



(12) 发明专利申请

(10) 申请公布号 CN 116250238 A

(43) 申请公布日 2023. 06. 09

(21) 申请号 202180067571.5

(74) 专利代理机构 永新专利商标代理有限公司
72002

(22) 申请日 2021.09.27

专利代理师 李光颖

(30) 优先权数据

20199751.7 2020.10.02 EP

(51) Int.Cl.

H04N 19/167 (2006.01)

(85) PCT国际申请进入国家阶段日

2023.03.31

(86) PCT国际申请的申请数据

PCT/EP2021/076447 2021.09.27

(87) PCT国际申请的公布数据

W02022/069388 EN 2022.04.07

(71) 申请人 皇家飞利浦有限公司

地址 荷兰艾恩德霍芬

(72) 发明人 C·韦雷坎普

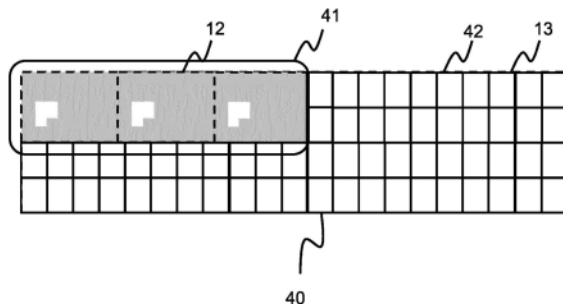
权利要求书3页 说明书10页 附图11页

(54) 发明名称

用于对场景的一幅或多幅视图进行编码和解码的方法和装置

(57) 摘要

提供了用于对包括场景的两幅或更多幅视图(10)的图像或视频数据进行编码和解码的方法。所述编码方法包括针对所述两幅或更多幅视图中的每幅视图获得(11)所述视图的相应块分割掩模(12)和所述视图的块图像数据(13)。所述编码方法还包括:生成(14)包含所述两幅或更多幅视图的所述两个或更多个块分割掩模和所述块图像数据的至少一个打包帧(40);并且将所述至少一个打包帧编码(15)成至少一个比特流(16)。每幅视图被划分成像素块(30),并且所述块分割掩模指示哪些像素块属于所述视图中的感兴趣区(31)。所述块图像数据包括属于所述感兴趣区的所述像素块。还提供了对应的编码器、解码器和比特流。



1. 一种对包括场景的两幅或更多幅视图(10)的图像或视频数据进行编码的方法,所述方法包括:

针对所述两幅或更多幅视图中的每幅视图获得(11)所述视图的相应块分割掩模(12)和所述视图的块图像数据(13);

生成(14)包含所述两幅或更多幅视图的所述两个或更多个块分割掩模和所述块图像数据的至少一个打包帧(40);并且

将所述至少一个打包帧编码(15)成至少一个比特流(16),

其中,每幅视图被划分成像素块(30),

其中,所述块分割掩模指示哪些像素块属于所述视图中的感兴趣区(31),

其中,所述感兴趣区仅包括所述视图的部分,

其中,所述块图像数据仅由属于所述感兴趣区的所述像素块组成。

2. 根据权利要求1所述的方法,其中,所述至少一个打包帧包括具有第一连续部分(41)和第二连续部分(42)的一个打包帧,

其中,所述第一连续部分包括所述两幅或更多幅视图的所述块分割掩模;并且

其中,所述第二连续部分包括所述两幅或更多幅视图的所述块图像数据。

3. 根据权利要求1或2所述的方法,其中,在视图内,该视图的所述块图像数据中的所述像素块都是相同大小的,

任选地,其中,在所有视图的所述块图像数据中,所述像素块都是相同大小的。

4. 根据前述权利要求中的任一项所述的方法,其中,相应不同视图的所述块图像数据以块交错布置的方式被打包在所述至少一个打包帧中,其中,第一视图的第一像素块之后紧跟着第二视图的第一像素块。

5. 根据权利要求1-3中的任一项所述的方法,其中,相应不同视图的所述块图像数据以行交错布置的方式被打包在所述至少一个打包帧中,其中,第一视图的第一行的所述像素块之后紧跟着第二视图的第一行的所述像素块。

6. 根据前述权利要求中的任一项所述的方法,其中,将所述至少一个打包帧编码成所述至少一个比特流包括使用视频压缩算法,任选地使用标准视频压缩算法,例如,H.264或HEVC。

7. 根据权利要求6所述的方法,包括:

选择(82)所述视频压缩算法的品质因数,使得至少所述块分割掩模能根据所述至少一个比特流来没有误差地重建;并且/或者

选择(80)在所述视频压缩算法中使用的量化水平的数量,使得至少所述块分割掩模能根据所述至少一个比特流来没有误差地重建。

8. 根据前述权利要求中的任一项所述的方法,包括:将所述块分割掩模(12)量化(81)为第一数量的量化水平,并且将所述块图像数据(13)量化(81)为第二数量的量化水平,其中,所述第一数量不同于所述第二数量。

9. 根据前述权利要求中的任一项所述的方法,其中,所述至少一个打包帧包括第三部分(50),其中,所述第三部分包括所述两幅或更多幅视图的深度数据(51);并且/或者

其中,所述至少一个打包帧包括第四部分(52),其中,所述第四部分包括所述两幅或更多幅视图的透明度数据(53)。

10. 一种对包括场景的两幅或更多幅视图的图像或视频数据进行解码的方法,所述方法包括:

接收(90)其中已经编码有至少一个打包帧的至少一个比特流,对于所述两幅或更多幅视图中的每幅视图,所述打包帧包含所述视图的相应块分割掩模和所述视图的块图像数据,

其中,每幅视图被划分成像素块,

其中,对于每幅视图,所述块图像数据仅由属于所述视图中的感兴趣区的所述像素块组成,并且所述块分割掩模指示所述视图中的属于所述感兴趣区的所述像素块的位置,

其中,所述感兴趣区仅包括所述视图的部分,

所述方法还包括:

对所述至少一个比特流进行解码(91)以获得所述至少一个打包帧;并且

通过根据由针对所述至少一幅视图的所述相应块分割掩模指示的所述位置布置所述至少一幅视图的所述块图像数据来重建(92)所述两幅或更多幅视图(93)中的至少一幅视图。

11. 根据前述权利要求中的任一项所述的方法,在直播多视图视频中进行使用。

12. 一种包括计算机程序代码的计算机程序,所述计算机程序代码被配置为当所述计算机程序在一个或多个物理计算设备上运行时使所述一个或多个物理计算设备执行根据前述权利要求中的任一项所述的方法的所有步骤。

13. 一种被配置为对包括场景的两幅或更多幅视图的图像或视频数据进行编码的编码器(100),所述编码器包括:

输入部(110),其被配置为针对所述两幅或更多幅视图中的每幅视图获得(11)所述视图的相应块分割掩模(12)和所述视图的块图像数据(13),

其中,每幅视图被划分成像素块,

其中,所述块分割掩模指示哪些像素块属于所述视图中的感兴趣区,

其中,所述感兴趣区仅包括所述视图的部分,

其中,所述块图像数据仅由属于所述感兴趣区的所述像素块组成;

打包单元(120),其被配置为生成(14)包含所述两幅或更多幅视图的所述两个或更多个块分割掩模和所述块图像数据的至少一个打包帧;以及

视频编码器(130),其被配置为将所述至少一个打包帧编码(15)成至少一个比特流。

14. 一种被配置为对包括场景的两幅或更多幅视图的图像或视频数据进行解码的解码器(200),所述解码器包括:

输入部(210),其被配置为接收(90)其中已经编码有至少一个打包帧的至少一个比特流,对于所述两幅或更多幅视图中的每幅视图,所述打包帧包含所述视图的相应块分割掩模和所述视图的块图像数据,

其中,每幅视图被划分成像素块,

其中,对于每幅视图,所述块图像数据仅由属于所述视图中的感兴趣区的所述像素块组成,并且所述块分割掩模指示所述视图中的属于所述感兴趣区的所述像素块的位置,

其中,所述感兴趣区仅包括所述视图的部分,

视频解码器(220),其被配置为对所述至少一个比特流进行解码(91)以获得所述至少

一个打包帧;以及

重建单元(230),其被配置为通过根据由针对所述至少一幅视图的相应块分割掩模指示的所述位置布置所述至少一幅视图的所述块图像数据来重建(92)所述两幅或更多幅视图(93)中的至少一幅视图。

15.至少一种比特流,其中已经编码有包括场景的两幅或更多幅视图的图像或视频数据,所述至少一个比特流包括:

至少一个打包帧,对于所述两幅或更多幅视图中的每幅视图,所述打包帧包含所述视图的相应块分割掩模和所述视图的块图像数据,

其中,每幅视图被划分成像素块,

其中,对于每幅视图,所述块图像数据仅由属于所述视图中的感兴趣区的所述像素块组成,并且所述块分割掩模指示所述视图中的属于所述感兴趣区的所述像素块的位置,

其中,所述感兴趣区仅包括所述视图的部分。

用于对场景的一幅或多幅视图进行编码和解码的方法和装置

技术领域

[0001] 本发明涉及对场景的一幅或多幅视图的图像或视频数据的译码。本发明特别涉及用于对从多个视点捕获的虚拟现实 (VR) 或沉浸式视频应用的视频序列进行编码和解码的方法和装置。

背景技术

[0002] 虚拟现实能够是观看场景的图像或视频的非常具有沉浸感的方式。当使用虚拟现实来观看所捕获的场景的图像或视频时,通常需要多个相机来从不同角度捕获场景的许多视图,以允许观看者在虚拟现实场景内四处移动。从不同角度捕获的视图越多,观看者能够在虚拟现实场景内移动的自由度就越大,并且所绘制的场景的视图就能够越准确。然而,增加捕获的视图的数量增加了必须处理和传输的数据量。对于有限的带宽,这可能降低观看者所体验的虚拟现实场景的图像或视频质量,因为必须更高度地压缩数据。

[0003] 场景的多幅视图常常与向解码器指示如何恢复原始视图的元数据一起编码。有效的编码常常需要计算上昂贵的确定步骤并且引起延迟,因为延迟了向观看者的数据传输。在效率(在针对给定带宽的比特率或像素速率方面)与延迟之间可以存在权衡。对于直播视频,延迟是一个特别关注的问题,因为观看者想要在没有延迟的情况下体验虚拟现实场景,特别是在诸如视频会议之类的双向流场景中。

发明内容

[0004] 就计算工作量和数据速率(带宽)而言,将期望有效地对场景的一幅或多幅视图进行编码和解码。

[0005] 本发明由权利要求来限定。根据本发明的一个方面,提供了一种根据权利要求1所述的对图像或视频数据进行编码的方法。

[0006] 对于每幅视图,块分割掩模指示属于感兴趣区的像素块的位置。在任何给定视图中可能存在一个以上的感兴趣区。

[0007] 该方法的实施例能够促进对多视图视频的简单且低延迟的编码。实际上,块分割掩模能够提供允许解码器根据至少一个打包帧来快速容易地重建视图中的一幅或多幅视图的隐式元数据。同时,能够降低像素速率,因为仅对每幅视图的部分(即,感兴趣区)进行编码/传输。

[0008] 在一些实施例中,至少一个打包帧可以是单个打包帧。至少一个比特流可以是单个比特流。

[0009] 对于视图中的每个像素块,在指示像素块是否属于感兴趣区的块分割掩模中可以存在对应的像素。因此,块分割掩模中的像素可以与相应视图中的像素块一样多。在一些实施例中,在与视图中的像素块相对应的块分割掩模中可以存在一个以上的像素。例如,视图中的像素块可以在指示视图中的像素块是否属于感兴趣区的块分割掩模中具有对应的像素块。块分割掩模中的像素块可以小于视图中的像素块。

[0010] 在每个块分割掩模中,每个像素可以包括指示对应的像素块是否是感兴趣区的部分的像素值。像素值可以是亮度值或另一像素值,例如,色度、深度或透明度值。

[0011] 用于指示属于感兴趣区的块的像素值可以以未使用的水平与用于指示不在感兴趣区中的块的像素值分开。所述的未使用的水平能够产生对像素值中的小偏差的鲁棒性,所述的像素值中的小偏差可能是通过将传统的有损视频压缩技术应用于打包帧而引入的。假设所得到的像素值范围保持不同且可分离,虽然存在这样的偏差,但是可以在解码器处没有误差地重建块分割映射图。

[0012] 在一些实施例中,可以存在一个以上的感兴趣区。块分割掩模的像素值可以充当针对感兴趣区的索引。例如,可以在块分割掩模中用第一非零像素值标记第一感兴趣区;并且可以在块分割掩模中用第二非零像素值标记第二感兴趣区。不属于任何感兴趣区的块可以用像素值零来标记。

[0013] (一个或多个)感兴趣区可以包括(一个或多个)前景对象或者由(一个或多个)前景对象组成。视图的其余部分可以包括背景或者由背景组成。

[0014] 块图像数据可以仅由针对属于感兴趣区的像素块的图像数据组成。

[0015] 图像或视频数据包括多幅视图。

[0016] 对于每幅视图,可以以基于相应块分割掩模的扫描顺序的序列将属于感兴趣区的像素块打包在打包帧中。

[0017] 因此,打包帧可以包括与第一视图相关联的第一块分割掩模,第一块分割掩模后面紧跟着与第二视图相关联的第二块分割掩模。块图像数据可以包括第一视图的一个或多个像素块,第一视图的一个或多个像素块后面紧跟着第二视图的一个或多个像素块。可以以相同的方式添加第三视图、第四视图和另外的视图。

[0018] 块分割掩模的扫描顺序可以是光栅扫描顺序(即,在沿着列扫描之前跨行扫描)。

[0019] 获得块分割掩模可以包括分割每幅视图以生成块分割掩模。例如,如果针对色度键背景(例如,绿色屏幕)捕获视图,那么分割可以包括通过颜色分离(颜色键控)将像素分类为前景或背景。可以在块分割映射图中将包括预定最小数量的前景像素的任何像素块标记为前景。

[0020] 所述至少一个打包帧可以包括这样一个打包帧,所述一个打包帧具有第一连续部分和第二连续部分,所述第一连续部分包括所述两幅或更多幅视图的所述块分割掩模,所述第二连续部分包括所述两幅或更多幅视图的所述块图像数据。

[0021] 能够将两个或更多个块分割掩模连续地打包在打包帧中,由此每个块分割掩模与至少一个其他块分割掩模相邻。第一部分可以是打包帧的最左上部分。

[0022] 至少一个比特流可以由一个比特流组成。第一部分可以被编码为使得它出现在紧接在第二部分之前的一个比特流中。这能够促进快速高效的解码,其中,解码器能够在接收并任选地解码块图像数据之前接收并任选地解码块分割掩模。因此,当接收/解码块图像数据时,解码器将已经具有在重建视图中定位像素块的必要信息。

[0023] 在其他实施例中,至少一个打包帧可以包括两个或更多个打包帧。第一打包帧可以包括两个或更多个块分割掩模。第二打包帧可以包括块图像数据。这能够允许块分割掩模与块图像数据分开编码——例如使用不同的视频压缩参数来实现分开编码。可以对包括第一打包帧的第一比特流进行编码,并且可以对包括第二打包帧的第二比特流进行编码。

[0024] 在视图内,该视图的块图像数据中的像素块可以都是相同大小的,或者任选地,在所有视图的块图像数据中,块可以都是相同大小的。在前一种情况下,块大小在视图内是恒定的(或者在每幅视图内是独立恒定的)。这能够促进高效打包以及简单的编码和解码。在后一种情况下,块大小在所有视图上都是恒定的。这还能够促进高效打包以及简单的编码和解码。

[0025] 可以以块交错布置的方式将相应不同视图的块图像数据打包在至少一个打包帧中,其中,在第一视图的第一像素块后面紧跟着第二视图的第一像素块。在第二视图的第一像素块后面可以紧跟着第三视图的第一像素块,以此类推,直到来自每幅视图的第一像素块都已经被打包到打包帧中为止。只要在每幅视图中存在足够数量的块以维持这种模式,就可以针对每幅视图的后续块重复从每幅视图一次获取单个像素块并在打包帧中交错彼此相邻的块的这种模式。

[0026] 该方法可以得到良好的视频压缩效率,特别是当感兴趣区的内容和位置在不同视图上相似时。在这种情况下,在打包帧中交错的相邻像素块的内容可能是相似的。对于给定的比特率,标准视频压缩算法可以利用这种空间一致性来降低编码块所需的比特率和/或提高压缩质量。

[0027] 可以以行交错布置的方式将相应不同视图的块图像数据打包在至少一个打包帧中,其中,在第一视图的第一行的像素块后面紧跟着第二视图的第一行的像素块。在第二视图的第一行的像素块后面可以紧跟着第三视图的第一行的像素块,以此类推,直到来自每幅视图的第一行的块都已经被打包为止。只要在每幅视图中存在足够数量的行以维持这种模式,就可以继续一次从每幅视图获取整行块并在打包帧中交错彼此相邻的行的这种模式。再次地,对于包含典型视觉内容的场景,这种方法可以得到良好的视频压缩效率。

[0028] 将至少一个打包帧编码成至少一个比特流任选地包括使用视频压缩算法,任选地包括使用标准视频压缩算法,例如,H.264或HEVC。这能够促进与现有的编码和解码硬件和软件的至少部分向后兼容。它还能够避免对专门支持多视图视频的不同视频压缩标准的扩散的需求。

[0029] 该方法可以包括:选择所述视频压缩算法的品质因数,使得至少所述块分割掩模能根据所述至少一个比特流来没有误差地重建;并且/或者选择在所述视频压缩算法中使用的量化水平的数量,使得至少所述块分割掩模能根据所述至少一个比特流来没有误差地重建。任选地,该方法可以包括联合地选择在所述视频压缩算法中使用的量化水平的数量和品质因数,以确保所述块分割掩模能根据所述至少一个比特流来没有误差地重建。

[0030] 至少一个打包帧可以包括两个打包帧。第一打包帧可以包括两个或更多个块分割掩模。第二打包帧可以包括块图像数据。可以使用无损视频压缩对第一打包帧进行编码。可以使用有损视频压缩对第二打包帧进行编码。

[0031] 该方法可以包括:将所述块分割掩模量化为第一数量的量化水平,并且将所述块图像数据量化为第二数量的量化水平,其中,所述第一数量不同于所述第二数量。通过独立地设置量化参数,该方法能够确保块分割掩模在视频压缩后能没有误差地重建,同时避免在块图像数据上浪费不必要的带宽。例如,可以以在解码器处在所重建的块图像数据中引起一些误差的方式量化块图像数据。以这种方式,能够以无损方式有效地压缩块分割掩模,同时以有损方式压缩块图像数据。

[0032] 所述至少一个打包帧可以包括第三部分,其中,所述第三部分包括两幅或更多幅视图的深度数据;并且/或者所述至少一个打包帧可以包括第四部分,其中,所述第四部分包括所述两幅或更多幅视图的透明度数据。可以将深度数据以比块图像数据更低的空间分辨率包括在至少一个打包帧中。例如,可以以更小的块和/或减少数量的块提供深度数据。

[0033] 还提供了根据权利要求10所述的解码方法。

[0034] 块图像数据仅由属于感兴趣区的像素块组成。可以重建所有两幅或更多幅视图。该方法还可以包括将感兴趣区合成到背景场景上以合成完整视图。

[0035] 可以在直播多视图视频中使用如上概述或如权利要求保护的任何方法。译码方案可以非常适合于实况视频流的需求,因为其简单性存进了编码器端和解码器端的低延迟。

[0036] 还提供了根据权利要求12所述的计算机程序。计算机程序可以被存储在计算机可读介质上,任选地被存储在非瞬态计算机可读介质上。

[0037] 还提供了根据权利要求13所述的编码器和根据权利要求14所述的解码器。

[0038] 还提供了根据权利要求15所述的比特流。块分割掩模能够紧接在块图像数据之前被打包到打包帧中。块分割掩模能够在块图像数据之前由解码器解码。

附图说明

[0039] 为了更好地理解本发明并且为了更清楚地示出可以如何将本发明付诸实践,现在将会仅以示例的方式参考附图,在附图中:

[0040] 图1是根据本发明的实施例的编码器的框图;

[0041] 图2是图示由图1的编码器执行的编码方法的流程图;

[0042] 图3A图示了根据实施例的被划分成像素块的场景的视图;

[0043] 图3B图示了针对图3A的视图(放大)的块分割掩模;

[0044] 图3C图示了针对图3A的视图的块图像数据;

[0045] 图4图示了根据实施例的打包帧;

[0046] 图5A图示了根据实施例的打包帧;

[0047] 图5B图示了根据实施例的打包帧;

[0048] 图6图示了根据实施例的打包方法;

[0049] 图7图示了根据实施例的打包方法;

[0050] 图8图示了根据实施例的编码方法;

[0051] 图9是根据实施例的解码器的框图;

[0052] 图10图示了由图9的解码器执行的解码方法;并且

[0053] 图11图示了根据实施例的重建方法。

具体实施方式

[0054] 将参考附图来描述本发明。

[0055] 应当理解,详细描述和具体示例在指示装置、系统和方法的示例性实施例的同时,仅旨在用于图示的目的,而并不旨在限制本发明的范围。根据以下描述、权利要求和附图将更好地理解本发明的装置、系统和方法的这些和其他特征、方面和优点。应当理解,附图仅仅是示意性的且并未按比例绘制。还应当理解,在整个附图中使用相同的附图标记来指示

相同或相似的部件。

[0056] 图1是根据实施例的编码器100的框图。编码器100包括输入部110,输入部110被配置为获得两个或更多个块分割掩模12和块图像数据13。输入部100被连接到打包单元120,打包单元120被配置为生成打包帧40。打包单元120被连接到视频编码器130,视频编码器130被配置为将打包帧40编码成至少一个比特流16。

[0057] 现在将参考图2、图3、图4和图6来描述由编码器100执行的方法。图2是图示该方法的流程图。在下面将详细描述的实施例中,已经捕获了场景的三幅视图10。图3A图示了场景的视图10中的一幅视图。视图10被均匀地划分成像素块30。其他视图(未图示)中的每幅视图也被均匀地划分成像素块。在该示例中,视图10中的像素块30都是相同的大小的。每幅视图10包含前景区域和背景区域。如本领域中常见的,绿色屏幕能够用作背景。其示例是包含被放置在绿色屏幕前面的对象的视图10。使用绿色屏幕作为背景允许通过色度键控来分割每幅视图10。与背景屏幕的颜色(在这种情况下为绿色)匹配的像素被识别为背景像素。

[0058] 图3B图示了针对图3A的视图10的块分割掩模12的放大视图。色度键控已经用于识别前景区域(在本文中被称为感兴趣区31)和背景区域。块分割掩模12是视图10的映射图,其指示视图10中的哪些像素块30属于感兴趣区31。视图10中的每个像素块30由块分割掩模12中的像素表示,使得块分割掩模12成为视图10的低分辨率分割掩模。

[0059] 块分割掩模12通过将块分割掩模12中的每个像素的像素值设置为第一值或第二值来指示视图10中的像素块30是否属于感兴趣区31。在通过色度键控的分割的情况下,如果块包含至少一个前景像素(即,至少一个除了绿色之外的颜色的像素),那么可以确定该块属于感兴趣区。

[0060] 在图3B中,块分割掩模中的像素值是亮度值。块分割掩模12的亮区域32指示视图10中属于感兴趣区31的像素块30。块分割掩模12的暗区域33指示视图10中不属于感兴趣区31的像素块30。图3C图示了图3A的视图10的块图像数据13,如通过图3B的块分割掩模12所指示的那样。块图像数据13仅包括属于感兴趣区31的像素块30。

[0061] 现在参考图2的流程图。在步骤11中,输入部110针对视图10中的每幅视图获得相应的块分割掩模12和块图像数据13。在该示例中,使用色度键控来创建块分割掩模12和块图像数据13。将块分割掩模12和块图像数据13传送到打包单元120。

[0062] 在步骤14中,打包单元120生成包含块分割掩模12和块图像数据13的打包帧40。图4图示了打包帧40。打包单元120将块分割掩模12连续地打包在打包帧40的第一部分41中。打包帧40的第一部分41是将被编码并且随后被解码的第一部分。

[0063] 打包单元120将块图像数据13打包到打包帧40的第二部分42中。也连续地对块图像数据13进行打包,不在像素块30之间或像素块30与块分割掩模12之间留下空间。

[0064] 在图5中示出了打包帧的其他可能配置,并且将稍后在下文中解释打包帧的其他可能配置。

[0065] 现在将参考图6解释用于生成图4所示的打包帧40的打包方法。块分割掩模60、61和62指示第一视图600、第二视图610和第三视图620中的相应感兴趣区。

[0066] 首先,将块分割掩模60、61、62成行地打包在打包帧40的第一部分中。将第一块分割掩模60打包在行中的最左侧位置中。将第二块分割掩模61紧挨着第一块分割掩模60进行打包,并且将第三块分割掩模62紧挨着第二块分割掩模61进行打包。块分割掩模60、61、62

在行中从左到右的顺序定义了序列。

[0067] 接下来,通过块交错将来自三幅视图600、610、620中的每幅视图的块图像数据打包到打包帧40的第二部分中。用像素块按光栅扫描顺序对打包帧40进行打包。首先,打包单元120将来自第一视图600的第一像素块打包到打包帧40中。然后,打包单元120将来自第二视图610的第一像素块打包到打包帧40中。然后,打包单元120将来自第三视图620的第一像素块打包到打包帧40中。该循环然后重复,对来自第一视图600的第二像素块进行打包,以此类推,重复直到从每幅视图没有足够的像素来继续该序列为止。以这种方式,打包帧40中的像素块发生交错,使得没有像素块紧接在来自同一视图的另一像素块之前或之后,除非不再有足够的像素块来维持交错。打包帧40中的像素块的序列与打包帧中的块分割掩模60、61、62的顺序所定义的序列相同。

[0068] 以与按光栅扫描顺序将块图像数据打包到打包帧40中的相同方式,选择来自视图600、610、620的像素块以用于按光栅扫描顺序进行打包,如图6所示。从每幅视图600、610、620中选择像素块以用于从视图600、610、620的顶部到底部成行地打包。图6示出了来自视图600的最左上像素块是从视图中首先打包的,因为它是根据光栅扫描顺序的感兴趣区中的第一个块。从视图600中打包的下一个块是根据光栅扫描顺序紧接在第一个块右侧的块——第二个块。

[0069] 图7图示了打包像素块的替代方法,被称为行交错方法,这将在稍后在下文中进行解释。

[0070] 在打包之后,在步骤14中,将打包帧40传送到视频编码器130。视频编码器130将打包帧40编码成比特流16。对打包帧进行编码包括压缩15打包帧。压缩15打包帧40减少了必须被编码到比特流16中并被传输的数据量。在该示例中,压缩15以有损方式完成。可以使用标准视频压缩编解码器。在本示例中,使用高效视频译码(HEVC)压缩标准。

[0071] 压缩15打包帧40能够减少要编码和传输的数据量,具体是由于打包帧40已经被打包的方式。在相应视图10中的感兴趣区31包含类似的视觉内容(很可能场景的多视图图像或视频也包含类似的视觉内容)的情况下,来自每幅视图10的块图像数据13在视觉内容方面可能是类似的。更具体地,由于块交错打包方法,打包帧中的相邻像素块可能是与从不同视图看到的感兴趣区的类似部分相对应的那些像素块。这些像素块在视觉内容方面可能具有最大的相似性。现有视频压缩算法(例如,HEVC)能够利用这种相似性(冗余)。以这种方式,通过使用块交错或行交错,并且使用视频压缩算法来压缩打包帧,本发明的实施例能够减少需要传输的数据量,同时利用现有的视频压缩软件和/或硬件。

[0072] 在图8中示出了在一些实施例中能够由编码器执行的附加步骤,并且将稍后在下文中解释这些附加步骤。

[0073] 图9是根据实施例的解码器200的框图。解码器200包括被配置为接收比特流16的输入部210。输入部210被连接到视频解码器220,视频解码器220被配置为对比特流16进行解码。视频解码器220被连接到重建单元230,重建单元230被配置为重建场景的一幅或多幅视图93。假定由解码器200接收的比特流16是由上述实施例中的编码器100产生的。

[0074] 现在将参考图10来描述由解码器200执行的方法。在步骤90中,输入部210接收比特流16。比特流16包括打包帧40。将比特流16传送到视频解码器220,在步骤91中,在视频解码器220中对比特流16进行解码。解码包括(使用HEVC)解压缩和检索打包帧40。在解码之

后,将打包帧传送到重建单元230。

[0075] 在步骤92中,重建单元230重建场景的三幅视图93。为了简单起见,首先将参考仅重建第一视图690来解释重建。图11所示的重建方法包括创建空白的第一重建视图690。重建单元230在由第一视图的块分割掩模60指示的前景位置处将第一视图的像素块插入到第一重建视图690中。

[0076] 块分割掩模60充当映射图,其指示重建视图690中的前景块的位置(由直箭头图示)。编码器和解码器都使用相同的预定义扫描顺序来扫描块分割掩模和块图像数据。当解码器找到块分割掩模的第一前景像素时,它在重建视图690中由块分割掩模的像素指示的位置(由弯曲箭头图示)处插入第一解码像素块。当它找到块分割掩模60的下一个前景像素时,它插入针对该视图的下一个解码像素块,依此类推,直到针对该视图的所有前景像素块都已经在其正确位置中被布置在重建视图690中为止。

[0077] 该方法同样适用于重建所有视图93,与在编码器处执行的打包过程是反向的。

[0078] 解码器200不需要任何附加元数据来知道在何处找到相关的像素块以及在何处定位相关的像素块。取而代之的是,打包帧中的块分割掩模12和像素块30的序列用作隐式元数据,其向解码器200指示哪个块分割掩模12与哪些像素块相关。避免与打包帧40一起包括附加元数据的需求能够有助于减少必须编码的数据量。并且使用块分割映射图的重建实施起来相对简单——它不涉及可能引起显著的附加延迟的复杂或耗时的操作。

[0079] 前景块(即,感兴趣区的块图像数据)可以由解码器叠加在任何期望的背景(例如,计算机图形背景)或来自不同视频流的图像或视频上。在一些实施例中,可以使用多幅重建视图93来创建具有与由相机最初捕获的任何视图不同的视点的新的虚拟视图。

[0080] 现在将讨论如图5A和图5B所示的打包帧40的可能替代配置。图5A图示了除了第一部分和第二部分(未图示)之外还具有第三部分50的打包帧40。第三部分在打包帧的底部处,并且是要编码和解码的帧的最后部分。第三部分50包括针对两幅或更多幅视图10的块深度数据51。将块深度数据51连续地打包在打包帧40中。可以以比块图像数据13更低的空间分辨率将块深度数据51包括在至少一个打包帧40中。例如,可以以更小的块和/或减少数量的块来提供块深度数据51,这是因为人类视觉系统可能对深度误差比对图像数据的误差更不敏感。

[0081] 图5B图示了除了第一部分、第二部分和第三部分(未图示)之外还具有第四部分52的另一打包帧40。第四部分52包括两幅或更多幅视图10的块透明度数据53。将块透明度数据53连续地打包在第四部分52内。能够在对两幅或更多幅视图的分割期间确定透明度数据,并且将透明度数据存储存储在 α 通道10中。在感兴趣区31与背景之间的边界处的像素块将可能既包含一些(绿色)背景像素又包含一些前景像素(非绿色)。针对像素块的块透明度数据为块中的每个像素标识像素的透明程度。在色度键控的情况下,这能够通过其绿色程度来确定,或者通过任何其他合适的像素水平分割来确定。在重建期间,透明像素能够被视为背景像素。这能够有助于防止并入感兴趣区31的块图像数据中的绿色(背景)像素在所重建的图像93中表现为对象周围的绿色光晕。部分透明像素(包含一些绿色但不完全是绿色的像素)能够通过 α 混合对其颜色值进行平均化。如果透明度数据53不被包括在打包帧中,那么在一些实施例中,解码器200本身能够对像素块执行色度键控以确定透明度数据。也就是说,在视图93的重建期间,解码器200能够检查每个像素块31(特别是外围块)的绿色像素,

以便确定所重建的视图93的块透明度数据。

[0082] 在打包帧包括块深度数据51或块透明度数据53的情况下,能够以与将块图像数据打包到打包帧的第二部分中的相同模式将块深度和块透明度数据打包到打包帧的第三部分和第四部分中。这意味着当解码器正在扫描打包帧40并重建视图93时,它能够利用与用于生成所重建的视图93的相同的反向块交错方法将块深度/透明度数据51、53应用于所重建的视图。

[0083] 在一些实施例中,能够修改参考图6描述的块交错方法,以交错来自每幅视图的多个块——例如,一次取两个、三个或更多个块作为交错单元。

[0084] 图7示出了基于行交错的打包方法的示例。行交错操作类似于块交错,不同之处在于不是一次打包来自每幅视图的像素块,而是打包单元120打包来自每幅视图700、710、720的块的行。在一些实施例中,通过以下操作来进一步扩展这个原理:连续地打包来自第一视图10的所有像素块30,随后是来自第二视图10的所有块30,对所有视图10以此类推。解码器200将总是使用用于对打包帧40进行打包的过程的反向过程来重建视图93。

[0085] 在期望仅对视图10的子集进行解码和重建的情况下,行交错可以比块交错更有利。利用行交错,更可能的是(在解码器处)对像素块30的预测将基于来自同一视图10的另一像素块30,因为压缩算法常常基于相邻块进行预测,并且整行块30被打包为一个连续组。当希望对给定视图10进行解码时,这能够有助于避免对来自附加视图10的像素块30进行解码的需求。以相同的方式,可能有利的是,将打包帧40打包为具有来自第一视图10的连续放置的所有像素块30,随后是来自第二视图10的所有块30,对于所有视图10以此类推。

[0086] 现在将解释图8所示的编码方法的实施例。图8是示出包括量化和压缩的编码方法的流程图。步骤11、14和15与上面描述的图2的实施例中的对应步骤基本上相同。

[0087] 在步骤11中,编码器100从场景的两幅或更多幅视图10获得块分割掩模12和块图像数据13。

[0088] 在步骤80中,编码器100为块分割掩模12选择多个量化水平。量化块分割掩模12以便减少需要传输的数据量可以是有益的。通过示例的方式,应当考虑使用亮度值作为像素值的块分割掩模12。能够使用第一值和第二值来区分前景与背景——例如通过将所有前景像素设置为255并且将所有背景像素设置为0来实现这一点。对值0和255进行编码将需要8比特。为了减少数据量,能够将像素值量化81为更少数量的水平。代替使用0和255作为第一值和第二值,可以使用0和127,这例如仅需要7比特来表示。在最极端的减少中,每个像素能够具有仅由单个比特表示的0或1的亮度值。

[0089] 在步骤81中,将所选择的量化参数应用于块分割掩模12。在步骤14中,打包单元120生成打包帧40。在步骤82中,编码器选择视频压缩的品质因数。品质因数越高,由于有损压缩而丢失的数据就越少。

[0090] 然后将打包帧编码成比特流16,所述编码包括使用HEVC编解码器(具有所选择的品质因数)来压缩15打包帧。优选的是,解码器200能够根据所压缩的打包帧40来准确地重建块分割掩模12,否则解码器200无法准确地重建视图93。当为块分割掩模12选择量化水平时,在第一像素值与第二像素值之间维持一定的未使用水平会是有用的。这能够有助于使该方法对可能由有损视频压缩引入的像素值中的小偏差具有鲁棒性。

[0091] 为了确保解码器200能够准确地重建块分割掩模12,必须通过无损压缩或通过具

有最小品质因数的有损压缩来压缩块分割掩模12。例如,可以通过无损压缩或利用某个最小质量的有损压缩来压缩使用值为0和1的二元分割掩模,使得没有像素改变值。替代地,可以通过有损压缩来量化和压缩使用值0和255的分割掩模。有损压缩能够允许值偏离其初始水平,只要量化对这些偏离是鲁棒的,使得决不存在重建误差。存在各种方式来确保这一点。在一个示例中,可以选择量化和品质因数参数,使得在视图的大训练数据集上,误差的可能性很低或为零。替代地,对于正被编码的视图的给定集合,编码器能够在线优化参数。为此目的,编码器100能够包括本地解码器200,本地解码器200在传输比特流16之前对打包帧40进行解码并重建块分割掩模12。编码器100能够通过将来自本地解码器的经解码的块分割掩模与原始块分割掩模进行比较来检查是否已经发生了准确的重建。编码器100能够迭代地量化块分割掩模12;生成、压缩和编码打包帧40,解码打包帧40,并且将所重建的块分割掩模与原始块分割掩模进行比较,每次改变量化和/或压缩条件,以便以最小的数据量实现准确重建的块分割掩模。然后能够将最优解应用于打包帧40并传输所得到的比特流16。

[0092] 图2、图8和图10的编码方法和解码方法以及图1和图9的编码器100和解码器200可以以硬件或软件或这两者的混合(例如作为在硬件设备上运行的固件)方式来实现。在实施例部分或全部地以软件方式实施的程度上,过程流程图中所示的功能步骤可以由经适当编程的物理计算设备(例如,一个或多个中央处理单元(CPU)或图形处理单元(GPU))来执行。每个过程(以及如流程图中所示的其个体组成步骤)可以由相同或不同的计算设备来执行。根据实施例,一种计算机可读存储介质存储包括计算机程序代码的计算机程序,该计算机程序代码被配置为当程序在一个或多个物理计算设备上运行时使一个或多个物理计算设备执行如上所述的编码或解码方法。

[0093] 对上述示例的各种修改是可能的。例如,在上述示例中,将块分割映射图提供给编码器。在一些实施例中,编码器可以包括分割单元,该分割单元被配置为接收视图10并且被配置为使用色度键控(或另一分割算法)来产生块分割掩模12。

[0094] 在一些实施例中,块分割掩模12中使用的第一像素值和第二像素值可以是色度值、深度值或透明度值。在块分割掩模12包含两个感兴趣区的情况下,第一像素值和第二像素值能够用于分别指示第一感兴趣区和第二感兴趣区,并且第三像素值能够用于指示背景区。例如,值0可以表示背景,并且值128和255可以表示第一感兴趣区和第二感兴趣区(例如,前景对象)。像这些值的宽间隔值可以适合用于有损压缩。替代地,值1和2可以用于感兴趣区,而值0用于背景。像这些值的相邻值可以适合用于无损压缩,其中,不存在在值中引入偏差的风险。

[0095] 在一些实施例中,打包单元120可以生成两个或更多个打包帧。第一打包帧可以包括两个或更多个块分割掩模12。第二打包帧可以包括块图像数据13。将两个或更多个打包帧传送到视频编码器130。视频编码器130可以对包括第一打包帧的第一比特流和包括第二打包帧的第二比特流进行编码。第一比特流的压缩可以是无损的,第二比特流的压缩可以是有损的。

[0096] 并不一定需要在所有视图中像素块都是相同大小的。在一些实施例中,一些或所有视图的像素块可以是不同大小的。能够利用本领域已知的任何合适算法(例如,H.264或一定范围的MPEG编解码器)来完成对打包帧的压缩和解压缩。

[0097] 在一些实施例中,不需要连续对块分割掩模12和块图像数据13进行打包。例如,在第一视图的块分割掩模后面可以紧跟着第一视图的块图像数据,在第一视图的块图像数据后面可以紧跟着第二视图的块分割掩模和第二视图的块图像数据,对于所有视图以此类推。

[0098] 存储介质可以包括易失性和非易失性计算机存储器,例如,RAM、PROM、EPROM和EEPROM。各种存储介质可以被固定在计算设备内,也可以是可转移的,使得存储在其上的一个或多个程序能够被加载到处理器中。

[0099] 本领域技术人员通过研究附图、公开内容以及权利要求,在实践请求保护的发明时能够理解并实现对所公开的实施例的变型。在权利要求中,“包括”一词并不排除其他元件或步骤,并且词语“一”或“一个”并不排除多个。单个处理器或其他单元可以实现在权利要求中记载的若干项的功能。虽然某些措施被记载在互不相同的从属权利要求中,但是这并不指示不能有利地使用这些措施的组合。如果上面讨论了计算机程序,那么它可以被存储/分布在合适的介质上,例如与其他硬件一起或作为其他硬件的部分而供应的光学存储介质或固态介质,但是也可以以其他形式分布,例如经由互联网或其他有线或无线的电信系统进行分布。如果在权利要求书或说明书中使用了术语“适于”,那么应当注意,术语“适于”旨在等同于术语“被配置为”。权利要求中的任何附图标记都不应被解释为对范围的限制。

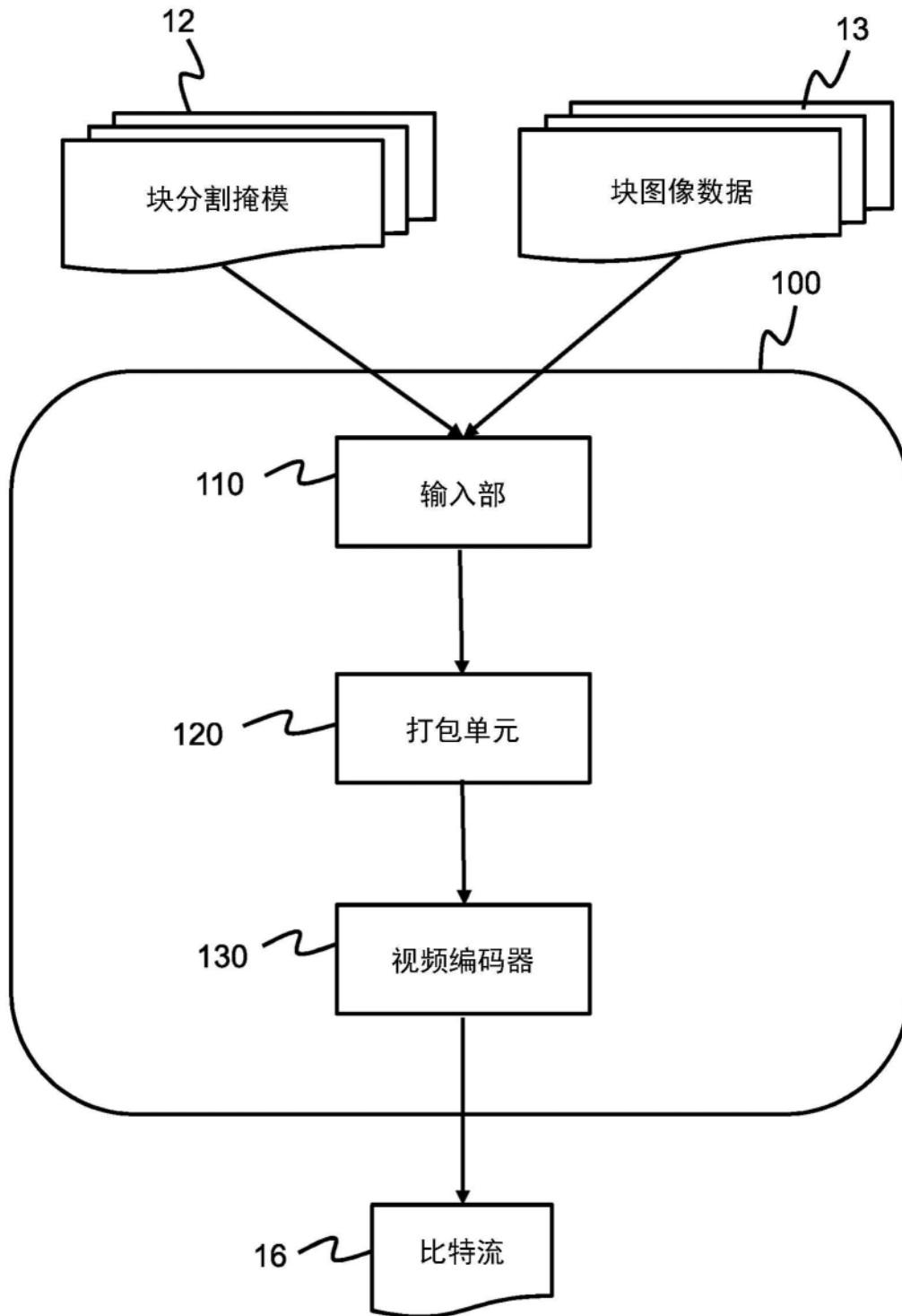


图1

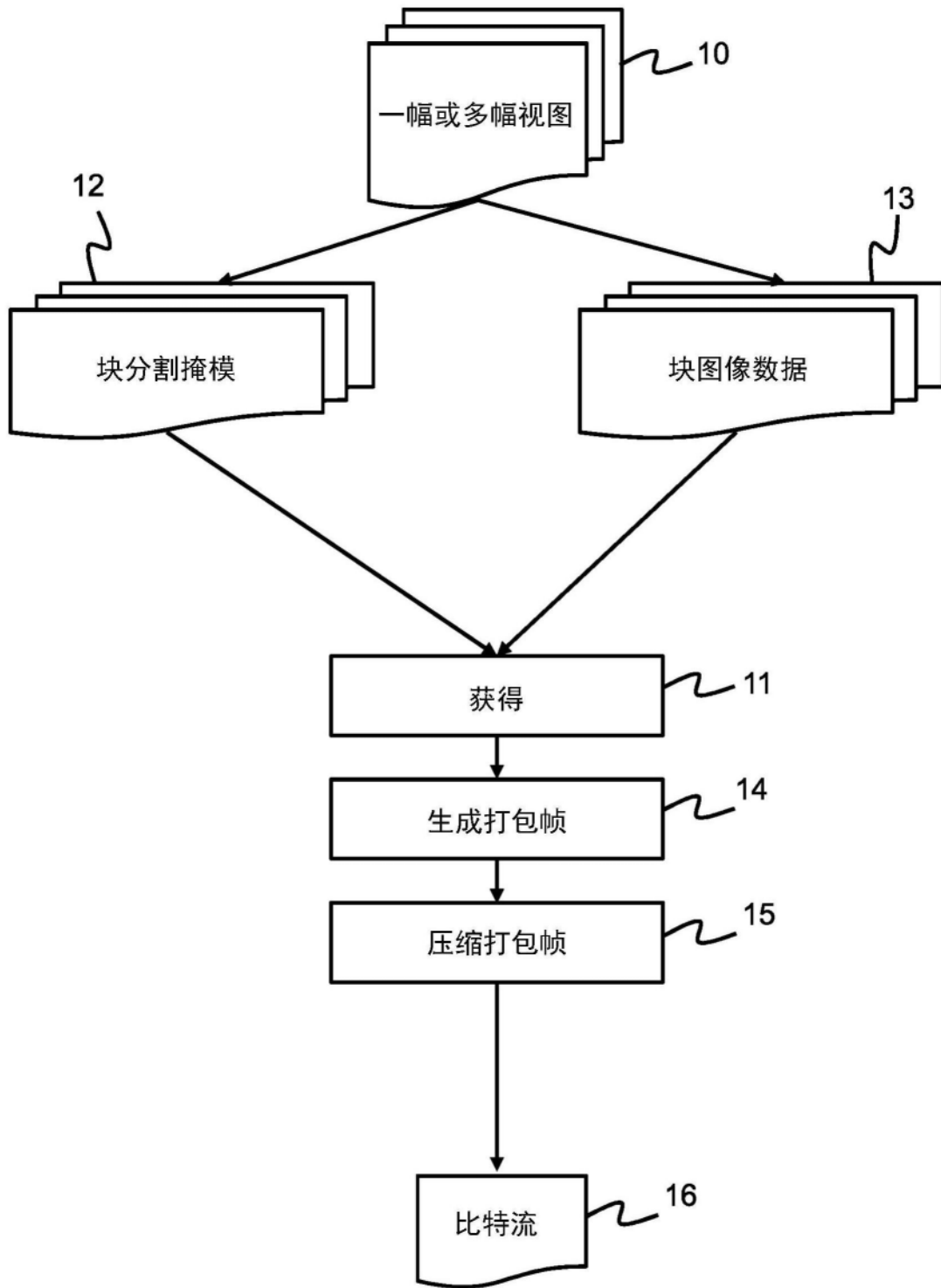


图2

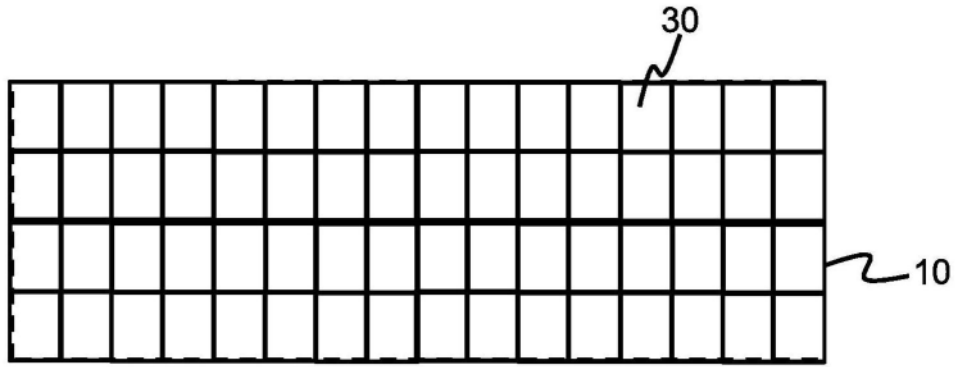


图3A

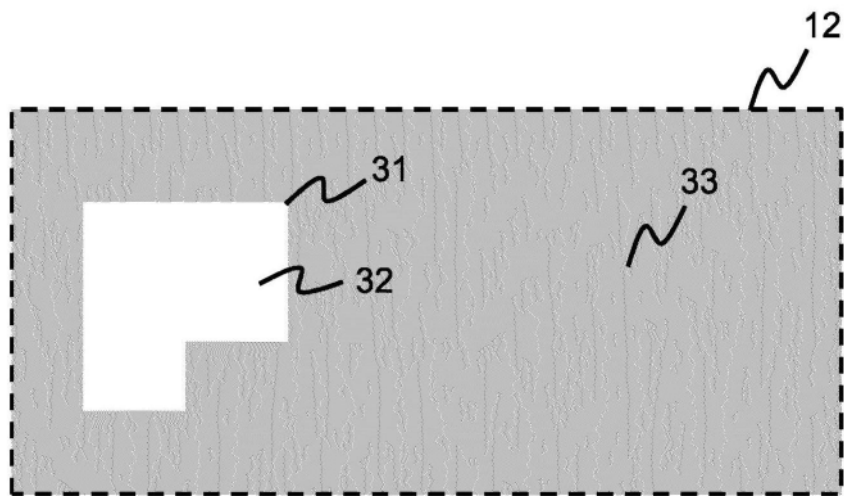


图3B

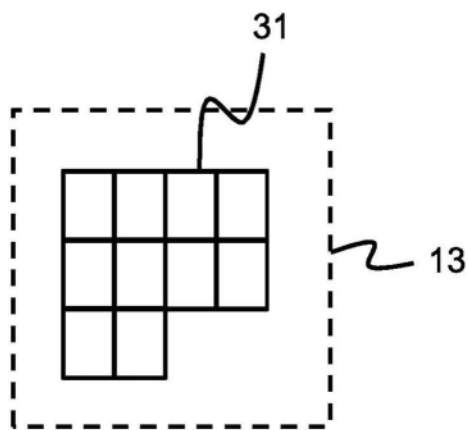


图3C

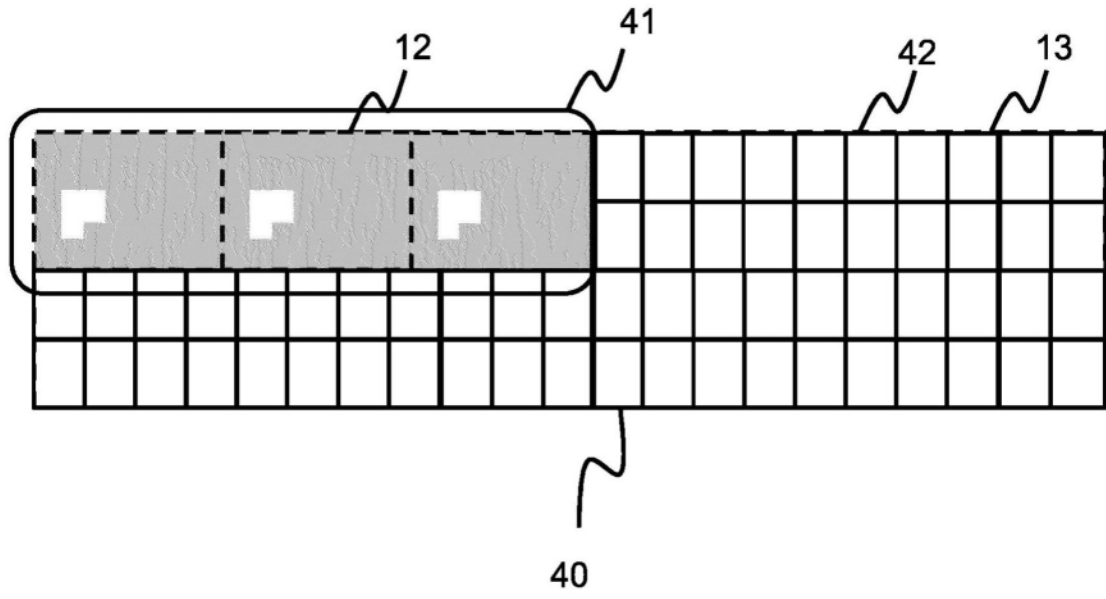


图4

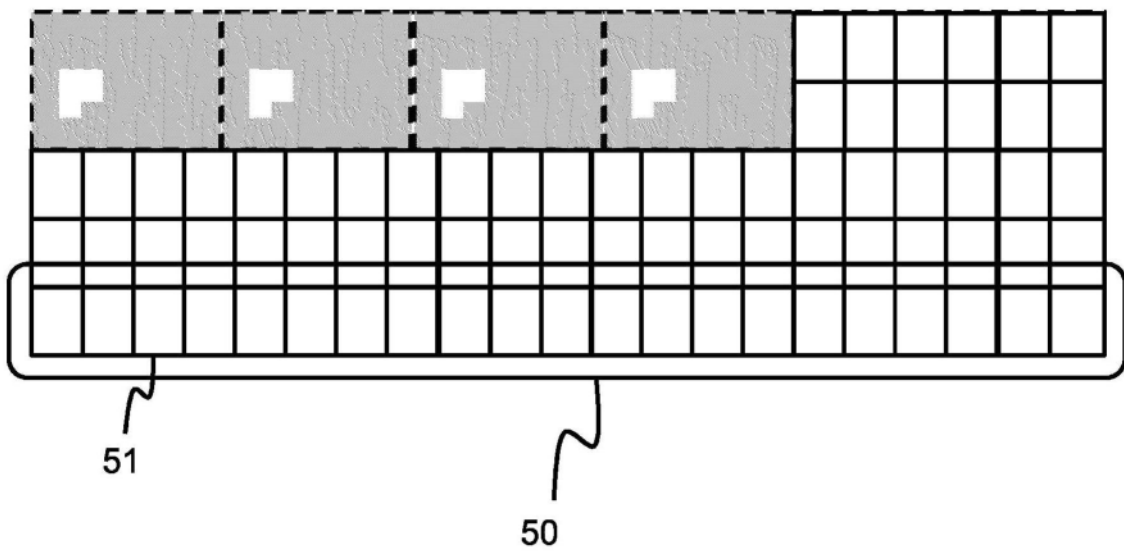


图5A

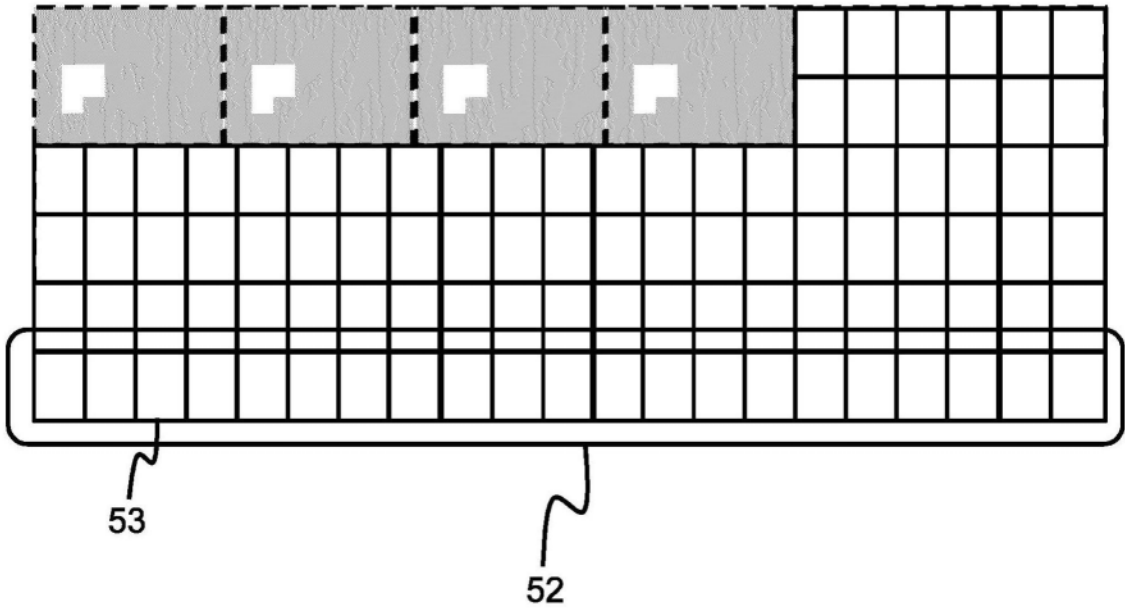


图5B

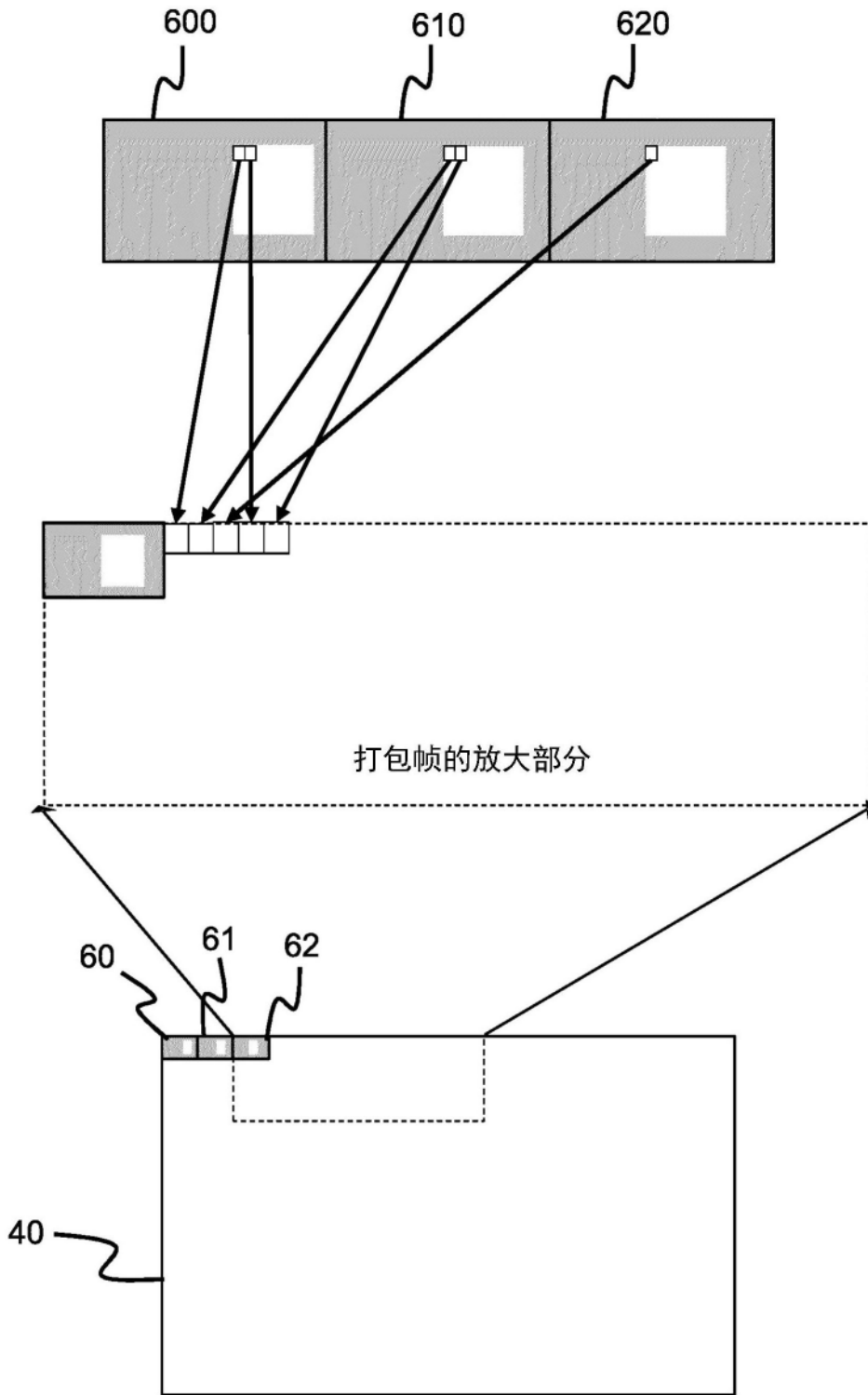


图6

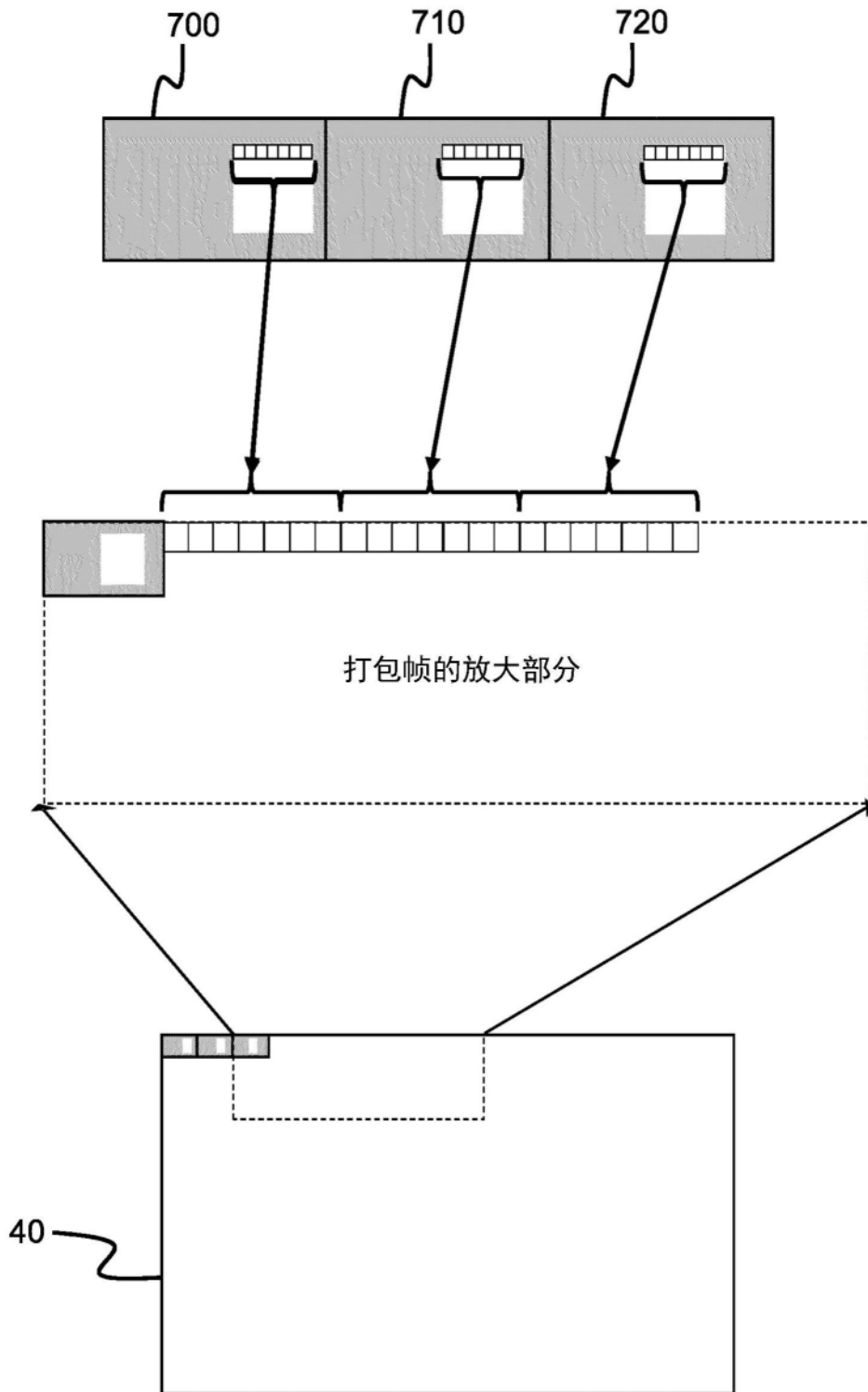


图7

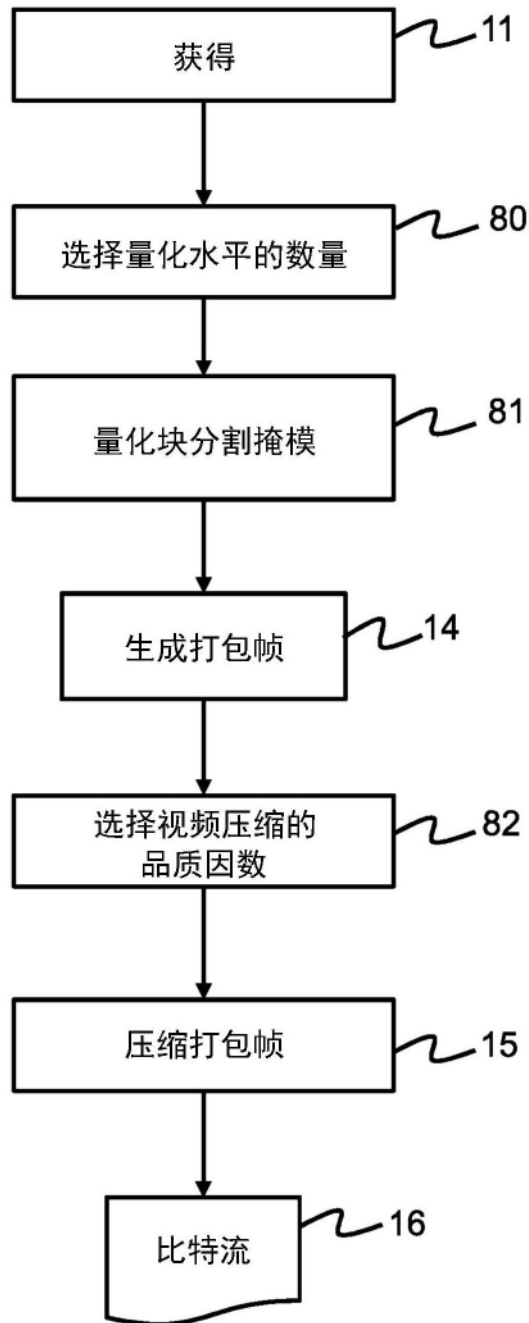


图8

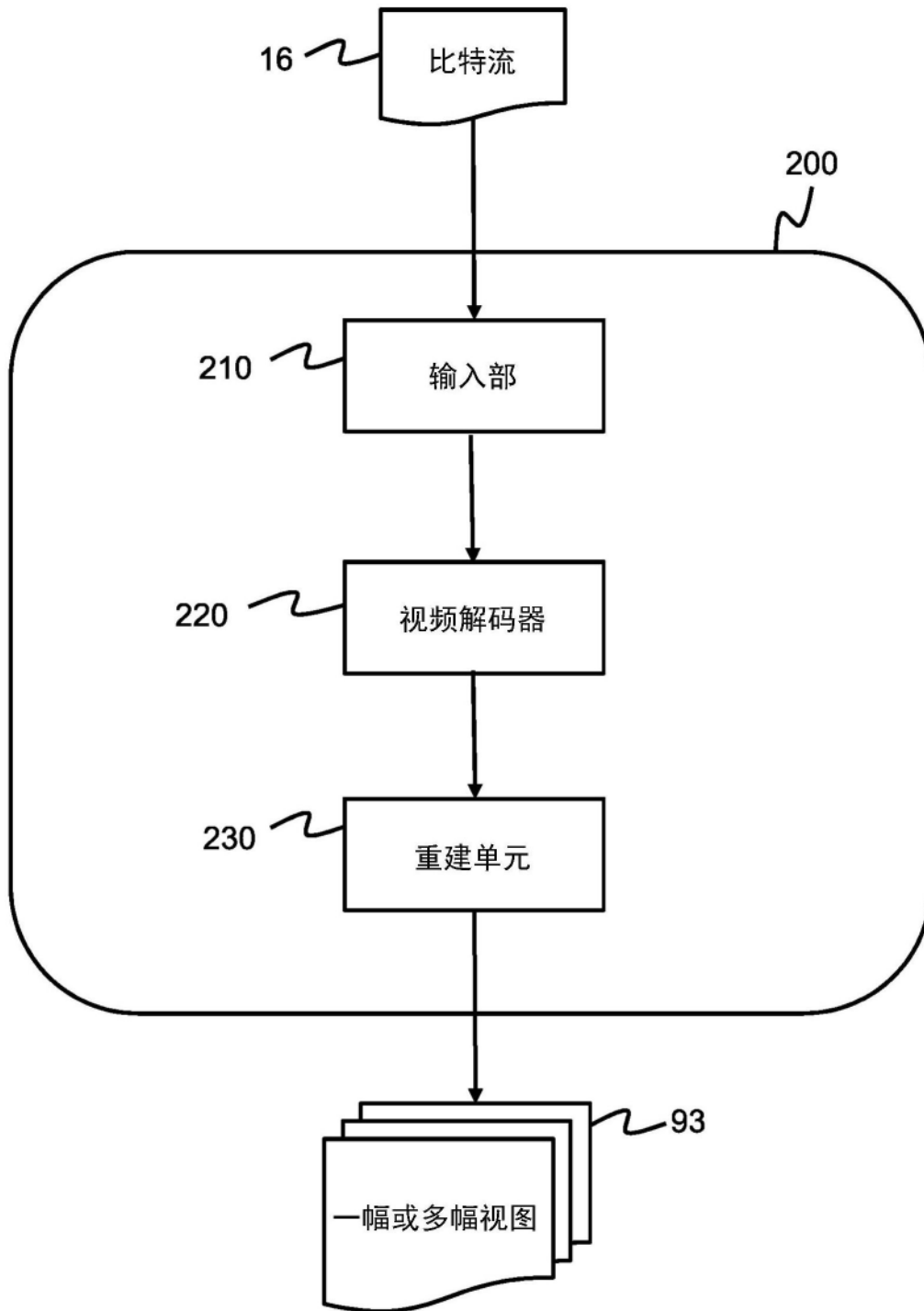


图9

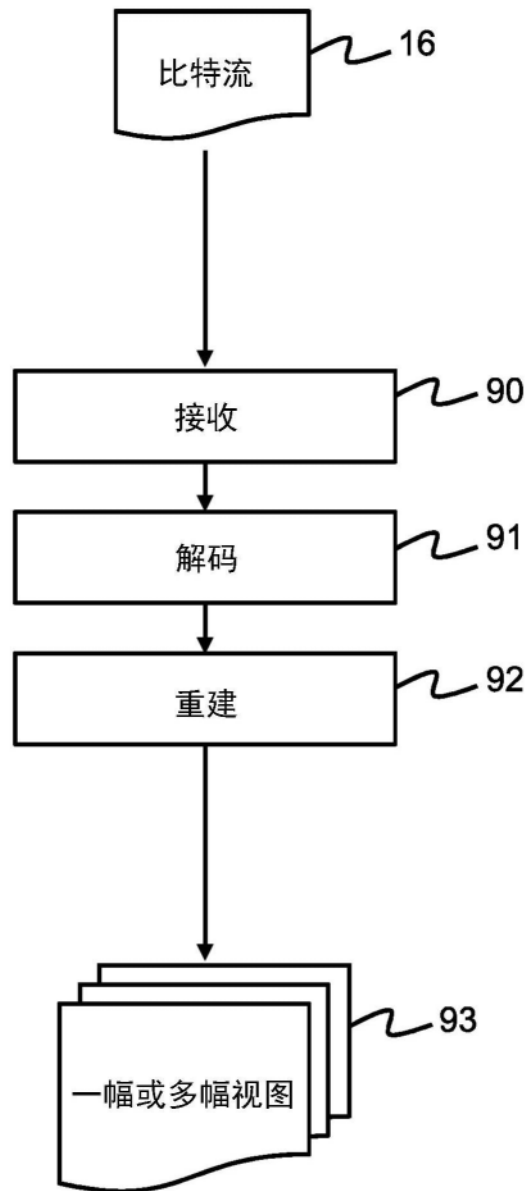


图10

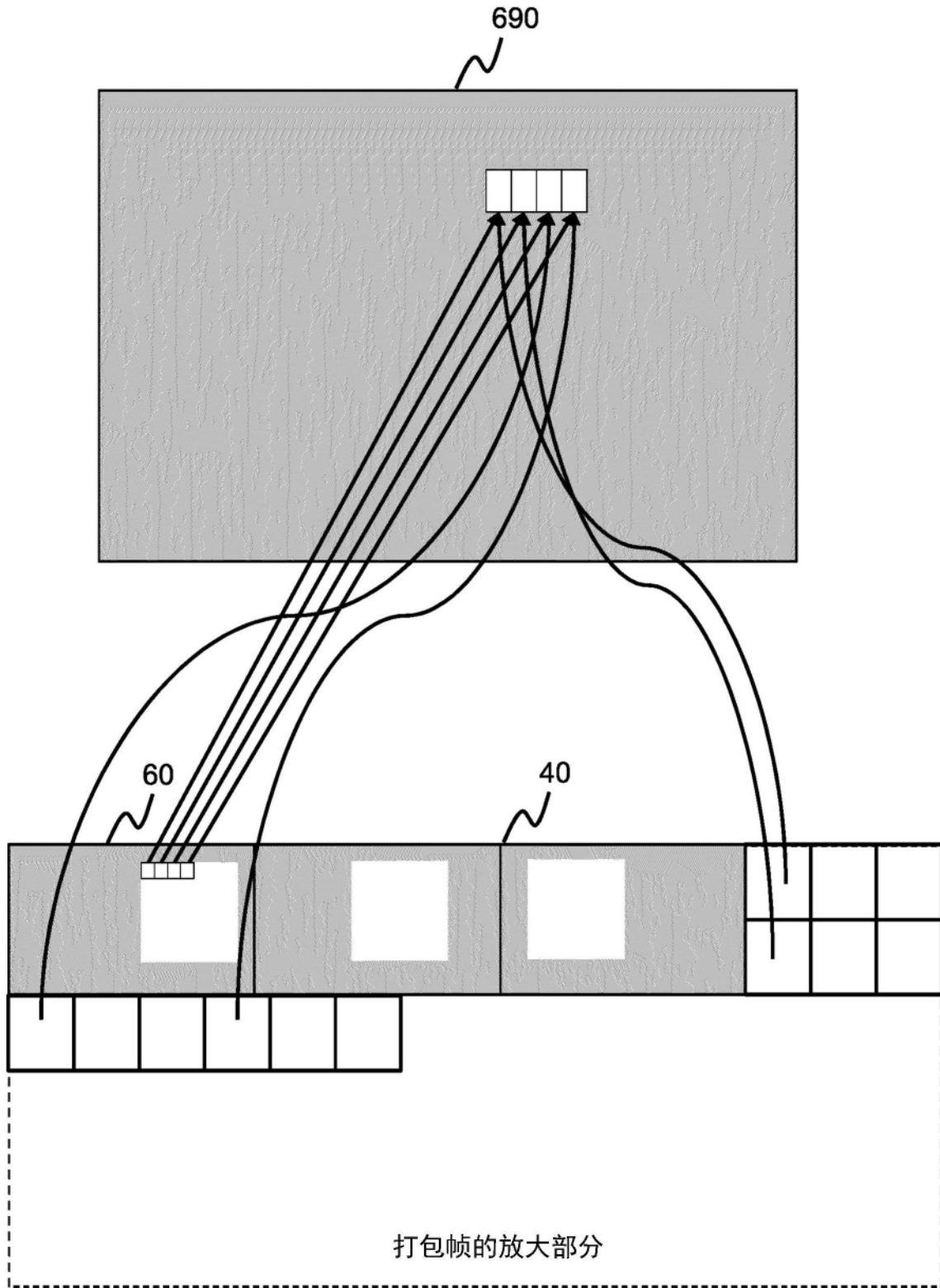


图11