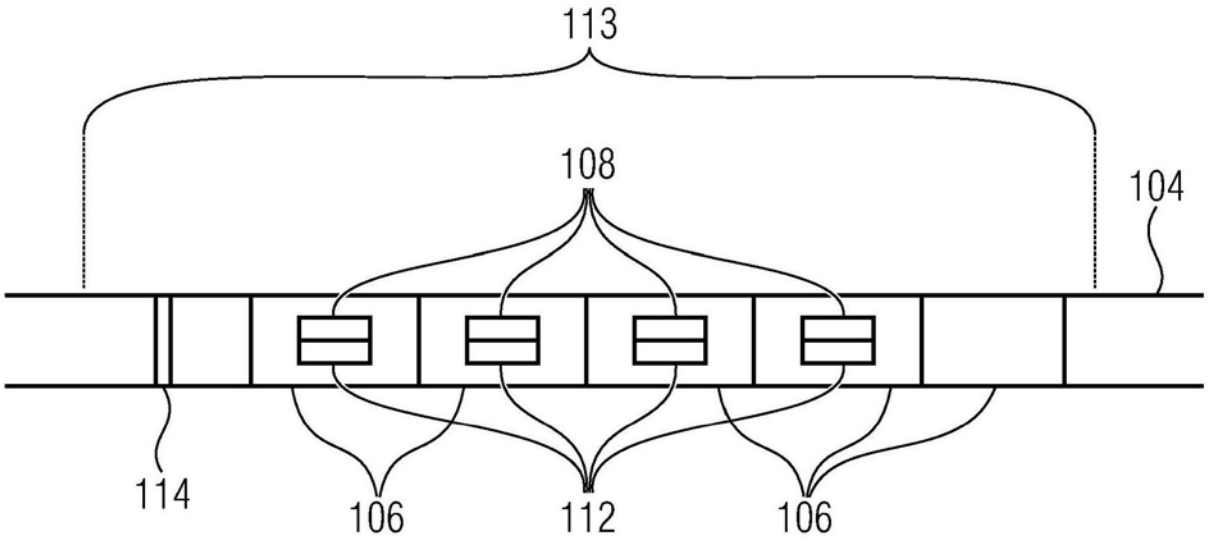

48
vps_extension( ) {
描述符
...
for(i = 1; i <= vps_max_layers_minus1; i++)
for(j = 0; j < i + (layer_id_ext_len > 0 ? l : 0); j++)
24
direct_dependency_flag[i][j]
u(1)
number_of_cluster_dependency_sets
for(n = 0; n < number_of_cluster_dependency_sets; n++) {
for(j = 0; j < max_layer_id_ext; j++)
26
for(k = 0; k < max_layer_id_ext; k++)
direct_ext_dependency_flag[n][j][k]
u(1)
for(i = 1; i <= vps_max_layers_minus1; i++)
for(j = 0; j < i + (layer_id_ext_len > 0 ? l : 0); j++)
if(direct_dependency_flag[i][j])
dependency_set_applies_flag[n][i][j]
u(1)
}
50
56

图10



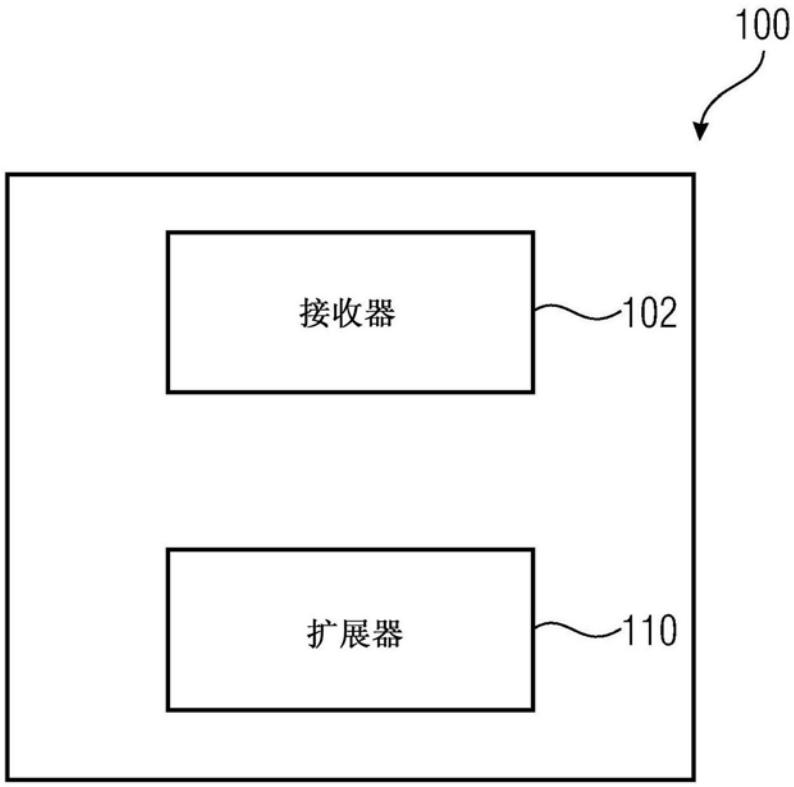


图11

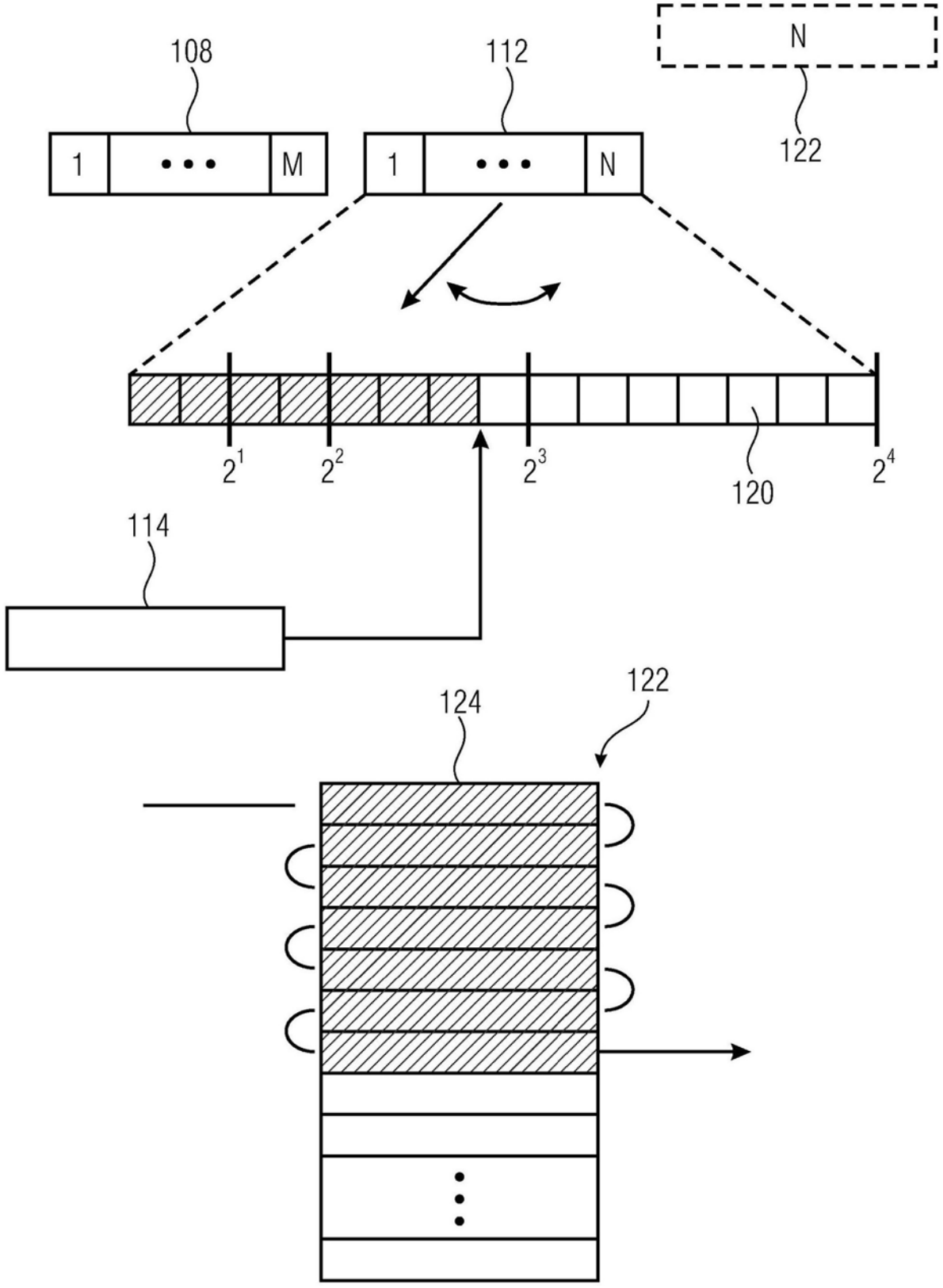


图12

slice_segment_header( ) {	描述符
first_slice_segment_in_pic_flag	u(1)
…	
if(!dependent_slice_segment_flag) {	
if (nuh_layer_id > 0) {	
layer_id_ext	u(v)
}	
for(i = 0; i < (num_extra_slice_header_bits); i++)	
slice_reserved_flag[i]	u(1)
…	
}	

图13

video_parameter_set_extension( ) {	描述符
…	
layer_id_ext_len	u(3)
…	
}	

图14

video_parameter_set_extension( ) {	描述符
…	
max_layer_id_ext	u(3)
…	
}	

图15

generic_sei_extension( ) {	描述符
...	
layer_id_ext	ue(v)
...	
}	

图16A

sps_extension( ) {	描述符
...	
layer_id_ext	ue(v)
...	
}	

图16B

pps_extension( ) {	描述符
...	
layer_id_ext	ue(v)
...	
}	

图16C

vps_extension( ) {	描述符
...	
Vps_nuh_layer_id_present_flag	u(1)
for(i = 1; i <= VpsMaxNumLayers - 1; i++) {	
if(vps_nuh_layer_id_present_flag)	
layer_id_in_nalu[i]	u(v)
if(!splitting_flag)	
for(j = 0; j < NumScalabilityTypes; j++)	
dimension_id[i][j]	u(v)
}	
for(i = 1; i <= VpsMaxNumLayers - 1; i++)	
for(j = 0; j < i; j++)	
direct_dependency_flag[i][j]	u(1)
for(i = 0; i< VpsMaxNumLayers - 1; i++)	
max_tid_il_ref_pics_plus1[i]	u(3)
vps_number_layer_sets_minus1	u(10)
vps_num_profile_tier_level_minus1	u(6)
for(i = 1; i <= VpsMaxNumLayers - 1; i ++) {	
vps_profile_present_flag[i]	u(1)
if(!vps_profile_present_flag[i])	
profile_ref_minus1[i]	u(6)
profile_tier_level(vps_profile_present_flag[i], vps_max_sub_layers_minus1)	
}	
...	
direct_dep_type_len_minus2	ue(v)
for(i = 1; i <= VpsMaxNumLayers - 1; i++)	
for(j = 0; j < i; j++)	
if(direct_dependency_flag[i][j])	
direct_dependency_type[i][j]	u(v)
...	
}	

图17

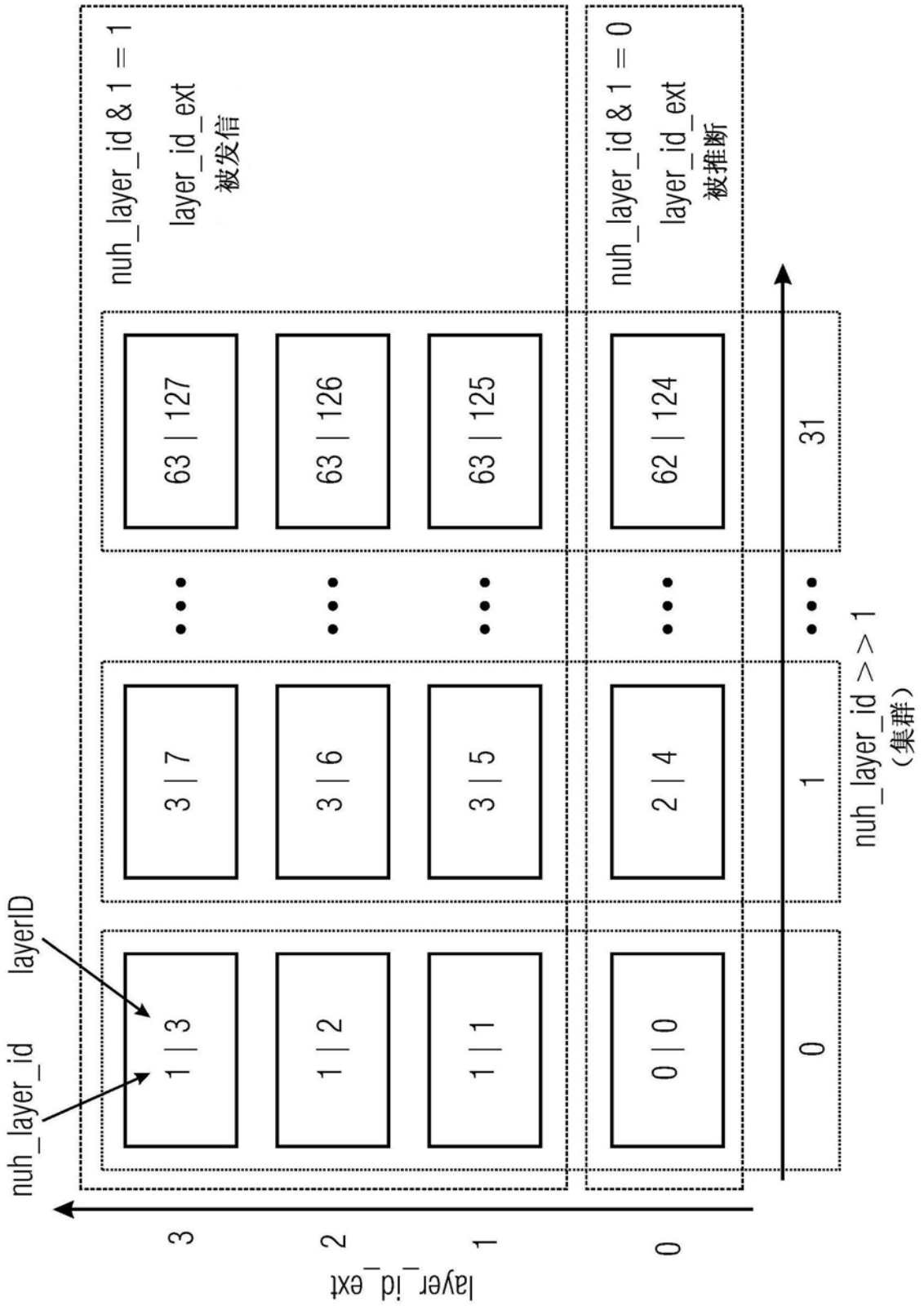

nuh_layer_id & 1 = 1
layer_id_ext
被发信
nuh_layer_id & 1 = 0
layer_id_ext
被推断
63 | 127
63 | 126
63 | 125
62 | 124
3 | 7
3 | 6
3 | 5
2 | 4
1 | 3
1 | 2
1 | 1
0 | 0
31
1
0
nuh_layer_id >> 1
（集群）
layerID
nuh_layer_id
layer_id_ext
3
2
1
0

图18

slice_segment_header( ) {	描述符
first_slice_segment_in_pic_flag	u(1)
...	
if(!dependent_slice_segment_flag) {	
if (nuh_layer_id & > 0) {	
layer_id_ext	u(v)
}	
for(i = 0; i < (num_extra_slice_header_bits); i++)	
slice_reserved_flag[i]	u(1)
...	
}	

图19

video_parameter_set_extension( ) {	描述符
...	
add_layer_id_ext_len	u(3)
...	
}	

图20A

slice_segment_header( ) {	描述符
first_slice_segment_in_pic_flag	u(1)
...	
if(!dependent_slice_segment_flag) {	
maxLayerIdExtOffset = (1<<add_layer_id_ext_len) - 1	
layerIdExtOffset = nuh_layer_id & maxLayerIdExtOffset	
if (layerIdExtOffset = = maxLayerIdExtOffset) {	
layer_id_ext	u(v)
}	
for(i = 0; i < (num_extra_slice_header_bits); i++)	
slice_reserved_flag[i]	u(1)
...	
}	

图20B

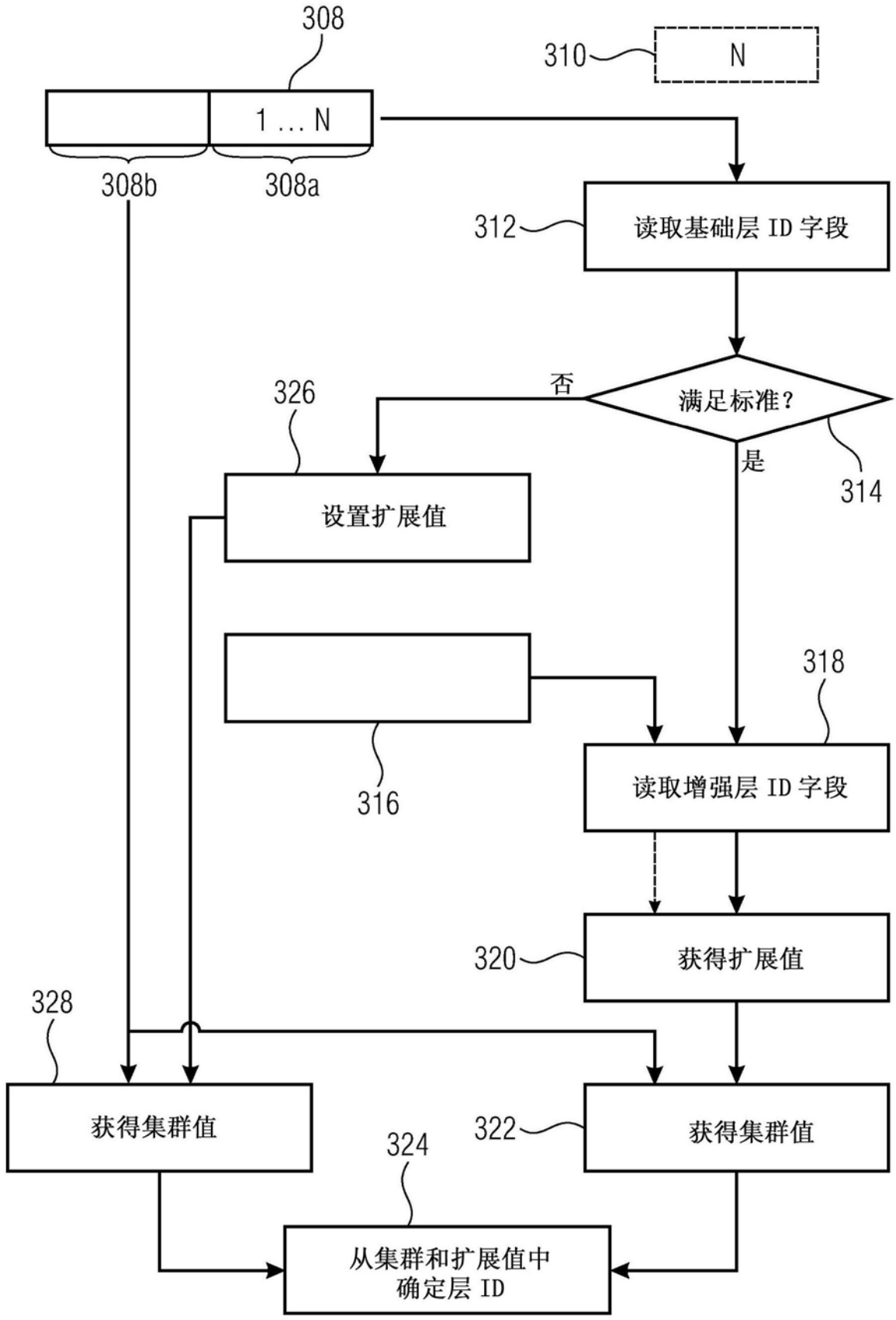

308
310
N
1 ... N
308b
308a
312
读取基础层 ID 字段
否
满足标准？
326
是
314
设置扩展值
316
318
读取增强层 ID 字段
320
获得扩展值
328
获得集群值
322
获得集群值
324
从集群和扩展值中
确定层 ID

图21