(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112956204 B
(45)授权公告日 2025.02.11

(21)申请号 201980065680.6
(22)申请日 2019.10.04
(65)同一申请的已公布的文献号
申请公布号 CN 112956204 A
(43)申请公布日 2021.06.11
(30)优先权数据
18306317.1 2018.10.05 EP
18306334.6 2018.10.09 EP
(85)PCT国际申请进入国家阶段日
2021.03.30
(86)PCT国际申请的申请数据
PCT/US2019/054616 2019.10.04
(87)PCT国际申请的公布数据
WO2020/072853 EN 2020.04.09

(73)专利权人 交互数字VC控股公司
地址 美国特拉华州
(72)发明人 J·里卡德 C·盖德 J·拉奇
(74)专利代理机构 北京润平知识产权代理有限公司 11283
专利代理师 肖冰滨 王晓晓

(51)Int.Cl.
H04N 19/597(2006.01)
(56)对比文件
US 2019087979 A1,2019.03.21
US 2019197739 A1,2019.06.27
审查员 夏团兵

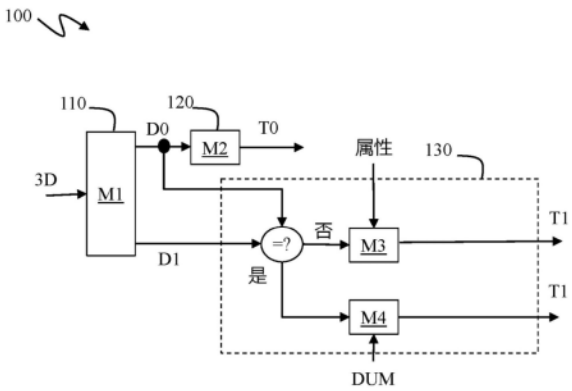
权利要求书3页 说明书12页 附图6页

(54)发明名称

一种用于编码/重建3D点的方法和设备

(57)摘要

本实施例涉及一种用于对3D点进行编码的方法,该3D点的几何形状由几何图像表示并且属性由属性图像表示。该方法检查所述几何图像中的第一几何图像中的像素的深度值和所述几何图像中的第二几何图像中的共位像素的深度值是否不相同(不相同的值)。当所述第一几何图像中的像素的深度值和所述第二几何图像中的共位像素的深度值不相同时,则该方法指派(编码)根据所述第一几何图像中的所述像素的2D空间坐标和所述第二几何图像中的所述共位像素的深度值所定义的3D点的属性。



CN 112956204 B

1.一种方法,包括对3D点进行编码,所述3D点的几何形状由第一几何图像和第二几何图像表示,并且属性由第一属性图像和第二属性图像表示,其中对3D点进行编码包括:

对与一组点云帧有关的语法元素进行编码,

所述语法元素的第一值指示第一几何图像中的像素的深度值和第二几何图像中的共位像素的深度值将在使用所述第二几何图像中的所述共位像素的所述深度值来重建3D点之前被比较,

所述语法元素的第二值指示所述第一几何图像中的所述像素的所述深度值和所述第二几何图像中的所述共位像素的所述深度值将不被比较,编码所述第一几何图像和所述第二几何图像,所述第一几何图像和所述第二几何图像存储了被投影到投影平面的点云帧的3D点的深度值。

2.根据权利要求1所述的方法,其中3D点的属性为颜色值或纹理值。

3.根据权利要求1所述的方法,其中,所述第一几何图像、所述第二几何图像和所述语法元素在被构造为多个块、块的面片和面片的帧的比特流中被编码,并且其中,响应于确定所述语法元素等于所述第一值,另一语法元素在帧级别、在面片级别或在块级别在所述比特流中被编码,所述另一语法元素指示与所述语法元素相同的内容。

4.根据权利要求1所述的方法,还包括:

对第一属性图像进行编码,所述第一属性图像存储了投影在所述投影平面上的所述3D点的属性值,所述3D点的深度值存储在所述第一几何图像中,其中,第一3D点的第一属性值存储在所述第一属性图像中的共位像素中,

对第二属性图像进行编码,所述第二属性图像存储了投影在所述投影平面上的所述3D点的属性值,其中,响应于确定第一几何图像中的像素的第一深度值和所述第二几何图像中的共位像素的第二深度值是相同的,向所述第二属性图像中的所述共位像素指派伪属性值。

5.根据权利要求4所述的方法,其中,所述伪属性是另一属性图像的所述共位像素的属性,或者所述伪属性是与位于所述像素周围的相邻像素相关联的属性的平均值。

6.一种方法,包括:

对与一组点云帧有关的语法元素进行解码,

所述语法元素的第一值指示第一几何图像中的像素的深度值和第二几何图像中的共位像素的深度值将在使用所述第二几何图像中的所述共位像素的所述深度值来重建3D点之前被比较,

所述语法元素的第二值指示所述第一几何图像中的所述像素的所述深度值和所述第二几何图像中的所述共位像素的所述深度值将不被比较,解码所述第一几何图像和所述第二几何图像,所述第一几何图像和所述第二几何图像存储了被投影到投影平面的点云帧的3D点的深度值,

根据所述第一几何图像中的第一2D坐标和所述第一几何图像的在所述第一2D坐标处的第一像素的第一深度值,重建第一3D点,

响应于确定所解码的语法元素等于所述第一值,确定所述第一像素的所述第一深度值与所述第二几何图像中的共位像素的第二深度值是否相同,

响应于确定所述第一深度值和所述第二深度值不相同,从所述第一2D坐标和所述第二

深度值重建第二3D点。

7.根据权利要求6所述的方法,其中携带3D点的编码属性的比特流被构造为多个块、块的面片和面片的帧,其中响应于确定所解码的语法元素等于所述第一值,另一语法元素在帧级别、在面片级别或在块级别被用信号发送,该另一语法元素指示与所述语法元素相同的内容。

8.根据权利要求6所述的方法,其中响应于确定所述第一深度值和所述第二深度值是相同的,跳过从所述第二几何图像中的共位像素的2D空间坐标和所述第二深度值重建所述第二3D点。

9.一种用于对3D点进行编码的装置,所述3D点的几何形状由第一几何图像和第二几何图像表示并且属性由第一属性图像和第二属性图像表示,其中所述装置包括处理器,所述处理器被配置为:

对与一组点云帧有关的语法元素进行编码,

所述语法元素的第一值指示第一几何图像中的像素的深度值和第二几何图像中的共位像素的深度值将在使用所述第二几何图像中的所述共位像素的所述深度值来重建3D点之前被比较,

所述语法元素的第二值指示所述第一几何图像中的所述像素的所述深度值和所述第二几何图像中的所述共位像素的所述深度值将不被比较,编码所述第一几何图像和所述第二几何图像,所述第一几何图像和所述第二几何图像存储了被投影到投影平面的点云帧的3D点的深度值。

10.根据权利要求9所述的装置,其中3D点的属性为颜色值或纹理值。

11.根据权利要求9所述的装置,其中,所述第一几何图像、所述第二几何图像和所述语法元素在被构造为多个块、块的面片和面片的帧的比特流中被编码,并且其中,响应于确定所述语法元素等于所述第一值,另一语法元素在帧级别、在面片级别或在块级别在所述比特流中被编码,所述另一语法元素指示与所述语法元素相同的内容。

12.根据权利要求9所述的装置,其中所述处理器进一步被配置为:

对第一属性图像进行编码,所述第一属性图像存储了投影在所述投影平面上的所述3D点的属性值,所述3D点的深度值存储在所述第一几何图像中,其中,第一3D点的第一属性值存储在所述第一属性图像中的共位像素中,

对第二属性图像进行编码,所述第二属性图像存储了投影在所述投影平面上的所述3D点的属性值,其中,响应于确定所述第一几何图像中的像素的第一深度值和所述第二几何图像中的共位像素的第二深度值是相同的,向所述第二属性图像中的所述共位像素指派伪属性值。

13.根据权利要求12所述的装置,其中所述伪属性是另一属性图像的所述共位像素的属性,或所述伪属性是与位于所述像素周围的相邻像素相关联的属性的平均值。

14.一种用于从表示3D点的几何形状的几何图像重建所述3D点的装置,其中所述装置包括处理器,所述处理器被配置为:

对与一组点云帧有关的语法元素进行解码,

所述语法元素的第一值指示第一几何图像中的像素的深度值和第二几何图像中的共位像素的深度值将在使用所述第二几何图像中的所述共位像素的所述深度值来重建3D点

之前被比较，

所述语法元素的第二值指示所述第一几何图像中的所述像素的所述深度值和所述第二几何图像中的所述共位像素的所述深度值将不被比较，解码所述第一几何图像和所述第二几何图像，所述第一几何图像和所述第二几何图像存储了被投影到投影平面的点云帧的3D点的深度值，

根据所述第一几何图像中的第一2D坐标和所述第一几何图像的在所述第一2D坐标处的第一像素的第一深度值，重建第一3D点，

响应于确定所解码的语法元素等于所述第一值，确定所述第一像素的所述第一深度值与所述第二几何图像中的共位像素的第二深度值是否相同，

响应于确定所述第一深度值和所述第二深度值不相同，从所述第一2D坐标和所述第二深度值重建第二3D点。

15.根据权利要求14所述的装置，其中携带3D点的编码属性的比特流被构造为多个块、块的面片和面片的帧，其中响应于确定所解码的语法元素等于所述第一值，另一语法元素在帧级别、在面片级别或在块级别用信号发送，该另一语法元素指示与所述语法元素相同的内容。

16.根据权利要求14所述的装置，其中响应于确定所述第一深度值和所述第二深度值是相同的，跳过从所述第二几何图像中的共位像素的2D空间坐标和所述第二深度值重建所述第二3D点。

17.一种非暂时性存储介质，承载程序代码的指令，所述程序代码用于当所述程序在计算设备上执行时执行根据权利要求1所述的方法的步骤。

18.一种非暂时性存储介质，承载程序代码的指令，所述程序代码用于当所述程序在计算设备上执行时执行根据权利要求6所述的方法的步骤。

一种用于编码/重建3D点的方法和设备

技术领域

[0001] 本实施例主要涉及3D点的编码和重建。具体地,但不排他地,本实施例的技术领域涉及表示3D对象的外表面的点云的编码/重建。

背景技术

[0002] 本部分旨在向读者介绍可能与下面描述和/或要求保护的本发明实施例的各方面相关的技术的各方面。相信该讨论有助于向读者提供背景信息,以便于更好地理解本实施例的各个方面。因此,应当理解,这些陈述应就此而论来阅读,而不是作为对现有技术的承认。

[0003] 点云是某一坐标系中的一组数据点。在三维坐标系(3D空间)中,这些点通常旨在表示3D对象的外表面。点云的每个点通常由其位置(3D空间中的X、Y和Z坐标)定义,并且可能由其他相关属性定义,例如颜色(在例如RGB或YUV颜色空间中表示的颜色)、透明度、反射率、两分量法向量等。

[0004] 通常将点云表示为一组6分量点(X,Y,Z,R,G,B)或等效地(X,Y,Z,Y,U,V),其中(X,Y,Z)定义3D空间中的色点的坐标,并且(R,G,B)或(Y,U,V)定义该色点的颜色。

[0005] 点云可以是静态的或动态的,这取决于该云是否相对于时间演变。应当注意,在动态点云的情况下,点的数目不是恒定的,而是相反,通常随着时间演变。因此,动态点云是一组点的时间顺序列表。

[0006] 实际上,点云可以用于各种目的,诸如文化遗产/建筑,其中像雕像或建筑的对象以3D扫描,以便共享该对象的空间配置而不发送或访问它。而且,这是一种确保在对象可能被破坏(例如被地震破坏的寺庙)的情况下保留关于该对象的知识的方式。这种点云通常是静态的、有颜色的和巨大的。

[0007] 另一使用情况是在使用3D表示的地形图和制图中,地图不限于平面且可包含起伏。谷歌地图现在是3D地图的良好示例,但是使用的是网格而不是点云。然而,点云可以是用于3D地图的合适的数据格式,并且这样的点云通常是静态的、有颜色的和巨大的。

[0008] 汽车工业和自动驾驶汽车也是其中可以使用点云的领域。自动驾驶汽车应当能够“探测”它们的环境,以基于它们紧邻的现实来做出良好的驾驶决定。像LIDAR的典型传感器产生由决策引擎所使用的动态点云。这些点云不旨在被人类查看,并且它们通常是小的,不一定是有色的,并且是动态的,具有高捕获频率。它们可以具有其他属性,例如由LIDAR提供的反射率,因为该属性是关于被感测物体的材料的良好信息并且可以帮助进行所述决策。

[0009] 虚拟现实和沉浸式世界近来已经成为热点主题,并且被很多人预见为2D平面视频的未来。基本思想是使观看者沉浸在他周围的环境中,这与标准TV相反,在标准TV中,观看者仅能够看到他前方的虚拟世界。根据观看者在环境中的自由度,在所述沉浸度中存在若干层次。有色点云是分布虚拟现实(或VR)世界的良好格式候选。它们可以是静态的或动态的,并且通常具有平均的大小的,比如一次不超过数百万个点。

[0010] 仅当比特流的大小足够低以允许对终端用户的实际存储/传输时,点云压缩才将

成功地存储/传输用于沉浸式世界的3D对象。

[0011] 能够以比特率的合理消耗而将动态点云分发给终端用户同时保持可接受的(或者优选地非常好的)体验质量是至关重要的。为了使沉浸式世界的分发链切实可行,这些动态点云的有效压缩是关键点。

[0012] 基于图像的点云压缩技术由于其压缩效率和低复杂度的组合而变得越来越流行。它们以两个主要步骤进行:首先,它们将点云(即,3D点)投影(正交投影)到至少一个2D图像平面上。例如,因此获得至少一个2D几何形状(也表示为深度)图像以表示所述点云的几何形状,即,所述3D点在3D空间中的空间坐标,并且还获得至少一个2D属性(也表示为纹理)图像以表示与所述点云的所述3D点相关联的属性,例如与那些3D点相关联的纹理/颜色信息。接着,这些技术用传统视频编码器将这种几何和属性图像编码成至少一个几何和属性层。

[0013] 基于图像的点云压缩技术通过利用2D视频编码器的性能来实现良好的压缩性能,同时它们通过使用简单的投影方案来保持低复杂度,所述编码器例如HEVC(“ITU的ITU-TH.265电信标准化部门(10/2014),系列H:视听和多媒体系统,视听服务的基础设施-运动视频的编码、高效视频编码,ITU-TH.265建议(ITU-T H.265Telecommunication standardization sector of ITU(10/2014),series H:audiovisual and multimedia systems,infrastructure of audiovisual services-coding of moving video,High efficiency video coding,Recommendation ITU-T H.265)”)。

[0014] 基于图像的点云压缩技术的挑战之一在于:点云可能不适合投影到图像上,尤其是在点分布遵循具有许多褶皱的表面(凹/凸区域,如在衣服中)时或者在点分布根本不遵循表面(如在皮毛或毛发中)时。在这些情况下,基于图像的点云压缩技术遭受低压缩效率(需要许多小投影,从而降低2D视频压缩的效率)或质量差(由于将点云投影到表面上的困难)。

[0015] 在现有技术中使用的减轻这个问题的方法之一在于将多个几何形状和属性信息投影到图像的相同空间位置上。这意味着,可以为点云的每个3D点生成具有相同投影坐标(像素的相同2D空间坐标)的若干几何和/或属性图像。

[0016] 例如,在ISO/IEC JTC1/SC29/WG11 MPEG2018/N17767,Ljubljana,2018年7月(附录A)中定义的所谓的测试模型类型2点云编码器(TMC2)中便是这种情况,其中点云被正交地投影到投影平面上。针对所述投影平面的每个坐标,生成两个几何图像:一个代表与最近点相关联的深度值(最小深度值),另一个代表最远点的深度值(最大深度值)。然后,从最小深度值(D0)生成第一几何图像,从最大深度值的绝对值(D1)生成第二几何图像,其中D1-D0小于或等于最大表面厚度。还产生与第一(D0)和第二(D1)几何图像相关联的第一和第二属性图像。然后,使用诸如HEVC的任何传统视频编解码器对所述属性图像和几何图像进行编码和解码。因此,通过对包括在所解码的第一和第二几何图像中的信息进行去投影来重建所述点云的几何形状,并且将属性与根据包括在所解码的属性图像中的信息而被重建的3D相关联。

[0017] 捕获两个几何(和两个属性)值的缺点是:从两个几何图像系统地重建两个3D点,当第一几何图像中的像素的深度值等于第二几何图像中的共位(co-located)像素的深度值时,创建这样重复的重建3D点。接下来,对不必要的重复的点进行编码增加了用于发送编码的3D点集合的比特率。此外,在编码和解码侧,用于处理这种假复制3D点的计算和存储资

源也被浪费。

发明内容

[0018] 以下给出了本发明实施例的简化概述,以便提供对本发明实施例的一些方面的基本理解。该概述不是对本实施例的广泛综述。其不旨在标识本实施例的关键或重要元素。以下概述仅以简化形式呈现本实施例的一些方面,作为以下提供的更详细描述的序言。

[0019] 本发明的实施例旨在利用用于对其几何形状由几何图像表示并且属性由属性图像表示的3D点进行编码的方法和装置来弥补现有技术的至少一个缺点。当所述几何图像中的第一几何图像中的像素的深度值和所述几何图像中的第二几何图像中的共位像素的深度值不相同时,该方法向所述属性图像的共位像素指派3D点的属性,该3D点的几何形状是根据所述第一几何图像中的共位像素的2D空间坐标和所述第二几何图像中的所述共位像素的深度值来定义的。

[0020] 根据实施例,该方法还包括:向所述属性图像的像素指派伪(dummy)属性,这发生在当所述第一几何图像中的共位像素的深度值和所述第二几何图像中的共位像素的深度值相同时。

[0021] 根据一个实施例,所述伪属性是另一属性图像的所述共位像素的属性。

[0022] 根据实施例,所述伪属性是与位于所述像素周围的相邻像素相关联的属性的平均值。

[0023] 根据实施例,该方法还包括:发送信息数据,该信息数据指示了在从所述几何图像重建3D点之前,所述第一几何图像中的像素的深度值和所述第二几何图像中的共位像素的深度值是否被比较。

[0024] 根据本发明的另一方面,本实施例涉及一种携带3D点的编码属性的比特流,所述比特流被构造为多个块、块的面片(patch)和面片的帧,其中所述信息数据在帧组级别、帧级别、面片级别或块级别有效。

[0025] 根据本发明的另一方面,本实施例涉及一种用于从表示3D点的几何形状的几何图像重建所述3D点的方法,其中该方法包括:根据所述几何图像中的第一几何图像中的像素的2D空间坐标和所述几何图像中的第二几何图像中的共位像素的深度值来重建3D点,这发生在当所述第一几何图像中的所述像素的深度值和所述第二深度图像中的所述共位像素的深度值不相同时。

[0026] 根据实施例,该方法还包括:接收信息数据,该信息数据指示了在从所述几何图像重建3D点之前,所述第一几何图像中的像素的深度值和所述第二几何图像中的共位像素的深度值是否被比较。

[0027] 根据一个实施例,携带3D点的编码属性的比特流被构造为多个块、块的面片和面片的帧,则所述信息数据在帧组级别、在帧级别、在面片级别或在块级别有效。

[0028] 根据实施例,3D点的属性是颜色值或纹理值。

[0029] 至少一个实施例中的一个或多个还提供了一种装置、计算机程序产品、非瞬态计算机可读介质和比特流。

[0030] 从结合附图进行的以下示例描述中,本发明实施例的特定性质以及本发明实施例的其它目的、优点、特征和用途将变得显而易见。

附图说明

[0031] 在附图中,示出了本实施例的示例。其显示了:

[0032] -图1示意性地示出了根据本实施例的示例的用于对与3D点相关联的属性进行编码的方法100的步骤的图;

[0033] -图2示意性地示出了根据本实施例的示例的用于从表示3D点的几何形状的几何图像重建所述3D点的方法200的步骤的图。

[0034] -图3示意性地示出了TMC2中定义的用于对点云的几何形状和属性进行编码的方法;

[0035] -图4示意性地示出了TMC2中定义的用于解码点云的几何形状和属性的方法;

[0036] -图5示出了其中实现了各个方面和实施例的系统的示例的框图;

[0037] -图6示出根据本实施例修改的TCM2的语法元素

[0038] “group_of_frames_header()”的示例;

[0039] -图7、图7a和图7b示出了根据本发明实施例修改的TCM2的表示为

[0040] “frame_auxiliary_information(frame_index)”的语法元素的示例;以及

[0041] -图8、图8a和图8b示出了根据本发明实施例修改的TCM2的表示为

[0042] “frame_auxiliary_information(frame_index)”的语法元素的另一示例。

[0043] 相似或相同的元件用相同的参考数字表示。

[0044] 本发明实施例的描述

[0045] 以下将参照附图更全面地描述本发明,在附图中示出了本发明的示例。然而,本发明的实施例可以以许多替代形式实施,且不应解释为限于本文所陈述的示例。因此,尽管本发明的实施例易于进行各种修改和替换形式,但是其具体示例在附图中以示例的方式示出并且将在此详细描述。然而,应理解,不希望将本发明实施例限于所揭示的特定形式,而是相反,本说明书将涵盖属于如权利要求书所界定的本发明实施例的精神和范围内的所有修改、等效物和替代方案。

[0046] 本文所用的术语仅是为了描述特定示例的目的,而不是要限制本发明的实施例。如本文所用,单数形式“一(a)”、“一(an)”和“该”旨在也包括复数形式,除非上下文另有明确指示。还将理解,术语“包括”和/或“包含”在本说明书中使用时,指定所陈述的特征、整数、步骤、操作、元件和/或组件的存在,但不排除一个或多个其它特征、整数、步骤、操作、元件、组件和/或其群组的存在或添加。此外,当元件被称为“响应”或“连接”到另一元件时,它可以直接响应或连接到另一元件,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”到其它元件时,不存在中间元件。如本文所用,术语“和/或”包括一个或多个相关的所列项目的任意者和所有组合,并且可以缩写为“/”。

[0047] 应当理解,尽管术语第一、第二等可以在这里用来描述各种元件,但是这些元件不应当受这些术语的限制。这些术语仅用于将一个元件与另一个元件区分。例如,在不脱离本实施例的教导的情况下,第一元件可以被称为第二元件,并且类似地,第二元件可以被称为第一元件。

[0048] 尽管一些图包括通信路径上的箭头以示出通信的主要方向,但是应当理解,通信可以在与所描绘的箭头相反的方向上发生。

[0049] 关于框图和操作流程图描述了一些示例,其中每个框表示电路元件、模块或代码

的一部分,其包括用于实现指定的逻辑功能(多个)的一个或多个可执行指令。还应当注意,在其它实现中,在所述框中所提到的(一个或多个)功能可以不按所提到的顺序发生。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。

[0050] 本文中提到“根据示例”或“在示例中”意味着结合该示例描述的特定特征、结构或特性可包含在本实施例的至少一个实施方式中。在说明书中的各个地方出现的根据示例“或”在示例中“的短语不一定全部指代相同的示例,也不是必须与其他示例互斥的单独的或替代的示例。

[0051] 权利要求中出现的附图标记仅作为说明,并且不应对权利要求的范围具有限制作用。

[0052] 虽然没有明确描述,但是本示例和变型可以以任何组合或子组合来采用。

[0053] 本实施例被描述用于编码/重建表示点云的两个几何图像和两个属性图像,但是可扩展到编码/重建表示点云序列(时间动态点云)的两个序列(视频)的几何图像和两个序列(视频)的属性图像,因为所述点云序列的点云的几何形状(两个几何图像)和属性(纹理/颜色)然后独立于所述点云序列的另一点云的几何形状(两个几何图像)和属性(纹理/颜色)而被编码/重建。

[0054] 一般而言,本实施例涉及一种用于对其几何形状由几何图像表示的3D点的属性进行编码的方法。所述3D点可以形成表示例如3D对象的外表面的点云。但是该方法不限于点云的编码,并且可以扩展到任何其他3D点的集合。所述3D点的几何形状(3D坐标)被表示为几何图像。

[0055] 所述方法检查所述几何图像中的第一几何图像中的像素的深度值和所述几何图像中的第二几何图像中的共位像素的深度值是否不相同(不相同的值)。当所述第一几何图像中的像素的深度值和所述第二几何图像中的共位像素的深度值不相同时,则该方法指派(编码)根据所述第一几何图像中的所述像素的2D空间坐标和所述第二几何图像中的所述共位像素的深度值所定义的3D点的属性。否则,该方法指派伪值作为所述3D点的属性。

[0056] 因此,该方法通过避免例如在TMC2中发生的对假复制3D点的编码,修改了一组3D点的通常几何形状和属性编码。这避免了浪费编码侧的计算和存储资源,并且限制了发送编码3D点的比特率。

[0057] 本实施例还涉及一种用于从表示3D点的几何形状的几何图像重建所述3D点的方法。

[0058] 该方法检查所述几何图像中的第一几何图像中的像素的深度值和所述几何图像中的第二几何图像中的共位像素的深度值是否不相同。然后,当第一几何图像中的所述像素的深度值和第二深度图像中的所述共位像素的深度值不相同时,该方法从第一几何图像中的所述像素的2D空间坐标和第二几何图像中的所述共位像素的深度值,重建3D点。否则,不重建3D点。

[0059] 因此,该方法通过避免例如在TMC2中发生的假复制3D点的创建,修改了3D点的通常几何形状和属性重建。这避免了在解码侧浪费计算和存储资源。

[0060] 从图像中提取的属性的示例可以是颜色、纹理、法向量等。.

[0061] 图1示意性地示出了根据本实施例的示例的用于对3D点的属性进行编码的方法

100的步骤的图。

[0062] 例如,所述3D点可以形成点云,但是该方法不限于点云并且可以应用于任何3D点集合。

[0063] 在步骤110中,模块M1可获得表示3D点的几何形状的几何图像:所述3D点的三个坐标中的两个由所述几何图像中的像素的2D坐标表示,并且像素值表示所述3D点的第三坐标(深度值)。

[0064] 例如,在TMC2中,所述3D点可以正交地投影到投影平面上,并且两个几何图像D0和D1可以从与所述投影的3D点相关联的深度值获得。D0是第一几何图像,其表示投影平面的最近3D点的深度值,D1是第二几何图像,其表示最远3D点的深度值。所述几何图像可以使用例如诸如HEVC的传统图像/视频编码器来编码。

[0065] 在步骤120中,模块M2可以获得第一属性图像,例如T0,其表示从所述获得的几何图像的第一几何图像D0中的像素的2D空间坐标和深度值定义的3D点RP的属性。

[0066] 例如,在TMC2中,从原始3D点获得3D点RP的属性(更多细节参见附录A的2.5节)。使用例如诸如HEVC的传统图像/视频编码器来编码第一属性图像T0(图1中未示出)。

[0067] 在步骤130中,模块可以获得第二属性图像,例如T1。所述第二属性图像T1表示根据所述第一几何图像中的像素的2D空间坐标和深度值定义的补充3D点SP的属性。

[0068] 首先,在步骤130中,模块比较第一几何图像(例如D0)中的像素P的深度值和第二几何图像(例如D1)中的共位像素CP的深度值。接下来,当第一几何图像D0中的所述像素P的深度值和第二几何图像D1中的所述共位像素CP的深度值不相同时,模块M3可以向第二属性图像T1中的所述共位像素指派3D点的属性,该3D点的几何形状是根据所述第一几何图像中的所述像素P的2D空间坐标和所述第二几何图像中的所述共位像素CP的深度值来定义的。否则,模块M4可以将伪属性DUM指派给所述第二属性图像中的所述共位像素。

[0069] 使用例如诸如HEVC的传统图像/视频编码器来编码所述第二属性图像T1(图1中未示出)。

[0070] 根据实施例,所述伪属性是所述第一属性图像T0的所述共位像素的属性。

[0071] 根据实施例,所述伪属性是与位于所述像素P周围的相邻像素相关联的属性的平均值。

[0072] 图2示意性地示出了根据本实施例的示例的用于从表示3D点的几何形状的几何图像重建所述3D点的方法200的步骤的图。

[0073] 在步骤210中,模块可以比较所述几何图像的第一几何图像(例如D0)中的像素P的深度值和所述几何图像的第二几何图像(例如D1)中的共位像素CP的深度值。

[0074] 在步骤220中,模块M5可从第一几何图像D0中的像素P的2D空间坐标和深度值定义3D点RP。

[0075] 在步骤230中,当所述第一几何图像D0中的像素P的深度值和所述第二几何图像D0中的共位像素CP的深度值不相同时,模块M6根据第一几何图像(例如D0)的像素P的2D空间坐标和第二几何图像(例如D1)中的共位像素CP的深度值,定义一补充3D点SP。

[0076] 根据第一几何图像D0中的像素的2D空间坐标和深度值定义的3D点RP的属性是第一属性图像T0中的共位像素的值。根据第一几何图像D0中的像素的2D空间坐标和第二几何图像D1中的共位像素的深度值定义的3D点SP的属性是第二属性图像T1中的共位像素的值

(不等于伪值DUM的值)。

[0077] 所述方法100编码3D点的属性,并且特别地,当第一几何图像D0中的共位像素的深度值和第二几何图像D1中的共位像素的深度值相同时,实现第一功能,该第一功能向第二属性图像T1中的像素值指派伪值(步骤130)。

[0078] 因此,该第一功能限制了发送3D点的属性所需的比特率,并且减少了计算和存储资源。

[0079] 所述方法200从表示3D点的几何形状的几何图像重建3D点,并且特别地,当所述第一几何图像的所述像素的深度值和所述第二几何图像的所述共位像素的深度值不相同时,实现第二功能,该第二功能根据所述几何图像中的第一几何图像的像素的2D空间坐标和所述几何图像中的第二几何图像的共位像素的深度值,定义3D点(步骤230)。

[0080] 根据一个变型,当信息数据ID表示第一值时,所述第一和第二功能被启用,并且当所述信息数据ID表示第二值时,所述第一和第二功能被禁用。因此,所述信息数据ID指示了所述方法100或200在编码3D点的属性之前(方法100)或在重建3D点之前(方法200)是否检查第一几何图像的像素的深度值和第二几何图像的所述共位像素的深度值是否不相同。

[0081] 根据一个实施例,在帧组级别上启用/禁用所述第一和第二功能。

[0082] 然后,将信息数据ID与关于一组帧的语法元素相关联。

[0083] 图6展示了TCM2的语法元素“group_of_frames_header()”,其包含表示与一组帧相关联的信息数据ID的“remove_duplicate_coding_group_of_frames”的字段。

[0084] 根据本实施例,图6的这个语法元素可以用于用信号通知所述信息数据ID。

[0085] 根据实施例,在帧组级别和帧级别,启用/禁用所述第一和第二功能。

[0086] 然后,信息数据ID与相对于帧组的语法元素相关联,例如TCM2的语法元素“group_of_frames_header()”(图6),并且信息数据还与相对于帧的语法元素相关联,如图7、图7a和图7b所示。

[0087] 图7展示了TCM2的表示为“frame_auxiliary_information(frame_index)”且如图7、图7a和图7b中所示修改(灰色阴影区域)的语法元素的示例。

[0088] 根据该实施例,图6的语法元素和图7、图7a和图7b的该语法元素可以用于用信号通知所述信息数据ID。

[0089] 根据实施例,在帧组级别、帧级别和面片级别,启用/禁用所述第一和第二功能。面片可以被定义为图像的一部分。

[0090] 然后,信息数据ID与相对于帧组的语法元素相关联,例如TCM2的语法元素“group_of_frames_header()”(图6),信息数据还与相对于帧的语法元素相关联,如图8、图8a和图8b所示。

[0091] 图8展示了TCM2的表示为“frame_auxiliary_information(frame_index)”且如图8、图8a和图8b所示修改(灰色阴影区域)的语法元素的示例。根据该实施例,图6的语法元素和图8、图8a和图8b的该语法元素可以用于用信号通知所述信息数据ID。

[0092] 根据所述最后一个实施例的变型,在帧块级,启用/禁用所述第一和第二功能。

[0093] 例如,当面片与至少一个图像块重叠时,可以用信号通知信息数据ID以指示是否为图像块启用了(或未启用)第一和第二功能。

[0094] 这在面片的某些部分产生密集的3D点。

[0095] 图3示意性地示出了TMC2(附录A)中定义的用于编码点云的几何形状和属性的方法。

[0096] 基本上,编码器在第一(D0)和第二(D1)几何图像中捕获点云PC的几何信息。

[0097] 作为示例,所述第一和第二几何图像在TMC2中如下获得。

[0098] 通过根据点云PC的点处的法向量而对该点云PC的该点进行聚类来获得几何面片(点云PC的3D点的集合)。然后,将所有提取的几何面片投影到2D网格上并打包,同时试图最小化未使用的空间,并保证所述网格的每个TxT(例如,16×16)块与唯一面片相关联,其中T是用信号发送到所述比特流中的用户定义参数。

[0099] 然后,通过利用在所述打包过程期间计算的3D到2D映射,更具体地,利用每个面片的投影区域的打包位置和尺寸,来生成几何图像。更准确地说,令H(u,v)为当前面片的投影到同一像素(u,v)的点的集合。第一层(也称为最近层或第一几何图像D0)存储具有H(u,v)的最小几何值的点。第二层(称为最远层或第二几何图像D1)捕捉H(u,v)的区间[D,D+Δ]内具有最高几何值的点,其中D是第一几何图像D0中的像素的几何值,并且是描述表面厚度的用户定义参数。

[0100] 第一几何图像D0然后输出所述打包过程。还使用填充过程来填充面片之间的空白空间,以便生成适合于视频压缩的分段平滑的第一几何图像。

[0101] 然后,所生成的几何图像/层D0和D1被存储为视频帧,并且使用诸如HEVC的任何传统视频编解码器而被压缩。

[0102] 所述编码器还通过以下而在两个纹理(属性)图像中捕获原始点云PC的属性信息:对所述第一和第二几何图像进行编码/解码,并通过对所述解码的第一和第二几何图像$\widehat{D_0},\widehat{D_1}$进行去投影来重建所述点云的几何形状。一旦重建,就以使颜色信息编码误差最小化的方式,根据原始点云PC的颜色信息而将颜色指派(颜色转移)给所重建的点云的每个点。

[0103] 根据一个实施例,对于每个所重建的点,原始点云中其最近点的颜色被指派为其要被编码的颜色。

[0104] 然后,通过将每个所重建的点的要编码的颜色信息存储与所述几何图像中相同位置(即,(i,u,v)),生成第一和第二属性图像T0、T1。

[0105] 例如,当需要重建点云时,即当需要该点云的几何形状和可能的属性时,方法100和200可用在TMC2(附录A的图1)的编码侧。例如,这是用于生成属性图像和用于重建几何图像的情况。

[0106] 图4示意性地示出了如TMC2所定义的用于解码点云的几何形状和属性的方法。

[0107] 通过对比特流BT进行解码,获得解码的第一几何图像$\widehat{D_0}$和解码的第二几何图像$\widehat{D_1}$。可能地,元数据也被解码以重建点云$\widehat{PC}$的几何形状。

[0108] 因此,通过对所述解码的第一和第二几何图像以及可能的所述元数据进行去投影来重建所述点云的几何形状。

[0109] 当需要所重建的点云时,即,当需要该点云的几何形状时,方法200也可以用在TMC2(附录A的图2)的解码侧。例如,对于重建所述点云的几何形状,就是这种情况。

[0110] 在图1-8b中,所述模块是功能单元,其可以与可区分的物理单元相关或不相关。例

如,这些模块或其中的一些可以被一起引入到唯一的组件或电路中,或者有助于软件的功能。相反,一些模块可能由单独的物理实体组成。与本实施例兼容的设备使用纯硬件来实现,例如使用专用硬件,例如ASIC(专用集成电路)或FPGA(现场可编程门阵列)或VLSI(超大规模集成),或通过嵌入在一器件中的若干集成电子部件或通过硬件和软件部件的共混物来实现。

[0111] 图5示出了其中实现了各个方面和实施例的系统的示例的框图。系统5000可以被实现为包括以下描述的各种组件的设备,并且被配置为执行本申请中描述的一个或多个方面。此类设备的示例包括但不限于各种电子设备,诸如个人计算机、膝上型计算机、智能电话、平板计算机、数字多媒体机顶盒、数字电视接收机、个人视频记录系统、连接的家用电器和服务器。系统5000的元件可以单独地或组合地被实现在单个集成电路、多个IC和/或分立组件中。例如,在至少一个实施例中,系统5000的处理和编码器/解码器元件分布在多个IC和/或分立组件上。在各种实施例中,所述系统5000经由例如通信总线或通过专用输入和/或输出端口而被通信地耦合到其他类似系统或其他电子设备。在各种实施例中,所述系统5000被配置为实现本文中描述的一个或多个方面。

[0112] 所述系统5000包括至少一个处理器5010,其被配置为执行加载在其中的指令,以用于实现例如本文中描述的各个方面。处理器5010可以包括嵌入式存储器、输入输出接口和本领域已知的各种其它电路。所述系统5000包括至少一个存储器5020(例如,易失性存储器设备和/或非易失性存储器设备)。系统5000包括存储设备5040,其可以包括非易失性存储器和/或易失性存储器,这其中包括但不限于EEPROM、ROM、PROM、RAM、DRAM、SRAM、闪存、磁盘驱动器和/或光盘驱动器。作为非限制性示例,所述存储设备5040可以包括内部存储设备、附接的存储设备和/或网络可访问的存储设备。

[0113] 系统5000包括编码器/解码器模块5030,其被配置为例如处理数据以提供所编码的视频或所解码的视频,并且所述编码器/解码器模块5030可以包括其自己的处理器和存储器。所述编码器/解码器模块5030表示可包括在设备中以执行编码和/或解码功能的模块(一个或多个)。如已知的,设备可以包括所述编码模块和解码模块中的一个或两个。另外,编码器/解码器模块5030可实施为系统5000的单独元件或可并入处理器5010内作为如所属领域的技术人员已知的硬件与软件的组合。

[0114] 要加载到处理器5010或编码器/解码器5030上以执行本文档中描述的各个方面的程序代码可以存储在存储设备5040中,并且随后加载到存储器5020上以供处理器5010执行。根据各种实施例,处理器5010、存储器5020、存储设备5040和编码器/解码器模块5030中的一者或多者可以在执行本文中描述的过程期间存储各种项中的一者或多者。这些存储的项可以包括但不限于输入视频、所解码的视频或该解码的视频的部分、比特流、矩阵、变量以及来自方程式、公式、运算和运算逻辑的处理的中间或最终结果。

[0115] 在若干实施例中,所述处理器5010和/或所述编码器/解码器模块5030内的存储器用于存储指令,并且提供用于在编码或解码期间需要的处理的工作存储器。

[0116] 然而,在其它实施例中,所述处理设备(例如,所述处理设备可为所述处理器5010或所述编码器/解码器模块5030)外部的存储器用于这些功能中的一者或多者。外部存储器可以是存储器5020和/或存储设备5040,例如,动态易失性存储器和/或非易失性闪存。在若干实施例中,外部非易失性闪存用于存储电视的操作系统。在至少一个实施例中,诸如RAM

的快速外部动态易失性存储器被用作视频编码和解码操作的工作存储器,诸如用于MPEG-2、HEVC、VVC(通用视频编码)或TMC2的工作存储器。

[0117] 如框5130中所示,可以通过各种输入设备来提供对系统5000的元件的输入。这样的输入设备包括但不限于:(i)接收例如由广播者通过空中传输的RF信号的RF部分,(ii)复合输入端子,(iii)USB输入端子,和/或(iv)HDMI输入端子。

[0118] 在各种实施例中,框5130的输入设备具有本领域已知的相关联的相应输入处理元件。例如,所述RF部分可以与用于以下所必须的元件相关联:(i)选择期望频率(也称为选择信号,或将信号频带限制到一频带),(ii)将所选择的信号下变频,(iii)再次将频带限制到较窄频带,以选择(例如,)在某些实施例中可以称为信道的信号频带,(iv)解调所述下变频且频带限制的信号,(v)执行纠错,和(vi)解复用以选择期望的数据分组流。各种实施例的RF部分包括一个或多个元件以执行这些功能,例如,频率选择器、信号选择器、限带器、信道选择器、滤波器、下变频器、解调器、纠错器和解复用器。所述RF部分可以包括执行各种这些功能的调谐器,这些功能包括例如将所接收的信号下变频到较低频率(例如,中频或近基带频率)或基带。

[0119] 在一个机顶盒实施例中,所述RF部分及其相关的输入处理元件接收通过有线(例如,电缆)介质发送的RF信号,并通过滤波、下变频和再次滤波来执行到期望频带的频率选择。

[0120] 各种实施例重新安排上述(和其它)元件的顺序,移除这些元件中的一些,和/或添加执行类似或不同功能的其它元件。

[0121] 添加元件可以包括在现有元件之间插入元件,例如插入放大器和模数转换器。在各种实施例中,所述RF部分包括天线。

[0122] 另外,USB和/或HDMI终端可以包括用于通过USB和/或HDMI连接将系统5000连接到其它电子设备的相应接口处理器。应当理解,输入处理的各个方面(例如,所罗门纠错)可以根据需要在例如单独的输入处理IC或处理器5010内实现。类似地,USB或HDMI接口处理的各方面可以根据需要在单独的接口IC内或在处理器5010内实现。解调、纠错和解复用的流被提供给各种处理元件,这其中包括例如处理器5010和编码器/解码器5030,其与存储器和存储元件结合操作以根据需要来处理所述数据流以便在输出设备上呈现。

[0123] 系统5000的各种元件可以设置在集成壳体内。在该集成壳体内,各种元件可以使用合适的连接布置(例如,本领域已知的内部总线,包括I2C总线、布线和印刷电路板)互连并在其间传输数据。

[0124] 所述系统5000包括通信接口5050,其使得能够经由通信信道5060与其他设备通信。所述通信接口5050可以包括但不限于被配置为通过通信信道5060发送和接收数据的收发器。所述通信接口5050可以包括但不限于调制解调器或网卡,并且所述通信信道5060可以例如在有线和/或无线介质内实现。

[0125] 在各种实施例中,使用Wi-Fi网络(例如IEEE 802.11),将数据流式传输给所述系统5000。这些实施例的Wi-Fi信号通过适用于Wi-Fi通信的通信信道5060和通信接口5050来接收。这些实施例的通信信道5060通常连接到接入点或路由器,所述接入点或路由器提供对包括因特网的外部网络的接入以允许流式传输应用和其它云上通信。

[0126] 其它实施例使用机顶盒向系统5000提供流式传输数据,该机顶盒通过输入框5130

的HDMI连接来传递所述数据。

[0127] 还有一些实施例使用输入框5130的RF连接来向系统5000提供流式传输数据。

[0128] 所述流式传输数据可用作由系统5000使用的信令信息的方式。该信令信息可以包括如上所述的信息数据ID。

[0129] 应当理解,可以以各种方式来实现信令。例如,在各种实施例中,一个或多个语法元素、标志等被用于将信息用信号通知给对应的解码器。

[0130] 所述系统5000可以向各种输出设备(包括显示器5100、扬声器5110和其他外围设备5120)提供输出信号。在各实施例的各示例中,所述其它外围设备5120包括以下中的一者或多者:独立DVR、盘播放器、立体声系统、照明系统和基于系统5000的输出来提供功能的其他设备。

[0131] 在各种实施例中,使用信令(诸如,AV.Link(AV.链路)、CEC、或在有或没有用户干预的情况下实现设备到设备控制的其他通信协议)在系统5000和显示器5100、扬声器5110或其它外围设备5120之间传送控制信号。

[0132] 所述输出设备可以经由通过相应接口5070、5080和5090的专用连接而通信地耦合到系统5000。

[0133] 作为替代,所述输出设备可以使用通信信道5060经由通信接口5050连接到系统5000。所述显示器5100和扬声器5110可以与系统5000的其它组件一起集成在电子设备(例如,电视机)中的单个单元中。

[0134] 在各种实施例中,显示接口5070包括显示驱动器,例如定时控制器((T Con)芯片。

[0135] 例如,如果输入框5130的RF部分是单独机顶盒的一部分,则所述显示器5100和扬声器5110可以备选地与其它组件中的一个或多个分离。在所述显示器5100和扬声器5110是外部组件的各种实施例中,所述输出信号可以经由专用输出连接来提供,所述专用输出连接例如包括HDMI端口、USB端口或COMP输出。

[0136] 本文所述的各种过程和特征的实施方式可以体现在各种不同的设备或应用中。这种设备的示例包括编码器、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、以及用于处理图片或视频的任何其他设备或其他通信设备。应当清楚,该设备可以是移动的,甚至可以安装在移动车辆中。

[0137] 另外,所述方法可由正被处理器执行的指令实施,且此类指令(和/或由实施方案产生的数据值)可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,该计算机可读程序产品被嵌入在一个或多个计算机可读介质中,并且其上嵌入有计算机可执行的计算机可读程序代码。如本文所使用的计算机可读存储介质被认为是非暂时性存储介质,其被赋予在其中存储信息的固有能力以及提供从其检索信息的固有能力。计算机可读存储介质可以是例如但不限于:电、磁、光、电磁、红外或半导体系统、装置或设备,或前述的任何合适的组合。应当理解,尽管提供了可以应用本实施例的计算机可读存储介质的更具体的示例,但是如本领域普通技术人员容易理解的,以下仅是说明性的而非穷举性的列举:便携式计算机磁盘;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);光学存储设备;磁存储装置;或前述的任何合适的组合。

[0138] 所述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。

[0139] 指令可以是例如硬件、固件、软件或其组合。指令可以在例如操作系统、单独的应用程序或两者的组合中找到。因此,处理器可以被表征为例如被配置为执行处理的设备和包括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)的设备这两者。此外,处理器可读介质可以存储由实现方式产生的数据值,作为指令的补充或替代。

[0140] 如对于本领域技术人员将显而易见的,实现方式可以产生被格式化以携带例如可以被存储或发送的信息的各种信号。该信息可以包括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将用于写入或读取本实施例的所述示例的语法的规则作为数据来携带,或者将由本实施例的所述示例写入的实际语法值作为数据来携带。这种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。所述格式化可以包括例如对数据流进行编码并且利用所编码的数据流对载波进行调制。所述信号携带的信息可以是例如模拟或数字信息。如已知的,所述信号可以通过各种不同的有线或无线链路来传输。所述信号可以存储在处理器可读介质上。

[0141] 已经描述了许多实现方式。然而,应当理解,可以进行各种修改。例如,不同实现方式的元素可以被组合、补充、修改或移除以产生其他实现方式。另外,本领域技术人员将理解,其他结构和过程可以替代所公开的那些,并且所得到的实现方式将以至少基本上相同的方式(一个或多个)执行至少基本上相同的功能(一个或多个),以实现与所公开的实现方式至少基本上相同的结果(一个或多个)。因此,本申请考虑了这些和其他实现方式。

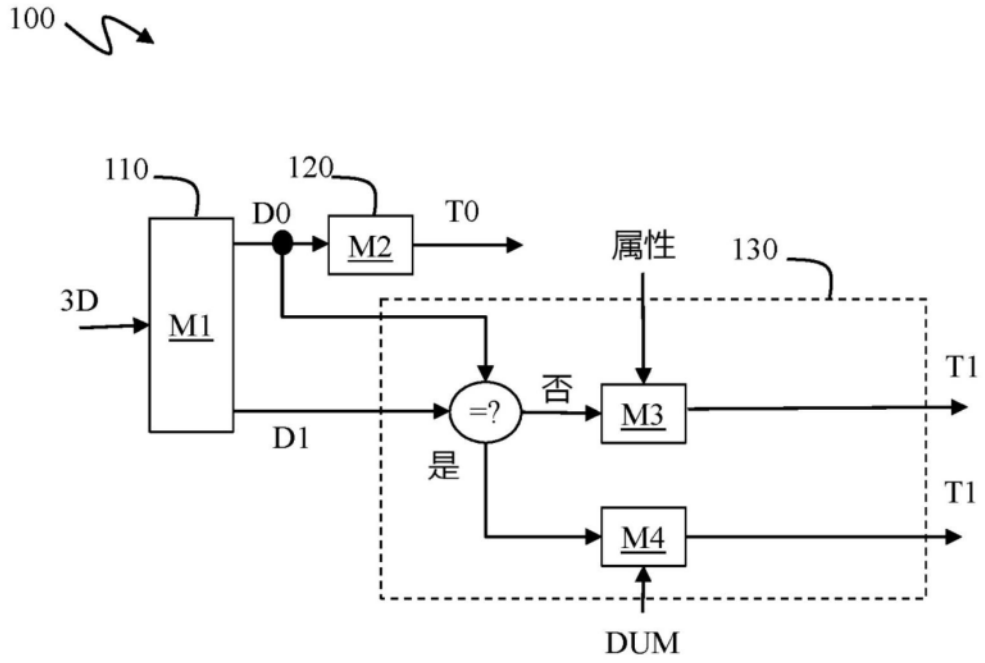

100
110
120
D0
T0
属性
130
3D
M1
M2
=?
否
M3
T1
D1
是
T1
M4
DUM

图1

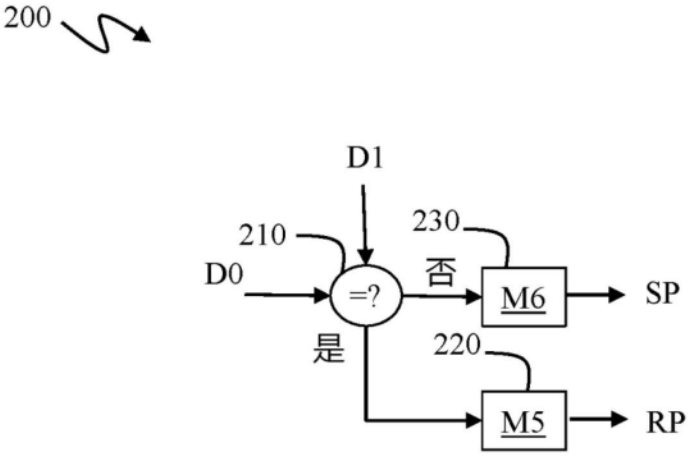

200
D1
210
230
D0
=?
否
M6
SP
是
220
M5
RP

图2

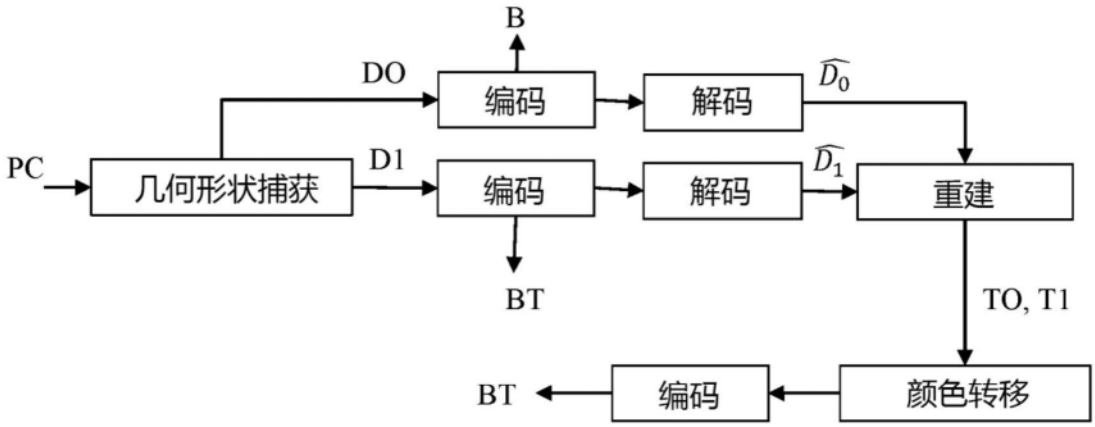

B
DO
编码
解码
$\widehat{D_0}$
PC
几何形状捕获
D1
编码
解码
$\widehat{D_1}$
重建
BT
TO, T1
BT
编码
颜色转移

图3

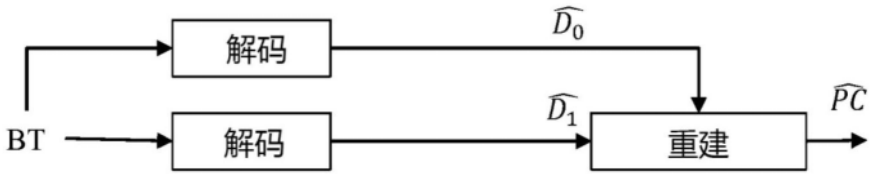

解码
$\widehat{D_0}$
BT
解码
$\widehat{D_1}$
重建
$\widehat{PC}$

图4

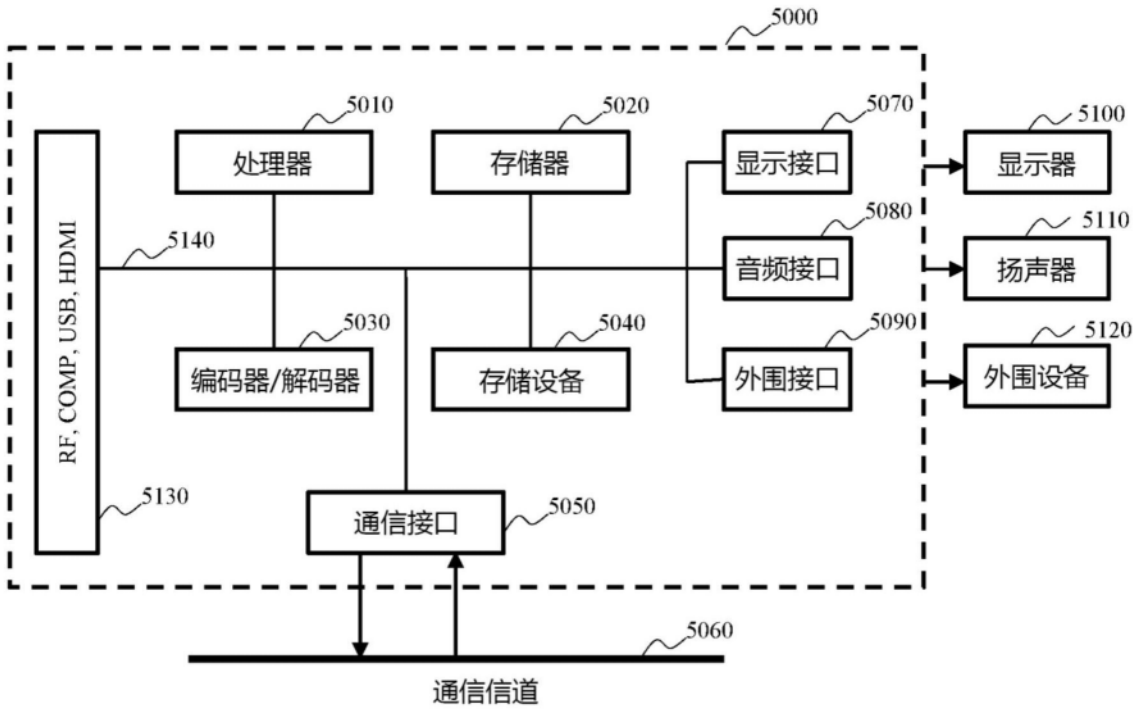

5000
5010
处理器
5020
存储器
5070
显示接口
5100
显示器
RF, COMP, USB, HDMI
5140
5080
音频接口
5110
扬声器
5030
编码器/解码器
5040
存储设备
5090
外围接口
5120
外围设备
5130
通信接口
5050
5060
通信信道

图5

group_of_frames_header( ) {	描述符
group_of_frames_size	u(8)
frame_width	u(16)
frame_height	u(16)
occupancy_resolution	u(8)
radius_to_smoothing	u(8)
neighbor_count_smoothing	u(8)
radius2_boundary_detection	u(8)
threshold_smoothing	u(8)
lossless_geometry	u(8)
lossless_texture	u(8)
no_attributes	u(8)
lossless_geometry_444	u(8)
absolute_d1_coding	u(8)
binary_arithmetic_coding	u(8)
auxillary_information_delta_coding	u(8)
remove_duplicate_coding_group_of_frames	u(8)
ByteCount += 17	
}	

图6

frame_auxiliary_information(frame_index) {	描述符
if((frame_index==0)||(!auxillary_information_delta_coding)||(lossless_...)) {	
patch_count	u(32)
occupancy_precision	u(8)
max_candidate_Count	u(8)
bit_count_u0	u(8)
bit_count_v0	u(8)
bit_count_u1	u(8)
bit_count_v1	u(8)
bit_count_d1	u(8)
If(remove_duplicate_coding_group_of_frames) {	
remove_duplicate_coding_frame	u(8)
}	
occupancy_aux_stream_size	u(32)
ByteCount += 15	
for(i = 0; i < patchCount; i++) {	
patchList[i].patch_u0	ae(v)
patchList[i].patch_v0	ae(v)
patchList[i].patch_u1	ae(v)
patchList[i].patch_v1	ae(v)
patchList[i].patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)
patchList[i].normal_axis	ae(v)
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
} else {	
patch_count	u(32)
occupancy_precision	u(8)
max_candidate_Count	u(8)

图7

remove_duplicate_coding_frame	u(8)
occupancy_aux_stream_size	u(32)
ByteCount += 15	
maximum_from_umatched_patch	u(1)
if(maximum_from_umatched_patch) {	
maximum_item_indication	u(5)
for(i = 0; i < 5; i++) {	
if(maximum_item_indication[i]) {	
bit_count_unmatched_patch_item[i]	u(8)
}	
}	
}	
matched_patch_count	ae(v)
for(i = 0; i < matched_patch_count; i++) {	
patchList[i].delta_patch_index	
patchList[i].delta_patch_u0	ae(v)
patchList[i].delta_patch_v0	ae(v)
patchList[i].delta_patch_u1	ae(v)
patchList[i].delta_patch_v1	ae(v)
patchList[i].delta_patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
for(i = matched_patch_count; i < patchCount; i++) {	
patchList[i].patch_u0	ae(v)
patchList[i].patch_v0	ae(v)
patchList[i].patch_u1	ae(v)
patchList[i].patch_v1	ae(v)
patchList[i].patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)

图7a

patchList[i].normal_axis	ae(v)
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
}	
for(i = 0; i < blockCount; i++) {	
if(candidatePatches[i].size( ) == 1)	
blockToPatch[i] = candidatePatches[i][0]	
else {	
candidate_index	ae(v)
if(candidate_index == max_candidate_count)	
blockToPatch[i] = patch_index	ae(v)
else	
blockToPatch[i] = candidatePatches[i][candidate_index]	
}	
}	
ByteCount += occupancy_auxilary_stream_size	
}	

图7b

frame_auxiliary_information(frame_index) {	描述符
if((frame_index==0)||(!auxillary_information_delta_coding)||(lossless ...)) {	
patch_count	u(32)
occupancy_precision	u(8)
max_candidate_Count	u(8)
bit_count_u0	u(8)
bit_count_v0	u(8)
bit_count_u1	u(8)
bit_count_v1	u(8)
bit_count_d1	u(8)
If(remove_duplicate_coding_group_of_frames) {	
remove_duplicate_coding_frame	u(8)
}	
occupancy_aux_stream_size	u(32)
ByteCount += 15	
for(i = 0; i < patchCount; i++) {	
patchList[i].patch_u0	ae(v)
patchList[i].patch_v0	ae(v)
patchList[i].patch_u1	ae(v)
patchList[i].patch_v1	ae(v)
patchList[i].patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)
patchList[i].normal_axis	ae(v)
If(remove_duplicate_coding_frame) {	
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
}	
} else {	

图8

patch_count	u(32)
occupancy_precision	u(8)
max_candidate_Count	u(8)
If(remove_duplicate_coding_group_of_frames) {	
remove_duplicate_coding_frame	u(8)
}	
occupancy_aux_stream_size	u(32)
ByteCount += 15	
maximum_from_umatched_patch	u(1)
if(maximum_from_umatched_patch) {	
maximum_item_indication	u(5)
for(i = 0; i < 5; i++) {	
if(maximum_item_indication[i]) {	
bit_count_unmatched_patch_item[i]	u(8)
}	
}	
}	
matched_patch_count	ae(v)
for(i = 0; i < matched_patch_count; i++) {	
patchList[i].delta_patch_index	
patchList[i].delta_patch_u0	ae(v)
patchList[i].delta_patch_v0	ae(v)
patchList[i].delta_patch_u1	ae(v)
patchList[i].delta_patch_v1	ae(v)
patchList[i].delta_patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)
If(remove_duplicate_coding_frame) {	
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
}	

图8a

for(i = matched_patch_count; i < patchCount; i++) {	
patchList[i].patch_u0	ae(v)
patchList[i].patch_v0	ae(v)
patchList[i].patch_u1	ae(v)
patchList[i].patch_v1	ae(v)
patchList[i].patch_d1	ae(v)
patchList[i].delta_size_u0	se(v)
patchList[i].delta_size_v0	se(v)
patchList[i].normal_axis	ae(v)
if(remove_duplicate_coding_frame) {	
patchList[i].remove_duplicate_coding_patch	ae(v)
}	
}	
}	
for(i = 0; i < blockCount; i++) {	
if(candidatePatches[i].size( ) == 1)	
blockToPatch[i] = candidatePatches[i][0]	
else {	
candidate_index	ae(v)
if(candidate_index == max_candidate_count)	
blockToPatch[i] = patch_index	ae(v)
else	
blockToPatch[i] = candidatePatches[i][candidate_index]	
}	
}	
ByteCount += occupancy_auxilary_stream_size	
}	

图8b