



(12)发明专利申请

(10)申请公布号 CN 110915219 A

(43)申请公布日 2020.03.24

(21)申请号 201880046722.7

(22)申请日 2018.06.25

(30)优先权数据

17305928.8 2017.07.13 EP

(85)PCT国际申请进入国家阶段日

2020.01.13

(86)PCT国际申请的申请数据

PCT/EP2018/066906 2018.06.25

(87)PCT国际申请的公布数据

WO2019/011634 EN 2019.01.17

(71)申请人 交互数字VC控股公司

地址 美国特拉华州

(72)发明人 S.拉塞尔 M-L.查姆佩尔 蔡康颖

(74)专利代理机构 北京市柳沈律师事务所

11105

代理人 刘茵

(51)Int.Cl.

H04N 19/96(2006.01)

H04N 19/597(2006.01)

G06T 17/00(2006.01)

G06T 17/10(2006.01)

G06T 9/40(2006.01)

H04N 19/20(2006.01)

G06T 9/00(2006.01)

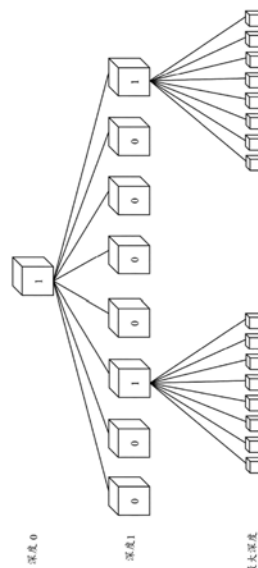
权利要求书2页 说明书13页 附图12页

(54)发明名称

用于对几何形状由基于八叉树的结构表示的有色点云的颜色进行编码/解码的方法和装置

(57)摘要

本原理涉及一种用于对点云的颜色进行编码的方法和装置,其中,该方法包括:-为表示点云的几何形状的基于八叉树的结构每个叶节点分配特定编号(LN<sub>k</sub>),特定编号(LN<sub>k</sub>)是通过根据先前分配给父节点的取向(Ω<sub>p</sub>)并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向({Ω<sub>0</sub>,...,Ω<sub>7</sub>})(201)和局部编号({n<sub>0</sub>,...,n<sub>7</sub>})(202)而分配给叶节点的;-通过根据所分配的特定编号扫描基于八叉树的结构叶节点来对点云的点进行排序(200);-根据特定2D路径将有色点云的有序点的颜色分配(210)给图像的像素;以及-对图像进行编码(220)。本原理还涉及一种用于对点云进行解码的方法和装置。



1. 一种用于对点云的颜色进行编码的方法,其中,所述方法包括:

-为表示所述点云的几何形状的基于八叉树的结构每个叶节点分配特定编号(LN<sub>k</sub>),特定编号(LN<sub>k</sub>)是通过根据先前分配给父节点的取向( $\Omega_p$ )并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向( $\{\Omega_0, \dots, \Omega_7\}$ ) (201)和局部编号( $\{n_0, \dots, n_7\}$ ) (202)而分配给叶节点的;

-通过根据所分配的特定编号扫描所述基于八叉树的结构叶节点来对所述点云的点进行排序(200);

-根据特定2D路径将有色点云的有序点的颜色分配(210)给图像的像素;以及

-对所述图像进行编码(220)。

2. 一种用于对点云的颜色进行编码的设备,其中,所述设备包括处理器,所述处理器被配置为:

-为表示所述点云的几何形状的基于八叉树的结构每个叶节点分配特定编号(LN<sub>k</sub>),特定编号(LN<sub>k</sub>)是通过根据先前分配给父节点的取向( $\Omega_p$ )并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向( $\{\Omega_0, \dots, \Omega_7\}$ ) (201)和局部编号( $\{n_0, \dots, n_7\}$ ) (202)而分配给叶节点的;

-通过根据所分配的特定编号扫描所述基于八叉树的结构叶节点来对所述点云的点进行排序;

-根据特定2D路径将有色点云的有序点的颜色分配给图像的像素;以及

-对所述图像进行编码。

3. 如权利要求1所述的方法或如权利要求2所述的设备,其中所述特定2D路径被编码为在比特流中发送的信息数据。

4. 如权利要求1、3中的一项所述的方法或如权利要求2-3中的一项所述的设备,其中,定义第一查找表,以根据先前分配给父节点的取向( $\Omega_p$ )和所述特定子节点编号来分配(201)所述父节点的八个子节点的取向。

5. 如权利要求4所述的方法或设备,其中,定义第二查找表,以根据父节点的取向和所述特定子节点编号顺序来分配(202)所述父节点的八个子节点的局部编号。

6. 一种用于对点云的颜色进行解码的方法,其中,所述方法包括:

-对图像进行解码(1400);

-为表示所述点云的几何形状的基于八叉树的结构每个叶节点分配特定编号(LN<sub>k</sub>),特定编号(LN<sub>k</sub>)是通过根据先前分配给父节点的取向( $\Omega_p$ )并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向( $\{\Omega_0, \dots, \Omega_7\}$ ) (201)和局部编号( $\{n_0, \dots, n_7\}$ ) (202)而分配给叶节点的;

-通过根据所分配的特定编号扫描所述基于八叉树的结构叶节点来对所述基于八叉树的结构叶节点进行排序(1410);

-通过根据特定2D路径和与所述基于八叉树的结构有序叶节点相关联的有序叶立方体扫描所解码的图像(I'),来获得(1420)有色点云(IIPC)的点的颜色。

7. 如权利要求6所述的方法,其中,所述方法还包括:根据针对所述基于八叉树的结构每个节点从比特流中解码的节点信息数据、以及针对所述基于八叉树的结构每个叶节点从比特流中解码的叶信息数据,获得所述基于八叉树的结构,节点信息数据指示与所述

节点相关联的立方体是否被分割,并且叶信息数据指示与所述基于八叉树的结构 of 的所述叶节点相关联的叶立方体是否包含所述有色点云的点。

8. 一种用于对点云的颜色进行解码的设备,其中,所述设备包括处理器,所述处理器被配置为:

-对图像进行解码(1400);

-为表示所述点云的几何形状的基于八叉树的结构 of 的每个叶节点分配特定编号( $LN_k$ ),特定编号( $LN_k$ )是通过根据先前分配给父节点的取向( $\Omega_p$ )并根据特定子节点编号顺序给所述基于八叉树的结构 of 的八个子节点中的每一个递归地分配取向( $\{\Omega_0, \dots, \Omega_7\}$ )(201)和局部编号( $\{n_0, \dots, n_7\}$ )(202)而分配给叶节点的;

-通过根据所分配的特定编号扫描所述基于八叉树的结构 of 的叶节点来对所述基于八叉树的结构 of 的叶节点进行排序(1410);

-通过根据特定2D路径和与所述基于八叉树的结构 of 的有序叶节点相关联的有序叶立方体扫描所解码的图像( $I'$ ),来获得(1420)有色点云(IIPC)的点的颜色。

9. 如权利要求8所述的设备,其中,所述处理器还被配置为:根据针对所述基于八叉树的结构 of 的每个节点从比特流中解码的节点信息数据、以及针对所述基于八叉树的结构 of 的每个叶节点从比特流中解码的叶信息数据,获得所述基于八叉树的结构 of 的节点信息数据指示与所述节点相关联的立方体是否被分割,并且叶信息数据指示与所述基于八叉树的结构 of 的所述叶节点相关联的叶立方体是否包含所述有色点云的点。

10. 如权利要求6-7中的一项所述的方法或如权利要求8-9中的一项所述的设备,其中,所述特定2D路径是从自比特流获得的信息数据中解码的。

11. 一种计算机程序产品,包括程序代码指令,以当该程序在计算机上被执行时执行根据权利要求1或6所述的方法的步骤。

12. 一种包括指令的计算机可读介质,所述指令在由计算机执行时使所述计算机执行根据权利要求1或6所述的方法。

## 用于对几何形状由基于八叉树的结构表示的有色点云的颜色进行编码/解码的方法和装置

### 技术领域

[0001] 本原理一般地涉及对表示3D对象的几何形状和颜色的有色点云 (colored point cloud) 进行编码和解码。具体地但非排他地,本原理的技术领域涉及对有色点云的颜色进行编码/解码。

### 背景技术

[0002] 本部分旨在向读者介绍技术的各个方面,这些方面可能与下面描述和/或要求保护的本原理的各个方面有关。相信该讨论有助于向读者提供背景信息,以便更好地理解本原理的各个方面。因此,应该理解的是,这些陈述要在该角度上来阅读,而不是作为对现有技术的承认。

[0003] 点云是通常旨在表示3D对象的外表面以及更复杂的几何形状(如头发、毛发)的点的集合,这些更复杂的几何形状可能无法通过其他数据格式(如网格)有效地表示。点云的每个点往往通过3D空间位置(3D空间中的X、Y和Z坐标)来定义,也可能通过其他相关属性来定义,诸如例如表示在RGB或YUV颜色空间中的颜色、透明度、反射率、两分量法向矢量等。

[0004] 在下文中,考虑有色点云,即6分量点(X,Y,Z,R,G,B)或等效地(X,Y,Z,Y,U,V)的集合,其中(X,Y,Z)定义点在3D空间中的空间位置,而(R,G,B)或(Y,U,V)定义该点的颜色。

[0005] 有色点云可以是静态的或者动态的,取决于云是否相对于时间而演变。应当注意的是,在动态点云的情况下,点的数量不是恒定的,而相反一般是随时间而演变的。因此,动态点云是点的集合的时间顺序列表。

[0006] 实际上,有色点云可以用于各种目的,诸如文化遗产/建筑物,其中以3D形式扫描如雕像或建筑物的对象,以便在不传送或访问对象的情况下共享对象的空间配置。此外,它是一种确保在对象可能被破坏的情况下保存对象的知识的方式;例如地震后的寺庙。这种有色点云通常是静态的且巨大的。

[0007] 另一个用例是在地形学和制图学中,其中通过使用3D表示,地图不限于平面,而是可以包含地势。

[0008] 汽车工业和自动汽车也是可以使用点云的领域。自动汽车应当能够“探测”其环境,从而基于其近邻的实际情况做出安全驾驶决策。典型的传感器产生由决策引擎使用的动态点云。这些点云并不意图供人类观看。它们通常是小的、未必有颜色、并且是动态的且具有高捕获频率。它们可能具有其他属性,如反射率,它是与感测对象的物理表面的材料相关的有价值的信息并且可以帮助做出决策。

[0009] 虚拟现实(VR)和沉浸式世界近来已成为热门话题,并被许多人预见为2D平面视频的未来。基本思想是让观看者沉浸在其周围的环境中,与标准TV相反,在标准TV中,他只能看到面前的虚拟世界。取决于观看者在环境中的自由度,沉浸感有若干等级。有色点云是分发VR世界的良好格式候选。它们可以是静态的或动态的,并且通常具有平均大小,例如一次不超过几百万个点。

[0010] 只有当比特流的大小足够小到允许向终端用户进行实际存储/发送时,点云压缩才能成功存储/发送用于沉浸式世界的3D对象。

[0011] 同样重要的是,能够以合理的带宽消耗向终端用户分发动态点云,同时保持可接受的(或优选地是非常好的)体验质量。与视频压缩类似,良好地使用时间相关性被认为是会导致动态点云的有效压缩的重要元素。

[0012] 公知的方法(例如,在2008年9月计算机图形论坛中Tilo Ochotta和Dietmar Saupe的“基于图像的表面压缩(Image-Based Surface Compression)”中)将表示3D对象的几何形状和颜色的有色点云投影到位于3D对象周围的表面上,以获得有关纹理和深度的视频并使用诸如3D-HEVC(HEVC的扩展,其规范可在下面找到:ITU网站,T推荐,H系列,h265,<http://www.itu.int/rec/T-REC-H.265-201612-1/en> annex G and I)的传统编码器对纹理和深度视频进行编码。

[0013] 对于每个投影点,压缩的性能都接近于视频压缩,但是当考虑动态点云时,由于遮挡、冗余和时间稳定性,某些内容可能会更复杂。因此,就比特率而言,点云压缩比视频压缩的要求更多。

[0014] 关于遮挡,几乎不可能在不使用许多投影的情况下得到复杂拓扑的完整几何形状。因此,用于对所有这些投影进行编码/解码的所需资源(计算能力、存储内存)通常太高。

[0015] 关于冗余,如果在两个不同的投影上两次看到一个点,则其编码效率被除以二,并且如果使用大量的投影,这可能容易变得更糟。可以在投影之前使用非重叠的面片(patch),但这使得投影的分区边界不平滑,因此难以编码,并且这对编码性能产生负面影响。

[0016] 关于时间稳定性,可以在给定时间针对对象来优化投影之前的非重叠面片,但是,当该对象移动时,面片边界也移动,并且丢失难以编码的区域(=该边界)的时间稳定性。实际上,由于时间帧间预测在该背景下效率低下,因此获得的压缩性能并不比全帧内编码好得多。

[0017] 基于八叉树的编码是用于对有色点云的几何形状进行编码的另一个公知方法。

[0018] 通过递归地分割涵盖有色点云的立方体,直到与基于八叉树的结构 of 的叶节点相关联的叶立方体(leaf cube)包含不超过点云的一个点,来获得用于表示点云的几何形状的所述基于八叉树的结构。因此,基于八叉树的结构 of 的叶节点的空间位置表示点云的点的空间位置,即其几何形状。

[0019] 基于八叉树的编码是用于对有色点云的点的几何形状进行编码的有效方法,但是不提供对有色点云的点的颜色的编码。

[0020] 因此,需要找到一种用于对几何形状由基于八叉树的结构表示的点云的颜色进行编码的解决方案。

## 发明内容

[0021] 以下呈现本原理的简化的概述,以便提供对本原理的某些方面的基本理解。该概述并非对本原理的广泛概论。它并不旨在识别本原理的关键或重要元素。以下概述仅以简化形式来呈现本原理的一些方面,作为下面提供的更详细描述 of 的序言。

[0022] 一般来说,本原理涉及对有色点云的颜色进行编码,该有色点云的几何形状由基

于八叉树的结构表示。所述方法在编码方面是有效的,因为有色点云的颜色被映射到可以由传统编码器编码的图像上。

[0023] 本原理涉及一种用于对点云的颜色进行编码的方法和设备,其中,该方法包括:

[0024] -为表示点云的几何形状的基于八叉树的结构每个叶节点分配特定编号,特定编号是通过根据先前分配给父节点的取向(orientation)并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向和局部编号而分配给叶节点的;

[0025] -通过根据所分配的特定编号扫描基于八叉树的结构叶节点来对点云的点进行排序;

[0026] -根据特定2D路径将有色点云的有序点的颜色分配给图像的像素;以及

[0027] -对图像进行编码。

[0028] 根据实施例,特定2D路径被编码为在比特流中发送的信息数据。

[0029] 根据实施例,定义第一查找表,以根据先前分配给父节点的取向和特定子节点编号来分配父节点的八个子节点的取向。

[0030] 根据实施例,定义第二查找表,以根据父节点的取向和特定子节点编号顺序来分配父节点的八个子节点的局部编号。

[0031] 根据它们的另一方面,本原理涉及一种用于对点云的颜色进行解码的方法和设备。该方法包括:

[0032] -对图像进行解码;

[0033] -为表示点云的几何形状的基于八叉树的结构每个叶节点分配特定编号,特定编号是通过根据先前分配给父节点的取向并根据特定子节点编号顺序给所述基于八叉树的结构八个子节点中的每一个递归地分配取向和局部编号而分配给叶节点的;

[0034] -通过根据所分配的特定编号扫描基于八叉树的结构叶节点来对基于八叉树的结构叶节点进行排序;

[0035] -通过根据特定2D路径和与基于八叉树的结构有序叶节点相关联的有序叶立方体扫描所解码的图像,来获得有色点云的点的颜色。

[0036] 根据实施例,该方法还包括:根据针对基于八叉树的结构每个节点从比特流中解码的节点信息数据、以及针对基于八叉树的结构每个叶节点从比特流中解码的叶信息数据,获得基于八叉树的结构,节点信息数据指示与节点相关联的立方体是否被分割,并且叶信息数据指示与基于八叉树的结构所述叶节点相关联的叶立方体是否包含有色点云的点。

[0037] 根据实施例,特定2D路径是从自比特流获得的信息数据中解码的。

[0038] 根据它们的另一方面,本原理涉及一种计算机程序产品,包括程序代码指令,以当该程序在计算机上被执行时执行以上方法的步骤。

[0039] 根据它们的另一方面,本原理涉及一种包括指令的计算机可读介质,该指令在由计算机执行时使计算机执行以上方法之一。

[0040] 通过以下结合附图对示例的描述,本原理的具体性质以及本原理的其他目的、优点、特征和用途将变得明显。

## 附图说明

- [0041] 在附图中,图示了本原理的示例。其示出了:
- [0042] 图1图示了基于八叉树的结构示例;
- [0043] 图2示意性地示出了根据本原理的示例的用于对表示3D对象的几何形状和颜色的有色点云进行编码的方法的步骤的图;
- [0044] 图3图示了图2的方法;
- [0045] 图4示出了3D希尔伯特(Hilbert)路径的示例;
- [0046] 图5示出了定向系统的图示;
- [0047] 图6示出了当通过图5的定向系统来表征3D希尔伯特路径时的取向的示例;
- [0048] 图7示出了根据本原理的实施例的步骤200的子步骤的图;
- [0049] 图8示出了特定节点编号顺序的示例;
- [0050] 图9示出了遵循特定节点编号顺序分配给子立方体的取向和局部编号的示例;
- [0051] 图10示出了取向的查找表的示例;
- [0052] 图11示出了局部编号的查找表的示例;
- [0053] 图12示出了连续的2D Z-扫描曲线的图示;
- [0054] 图13示出了连续的2D Peano曲线的图示;
- [0055] 图14示出了基于块的2D栅格(raster)扫描路径的图示;
- [0056] 图15示出了基于块的2D希尔伯特扫描路径的图示;
- [0057] 图16示意性地示出了根据本原理的示例的用于从至少一个比特流中解码有色点云的颜色的方法的步骤的图,该有色点云的几何形状由基于八叉树的结构表示。
- [0058] 图17图示了图16的方法。
- [0059] 图18示出了根据本原理的示例的设备的架构的示例;
- [0060] 图19示出了根据本原理的示例的通过通信网络进行通信的两个远程设备;以及
- [0061] 图20示出了根据本原理的示例的信号的语法。
- [0062] 类似的或相同的元件用相同的参考标号来引用。

## 具体实施方式

[0063] 在下文中将参考附图更全面地描述本原理,在附图中示出了本原理的示例。然而,本原理可以以许多替代形式来体现,并且不应被解释为限于在此阐述的示例。因此,尽管本原理易于有各种修改和替代形式,但是其具体示例通过附图中的示例的方式示出,并且将在这里进行详细描述。然而,应当理解的是,无意将本原理限制为所公开的特定形式,而是相反,本公开将涵盖落入如由权利要求所限定的本原理的精神和范围内的所有修改、等同物和替代物。

[0064] 在此使用的术语仅出于描述特定示例的目的,而无意于限制本原理。如在此所使用的,单数形式“一(a)”,“一(an)”和“该”也意图包含复数形式,除非上下文另外明确指出。还将理解的是,当在本说明书中使用时,术语“包括(comprise)”、“包括(comprising)”、“包含(include)”和/或“包含(including)”指定存在所陈述的特征、整数、步骤、操作、元件和/或组件,但不排除一个或多个其他特征、整数、步骤、操作、元件、组件和/或其群组的存在或附加。此外,当元件被称为“响应”或“连接”于另一元件时,它可以直接响应或连接于另一元

件,或者可以存在中间元件。相反,当元件被称为“直接响应”或“直接连接”于其他元件时,不存在中间元件。如在此所使用的,术语“和/或”包含相关联的所列项中的一个或多个的任何和所有组合,并且可以缩写为“/”。

[0065] 将理解的是,尽管可以在此使用术语第一、第二等来描述各种元件,但是这些元件不应受到这些术语的限制。这些术语仅用于将一个元件与另一个元件区分开。例如,在不脱离本原理的教导的情况下,第一元件可以被称作第二元件,并且类似地,第二元件可以被称作第一元件。

[0066] 尽管一些图包含通信路径上的箭头以示出通信的主要方向,但是要理解,通信可以发生在与所描绘的箭头相反的方向上。

[0067] 关于框图 and 操作流程图描述一些示例,其中每个框代表电路元件、模块或包括用于实现(多个)指定的逻辑功能的一个或多个可执行指令的代码的一部分。还应当注意,在其他实现方式中,框中注释的(多个)功能可以不按所注释的顺序发生。例如,取决于所涉及的功能,连续示出的两个框实际上可以基本上同时执行,或者这些框有时可以以相反的顺序执行。

[0068] 在此对“根据示例”或“在示例中”的引用是指结合示例描述的特定特征、结构或特性可以包含在本原理的至少一个实现方式中。在说明书中各处的短语“根据示例”或“在示例中”的出现不一定都指代相同的示例,单独的或替代的示例也不一定与其他示例相互排斥。

[0069] 权利要求中出现的参考标号仅作为说明,并且应对权利要求的范围没有限制作用。

[0070] 尽管没有明确描述,但是可以以任何组合或子组合来采用本示例和变型。

[0071] 本原理针对编码/解码有色点云进行描述,但是扩展到有色点云序列的编码/解码,因为序列的每个有色点云被顺序地编码/解码,如下所述。

[0072] 在下文中,图像包含呈特定图像/视频格式的一个或几个样本阵列(像素值),其指定与图像(或视频)的像素值有关的所有信息,以及可以由显示器和/或任何其他设备使用以例如对图像(或视频)进行可视化和/或解码的所有信息。图像包括呈第一样本阵列的形状的至少一个分量,通常是照度(luma)(或亮度(luminance))分量,并且可能包括呈至少一个其他样本阵列的形状的至少一个其他分量,通常是颜色分量。或者,等同地,相同的信息也可以由颜色样本阵列的集合来表示,诸如传统的三色RGB表示。

[0073] 像素值由 $n_v$ 个值的矢量表示,其中 $n_v$ 是分量的数量。矢量的每个值利用定义像素值的最大动态范围的多个比特来表示。

[0074] 基于八叉树的编码是用于对有色点云IPC的几何形状、即有色点云IPC的点的空间位置进行编码的公知方法。

[0075] 基于八叉树的结构0是这样的结构,其中每个父节点最多可以包括八个子节点,并且其中立方体与该节点中的每个相关联。根节点(深度0)是没有任何父节点并且每个子节点(深度大于0)具有单个父节点的唯一节点。叶节点是没有子节点的节点。叶立方体是与基于八叉树的结构0的叶节点相关联的立方体。

[0076] 可以通过递归地分割涵盖有色点云IPC的立方体,直到与基于八叉树的结构0的叶节点相关联的叶立方体包含不超过有色点云IPC的一个点,来获得用于表示有色点云IPC的

几何形状的所述基于八叉树的结构0。位于叶立方体中的点表示包含在所述叶立方体中的有色点云IPC的点。因此,包含点(每个叶立方体一个点)的基于八叉树的结构0的叶立方体表示有色点云IPC的那些点的空间位置。

[0077] 在图1上所示的示例中,将与根节点(深度0)相关联的立方体分割为8个子立方体(深度1),然后将深度为1的两个子立方体分割为8个子立方体(最大深度=2)。

[0078] 相同深度的立方体的大小通常是相同的,但是本原理不限于该示例。当分割立方体时,特定过程还可以确定每个深度的不同数量的子立方体,和/或相同深度或根据其深度的立方体的多个尺寸。

[0079] 图2示意性地示出了根据本原理的示例的用于对有色点云的颜色进行编码的方法的步骤的图,该有色点云的几何形状由基于八叉树的结构0表示。

[0080] 在步骤200中,模块M1通过根据特定3D路径扫描基于八叉树的结构0的叶节点来对有色点云IPC的点进行排序。

[0081] 例如,如图3中所示,将编号分配给基于八叉树的结构0的每个叶节点,并根据所分配的编号对包含有色点云的点的叶立方体进行排序。在图3中,仅示出了与包含有色点云IPC的点的叶立方体相关联的叶节点的编号(25,2,...)。

[0082] 有序叶立方、即有色点云IPC的有序点可以形成如图3中所示的有序编号的列表。每个编号指代包含有色点云IPC的点的特定叶立方体。

[0083] 根据步骤200的实施例,针对基于八叉树的结构0的每个节点,将节点信息数据编码在比特流F1中,指示与节点相关联的立方体是否被分割。

[0084] 根据步骤200的实施例,节点信息数据是二进制标志,该二进制标志等于1以指示与节点相关联的立方体被分割,否则为0。

[0085] 根据可选变型,立方体分割的最大深度也被编码在比特流F1中。

[0086] 这避免针对具有最大深度的所有立方体发信号通知节点信息数据。

[0087] 根据步骤200的实施例,针对基于八叉树的结构0的每个节点,将叶信息数据编码在比特流F1中,指示与叶节点相关联的叶立方体是否包含有色点云的点。

[0088] 根据步骤200的实施例,叶信息数据是二进制标志,该二进制标志等于1以指示叶立方体被分割,否则为0。

[0089] 在步骤210中,模块M2根据特定2D路径将有色点云IPC的每个有序点的颜色分配给图像I的像素。

[0090] 如图3中所示,2D路径可以是通常在图像处理中使用的栅格顺序。遵循这样的2D路径,将有序点列表的第一点(25)的颜色分配给图像I的左上像素,将有序点列表的第二点(2)的颜色分配给同一行上的相邻像素,等等。

[0091] 在步骤220中,编码器ENC1对图像I进行编码。

[0092] 根据实施例,特定3D和/或特定2D路径被编码为信息数据,诸如比特的集合。所述信息数据可以表示列出若干候选3D和/或2D路径的表的索引。

[0093] 根据实施例,所述信息数据被存储和/或在比特流F1中发送。

[0094] 编码的图像I可以被存储和/或在比特流F2中发送。

[0095] 这些比特流F1和F2可以由单个或多个信号承载。

[0096] 根据步骤220的实施例,编码器ENC1与传统编码器兼容,传统编码器诸如例如:

[0097] • JPEG, 规范ISO/CEI 10918-1UIT-T推荐T.81, <https://www.itu.int/rec/T-REC-T.81/en>,

[0098] • AVC, 又名MPEG-4AVC或h264。在UIT-T H.264和ISO/CEI MPEG-4部分10 (ISO/CEI 14496-10) 二者中规定, <http://www.itu.int/rec/T-REC-H.264/en>,

[0099] • 或者HEVC (其规范可在下面找到: ITU网站, T推荐, H系列, h265, <http://www.itu.int/rec/T-REC-H.265-201612-1/en>)。

[0100] 图像I的行数和列数是预先定义的。

[0101] 有利地, 它可以是图像或视频编码器ENC1的编码单元(或块)的大小的倍数。例如, 当使用HEVC时, 行数和列数二者优选地是64的倍数。图像I的像素数至少应为点云IPC的点数, 以确保所有点的颜色都可以被映射到图像I上。例如, 一百万点的有色点云IPC可以被映射到 $1024 \times 1024$ 像素的图像I上。

[0102] 根据步骤200的实施例, 模块M1给基于八叉树的结构0的每个节点分配特定编号 $LN_k$ , 以唯一地识别包含在与叶节点相关联的叶立方体中的每个点。

[0103] 例如, 分配给子节点的特定编号 $LN_k$ 由下式给出:

$$[0104] \quad LN_k = 8 * PN_k + n_k \quad (1)$$

[0105] 其中 $PN_k$ 是分配给子节点的父节点的特定编号, 并且 $n_k$ 是分配给子节点的局部编号。

[0106] 根据步骤200的实施例, 特定3D路径是被定义为如图4上所示的3D希尔伯特曲线的离散近似的3D希尔伯特路径。

[0107] 在图4上, 基于八叉树的结构0的父节点的八个子节点被表示为编号为0到7的八个顶点, 这些顶点通过线段象征性地连接。换句话说, 与父节点相关联的立方体被分割, 并且八个子立方体与顶点相关联。因此, 对于基于八叉树的结构0的给定深度级别遵循3D希尔伯特路径是通过首先考虑顶点0、然后考虑顶点1...、最后考虑顶点7来定义的。

[0108] 因此, 3D希尔伯特路径允许根据特定顺序扫描基于八叉树的结构0的父节点的子节点。

[0109] 图4中所示的3D希尔伯特路径和特定顺序应在基于八叉树的结构0的任何深度级别上进行再现。

[0110] 根据步骤200的实施例, 给基于八叉树的结构0的每个节点分配特定编号 $LN_k$ 使用定向系统, 该定向系统的表征使用沿着连接八个顶点的线段移动的“飞机”的概念, 如图5中所示。

[0111] “飞机”可以沿着三个轴(前/后、上/下、右/左)移动并且可以绕每个轴旋转0度、90度、180度或270度的角度。

[0112] 因此, 如图6中所示, 定义了“飞机”的24个取向的集合 $\Omega$ , 每个取向通过方向(前/后、上/下、右/左)和旋转角度(0、90、180或270度)来定义, 并通过编号0-23标记。

[0113] 例如, 取向 $\Omega = 16$ 对应于方向“后”结合0度的旋转角度, 方向 $\Omega = 10$ 对应于方向“上”结合270度的旋转角度。

[0114] 根据步骤200的实施例, 如图7上所示, 模块M1根据先前分配给父节点的取向 $\Omega_p$ 并根据特定子节点编号顺序给八个子节点中的每一个分配取向 $\{\Omega_0, \dots, \Omega_7\}$  (步骤201) 和局部编号 $\{n_0, \dots, n_7\}$  (步骤202)。该特定子节点编号顺序在所有节点上定义一次, 并且与取向

$\Omega_p$ 无关。

[0115] 步骤201和202是递归的,以根据3D希尔伯特路径给基于八叉树的结构0的每个节点分配取向( $\Omega_m$ )和局部编号( $n_m$ )。

[0116] 然后,将特定编号 $LN_k$ 分配给基于八叉树的结构0的每个节点(等式1),然后根据这些特定编号 $LN_k$ 对包含有色点云IPC的点的叶立方体(与基于八叉树的结构0的叶节点相关联)进行排序。

[0117] 例如,当使用图8上所示的特定节点编号顺序时,如图9中所示,将( $\Omega_m, n_m$ )的对分配给叶立方体的每个顶点。

[0118] 本原理不限于任何特定节点编号顺序,而是扩展到任何节点编号顺序。

[0119] 根据步骤201的实施例,如图10中所示,定义第一查找表,以根据父节点的取向和特定子节点编号来分配父节点的八个子节点的取向。父节点的取向定义查找表的行,并且查找表的列定义子节点的取向的不同值。

[0120] 例如,当父节点的取向等于16时,子节点的取向为(18,5,10,5,18,3,14,3),如图10中所示。

[0121] 根据步骤202的实施例,如图11中所示,定义第二查找表,以根据父节点的取向和特定子节点编号顺序来分配父节点的八个子节点的局部编号。父节点的取向定义查找表的行,而查找表的列定义子节点的局部编号的不同值。

[0122] 这两个后面的实施例提供了该方法的非常快速的实现方式,因为给基于八叉树的结构0的节点分配取向和局部编号仅需要对查找表的多次访问。

[0123] 例如,当通过图8给出特定编号顺序并且父节点的取向等于16时,子节点的局部编号为(3,2,0,1,4,5,7,6),如图11中所示。

[0124] 然后,根据等式1,例如,如果分配给父节点的特定编号 $PN_k$ 等于1,则分配给该父节点的子节点0的编号 $LN_0$ 等于11( $8*1+3$ ),分配给该父节点的子节点1的编号 $LN_1$ 等于10( $8*1+2$ ),分配给该父节点的子节点2的编号 $LN_2$ 等于8( $8*1+0$ ),分配给该父节点的子节点3的编号 $LN_3$ 等于9( $8*1+1$ ),等等。

[0125] 然后,如果子节点0不是叶节点,则与(特定编号,取向)=(11,18)的对相关联的该节点被视为父节点,并且根据取向18将取向和局部编号分配给其子节点。其八个子节点的取向是(1,12,1,16,7,8,7,16),并且遵循等式1,局部编号(1,0,2,3,6,7,5,4)导致这八个子节点的特定编号为(89,88,90,91,94,95,93,92)。这些子节点(如果不是叶节点)可以变为父节点,并且递归过程继续。

[0126] 接下来,如果子节点1不是叶节点,则与(10,5)的对相关联的该节点被视为父节点,并且根据取向5将取向和局部编号分配给其子节点,等等。

[0127] 如果子节点是叶节点,则为其分配的编号是特定编号 $LN_k$ 。

[0128] 本原理不限于3D希尔伯特路径,并且可以遵循或不遵循与上面针对3D希尔伯特路径详述的节点编号类似的节点编号顺序来使用其他特定3D路径。

[0129] 图12和图13示出了特定2D路径的其他示例,诸如Z-扫描曲线(图12)或Peano曲线(图13)。读者将理解的是,这些特定2D路径也可以扩展到3D情况。

[0130] 遵循步骤200,根据特定编号 $LN_k$ 对包含在基于八叉树的结构0的叶节点的叶立方体中的有色点云IPC的点进行排序。

[0131] 根据步骤210的实施例,如图14中所示,根据栅格扫描将有色点云IPC的每个有序点的颜色分配给图像的像素。

[0132] 根据步骤210的实施例,根据如图15中所示的连续2D希尔伯特曲线的离散近似,将有色点云IPC的每个有序点的颜色分配给图像的像素。

[0133] 连续的希尔伯特曲线及其离散近似都是有用的,因为它们给出了相当好地保留局部性的3D到1D(步骤200)或1D到2D(步骤210)空间映射。基本上这意味着,如果 $(x, y)$ 是单位正方形内的2D点的坐标,并且 $d$ 是当其到达该2D点时沿2D曲线的距离,则具有邻近的 $d$ 值的2D点也将具有邻近的 $(x, y)$ 值。反过来则无法总是真的。有时会存在 $(x, y)$ 坐标接近但其 $d$ 值相距甚远的点。这在具有单位立方体中的坐标点 $(x, y, z)$ 的3D空间中也是有效的。保留局部性是有利的,因为当对点执行从3D几何图形到图像I的扫描时,其允许保留颜色(纹理)空间相关性。2D图像/视频编解码器的压缩能力主要来自纹理空间相关性。因此,不保留这种相关性将导致图像I的压缩非常差。

[0134] 图16示意性地示出了根据本原理的示例的用于从至少一个比特流中解码重构点云RPC的颜色的方法的步骤的图,该重构点云RPC的几何形状由基于八叉树的结构O表示。

[0135] 如上所述,可以通过递归地分割涵盖有色点云IPC的立方体,直到与基于八叉树的结构O的叶节点相关联的叶立方体包含不超过有色点云IPC的一个点,来获得用于表示有色点云IPC的几何形状的所述基于八叉树的结构O。位于叶立方体中的点表示包含在所述叶立方体中的有色点云IPC的点。因此,包含点(每个叶立方体一个点)的基于八叉树的结构O的叶立方体表示有色点云IPC的那些点的空间位置。

[0136] 根据实施例,根据针对基于八叉树的结构O的每个节点从比特流F1中解码的节点信息数据、以及针对基于八叉树的结构O的每个叶节点从比特流F1中解码的叶信息数据,获得基于八叉树的结构O。

[0137] 节点信息数据指示与节点相关联的立方体是否被分割,并且叶信息数据指示与基于八叉树的结构O的所述叶节点相关联的叶立方体是否包含有色点云的点。

[0138] 根据可选变型,还从比特流F1中解码立方体分割的最大深度。

[0139] 在步骤1400中,解码器DEC1从比特流F2中对编码图像I进行解码,以获得解码图像I'。

[0140] 根据步骤1400的实施例,解码器DEC1与诸如例如JPEG、AVC或HEVC的传统解码器兼容。

[0141] 在步骤1410中,模块M3通过根据特定3D路径扫描基于八叉树的结构O的叶节点来对基于八叉树的结构O的叶节点进行排序。

[0142] 步骤1410的排序类似于步骤200的排序,并且可以基于分配给基于八叉树的结构O的每个叶节点的特定编号 $LN_k$ (图2)。

[0143] 然后,获得包含有色点云IPC的点的有序叶立方体的列表。

[0144] 在步骤1420中,模块M4通过根据特定2D路径和与基于八叉树的结构O的叶节点相关联的有序叶立方体扫描解码图像I',来获得重构点云RPC的点的颜色。

[0145] 例如,如图17中所示,当特定2D路径是栅格顺序时,解码图像I'的左上像素的颜色是有序叶立方体的列表的第一叶立方体(点)(25)的颜色,相邻像素的颜色是有序叶立方体的列表的第二点(2)的颜色,等等。

[0146] 根据实施例,特定3D路径和/或特定2D路径是从诸如比特集合的信息数据中解码的。所述信息数据可以表示列出若干候选3D和/或2D路径的表的索引。

[0147] 根据实施例,从存储内存和/或从比特流F1获得所述信息数据。

[0148] 根据步骤1420的实施例,如图14中所示,根据栅格扫描从解码图像I'的像素获得重构点云PRC的每个有序点的颜色。

[0149] 根据步骤210的实施例,根据如图15中所示的连续2D希尔伯特曲线的离散近似,从解码图像I'的像素获得重构点云RPC的每个有序点的颜色。

[0150] 在图1-图17中,模块是功能单元,其可以与可区分的物理单元相关或不相关。例如,这些模块或它们中的一些可以在唯一的组件或电路中集成在一起,或者有助于软件的功能。相反,某些模块可以可能地由单独的物理实体组成。使用纯硬件(例如使用诸如ASIC或FPGA或VLSI(分别是“专用集成电路”、“现场可编程门阵列”、“超大规模集成”)的专用硬件)、或者来自嵌入设备中的若干集成电子组件、或者来自硬件和软件组件的混合来实现与本原理兼容的装置。

[0151] 图18表示设备1800的示例性架构,设备1800可以被配置为实现关于图1-图17描述的方法。

[0152] 设备1800包括通过数据和地址总线1801链接在一起的以下元件:

[0153] ——微处理器1802(或CPU),其为例如DSP(或数字信号处理器);

[0154] ——ROM(或只读存储器)1803;

[0155] ——RAM(或随机存取存储器)1804;

[0156] ——I/O接口1805,用于从应用接收要发送的数据;和

[0157] ——电池1806。

[0158] 根据示例,电池1806在设备外部。在提到的存储器的每一个中,说明书中使用的词语“寄存器”可以对应于小容量的区域(一些比特)或非常大的区域(例如整个程序或大量接收或解码的数据)。ROM 1803至少包括程序和参数。ROM 1803可以存储算法和指令以执行根据本原理的技术。当接通时,CPU 1802将RAM中的程序上载并执行对应的指令。

[0159] RAM 1804在寄存器中包括由CPU 1802执行并在设备1800接通之后上载的程序、寄存器中的输入数据、寄存器中的处于方法的不同状态的中间数据以及寄存器中的用于执行方法的其他变量。

[0160] 在此描述的实现方式可以例如以方法或过程、装置、软件程序、数据流或信号来实现。即使仅在单一形式的实现方式的上下文中进行讨论(例如,仅作为方法或设备进行讨论),但是所讨论的特征的实现方式也可以以其他形式(例如程序)来实现。装置可以例如以适当的硬件、软件和固件来实现。方法可以在诸如例如处理器之类的装置中实现,处理器通常是指处理设备,包含例如计算机、微处理器、集成电路或可编程逻辑器件。处理器还包含通信设备,诸如例如计算机、蜂窝电话、便携式/个人数字助理(“PDA”),以及有助于终端用户之间的信息通信的其他设备。

[0161] 根据编码或编码器的示例,从源获得原始的有色点云IPC。例如,源属于包括以下的集合:

[0162] ——本地存储器(1803或1804),例如视频存储器或RAM(或随机存取存储器)、闪存、ROM(或只读存储器)、硬盘;

[0163] ——存储接口 (1805), 例如与大容量存储器、RAM、闪存、ROM、光盘或磁性支撑件的接口;

[0164] ——通信接口 (1805), 例如有线接口 (例如总线接口、广域网接口、局域网接口) 或无线接口 (诸如IEEE 802.11接口或蓝牙®接口); 和

[0165] ——图像捕获电路 (例如传感器, 诸如例如CCD (或电荷耦合器件) 或CMOS (或互补金属氧化物半导体))。

[0166] 根据解码或解码器的示例, 将重构的有色点云CPC传送到目的地; 具体地, 目的地属于包括以下的集合:

[0167] ——本地存储器 (1803或1804), 例如视频存储器或RAM、闪存, 硬盘;

[0168] ——存储接口 (1805), 例如与大容量存储器、RAM、闪存、ROM、光盘或磁性支撑件的接口;

[0169] ——通信接口 (1805), 例如有线接口 (例如总线接口 (例如USB (或通用串行总线))、广域网接口、局域网接口、HDMI (高清多媒体接口) 接口或无线接口 (诸如IEEE 802.11接口、WiFi®或蓝牙®接口);

[0170] ——呈现设备; 以及

[0171] ——显示器。

[0172] 根据编码或编码器的示例, 比特流F1-F2被传送到目的地。作为示例, 比特流F1-F2中的至少一个被存储在本地或远程存储器中, 例如, 视频存储器 (1804) 或RAM (1804)、硬盘 (1803)。在变型中, 比特流F1-F2中的至少一个被传送到存储接口 (1805) (例如, 与大容量存储器、闪存、ROM、光盘或磁性支撑件的接口), 和/或通过通信接口 (1805) (例如与点对点链路、通信总线, 点对多点链路或广播网络的接口) 发送。

[0173] 根据解码或解码器的示例, 从源获得比特流F1-F2中的至少一个。示例性地, 从本地存储器 (例如视频存储器 (1804)、RAM (1804)、ROM (1803)、闪存 (1803) 或硬盘 (1803)) 读取比特流。在变型中, 从存储接口 (1805) (例如与大容量存储器、RAM、ROM、闪存、光盘或磁性支撑件的接口) 接收比特流, 和/或从通信接口 (1805) (例如与点对点链路、总线、点对多点链路或广播网络的接口) 接收比特流。

[0174] 根据示例, 被配置为实现关于图1-图15描述的编码方法的设备1800属于包括以下的集合:

[0175] ——移动设备;

[0176] ——具有3D捕获能力的智能电话或电视机;

[0177] ——通信设备;

[0178] ——游戏设备;

[0179] ——平板 (或平板计算机);

[0180] ——膝上型计算机;

[0181] ——静止图像相机;

[0182] ——摄像机;

[0183] ——编码芯片;

[0184] ——静止图像服务器; 以及

- [0185] ——视频服务器(例如,广播服务器、视频点播服务器或网络服务器)。
- [0186] 根据示例,被配置为实现关于图16-图17描述的解码方法的设备1800属于包括以下的集合:
- [0187] ——移动设备;
  - [0188] ——头戴式显示器(HMD);
  - [0189] ——(混合现实)智能眼镜;
  - [0190] ——全息设备;
  - [0191] ——通信设备;
  - [0192] ——游戏设备;
  - [0193] ——机顶盒;
  - [0194] ——电视机;
  - [0195] ——平板(或平板计算机);
  - [0196] ——膝上型计算机;
  - [0197] ——显示器;
  - [0198] ——立体显示器;以及
  - [0199] ——解码芯片。
- [0200] 根据图19中所示的本原理的示例,在两个远程设备A和B之间通过通信网络NET的发送背景中,设备A包括与存储器RAM和ROM有关的处理器,该存储器RAM和ROM被配置为实现关于图1-图15描述的用于编码有色点云的方法,并且设备B包括与存储器RAM和ROM有关的处理器,该存储器RAM和ROM被配置为实现关于图16-图17描述的用于解码的方法。
- [0201] 根据示例,该网络是广播网络,适用于从设备A向包含设备B的解码设备广播编码的有色点云。
- [0202] 意在由设备A发送的信号承载比特流F1-F2中的至少一个。
- [0203] 因此,该信号可以承载基于八叉树的结构O的每个节点的节点信息数据,其指示与节点相关联的立方体是否被分割。
- [0204] 该信号还可以承载立方体分割的最大深度。
- [0205] 该信号还可以承载基于八叉树的结构O的每个叶节点的叶信息数据,其指示与叶节点相关联的叶立方体是否包含有色点云的点。
- [0206] 该信号还可以承载表示特定3D路径和/或特定2D路径的信息数据。
- [0207] 图20示出了当通过基于分组的传输协议来发送数据时这种信号的语法的示例。每个发送的分组P包括报头H和有效载荷PAYLOAD。
- [0208] 根据实施例,有效载荷PAYLOAD可以包括以下元素中的至少一个:
- [0209] ——二进制标志,其指示与基于八叉树的结构O的节点相关联的立方体是否被分割;
  - [0210] ——二进制标志,其指示与基于八叉树的结构O的叶节点相关联的叶立方体是否被分割;
  - [0211] ——表示3D和/或2D路径的比特。
- [0212] 在此描述的各种过程和特征的实现方式可以体现在各种不同的设备或应用中。这种设备的示例包含编码器、解码器、处理来自解码器的输出的后处理器、向编码器提供输入

的预处理器、视频编码器、视频解码器、视频编解码器、网络服务器、机顶盒、膝上型计算机、个人计算机、蜂窝电话、PDA、HMD、智能眼镜以及用于处理图像或视频的任何其他设备或其他通信设备。应当清楚的是,设备可以是移动的,并且甚至安装在移动交通工具中。

[0213] 另外,可以通过由处理器执行的指令来实现方法,并且可以将这样的指令(和/或由实现方式产生的数据值)存储在计算机可读存储介质上。计算机可读存储介质可以采取计算机可读程序产品的形式,计算机可读程序产品体现在一个或多个计算机可读介质中,并且具有可由计算机执行的在其上体现的计算机可读程序代码。如在此使用的计算机可读存储介质被认为是非暂时性存储介质,给出在其中存储信息的固有能力以及提供从中取回信息的固有能力。计算机可读存储介质可以是例如但不限于电子的、磁性的、光学的、电磁的、红外的或半导体系统、装置或设备,或前述的任何适当组合。应当理解的是,以下内容虽然提供可以应用本原理的计算机可读存储介质的更具体示例,但是仅仅是本领域普通技术人员容易理解的说明性且非详尽的列表:便携式计算机软盘;硬盘;只读存储器(ROM);可擦除可编程只读存储器(EPROM或闪存);便携式光盘只读存储器(CD-ROM);光存储设备;磁存储设备;或前述内容的任何适当组合。

[0214] 指令可以形成有形地体现在处理器可读介质上的应用程序。

[0215] 指令可以在例如硬件、固件、软件或其组合中。指令可以存在于例如操作系统、单独的应用或这两者的组合中。因此,处理器可以表征为例如被配置为执行处理的设备和包含具有用于执行处理的指令的处理器可读介质(诸如存储设备)的设备两者。此外,除了或代替指令,处理器可读介质可以存储由实现方式产生的数据值。

[0216] 如对本领域技术人员将明显的,实现方式可以产生各种信号,这些信号被格式化以承载例如可以被存储或发送的信息。该信息可以包含例如用于执行方法的指令,或者由所描述的实现方式之一产生的数据。例如,信号可以被格式化为承载用于写入或读取本原理的所描述示例的语法的规则作为数据,或者承载由本原理的所描述示例写入的实际语法值作为数据。这样的信号可以被格式化为例如电磁波(例如,使用频谱的射频部分)或基带信号。格式化可以包含例如对数据流进行编码以及利用经编码数据流来调制载波。信号承载的信息可以是例如模拟或数字信息。如已知的,信号可以通过各种不同的有线或无线链路来发送。信号可以存储在处理器可读介质上。

[0217] 已经描述了多个实现方式。然而,将理解的是,可以进行各种修改。例如,可以组合、补充,修改或移除不同实现方式的元素以产生其他实现方式。另外,本领域普通技术人员将理解,其他结构和处理可以代替所公开的结构和处理,并且所得到的实现方式将与所公开的实现方式至少基本相同的方式执行与所公开的实现方式至少基本相同的功能,以达到与所公开的实现方式至少基本相同的结果。因此,本申请考虑这些和其他实现方式。

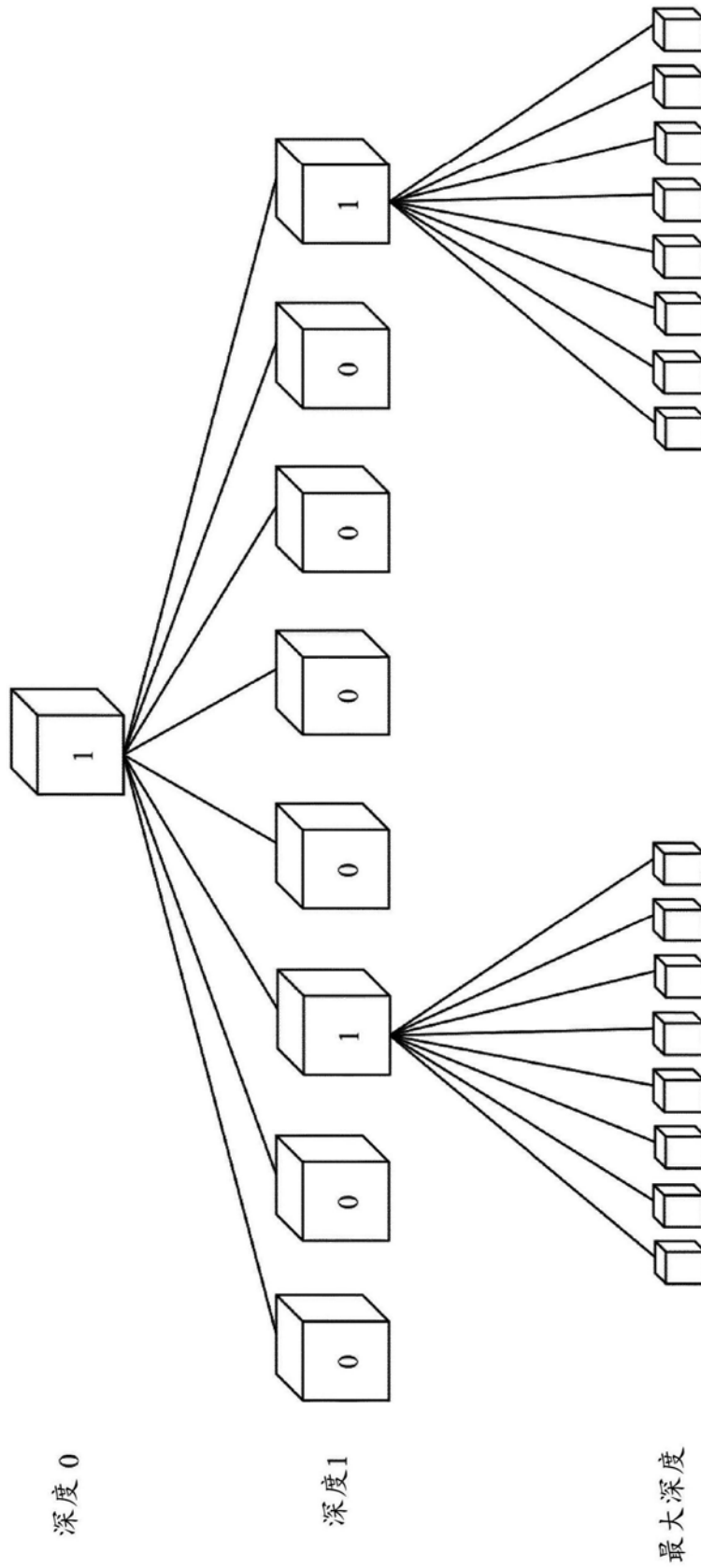


图1

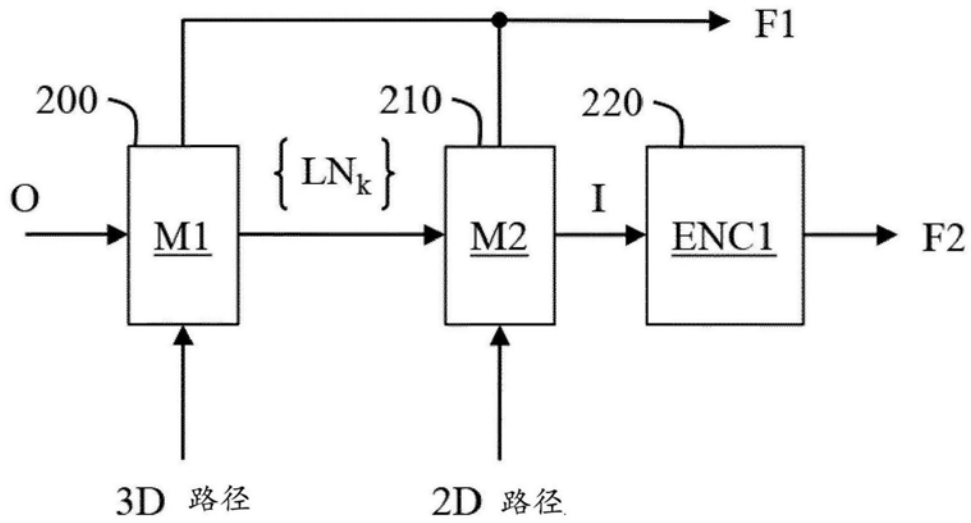


图2

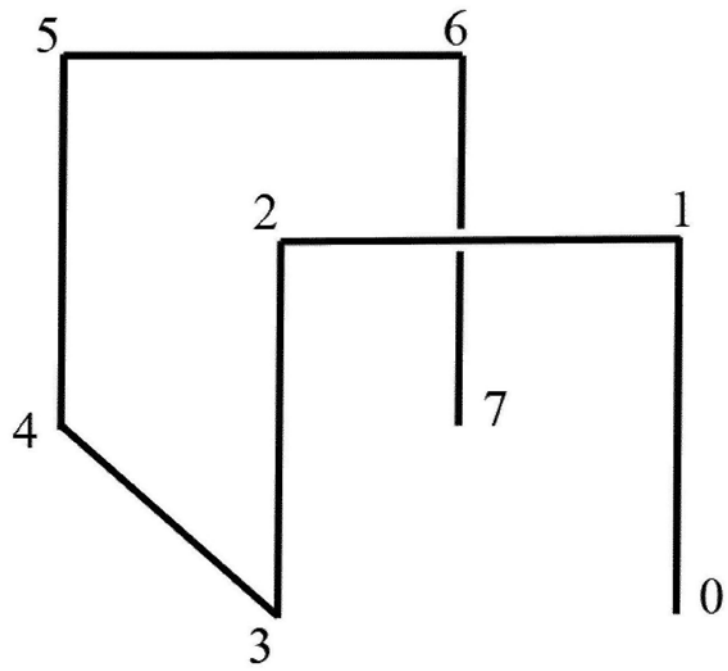


图4

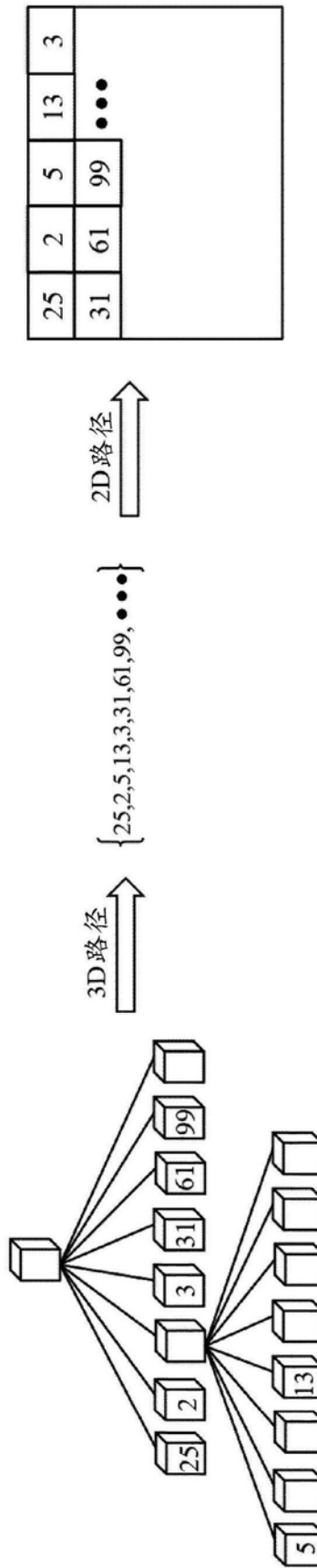


图3

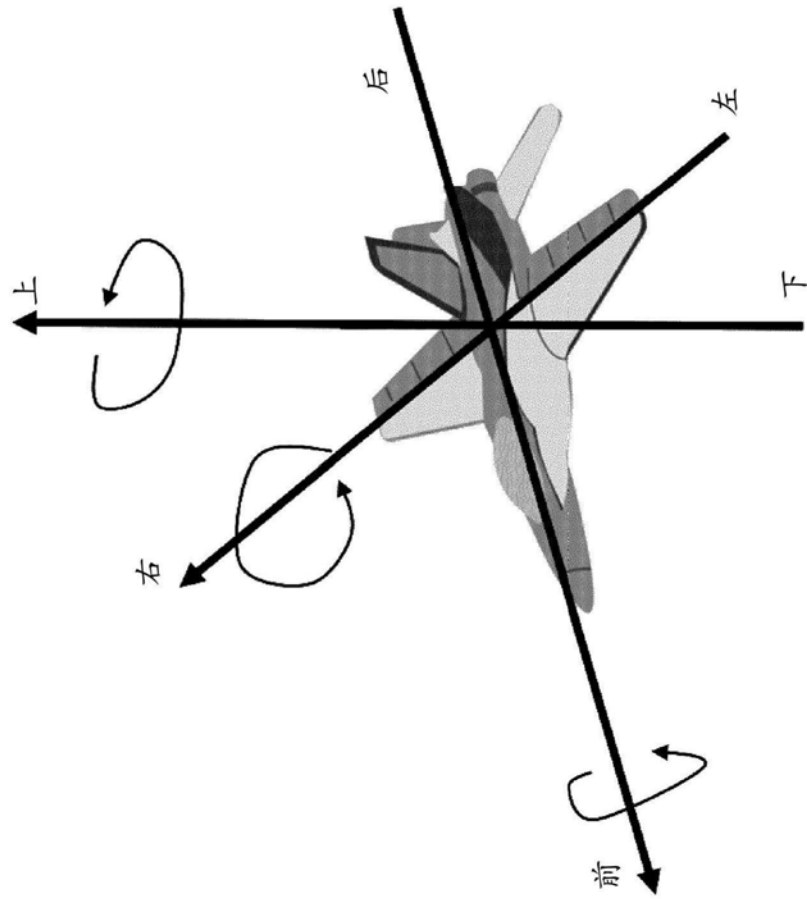


图5

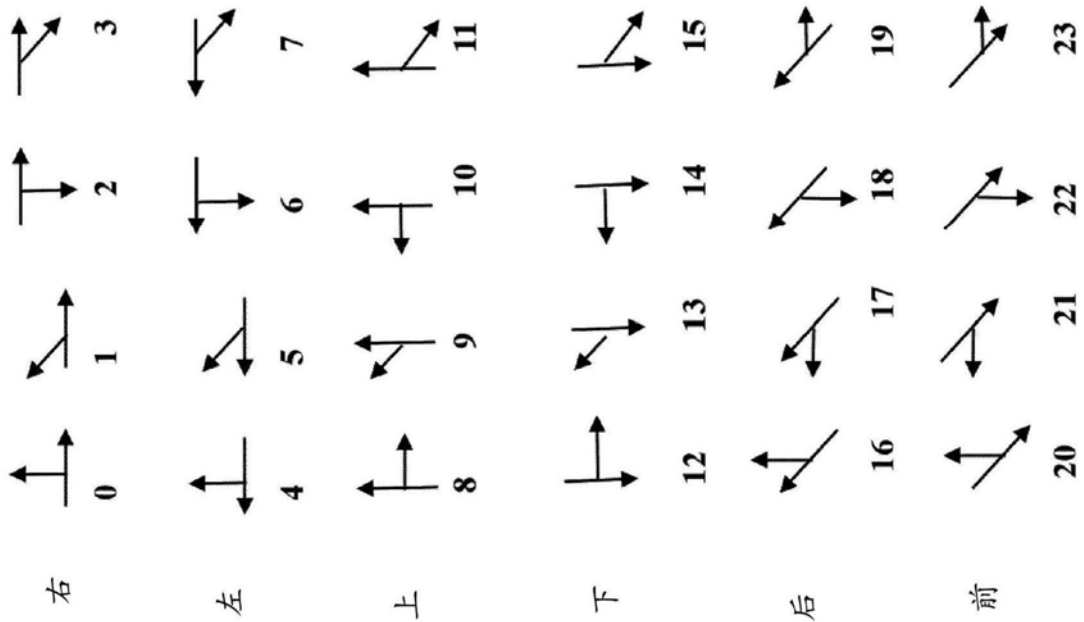


图6

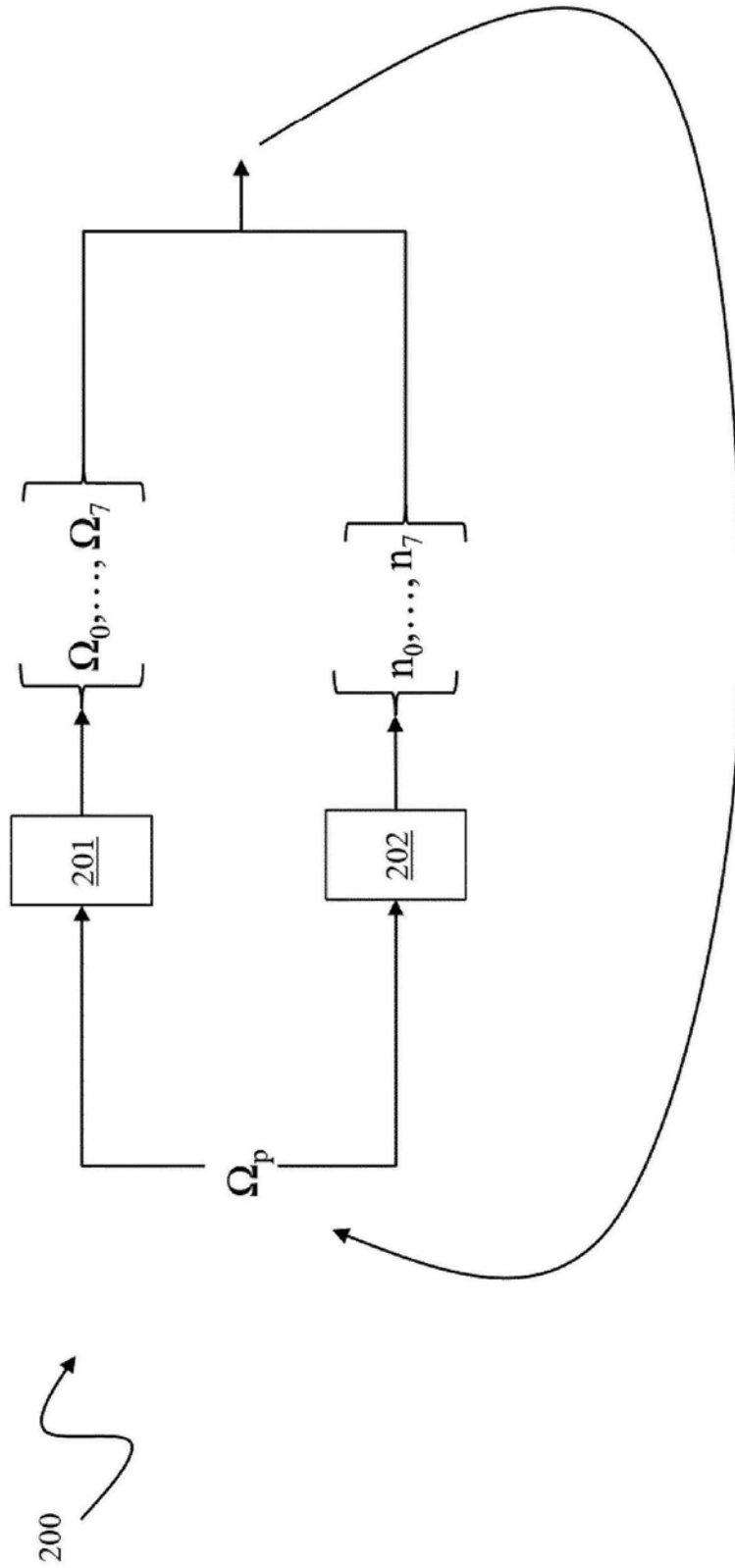


图7

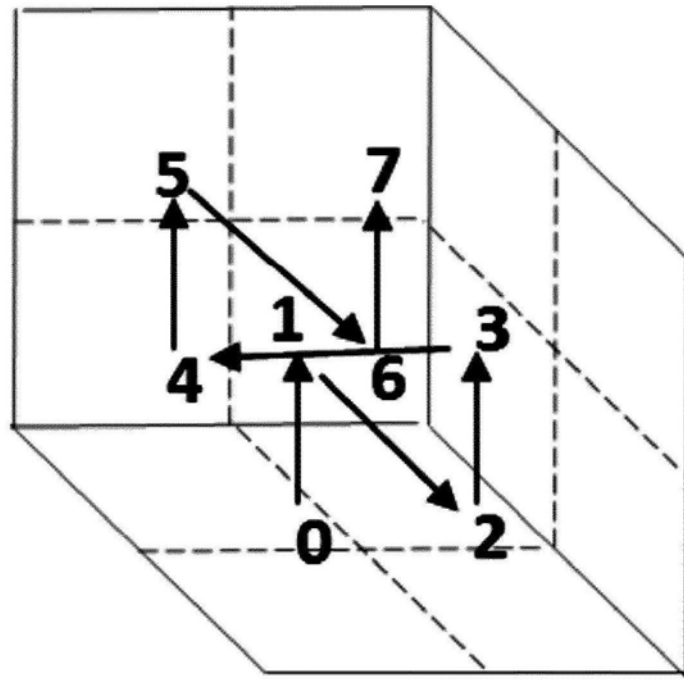


图8

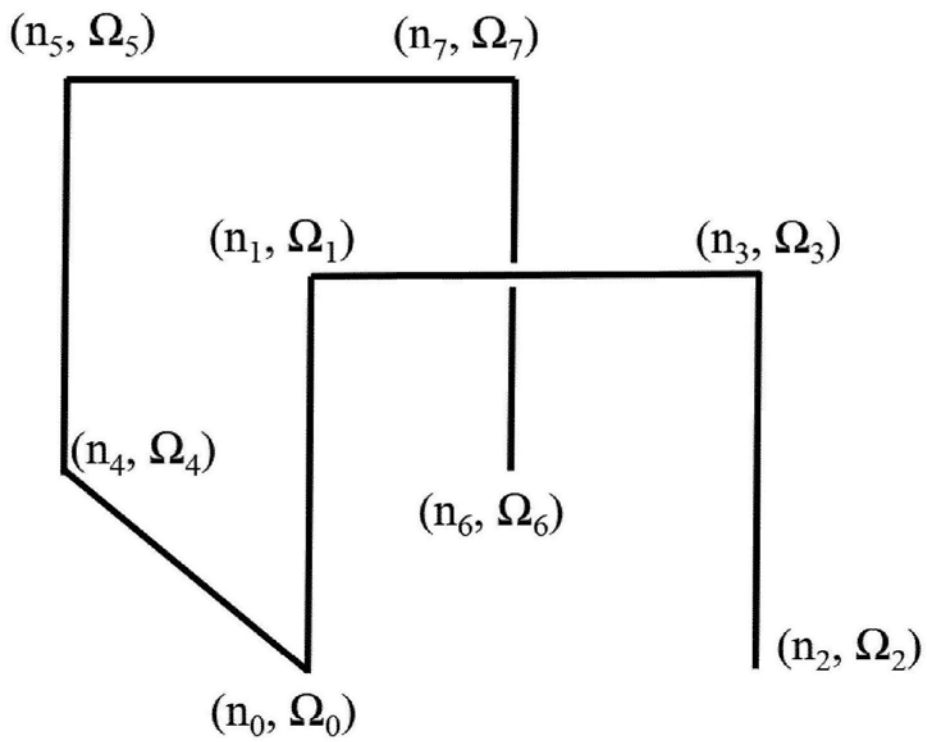


图9



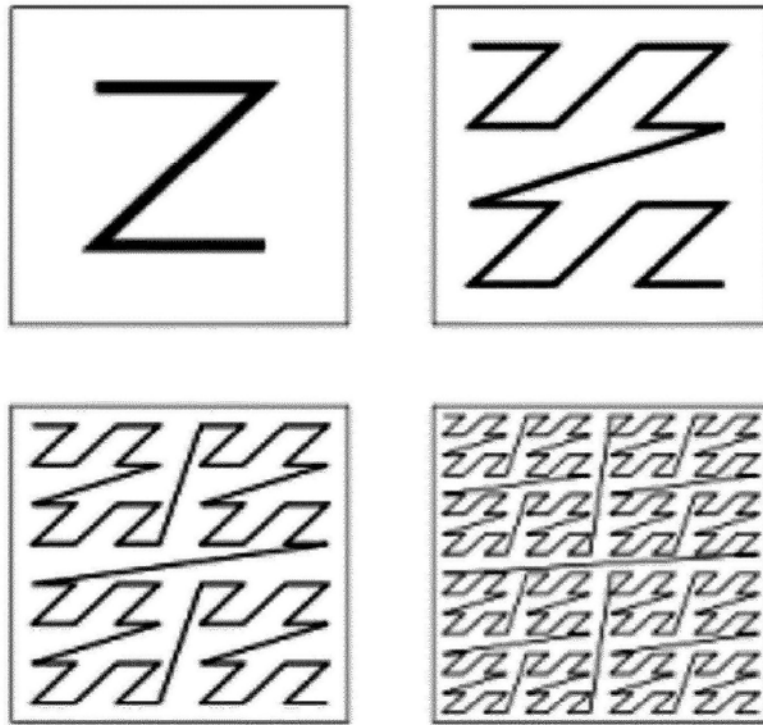


图12

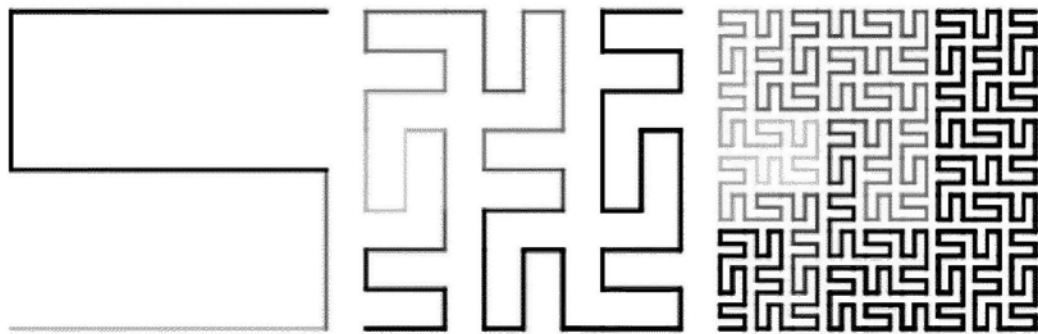


图13

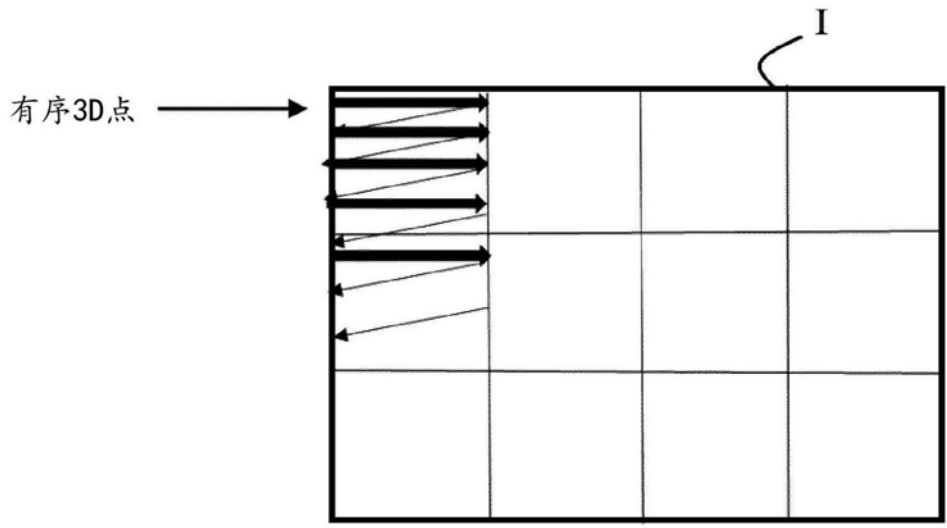


图14

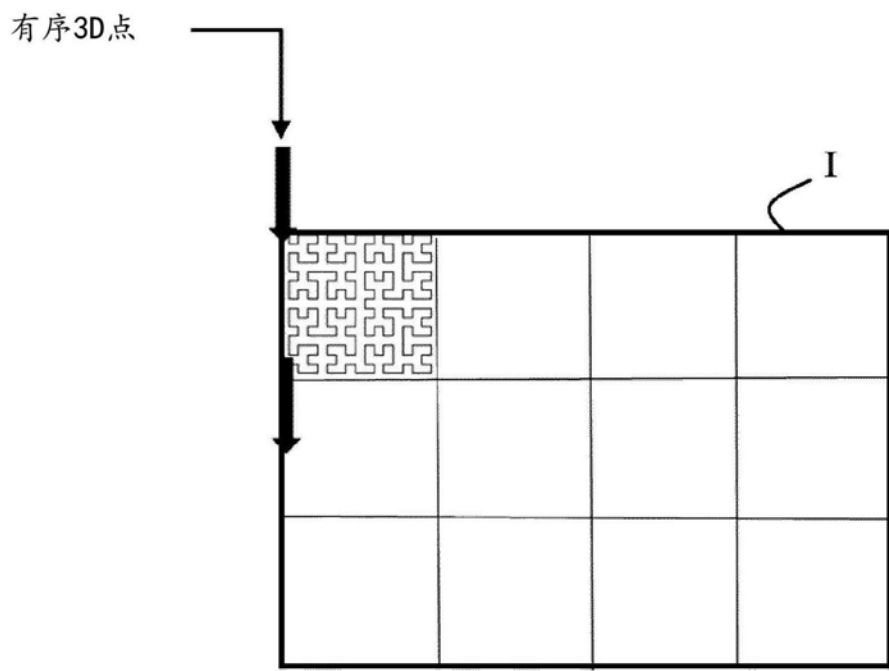


图15

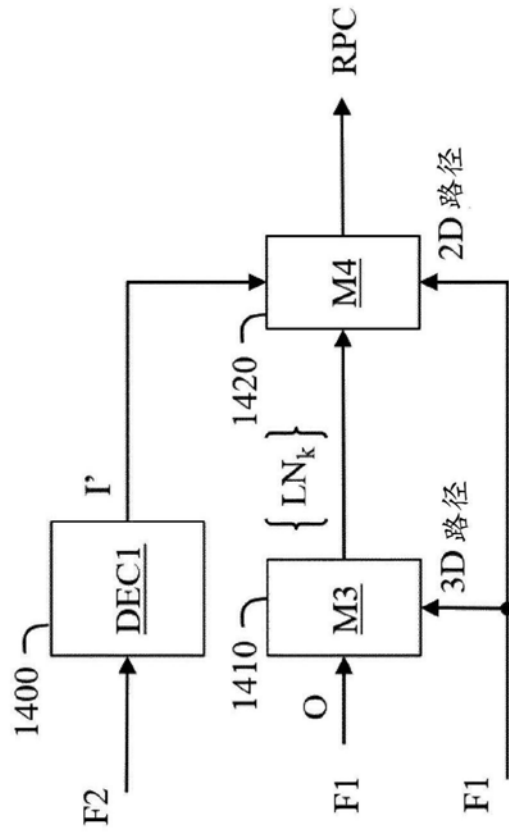


图16

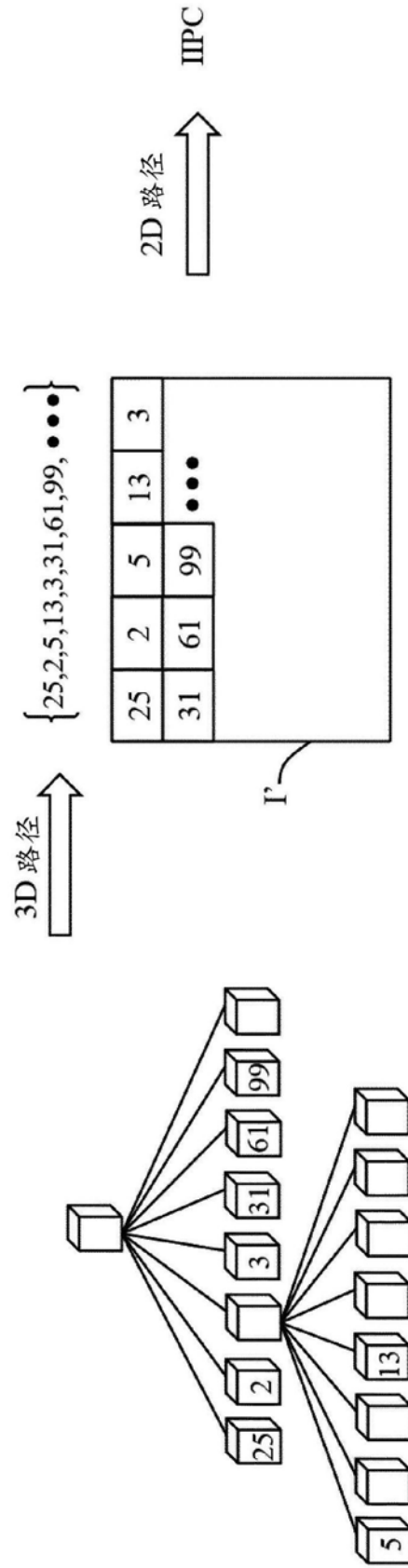


图17

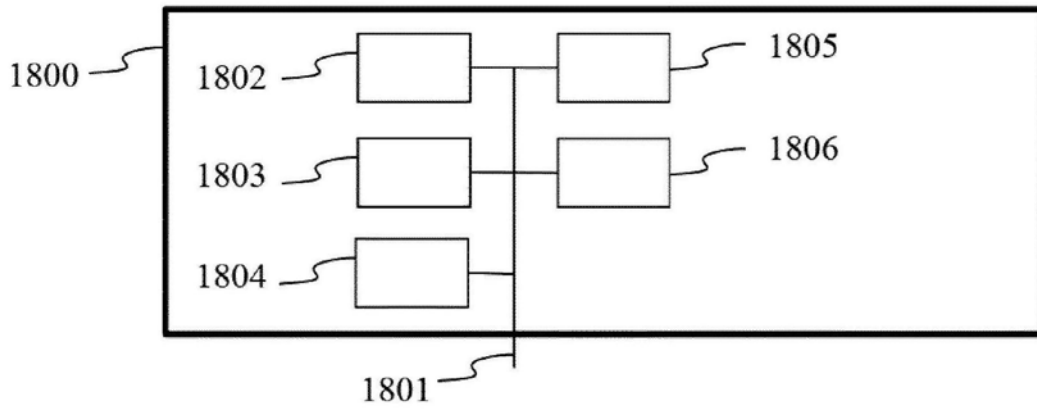


图18



图19

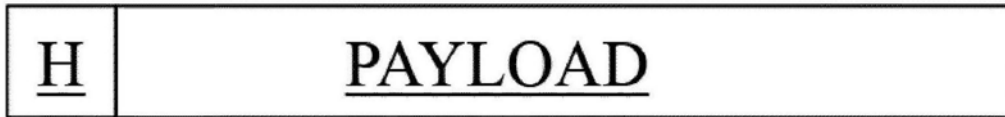


图20