



# (12)发明专利申请

(10)申请公布号 CN 109792517 A

(43)申请公布日 2019.05.21

(21)申请号 201780060867.8

(22)申请日 2017.09.14

(30)优先权数据

16306262.3 2016.09.30 EP

(85)PCT国际申请进入国家阶段日

2019.03.29

(86)PCT国际申请的申请数据

PCT/EP2017/073101 2017.09.14

(87)PCT国际申请的公布数据

W02018/059946 EN 2018.04.05

(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 加甘·拉斯 弗兰克·加尔平

法比安·拉卡佩 坦吉·波里尔

法布里斯·勒林内克

(74)专利代理机构 中科专利商标代理有限责任  
公司 11021

代理人 闫晔

(51)Int.Cl.

H04N 19/105(2006.01)

H04N 19/13(2006.01)

H04N 19/176(2006.01)

H04N 19/52(2006.01)

H04N 19/597(2006.01)

H04N 19/593(2006.01)

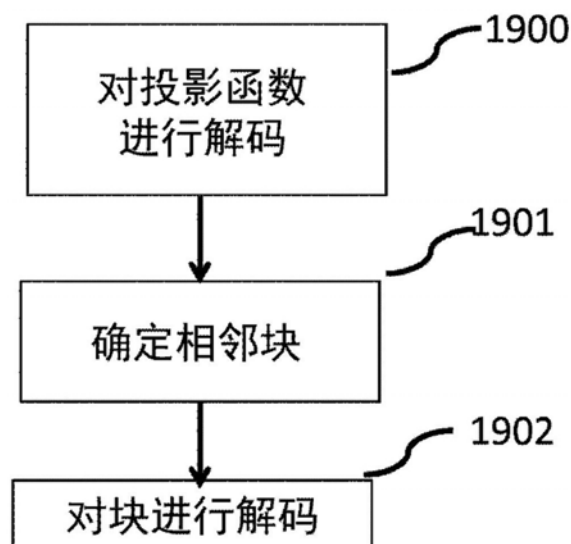
权利要求书4页 说明书21页 附图25页

(54)发明名称

用于编码和解码大视场视频的方法和装置

(57)摘要

公开了一种用于将大视场视频编码成比特流的方法和装置。所述大视场视频的至少一个图片被表示为表面,使用投影函数将所述表面投影到至少一个2D图片上。对于所述至少一个2D图片的至少一个当前块,根据所述投影函数确定所述2D图片的至少一个相邻块,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,并且在所述表面上所述至少一个相邻块与所述当前块在空间上相邻。然后,至少使用所述确定的相邻块对所述当前块进行编码。还公开了对应的解码方法和装置。



1. 一种用于将大视场视频编码成比特流的方法,所述大视场视频的至少一个图片被表示为表面,使用投影函数将所述表面投影到至少一个2D图片上,所述方法包括,对于根据当前帧内预测模式 $m$ 编码的所述至少一个2D图片的至少一个当前块:

-根据所述投影函数确定所述2D图片的被称为第一相邻块C的至少一个相邻块,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,所述至少一个相邻块在所述表面上与所述当前块在空间上相邻;

-基于所述第一相邻块C的帧内预测模式 $m_C$ ,并且还至少基于第二相邻块A的帧内预测模式 $m_A$ 和第三相邻块B的帧内预测模式 $m_B$ ,确定最可能模式的列表,所述第二相邻块和所述第三相邻块在所述2D图片中与所述当前块在空间上相邻;

-根据所述最可能模式的列表对所述当前帧内预测模式进行编码。

2. 根据权利要求1所述的方法,其中,所述确定最可能模式的列表包括:

-如果 $m_A$ 与 $m_B$ 不同,则按如下方式确定所述列表:

如果 $m_C$ 等于 $m_A$ 或 $m_B$ ,则所述最可能模式的列表包括 $m_A$ 和 $m_B$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m_A$ 和 $m_B$ 都不是平面模式的情况下等于平面模式,在 $m_A$ 或 $m_B$ 是平面模式但 $m_A$ 和 $m_B$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式;

否则,所述最可能模式的列表包括 $m_A$ 、 $m_B$ 和 $m_C$ ;

-如果 $m_A$ 与 $m_B$ 相等,则按如下方式确定所述列表:

如果 $m_C$ 等于 $m_A$ ,则所述最可能模式的列表在 $m_A$ 与平面模式和DC模式不同的情况下包括 $m_A$ 和 $m_A$ 的两个相邻角度模式,在其他情况下所述最可能模式的列表包括平面模式、DC模式和垂直模式,

否则,所述最可能模式的列表包括 $m_A$ 和 $m_C$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m_A$ 和 $m_C$ 都不是平面模式的情况下等于平面模式,在 $m_A$ 或 $m_C$ 是平面模式但 $m_A$ 和 $m_C$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式。

3. 根据权利要求1或2所述的方法,其中,对所述当前帧内预测模式进行编码包括:

-对标志进行编码,所述标志指示所述当前帧内预测模式是否等于所述最可能模式的列表中的一种模式;

-在所述当前帧内预测模式等于所述最可能模式的列表中的一种模式的情况下,对标识所述列表中的与所述当前帧内预测模式相等的最可能模式的索引进行编码,在其他情况下,对标识所述当前帧内预测模式的索引进行编码。

4. 根据权利要求1至3中任一项所述的方法,还包括:对与所述投影函数有关的信息项进行编码。

5. 根据权利要求1至4中任一项所述的方法,其中,所述3D表面是球面,并且所述投影函数是等矩形投影。

6. 一种用于将大视场视频编码成比特流的装置,所述大视场视频的至少一个图片被表示为表面,使用投影函数将所述表面投影到至少一个2D图片上,所述装置包括:

-用于对于根据当前帧内预测模式 $m$ 编码的所述至少一个2D图片的至少一个当前块,根据所述投影函数确定所述2D图片的被称为第一相邻块C的至少一个相邻块的单元,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,所述至少一个相邻块在所述

表面上与所述当前块在空间上相邻；

-用于基于所述第一相邻块C的帧内预测模式 $m_C$ ，并且还至少基于第二相邻块A的帧内预测模式 $m_A$ 和第三相邻块B的帧内预测模式 $m_B$ ，来确定最可能模式的列表的单元，所述第二相邻块和所述第三相邻块在所述2D图片中与所述当前块在空间上相邻；

-用于根据所述最可能模式的列表对所述当前帧内预测模式进行编码的单元。

7. 根据权利要求6所述的装置，其中，用于确定最可能模式的列表的单元被配置为按如下方式确定所述最可能模式的列表：

-如果 $m_A$ 与 $m_B$ 不同：

如果 $m_C$ 等于 $m_A$ 或 $m_B$ ，则所述最可能模式的列表包括 $m_A$ 和 $m_B$ 以及附加帧内预测模式，所述附加帧内预测模式在 $m_A$ 和 $m_B$ 都不是平面模式的情况下等于平面模式，在 $m_A$ 或 $m_B$ 是平面模式但 $m_A$ 和 $m_B$ 都不是DC模式的情况下等于DC模式，在其他情况下等于垂直帧内预测模式；

否则，所述最可能模式的列表包括 $m_A$ 、 $m_B$ 和 $m_C$ ；

-如果 $m_A$ 与 $m_B$ 相等：

如果 $m_C$ 等于 $m_A$ ，则所述最可能模式的列表在 $m_A$ 与平面模式和DC模式不同的情况下包括 $m_A$ 和 $m_A$ 的两个相邻角度模式，在其他情况下所述最可能模式的列表包括平面模式、DC模式和垂直模式，

否则，所述最可能模式的列表包括 $m_A$ 和 $m_C$ 以及附加帧内预测模式，所述附加帧内预测模式在 $m_A$ 和 $m_C$ 都不是平面模式的情况下等于平面模式，在 $m_A$ 或 $m_C$ 是平面模式但 $m_A$ 和 $m_C$ 都不是DC模式的情况下等于DC模式，在其他情况下等于垂直帧内预测模式。

8. 根据权利要求6至7中任一项所述的装置，其中，用于对所述当前帧内预测模式进行编码的单元被配置为：

-对标志进行编码，所述标志指示所述当前帧内预测模式是否等于所述最可能模式的列表中的一种模式；

-在所述当前帧内预测模式等于所述最可能模式的列表中的一种模式的情况下，对标识所述列表中的与所述当前帧内预测模式相等的最可能模式的索引进行编码，在其他情况下，对标识所述当前帧内预测模式的索引进行编码。

9. 根据权利要求6至8中任一项所述的装置，所述用于编码的单元还被配置为对与所述投影函数有关的信息项进行编码。

10. 根据权利要求6至9中任一项所述的装置，其中，所述3D表面是球面，并且所述投影函数是等矩形投影。

11. 一种用于解码表示大视场视频的比特流的方法，所述大视场视频的至少一个图片被表示为表面，使用投影函数将所述表面投影到至少一个2D图片上，所述方法包括，对于根据当前帧内预测模式 $m$ 编码的所述至少一个2D图片的至少一个当前块：

-根据所述投影函数确定所述2D图片的被称为第一相邻块C的至少一个相邻块，所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻，所述至少一个相邻块在所述表面上与所述当前块在空间上相邻；

-基于所述第一相邻块C的帧内预测模式 $m_C$ ，并且还至少基于第二相邻块A的帧内预测模式 $m_A$ 和第三相邻块B的帧内预测模式 $m_B$ ，确定最可能模式的列表，所述第二相邻块和所

述第三相邻块在所述2D图片中与所述当前块在空间上相邻;以及

-根据所述最可能模式的列表对所述当前帧内预测模式进行解码。

12. 根据权利要求11所述的方法,其中,所述确定最可能模式的列表包括:

-如果 $m_A$ 与 $m_B$ 不同,则按如下方式确定所述列表:

如果 $m_C$ 等于 $m_A$ 或 $m_B$ ,则所述最可能模式的列表包括 $m_A$ 和 $m_B$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m_A$ 和 $m_B$ 都不是平面模式的情况下等于平面模式,在 $m_A$ 或 $m_B$ 是平面模式但 $m_A$ 和 $m_B$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式;

否则,所述最可能模式的列表包括 $m_A$ 、 $m_B$ 和 $m_C$ ;

-如果 $m_A$ 与 $m_B$ 相等,则按如下方式确定所述列表:

如果 $m_C$ 等于 $m_A$ ,则所述最可能模式的列表在 $m_A$ 与平面模式和DC模式不同的情况下包括 $m_A$ 和 $m_A$ 的两个相邻角度模式,在其他情况下所述最可能模式的列表包括平面模式、DC模式和垂直模式,

否则,所述最可能模式的列表包括 $m_A$ 和 $m_C$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m_A$ 和 $m_C$ 都不是平面模式的情况下等于平面模式,在 $m_A$ 或 $m_C$ 是平面模式但 $m_A$ 和 $m_C$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式。

13. 根据权利要求11或12所述的方法,其中,对所述当前帧内预测模式进行解码包括:

-对标志进行解码,所述标志指示所述当前帧内预测模式是否等于所述最可能模式的列表中的一种模式;

-在所述当前帧内预测模式等于所述最可能模式的列表中的一种模式的情况下,对标识所述列表中的与所述当前帧内预测模式相等的最可能模式的索引进行解码,在其他情况下对标识所述当前帧内预测模式的索引进行解码。

14. 根据权利要求11至13中任一项所述的方法,还包括:对与所述投影函数有关的信息项进行解码。

15. 根据权利要求11至14中任一项所述的方法,其中,所述3D表面是球面,并且所述投影函数是等矩形投影。

16. 一种用于解码表示大视场视频的比特流的装置,所述大视场视频的至少一个图片被表示为表面,使用投影函数将所述表面投影到至少一个2D图片上,所述装置包括:

-用于对于根据当前帧内预测模式 $m$ 编码的所述至少一个2D图片的至少一个当前块,根据所述投影函数确定所述2D图片的被称为第一相邻块C的至少一个相邻块的单元,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,所述至少一个相邻块在所述表面上与所述当前块在空间上相邻;

-用于基于所述第一相邻块C的帧内预测模式 $m_C$ ,并且还至少基于第二相邻块A的帧内预测模式 $m_A$ 和第三相邻块B的帧内预测模式 $m_B$ ,来确定最可能模式的列表的单元,所述第二相邻块和所述第三相邻块在所述2D图片中与所述当前块在空间上相邻;以及

-用于根据所述最可能模式的列表对所述当前帧内预测模式进行解码的单元。

17. 根据权利要求16所述的装置,其中,所述用于确定最可能模式的列表的单元被配置为按如下方式确定所述最可能模式的列表:

-如果 $m_A$ 与 $m_B$ 不同:

如果 $m\_C$ 等于 $m\_A$ 或 $m\_B$ ,则所述最可能模式的列表包括 $m\_A$ 和 $m\_B$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m\_A$ 和 $m\_B$ 都不是平面模式的情况下等于平面模式,在 $m\_A$ 或 $m\_B$ 是平面模式但 $m\_A$ 和 $m\_B$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式;

否则,所述最可能模式的列表包括 $m\_A$ 、 $m\_B$ 和 $m\_C$ ;

-如果 $m\_A$ 与 $m\_B$ 相等:

如果 $m\_C$ 等于 $m\_A$ ,则所述最可能模式的列表在 $m\_A$ 与平面模式和DC模式不同的情况下包括 $m\_A$ 和 $m\_A$ 的两个相邻角度模式,在其他情况下所述最可能模式的列表包括平面模式、DC模式和垂直模式,

否则,所述最可能模式的列表包括 $m\_A$ 和 $m\_C$ 以及附加帧内预测模式,所述附加帧内预测模式在 $m\_A$ 和 $m\_C$ 都不是平面模式的情况下等于平面模式,在 $m\_A$ 或 $m\_C$ 是平面模式但 $m\_A$ 和 $m\_C$ 都不是DC模式的情况下等于DC模式,在其他情况下等于垂直帧内预测模式。

18. 根据权利要求17所述的装置,其中,所述用于对所述当前帧内预测模式进行解码的单元被配置为:

-对标志进行解码,所述标志指示所述当前帧内预测模式是否等于所述最可能模式的列表中的一种模式;

-在所述当前帧内预测模式等于所述最可能模式的列表中的一种模式的情况下,对标识所述列表中的与所述当前帧内预测模式相等的最可能模式的索引进行解码,在其他情况下对标识所述当前帧内预测模式的索引进行解码。

19. 根据权利要求16至18中任一项所述的装置,所述用于解码的单元还被配置为对与所述投影函数有关的信息项进行解码。

20. 根据权利要求16至19中任一项所述的装置,其中,所述3D表面是球面,并且所述投影函数是等矩形投影。

21. 一种沉浸式呈现设备,包括根据权利要求16至20中任一项所述的用于解码表示大视场视频的比特流的装置。

22. 一种用于沉浸式呈现编码成比特流的大视场视频的系统,至少包括:

-网络接口(600),用于从数据网络接收所述比特流,

-根据权利要求16至20中任一项所述的用于解码所述比特流的装置(700),

-沉浸式呈现设备(900)。

## 用于编码和解码大视场视频的方法和装置

### 技术领域

[0001] 本公开涉及例如当在虚拟现实、增强现实或增强虚拟系统中处理沉浸式视频时，以及例如当在头戴式显示设备中进行显示时，对沉浸式视频进行编码和解码。

### 背景技术

[0002] 最近，可用的大视场内容（高达360°）有所增加。观看沉浸式显示设备（例如头戴式显示器、智能眼镜、PC屏幕、平板电脑、智能电话等）上的内容的用户可能无法完整地看到这样的内容。这意味着在给定时刻，用户可能仅查看内容的一部分。然而，用户通常可以通过诸如头部移动、鼠标移动、触摸屏、语音等各种手段在内容内导航。通常希望对该内容进行编码和解码。

### 发明内容

[0003] 根据本原理的一个方面，公开了一种用于将大视场视频编码成比特流的方法。所述大视场视频的至少一个图片被表示为3D表面，所述3D表面使用投影函数投影到至少一个2D图片上。所述编码方法包括，对于所述至少一个2D图片的至少一个当前块：

[0004] -根据所述投影函数确定所述2D图片的至少一个相邻块，所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻，在所述3D表面上与所述当前块在空间上相邻；

[0005] -至少使用所确定的相邻块对所述当前块进行编码。

[0006] 本原理允许当在3D表面上空间邻近的相邻块可用来对当前块进行编码时，根据用于将3D表面投影到一个或多个图片上的投影函数，来确定要编码的当前块的新的相邻块。例如，在当前块位于2D矩形图片的边界处或立方体投影的面的边界处时，在3D表面上与当前块空间相邻的已经进行了编码和解码的块可以被确定为用于对当前块进行编码的新的相邻块。当在2D图片上投影3D表面时，当前块的这种适应后的邻域允许恢复这样的3D表面中的区域的空间相邻的邻域。

[0007] 适应后的邻域可被2D视频编码器的任何编码模块用于对当前块进行编码，从而允许增加应用于大视场视频的2D视频编码方案的压缩效率。

[0008] 根据本公开的实施例，对所述当前块进行编码属于至少包括以下项的组：

[0009] -确定最可能模式的列表，该最可能模式的列表用于对所述当前块的帧内预测模式进行编码，

[0010] -导出运动矢量预测值，该运动矢量预测值用于对所述当前块的运动矢量进行编码，

[0011] -在帧间预测合并模式下导出用于对所述当前块进行编码的运动信息，

[0012] -对所述当前块进行上下文算术熵编码，

[0013] -对所述当前块的至少一个样本进行样本自适应偏移滤波。

[0014] 根据本公开的另一个实施例，该编码方法还包括对与所述投影函数有关的信息项

进行编码。

[0015] 还公开了一种用于解码表示大视场视频的比特流的方法。至少一个2D图片表示所述大视场视频的被表示为3D表面的图片的投影,所述解码方法包括,对于所述至少一个2D图片的至少一个当前块:

[0016] -根据所述投影函数确定所述2D图片的至少一个相邻块,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,在所述3D表面上与所述当前块在空间上相邻;

[0017] -至少使用所确定的相邻块对所述当前块进行解码。

[0018] 根据本公开的实施例,对所述当前块进行解码属于至少包括以下项的组:

[0019] -确定最可能模式的列表,该最可能模式的列表用于对所述当前块的帧内预测模式进行解码,

[0020] -导出运动矢量预测值,该运动矢量预测值用于对所述当前块的运动矢量进行重建,

[0021] -在帧间预测合并模式下导出用于对所述当前块进行重建的运动信息,

[0022] -对所述当前块进行上下文算术熵解码,

[0023] -对所述当前块的至少一个样本进行样本自适应偏移滤波。

[0024] 根据本公开的另一个实施例,该解码方法还包括对与所述投影函数有关的信息项进行解码。

[0025] 根据本公开的另一个实施例,所述3D表面是球面,并且所述投影函数是等矩形投影。根据该实施例的变型,当前块位于2D图片右边界上,所述至少一个相邻块位于2D图片左边界上。

[0026] 根据本公开的另一个实施例,编码或解码所述当前块包括:构造预测值列表,该预测值列表至少包括从所述至少一个相邻块获得的预测数据,其中,来自所述当前块的数据使用来自所述预测值列表的选定的预测数据候选进行编码或解码。

[0027] 根据本实施例,将由当前块的新确定的相邻块提供的预测数据添加到用于编码或解码当前块的预测值列表中。例如,在对当前块进行帧内编码时,这样的预测值列表可以是帧内预测模式的最可能模式。在对当前块进行帧间预测时,预测值列表可以对应于用于预测当前块的运动矢量的运动矢量预测值集合,或者对应于运动候选块集合,当前块从该运动候选块集合继承用于预测当前块的运动信息。

[0028] 预测值列表还可以对应于滤波参数,例如样本自适应偏移参数(在HEVC标准中也称为SAO),当前块继承这些参数以处理当前块的重建像素。

[0029] 还公开了一种用于将大视场视频编码成比特流的装置。至少一个2D图片表示所述大视场视频的被表示为3D表面的图片的投影,这种装置包括,对于所述至少一个2D图片的至少一个当前块:-用于根据所述投影函数确定所述2D图片的至少一个相邻块的单元,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,在所述3D表面上与所述当前块在空间上相邻;以及用于至少使用所确定的相邻块对所述当前块进行编码的单元。

[0030] 还公开了一种用于解码表示大视场视频的比特流的装置。至少一个2D图片表示所述大视场视频的被表示为3D表面的图片的投影,这种装置包括,对于所述至少一个2D图片的至少一个当前块:

[0031] -用于根据所述投影函数确定所述2D图片的至少一个相邻块的单元,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,在所述3D表面上与所述当前块在空间上相邻;

[0032] -用于至少使用所确定的相邻块对所述当前块进行解码的单元。

[0033] 还公开了一种表示编码了的大视场视频的比特流。所述大视场视频的至少一个图片被表示为3D表面,所述3D表面使用投影函数投影到至少一个2D图片上。-该比特流包括表示所述2D图片的至少一个当前块的编码数据,所述当前块使用所述2D图片的至少一个相邻块进行编码,所述至少一个相邻块在所述2D图片中不与所述当前块在空间上相邻,在所述3D表面上与所述当前块在空间上相邻;

[0034] 根据本公开的一个实施例,该比特流还包括表示与所述投影函数有关的信息项的编码数据。

[0035] 根据本公开的另一个实施例,该比特流存储在非易失性处理器可读介质中。

[0036] 还公开了一种沉浸式呈现设备,其包括用于解码表示大视场视频的比特流的装置。

[0037] 还公开了一种用于沉浸式呈现编码成比特流的大视场视频的系统。这种系统至少包括:

[0038] -网络接口,用于从数据网络接收所述比特流,

[0039] -根据本文中公开的任一实施例所述的用于解码所述比特流的装置,

[0040] -用于呈现解码后的大视场视频的沉浸式呈现设备。

[0041] 根据一个实施方案,如上文所述的用于编码大视场视频的方法或用于解码表示大视场视频的比特流的方法的不同步骤,由一个或多个包括软件指令在内的软件程序或软件模块程序实现,这些软件指令旨在由用于编码大视场视频的装置或用于解码表示大视场视频的比特流的装置的数据处理器执行,这些软件指令被设计用来指示执行根据本原理的方法的不同步骤。

[0042] 还公开了一种计算机程序,其能够由计算机或数据处理器执行,该程序包括用于指示方法步骤的执行的指令,所述方法是如上所述的用于编码大视场视频的方法,或者用于解码表示大视场视频的比特流的方法。

[0043] 该程序可以使用任何编程语言,并可以是源代码、目标代码或源代码和目标代码之间的中间代码的形式,例如,部分编译形式或任何其它所需形式。

[0044] 信息载体可以是能够存储程序的任何实体或装置。例如,载体可以包括存储装置,诸如ROM(例如CD-ROM或微电子电路ROM)或磁性记录装置(例如软盘或硬盘驱动器)。

[0045] 此外,信息载体可以是可传输载体,例如,由无线电或其它装置通过电缆或光缆传递的电信号或光信号。根据本原理的程序可以特别地上载到因特网类型的网络。

[0046] 作为替代,信息载体可以是其中包含程序的集成电路,该电路适于执行或用于执行所时论的方法。

[0047] 根据一个实施例,可以通过软件和/或硬件组件来实现方法/装置。从这个观点,本文中的术语“模块”或“单元”可以对应于软件组件以及对应于硬件组件,或对应于硬件与软件组件的集合。

[0048] 软件组件对应于一个或更多计算机程序、程序的一个或更多子程序,或更一般地,



对应于能够实现根据以下针对所涉及模块描述的功能或功能集合的程序或软件的任何元素。这种软件组件由物理实体(终端、服务器等)的数据处理器来执行,并能够访问该物理实体的硬件资源(存储器、记录介质、通信总线、输入/输出电子板、用户接口等)。

[0049] 类似地,硬件组件对应于能够实现根据以下针对所涉及模块描述的功能或功能集合的硬件单元的任何元素。硬件组件可以是可编程的硬件组件或具有用于执行软件的集成处理器的组件,例如,集成电路、智能卡、存储卡、用于执行固件等的电子板等。

[0050] 除全向视频外,本原理还适用于大视场视频内容(例如180°)。

## 附图说明

[0051] 图1表示根据本公开实施例的优选环境的编码和解码系统的功能概述,

[0052] 图2表示根据本公开的系统的第二实施例,

[0053] 图3表示根据本公开的系统的第二实施例,

[0054] 图4表示根据本公开的系统的第二实施例,

[0055] 图5表示根据本公开的系统的第二实施例,

[0056] 图6表示根据本公开的系统的第二实施例,

[0057] 图7表示根据本公开的系统的第二实施例,

[0058] 图8表示根据本公开的系统的第二实施例,

[0059] 图9表示根据本公开的系统的第二实施例,

[0060] 图10表示根据本公开的沉浸式视频呈现设备的第一实施例,

[0061] 图11表示根据本公开的沉浸式视频呈现设备的第一实施例,

[0062] 图12表示根据本公开的沉浸式视频呈现设备的第一实施例,

[0063] 图13A示出了从球面S到矩形图片F的投影的示例,

[0064] 图13B示出了图片F的XY平面参考系统,

[0065] 图13C示出了球体S上的角度参考系统,

[0066] 图14A示出了从立方体表面S到6个图片的投影的示例,

[0067] 图14B示出了立方体参考系统,

[0068] 图14C示出了2D图片F的XY平面参考系统,

[0069] 图14D示出了投影在2D图片上的立方体的6个面的布局,

[0070] 图14E示出了根据图14D所示的布局的相应重新排列后的矩形图片,

[0071] 图15示出了传统视频编码方案中的因果空间邻域,

[0072] 图16A示出了矩形图片,已经使用等矩形投影将表示为球体的全向视频投影到该矩形图片上,

[0073] 图16B示出了矩形图片,已经使用立方体投影和根据图14D的6个面的布局将表示为立方体的全向视频投影到该矩形图片上,

[0074] 图17示出了根据本公开实施例的用于对2D图片的当前块进行编码的示例性方法的框图,该2D图片是全向视频的投影,

[0075] 图18示出了根据本公开实施例的用于将全向视频编码为比特流的示例性方法的框图,

[0076] 图19示出了根据本公开实施例的用于对2D图片的当前块进行解码的示例性方法

的框图,该2D图片是全向视频的投影,

[0077] 图20示出了根据本公开实施例的用于从比特流中解码2D图片的当前块的示例性方法的框图,该2D图片表示全向视频的3D图片,

[0078] 图21示出了根据一个实施例的用于将全向视频编码为比特流的示例性装置,

[0079] 图22示出了根据一个实施例的用于对表示全向视频的比特流进行解码的示例性装置,

[0080] 图23示出了根据本公开实施例的用于确定最可能的帧内预测模式的当前块的自适应邻域,

[0081] 图24示出了根据本公开实施例的用于导出运动矢量预测值或运动信息的当前块的自适应邻域,

[0082] 图25示出了根据本公开实施例的用于导出用于上下文算术二进制编码的上下文的当前块的自适应邻域,

[0083] 图26示出了根据本公开实施例的用于导出样本自适应偏移参数的当前块的自适应邻域,

[0084] 图27是描绘HEVC中的帧内预测方向的图示示例。

### 具体实施方式

[0085] 大视场内容可以是三维计算机图形图像场景(3D CGI场景)、点云或沉浸式视频等。可以使用许多术语来设计这样的沉浸式视频,例如虚拟现实(VR)、360、全景、4 $\pi$ 球面度、沉浸式、全向、大视场等。

[0086] 为了将全向视频编码成比特流(例如用于通过数据网络进行传输),可以使用传统的视频编解码器(例如HEVC、H.264/AVC)。因此,全向视频的每个图片首先使用合适的投影函数投影在一个或多个2D图片(例如一个或多个矩形图片)上。在实践中,来自全向视频的图片被表示为3D表面。为了便于投影,通常使用凸起和简单的表面(例如球体、立方体或金字塔)用于投影。然后,使用传统的视频编解码器对表示全向视频的投影2D图片进行编码。

[0087] 图13A示出了使用等矩形投影将全向视频的帧投影到一个矩形图片I上的示例,该全向视频的帧映射在表示为球体的表面S上。

[0088] 图14A示出了将全向视频的帧投影到六个图片或面上的另一示例,该全向视频的帧映射在表示为立方体的表面S上。图14B中示出了立方体的这些面的参考系统,立方体的这些面可以使用图14D所示的布局重新布置成如图14E所示的一个矩形图片。

[0089] 为了对全向视频进行编码,可以使用诸如HEVC、H.264/AVC等的传统视频编码标准对表面的投影矩形图片进行编码。根据这样的标准,通过首先将2D图片分成小的非重叠的块,然后单独编码这些块,来对2D图片进行编码。为了减少冗余,传统视频编码器使用因果空间相邻块数据来预测当前待编码块的值。图15示出了这种因果空间相邻块的示例,其中当前待编码块BK具有4个相邻块A、B、C和D,这些相邻块已经被编码/解码并且可以在当前块BK的编码/解码处理的编码步骤中使用。这种邻域可以用于帧内预测、最可能的编码模式确定(在HEVC或H.264/AVC中称为MPM确定)、帧间图片编码中的运动矢量预测。例如,在去块滤波处理或样本自适应偏移处理(在HEVC中也称为SAO)中,这种邻域也可以用于在编码之后对当前块进行滤波。取决于要执行的处理,可以使用其他因果空间/时间邻域。

[0090] 因果空间相邻块在此被理解为根据图片的扫描顺序(例如光栅扫描顺序)已经进行了编码和解码的块。

[0091] 在全向视频中,当使用等矩形投影时,由于球体的圆对称性,来自3D表面的块在该块的左侧和该块的右侧具有相邻块。然而,当将3D表面投影到2D矩形图片上时,一些相邻块可能不再可用于块。例如,在等矩形投影中,投影到位于2D图片右边界处的2D图片的块上的3D表面的块,在3D表面上与投影到位于2D图片左边界处的2D块上的3D表面的块相邻。例如,如图16A所示,位于2D图片的右边界上的块F和J在3D表面上与块A和G是空间相邻的。

[0092] 然而,在2D图片中,位于图片右边界的块不再与位于图片左边界的块相邻。如图16A所示,在2D图片上,块F和J在空间上不再与块A和G相邻。

[0093] 与不位于图片边界的2D块的编码相比,传统的视频编码方案以特殊方式对这些块进行编码。实际上,传统的视频编码方案必须应对丢失的邻域。

[0094] 此外,当在2D图片解码之后将2D图片重新投影到3D表面上时,因为3D表面的连续性已被破坏,所以在包括从位于2D图片边界处的2D块投影的3D块在内的范围中可能出现一些视觉伪像,可以在2D图片的编码处理中应用自适应处理(例如像素传播或填充)。

[0095] 当将全向视频表示为3D立方体并将立方体的投影后的6个面重新排列在矩形图片上(如图14E或14F所示)时,会出现类似的问题。在这样的投影图片中,位于立方体的投影面边界上的块在2D图片中不能从那些属于3D立方体上的相邻面的因果相邻块中受益。

[0096] 因此,需要一种新颖的全向视频编码和解码方法。

[0097] 这里在全向视频的情况下公开本原理,但本原理也可以应用于以非常大的视场获得的传统平面图像的情况,即,像鱼镜头一样以非常小的焦距获得的传统平面图像。

[0098] 图1示出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300可以准备内容以供编码设备400进行编码。预处理模块300可以执行多图像采集,将所获取的多个图像在共同空间(如果我们编码多个方向,则通常是3D球体)中进行合并,并且使用例如(但不限于)等矩形映射或立方体映射将3D球体映射到2D帧。预处理模块300还可以接受特定格式(例如,等矩形)的全向视频作为输入,并对该视频进行预处理以将映射改变为更适合于编码的格式。取决于所获取的视频数据表示,预处理模块可以执行映射空间改变。将参照说明书的其他附图描述编码设备400和编码方法。在编码之后,将数据(例如,可以对沉浸式视频数据进行编码,或者是3D CGI编码数据)发送到网络接口500,网络接口500通常可以以(例如存在于网关中的)任何网络接口的形式实现。然后,通过诸如因特网之类的通信网络(但可以预见任何其他网络)来传输数据。然后,经由网络接口600接收数据。网络接口600可以在网关、电视、机顶盒、头戴式显示设备、沉浸式(投影)墙或任何沉浸式视频呈现设备中实现。在接收之后,将数据发送到解码设备700。解码功能是以下图2至图12中描述的处理功能之一。然后,由播放器800处理解码的数据。播放器800准备数据以用于呈现设备900,并且可以接收来自传感器的外部数据或用户输入数据。更确切地说,播放器800准备视频内容中将由呈现设备900显示的那部分内容。解码设备700和播放器800可以集成在单个设备(例如,智能电话、游戏机、STB、平板电脑、计算机等)中。在变型中,播放器800集成在呈现设备900中。

[0099] 可以设想几种类型的系统来例如在呈现沉浸式视频时执行沉浸式显示设备的解码、播放和呈现功能。

[0100] 用于处理增强现实、虚拟现实或增强虚拟内容的第一系统在图2至图6中示出。这样的系统包括处理功能和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)、平板电脑或智能电话,并且可以包括传感器。沉浸式视频呈现设备还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多个设备执行。它们可以集成到沉浸式视频呈现设备中,也可以集成到一个或多个处理设备中。处理设备包括一个或多个处理器和与沉浸式视频呈现设备通信的通信接口(例如无线或有线通信接口)。

[0101] 处理设备还可以包括与诸如因特网之类的广域接入网络通信的第二通信接口,并且直接地或通过诸如家庭网关或本地网关之类的网络设备访问位于云上的内容。处理设备还可以通过第三接口(例如以太网类型的本地接入网接口)访问本地存储设备。在一个实施例中,处理设备可以是具有一个或多个处理单元的计算机系统。在另一个实施例中,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,或者可以插入沉浸式视频呈现设备的外壳中并通过连接器或无线地与沉浸式视频呈现设备通信。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(例如IEEE 802.11接口或蓝牙®接口)。

[0102] 当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直接地或通过网关设置有与网络的接口,以接收和/或发送内容。

[0103] 在另一实施例中,该系统包括辅助设备,该辅助设备与沉浸式视频呈现设备和处理设备通信。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。

[0104] 沉浸式视频呈现设备可以包括一个或多个显示器。该设备可以在其每个显示器的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,如智能电话或平板电脑那样。在另一个实施例中,显示器和光学器件可以嵌入头盔、眼镜或用户可以佩戴的面罩中。沉浸式视频呈现设备还可以集成若干传感器,如稍后所述。沉浸式视频呈现设备还可以包括多个接口或连接器。它可能包括一个或多个无线模块,以便与传感器、处理功能、手持设备或其他与身体部位相关的设备或传感器进行通信。

[0105] 沉浸式视频呈现设备还可以包括由一个或多个处理器执行的处理功能,并且被配置为解码内容或处理内容。在此,处理内容可以理解为准备可以显示的内容的所有功能。例如,这可以包括对内容进行解码、在显示内容之前合并内容、以及修改内容以适合于显示设备。

[0106] 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卷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。该系统还可以包括与环境相关的其他传感器,例如用来测量照明、温度或声音条件。这样的传感器还可以与用户的身体相关,例如用来测量出汗或心率。通过这些传感器获取的信息可用于对内容进行处理。该系统还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用于对内容进行处理、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/或沉浸式呈现设备通信。

[0107] 使用图2至图6,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系统的若干实施例。图2示出了被配置为解码、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)。

[0108] 如图10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、平板电脑或智能电话。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连接的存储器105、以及至少一个通信接口106。至少一个处理器104处理从传感器20接收的信号。来自传感器的一些测量值用于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。也可以使用更复杂的系统,例如使用一套摄像头。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。使用一些其他测量值来根据环境条件或用户反应处理内容。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,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执行预期测量。传感器20和用户输入设备30数据也可以传输到计算机40,计算机40将根据这些传感器的输入来处理数据。

[0109] 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还可以包括从传感器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(例如IEEE 802.11接口或蓝牙®接口)。计算机40将数据和可选的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,即准备数据以供沉浸式视频呈现设备10显示。处理可以仅由计算机40完成,或者处理的一部分可以由计算机完成,一部分由沉浸式视频呈现设备10完成。计算机40直接地或通过网关或网络接口50连接到因特网。计算机40从因特网接收表示沉浸式视频的数据,处理这些数据(例如,对数据进行解码,并且可能准备视频内容中将由沉浸式视频呈现设备10显示的那部分内容),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。在一种变型中,系统还可以包括存储表示沉浸式视频的数据的本地存储设备(未示出),所述本地存储设备可以位于计算机40上或位于例如可通过局域网访问的本地服务器上(未示出)。

[0110] 图3表示第二实施例。在该实施例中,STB 90直接地(即,STB 90包括网络接口)或经由网关或网络接口50连接到诸如因特网之类的网络。STB 90通过无线接口或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的经典功能之外,STB 90还包括处理视频内容以便在电视机100或任何沉浸式视频呈现设备200上呈现的处理功能。这些处理功能与为计算机40描述的处理功能相同,这里不再描述。传感器20和用户输入设备30也与前面关于图2描述的类型相同。STB 90从因特网获得表示沉浸式视频的数据。在一种变型中,STB 90从存储表示沉浸式视频的数据的本地存储设备(未示出)获得表示沉浸式视频的数据,所述本地存储设备可以位于例如可通过局域网访问的本地服务器(未示出)上。

[0111] 图4表示与图2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将数据和可选的控制命令发送到沉浸式视频呈现设备10。游戏控制台60被配置为处理表示沉浸式视频的数据,并将处理后的数据发送到沉浸式视频呈现设备10以供显示。处理可以由游戏控制台60单独完成,或者部分处理可以由沉浸式视频呈现设备10完成。

[0112] 游戏控制台60直接地或通过网关或网络接口50连接到因特网。游戏控制台60从因

特网获得表示沉浸式视频的数据。在一个变型中,游戏控制台60从存储表示沉浸式视频的数据的本地存储设备(未示出)获得表示沉浸式视频的数据,所述本地存储设备可以位于游戏控制台60上,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0113] 游戏控制台60从因特网接收表示沉浸式视频的数据,处理这些数据(例如,对数据进行解码,并且可能准备将要显示的那部分视频内容),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因特网或从本地存储设备获得的表示沉浸式视频的数据。

[0114] 图5表示所述第一类系统的第四实施例,其中沉浸式视频呈现设备70由插入外壳705中的智能电话701形成。智能电话701可以连接到因特网,因此可以从因特网获得表示沉浸式视频的数据。在一个变型中,智能电话701从存储表示沉浸式视频的数据的本地存储设备(未示出)获得表示沉浸式视频的数据,所述本地存储设备可以位于智能电话701上,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0115] 参考图11描述了沉浸式视频呈现设备70,图11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的外壳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的显示器。因此,不包括除智能电话701的显示器之外的显示器。然而,包括诸如镜头之类的光学器件704,用于看到智能电话显示器上的数据。智能电话701被配置为处理(例如,解码并准备显示)表示沉浸式视频的数据,该处理可能根据从传感器20和用户输入设备30接收的数据。来自传感器的一些测量值用于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。也可以使用更复杂的系统,例如使用一套摄像头。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。使用一些其他测量值来根据环境条件或用户反应处理内容。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,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执行预期测量。

[0116] 图6表示所述第一类系统的第五实施例,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被配置为处理(例如,解码并准备显示)表示沉浸式视频的数据,该处理可能根据从传感器20和用户输入设备30接收的数据。沉浸式视频呈现设备80可以连接到因特网,因此可以从因特网获得表示沉浸式视频的数据。在一个变型中,沉浸式视频呈现设备80从存储表示沉浸式视频的数据的本地存储设备(未示出)获得表示沉浸式视频的数据,所述本地存储设备可以位于沉浸式视频呈现设备80上,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0117] 沉浸式视频呈现设备80在图12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)802、相机(可选)803、与至少一个处理器804连接的存储器805、以及至少一个通信接口806。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还可以包括从传感器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表示沉浸式视频内容的数据。存储器805可以是不同类型(SD卡、硬盘、易失性或非易失性存储器等)。通信接口806使沉浸式视频呈现设备能够与因特网通信。处理器804处理表示视频的数据,以便在显示器801上显示它们。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。

[0118] 用于处理增强现实、虚拟现实或增强虚拟内容的第二系统在图7至图9中示出。这种系统包括沉浸式墙。

[0119] 图7表示第二类系统。它包括显示器1000,其是从计算机4000接收数据的沉浸式(投射)墙。计算机4000可以从因特网接收沉浸式视频数据。计算机4000通常直接地或通过网关5000或网络接口连接到因特网。在一个变型中,沉浸式视频数据由计算机4000从存储表示沉浸式视频的数据的本地存储设备(未示出)获得,所述本地存储设备可以位于计算机4000中,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0120] 该系统还可以包括传感器2000和用户输入设备3000。沉浸式墙1000可以是OLED或LCD类型。它可以配备有一个或几个摄像头。沉浸式墙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接收的数据可以与照明条件、温度、用户的环境(例如,物体的位置)有关。

[0121] 沉浸式墙1000还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,以便提供关于用户情绪的反馈。用户输入设备3000的示例是手持设备,例如智能电话、遥控器和具有陀螺仪功能的设备。

[0122] 传感器2000和用户输入设备3000的数据也可以传输到计算机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,解码它们并准备它们以用于显示)。传感器信号可以通过沉浸式墙的通信接口接收。该通信接口可以是蓝牙类型、WIFI类型或任何其他类型的连接,优选地是无线连接,但也可以是有线连接。

[0123] 计算机4000将处理后的数据和可选的控制命令发送到沉浸式墙1000。计算机4000被配置为处理要由沉浸式墙1000显示的数据(即准备它们以供显示)。处理可以由计算机4000单独完成,或者处理的一部分可以由计算机4000完成,一部分由沉浸式墙1000完成。

[0124] 图8表示另一个第二类系统。它包括沉浸式(投影)墙6000,其被配置为处理(例如,解码和准备数据以用于显示)并显示视频内容。它还包括传感器2000和用户输入设备3000。

[0125] 沉浸式墙6000通过网关5000或直接从因特网接收来自因特网的沉浸式视频数据。在变型中,沉浸式视频数据由沉浸式墙6000从存储表示沉浸式视频的数据的本地存储设备(未示出)获得,所述本地存储设备可以位于沉浸式墙6000中,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0126] 该系统还可以包括传感器2000和用户输入设备3000。沉浸式墙6000可以是OLED或LCD类型。它可以配备有一个或几个摄像头。沉浸式墙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接收的数据可以与照明条件、温度、用户的环境(例如,物体的位置)有关。

[0127] 沉浸式墙6000还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,以便提供关于用户情绪的反馈。用户输入设备3000的示例是手持设备,例如智能电话、遥控器和具有陀螺仪功能的设备。

[0128] 沉浸式墙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,解码它们并准备它们以用于显示)。传感器信号可以通过沉浸式墙的通信接口接收。该通信接口可以是蓝牙类型、WIFI类型或任何其他类型的连接,优选地是无线连接,但也可以是有线连接。沉浸式墙6000可以包括至少一个通信接口,以与传感器和因特网通信。



[0129] 图9示出了第三实施例,其中沉浸式墙用于游戏。一个或多个游戏控制台7000优选地通过无线接口连接到沉浸式墙6000。沉浸式墙6000通过网关5000或直接从因特网接收来自因特网的沉浸式视频数据。在变型中,沉浸式视频数据由沉浸式墙6000从存储表示沉浸式视频的数据的本地存储设备(未示出)获得,所述本地存储设备可以位于沉浸式墙6000中,或者位于例如可通过局域网访问的本地服务器(未示出)上。

[0130] 游戏控制台7000将指令和用户输入参数发送到沉浸式墙6000。沉浸式墙6000可能根据从传感器2000、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6000还可以包括内部存储器以存储要显示的内容。沉浸式墙6000可以是OLED或LCD类型。它可以配备有一个或几个摄像头。

[0131] 从传感器2000接收的数据可以与照明条件、温度、用户的环境(例如,物体的位置)有关。沉浸式墙6000还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,以便提供关于用户情绪的反馈。用户输入设备3000的示例是手持设备,例如智能电话、遥控器和具有陀螺仪功能的设备。

[0132] 沉浸式墙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沉浸式视频数据(例如,解码它们并准备它们以用于显示)。传感器信号可以通过沉浸式墙的通信接口接收。该通信接口可以是蓝牙类型、WIFI类型或任何其他类型的连接,优选地是无线连接,但也可以是有线连接。沉浸式墙6000可以包括至少一个通信接口,以与传感器和因特网通信。

[0133] 图17示出了根据本公开实施例的用于对2D图片的当前块进行编码的示例性方法的框图,该2D图片是全向视频的投影。所述全向视频的至少一个图片被表示为3D表面,例如,如上所述的球体或立方体。然而,本原理可以应用于全向视频的任何3D表示。使用投影函数将3D表面投影到至少一个2D图片上。例如,这种投影函数可以是等矩形投影或其他类型的投影函数。然后,将2D结果图片划分成非重叠的像素块。这里公开了用于2D图片的至少一个当前块的方法,所述当前块要使用传统的邻域(例如图15中公开的邻域)并使用传统的2D视频编码方案进行编码。

[0134] 在块1700中,根据用于将3D表面投影到2D图片上的投影函数,为所述当前块确定所述2D图片的至少一个相邻块。所确定的相邻块在2D图片中不与所述当前块空间相邻,但是该相邻块在所述3D表面上与所述当前块空间相邻。

[0135] 根据本原理的一个实施例,投影函数是等矩形投影,3D表面是球面。位于2D图片右边界上的当前块的相邻块通过使用两个坐标之间的关系来确定,这两个坐标是如图13B所示的XY平面上的笛卡尔坐标(归一化)以及如图13C所示的球面上的角坐标,该关系如下:

[0136]  $y = \phi / \pi, -0.5 \leq y \leq 0.5, -\pi/2 \leq \phi \leq \pi/2$

[0137]  $x = \theta / 2\pi, 0 \leq x \leq 1, 0 \leq \theta \leq 2\pi,$

[0138] 其中  $(x, y)$  与2D图片的归一化XY平面上的点M的位置相对应,并且  $(\theta, \phi)$  是球面上的对应点M'的坐标。

[0139] 在图16A的情况下,对于2D图片的最后一列的块(例如,块F)的右相邻块,该块的左上角的点位于2D图片中的列索引w处,并且在XY平面上位于  $x=1$  处。该点在球面上的对应点具有角坐标  $(2\pi, \phi) = (0, \phi)$ 。因此,对于位于2D图片右边界(即2D图片的最后一列)上的当前块,相邻块被确定为与当前块位于同一行上的2D图片的第一个块。例如,如图16A所示,对



于当前块F和J,所确定的相邻块分别是A和G。

[0140] 根据本公开的一个实施例,位于2D图片右边界上的当前块的相邻块也可以是位于当前块行下方或上方的行上的、2D图片左边界上的相邻块。由于采用等矩形投影,所以这种相邻块在球面上至少部分地与当前块空间相邻。例如,对于图16A所示的块J,可以根据本原理将块A和/或K确定为相邻块,因为这些块在3D表面上至少部分地与块J空间相邻。然而,要被确定为相邻块,目标块(例如K)必须可用于对当前块进行编码,即,根据用于编码/解码的扫描顺序,相邻块必须在当前块之前被编码/解码。例如,对于相邻块K和当前块J,这种情况可能发生在以下情形下,即,当块K和J属于包括位于同一行编码单元上的一组块的编码单元时。

[0141] 根据另一实施例,当投影函数是立方体投影时,对于位于投影在2D图片上的立方体的当前面的边界上(例如,位于立方体的当前面的最后一列或第一列上)的当前块,相邻块被确定为位于立方体的另一面的边界上的块,该另一面在3D表面上与当前面共用一条边。例如,图16B示出了2D图片,其上根据图16C所示的布局重新排列了立方体的6个投影面。对于当前块C、A、E和K,根据本原理确定的相邻块分别为D、F、I和H。为了根据立方体投影确定相邻块,可以使用下面公开的图14C所示的XY平面中的点的笛卡尔坐标与立方体上的点的笛卡尔坐标之间的关系:

$$[0142] \quad f \left\{ \begin{array}{l} \text{左: } x < w, y > h: u = \frac{2x}{w} - 1, v = \frac{2(y-h)}{h} - 1, k = 0 \\ \text{前: } w < x < 2w, y > h: u = \frac{2(x-w)}{w} - 1, v = \frac{2(y-h)}{h} - 1, k = 1 \\ \text{右: } 2w < x, y > h: u = \frac{2(x-2w)}{w} - 1, v = \frac{2(y-h)}{h} - 1, k = 2 \\ \text{底: } x < w, y < h: u = \frac{2y}{h} - 1, v = \frac{2(w-x)}{w} - 1, k = 3 \\ \text{后: } w < x < 2w, y < h: u = \frac{2y}{h} - 1, v = \frac{2(2w-x)}{w} - 1, k = 4 \\ \text{顶: } 2w < x, y < h: u = \frac{2y}{h} - 1, v = \frac{2(3w-x)}{w} - 1, k = 5 \end{array} \right.$$

[0143] 其中相应的布局在图16C中示出。坐标k表示面的编号,(u,v),(其中 $u, v \in [-1, 1]$ ),表示该面上的坐标。立方体的每个面具有宽度w和高度h。

[0144] 在块1701中,一旦根据本原理确定了至少一个相邻块,则使用至少所确定的相邻块将当前块编码到所述比特流中。在编码当前块期间,所有编码模块或仅其中一些编码模块可以使用所确定的相邻块,如下面将详细描述。

[0145] 在块1702中,将与投影函数有关的至少一项信息编码到比特流中。这样的信息项允许向解码器指示用于将3D表面投影到2D图片上的投影函数的种类。因此,解码器可以确定当前块在执行编码期间的邻域,并使用该邻域。

[0146] 根据不同的变型,与投影函数有关的信息项可以编码在以下元素中:(例如由

H.264/AVC标准或HEVC标准定义的)序列参数集语法元素、或(例如由H.264/AVC标准或HEVC标准定义的)图片参数集语法元素、或(例如由H.264/AVC标准或HEVC标准定义的)与所述2D图片对应的切片头部语法元素。与投影函数有关的信息项可以编码在任何合适的语法元素中,以允许在图片或序列级别通知这样的信息项。

[0147] 图18是示出示例性视频编码器400的示意框图。根据本原理的实施例,这样的视频编码器400将表示全向视频的投影的一组图片编码到比特流中。视频编码器400被公开为符合HEVC编码器,但是本原理可以应用于将视频处理为2D图片序列的任何2D视频编码方案。

[0148] 典型地,如图18所示,视频编码器400可以包括用于基于块的视频编码的若干模块。表示来自全向视频的投影图片的待编码的2D图片I被输入到编码器400。

[0149] 首先,细分模块将图片I划分为一组像素单元。

[0150] 取决于所使用的视频编码标准,由细分模块传送的像素单元可以是(例如H.264/AVC中的)宏块(MB)或者(例如HEVC中的)编码树单元(CTU)。

[0151] 根据HEVC编码器,编码树单元包括亮度样本的编码树块(CTB)、色度样本的两个编码树块、以及关于编码树块的进一步细分的对应语法元素。亮度样本的编码树块可具有 $16 \times 16$ 像素、 $32 \times 32$ 像素或 $64 \times 64$ 像素的大小。可以使用树结构和类似四叉树的信令将每个编码树块进一步细分为更小的块(称为编码块CB)。四叉树的根与编码树单元相关联。亮度编码树块的大小是亮度编码块的最大支持大小。一个亮度编码块和通常两个色度编码块形成了编码单元(CU)。编码树单元可以包含一个编码单元,或者可以被分割以形成多个编码单元,并且每个编码单元具有相关联的、划分为预测单元(PU)和变换树单元(TU)的分区。在编码单元级别决定是使用帧间图片预测还是帧内图片预测来对图片区域进行编码。预测单元分区结构的根位于编码单元级别。根据基本预测类型决定,然后可以进一步划分亮度和色度编码块的大小,并根据亮度和色度预测块(PB)进行预测。HEVC标准支持从 $64 \times 64$ 个样本到 $4 \times 4$ 个样本的可变预测块大小。使用块变换对预测残差进行编码。变换单元(TU)树结构的根位于编码单元级别。亮度编码块残差可以与亮度变换块相同,或者可以进一步划分为更小的亮度变换块。这同样适用于色度变换块。变换块可具有 $4 \times 4$ 、 $8 \times 8$ 、 $16 \times 16$ 或 $32 \times 32$ 个样本的大小。

[0152] 编码过程在下面被描述为应用于称为块BLK的像素单元。这样的块BLK可以对应于宏块或编码树单元、或来自上述多种单元之一的任何子块、或者包括亮度样本和色度样本(或仅包括亮度样本)的图片I的任何其他细分局。

[0153] 下面描述的编码和解码过程用于说明目的。根据一些实施例,相对于以下模块,可以添加、移除或改变编码或解码模块。然而,本文公开的原理仍然可以应用于这些实施例。

[0154] 在这里,在等矩形投影的情况下公开了本原理。然而,可以使用其他投影函数。因此,根据该投影函数,确定根据块1700确定的位于2D图片边界处的当前块的相邻块。

[0155] 编码器400如下对图片I的每个块执行编码。编码器400包括模式选择单元,该模式选择单元用于例如基于速率/失真优化为待编码的图片的块BLK选择编码模式。这种模式选择单元包括:

[0156] -运动估计模块,用于估计待编码的图片的一个当前块与参考图片之间的运动,

[0157] -运动补偿模块,用于使用所估计的运动来预测当前块,

[0158] -帧内预测模块,用于在空间上预测当前块。

[0159] 模式选择单元还可以根据例如速率/失真优化来决定是否需要块进行细分。在需要细分情况下,模式选择单元然后对块BLK的每个子块进行操作。

[0160] 模式选择单元可以应用关于图17公开的原理,以用于导出位于2D图片右边界上的当前BLK的相邻块。

[0161] 根据变型,当确定用于对当前块BLK的帧内预测编码模式进行编码的最可能模式的列表时,按照所公开的原理执行操作。根据该变型,帧内预测模式编码在预测单元级别执行,因此这里的当前块BLK对应于预测单元(如图23所示的当前PU)。如图27所示,HEVC标准规定了与33个方向取向相对应的33个方向预测模式(索引从2到34)、平面预测模式(索引0)和DC预测模式(索引1),从而产生由35种可能的帧内预测模式组成的集合,以用于对当前预测单元进行空间预测。为了减少用于编码当前预测单元的帧内预测模式的通知所需的比特率,构造了最可能模式(MPM)列表。MPM列表包括用于待编码的当前块的三个最可能的帧内预测模式。根据在对当前块的相邻块进行编码时使用的帧内预测模式,来确定这三个MPM。根据HEVC,仅考虑当前块的左边和上方的相邻块,它们分别是图23(a)所示的当前PU的块A和B。如果这两个块A和/或B中的任何一个都不可用或不是帧内编码的块,则假定该块采用DC预测模式。在下文中,块A的帧内预测模式表示为 $m_A$ ,块B的帧内预测模式表示为 $m_B$ 。

[0162] 在HEVC中,MPM的集合构造如下:

[0163] (HEVC\_1) 如果 $m_A$ 与 $m_B$ 不相等,则 $MPM[0] = m_A$ ,  $MPM[1] = m_B$ 。该集合的表示为MPM[2]的第三个最可能的模式确定如下:

[0164] 如果 $m_A$ 和 $m_B$ 都不是平面模式(图27中的索引0),则 $MPM[2] =$ 平面模式(0),

[0165] 否则,如果二者中的一个(即 $m_A$ 或 $m_B$ )是平面模式,但 $m_A$ 和 $m_B$ 都不是DC模式,则 $MPM[2] =$ DC模式(1),

[0166] 否则,如果 $m_A$ 和 $m_B$ 中的一个平面模式并且另一个是DC模式,则 $MPM[2] =$ 垂直角度帧内预测模式(图27中的方向模式26)。

[0167] (HEVC\_2) 如果 $m_A$ 与 $m_B$ 相等但它们与平面模式或DC模式不同,则 $MPM[0] = m_A$ ,  $MPM[1] = m_{A-}$ ,并且 $MPM[2] = m_{A+}$ ,其中 $m_{A-}$ 和 $m_{A+}$ 表示由HEVC标准规定的块A的帧内预测模式的两个相邻角度模式,否则, $MPM[0] =$ 平面模式(HEVC的索引0),  $MPM[1] =$ DC模式(HEVC的索引1),并且 $MPM[2] =$ 垂直角度帧内预测模式“模式26”(图27中的方向模式26)。 $+$ 和 $-$ 指的是位于 $m_A$ 的当前角度方向两侧的角度方向。作为示例,如果 $m_A$ 等于图17上的索引14的模式,则 $m_{A-}$ 等于索引13的模式,并且 $m_{A+}$ 等于索引15的模式。有两种特殊情况:模式2和34。如果 $m_A$ 为2,则 $m_{A-}$ 为33, $m_{A+}$ 为3。如果 $m_A$ 为34,则 $m_{A-}$ 为33, $m_{A+}$ 为3。

[0168] 根据本原理,仅针对位于图片右侧的块,修改最可能模式集的构造。

[0169] 如图23(b)所示,当前块的新的相邻块被确定为当前块右侧的块C,即,根据投影函数,该相邻块是与当前块所属的当前CTU位于同一行的第一个CTU中的块。由于等矩形投影的连续性,沿着一行的第一个和最后一个CTU是空间邻居。

[0170] 然后,除了当前块的上方块和左侧块之外,右侧的相邻块也用于确定MPM列表。编码器/解码器保存一行的第一个CTU中的帧内编码的(左)边界块的帧内预测模式,并将它们用于编码最后一个CTU中的(右)边界块。存储器需求非常低,因为只需要保存一个CTU中的边界块的信息。当前块的MPM集现在基于在如图23(b)所示的三个相邻块(A、B和C)中使用的预测模式。MPM集构造如下:

[0171] 如果 $m\_A$ 与 $m\_B$ 不相等,但是 $C$ 的帧内预测模式(表示为 $m\_C$ )等于 $m\_A$ 或 $m\_B$ ,则使用规则HEVC\_1构造该集合。但是如果 $m\_C$ 不等于它们中的任何一个,即 $m\_C$ 与 $m\_A$ 和 $m\_B$ 都不同,则 $MPM[0]=m\_A$ , $MPM[1]=m\_B$ ,并且 $MPM[2]=m\_C$ 。

[0172] 如果 $m\_A$ 与 $m\_B$ 相等且 $m\_C$ 等于它们两者,则使用规则HEVC\_2构造该集合,

[0173] 否则,如果 $m\_A$ 与 $m\_B$ 相等但 $m\_C$ 不等于它们,则使用规则HEVC\_1构造该集合,其中 $B$ 由 $C$ 替换。

[0174] 对于色度预测,HEVC色度预测可以对于当前块保持不变,因为色度的预测模式并不基于相邻块的预测模式来导出。

[0175] 用于对当前块的帧内预测模式进行编码的其余过程仍然与HEVC中定义的相同。如果当前块的帧内预测模式属于MPM集,则设置标志`prev_intra_luma_pred_flag`,并且称为`mpm_idx`的另一个标志通知来自MPM集的候选者。如果未设置标志`prev_intra_luma_pred_flag`,则标志`rem_intra_luma_pred_mode`通知来自剩余的32个预测模式中的特定模式。

[0176] 当对当前块进行帧内编码时,通过根据为当前块选择的帧内预测模式执行帧内预测来计算预测块。这种过程是本领域技术人员公知的,不再进一步讨论。

[0177] 根据另一变型,在对当前块进行帧间预测时,当导出用于对当前块的运动矢量进行编码的运动矢量预测值时,实施所公开的原理。在对当前块进行帧间预测时,根据可用的预测方向,使用参考图片列表10和/或11中存在的参考图片来估计其运动矢量。通过使用估计的运动矢量对当前块进行运动补偿来计算预测块。然后将当前块的运动矢量编码到比特流中。

[0178] HEVC在对具有帧间预测的CU的运动矢量进行编码之前使用高级运动矢量预测(AMVP)。在H.264中,根据相邻的运动矢量构造单个运动矢量作为当前运动矢量的预测,与此不同,在HEVC中,使用来自如图24(a)所示的五个空间相邻块的运动矢量和位于相同位置的时间运动矢量,获得包括有两个运动矢量的集合。按如下方式选择这两个候选运动矢量A和B。

[0179] 基于空间邻居A0和A1的运动矢量构造候选运动矢量A,并且基于空间邻居B0、B1和B2的运动矢量构造候选矢量B,这些空间邻居也称为候选块。这是一个两遍过程。在第一遍中,检查任何候选块是否具有与当前块的参考索引相等的参考索引。依次检查A0和A1。将第一个运动矢量作为候选A。在来自A0和A1的两个参考索引都指向与当前块的参考索引不同的参考图片的情况下,不能按原样使用相关联的运动矢量。因此,在第二遍中,首先检查当前参考图片(即当前块的参考图片)和候选参考图片(即候选块的参考图片)(按顺序搜索A0然后搜索A1)是否都是短期的。如果验证了检查,即如果当前参考图片和候选参考图片都是短期的,则缩放相应候选块的运动矢量并将其用作候选运动矢量A。缩放因子取决于候选参考图片与当前图片之间的时间距离,并且还取决于当前参考图片与当前图片之间的时间距离。因此,在A0具有与当前块的参考索引相等的参考索引的情况下,在第一遍中不需要检查A1。将A0的运动矢量作为候选A。如果A0的参考索引与当前块的参考索引不同,则检查A1是否具有与当前块的参考索引相等的参考索引。如果是这种情况,则将A1的运动矢量作为候选A,否则应用第二遍过程。

[0180] 对于候选运动矢量B,在第一遍中如A0和A1那样按顺序搜索候选块B0、B1和B2。具有与当前块相同的参考索引的第一个候选块被用作运动矢量B。如果A0和A1不可用或者是

帧内预测,则候选A被设置为等于B。在这种情况下,在第二遍中,搜索具有与当前块相同的参考索引的第二候选块,如果找到,则将其运动矢量用作候选B。否则,在当前参考图片和候选参考图片都是短期图片的情况下,计算缩放的运动矢量并将其用作候选B。在第一遍没有找到具有与当前块相同的参考索引的候选的情况下,如果A0和A1不可用或者是帧内预测,则执行第二遍过程。在这种情况下,在当前参考图片和候选参考图片都是短期图片的情况下,计算缩放的运动矢量并将其用作候选B。因此,仅当块A0和A1不包含任何运动信息时才执行第二遍过程。仅当两个空间候选块不可用时或者它们是相同的时,才考虑时间候选块。

[0181] 在这两个运动矢量中,选择其中一个作为用于预测当前运动矢量的候选运动矢量。使用与list\_0和list\_1参考图片相对应的标志mvp\_l0\_flag和mvp\_l1\_flag来指示所选择的运动向量。

[0182] 根据该实施例,为了选择这两个候选运动矢量,考虑位于右边界处的当前块的七个相邻块的运动矢量,如图24 (b) 所示。

[0183] 对于这样的当前块,如果使用传统的HEVC方法,则相邻块B0不可用。

[0184] 根据本文公开的原理,块B0、C0和C1是在帧的左边界处位于同一行上的第一个CTU的一部分。当对位于右边界处的当前块进行编码时,这些块已经被编码,并且它们的运动矢量信息是可用的。因此,它们的运动矢量信息可用于改善候选运动矢量的集合。

[0185] 根据该变型,对于候选A,依次考虑块A0、A1、C0和C1的运动矢量。

[0186] 除了考虑四个候选块之外,该算法与HEVC中的算法保持相同。

[0187] 在第一遍中,顺序检查A0、A1、C0和C1。将四个候选块中的第一个具有与当前块的参考索引相等的参考索引的块作为候选A。如果四个候选块中没有一个块具有相同的参考索引,则在第二遍中,检查当前参考图片和候选参考图片(以与上述相同的顺序)是否都是短期的。如果验证了检查,则缩放所考虑的候选块的运动矢量并将其用作候选运动矢量A。缩放因子取决于它们的参考图片与当前图片之间的时间距离,并且还取决于当前块的参考图片与当前图片之间的时间距离。该缩放如在HEVC中那样完成。

[0188] 对于候选B,算法保持与HEVC标准中的相同。在第一遍中,以与检查A0和A1相同的方式顺序检查候选B0至B2。然而,仅当块A0和A1不包含任何运动信息(即块A0和A1不可用或者是帧内预测)时,才执行第二遍过程。

[0189] 1位语法元素mvp\_l0\_flag和mvp\_l1\_flag编码保持与HEVC中相同,因为不需要编码额外的信息。根据该实施例,产生预测值集合的预测值的数量增加,并且集合的构造方式发生了修改,但是集合中的预测值的数量保持相同,因此不需要编码额外的信息。

[0190] 根据另一变型,当估计位于2D图片边界上的当前块的运动向量时,运动估计模块可受益于估计运动的2D参考图片的左右边界处的连续性。在传统的基于块的运动估计技术中,搜索范围被截断,使得运动矢量不指向位于2D参考图片外部的不可用像素。搜索范围是搜索运动矢量的范围。

[0191] 根据该变型,在估计位于2D图片边界上的当前块的运动时,现在可以考虑全部搜索范围。通过使用来自对面边界的像素,可以对称地延伸2D参考图片的边界。结果,所估计的运动矢量可以从当前块指向位于参考图片外部的像素。位于参考图片外部的这种位置对应于2D参考图片的对称扩展块。在解码器侧,解码器仅需要对这样的运动矢量执行参考图片的对称扩展。

[0192] 根据另一变型,当在帧间预测合并模式下导出运动信息以用于对位于2D图片右边界处的当前块进行编码时,实施所公开的原理。除了AMVP之外,HEVC还使用预测块合并来减少对运动信息的编码。为此目的,HEVC编码器通过考虑相同的五个空间邻居(如图24(a)中所示)和可能的一个时间邻居的运动信息,来为要进行帧间编码的当前块生成合并列表。所选候选块的运动信息(预测方向(也称为参考图片的图片参考列表10和11)、图片参考列表中的参考图片的参考索引、以及运动矢量)直接用于预测当前块,而没有任何其他辅助信息。在该合并模式下,通过继承来自所选候选块的所有运动信息来预测当前块。因此,通过使用所继承的运动信息对当前块进行运动补偿来计算预测块。

[0193] 为了通知帧间预测合并模式,编码器使用称为merge\_flag的标志。如果merge\_flag是1,则语法元素merge\_idx通知所选择的候选块。合并列表中的候选块的最大数量使用称为cMax的参数来通知,该参数在切片报头中通知。合并列表可以包含从5个空间邻居导出的最多四个合并候选、一个时间候选、以及包括组合的双向预测候选和零运动矢量候选在内的附加合并候选。

[0194] 根据该实施例,位于右边界CU上的当前块的合并列表中的空间运动候选块的数量增加到5。该附加的运动候选块来自B0、C0和C1。该附加的运动候选块包括在由HEVC导出的其他四个候选块中。因此,对于边界CU,候选块的数量为cMax+1。

[0195] 回到图18,一旦为当前块BLK选择了编码模式,模式选择单元就传送预测块PRED和相应语法元素,该相应语法元素要在比特流中编码以在解码器处执行相同的块预测。

[0196] 然后,通过从原始块BLK中减去预测块PRED来获得残差块RES。

[0197] 然后,通过变换处理模块来变换残差块RES,该变换处理模块传送变换系数的变换块TCOEF。然后,每个传送的变换块TCOEF由量化模块量化,该量化模块传送量化后的残差变换系数的量化变换块QCOEF。

[0198] 然后,将语法元素和块QCOEF的量化后的残差变换系数输入到熵编码模块,以传送比特流STR的编码视频数据。

[0199] 根据另一变型,所公开的原理可以由熵编码模块使用。HEVC使用上下文算术熵编码,也称为CABAC。由熵编码模块执行的算术编码对整个比特流通过它们的联合概率(由(0, 1)中的区间表示)进行编码,该整个比特流是对要编码的符号(语法元素、量化的变换系数等)进行合适的二值化之后获得的。熵编码模块通过以下操作来执行算术编码:通过不同语法元素的上下文模型对符号的概率建模,并在编码每个比特之后更新模型状态。上下文模型基于邻域编码信息对概率进行初始化。

[0200] 在HEVC中,针对当前编码单元编码CU\_split\_flag和CU\_skip\_flag,以分别指示该编码单元是否被进一步划分以及是否跳过该编码单元(即,指示是否为当前编码单元编码其他信息)。用于编码这些标志的熵编码模块使用来自先前编码的相邻CU的信息,来选择用于编码这些标志的上下文模型。针对每种适用的初始化类型的三个上下文模型的集合是可用的。如图25(a)所示,基于相邻块A和B来决定当前块的CU\_split\_flag和CU\_skip\_flag的上下文模型。如果邻居可用并且其编码树深度大于当前块,则CU\_split\_flag上下文递增1。类似地,如果邻居可用并且其CU\_skip\_flag被设置,则CU\_skip\_flag的上下文递增1。

[0201] 根据本原理,对于位于2D图片右边界处的当前块,还考虑来自已经编码的作为左边界上的CTU的一部分的相邻块的信息。如图25(b)所示。对于位于2D图片右边界处的当前

块, `CU_split_flag`和`CU_skip_flag`均使用4个上下文模型。如在HEVC中那样,如果邻居(顶部、左侧、右侧)可用并且其编码树深度大于当前块,则`CU_split_flag`上下文递增1。类似地,如果邻居(顶部、左侧、右侧)可用并且其`CU_skip_flag`被设置,则`CU_skip_flag`的上下文递增1。

[0202] 回到图18,量化变换块`QCOEF`的量化后的残差变换系数由逆量化模块处理,该逆量化模块传送未量化的变换系数的块`TCOEFF'`。块`TCOEFF'`被传递到逆变换模块,以用于重建残差预测块`RES'`。

[0203] 然后,通过将预测块`PRED`与重建的残差预测块`RES'`相加,来获得块`BLK`的重建版本`REC`。重建的块`REC`存储在存储器中,供以后由图片重建模块用于重建图片`I`的解码版本`I'`。一旦图片`I`的所有块`BLK`都已被编码,图片重建模块就从重建块`REC`执行图片`I`的解码版本`I'`的重建。可选地,可以将去块滤波应用于重建的图片`I'`,以去除重建的块之间的块伪影。

[0204] 根据另一变型,对右边界CTU执行样本自适应偏移滤波的SAO模块可以使用所公开的原理。SAO是基于由编码器发送的查找表中的值,通过应用去块滤波器之后有条件地将偏移值添加到每个样本来修改解码后的样本的过程。基于每个CTU选择的滤波类型,基于区域执行SAO。

[0205] 在HEVC中,CTU可以使用三个选项来通知SAO参数:通过重新使用左侧CTU或顶部CTU的SAO参数(图26(a))或通过发送新的SAO参数。根据是使用左侧CTU还是顶部CTU的SAO信息,来设置称为`sao_merge_left_flag`和`sao_merge_top_flag`的两个标志。

[0206] 根据该实施例,为位于帧的右边界上的CTU添加称为`sao_merge_right_flag`的附加标志。位于同一行的左边界上的CTU用作右侧邻居。如果当前CTU使用右侧邻居的SAO信息,则设置`sao_merge_right_flag`。

[0207] HEVC使用一个上下文模型来编码`sao_merge_left_flag`和`sao_merge_top_flag`。在该变型中,使用该同一上下文来编码右边界CTU的`sao_merge_right_flag`。

[0208] 回到图18,一旦重建的图片`I'`已经被去块并且已经经过SAO滤波,则将所得到的重建图片添加到参考图片存储器中,供稍后用作参考图片,以编码要编码的图片集合中的后续图片。

[0209] 然后,通过数据网络发送通过上述编码处理生成的比特流,或者将其存储在用于沉浸式呈现从比特流STR解码的全向视频的存储器上。

[0210] 图19示出了根据本公开实施例的用于对2D图片的当前块进行解码的示例性方法的框图,该2D图片表示全向视频的使用投影函数的投影。这种方法可以例如通过诸如本文所公开的沉浸式系统的解码器(700)来执行。

[0211] 在块1900中,从表示全向视频的比特流中解码与所述投影函数有关的信息项。或者,解码器可以得知用于将全向视频投影到2D图片上的投影函数。例如,这种信息可以存储在解码器的存储器中。

[0212] 在块1901中,如图17中的块1700所公开的那样,对于所述2D图片的当前块,根据投影函数确定所述2D图片的至少一个相邻块。

[0213] 在块1902中,至少使用所确定的相邻块来解码当前块。在解码当前块期间,所确定的相邻块可以由解码器的所有解码模块或其中的一些解码模块使用。例如,这样的解码模块可以包括在包括以下模块的组中:

[0214] -用于确定最可能模式的列表的模块,该最可能模式的列表用于解码所述当前块的帧内预测模式,

[0215] -用于导出运动矢量预测值的模块,该运动矢量预测值用于重建所述当前块的运动矢量,

[0216] -用于在帧间预测合并模式下导出运动信息以重建所述当前块的模块,

[0217] -用于对所述当前块进行上下文算术熵解码的模块,

[0218] -用于样本自适应偏移的模块,该样本自适应偏移用于对所述当前块的至少一个样本进行滤波。

[0219] 根据本公开的实施例,图19所公开的方法的任一实施例可以在(例如在下文和图20中公开的)示例性解码器中实现,以用于解码表示全向视频的比特流。

[0220] 图20是示出适用于解码使用本原理编码的比特流的示例性视频解码器方法的示意性框图。比特流STR表示编码后的图片,该图片表示全向视频到2D图片上的投影,比特流STR包括表示所述2D图片的至少一个当前块BLK的编码数据。这样的当前块可以已经根据本公开的实施例进行了编码。

[0221] 根据一个实施例,比特流STR还可以包括表示与投影函数有关的信息项的编码数据。

[0222] 本文公开的视频解码器700根据HEVC视频编码标准执行图片的解码。然而,本原理可以容易地应用于任何视频编码标准。

[0223] 视频解码器700通过逐图片地从比特流中解码编码图片并且通过逐块地解码每个图片,来执行全向视频的重建。根据所使用的视频压缩方案,可以使用并行处理来基于图片或基于块解码比特流。因此,以如下方式从压缩比特流重建图片I'。

[0224] 编码数据被传递到视频解码器700的视频解码模块。如图20中所示,将编码数据传递到熵解码模块,该熵解码模块执行熵解码,并将量化变换系数的块QCOEF递送到逆量化模块,将语法元素传送到预测模块。根据本原理的实施例,熵解码模块可以执行如图19中的框1901和1902中所公开的处理,以导出用于执行当前块的CU\_split\_flag和/或CU\_skip\_flag的算术二进制解码的上下文模型。

[0225] 在熵解码之后,量化变换系数的块QCOEF由逆量化模块逆量化,以传送去量化变换系数的块TCOEF'。

[0226] 由逆变换模块对去量化变换系数的块TCOEF'进行逆变换,该逆变换模块传送残差预测块RES'。

[0227] 预测模块根据语法元素并使用运动补偿模块或帧内预测模块来构建预测块PRED,如果当前块已被帧间预测则使用运动补偿模块,或者,如果当前块已被空间预测则使用帧内预测模块。在当前块是位于2D图片边界上的块时,为了构建当前块的预测块PRED,预测模块可以执行如图19中的框1901和1902中所公开的处理,从而根据本原理确定用于当前块的相邻块,以用于以下目的:

[0228] -当残差运动矢量已被明确地编码到当前块的比特流中时,用于导出用于重建当前块的运动矢量的运动矢量预测值,或者,

[0229] -当已经使用帧间预测合并模式对当前块进行了编码时,用于导出当前块的运动信息,或者



- [0230] -当已经使用帧内预测模式对当前块进行了编码时,用于确定最可能模式的列表。
- [0231] 然后,通过将预测块PRED与重建的残差预测块RES'相加,来获得重建块REC。将重建块REC存储在存储器中,供图片重建模块稍后用于重建解码的图片I'。一旦图片I的所有块都被解码,图片重建模块就从重建块REC执行解码图片I'的重建。可选地,可以将去块滤波应用于重建的图片I',以去除重建的块之间的块伪影。
- [0232] 在根据本公开实施例的用于导出当前块的邻域的处理已经在SAO模块中的编码器处得到应用的情况下,SAO滤波也以与编码器中相同的方式在解码器处进行应用。因此,对于位于2D图片边界上的当前块,SAO模块可以执行如图19中的框1901和1902中所公开的处理,从而根据本原理确定当前块的相邻块。
- [0233] 然后,将重建的图片I'存储在参考图片存储器中,供稍后用作参考图片,以解码要解码的图片集中的后续图片。
- [0234] 然后,重建的图片I'被存储在存储器上,或由视频解码器设备700输出到如上所述的沉浸式呈现设备(10)。视频解码器装置700也可以包括在沉浸式呈现设备(80)中。在这种情况下,重建的图片I'由解码器装置输出到沉浸式呈现装置(80)的显示模块。
- [0235] 根据所实现的沉浸式呈现系统,所公开的解码器装置可以包括在诸如本文所公开的沉浸式呈现系统的任何一个处理设备中,例如,计算机(40)、游戏控制台(60)、智能电话(701)、沉浸式呈现设备(80)或沉浸式墙(6000)。
- [0236] 装置解码器700可以实现为硬件、软件、或硬件与软件的组合。
- [0237] 图21示出了根据实施例的用于编码全向视频的装置(400)的简化结构。这种装置被配置为实现用于根据本原理编码全向视频的方法,该方法已在上面参考图17和18进行了描述。
- [0238] 根据实施例,编码器装置包括处理单元PROC,其配备有例如处理器,并由存储在存储器MEM中的计算机程序PG驱动,并且实现用于根据本原理编码全向视频的方法。
- [0239] 在初始化时,计算机程序PG的代码指令例如被加载到RAM(未示出)中,然后由处理单元PROC的处理器执行。处理单元PROC的处理器根据计算机程序PG的指令实现上面已经描述的用于编码全向视频的方法的步骤。
- [0240] 编码器装置包括通信单元COMOUT,以将编码比特流STR发送到数据网络。
- [0241] 编码器装置还包括接口COMIN,以用于接收要编码的图片或要编码的全向视频。
- [0242] 图22示出了根据实施例的用于解码表示全向视频的比特流的装置(700)的简化结构。这种装置被配置为实现用于根据本原理解码表示全向视频的比特流的方法,该方法已在上面参考图19和20进行了描述。
- [0243] 根据实施例,解码器装置包括处理单元PROC,其配备有例如处理器,并由存储在存储器MEM中的计算机程序PG驱动,并且实现用于根据本原理解码表示全向视频的比特流的方法。
- [0244] 在初始化时,计算机程序PG的代码指令例如被加载到RAM(未示出)中,然后由处理单元PROC的处理器执行。处理单元PROC的处理器根据计算机程序PG的指令实现上面已经描述的用于解码表示全向视频的比特流的方法的步骤。
- [0245] 该装置可以包括通信单元COMOUT,以将视频数据的重建图片发送到呈现设备。
- [0246] 该装置还包括接口COMIN,用于从数据网络、网关或机顶盒接收要解码的表示全向

视频的比特流STR。

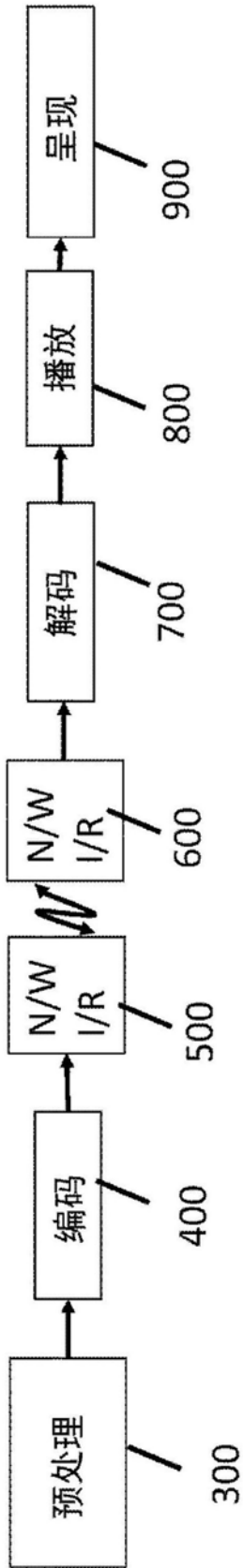


图1

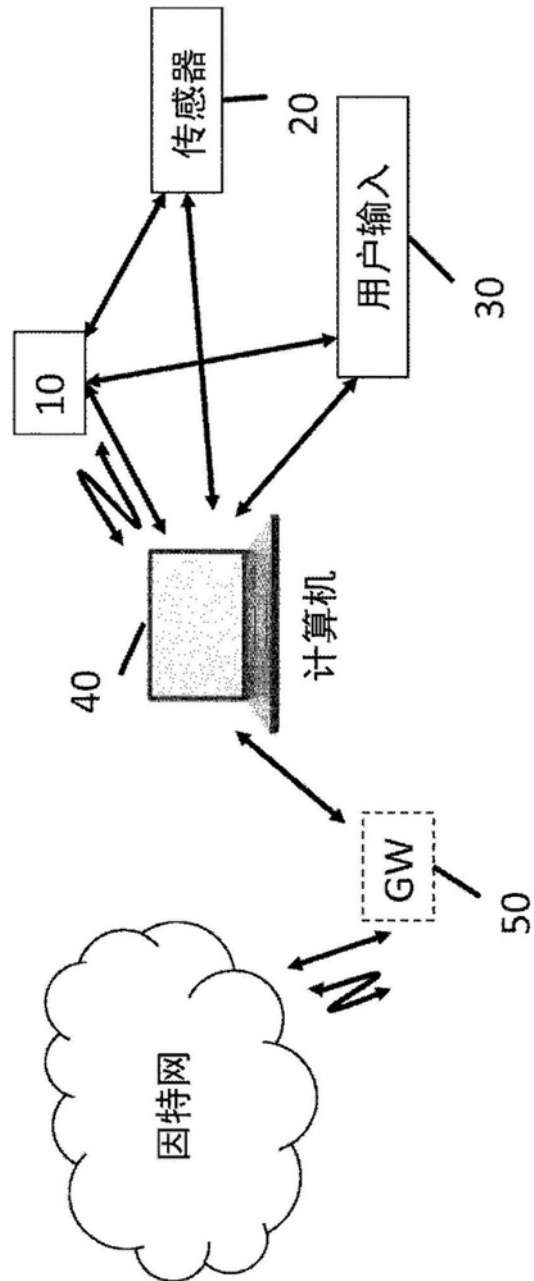


图2

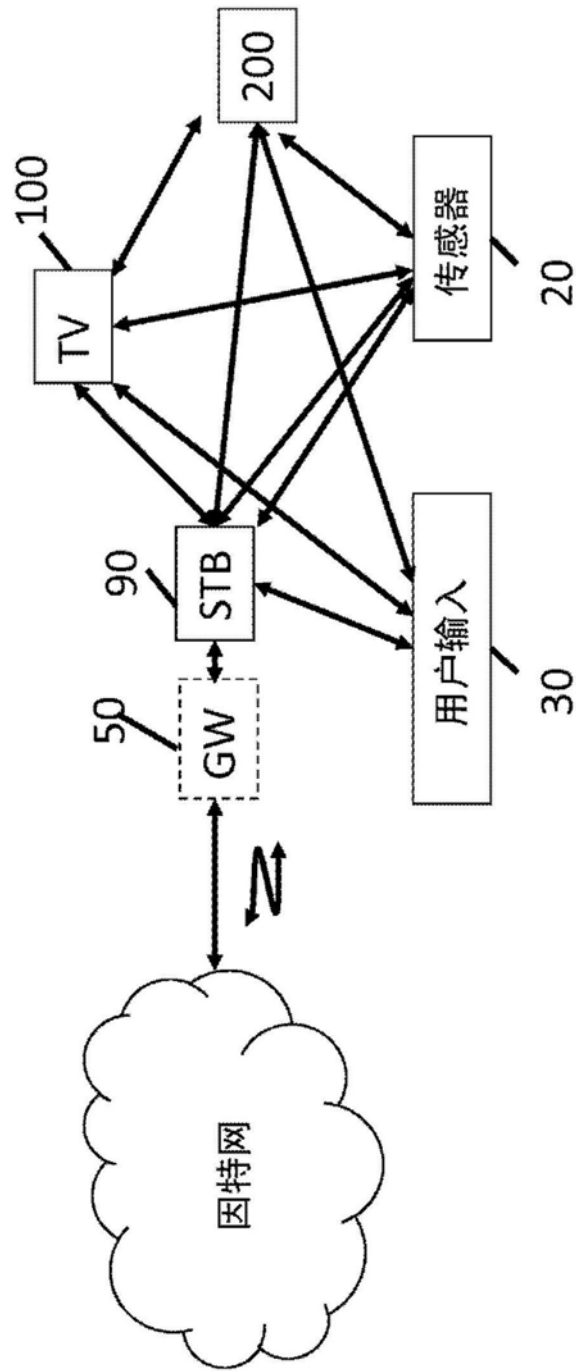


图3

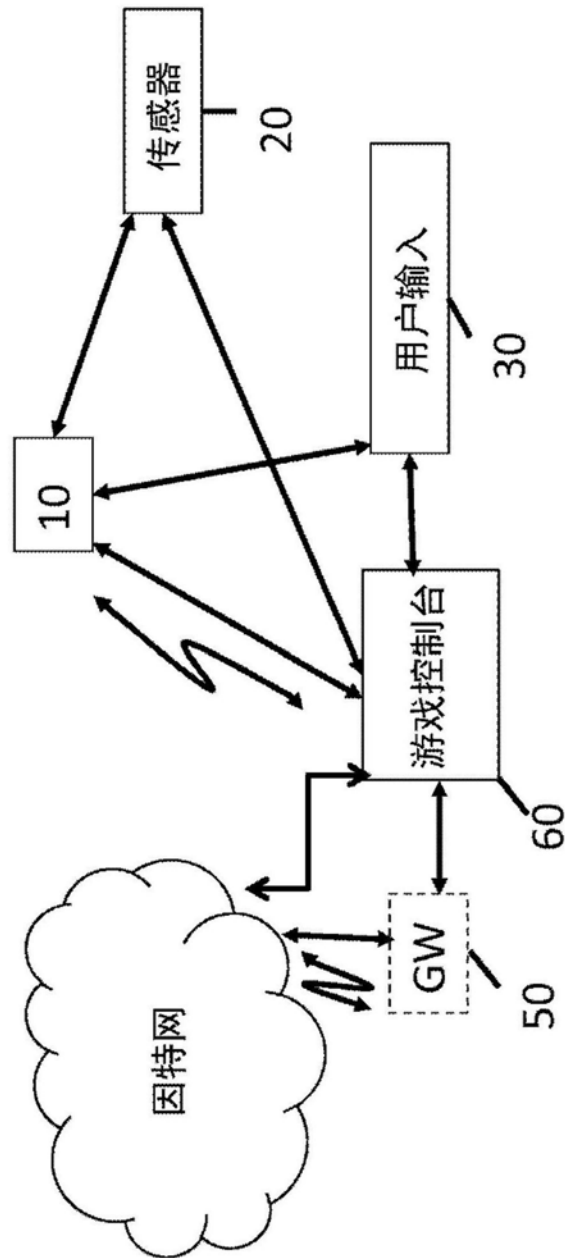


图4

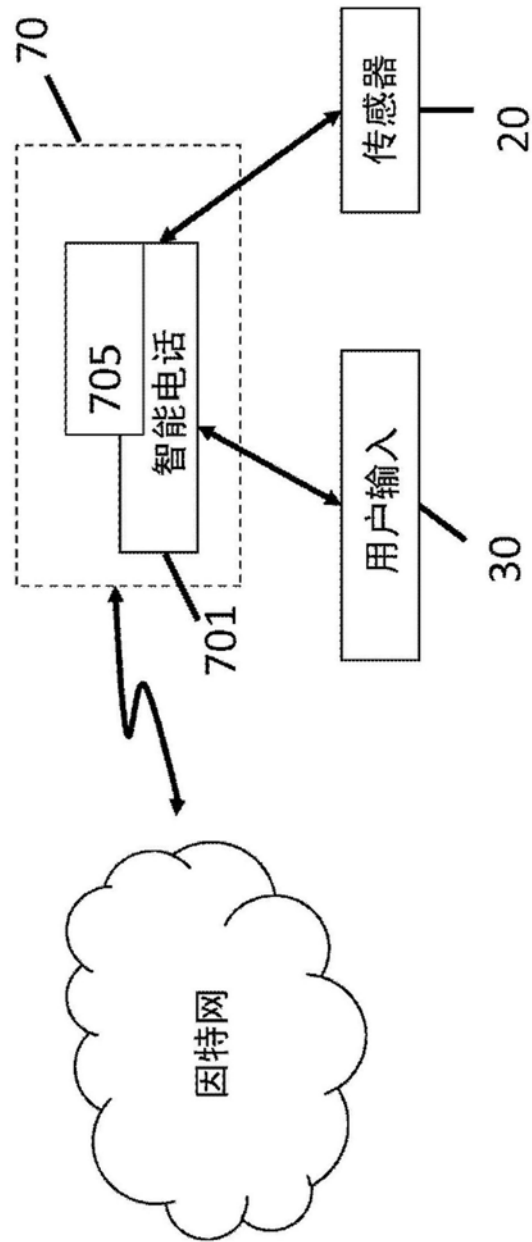


图5

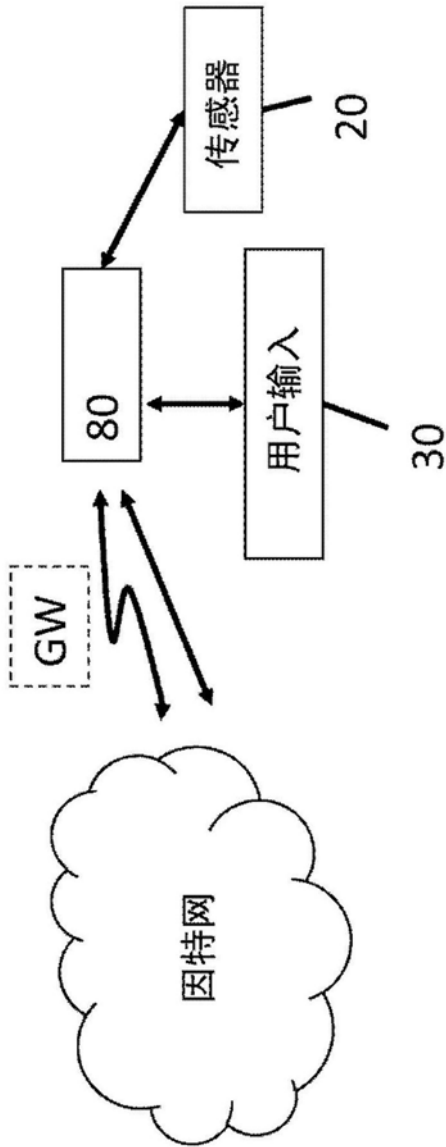


图6



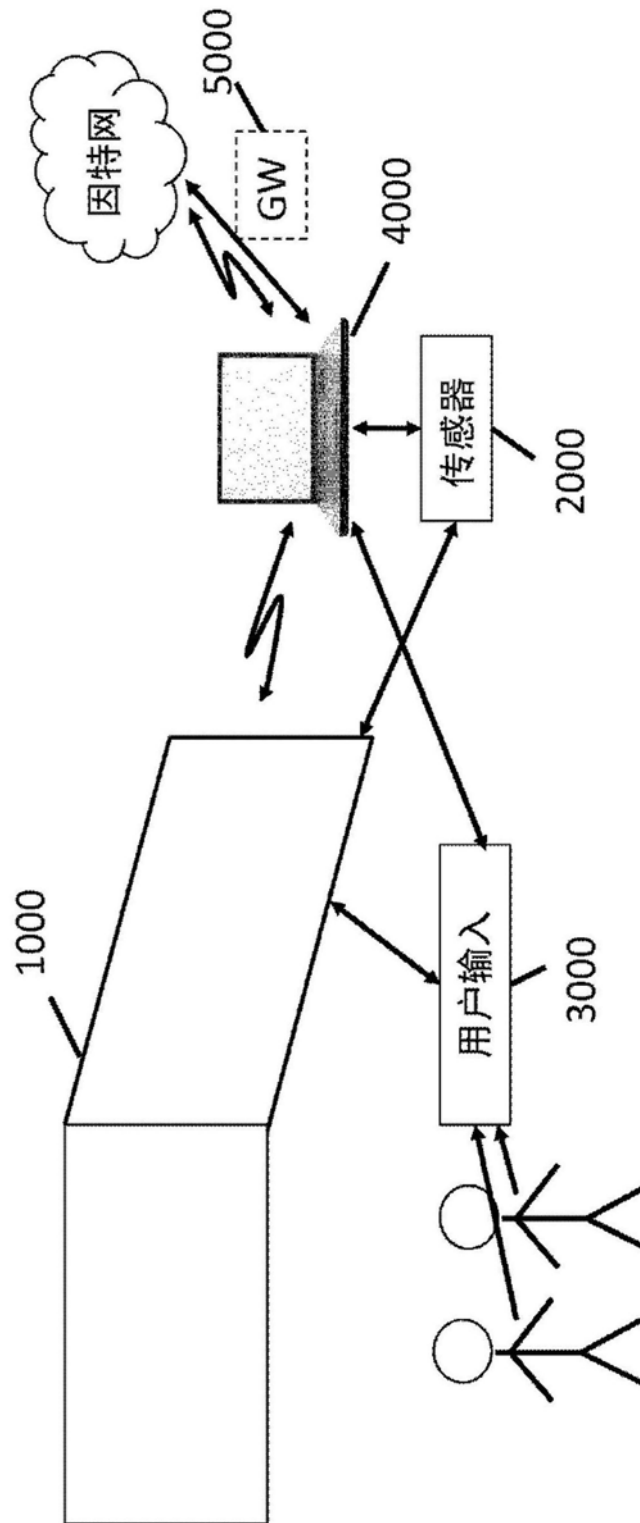


图7

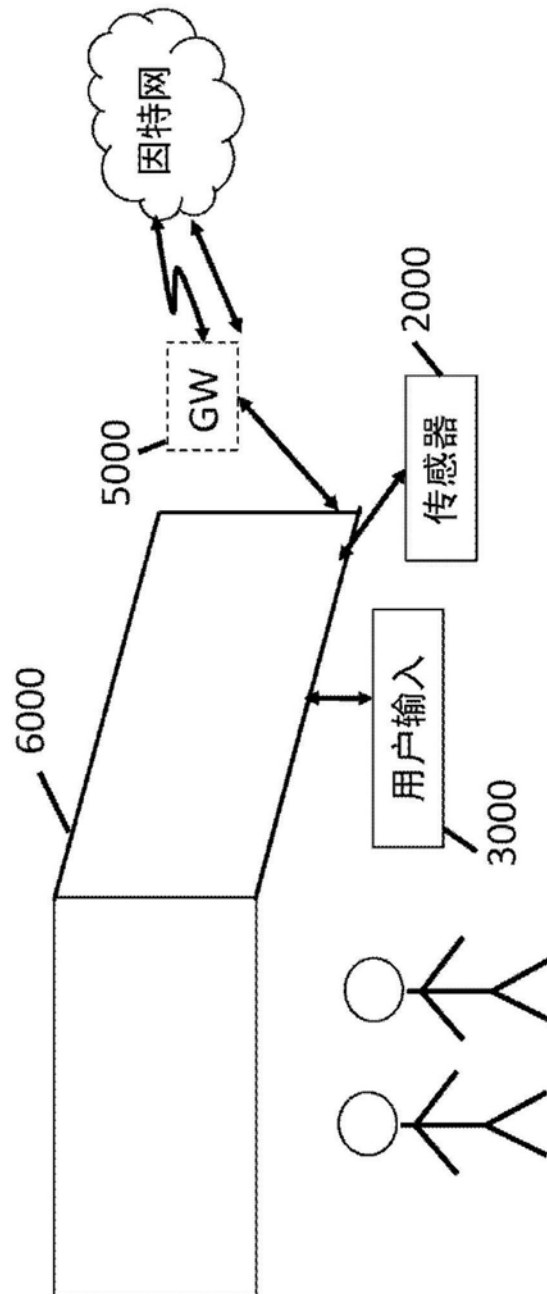


图8

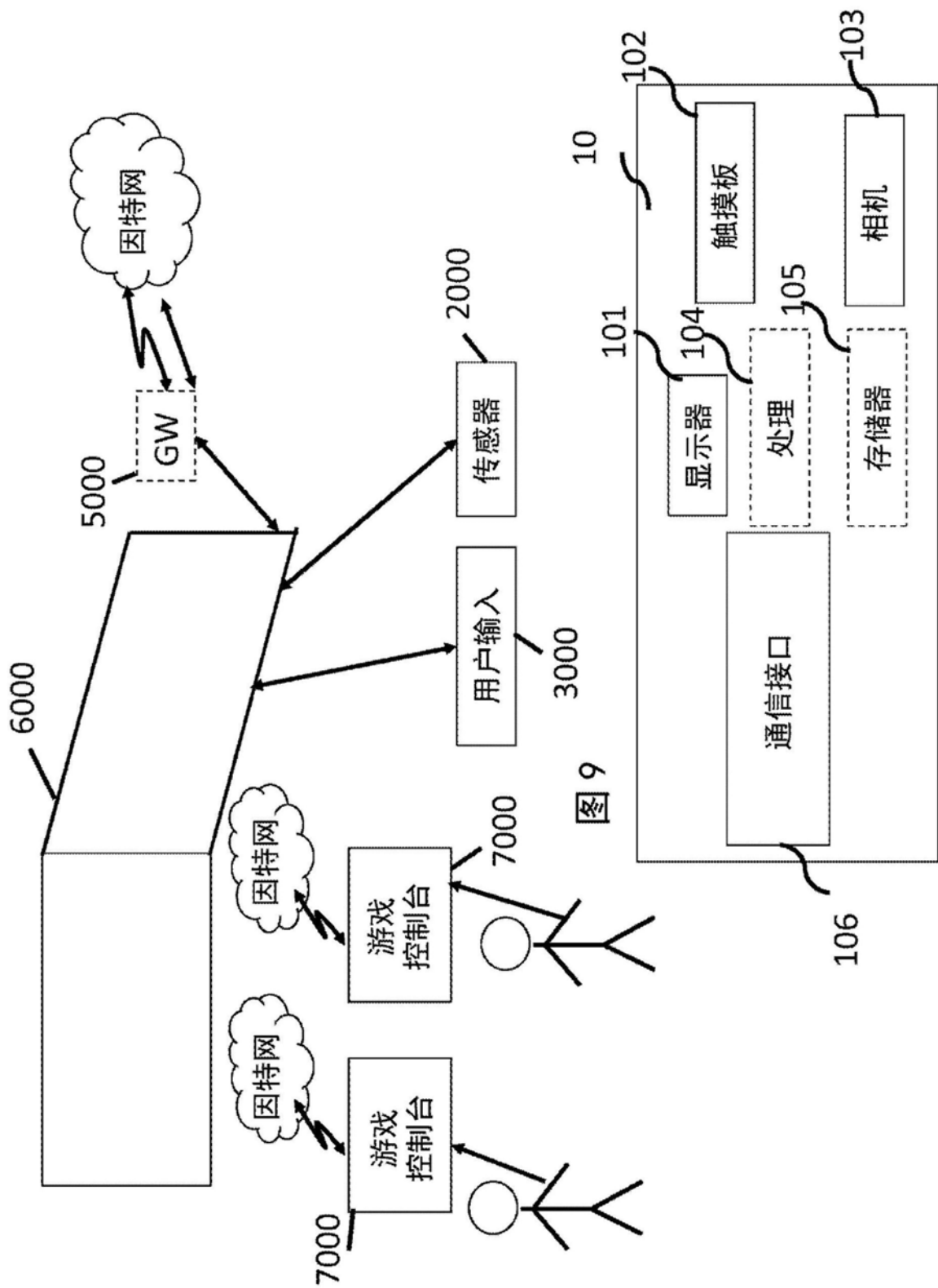


图 10

图 9

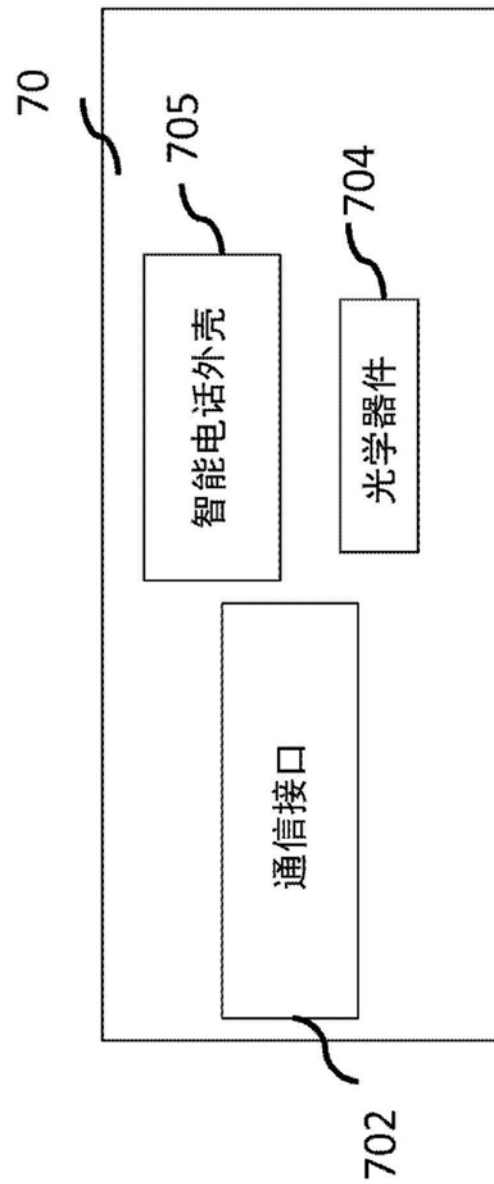


图11

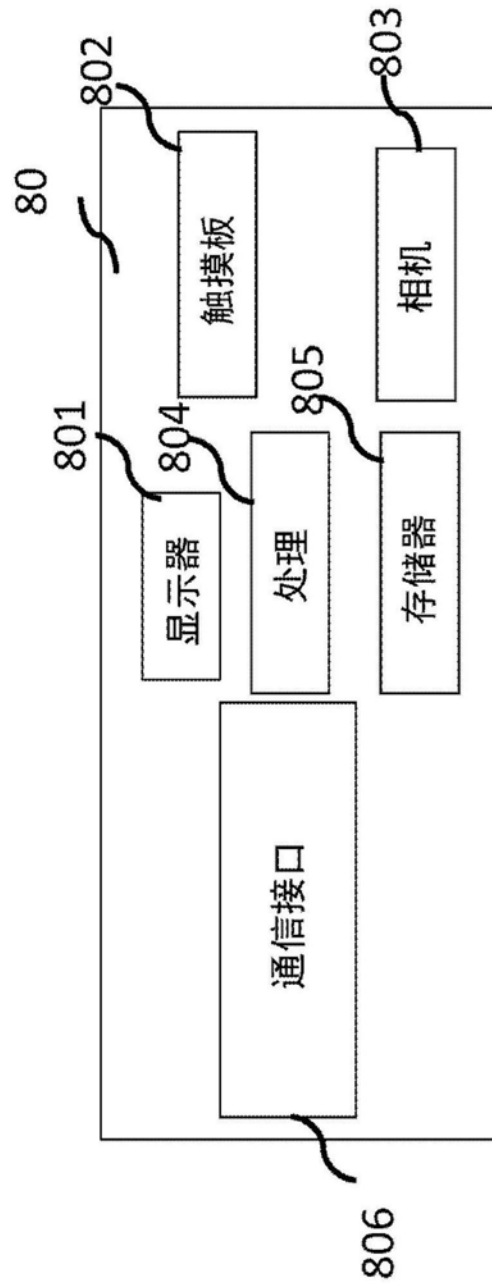


图12

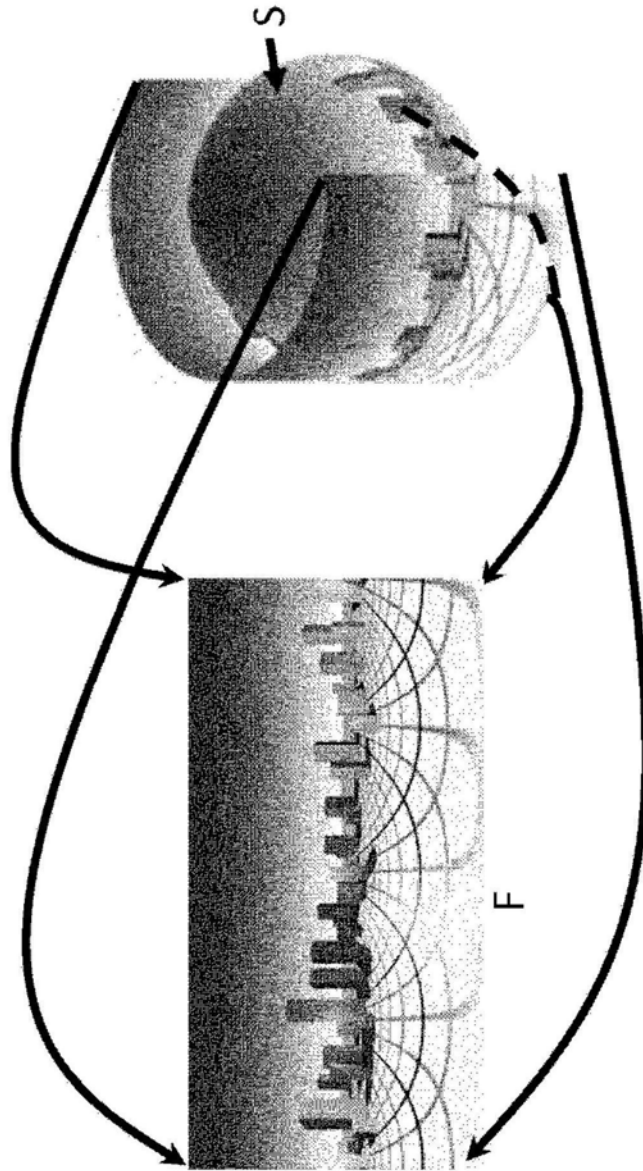


图13A

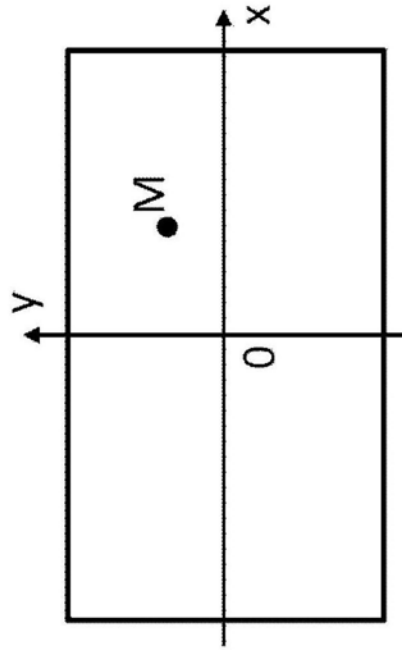


图13B

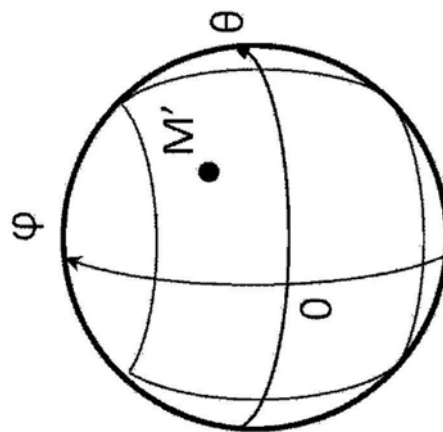


图13C

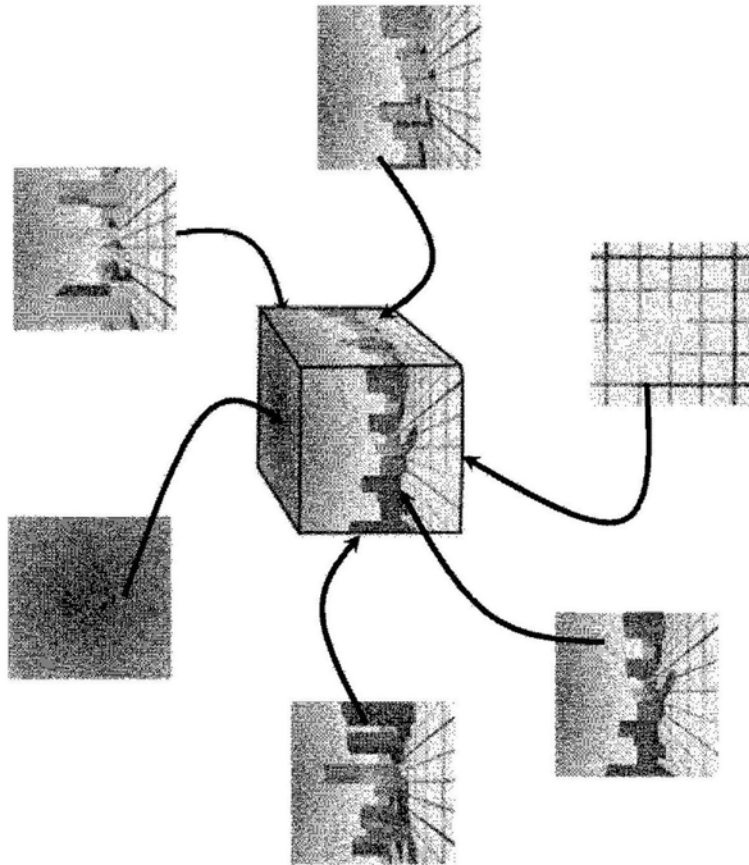


图14A

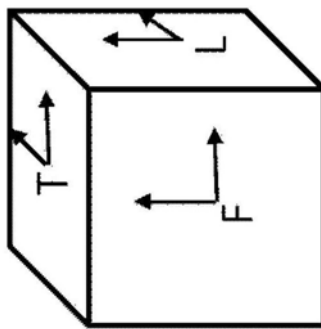


图14B



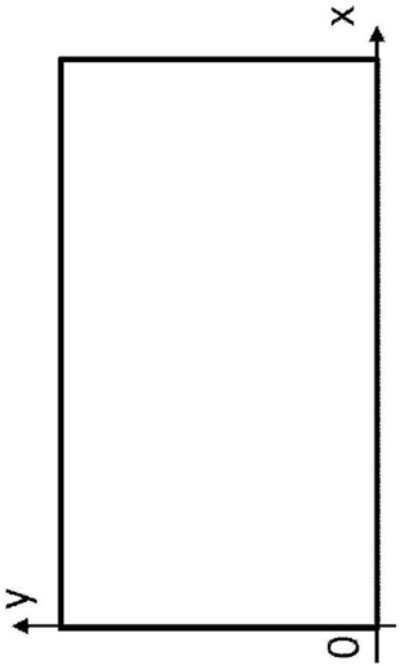


图14C

<div><math>v_R</math> 右</div>	<div><math>u_t</math> 后 <math>v_t</math></div>
<div><math>v_F</math> 前 <math>u_F</math></div>	<div><math>u_{Ba}</math> 后 <math>v_{Ba}</math></div>
<div><math>v_L</math> 左 <math>u_L</math></div>	<div><math>u_{Bo}</math> 后 <math>v_{Bo}</math></div>

图14D

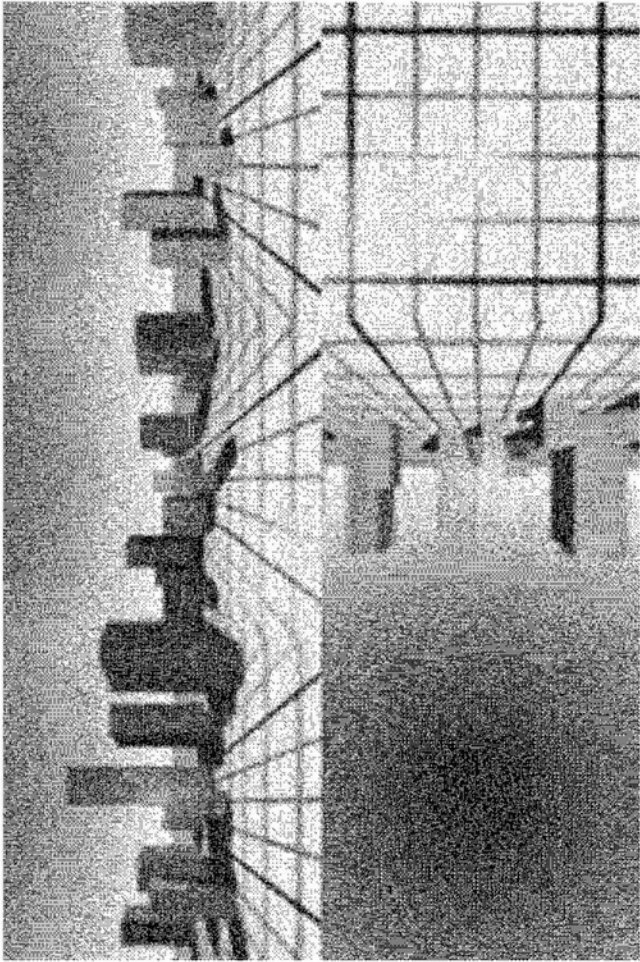


图14E

A	B	D
C	BK	

图15

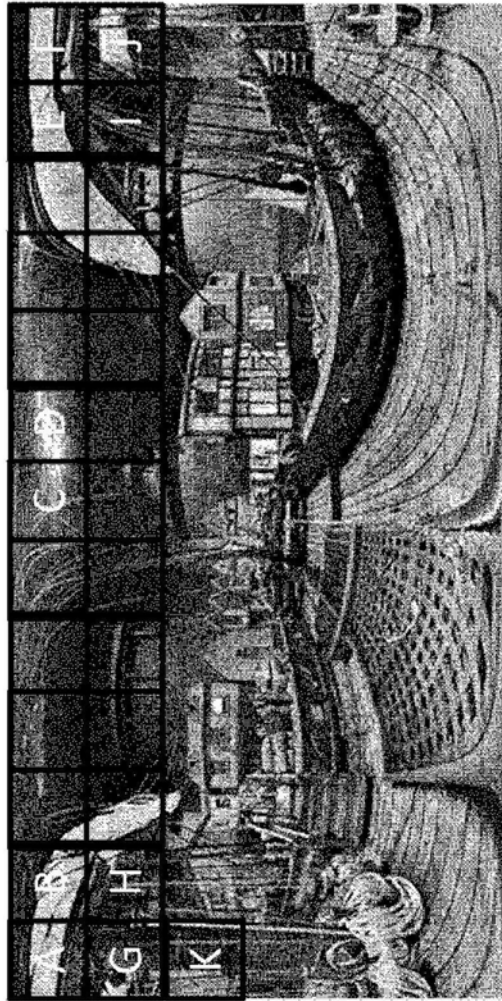


图16A

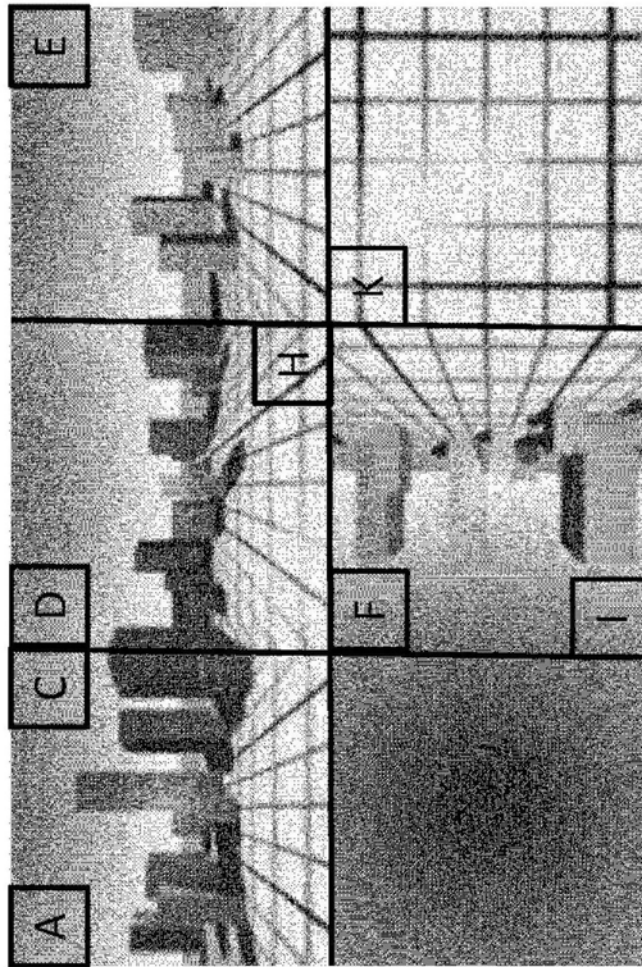


图16B

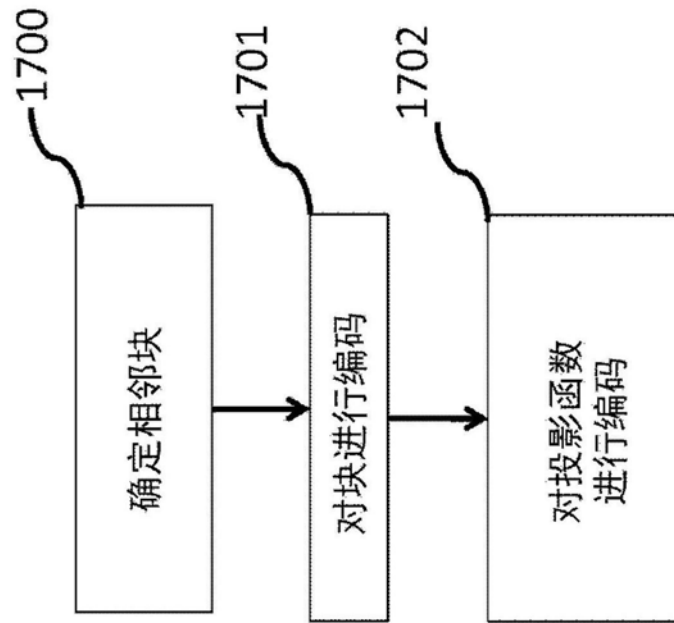


图17

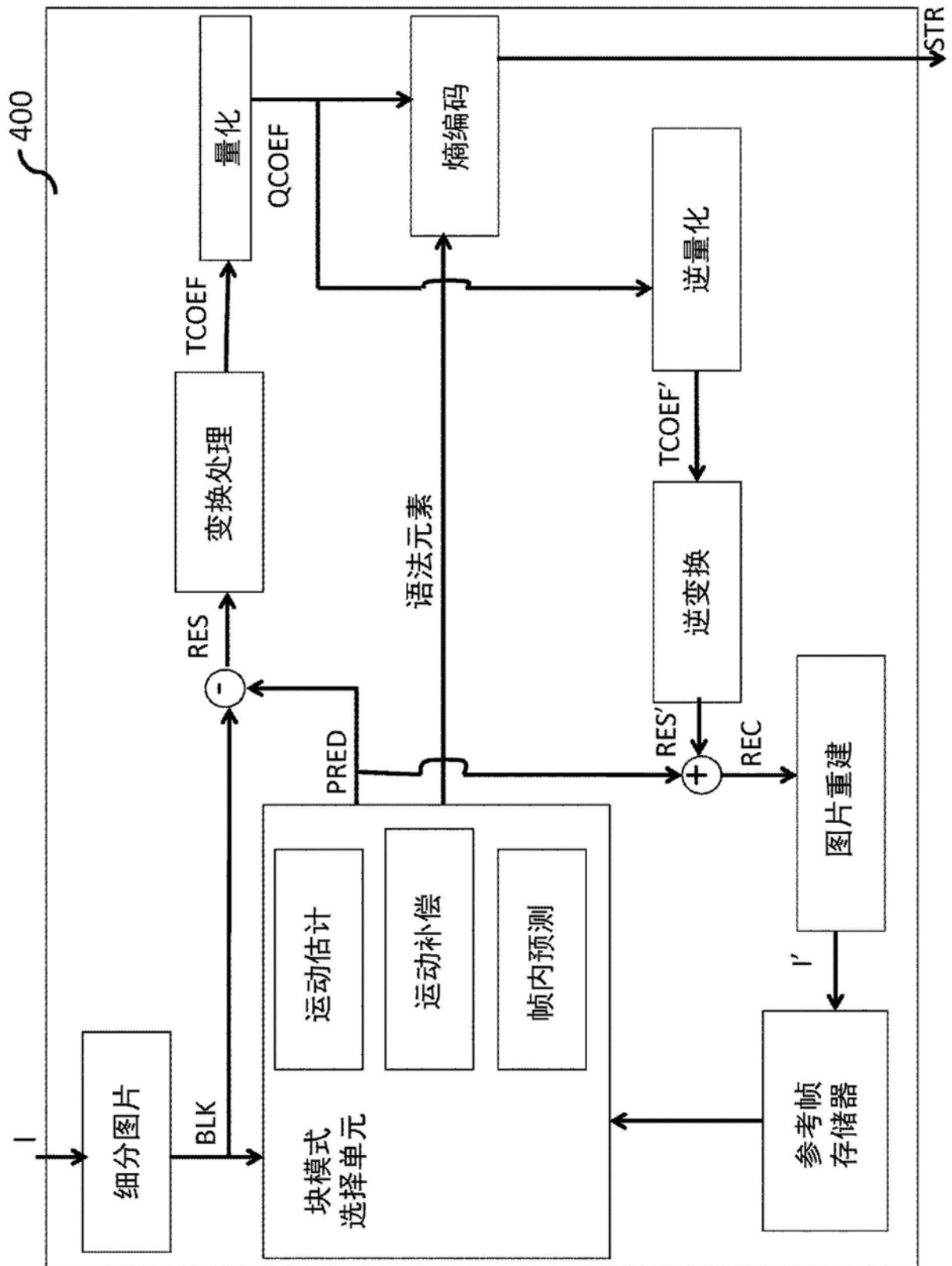


图18

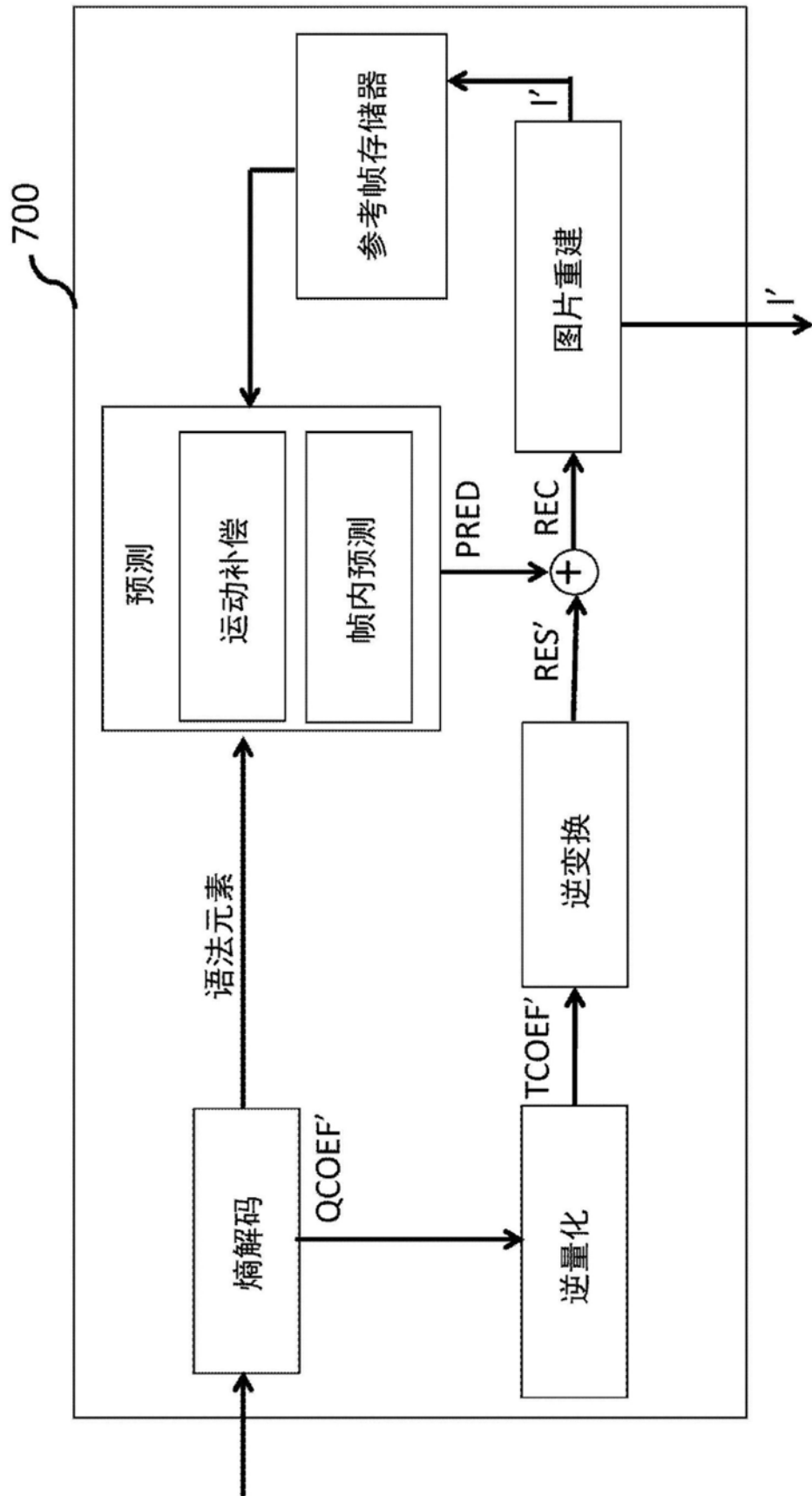


图20

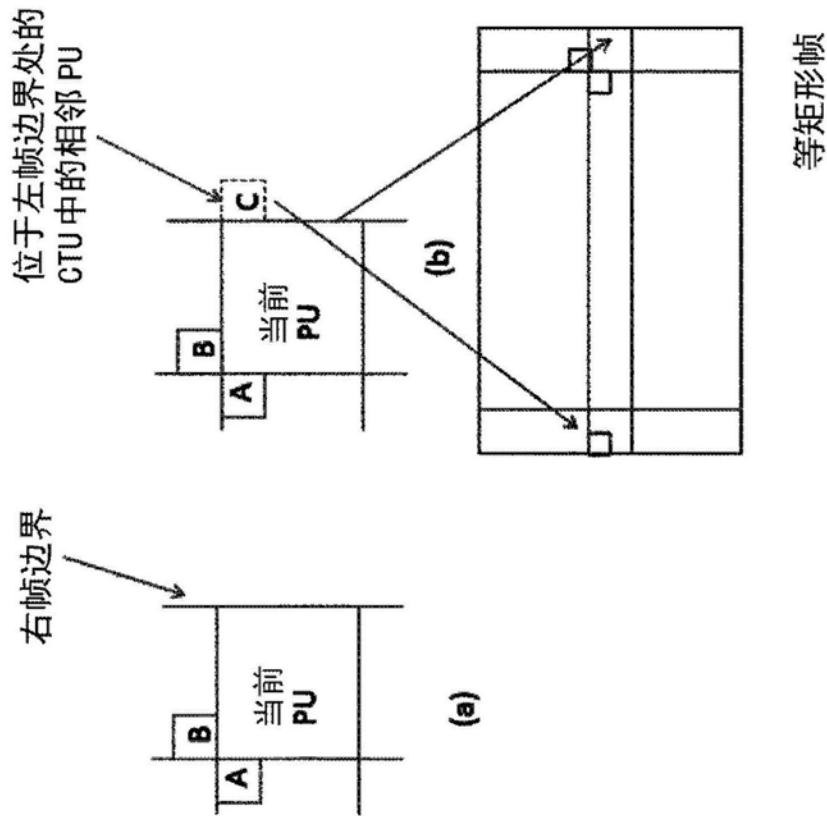


图23



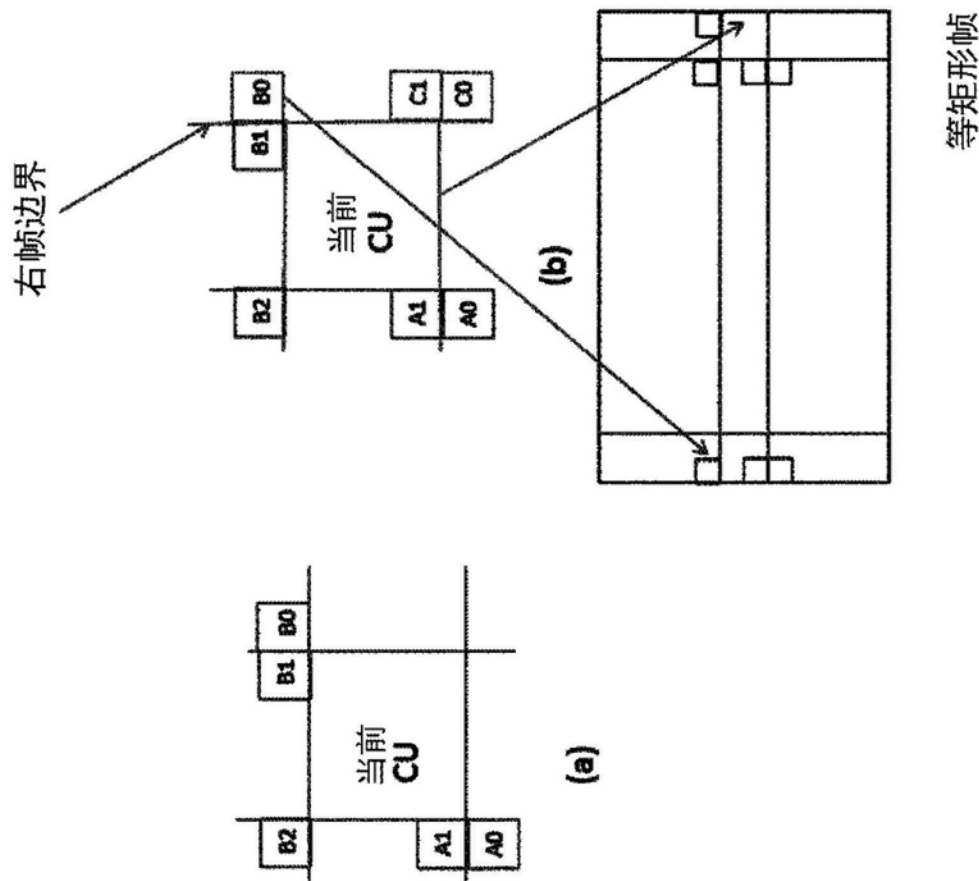


图24

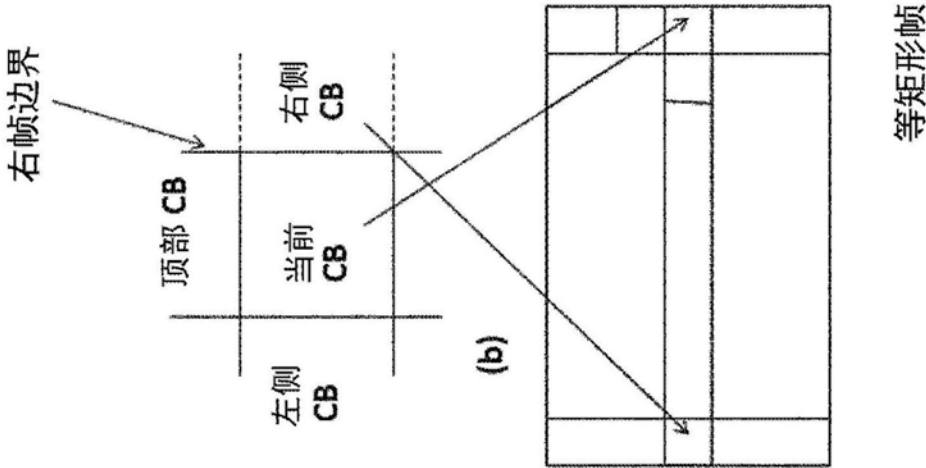


图 25

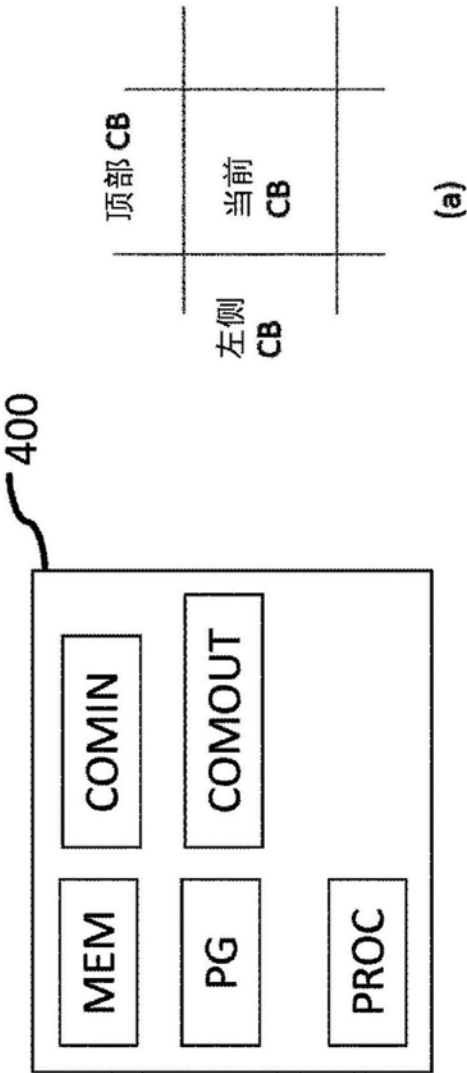


图 21

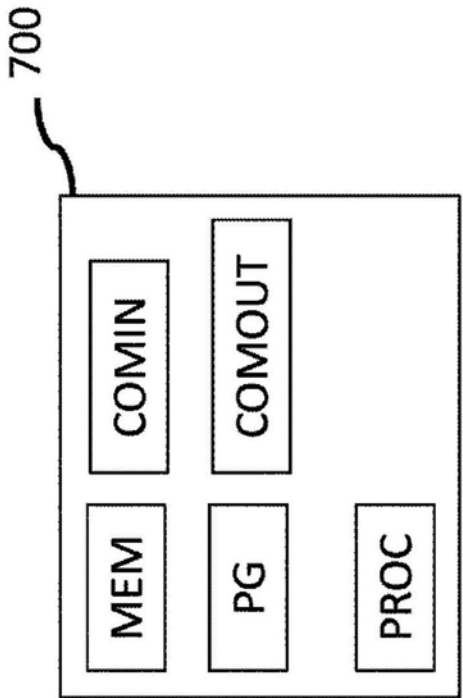


图 22

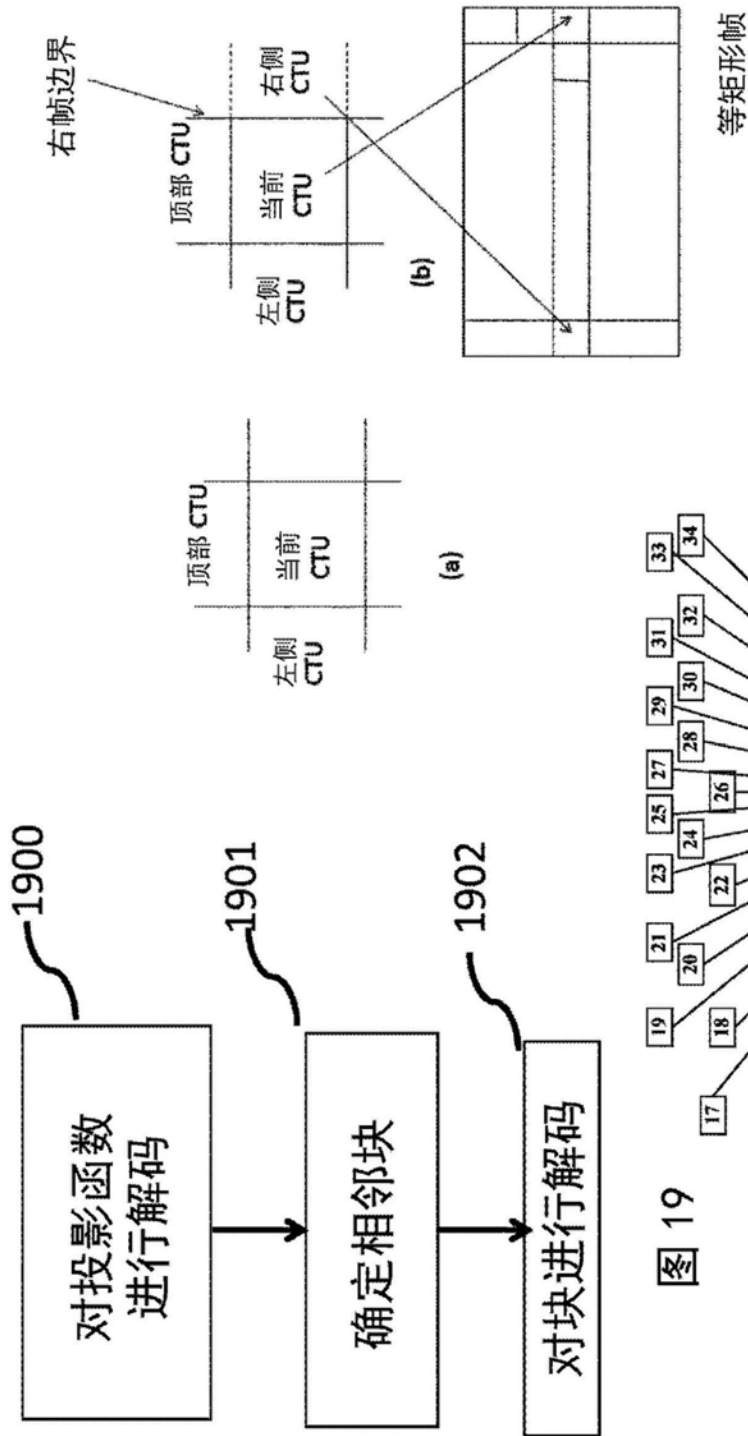


图 26

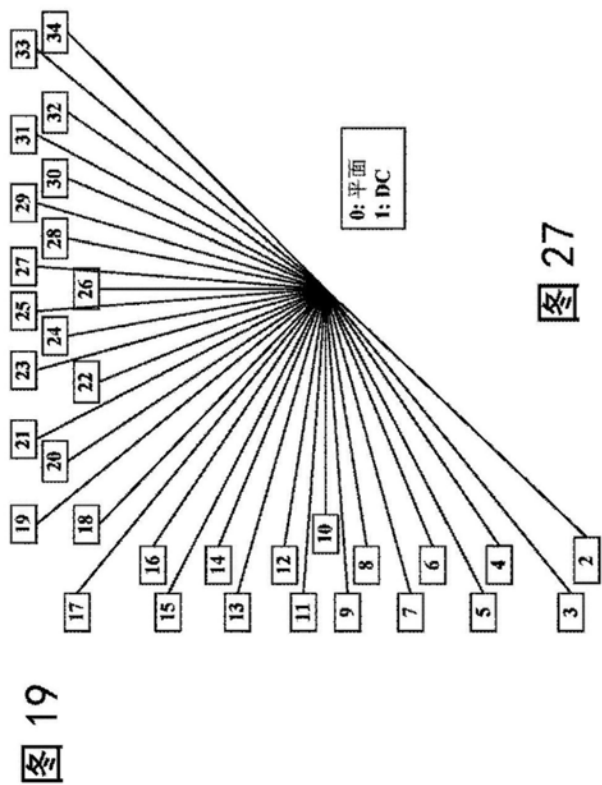


图 19