



(21) 申请号 201780060054.9
(22) 申请日 2017.09.28
(65) 同一申请的已公布的文献号
申请公布号 CN 109792520 A
(43) 申请公布日 2019.05.21
(30) 优先权数据
16306271.4 2016.09.30 EP
(85) PCT国际申请进入国家阶段日
2019.03.28
(86) PCT国际申请的申请数据
PCT/EP2017/074624 2017.09.28
(87) PCT国际申请的公布数据
W02018/060329 EN 2018.04.05
(73) 专利权人 交互数字麦迪逊专利控股公司
地址 法国巴黎
(72) 发明人 F.加尔平 F.拉卡普 T.维拉德
(74) 专利代理机构 北京市柳沈律师事务所
11105
专利代理师 于小宁

(51) Int.Cl.
H04N 19/11 (2006.01)
H04N 19/597 (2006.01)
H04N 19/105 (2006.01)
H04N 19/176 (2006.01)
H04N 19/70 (2006.01)
H04N 19/593 (2006.01)
H04N 19/157 (2006.01)
(56) 对比文件
WO 2014036162 A1,2014.03.06
CN 103947205 A,2014.07.23
CN 103597832 A,2014.02.19
US 2007047648 A1,2007.03.01
Elie-Gabriel Mora.Multiview video
plus depth coding for new multimedia
services.《https://tel.archives-
ouvertes.fr/tel-01061005v2》.2016,
Joel Jung.Depth video coding based on
intra mode inheritance from texture.
《Article in APSIPA Transactions on Signal
and Information Processing》.2014,
审查员 谢佳妮

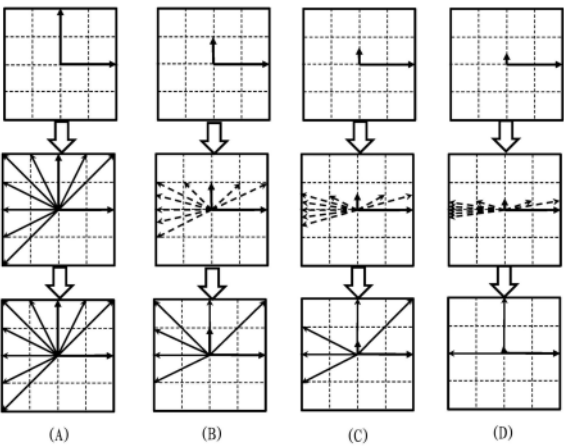
权利要求书2页 说明书26页 附图36页

(54) 发明名称

用于利用自适应帧内最可能模式的全向视频译码的方法和装置

(57) 摘要

在全向视频中,对于帧内预测,只有一些帧内预测方向可能是相关的。例如,对于从等矩形映射生成的视频,只有水平方向可能与极点周围的帧内预测相关。为了提高压缩效率,我们提出基于要编码或解码的块在帧中的定位来减少可能的方向的数量。在各种实施例中,我们可以使MPM的推导适配以便考虑取决于几何失真的帧内方向的减少。编码方法也可以被适配为减少针对全向视频中的块要估计的模式的数量。也可以改进表示帧内方向的语法元素的译码。



1. 一种对全向视频数据进行编码或解码的方法,包括:

访问等矩形图片的第一块;

从用于图片的第一块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量取决于所述第一块的垂直位置,以及其中当所述第一块比第二块在垂直方向上更远离所述图片的中心时,用于所述第一块的可用帧内预测模式的数量小于用于所述第二块的可用帧内预测模式的数量;以及

使用所确定的帧内预测模式对所述第一块进行编码或解码。

2. 如权利要求1所述的方法,其中使用固定长度代码对用于所述第一块的帧内预测模式进行编码,所述固定长度代码的长度基于所述集合中的可用帧内预测模式的数量,并且其中所述固定长度代码的长度取决于所述第一块的垂直位置。

3. 如权利要求1所述的方法,其中所述可用帧内预测模式中的两个邻近帧内预测模式之间的角度差基于所述第一块的定位。

4. 如权利要求1所述的方法,还包括:

基于相邻块的帧内预测模式来确定用于所述第一块的第一最可能模式;以及

基于所述第一最可能模式来确定用于所述第一块的第二最可能模式,其中所述第一最可能模式与所述第二最可能模式之间的角度差基于所述第一块的定位。

5. 如权利要求4所述的方法,其中用于所述第一块的帧内预测模式被选择为所述第一最可能模式和所述第二最可能模式之一。

6. 如权利要求1所述的方法,所述图片具有第一块和第二块,其中用于所述第一块的可用帧内预测模式的第一集合与用于所述第二块的可用帧内预测模式的第二集合不同,并且其中可用帧内预测模式的所述第一集合中的两个邻近帧内预测模式之间的第一角度差与可用帧内预测模式的所述第二集合中的两个邻近帧内预测模式之间的第二角度差相同。

7. 如权利要求1所述的方法,所述图片具有第一块和第二块,其中用于所述第一块的可用帧内预测模式的第一集合与用于所述第二块的可用帧内预测模式的第二集合不同,其中用于所述第一块的可用帧内预测模式的所述第一集合通过相连帧内预测模式索引的第一集合来表示,其中用于所述第二块的可用帧内预测模式的所述第二集合通过相连帧内预测模式索引的第二集合来表示。

8. 如权利要求7所述的方法,其中可用帧内预测模式的所述第一集合和可用帧内预测模式的所述第二集合中的相同帧内预测模式索引对应于相同帧内预测模式。

9. 如权利要求1所述的方法,其中使用查找表来将用于所述第一块的所述集合中的可用帧内预测模式的数量与所述第一块的定位相关联。

10. 一种用于对视频数据进行编码或解码的装置,包括:一个或多个处理器,其中所述一个或多个处理器被配置为:

从用于第一块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量取决于所述第一块的垂直位置,以及其中当所述第一块比第二块在垂直方向上更远离图片的中心时,用于所述第一块的可用帧内预测模式的数量小于用于所述第二块的可用帧内预测模式的数量;以及

使用所确定的帧内预测模式对所述第一块进行编码或解码。

11. 如权利要求10所述的装置,其中使用固定长度代码对用于所述第一块的帧内预测

模式进行编码,所述固定长度代码的长度基于所述集合中的可用帧内预测模式的数量,并且其中所述固定长度代码的长度取决于所述第一块的垂直位置。

12.如权利要求10所述的装置,其中所述可用帧内预测模式中的两个邻近帧内预测模式之间的角度差基于所述第一块的定位。

13.如权利要求10所述的装置,其中所述一个或多个处理器还被配置为:

基于相邻块的帧内预测模式来确定用于所述第一块的第一最可能模式;以及

基于所述第一最可能模式来确定用于所述第一块的第二最可能模式,其中所述第一最可能模式与所述第二最可能模式之间的角度差基于所述第一块的定位。

14.如权利要求13所述的装置,其中用于所述第一块的帧内预测模式被选择为所述第一最可能模式和所述第二最可能模式之一。

15.如权利要求10所述的装置,所述图片具有第一块和第二块,其中用于所述第一块的可用帧内预测模式的第一集合与用于所述第二块的可用帧内预测模式的第二集合不同,并且其中可用帧内预测模式的所述第一集合中的两个邻近帧内预测模式之间的第一角度差与可用帧内预测模式的所述第二集合中的两个邻近帧内预测模式之间的第二角度差相同。

16.如权利要求10所述的装置,所述图片具有第一块和第二块,其中用于所述第一块的可用帧内预测模式的第一集合与用于所述第二块的可用帧内预测模式的第二集合不同,其中用于所述第一块的可用帧内预测模式的所述第一集合通过相连帧内预测模式索引的第一集合来表示,其中用于所述第二块的可用帧内预测模式的所述第二集合通过相连帧内预测模式索引的第二集合来表示。

17.如权利要求16所述的装置,其中可用帧内预测模式的所述第一集合和可用帧内预测模式的所述第二集合中的相同帧内预测模式索引对应于相同帧内预测模式。

18.如权利要求10所述的装置,其中使用查找表来将用于所述第一块的所述集合中的可用帧内预测模式的数量与所述第一块的定位相关联。

用于利用自适应帧内最可能模式的全向视频译码的方法和装置

技术领域

[0001] 本实施例一般地涉及用于视频编码和解码的方法和装置,并且更具体地涉及用于利用自适应帧内可能模式的视频编码和解码的方法和装置。

背景技术

[0002] 最近,可用的大视场内容(高达360°)有所增长。这样的内容可能不会被观看诸如头戴式显示器(HMD)、智能眼镜、PC屏幕、平板、智能电话等的沉浸式显示设备上的内容的用户完全看得见。这意味着在给定时刻,用户可能只是查看内容的一部分。然而,用户典型地可以通过诸如头部移动、鼠标移动、触摸屏、语音等的各种手段在内容内导航。典型地希望对该内容进行编码和解码。

发明内容

[0003] 根据本原理的一般方面,一种对视频数据进行编码的方法,包括:访问图片的块;从用于所述块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量基于所述块的定位;以及使用所确定的帧内预测模式对所述块进行编码。

[0004] 根据本原理的另一方面,一种对视频数据进行解码的方法,包括:从用于图片的块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量基于所述块的定位;以及使用所确定的帧内预测模式对所述块进行解码。

[0005] 根据本原理的另一方面,一种用于对视频数据进行编码的装置,包括至少一个存储器和一个或多个处理器,其中所述一个或多个处理器被配置为:访问图片的块;从用于所述块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量基于所述块的定位;以及使用所确定的帧内预测模式对所述块进行编码。

[0006] 根据本原理的另一方面,一种用于对视频数据进行解码的装置,包括至少一个存储器和一个或多个处理器,其中所述一个或多个处理器被配置为:从用于图片的块的可用帧内预测模式的集合中确定帧内预测模式,其中所述集合中的可用帧内预测模式的数量基于所述块的定位;以及使用所确定的帧内预测模式对所述块进行解码。

[0007] 根据本原理的另一方面,一种比特流,其被格式化为包括:代表图片的块的译码数据;代表来自用于所述块的可用帧内预测模式的集合的帧内预测模式的译码数据,其中所述集合中的可用帧内预测模式的数量基于所述块的定位,并且其中使用所述帧内预测模式对所述块进行编码。

[0008] 集合中的可用帧内预测模式的数量可以取决于块的垂直位置。在一个实施例中,当块比第二块在垂直方向上更远离图片的中心时,用于块的可用帧内预测模式的数量小于用于第二块的可用帧内预测模式的数量。

[0009] 可用帧内预测模式中的两个邻近帧内预测模式之间的角度差可以基于块的定位。

[0010] 根据实施例,可以基于相邻块的帧内预测模式来导出用于块的第一最可能模式。然后,可以基于第一最可能模式来导出用于块的第二最可能模式,其中第一最可能模式与第二最可能模式之间的角度差基于块的定位。可以将用于块的帧内预测模式选择为第一最可能模式和第二可能模式之一。

[0011] 如果块的帧内预测模式是最可能模式之一,则针对块对MPM索引进行编码。否则,可以使用固定长度代码对块的帧内预测模式进行编码,其中固定长度代码的长度基于集合中的可用帧内预测模式的数量。当集合中的可用帧内预测模式的数量基于块的垂直位置时,固定长度代码的长度取决于块的垂直位置。可以将块的垂直位置判定为块上的点(例如块的中心或拐角点)的位置。

[0012] 对于图片中的第一块和第二块,其中用于第一块的可用帧内预测模式的第一集合与用于第二块的可用帧内预测模式的第二集合不同,可用帧内预测模式的第一集合中的两个邻近帧内预测模式之间的第一角度差可以与可用帧内预测模式的第二集合中的两个邻近帧内预测模式之间的第二角度差相同。

[0013] 对于图片中的第一块和第二块,其中用于第一块的可用帧内预测模式的第一集合与可用帧内预测模式的第二集合不同,用于第一块的可用帧内预测模式的第一集合可以通过相连帧内预测模式索引的第一集合来表示,并且用于第二块的可用帧内预测模式的第二集合也可以通过相连帧内预测模式索引的第二集合来表示。在一个实施例中,可用帧内预测模式的第一集合和可用帧内预测模式的第二集合中的相同帧内预测模式索引对应于相同帧内预测模式。

[0014] 可以使用查找表来将用于块的集合中的可用帧内预测模式的数量与块的定位相关联。使用查找表可以简化并加速以上处理。

[0015] 可以在比特流中指示用于块的集合中的可用帧内预测模式的数量。因此,解码器可以直接从比特流得到用于块的集合中的可用帧内预测模式的数量,而无需在解码器侧计算该数量。

[0016] 本实施例还提供了一种用于根据上述方法对视频数据进行编码或解码的装置。本实施例还提供了一种用于传送根据上述方法生成的比特流的装置。

[0017] 本实施例还提供了一种计算机可读存储介质,其上存储有用于根据上述方法对视频数据进行编码或解码的指令。本实施例还提供了一种计算机可读存储介质,其上存储有根据上述方法生成的比特流。

附图说明

[0018] 图1图示了根据本原理的实施例的用于编码和解码全向视频的示例性系统。

[0019] 图2A示出了使用等矩形投影将示例性球体表面S映射到2D帧F,以及图2B示出了使用立方体映射将示例性立方体表面映射到如图2C中所示的2D帧。

[0020] 图3A、图3B、图3C和图3D图示了示例性逆等矩形映射。

[0021] 图4A示出了将点从3D表面投影到呈现帧G,以及图4B示出了投影的近似。

[0022] 图5图示了示例性HEVC视频编码器的框图。

[0023] 图6A示出了在HEVC中可用的35个帧内预测模式的集合,以及图6B示出了沿着由角度预测模式指示的方向来复制参考样本。

- [0024] 图7图示了示例性HEVC视频解码器的框图。
- [0025] 图8图示了其中呈现帧中的直线在要编码的帧中不被视为直线的示例。
- [0026] 图9图示了根据本原理的示例性实施例的使MPM列表适配以用于对帧内预测模式进行编码或解码的示例性方法。
- [0027] 图10A示出了x-y坐标系中的归一化2D帧,以及图10B示出了用于等矩形映射的示例性变形索引(warping index)推导。
- [0028] 图11A图示了根据本原理的实施例的用于计算转换“T”的示例性工作流程,以及图11B图示了根据本原理的实施例的用于计算转换“T”的示例性工作流程。
- [0029] 图12图示了根据本原理的实施例的在使用等矩形映射时基于相邻左块获得MPM的示例性处理。
- [0030] 图13图示了赤道处的示例性当前块C,其相邻左块L也在赤道处。
- [0031] 图14A示出了立方体映射的示例,以及图14B示出了边缘处的方向改变的示例。
- [0032] 图15示出了全向图片的示例性等矩形布局以及局部参考帧沿垂直坐标的演变。
- [0033] 图16A、图16B、图16C、图16D和图16E图示了在等矩形视频的情况下来自沿Y轴的不同定位的示例性方向形变。
- [0034] 图17示出了JVET(联合视频探索小组)的联合探索软件(JEM)中的帧内方向模式。
- [0035] 图18示出了根据本原理的示例性实施例的水平线上方和下方的可用帧内预测模式的数量。
- [0036] 图19A、图19B、图19C和图19D提供了根据本原理的示例性实施例的针对 $a=1$ 、 0.5 、 0.25 和 0.125 计算可用帧内预测模式的示例。
- [0037] 图20示出了根据本原理的示例性实施例的使方向分组和帧内方向的重新排序的示例。
- [0038] 图21A图示了根据本原理的实施例的使用MPM列表对用于当前块的帧内预测模式进行编码或解码的示例性方法,以及图21B图示了根据本原理的实施例的使用固定长度代码对用于当前块的帧内预测模式进行编码或解码的示例性方法。
- [0039] 图22图示了在严重失真的情况下直的帧内模式如何在示例性帧上变化。
- [0040] 图23图示了映射在矩形帧上的示例性非正方形立方体。
- [0041] 图24图示了其中可以实现本原理的示例性实施例的各个方面的示例性系统的框图。
- [0042] 图25表示根据本原理的特定实施例的系统的第二实施例。
- [0043] 图26表示根据本原理的特定实施例的系统的第二实施例。
- [0044] 图27表示根据本原理的特定实施例的系统的第二实施例。
- [0045] 图28表示根据本原理的特定实施例的系统的第二实施例。
- [0046] 图29表示根据本原理的特定实施例的系统的第二实施例。
- [0047] 图30表示根据本原理的特定实施例的系统的第二实施例。
- [0048] 图31表示根据本原理的系统的第二实施例。
- [0049] 图32表示根据本原理的系统的第二实施例。
- [0050] 图33表示根据本原理的沉浸式视频呈现设备的第一实施例。
- [0051] 图34表示根据本原理的沉浸式视频呈现设备的第一实施例。

[0052] 图35表示根据本原理的沉浸式视频呈现设备的第一实施例。

具体实施方式

[0053] 大视场内容可以是三维计算机图形影像场景(3D CGI场景)、点云或沉浸式视频,等等。许多术语可以用来设计这样的沉浸式视频,诸如例如虚拟现实(VR)、360、全景、 4π 、球面度、沉浸式、全向、大视场。

[0054] 沉浸式视频典型地涉及在作为二维像素阵列(即,色彩信息的元素)的矩形帧上编码的视频,如“常规”视频。在许多实现方式中,可以执行以下处理。为了呈现,首先将帧映射在凸体积(volume)的内表面上,也称为映射表面(例如,球体、立方体、金字塔),其次,该体积的一部分由虚拟相机捕获。由虚拟相机捕获的图像被呈现在沉浸式显示设备的屏幕上。将立体视频编码在一个或两个矩形帧上,投影在两个映射表面上,这两个映射表面被组合以根据设备的特性由两个虚拟相机捕获。

[0055] 在帧中,可以根据映射函数对像素进行编码。映射函数可以取决于映射表面。对于相同的映射表面,若干个映射函数是可能的。例如,可以根据帧表面内的不同布局来构造立方体的面。例如,可以根据等矩形投影或球心投影来映射球体。产生自所选投影函数的像素的组织使线连续性、规格化正交(orthonormal)局部帧、像素密度修改或断开,并引入时间和空间上的周期性。这些是用于对视频进行编码和解码的典型特征。现有的编码和解码方法通常不考虑沉浸式视频的特异性。实际上,随着沉浸式视频可以是 360° 视频,例如摇拍引入运动和不连续性,这需要在场景的内容不改变的同时编码大量数据。在对视频帧进行编码和解码时考虑沉浸式视频特异性将为编码或解码方法带来有价值的优势。

[0056] 图1图示了根据示例实施例的编码和解码系统的总体概述。图1的系统是功能系统。预处理模块110可以准备用于由编码设备120进行编码的内容。预处理模块110可以执行多图像采集,在公共空间(如果我们对方向进行编码,则典型地为3D球体)中合并所采集的多个图像,以及使用例如但不限于等矩形映射或立方体映射将3D球体映射到2D帧中。预处理模块110也可以接受特定格式(例如,等矩形)的全向视频作为输入,并对视频进行预处理,以将映射改变成更适合于编码的格式。取决于所采集的视频数据表示,预处理模块110可以执行映射空间改变。

[0057] 将关于说明书的其他附图描述编码设备120和编码方法。在编码之后,数据(例如其可以对沉浸式视频数据或3D CGI编码数据进行编码)被发送到网络接口130,网络接口130典型地可以以任何网络接口来实现,例如存在于网关中。然后,数据通过通信网络(诸如互联网,但是可以预见任何其他网络)被传送。然后,数据经由网络接口140被接收。网络接口140可以实现在网关中、电视中、机顶盒中、头戴式显示设备中、沉浸式(投影式)墙壁中或者任何沉浸式视频呈现设备中。

[0058] 在接收之后,数据被发送到解码设备150。解码功能是在下面的图2至图12中描述的处理功能之一。然后,解码数据由播放器160处理。播放器160准备用于呈现设备170的数据,并且可以接收来自传感器的外部数据或用户输入数据。更准确地,播放器160准备将由呈现设备170显示的视频内容的部分。解码设备150和播放器160可以集成在单个设备(例如,智能电话、游戏控制台、STB、平板、计算机等)中。在其他实施例中,播放器160可以集成在呈现设备170中。

[0059] 例如在呈现沉浸式视频时,可以设想若干类型的系统来执行沉浸式显示设备的解码、播放和呈现功能。

[0060] 图2至图6中图示了用于处理增强现实、虚拟现实或增强的虚拟内容的第一系统。这样的系统包括处理功能、沉浸式视频呈现设备,沉浸式视频呈现设备可以是例如头戴式显示器(HMD)、平板或智能电话,并且可以包括传感器。沉浸式视频呈现设备也可以包括显示设备与处理功能之间的附加接口模块。处理功能可以由一个或若干个设备执行。它们可以集成到沉浸式视频呈现设备中,或者它们可以集成到一个或若干个处理设备中。处理设备包括一个或若干个处理器,以及与沉浸式视频呈现设备的通信接口,诸如无线或有线通信接口。

[0061] 处理设备也可以包括与诸如互联网之类的宽接入网络的第二通信接口,并且直接地或者通过诸如家庭或本地网关之类的网络设备访问位于云上的内容。处理设备也可以通过第三接口(诸如以太网类型的本地接入网接口)访问本地存储。在实施例中,处理设备可以是一个或若干个处理单元的计算机系统。在另一实施例中,它可以是智能电话,其可以通过有线或无线链路连接到沉浸式视频呈现设备,或者其可以在沉浸式视频呈现设备中插入壳体中并通过连接器或者无线地与之通信。处理设备的通信接口是有线接口(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)。

[0062] 当由沉浸式视频呈现设备执行处理功能时,沉浸式视频呈现设备可以提供有直接或者通过网关接收和/或传送内容的到网络的接口。

[0063] 在另一实施例中,该系统包括与沉浸式视频呈现设备并且与处理设备通信的辅助设备。在这样的实施例中,该辅助设备可以包含至少一个处理功能。

[0064] 沉浸式视频呈现设备可以包括一个或若干个显示器。该设备可以在其每个显示器前面采用诸如透镜之类的光学器件。显示器也可以是沉浸式显示设备的一部分,如智能电话或平板的情况。在另一实施例中,显示器和光学器件可以嵌入头盔中、眼镜中或用户可以佩戴的面罩中。沉浸式视频呈现设备也可以集成若干个传感器,如稍后所述。沉浸式视频呈现设备也可以包括若干个接口或连接器。它可以包括一个或若干个无线模块,以便与传感器、处理功能、手持或其他身体部位相关设备或传感器通信。

[0065] 沉浸式视频呈现设备也可以包括由一个或若干个处理器执行并且被配置为解码内容或处理内容的处理功能。通过在此处理内容,要理解所有功能用于准备可被显示的内容。这可以包括例如解码内容、在显示内容之前合并内容以及修改内容以适合显示设备。

[0066] 沉浸式内容呈现设备的一个功能是控制虚拟相机,该虚拟相机捕获被构造为虚拟体积的内容的至少一部分。该系统可以包括姿势跟踪传感器,其完全或部分地跟踪用户的姿势,例如,用户头部的姿势,以便处理虚拟相机的姿势。一些定位传感器可以跟踪用户的位移。该系统也可以包括与环境相关的其他传感器,例如以测量照明、温度或声音条件。这样的传感器也可以与用户的身体相关,例如以测量发汗或心率。通过这些传感器采集的信息可以用于处理内容。该系统也可以包括用户输入设备(例如,鼠标、键盘、遥控器、操纵杆)。来自用户输入设备的信息可以用于处理内容、管理用户界面或控制虚拟相机的姿势。传感器和用户输入设备通过有线或无线通信接口与处理设备和/或与沉浸式呈现设备通信。

[0067] 使用图25至图29,描述用于显示增强现实、虚拟现实、增强虚拟或从增强现实到虚

拟现实的任何内容的该第一类型的系统的若干个实施例。

[0068] 图25图示了被配置为解码、处理和呈现沉浸式视频的系统的特定实施例。该系统包括沉浸式视频呈现设备10、传感器20、用户输入设备30、计算机40和网关50(可选的)。

[0069] 图33中所示的沉浸式视频呈现设备10包括显示器101。显示器例如是OLED或LCD类型。沉浸式视频呈现设备10例如是HMD、平板或智能电话。设备10可以包括触摸表面102(例如,触摸板或触觉屏幕)、相机103、与至少一个处理器104和至少一个通信接口106连接的存储器105。至少一个处理器104处理从传感器20接收的信号。

[0070] 使用来自传感器的一些测量来计算设备的姿势和控制虚拟相机。用于姿势估计的传感器例如是陀螺仪、加速度计或罗盘。也可以使用(例如使用一套相机(a rig of cameras)的)更复杂的系统。在这种情况下,至少一个处理器执行图像处理以估计设备10的姿势。使用一些其他测量来根据环境条件或用户的反应处理内容。用于观察环境和用户的传感器例如是麦克风、光传感器或接触传感器。也可以使用更复杂的系统,比如例如跟踪用户眼睛的摄像机。在这种情况下,至少一个处理器执行图像处理以操作预期测量。来自传感器20和用户输入设备30的数据也可以被传送到计算机40,计算机40将根据这些传感器的输入来处理数据。

[0071] 存储器105包括用于处理器104的参数和代码程序指令。存储器105也可以包括从传感器20和用户输入设备30接收的参数。通信接口106使得沉浸式视频呈现设备能够与计算机40通信。处理设备的通信接口106可以是有线接口(例如总线接口、广域网接口、局域网接口)或无线接口(诸如IEEE802.11接口或蓝牙®接口)。

[0072] 计算机40将数据以及可选地将控制命令发送到沉浸式视频呈现设备10。计算机40负责处理数据,即,准备它们以用于由沉浸式视频呈现设备10显示。处理可以由计算机40专门完成,或者部分处理可以由计算机完成,并且部分由沉浸式视频呈现设备10完成。计算机40直接或者通过网关或网络接口50连接到互联网。计算机40从互联网接收代表沉浸式视频的数据,处理这些数据(例如,将它们解码以及可能地准备将由沉浸式视频呈现设备10显示的视频内容的部分),并将处理后的数据发送到沉浸式视频呈现设备10以用于显示。在另一实施例中,该系统也可以包括存储代表沉浸式视频的数据的本地存储器(未示出),所述本地存储器可以在计算机40上或者在例如可通过局域网访问的本地服务器上(未示出)。

[0073] 图26表示第二实施例。在该实施例中,STB 90直接(即,STB 90包括网络接口)或者经由网关50连接到诸如互联网之类的网络。STB 90通过无线接口或者通过有线接口连接到诸如电视机100或沉浸式视频呈现设备200之类的呈现设备。除了STB的传统功能之外,STB 90包括处理视频内容以用于呈现在电视100或者任何沉浸式视频呈现设备200上的处理功能。这些处理功能与针对计算机40描述的处理功能相同,并且这里不再描述。传感器20和用户输入设备30也与较早关于图25描述的类型相同。STB 90从互联网获得代表沉浸式视频的数据。在另一实施例中,STB 90从存储代表沉浸式视频的数据的本地存储器(未示出)获得代表沉浸式视频的数据。

[0074] 图27表示与图25中所示的实施例相关的第三实施例。游戏控制台60处理内容数据。游戏控制台60将数据以及可选地将控制命令发送到沉浸式视频呈现设备10。游戏控制台60被配置为处理代表沉浸式视频的数据并将处理后的数据发送到沉浸式视频呈现设备10以用于显示。处理可以由游戏控制台60专门完成,或者部分处理可以由沉浸式视频呈现

设备10完成。

[0075] 游戏控制台60直接或者通过网关或网络接口50连接到互联网。游戏控制台60从互联网获得代表沉浸式视频的数据。在另一实施例中,游戏控制台60从存储代表沉浸式视频的数据的本地储存器(未示出)获得代表沉浸式视频的数据,所述本地储存器可以在游戏控制台60上或者在例如可通过局域网访问的本地服务器上(未示出)。

[0076] 游戏控制台60从互联网接收代表沉浸式视频的数据,处理这些数据(例如,将它们解码以及可能地准备将要显示的視頻的部分),并将处理后的数据发送到沉浸式视频呈现设备10以用于显示。游戏控制台60可以从传感器20和用户输入设备30接收数据,并且可以使用它们来处理从互联网或者从本地储存器获得的代表沉浸式视频的数据。

[0077] 图28表示所述第一类型的系统的第四实施例,其中沉浸式视频呈现设备70由插入壳体705中的智能电话701形成。智能电话701可以连接到互联网,并且因此可以从互联网获得代表沉浸式视频的数据。在另一实施例中,智能电话701从存储代表沉浸式视频的数据的本地储存器(未示出)获得代表沉浸式视频的数据,所述本地储存器可以在智能电话701上或者在例如可通过局域网访问的本地服务器上(未示出)。

[0078] 关于图34描述沉浸式视频呈现设备70,图34给出了沉浸式视频呈现设备70的优选实施例。它可选地包括至少一个网络接口702和用于智能电话701的壳体705。智能电话701包括智能电话和显示器的所有功能。智能电话的显示器用作沉浸式视频呈现设备70显示器。因此,不包括除智能电话701的显示器之外的显示器。然而,包括诸如透镜之类的光学器件704以用于在智能电话显示器上看到数据。智能电话701被配置为可能地根据从传感器20和从用户输入设备30接收的数据来处理(例如,解码以及准备用于显示)代表沉浸式视频的数据。使用来自传感器的一些测量来计算设备的姿势和控制虚拟相机。用于姿势估计的传感器例如是陀螺仪、加速度计或罗盘。也可以使用(例如使用一套相机的)更复杂的系统。在这种情况下,至少一个处理器执行图像处理以估计设备10的姿势。使用一些其他测量来根据环境条件或用户的反应处理内容。用于观察环境和用户的传感器例如是麦克风、光传感器或接触传感器。也可以使用更复杂的系统,比如例如跟踪用户眼睛的摄像机。在这种情况下,至少一个处理器执行图像处理以操作预期测量。

[0079] 图29表示所述第一类型的系统的第五实施例,其中沉浸式视频呈现设备80包括用于处理和显示数据内容的所有功能。该系统包括沉浸式视频呈现设备80、传感器20和用户输入设备30。沉浸式视频呈现设备80被配置为可能地根据从传感器20和从用户输入设备30接收的数据来处理(例如,解码以及准备用于显示)代表沉浸式视频的数据。沉浸式视频呈现设备80可以连接到互联网,并且因此可以从互联网获得代表沉浸式视频的数据。在另一实施例中,沉浸式视频呈现设备80从存储代表沉浸式视频的数据的本地储存器(未示出)获得代表沉浸式视频的数据,所述本地储存器可以在呈现设备80上或者在例如可通过局域网访问的本地服务器上(未示出)。

[0080] 沉浸式视频呈现设备80在图35中示出。沉浸式视频呈现设备包括显示器801。显示器可以是例如OLED或LCD类型,触摸板(可选的)802,相机(可选的)803,与至少一个处理器804和至少一个通信接口806连接的存储器805。存储器805包括用于处理器804的参数和代码程序指令。存储器805也可以包括从传感器20和用户输入设备30接收的参数。存储器也可以足够大以存储代表沉浸式视频内容的数据。为此,可以存在若干类型的存储器,并且存储

器805可以是单个存储器,或者可以是若干类型的储存器(SD卡、硬盘、易失性或非易失性存储器…)。通信接口806使得沉浸式视频呈现设备能够与互联网通信。处理器804处理代表视频的数据,以便显示器801显示它们。相机803捕获环境的图像以用于图像处理步骤。从该步骤提取数据以便控制沉浸式视频呈现设备。

[0081] 用于处理增强现实、虚拟现实或增强的虚拟内容的第二系统在图30至图32中示出。这样的系统包括沉浸式墙壁。

[0082] 图30表示第二类型的系统。它包括显示器1000,其为从计算机4000接收数据的沉浸式(投影式)墙壁。计算机4000可以从互联网接收沉浸式视频数据。计算机4000通常直接或者通过网关5000或网络接口连接到互联网。在另一实施例中,由计算机4000从存储代表沉浸式视频的数据的本地储存器(未示出)获得沉浸式视频数据,所述本地储存器可以在计算机4000中或者在例如可通过局域网访问的本地服务器中(未示出)。

[0083] 该系统也可以包括传感器2000和用户输入设备3000。沉浸式墙壁1000可以是OLED或LCD类型。它可以配备有一个或若干个相机。沉浸式墙壁1000可以处理从传感器2000(或多个传感器2000)接收的数据。从传感器2000接收的数据可以相关于照明条件、温度、用户的环境,例如物体的位置。

[0084] 沉浸式墙壁1000也可以处理从用户输入设备3000接收的数据。用户输入设备3000发送诸如触觉信号之类的数据,以便给出关于用户情绪的反馈。用户输入设备3000的示例是手持设备,诸如智能电话、遥控器,以及具有陀螺仪功能的设备。

[0085] 传感器2000和用户输入设备3000数据也可以被传送到计算机4000。计算机4000可以根据从这些传感器/用户输入设备接收的数据来处理视频数据(例如,将它们解码以及准备它们以用于显示)。传感器信号可以通过沉浸式墙壁的通信接口来接收。该通信接口可以是蓝牙类型、WIFI类型或者任何其他类型的连接,优选地是无线的,但也可以是有线连接。

[0086] 计算机4000将处理后的数据以及可选地将控制命令发送到沉浸式墙壁1000。计算机4000被配置为处理将由沉浸式墙壁1000显示的数据,即准备它们以用于显示。处理可以由计算机4000专门完成,或者部分处理可以由计算机4000完成,并且部分由沉浸式墙壁1000完成。

[0087] 图31表示第二类型的另一系统。它包括沉浸式(投影式)墙壁6000,其被配置为处理(例如,解码以及准备数据以用于显示)并显示视频内容。它还包括传感器2000、用户输入设备3000。

[0088] 沉浸式墙壁6000通过网关5000从互联网或者直接从互联网接收沉浸式视频数据。在另一实施例中,由沉浸式墙壁6000从存储代表沉浸式视频的数据的本地储存器(未示出)获得沉浸式视频数据,所述本地储存器可以在沉浸式墙壁6000中或者在例如可通过局域网访问的本地服务器中(未示出)。

[0089] 该系统也可以包括传感器2000和用户输入设备3000。沉浸式墙壁6000可以是OLED或LCD类型。它可以配备有一个或若干个相机。沉浸式墙壁6000可以处理从传感器2000(或多个传感器2000)接收的数据。从传感器2000接收的数据可以相关于照明条件、温度、用户的环境,例如物体的位置。

[0090] 沉浸式墙壁6000也可以处理从用户输入设备3000接收的数据。用户输入设备3000发送诸如触觉信号之类的数据,以便给出关于用户情绪的反馈。用户输入设备3000的示例

是手持设备,诸如智能电话、遥控器,以及具有陀螺仪功能的设备。

[0091] 沉浸式墙壁6000可以根据从这些传感器/用户输入设备接收的数据来处理视频数据(例如,将它们解码以及准备它们以用于显示)。传感器信号可以通过沉浸式墙壁的通信接口来接收。该通信接口可以是蓝牙类型、WIFI类型或者任何其他类型的连接,优选地是无线的,但也可以是有线连接。沉浸式墙壁6000可以包括至少一个通信接口,以与传感器并且与互联网通信。

[0092] 图32图示了第三实施例,其中沉浸式墙壁用于游戏。一个或若干个游戏控制台7000优选地通过无线接口连接到沉浸式墙壁6000。沉浸式墙壁6000通过网关5000从互联网或者直接从互联网接收沉浸式视频数据。在另一实施例中,由沉浸式墙壁6000从存储代表沉浸式视频的数据的本地储存器(未示出)获得沉浸式视频数据,所述本地储存器可以在沉浸式墙壁6000中或者在例如可通过局域网访问的本地服务器中(未示出)。

[0093] 游戏控制台7000将指令和用户输入参数发送到沉浸式墙壁6000。沉浸式墙壁6000可能地根据从传感器2000和用户输入设备3000和游戏控制台7000接收的输入数据来处理沉浸式视频内容,以便准备用于显示的内容。沉浸式墙壁6000也可以包括内部存储器以存储要显示的内容。

[0094] 在一个实施例中,我们考虑全向视频以实现周围3D表面S到标准矩形帧F上的投影的格式来表示,该标准矩形帧F以适合于视频编解码器的格式表示。可以使用各种投影将3D表面投影到2D表面。例如,图2A示出了使用等矩形投影将示例性球体表面S映射到2D帧F,并且图2B示出了使用立方体映射将示例性立方体表面映射到如图2C中所示的2D帧。诸如金字塔、二十面体或八面体映射之类的其他映射可以将3D表面映射到2D帧中。

[0095] 然后,可以使用现有视频编解码器(例如,符合VP9、VP10、MPEG-2、H.264/AVC或H.265/HEVC的编解码器)对2D帧F进行编码。也可以利用适配于全向视频的属性的编解码器(例如,使用经调整的VP9、VP10、MPEG-2、H.264/AVC或H.265/HEVC编解码器)对2D帧F进行编码。在编码和解码之后,可以将经解码2D帧映射回到对应的3D表面(例如,用于等矩形映射的球体或者用于立方体映射的立方体)。然后,可以将3D表面投影到与用户视点相对应的“虚拟屏幕”上,以便获得最终的呈现帧。将2D帧解码以及从3D表面投影到呈现帧的步骤可以合并成单个步骤,其中解码帧的一部分被映射到呈现帧上。

[0096] 为了简化符号,我们可以将经解码2D帧也称为“F”,并且将呈现中使用的3D表面也称为S。应当理解,要编码的2D帧和要解码的2D帧可能由于视频压缩而不同,并且预处理中的3D表面和呈现中的3D表面也可能不同。在本申请中,我们可互换地使用术语“映射”和“投影”,可互换地使用术语“像素”和“样本”,并且可互换地使用术语“帧”和“图片”。

[0097] 等矩形映射

[0098] 图3图示了示例性逆等矩形映射,其中2D帧被映射到呈现帧。图3A示出了解码帧(也称为“等矩形帧”)中的x-y坐标系中的点P。图3B示出了使用 θ - Φ 极坐标从P映射到2D呈现表面上的点P'。图3C示出了使用笛卡尔坐标在3D呈现空间S中对应于P'的点P^{3d}。图3D示出了局部呈现帧G中的点P",其产生自投影到与P^{3d}处的表面相切的局部平面上的P^{3d}。通过构建,P"处于帧的中心。通常基于用户想要看到何物,例如,基于HMD或用户的手持设备指向何物,局部呈现帧G对应于2D帧F的一部分。局部呈现帧G也称为局部呈现版本。

[0099] 从P到P'、从P'到P^{3d}以及从P^{3d}到P"的映射分别表示为“f”、“3d”和“proj”。在图3

中,出于说明目的,以多个步骤(“f”、“3d”和“proj”)示出了逆等矩形映射。可以以更少的步骤来执行逆等矩形映射,例如,通过将“f”、“3d”和“proj”组合成一个或两个步骤。

[0100] 数学上,映射“f”可以表示为:

$$[0101] \quad f: M(x, y) \rightarrow M'(\theta, \varphi), \quad \theta = 2\pi x, \quad \varphi = \pi y \quad (1)$$

[0102] 当帧F中的像素M(x, y)被映射到点M'(\theta, \varphi)处的2D参数表面上时,假设归一化坐标。具有非归一化坐标:

$$[0103] \quad \theta = \frac{2\pi(x-\frac{w}{2})}{w}, \quad \varphi = \frac{\pi(\frac{h}{2}-y)}{h} \quad (2)$$

[0104] 从点M'(\theta, \varphi)到坐标M^{3d}的3D表面上的映射“3d”可以表示为:

$$[0105] \quad 3d: M' \rightarrow M^{3d}, \quad M^{3d} = \begin{bmatrix} \sin\left(\varphi - \frac{\pi}{2}\right) \cos(\theta) \\ \sin\left(\varphi - \frac{\pi}{2}\right) \sin(\theta) \\ \cos\left(\varphi - \frac{\pi}{2}\right) \end{bmatrix} \quad (3)$$

[0106] 为了从点M^{3d}返回到帧F,我们计算逆变换T⁻¹:

$$[0107] \quad T^{-1}: M^{3d} \rightarrow M, M = f^{-1}(3d^{-1}(M^{3d})). \quad (4)$$

[0108] 从3D表面S上的点M^{3d}(X, Y, Z),我们可以使用标准的笛卡尔到极坐标变换来返回到2D参数帧:

$$[0109] \quad 3d^{-1}: M^{3d} \rightarrow M', \quad \theta = \text{atan}\left(\frac{Y}{X}\right) + \pi, \quad \varphi = -\text{atan}\left(\frac{\sqrt{X^2+Y^2}}{Z}\right) + \frac{\pi}{2} \quad (5)$$

[0110] 为了进一步返回到2D帧F,我们可以使用:

$$[0111] \quad f^{-1}: M'(\theta, \varphi) \rightarrow M(x, y) \quad (6)$$

[0112] 对于奇点(典型地,在极点处),当x和y接近0时,我们可以直接设置:

$$[0113] \quad \theta = 0, \quad \varphi = \text{sign}(Z) \frac{\pi}{2} \quad (7)$$

[0114] 应当注意,针对模块化情况应当采取特别注意。

[0115] 图4A示出了将点从3D表面投影(“proj”)到呈现帧G。如果从G中的局部呈现帧查看的点Q'接近P',则与表面S相切的平面可以通过表面自身来近似。在图4B中,我们示出Q"可以通过Q在表面上的图像Q'来近似。

[0116] 从3D空间到2D表面的投影经常引入一些影响或变更,这对于以与针对传统平面获得的质量/比特率折衷类似的质量/比特率折衷对所得视频进行编码可能是有挑战性的。这些影响或变更可能包括:

[0117] -强几何失真:。

[0118] • 直线不再是直的,

[0119] • 规格化正交坐标系不再是规格化正交的

[0120] -非均匀像素密度:要编码的帧中的像素不总是表示要编码的表面上相同的大小(即,呈现阶段期间图像上的相同的大小)

[0121] -强不连续性:帧布局可能在两个邻近像素之间引入强不连续性

[0122] -帧中可能出现一些周期性(例如从一个边界到相对的边界)

[0123] 表1列出了针对不同映射的一些变更:

[0124] 表1

映射类型	等矩形	立方体映射	金字塔
3D表面	球体	立方体	金字塔
直线	连续地失真的	分段直的	分段直的
规格化正交局部帧	否	是,除了在面边界上	否,除了在正方形面上
像素密度	非均匀(在赤道线上较高)	几乎不变	非均匀,除了在正方形面上
不连续性	否	是,在面边界上	是,在面边界上
连续性	是,水平的	是,在一些面之间	是,在一些面之间

[0125] 在下文中,我们主要使用HEVC编码器和解码器来说明全向视频的编码和解码。应当注意,本原理可以应用于其他编码器或解码器。

[0127] 图5图示了示例性HEVC编码器500。为了对具有一个或多个图片的视频序列进行编码,将图片分割成一个或多个切片,其中每个切片可以包括一个或多个切片片段。将切片片段组织成译码单元、预测单元和变换单元。

[0128] 在本申请中,术语“重建的”和“解码的”可以互换使用。通常但不是必需地,在编码器侧使用术语“重建的”,而在解码器侧使用“解码的”。

[0129] HEVC规范在“块”与“单元”之间进行区分,其中“块”寻址样本阵列(例如,亮度,Y)中的特定区域,而“单元”包括所有编码色彩分量(Y,Cb,Cr或单色)的同位(collocated)块、与块相关联的语法元素和预测数据(例如,运动矢量)。

[0130] 对于译码,将图片分割成具有可配置大小的正方形形状的译码树块(CTB),并且将连贯的译码树块集合分组成切片。译码树单元(CTU)包含编码色彩分量的CTB。CTB是分割成译码块(CB)的四叉树的根,并且将译码块分割成一个或多个预测块(PB)并形成分割成变换块(TB)的四叉树的根。对应于译码块、预测块和变换块,译码单元(CU)包括预测单元(PU)和树形结构的变换单元(TU)集合,PU包括所有色彩分量的预测信息,并且TU包括每个色彩分量的残差译码语法结构。亮度分量的CB、PB和TB的大小适用于对应的CU、PU和TU。在本申请中,术语“块”可以用于指代CTU、CU、PU、TU、CB、PB和TB中的任何一个。另外,“块”也可以用于指代如H.264/AVC或其他视频译码标准中规定的宏块、分区和子块,并且更一般地指代各种大小的数据阵列。

[0131] 在示例性编码器500中,如下所述,由编码器元件对图片进行编码。要编码的图片以CU为单位进行处理。使用帧内或帧间模式对每个CU进行编码。在以帧内模式对CU进行编码时,其执行帧内预测(560)。在帧间模式中,执行运动估计(575)和补偿(570)。编码器判定(505)使用帧内模式或帧间模式中的哪一个对CU进行编码,并且通过预测模式标志来指示帧内/帧间判定。通过从原始图像块中减去(510)预测块来计算预测残差。

[0132] 根据相同切片内的重建的相邻样本来预测帧内模式中的PU。在HEVC中,35个帧内

预测模式的集合是可用的,其包括DC、平面和33个角度预测模式,如图6A中所示。根据邻近于当前块的行和列来重建帧内预测参考。使用来自先前重建块的可用样本,该参考在水平和垂直方向上延伸超过块大小的两倍。当角度预测模式用于帧内预测时,可以沿着由角度预测模式指示的方向来复制参考样本,如图6B中的示例所示。在本申请中,角度预测模式也称为方向预测模式。

[0133] 当预测地对亮度帧内预测模式进行译码时,HEVC考虑三个最可能模式(MPM),MPM0、MPM1和MPM2,如表2中所示,其中“L”表示相邻左块的帧内预测模式,并且“A”表示相邻上块的帧内预测模式。相邻块和左块可以具有与当前块不同的大小,如图6C中所示。在三个最可能模式之中,前两个模式通过上PB和左PB的亮度帧内预测模式来初始化(如果那些PB可用并且使用帧内预测模式进行译码)。任何不可用的帧内预测模式都被认为是DC模式。

[0134] 表2

条件			MPM0	MPM1	MPM2
L=A	L ≠ 平面 且 L ≠ DC		L	L+1	L-1
	否则		平面	DC	26 (垂直)
L≠A	L ≠ 平面 且 A ≠ 平面		L	A	平面
	否则	L ≠ DC 且 A ≠ DC	L	A	DC
		否则	L	A	26 (垂直)

[0136] 在前两个最可能模式不相等时,将第一最可能模式(MPM0)设置为L,将第二最可能模式(MPM1)设置为A,并且将第三最可能模式(MPM2)设置为等于平面模式、DC或垂直,这根据按该顺序这些模式中的哪一个不是前两个模式之一的副本。在前两个最可能模式相同时,如果该第一模式具有平面或DC的值,则将第二和第三最可能模式分配为平面、DC或垂直,这根据按该顺序这些模式中的哪一个不是副本。在前两个最可能模式相同并且第一模式具有角度值时,将第二和第三最可能模式选择为最接近第一的角度(即,k的值)的两个角度预测模式。

[0137] 如果当前亮度预测模式是三个MPM之一,则仅将MPM索引传送到解码器。否则,通过使用5比特固定长度代码,将除了三个MPM之外的当前亮度预测模式的索引传送到解码器。使用MPM对帧内预测模式进行编码依赖于相邻块可能共享相同预测模式的假设(当纹理内容相似时)。

[0138] 对于帧间CU,将对应的译码块进一步分割成一个或多个预测块。在PB级执行帧间预测,并且对应的PU包含关于如何执行帧间预测的信息。

[0139] 可以以两种方法、即“合并模式”和“高级运动矢量预测(AMVP)”来用信号通知运动信息(即,运动矢量和参考图片索引)。

[0140] 在合并模式中,视频编码器或解码器基于已译码块来组装候选列表,并且视频编码器用信号通知候选列表中的候选之一的索引。在解码器侧,基于用信号通知的候选来重建运动矢量(MV)和参考图片索引。

[0141] 在AMVP中,视频编码器或解码器基于根据已译码块确定的运动矢量来组装候选列表。然后,视频编码器用信号通知候选列表中的索引以识别运动矢量预测器(MVP)并用信号

通知运动矢量差 (MVD)。在解码器侧,将运动矢量 (MV) 重建为MVP+MVD。

[0142] 在HEVC中,用于运动补偿的运动信息的精度针对亮度分量是四分之一样本(也称为四分之一像素或1/4像素)并且针对色度分量是八分之一样本(也称为1/8像素)。7抽头或8抽头内插滤波器用于分数样本(fractional-sample)位置的内插,即,对于亮度可以寻址水平和垂直方向二者上的全样本定位的1/4、1/2和3/4。

[0143] 然后,对预测残差进行变换(525)和量化(530)。对量化的变换系数以及运动矢量和其他语法元素进行熵译码(545)以输出比特流。编码器也可以跳过变换并且在 4×4 TU的基础上将量化直接应用于非变换的残差信号。编码器也可以绕过变换和量化二者,即,在不应用变换或量化处理的情况下直接对残差进行译码。在直接PCM译码中,不应用预测,并且将译码单元样本直接译码到比特流中。

[0144] 编码器对编码块进行解码以提供用于进一步预测的参考。对量化的变换系数进行去量化(540)和逆变换(550)以对预测残差进行解码。将解码预测残差和预测块组合(555),重建图像块。将环内滤波器(565)应用于重建图片,例如,以执行去块/SAO滤波以减少编码伪像。将滤波图像存储在参考图片缓冲器(580)处。

[0145] 图7图示了示例性HEVC视频解码器700的框图。在示例性解码器700中,如下所述,由解码器元件对比特流进行解码。视频解码器700通常执行与如图5中描述的编码通道对等的解码通道,其执行视频解码,如编码视频数据的部分。

[0146] 具体地,解码器的输入包括视频比特流,其可以由视频编码器500生成。首先对比特流进行熵解码(730)以获得变换系数、运动矢量和其他译码信息。对变换系数进行去量化(740)和逆变换(750)以对预测残差进行解码。将解码预测残差和预测块组合(755),重建图像块。预测块可以从帧内预测(760)或运动补偿预测(即,帧间预测)(775)获得(770)。如上所述,AMVP和合并模式技术可以用于导出用于运动补偿的运动矢量,运动补偿可以使用内插滤波器来计算参考块的子整数样本的内插值。将环内滤波器(765)应用于重建图像。将滤波图像存储在参考图片缓冲器(780)处。

[0147] 对准(aligning)

[0148] 图8图示了其中呈现帧中的直线(820)在要编码的帧中不被视为直线(810)的示例。具体地,图8图示了当前预测块(830)和对应参考样本(850)的局部映射和呈现。在局部映射和呈现中,只有预测块周围的一小部分被映射到呈现版本。左图像是要编码的示例性等矩形帧F的裁剪部分。右图像是示例性呈现帧中的裁剪部分,以显示出预测块的局部呈现版本(840)。等矩形帧中的预测块(830)的相邻样本(850)被映射到呈现帧G中的相邻区域(860)。参考帧(870)以预测块的中心为中心。箭头(880)示出如果对呈现版本进行编码,则可以由传统编码器选择的可能的方向模式。

[0149] 在图8的右图像中,合理的是,假设帧内预测模式可以在呈现帧中沿着直线传播,并且MPM将提高译码效率。然而,在等矩形帧中,直线在预测块(830)中不再是直的,因此当在等矩形内容中对块的帧内预测模式进行编码时直接应用MPM可能不是非常有效。

[0150] 我们提出针对全向视频改进MPM推导过程,例如,通过使相邻块的帧内预测模式适配以遵循呈现帧中的几何形状。

[0151] 我们注意到,等矩形帧中的几何改变是在垂直方向上。在一个实施例中,我们首先将2D帧F中的相邻帧内预测模式转换成呈现帧G中的方向(即,角度),然后将G中的方向转换

为用于当前块的帧内预测模式。也就是说,相邻帧内预测模式现在与当前块对准并且表示在与当前块相同的垂直水平处,因此可以抵消由等矩形映射引起的几何改变的影响。

[0152] 图9示出了根据本原理的示例性实施例的使MPM列表适配以用于对帧内预测模式进行编码或解码的示例性方法900。可以在编码器或解码器侧使用方法900。当在编码器侧针对要编码的当前块使用时,可以访问(910)相邻的左块和上块。然后,可以转换(920)用于相邻块的帧内预测模式,以使它们与当前块对准。然后,可以基于用于相邻块的经转换的帧内预测模式来确定(930)MPM列表。然后,可以对针对当前块测试的当前帧内预测模式进行编码。如果当前帧内预测模式在MPM列表中,则仅对MPM列表中的帧内预测模式的索引进行编码。否则,可以通过固定长度编码对帧内预测模式进行编码。如果当前帧内预测模式的RD成本小于其他帧内预测模式的RD成本并且小于帧间预测的RD成本,则当前块可以选择将当前帧内预测模式用于当前块。

[0153] 当在解码器侧使用方法900时,如果比特流指示用于当前块的帧内预测模式被编码为MPM之一,则解码器对MPM索引进行解码。可以访问(910)相邻的左块和上块。然后,可以转换(920)用于相邻块的帧内预测模式,以使它们与当前块对准。然后,可以基于用于相邻块的经转换的帧内预测模式来确定(930)MPM列表。然后,可以基于经解码的MPM索引将帧内预测模式解码为MPM之一。

[0154] 方法900对于相邻块仅考虑左块和上块。更一般地,其他块也可以被认为是相邻块。

[0155] 使用根据HEVC导出的MPM作为示例,表3示出了示例性所调整的MPM,其中 T_L^{-1} 或 T_A^{-1} 分别基于例如由左块或上块的变形索引所指示的垂直定位,将左相邻块或上相邻块的帧内预测模式索引转换成呈现帧中的角度,并且 T_c 基于例如指示当前块的变形索引的垂直定位将呈现帧中的角度转换回帧内预测模式索引。“ T_c ”和“ T_L^{-1} ” (“ T_A^{-1} ”) 的转换可以组合为例如 $F = T_c(T_L^{-1}())$ 或 $F = T_c(T_A^{-1}())$, 其中使用“ T_L^{-1} ”还是“ T_A^{-1} ”取决于对应的MPM是来自相邻的左块还是上块。

[0156] 表3

条件			MPM0	MPM1	MPM2
$T_c(T_L^{-1}(L)) = T_c(T_A^{-1}(A))$	L≠平面 且 L≠DC		$T_c(T_L^{-1}(L))$	$T_c(T_L^{-1}(L+1))$	$T_c(T_L^{-1}(L-1))$
	否则		平面	DC	垂直
$T_c(T_L^{-1}(L)) \neq T_c(T_A^{-1}(A))$	L≠平面 且 A≠平面		$T_c(T_L^{-1}(L))$	$T_c(T_A^{-1}(A))$	平面
	否则	L≠DC 且 A≠DC	$T_c(T_L^{-1}(L))$	$T_c(T_A^{-1}(A))$	DC
		否则	$T_c(T_L^{-1}(L))$	$T_c(T_A^{-1}(A))$	垂直

[0158] 在表3中,当 $L \neq \text{平面}$ 且 $L \neq \text{DC}$ 时,将 $T_c(T_L^{-1}())$ 应用于 $L+1$ 或 $L-1$ 。可能发生 $T_c(T_L^{-1}(L+1))$ 或 $T_c(T_L^{-1}(L-1))$ 与 $T_c(T_L^{-1}(L))$ 相同。为了仍然得到三个不同MPM,在不同实施例中,针对模式 $L+1$ 和 $L-1$, $T_c(T_L^{-1}())$ 可以应用于 L ,并且分别将MPM1和MPM2调整为 $T_c(T_L^{-1}(L))+1$ 和 $T_c(T_L^{-1}(L))-1$ 。即,当“ $T_c(T_L^{-1}(L)) = T_c(T_A^{-1}(A))$ ”及“ $L \neq \text{平面}$ 且 $L \neq \text{DC}$ ”时,可将MPM导出为:

[0159] 表3A

	条件	MPM0	MPM1	MPM2
[0160]	$T_c(T_L^{-1}(L))=T_c(T_A^{-1}(A)), L \neq \text{平面} \text{ 且 } L \neq \text{DC}$	$T_c(T_L^{-1}(L))$	$T_c(T_L^{-1}(L))+1$	$T_c(T_L^{-1}(L))-1$

[0161] 在下文中,我们详细描述如何确定用于等矩形映射的转换“ T_i ”和“ T_i^{-1} ”,其中 T_i 将呈现帧中的角度映射到在解码帧(或要编码的帧)中表达的帧内预测模式索引,例如,在HEVC中,从角度 $[45^\circ, \dots, 135^\circ]$ 、DC或平面到索引 $[0, \dots, 35]$,并且 T_i^{-1} 将帧内预测模式索引映射成呈现帧中的角度,例如,在HEVC中,从帧内预测模式索引 $[0, \dots, 35]$ 到角度 $[45^\circ, \dots, 135^\circ]$ 、DC或平面。针对所有变形索引,我们通过构建定义 $T(\text{DC})=1, T^{-1}(1)=\text{DC}, T(\text{Planar})=0, T^{-1}(0)=\text{Planar}$ 。

[0162] 图10A示出了x-y坐标系中的归一化2D帧,以及图10B示出了用于等矩形映射的示例性变形索引推导。这里,变形索引仅取决于块的y坐标,例如但不限于,将块的中心、拐角用于块的y坐标。为了便于表示,我们将用于指示块的y坐标的点表示为锚点。由于等矩形映射的属性,变形索引的演变通过 $\cos(y)$ 引导。在实践中,给定变形索引的数量N,我们可以使用以下来计算块的变形索引w:

$$[0163] \quad w(x, y) = [N(1 - \cos(\pi y))]$$

[0164] 其中(x, y)是块在图片中的归一化坐标,如图10A中所示。利用该函数,我们可以得到最多(N+1)个不同的方向转换表,每个变形索引一个。可以使用其他缩放因子或变型来获得变形索引。通常,可以基于用于生成2D帧的映射来计算变形索引。

[0165] 图11A图示了根据本原理的实施例的用于计算转换“T”的示例性工作流程。

[0166] 因为等矩形映射仅在垂直方向上使图像失真,所以我们设置 $x=0$ 而不失一般性。对于给定的变形索引w,与y坐标y和 $x=0$ 相对应的点是:

$$[0167] \quad P = \begin{bmatrix} 0 \\ y \end{bmatrix} \quad (8)$$

[0168] 并且可以将变形索引计算为:

$$[0169] \quad w = [N(1 - \cos(\pi y))] \quad (9)$$

[0170] 可以将P通过f的图像P' 计算为:

$$[0171] \quad P' = f(P) \quad (10)$$

[0172] 可以将P' 通过3d的图像 P^{3d} 计算为:

$$[0173] \quad P^{3d} = 3d(P') \quad (11)$$

[0174] 然后,可以使用proj()函数来计算P": $P'' = \text{proj}(P^{3d})$ 。

[0175] 可以将从P指向方向d的点表达为:

$$[0176] \quad Q = P + dP \quad (12)$$

[0177] 其中

$$[0178] \quad dP = \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} \quad (13)$$

[0179] 其中 α 是帧内预测模式索引d的方向角(对应于变形索引0的每个方向索引的角度),即:

$$[0180] \quad \alpha = T_0^{-1}(d) \quad (14)$$

[0181] 可以将Q通过f的图像Q' 计算为:

$$[0182] \quad Q' = f(Q) \quad (15)$$

[0183] 可以将Q' 通过3d的图像Q^{3d}计算为:

$$[0184] \quad Q^{3d} = 3d(Q')$$

[0185] 可以将Q^{3d}通过proj的图像Q'' 计算为:

$$[0186] \quad Q'' = Proj(Q^{3d}) \quad (16)$$

[0187] 然后,可以将表示P'' 与Q'' 之间的差的矢量dP'' 计算为:

$$[0188] \quad dP'' = Q'' - P'' \quad (17)$$

[0189] 因此,可以将变形索引w在方向d的角度计算为:

$$[0190] \quad \beta = T_w^{-1}(d) = atan\left(\frac{dP''_y}{dP''_x}\right) \quad (18)$$

[0191] 其中dP''_x和dP''_y是dP''的x分量和y分量。然后,可以将T_w计算为T_w⁻¹的逆。

[0192] 在另一实施例中,在该点的局部呈现帧中解释该方向,如图11B中所示。这里我们计算函数S_i,其将呈现帧中的角度映射到呈现帧中的帧内预测模式索引。

[0193] 对于给定的变形索引w,对应于y坐标y,即:

$$[0194] \quad P = \begin{bmatrix} x \\ y \end{bmatrix} \quad (19)$$

[0195] 并且变形索引

$$[0196] \quad w = [N(1 - \cos(\pi y))] \quad (20)$$

[0197] 可以将P通过f的图像P' 计算为:

$$[0198] \quad P' = f(P) \quad (21)$$

[0199] 可以将P' 通过3d的图像P^{3d}计算为:

$$[0200] \quad P^{3d} = 3d(P') \quad (22)$$

[0201] 通过构建,来自P的投影函数为:

$$[0202] \quad P'' = Proj(P^{3d}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (23)$$

[0203] 在帧G中,计算从P'' 指向与角度α相对应的方向d的点:

$$[0204] \quad Q'' = P'' + \epsilon dP'' = \epsilon dP'' \quad (24)$$

[0205] 其中

$$[0206] \quad dP'' = \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} \quad (25)$$

[0207] 其中α是帧内预测模式索引d的方向角(对应于变形索引0的每个方向索引的角度),即:

$$[0208] \quad \alpha = S_0^{-1}(d) \quad (26)$$

[0209] 可以将Q^{3d}计算为:

$$[0210] \quad Q^{3d} = Proj^{-1}(Q'') \quad (27)$$

[0211] 然后,可以将 Q' 计算为:

$$[0212] \quad Q' = 3d^{-1}(Q^{3d}) \quad (28)$$

[0213] 可以将 Q 计算为:

$$[0214] \quad Q = f^{-1}(Q') \quad (29)$$

[0215] 可以将表示 Q 与 P 之间的差的矢量 dP 计算为:

$$[0216] \quad dP = Q - P \quad (30)$$

[0217] 因此,可以将变形索引 w 关于方向 d 的角度计算为:

$$[0218] \quad \beta = S_w^{-1}(d) = \text{atan}\left(\frac{dP_y}{dP_x}\right) \quad (31)$$

[0219] 注意,在如图11A中所示的第一实施例中,其中 d 是在解码帧中表达的方向的索引(对应于函数 T),对于呈现帧中的直线,索引 d 可以沿着垂直轴变化,而在如图11B中所示的第二实施例中,索引 d 对于所有块是相同的。然而,差别在于预测阶段:对于第一实施例,块预测方向将是与 d 相关联的方向,而在第二实施例中,从呈现帧到解码帧的转换是必要的。

[0220] 对于帧内方向模式,将呈现帧中的给定角度变换为方向模式索引的转换(函数 T 或 S),以及将给定方向模式索引变换回到呈现帧中的角度(角度 β)的逆转换(函数 T^{-1} 或 S^{-1}),可以被预先计算并存储在转化表中,诸如针对HEVC帧内预测模式生成的表4。

[0221] 表4

变形索引 w / 方向模式索引 d	$d = 2$	$d = 3$...	$d = 33$	$d = 34$
[0222] $w = 0$	135°	129.375°	...	39.375°	45°
$w = 1$	131°	126°	...	36°	39°
$w = 2$	130°	135°	...	35°	38°
...					

[0223] 对于表中的变形索引 w ,我们针对方向模式的各个索引提供对应的以度数为单位的角度。注意,通过构建,变形索引0表示传统HEVC中的角度。然后,我们将 $T_i(x)$ 定义为表中的帧内预测方向模式索引,在变形索引 i 处,最接近具有角度 x 的那个。例如,在表4中, $T_2(134.5) = 3$:对于变形索引2,关于 134.5° 的最接近的方向模式索引是索引3(对应于 135°)。类似地,对于逆变换, $T_i^{-1}(j)$ 是表4中在变形索引 i 处与方向模式索引 j 相对应的角度。例如, $T_1^{-1}(33) = 36^\circ$ 。

[0224] 图12图示了根据本原理的实施例的在使用等矩形映射时基于相邻左块获得MPM的示例性处理1200。方法1200可用于获得如表3或表3A中所示的 $T_c(T_L^{-1}(L))$ 。本原理也可以用于基于其他相邻块获得MPM。注意, $T_L^{-1}()$ 计算相邻左块的变形索引,并将左块的帧内预测模式索引转换成角度,并且 $T_c()$ 得到当前块的变形索引,并将角度从 T_L^{-1} 转换回到帧内预测模式索引。

[0225] 为了计算 $T_L^{-1}(L)$,获得(1210)左块 L 的坐标 (x_L, y_L) 。然后,可以获得(1220)左块的变形索引,作为 $L_w = \lceil N(1 - \cos(\pi y_L)) \rceil$,其中 y_L 指示左块的定位。然后,可以计算(1230)左块的变形角度,作为 $\alpha_L = T_{Lw}^{-1}(d_L)$,其中 d_L 是用于左块的帧内预测模式索引。

[0226] 为了计算 $T_c()$,获得(1240)当前块 L 的坐标 (x_c, y_c) 。然后,可以获得(1250)当前块

的变形索引,作为 $C_w = \lceil N(1 - \cos(\pi y_c)) \rceil$,其中 y_c 指示左块的定位。然后,可以计算(1260)基于左块的用于MPM的适配帧内预测模式索引,作为 $d_c = T_{C_w}(\alpha_L) = T_{C_w}(T_{L_w}^{-1}(d_L))$ 。

[0227] 应当注意,用于左块的新帧内预测模式索引与当前块处于相同的垂直水平,并且与当前块的帧内预测模式相当。也就是说,针对相邻左块和当前块的角度表示在呈现帧中,其中帧内模式更可能如上所述传播。

[0228] 图13图示了赤道处的示例性当前块C,其相邻的左块L也在赤道处。使用表4作为等矩形映射的示例,可能发生以下MPM推导:

[0229] -左块的变形索引为0,因为左块在赤道处。经解码的方向帧内模式为 $d_L = 2$ 。根据表4,我们推导左角度: $\alpha_L = 135^\circ$ (第一行,第一列)。

[0230] -当前块的变形索引也为0,因为当前块也在赤道处。来自左块的预测帧内方向则为2(最近的角度在第一行,第一列)。对于等矩形映射,在当前块和相邻块在相同的垂直水平时,例如,在当前块和相邻块的锚点在相同的垂直水平时,不需要转换。注意,对于不同的块大小,左块和当前块不一定共享相同的变形索引。

[0231] -上块A的变形索引为2。经解码帧内预测模式索引为 $d_A = 3$ 。根据表4,我们推导上角度: $\alpha_A = 135^\circ$ (第三行,第二列)。

[0232] -来自上块的预测帧内方向则为2(最近的角度在第一行,第一列)。

[0233] 根据MPM推导,两个相邻块提供相同的方向预测器,即, $T_c(T_L^{-1}(L)) = T_c(T_A^{-1}(A))$, $L \neq \text{平面}$ 且 $L \neq \text{DC}$ 。使用表3A,三个MPM则为:MPM0= $T_c(T_L^{-1}(L)) = 2$,MPM1= $T_c(T_L^{-1}(L)) + 1 = 3$,以及MPM2= $T_c(T_L^{-1}(L)) - 1 = 1$ 。

[0234] 在上文中,我们使用HEVC MPM作为示例,描述了基于相邻块和当前块的垂直水平来调整相邻帧内预测模式。当以不同方式导出MPM时,可以应用本原理。例如,在最近的发展中,JVET(联合视频探索小组)的联合探索软件(JEM)使用两倍的HEVC中的帧内方向模式,如图17中所示。此外,已经提出将MPM列表从3个帧内模式扩展到6个帧内模式。表5中详述了MPM列表的推导。该新列表也考虑从相邻的左块和上块导出的两个模式。代替如图6C中所示的针对左候选和上候选考虑来自左上拐角的第一块,最终的L候选和A候选分别对应于到当前块的左侧和右侧的相邻块之中的最频繁模式。

[0235] 表5

条件			MPM 0	MPM 1	MPM 2	MPM3	MPM4	MPM5
L = A	L ≠ 平面 且 L ≠ DC		L	平面	L+1	L-1	L+2	DC
	否则		平面	DC	26 (Ver)	10 (Hor)	2	18
L ≠ A	L ≠ 平面 且 A ≠ 平面	L = DC 或 A = DC	L	A	平面	Max-1	Max+1	Max+2
		否则	L	A	平面	DC	Max+1	Min-1
	否则	L + A < 2	L	A	26 (Ver)	10 (Hor)	2	18
		否则	L	A	DC	Max-1	Max+1	Max+2

[0237] 对于如表5中描述的MPM,可以将左侧的帧内预测模式(“L”)调整为($T_c(T_L^{-1}(L))$),

可以将上方的帧内预测模式 (“A”) 调整为 $(T_C(T_A^{-1}(A)))$, 可以将帧内预测模式 $\text{mode}=26$ 、10、2 或 18 调整为 $(T_C(T_L^{-1}(\text{mode})))$ 或调整为 $(T_C(T_A^{-1}(L)))$, 并且可以将帧内预测模式 (“Max”) 调整为 $(T_C(T_{\text{Max}}^{-1}(\text{Max})))$, 其中 Max 代表给定编解码器的最高方向索引, 在 HEVC 的情况下为 33, 在 JEM 的情况下为 64。

[0238] 当使用特定映射并在流中用信号通知时, 可以自动推导块坐标与变形索引之间的关系。

[0239] 第一个方法是用信号通知用于 MPM 转化的变形索引。可以在序列级、切片级或者在 CTU、CU 或 PU 级完成用信号通知。当在块级用信号通知时, 使用上下文和熵译码对变形索引进行编码。

[0240] 在编码器侧, 变形表可以被构建并且在序列级、切片级或块级被发送。表的构建被认为是编码方法。表的编码可以使用熵和上下文译码。

[0241] 当使用特定映射并在流中用信号通知时, 可以自动推导要使用的变形角度。在这种情况下, 编码器和解码器都使用相同的表而无需将其传送。

[0242] 在上文中, 我们讨论了针对等矩形映射来调整帧内预测模式。对于立方体映射, 映射不修改立方体面内的传统模式。然而, 修改出现在边缘处。图 14A 示出了立方体映射的示例。面可以在矩形帧中不同地分布。在一些边缘处, 必须实现特定映射以使编码器/解码器指向相应地旋转的另一面上的重建相邻者的正确定位。

[0243] 然后, 可以使用从球体到立方体的投影, 以相同的方式应用方向的改变。图 14B 示出了边缘处的方向改变的示例。在该示例中, 利用用线段描绘的方向模式将左面上的块译码。然后, 用于右侧上的当前块的 MPM 将被修改, 如利用线段所示。

[0244] 减少可用帧内预测模式的数量

[0245] 图 15 示出了全向图片的示例性等矩形布局以及局部参考帧沿垂直坐标的演变。如可以从图 15 中看出的那样, 专用的局部参考帧, 即示出由投影引入的形变的坐标系, 以球体的赤道 (1510) 为中心是规格化正交的。随着我们沿着垂直方向将参考帧远离中心平移, 参考帧 (1520, 1530) 不再是规格化正交的。具体地, 随着局部参考帧朝向顶部或底部边界 (对应于球体的极点) 移动得更接近, 局部参考帧在垂直方向上变得越来越“收缩”。

[0246] 图 16A、图 16B、图 16C、图 16D 和图 16E 示出了分别从不同纬度: 0° 、 45° 、 66° 、 80° 和 85° 查看的相同图案的图像。在如图 16A 中所示的赤道处, 所有方向都可用且有规律地分布。越接近极点来查看图案, (中心处的) 方向在矩形帧中变得越水平。

[0247] 如上面针对图 6A 和图 6B 所讨论的那样, 在方向帧内预测模式中, 使用周围的先前重建样本和预测的方向来计算当前块的预测, 其中样本沿着所选方向“传播”到当前预测块上。然而, 在全向视频中 (其中局部参考帧取决于其中心位置, 因此不总是规格化正交的), 可以减少帧中的定位处的可能方向的数量以提高压缩效率。例如, 在等矩形 (Er) 布局中, 图 16 示出了在赤道处, 所有方向都可用, 但是非常接近极点, 只有水平方向会是相关的。

[0248] 基于以上观察, 我们提出进一步改进帧内 MPM 推导。在各种实施例中, 我们可以使 MPM 的推导适配以便考虑取决于几何失真的帧内方向的减少。编码方法也可以被适配为减少针对全向视频中的块要估计的模式的数量。也可以改进表示帧内方向的语法元素的译码。

[0249] 表 6

[0250]

条件			MPM 0	MPM 1	MPM 2	MPM3	MPM4	MPM5
F(L)= F(A)	L≠平面 且 L≠DC		F(L)	平面	F(L)+ d	F(L)-d	F(L)+2d	DC
	否则		平面	DC	F(26) (Ver)	F(10) (Hor)	F(2)	F(18)
F(L)≠ F(A)	L≠平面 且 R≠平面	L=DC 或 A=DC	F(L)	F(A)	平面	F(Max)- d	F(Max)+ d	F(Max)+2 d
		否则	F(L)	F(A)	平面	DC	F(Max)+ d	F(Min)-d
	否则	L+A<2	F(L)	F(A)	F(26) (Ver)	F(10) (Hor)	F(2)	F(18)
		否则	F(L)	F(A)	DC	F(Max)- d	F(Max)+ d	F(Max)+2 d

[0251] 表6示出了MPM适配于d,其中参数d表示适配于几何布局的角度步长。例如,当“L≠平面且L≠DC”且“F(L)=F(A)”时,针对MPM2,“F(L)+d”代替“F(L)+1”。也就是说,我们认为与对应于“F(L)”的角度邻近的角度是“F(L)+d”,因为我们可以跳过“F(L)+1”与“F(L)+d-1”之间的帧内预测模式。在赤道处,d=1,并且当块变得更接近极点时d增加(并且跳过更多帧内预测模式)。将在下面进一步详述d的计算。

[0252] 我们使用以下符号:

[0253] -a表示局部参考帧的垂直轴与水平轴之间的比例因子。通过构建,a=1用于标准视频或在等矩形视频的赤道处,以及a=0在等矩形视频的极点处。

[0254] -对于“标准”局部帧(即,a=1),方向的数量为N(a),例如,在当前JVET探索模型(JEM)中为65或者在HEVC中为33。

[0255] -以度数为单位的“标准”角度步长为s(a=1)=180/(N(1)-1)。例如:在HEVC中为5.625°。

[0256] -我们将180°的“标准”帧内方向的预测角度定义为Z(a=1),并且将水平方向下方45°和水平方向上方135°的“标准”帧内方向的预测角度分别定义为Z⁻(a=1)=45°和Z⁺(a=1)=135°。

[0257] -N⁻(a)=(N(a)-1)/4是水平方向下方的方向的数量,并且N⁺(a)=3(N(a)-1)/4是水平上方的方向的数量,如图18中的示例中所示。

[0258] -通过构建,d(a=1)=1。

[0259] 为了利用x轴与y轴之间的比例因子a来推导用于给定局部帧F_L的方向,应用以下逻辑:

[0260] -针对比例因子a的角度通过以下给出:

$$[0261] \quad Z^-(a) = a \tan(a \cdot \tan(Z^-(1))), Z^+(a) = 3 \cdot Z^-(a) \quad (32)$$

[0262] -在帧中保持相同的角度步长,我们将可用帧内预测方向的数量推导为:

$$[0263] \quad N^-(a) = \left\lceil \frac{Z^-(a)}{s(1)} \right\rceil, N^+(a) = \left\lceil \frac{Z^+(a)}{s(1)} \right\rceil \quad (33)$$

[0264] -然后,我们推导参数d:

$$[0265] \quad d(a) = \left\lceil \frac{N^+(1)}{N^+(a)} + \frac{N^-(1)}{N^-(a)} \right\rceil \quad (34)$$

[0266] 图19提供了针对 $a=1, 0.5, 0.25$ 和 0.125 的以上计算的示例。在图19的顶部,示出了局部参考帧。在图19的中部,示出了如使用等式(32)计算的针对比例因子 a 的角度,并且对于每个比例因子 a 保持相同的方向数量。在图19的底部,我们以将角度步长维持在相同的水平为目标,并且随着比例因子 a 减小,方向的数量减少。如可以从图19A、图19B和图19C中看出的那样,虽然图19A中的可用的帧内预测模式集合不同于图19B或图19C中的可用的帧内预测模式集合,但是图19A中的水平方向周围的两个邻近帧内预测模式之间的角度差与图19B或图19C中的水平方向周围的两个邻近帧内预测模式之间的角度差相同。在图19D中,所有角度在水平方向上暴跌(collapse)。

[0267] 为了表示减少的帧内预测模式集合,我们可以针对剩余的帧内模式保持相同的帧内预测模式索引,并且使用角度步长 d 来指示两个邻近模式之间的角度差。例如,赤道处的可用帧内预测方向模式索引可以为 $2, 3, 4, \dots$,其中 $d=1$,并且另一定位处的索引可以为 $2, 5, 8$,其中 $d=3$,而相同的索引对应于相同的帧内预测模式。编码器和解码器仅使用可用帧内预测模式。参数 d 由编码器用来构建MPM列表,并且可以在解码器中计算,或者在比特流中传送。对于具有均匀步长 d 的编解码器,使用单个参数。对于非均匀步长,查找表可以由解码器先验已知,或者经由比特流在例如序列参数集或视频参数集中传送。

[0268] 在另一实施例中,取决于在等矩形投影的特定情况下给定块定位、纬度而设置的新步长,重新组织帧内预测模式索引。用于方向模式的新索引是相连的,并且范围可以从2到 $N-1$,其中 N 现在取决于定位。

[0269] 由于随着角度 a 减小,可用帧内预测方向的数量减小,所以也可以改进帧内方向的编码。如上所述,当帧内预测模式不包括在HEVC中的MPM列表中时,使用5比特固定长度代码对该帧内预测模式进行译码,并且6比特用于JEM。对于给定块,我们提出使固定长度代码的比特数适配为以下:

$$[0270] \quad n = \lceil \log_2(N_a^+ + N_a^- + 1) \rceil. \quad (35)$$

[0271] 为了效率目的,可以在编码器和解码器侧二者处判定方向译码和值,这仅取决于局部帧比例 a 。对于等矩形映射,可以将局部帧比例计算为:

$$[0272] \quad a = \cos(\varphi) \quad (36)$$

[0273] 其中纬度角度

$$[0274] \quad \varphi = \frac{\pi(\frac{h}{2} - y)}{h} \quad (37)$$

[0275] 我们从等式(37)注意到,纬度角度 φ 依赖于“ y ”,即块的垂直定位。随后,局部参考帧比例 a 也取决于垂直定位(参见等式(36)),并且参数 d 和固定长度代码的比特数 n 也依赖于垂直定位(参见等式(34)和等式(35))。

[0276] 为了简化编码以及方向数量与预测块的定位之间的关系,可以构建对应的表,如表7中的示例中所示。

[0277] 表7

[0278]

abs(φ) 值 (rad)	[0, 0.78]	[0.78, 1.26]	[1.26, 1.47]	[1.47, 1.50]	[1.50, $\pi/2$]
对应的像素 行	[0.25*h, 0.75* h]	[0.1*h, 0.25* h] [0.75*h, 0.9* h]	[0.05*h, 0.1* h] [0.9*h, 0.95* h]	[0.02*h, 0.05* h] [0.95*h, 0.98* h]	[0, 0.02*h] [0.98*h, h]
方向的数量	33	16	8	4	2

[0279] 对于 $\text{abs}(\varphi)$ 值 = [0, 0.78] (rad), [0.25*h, 0.75*h] 之间的像素 (其中 “h” 为图片的高度), 所有帧内预测方向模式都可用并且用于块的帧内预测方向模式可以从所有 33 个模式中选择。对于 $\text{abs}(\varphi)$ 值 = [0.78, 1.26] (rad), [0.1*h, 0.25*h] 或 [0.75*h, 0.9*h] 之间的像素, 约帧内预测方向模式的一半 (16 个) 可用, 索引为 0-15。因此, 用于块的帧内预测方向模式只能从这 16 个模式中选择。对于 $\text{abs}(\varphi)$ 值 = [1.26, 1.47] (rad), [0.05*h, 0.1*h] 或 [0.9*h, 0.95*h] 之间的像素, 约帧内预测方向模式的四分之一 (8 个) 可用, 索引为 0-7。

[0280] 图 20 示出了使方向分组和帧内方向的重新排序的示例, 以便于根据可用方向的数量导出模式。作为纬度角度 φ 的函数的方向的数量在表 7 中示出。不是如 HEVC 或 JEM 中那样以连续方式将索引分配给帧内预测方向模式, 我们可以以隔行方式来分配索引。在如表 7 和图 20 中所示的示例中, 针对 HEVC 帧内预测模式的方向来重新组织模式索引。

[0281] 这里在图 20 中, 当只有两个可用模式时, 保持模式 0 和 1。当有四个可用模式时, 保持模式 0-3。当有 8 个可用模式时, 保持模式 0-7。当有 16 个模式时, 保持模式 0-15。因此, 可用帧内预测模式总是通过相连帧内预测模式索引的集合来表示, 而范围可以取决于所保持模式的数量而不同。在模式索引的该布置中, 相同的帧内预测模式索引总是对应于相同的帧内预测方向。

[0282] 图 21A 图示了根据本原理的实施例的使用 MPM 列表对用于当前块的帧内预测模式进行编码或解码的示例性方法 2100。图 21B 图示了根据本原理的实施例的使用固定长度代码对用于当前块的帧内预测模式进行编码或解码的示例性方法 2200。

[0283] 当在编码器侧使用时, 获得 (2110) 用于相邻的左块和上块的帧内预测模式。可以例如使用等式 (34) 或将垂直定位与参数 d 相关联的查找表来计算 (2120) 角度步长 d。随着角度步长 d 改变, 可用帧内预测模式的数量也改变。然后, 可以基于例如如表 6 中针对 JEM 所示的角度步长 d 来调整 (2130) MPM 列表。如果针对编码的测试下的当前帧内预测模式是 MPM 之一, 则确定对应的 MPM 索引, 并且可以计算针对使用当前帧内预测模式的编码成本。可以将 RD 成本与其他预测模式进行比较, 以判定是否要为当前块选择当前帧内预测模式。

[0284] 如果测试下的当前帧内预测模式不是 MPM 之一, 则编码器可以计算 (2260) 可用帧内预测模式的数量和固定长度代码的比特数, 例如, 基于等式 (35)。然后, 编码器可以使用所计算的比特数对帧内预测模式进行编码 (2270), 并且编码器基于相关联的 RD 成本来判定是否要使用当前帧内预测模式。

[0285] 当在解码器侧使用时, 解码器确定帧内 MPM 是否用于当前块。如果解码器确定 MPM 被使用, 则解码器执行步骤 2110-2130, 类似于针对编码器所进行的。然后, 解码器可以基于 MPM 索引对帧内预测模式进行解码。如果解码器确定固定长度代码用于帧内预测模式, 则解

码器可以计算 (2260) 固定长度代码的比特数, 并且基于固定长度代码的比特数来对帧内预测模式进行解码 (2270)。

[0286] 所提出的方法依赖于基于已知的几何失真从相邻块导出帧内方向传播模式的一般原理。

[0287] 例如, 取决于视频的类型、所考虑的预测块的位置和大小, 可以将二次曲线添加为另一帧内模式。在以广角/短焦距拍摄视频的另一示例中, 一些失真可能使直线形变。有时使用圆柱形校正来减少失真。在视频失真的情况下, 弯曲的帧内预测通常可以通过使参考像素沿着形变曲线传播来提高压缩效率, 尤其是在边界上。当然, 直的帧内预测被保持为针对给定块表面的良好估计。在两种情况下, 使MPM推导适配于已知失真提高了其性能。

[0288] 图22图示了在严重失真的情况下直的帧内模式如何在示例性帧上变化。在针对等矩形映射的上文中, 我们描述了在垂直方向上使帧内预测适配。这里应当注意其他类型的视频, 例如, 短焦距视频, 因为可能存在距透镜中心的径向变化, 也可以在水平方向上使帧内预测适配。

[0289] 对于短焦距视频, 针对失真的传统模型要使用Brown-Conrady模型, 其中2D+3D失真被合并成简单的2D函数: $(x_d, y_d) = g(x_u, y_u, S)$ 其中 (x_d, y_d) 是失真之后的像素坐标 (帧F中), 并且 (x_u, y_u) 是失真之前的像素坐标 (帧G中)。失真函数 $g()$ 是组合: $g = f^{-1} \circ 3d^{-1} \circ \text{proj}^{-1}$ 。可以使用Brown-Conrady模型来获得失真函数 $g()$ 。

[0290] 在图22中, 靠近帧的边界, 方向模式从块到块变化, 因此需要使从相邻块导出的MPM适配。因此, 考虑到该新的传递函数 g , 可以应用与所提出的对传统MPM处理的修改相关的上述原理。

[0291] 在图22中, 也可以注意到, 可用方向的数量可能从一个定位到另一个定位不同。在极端定位处, 连贯的失真方向之间的角度步长可能太小而无法考虑。因此, 可以应用关于减少方向模式的数量并因此减少语法比特的上述原理。

[0292] 对于非正方形立方体映射, 局部参考帧在大部分图像中变得非规格化正交。图23图示了映射在矩形帧上的示例性非正方形立方体。在该情况下, 分辨率取决于立方体面。

[0293] 随着可用方向的数量改变, 即使所有方向都可用 (在编解码器中没有改变), 也仅在编码器侧测试可用方向。在该情况下, 通过改变 $N^+(a)/N^-(a)$ 和 $Z^+(a)/Z^-(a)$ 来使方向的数量相应地适配。

[0294] 为了使用该新的帧内方向列表处理, 可以设置标志:

[0295] - 在PPS (图片参数集) 级指示该工具是否可用。该标志可能与用于全向视频的新MPM过程相同。

[0296] - 在块 (CTU/CU/PU) 级指示帧内方向处理的策略。如果该标志被设置为1, 则帧内方向的数量减少, 如上所述。

[0297] 基于映射在编码器和解码器侧二者处计算可用帧内预测方向的列表。

[0298] 可用方向的数量也可以被传送到解码器, 因此独立于图像的映射类型。

[0299] - 在PPS级指示可用于当前帧的帧内方向的数量。可以每行或每列或者每个图像区域 (切片、瓦片 (tile) 等) 使可能的方向的数量索引化。

[0300] - 在CTU/CU/PU级指示可用于所考虑的块 (可能使用差分/上下文译码来译码, 考虑周围块) 的帧内方向的数量。

[0301] 在视频编码和解码期间关于用于帧内预测的MPM讨论了各种实施例。例如,我们可以使MPM对准,如表3中所示,或者减少帧内预测方向模式的数量,如图19中所示。这些不同的实施例可以单独使用或组合使用。

[0302] 图24图示了其中可以实现本原理的示例性实施例的各个方面的示例性系统2400的框图。系统2400可以实施为包括下面描述的各种组件并且被配置为执行上述处理的设备。这样的设备的示例包括但不限于HMD、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、数字电视接收器、个人视频记录系统、连接的家用电器和服务。系统2400可以包括传感器,并且可以经由如图24中所示并且如本领域技术人员已知的通信信道通信地耦接到其他类似系统,以实现上述示例性视频系统。

[0303] 系统2400可以包括至少一个处理器2410,其被配置为执行加载在其中的指令,以用于实现如上所述的各种处理。处理器2410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他电路。系统2400也可以包括至少一个存储器2420(例如,易失性存储器设备,非易失性存储器设备)。系统2400可以另外包括储存设备2440,其可以包括非易失性存储器,包括但不限于EEPROM、ROM、PROM、RAM、DRAM、SRAM、闪速存储器、磁盘驱动器和/或光盘驱动器。作为非限制性示例,储存设备2440可以包括内部储存设备、附接储存设备和/或网络可访问储存设备。系统2400也可以包括编码器/解码器模块2430,其被配置为处理数据以提供经编码视频或经解码视频。

[0304] 编码器/解码器模块2430表示可以包括在设备中以执行编码和/或解码功能的模块。编码器500和解码器700可以用在编码器/解码器模块2430中。如已知的那样,设备可以包括编码和解码模块中的一个或二者。另外,编码器/解码器模块2430可以实现为系统2400的单独元件,或者可以作为硬件和软件的组合并入处理器2410内,本领域技术人员已知的那样。

[0305] 系统2400还可以包括显示器(2490)或者可以经由通信信道通信地耦接到显示器。显示器例如是OLED或LCD类型。显示器也可以是沉浸式(投影式)墙壁,其通常有巨大的尺寸。

[0306] 系统2400还可以包括触摸表面2480(例如触摸板或触觉屏幕)和相机2470。处理器2410可以处理从传感器接收的信号,传感器可以是或者可以不是系统2400的部分。来自传感器的一些测量可以用于计算系统2400或连接到系统2400的另一设备的姿势。相机2470可以捕获环境的图像以用于图像处理。处理器2410也可以执行如图1中描述的预处理和后处理功能。

[0307] 要加载到处理器2410上以执行上文描述的各种处理的程序代码可以存储在储存设备2440中,并且随后加载到存储器2420上以供处理器2410执行。根据本原理的示例性实施例,一个或多个处理器2410、存储器2420、储存设备2440和编码器/解码器模块2430可以在执行上文讨论的处理期间存储各种项中的一个或多个,包括但不限于输入视频、比特流、等式、公式、矩阵、变量、运算和运算逻辑。

[0308] 系统2400也可以包括通信接口2450,其使得能够经由通信信道2460与其他设备进行通信。通信接口2450可以包括但不限于被配置为从通信信道2460传送和接收数据的收发器。通信接口可以包括但不限于调制解调器或网卡,并且通信信道可以实现在有线和/或无线介质中。系统2400的各种组件可以使用各种合适的连接(包括但不限于内部总线、布线和

印刷电路板)而连接或通信地耦接在一起。

[0309] 可以通过由处理器2410实现的计算机软件、通过硬件或者通过硬件和软件的组合来执行根据本原理的示例性实施例。作为非限制性示例,可以通过一个或多个集成电路来实现根据本原理的示例性实施例。存储器2420可以是适合于技术环境的任何类型,并且可以使用任何适当的数据储存技术来实现,诸如作为非限制性示例的光存储器设备、磁存储器设备、基于半导体的存储器设备、固定存储器和可移动存储器。处理器2410可以是适合于技术环境的任何类型,并且可以涵盖作为非限制性示例的微处理器、通用计算机、专用计算机和基于多核架构的处理器中的一个或多个。

[0310] 以上描述了各种方法,并且每个方法包括用于实现所描述的方法的一个或多个步骤或动作。除非为了方法的正确操作需要特定顺序的步骤或动作,否则可以修改或组合特定步骤和/或动作的顺序和/或使用。

[0311] 在本申请中使用各种数值,例如,用于帧内预测的角度和模式索引。应当注意,特定值针对示例性目的,并且本原理不限于这些特定值。

[0312] 在此描述的实现方式可以例如以方法或处理、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背景下进行了讨论(例如,仅作为方法进行了讨论),但是所讨论的特征的实现方式也可以以其他形式(例如,装置或程序)来实现。装置例如可以实现在适当的硬件、软件和固件中。方法例如可以实现在诸如例如处理器(其通常涉及处理设备,包括例如计算机、微处理器、集成电路或可编程逻辑器件)之类的装置中。处理器还包括通信设备,诸如例如计算机、蜂窝电话、便携式/个人数字助理(“PDA”),以及便于最终用户之间的信息通信的其他设备。

[0313] 对本原理的“一个实施例”或“实施例”或“一个实现方式”或“实现方式”的引用以及其他变型意指结合实施例描述的特定特征、结构、特性等包括在本原理的至少一个实施例中。因此,在整个说明书中出现在各处的短语“在一个实施例中”或“在实施例中”或“在一个实现方式中”或“在实现方式中”以及任何其他变型的出现不一定都涉及相同的实施例。

[0314] 另外,本申请或其权利要求可以涉及“确定”各种信息。确定信息可以包括例如估计信息、计算信息、预测信息或从存储器取回信息中的一个或多个。

[0315] 此外,本申请或其权利要求可以涉及“访问”各种信息。访问信息可以包括例如接收信息、(例如,从存储器)取回信息、存储信息、处理信息、传送信息、移动信息、复制信息、擦除信息、计算信息、确定信息、预测信息或估计信息中的一个或多个。

[0316] 另外,本申请或其权利要求可以涉及“接收”各种信息。与“访问”一样,接收旨在为广义术语。接收信息可以包括例如访问信息或(例如,从存储器)取回信息中的一个或多个。此外,在诸如例如存储信息、处理信息、传送信息、移动信息、复制信息、擦除信息、计算信息、确定信息、预测信息或估计信息的操作期间,典型地以某种方式涉及“接收”。

[0317] 如对于本领域技术人员将明显的那样,实现方式可以产生被格式化为携带例如可以被存储或传送的信息的各种信号。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,可以将信号格式化为携带所描述的实施例的比特流。可以将这样的信号格式化为例如电磁波(例如,使用频谱的射频部分)或者基带信号。格式化可以包括例如对数据流进行编码以及以经编码数据流来调制载波。信号携带的信息例如可

以是模拟或数字信息。如已知的那样,可以通过各种不同的有线或无线链路来传送信号。信号可以存储在处理器可读介质上。

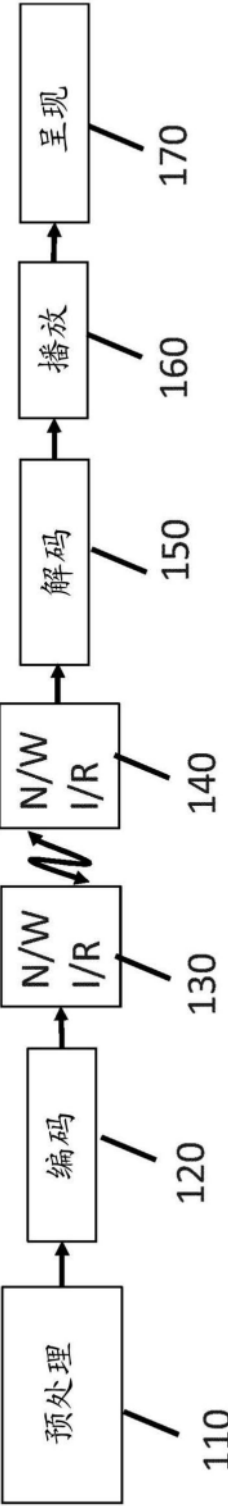


图1

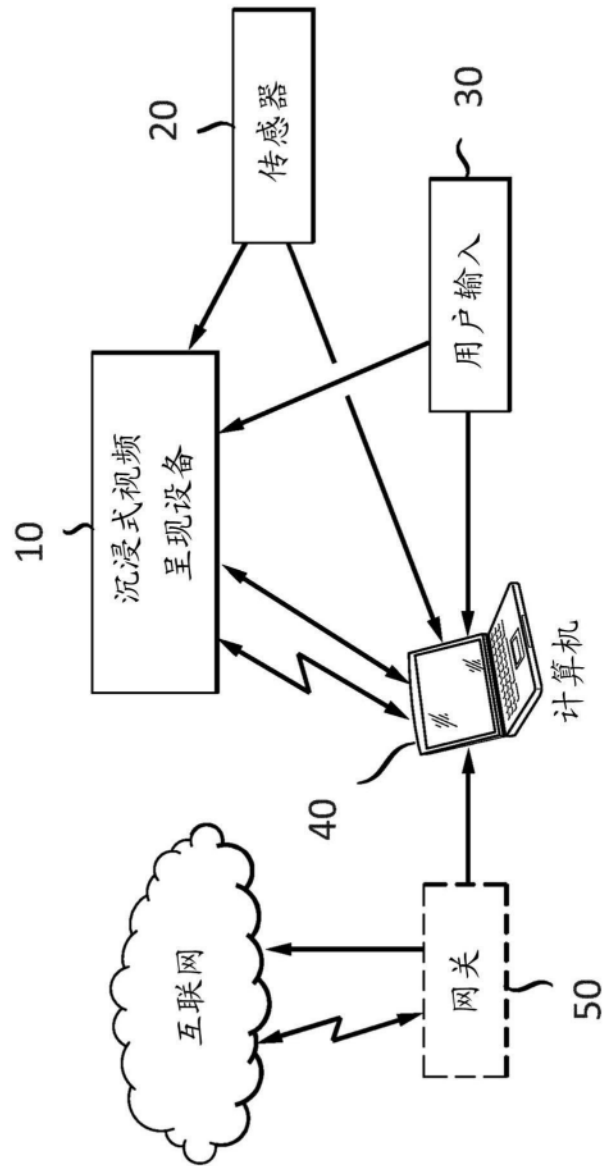


图25

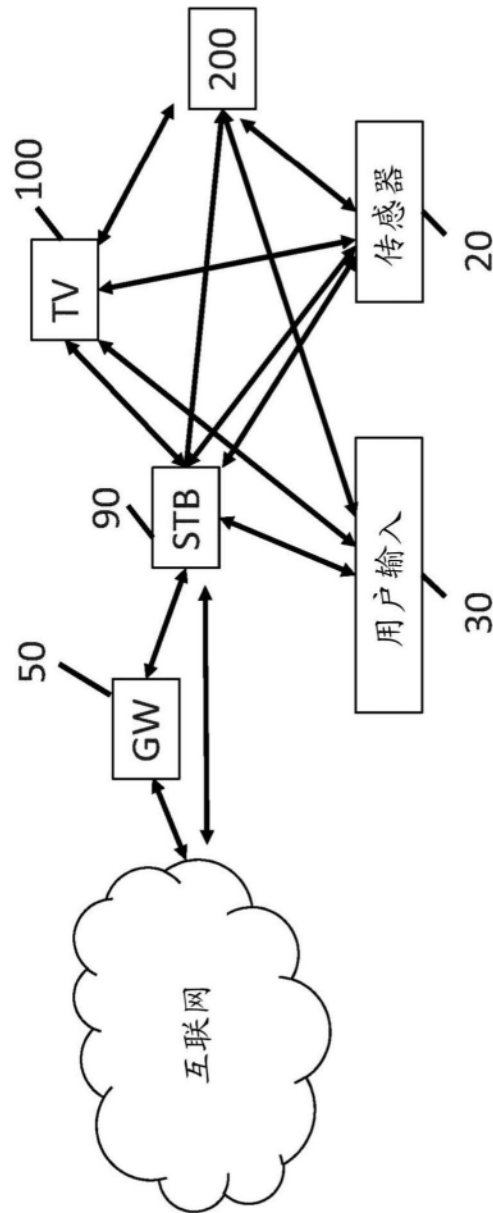


图26

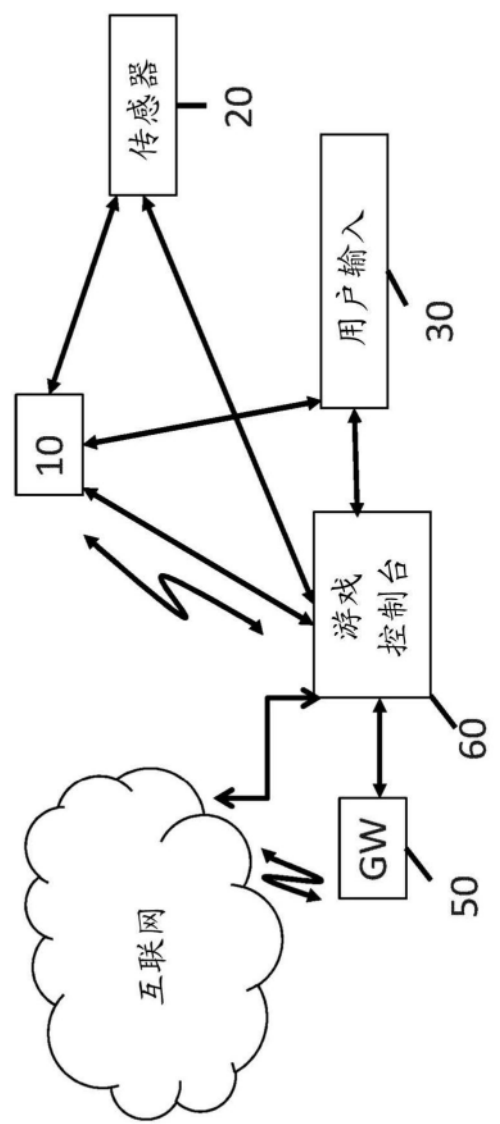


图27

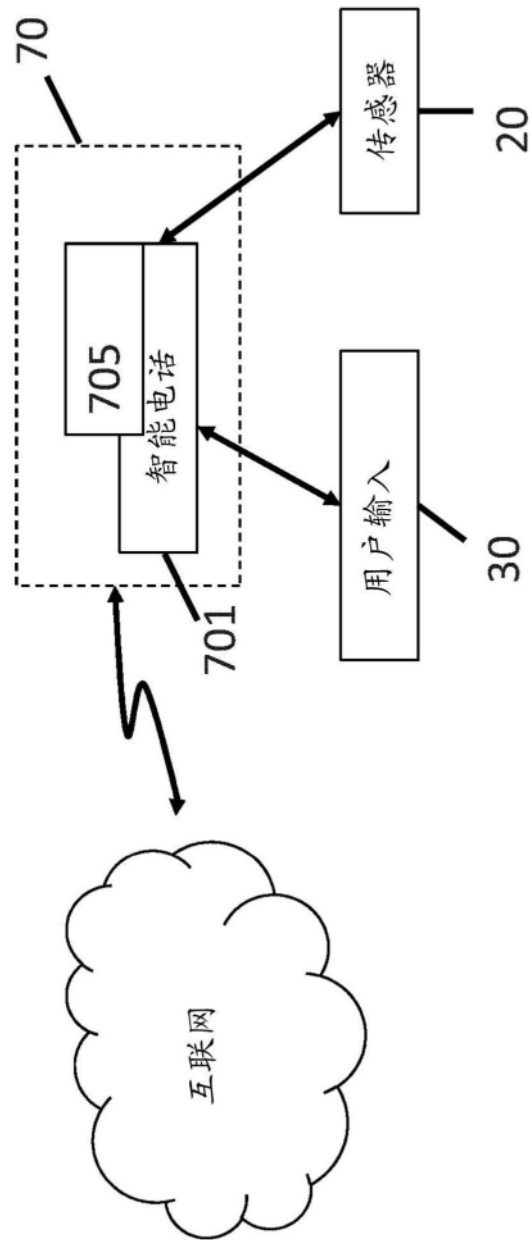


图28

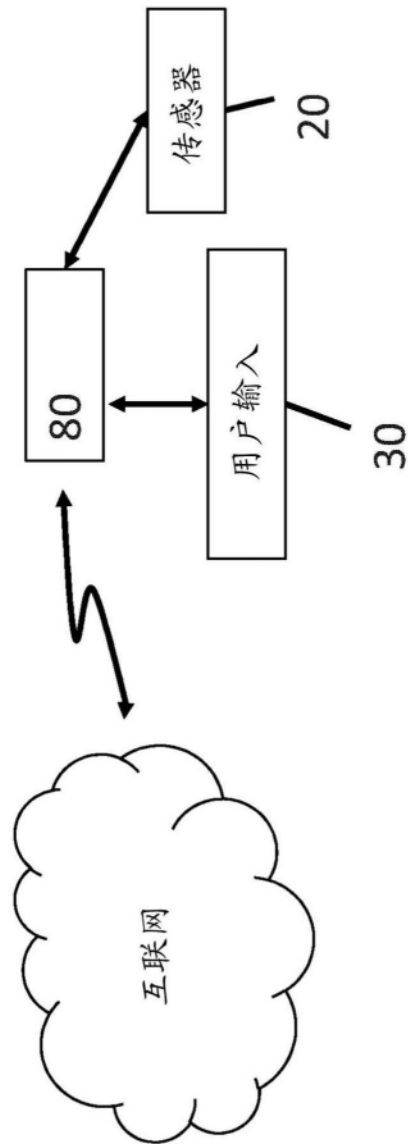


图29

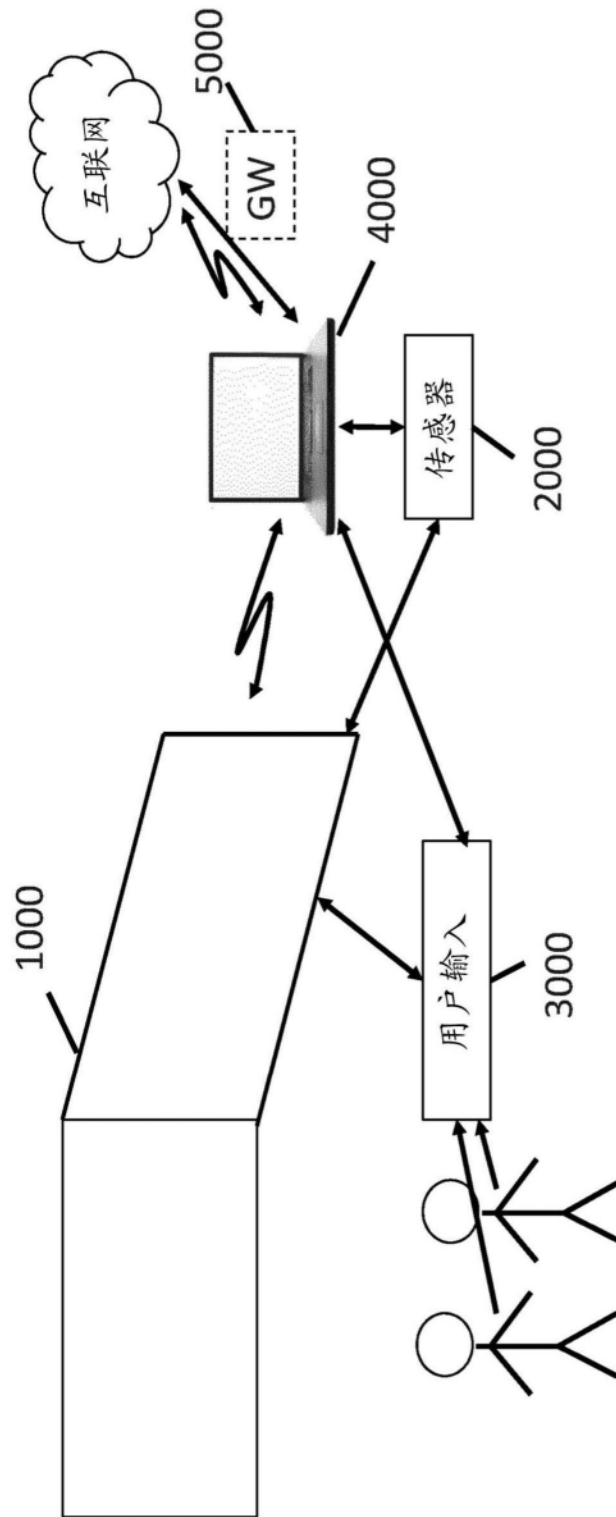


图30

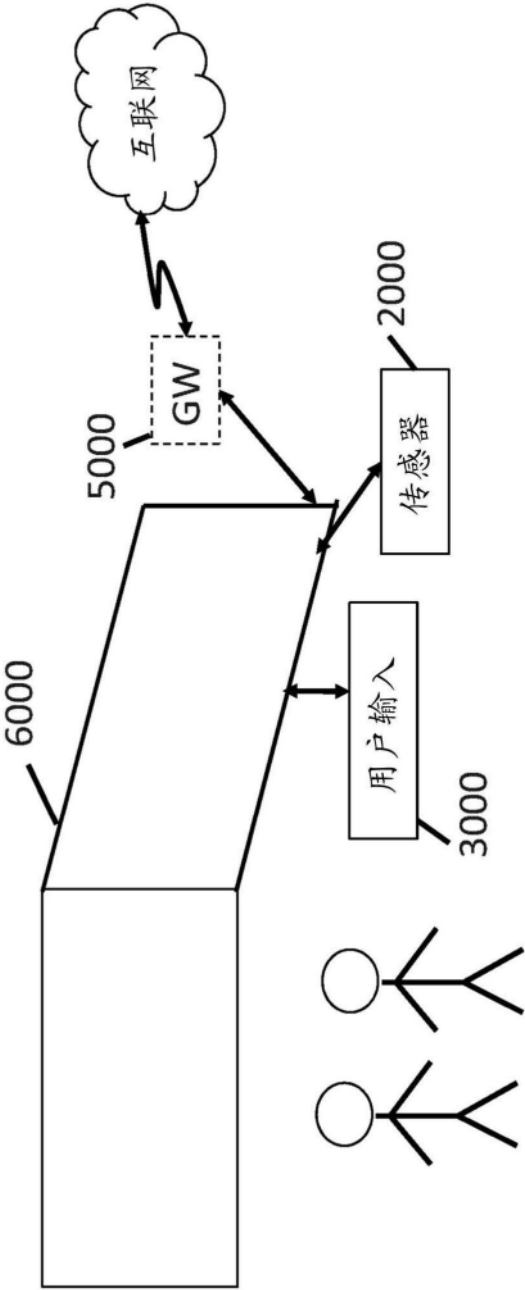


图31

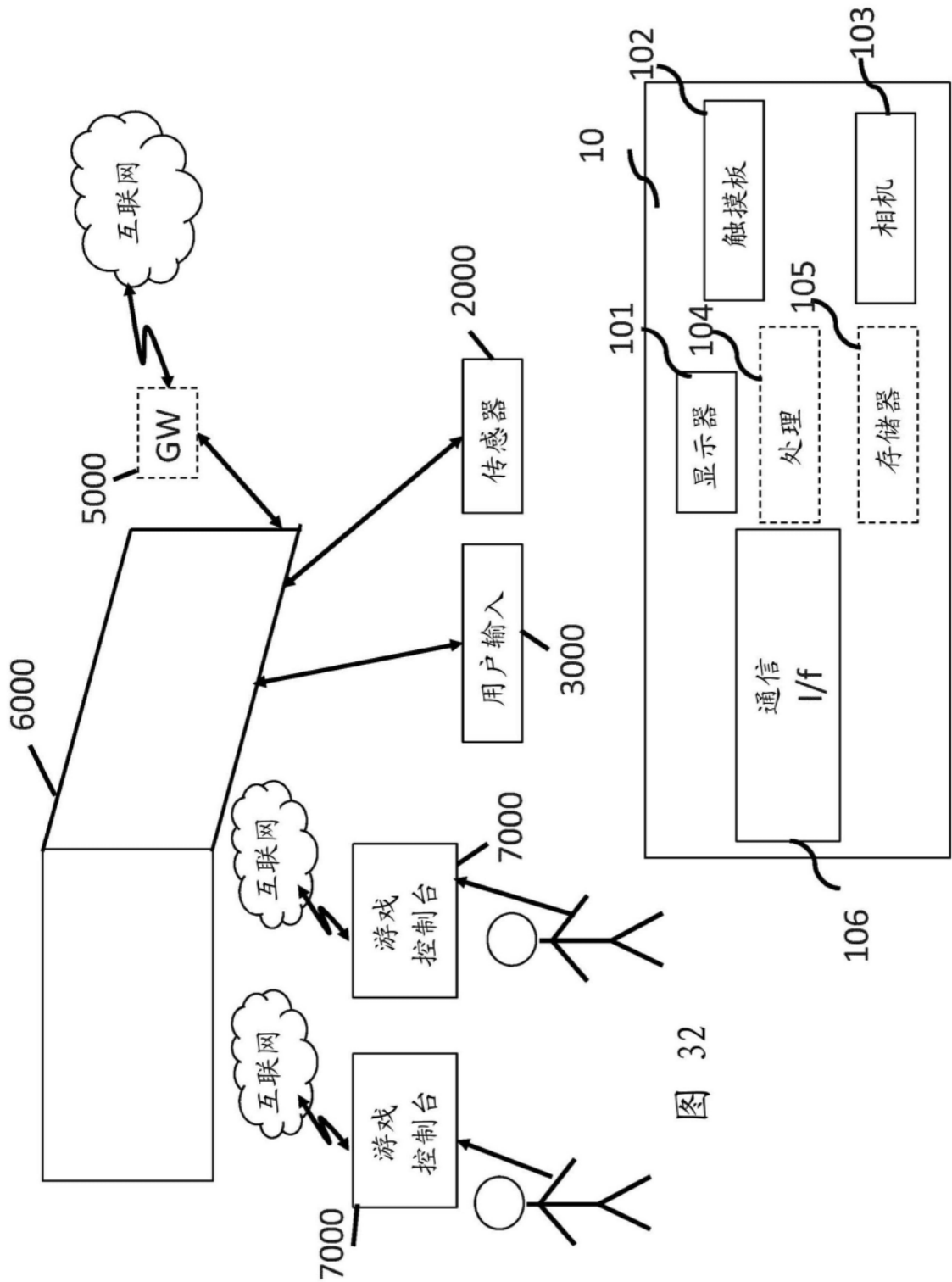


图 32

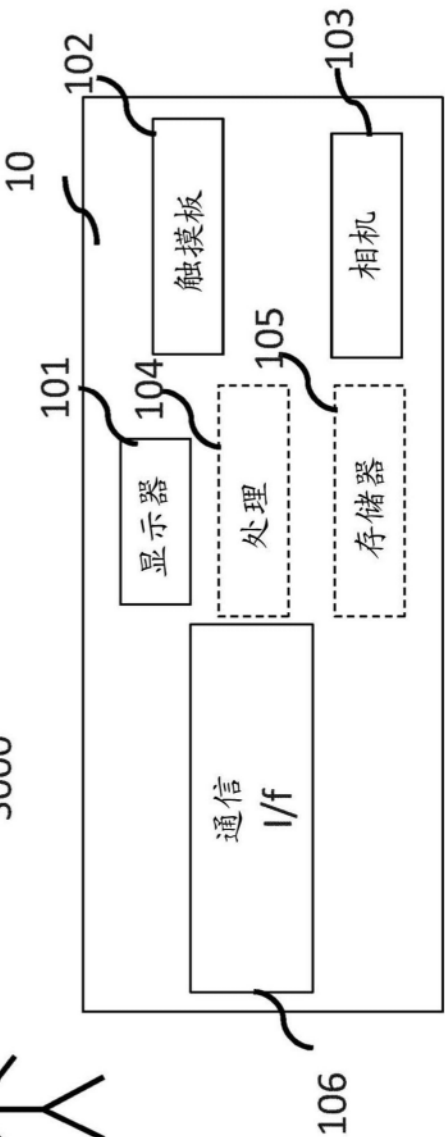


图 33

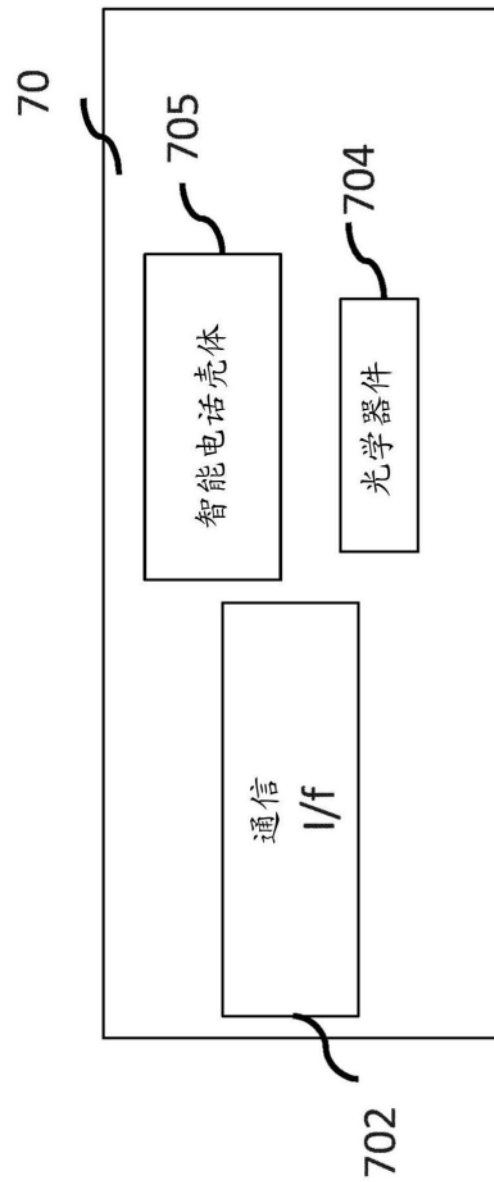


图34

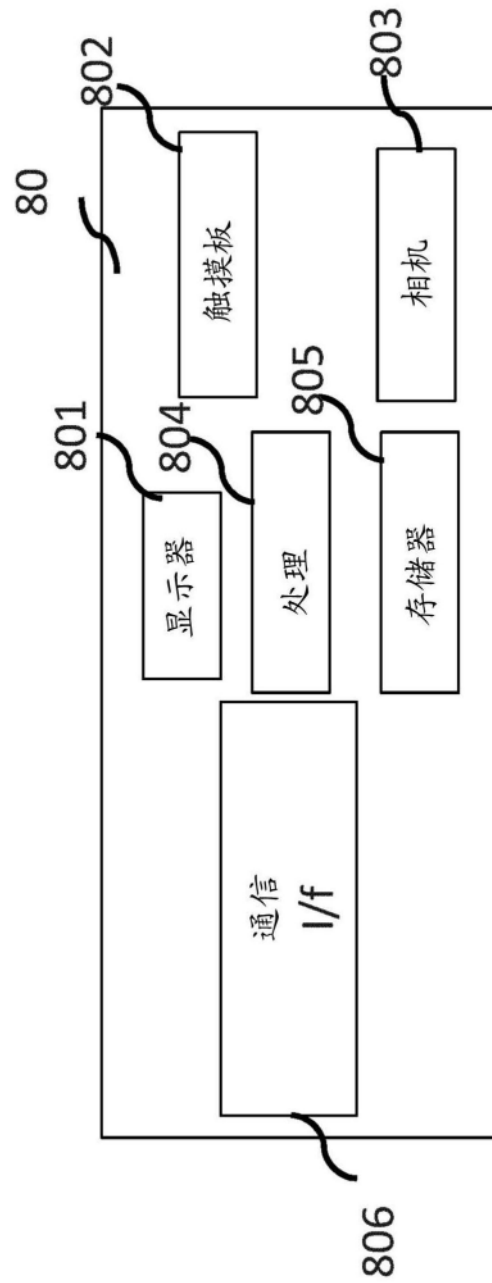


图35

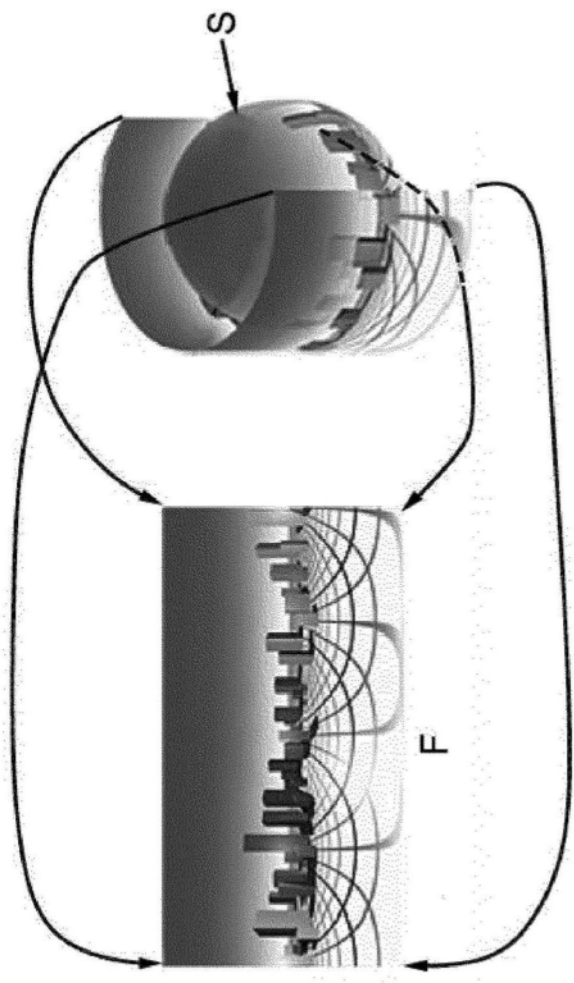


图2A

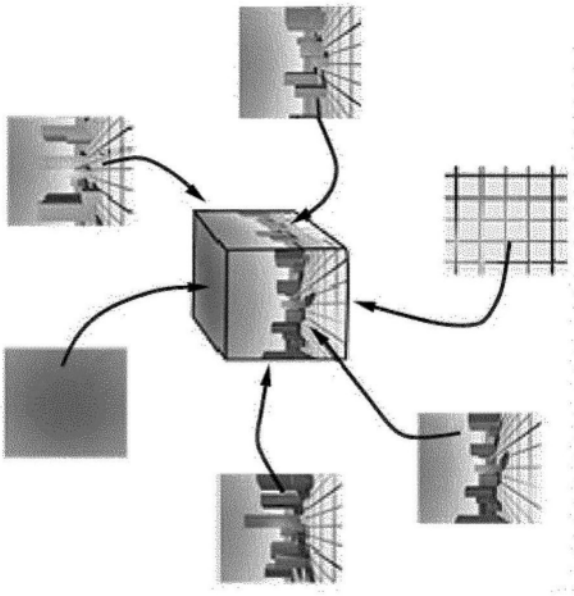


图2B

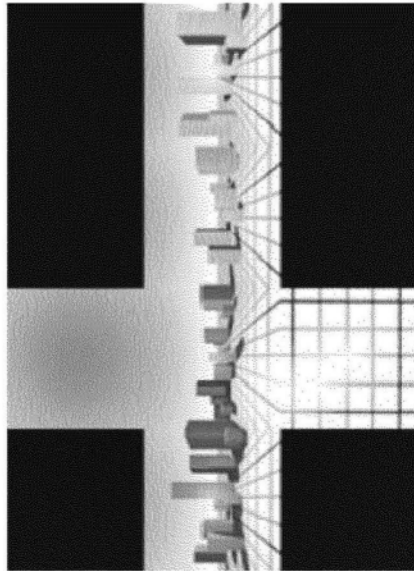


图2C

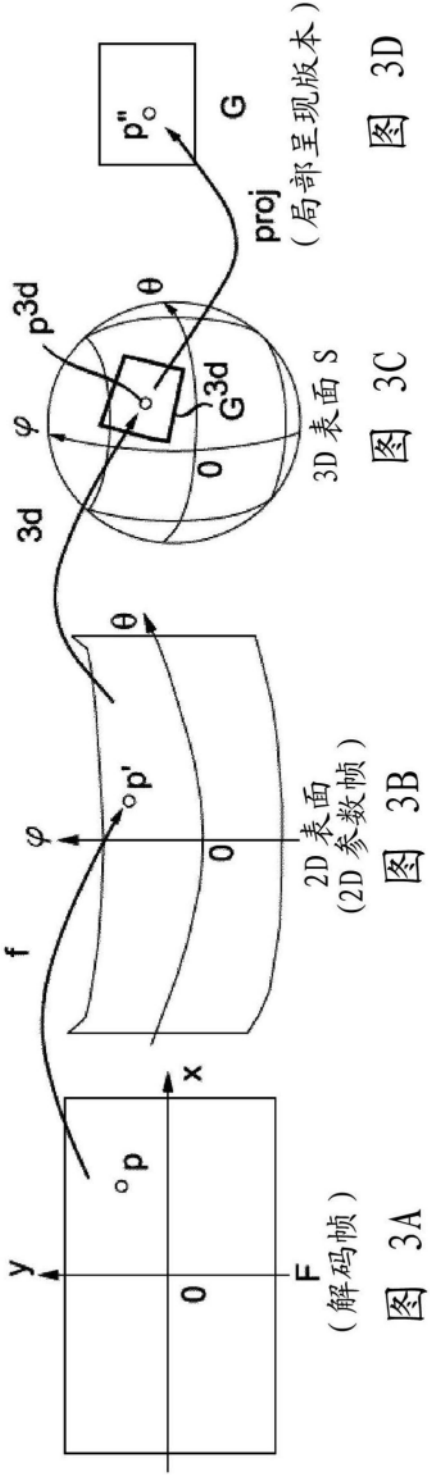


图 3A

图 3B

图 3C

图 3D

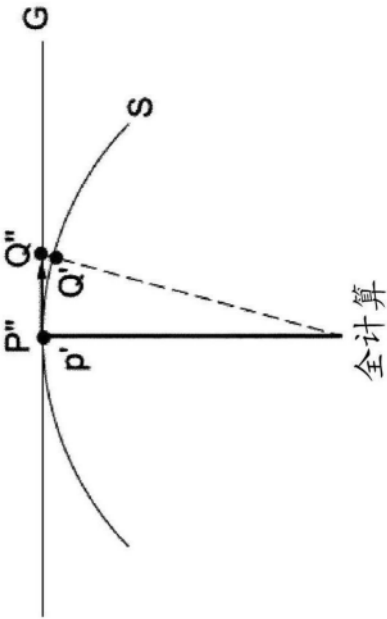


图4A

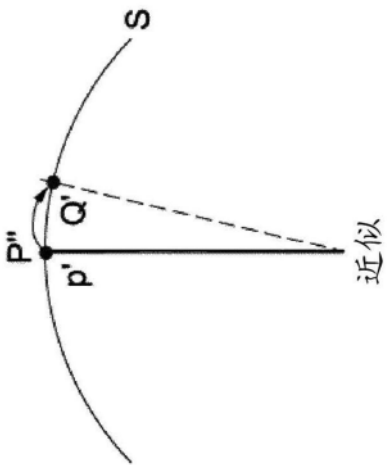


图4B

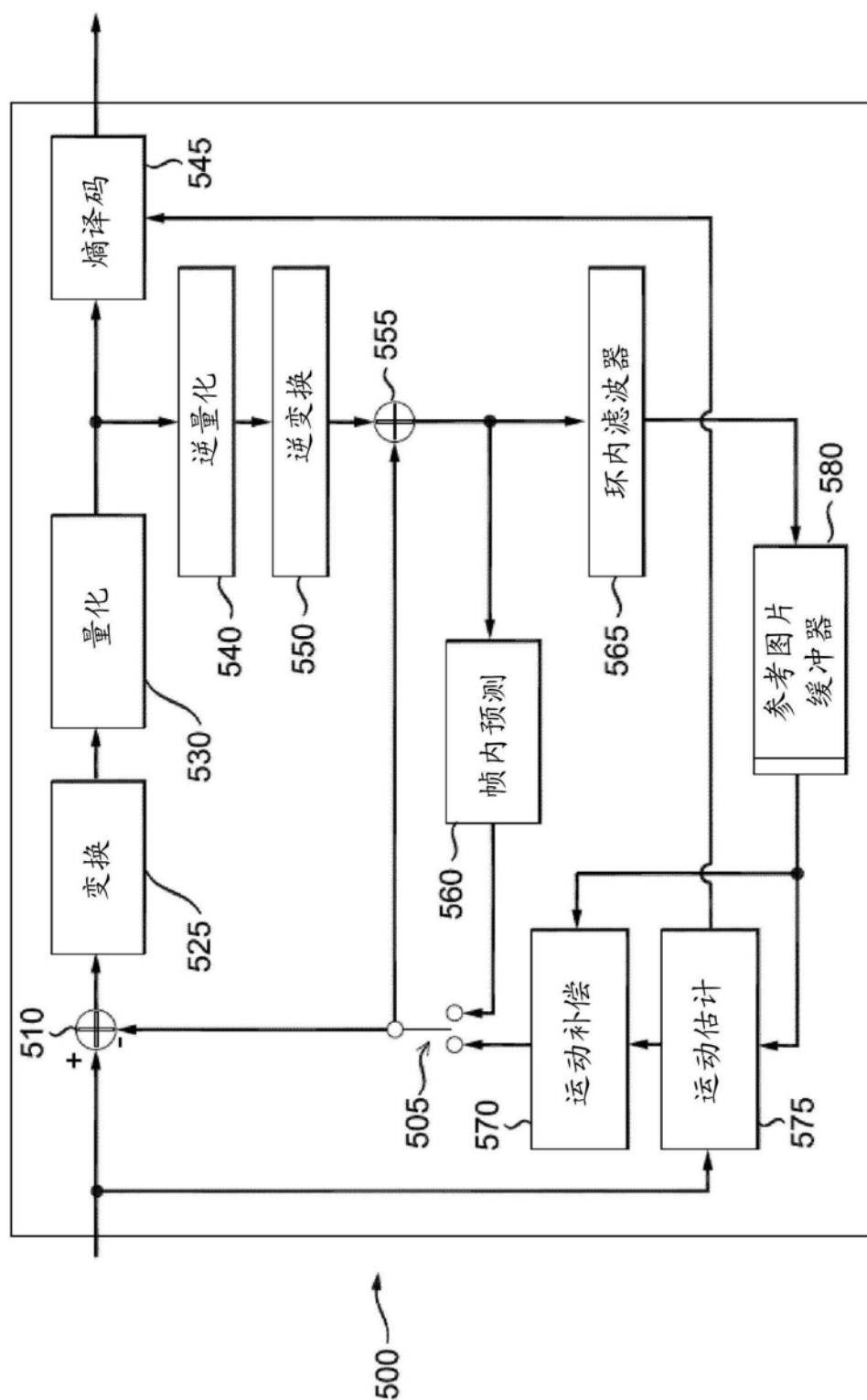


图5

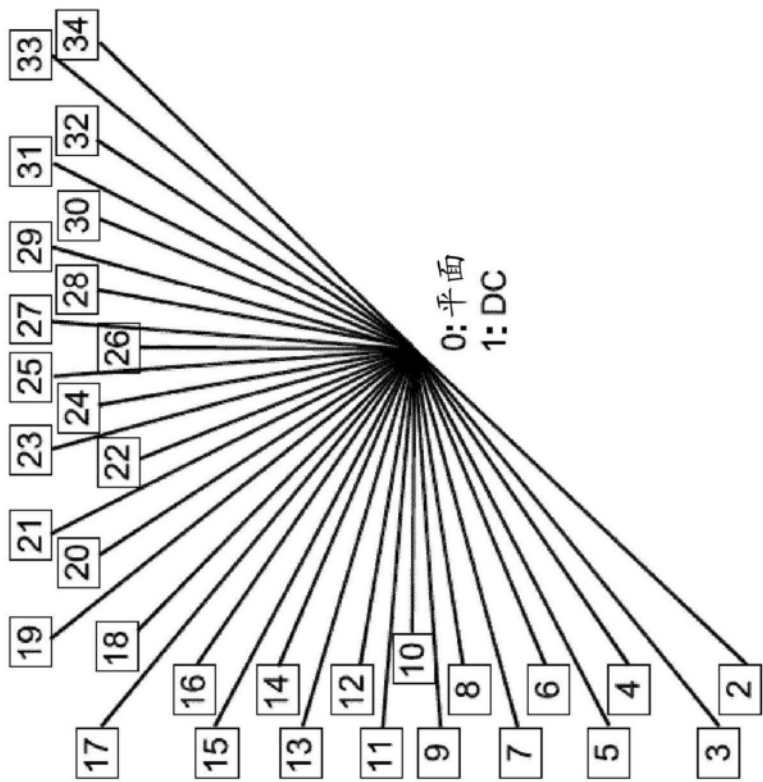


图6A

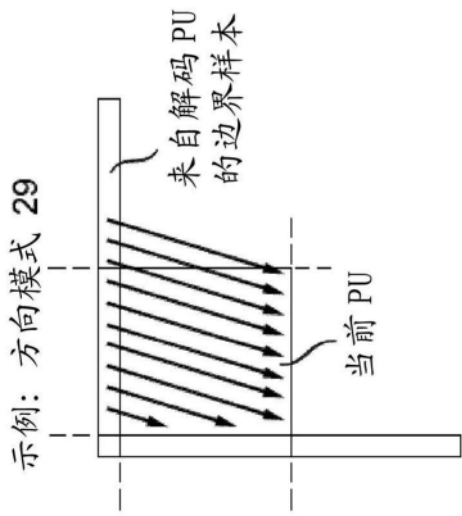


图6B

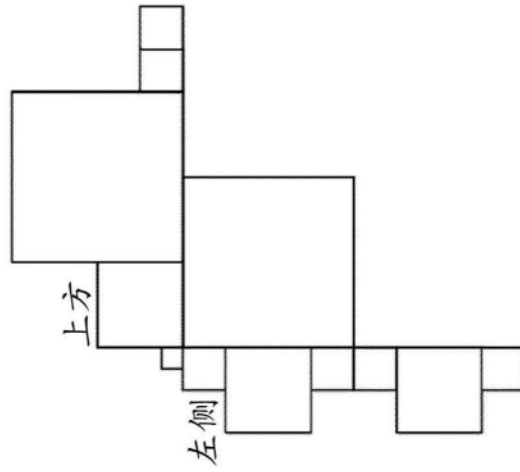


图6C

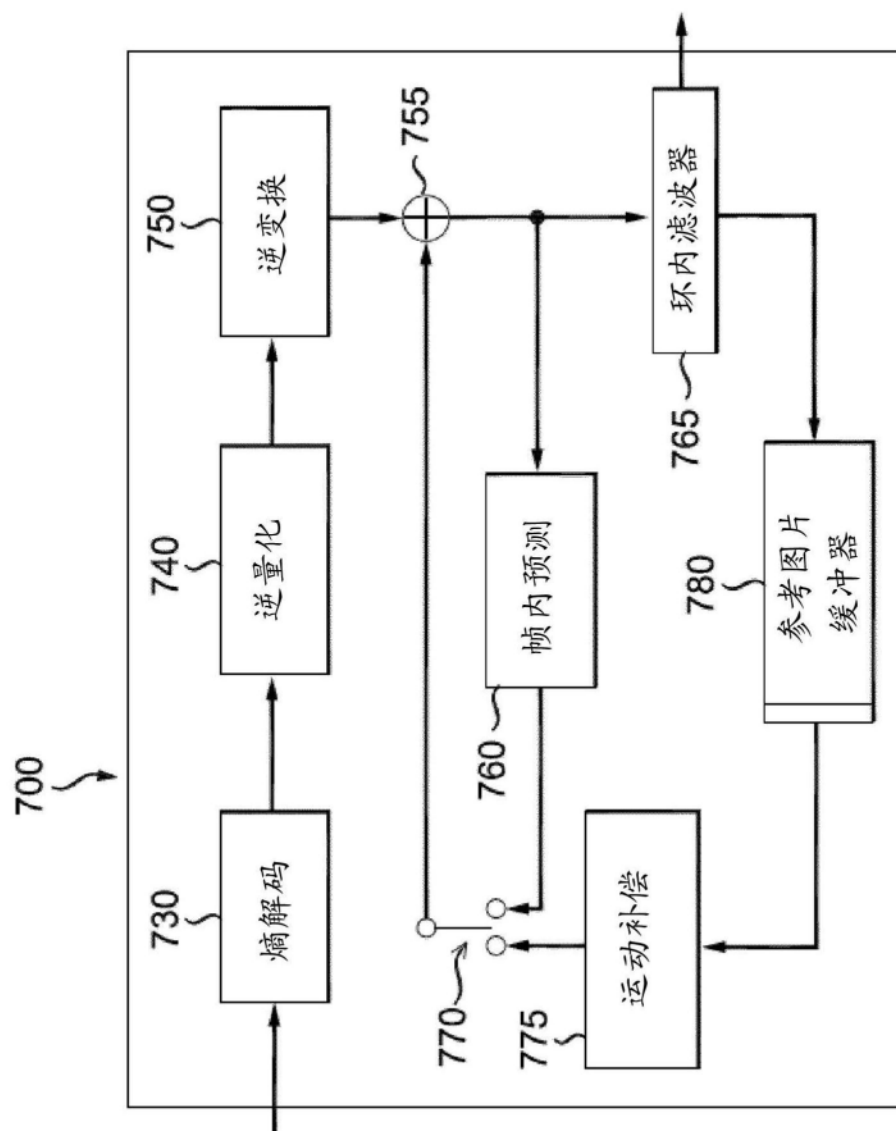


图7

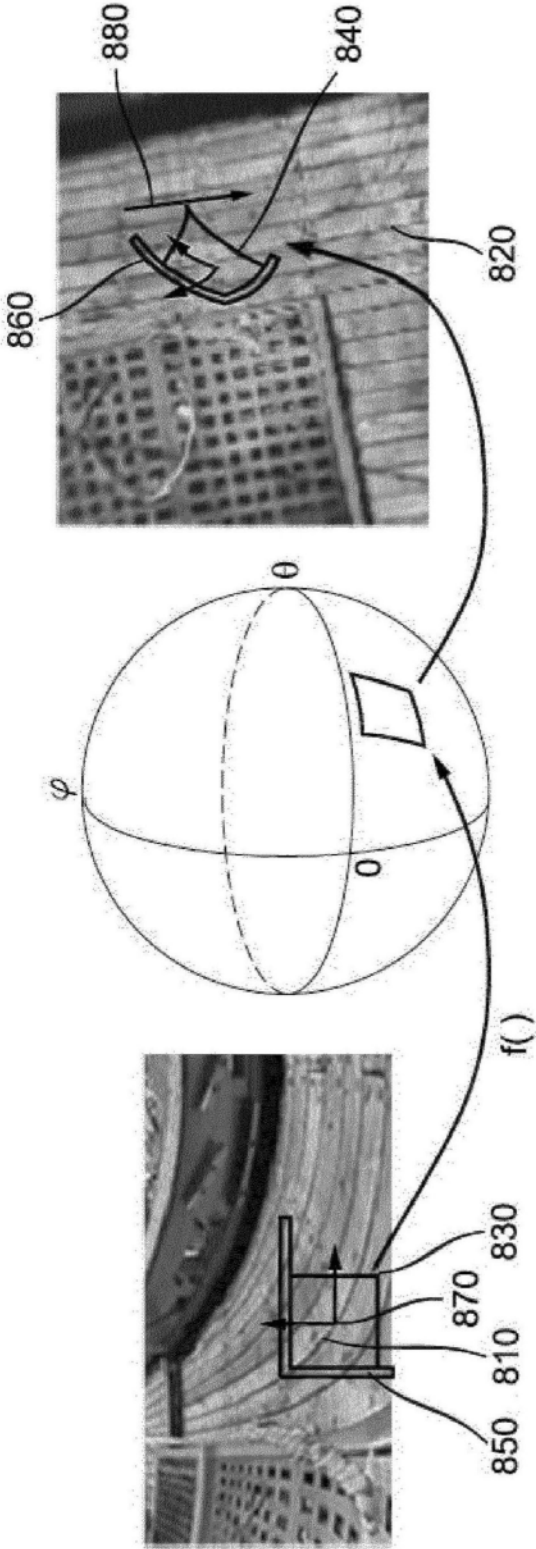


图8

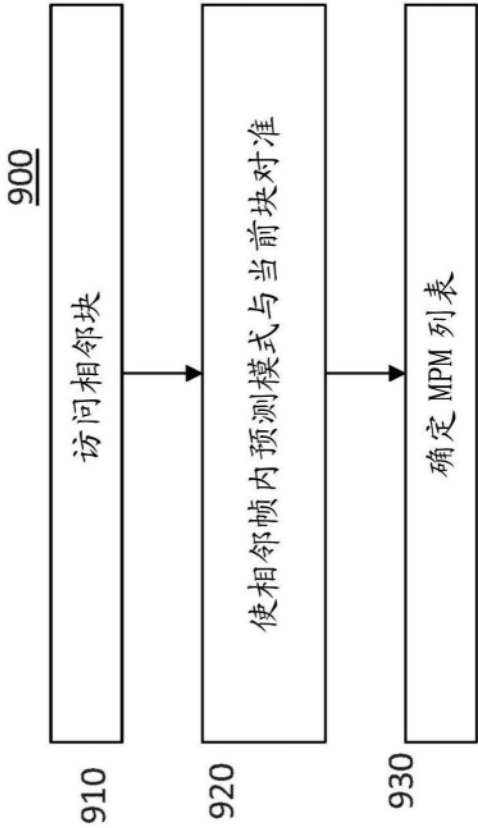


图9

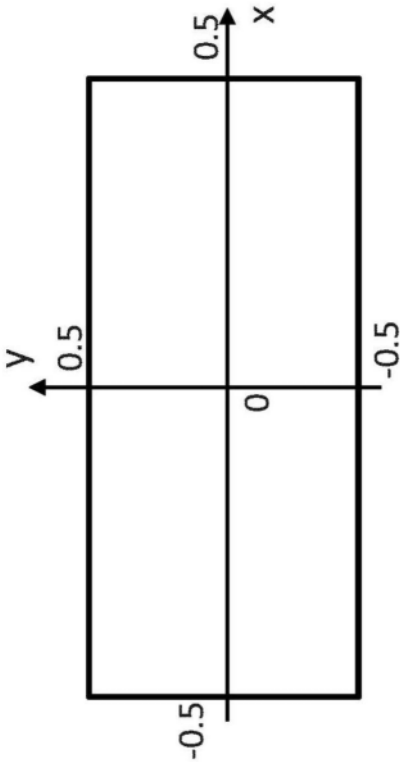


图10A

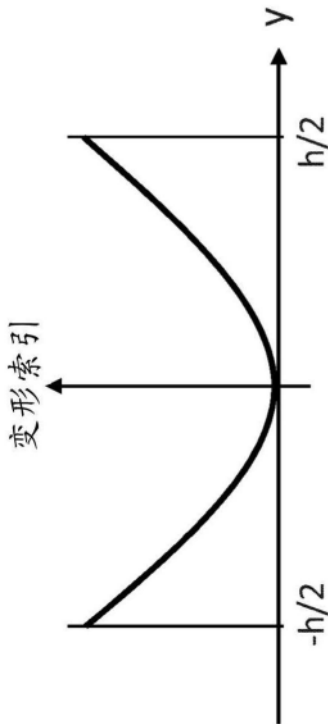


图10B

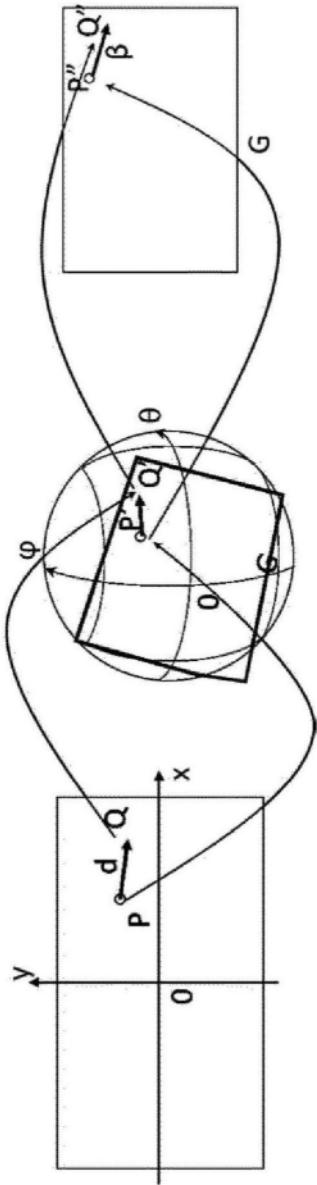


图11A

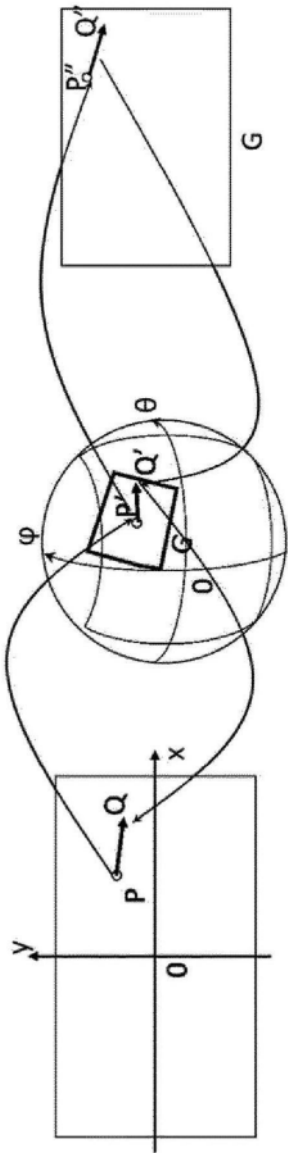


图11B

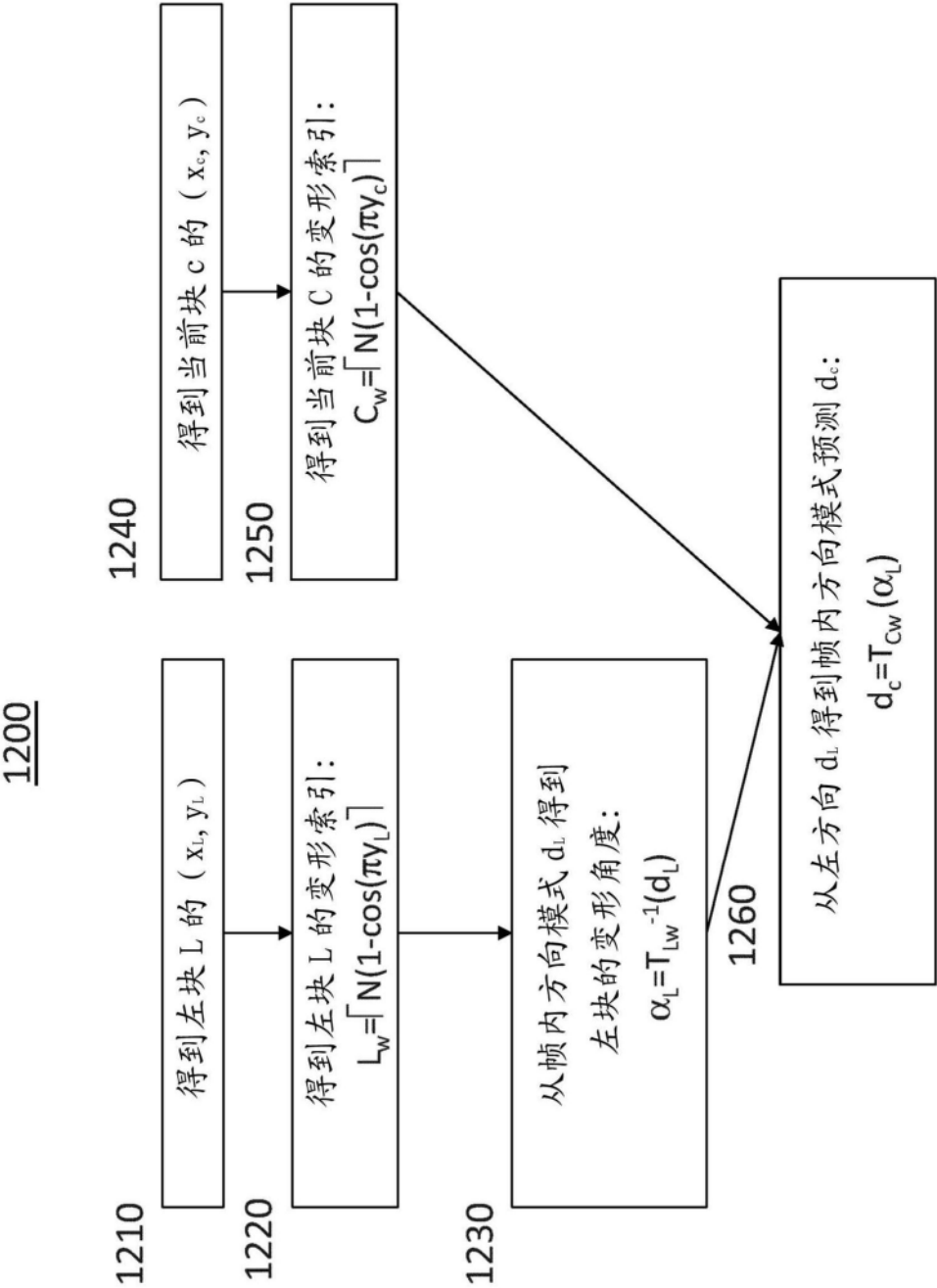


图12

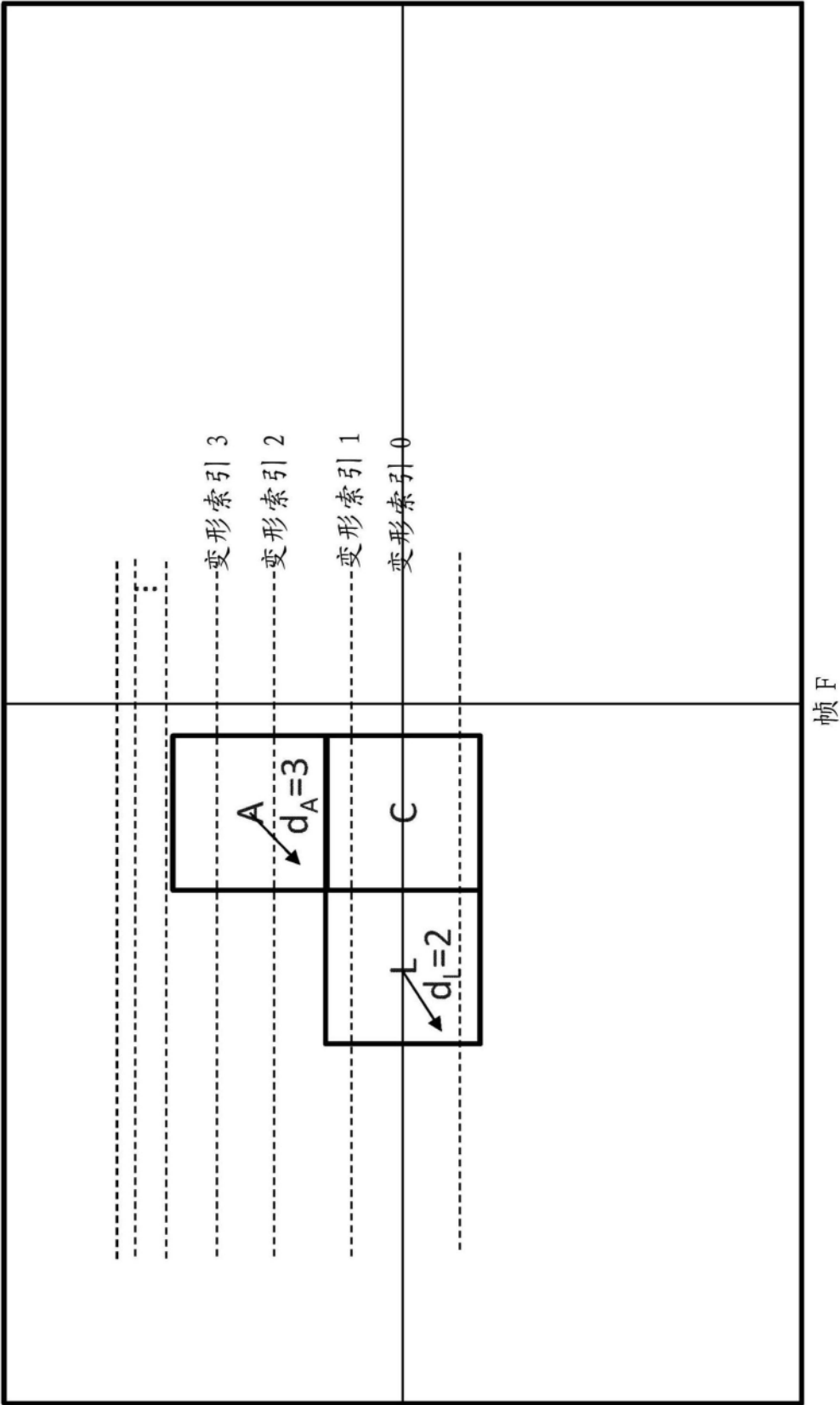


图13

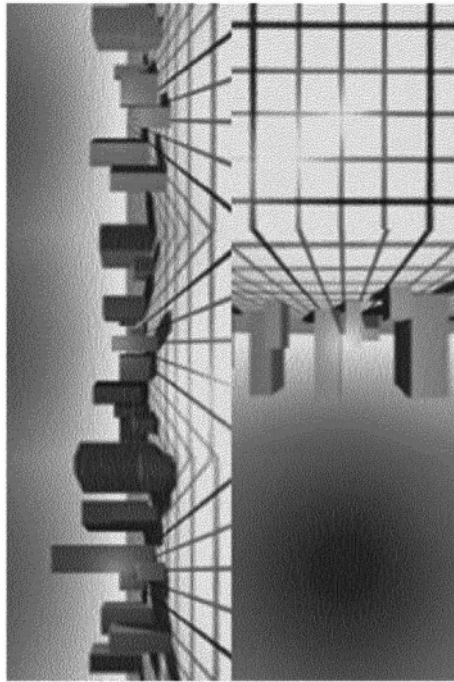


图14A

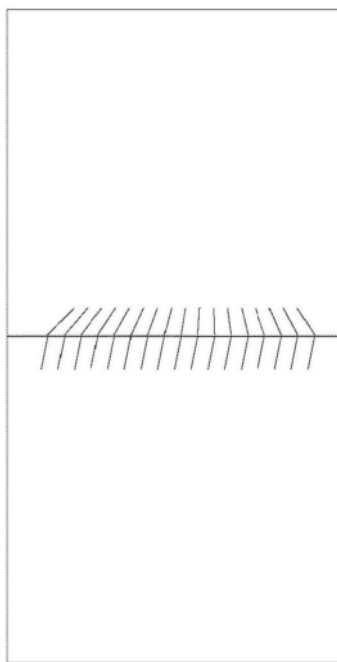


图14B

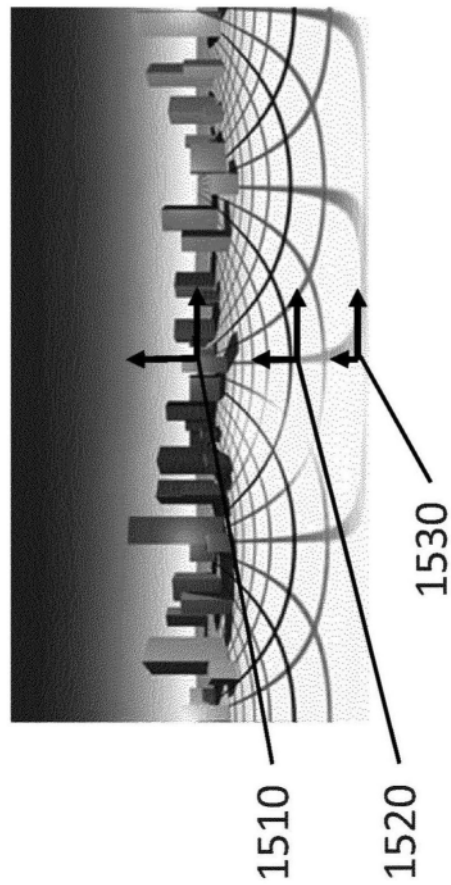


图15

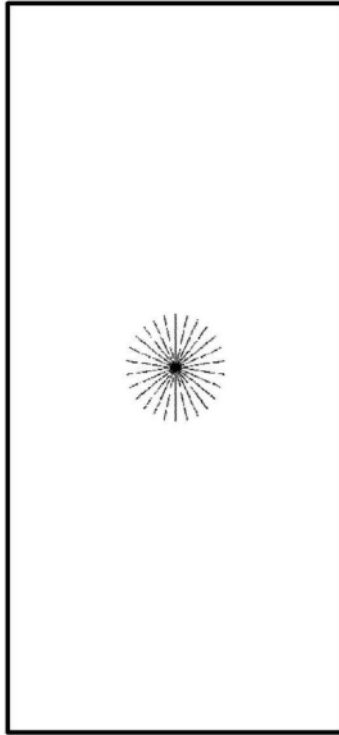


图16A

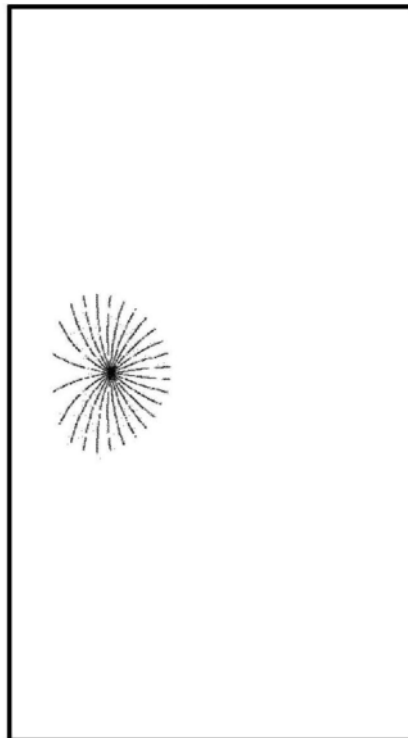


图16B

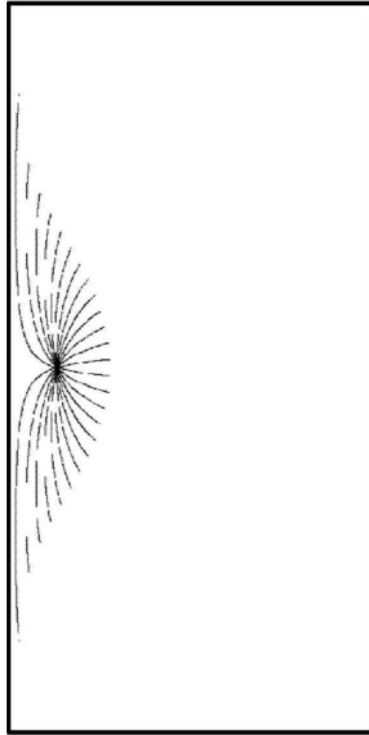


图16C

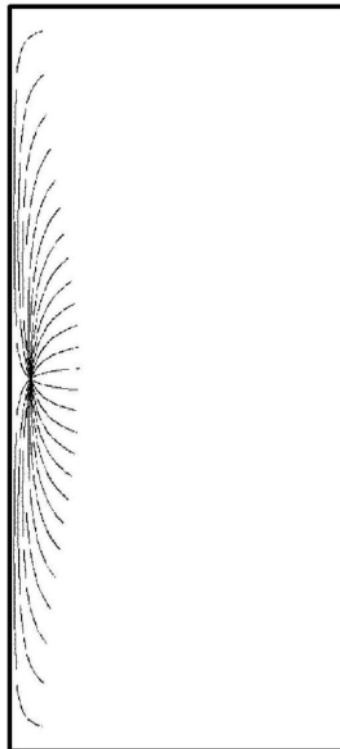


图16D

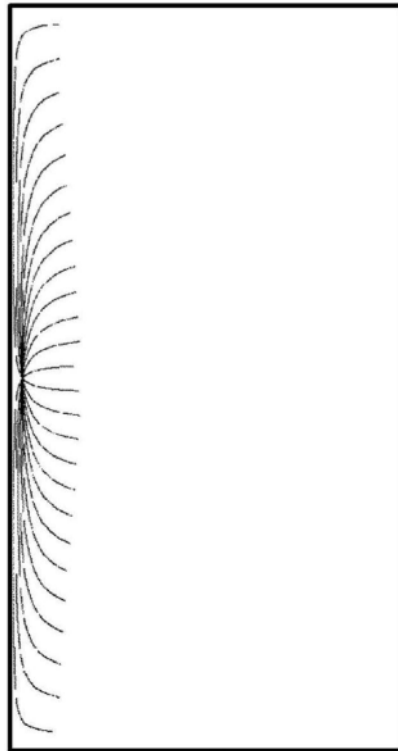


图16E

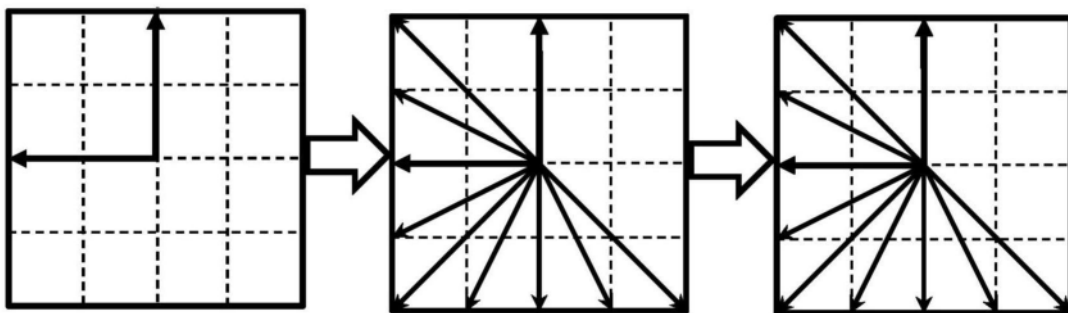


图19A

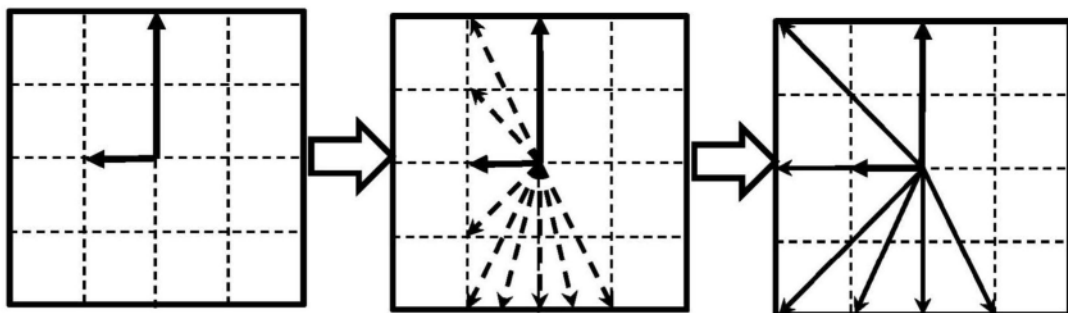


图19B

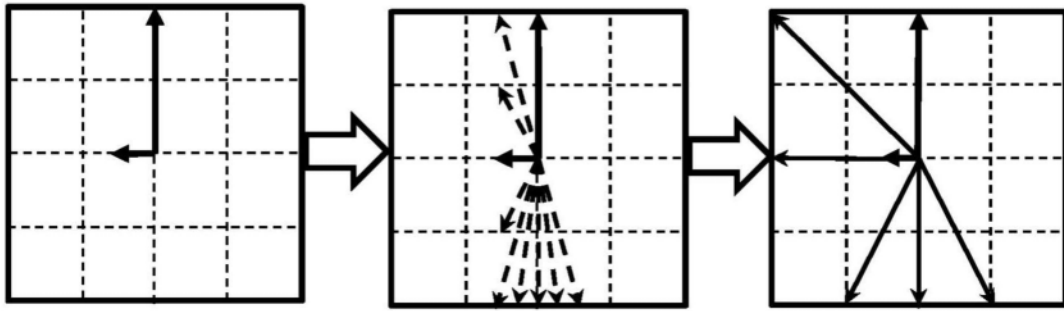


图19C

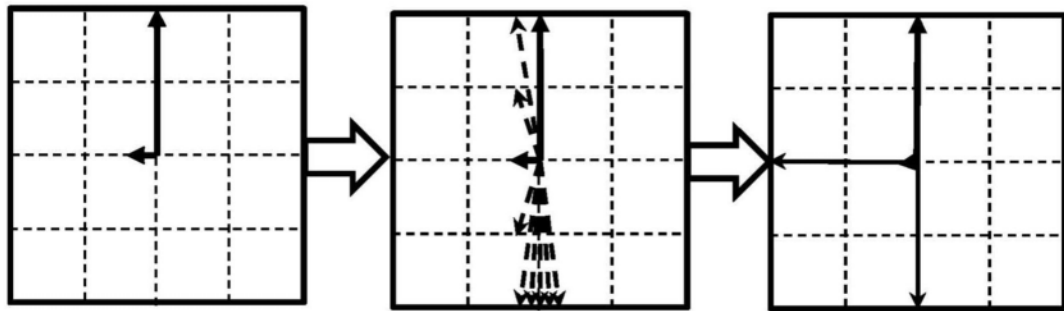


图19D

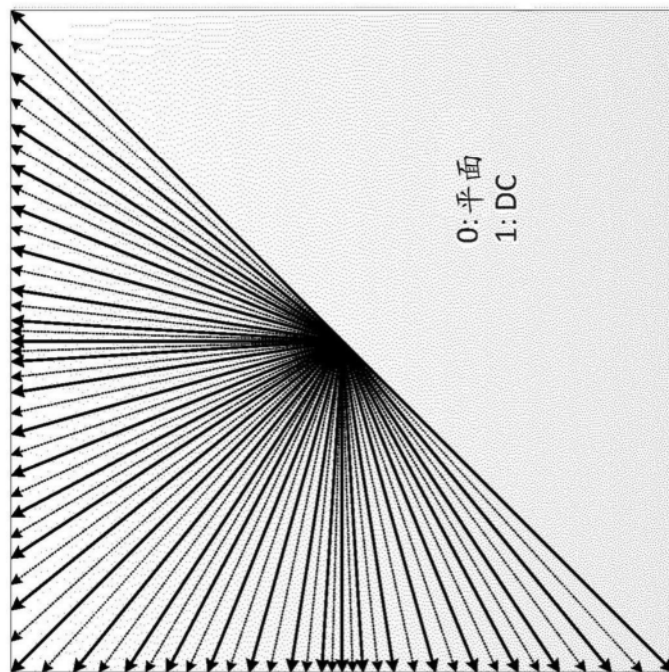


图17

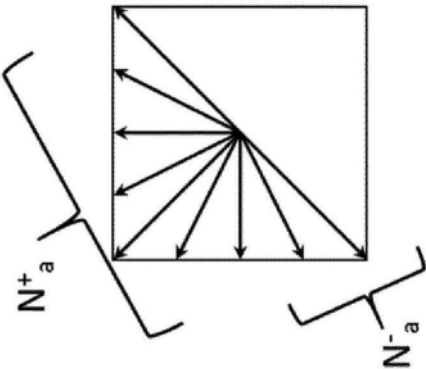


图18

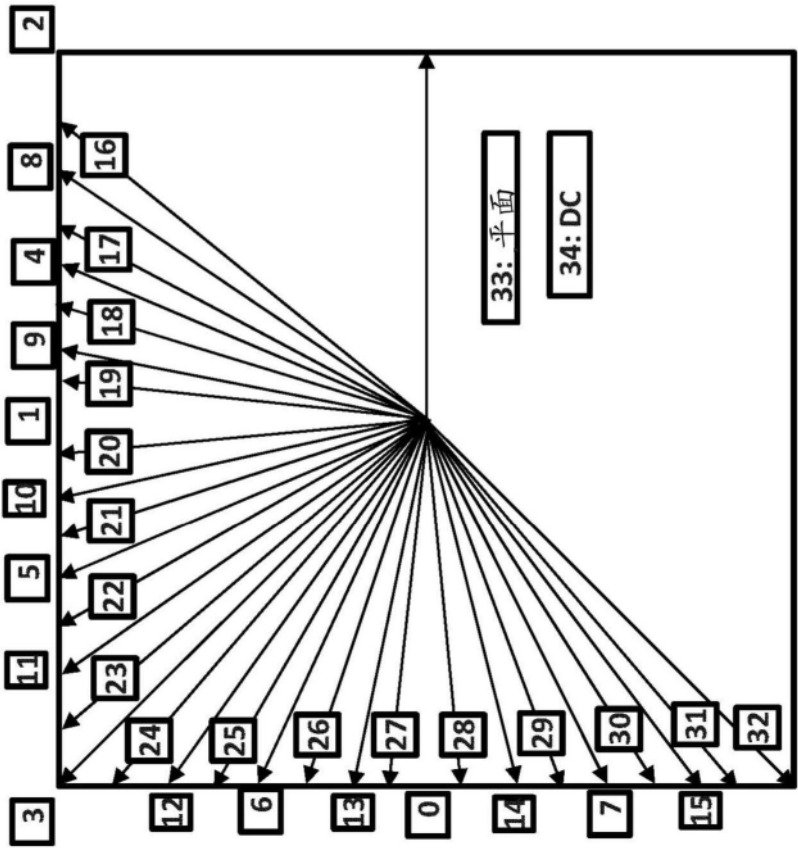


图20

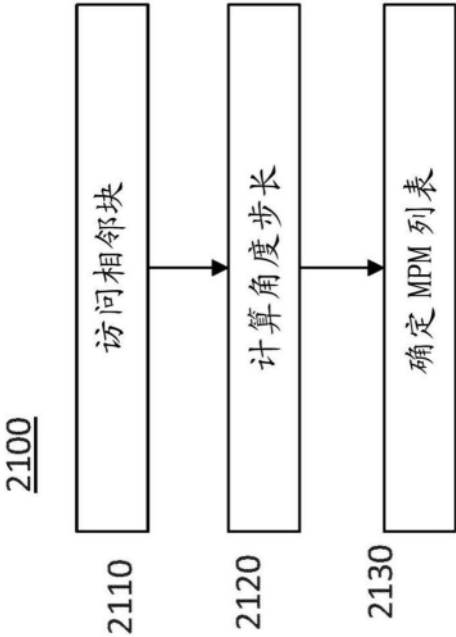


图21A

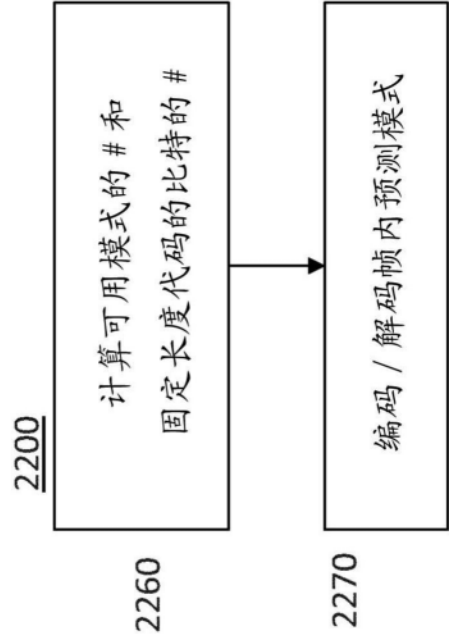


图21B

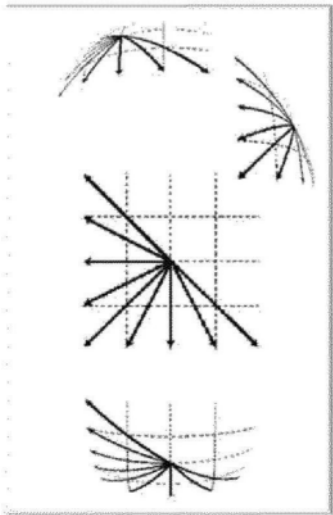


图22

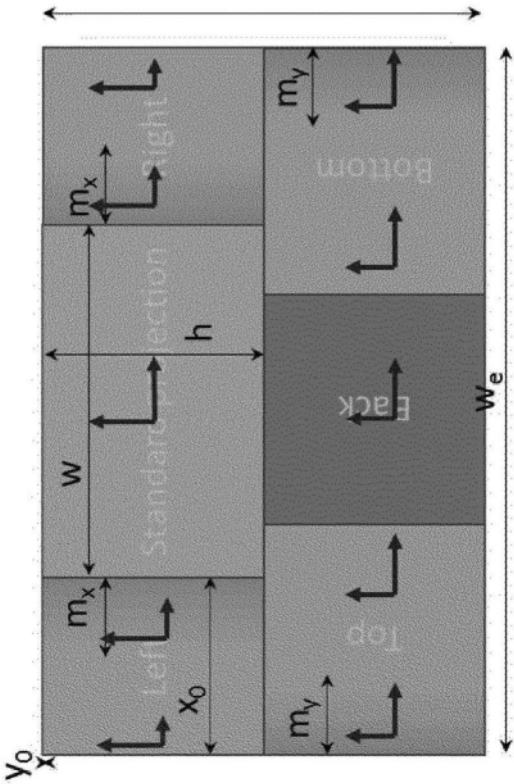


图23

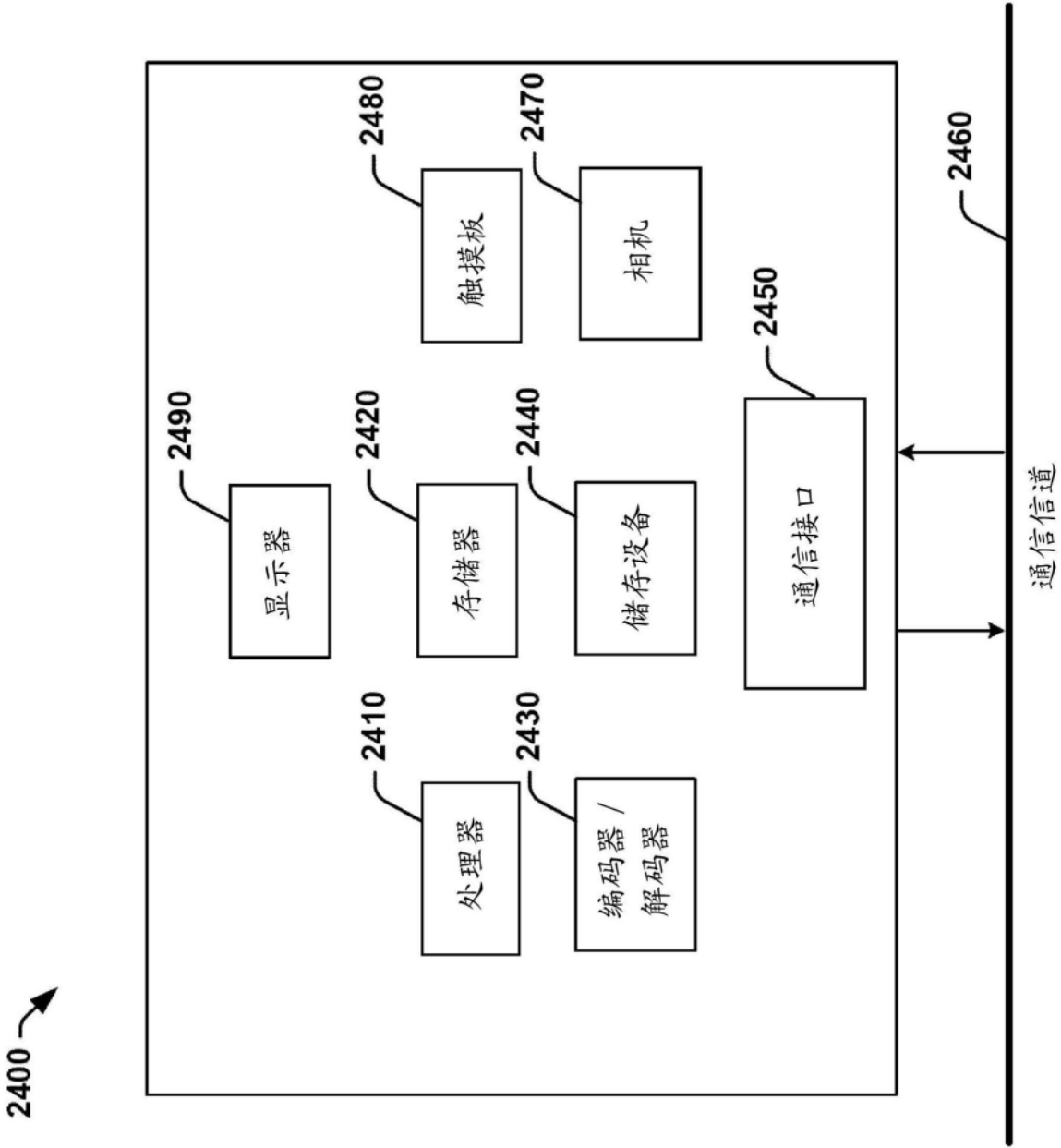


图24