



(10) 授权公告号 CN 115486078 B

(45) 授权公告日 2025. 04. 04

(21) 申请号 202280003790.1

(22) 申请日 2022.01.18

(65) 同一申请的已公布的文献号
申请公布号 CN 115486078 A

(43) 申请公布日 2022.12.16

(30) 优先权数据
63/175,929 2021.04.16 US
17/569,681 2022.01.06 US

(85) PCT国际申请进入国家阶段日
2022.10.27

(86) PCT国际申请的申请数据
PCT/US2022/012725 2022.01.18

(87) PCT国际申请的公布数据
W02022/220898 EN 2022.10.20

(73) 专利权人 腾讯美国有限责任公司
地址 美国加利福尼亚州帕洛阿尔托公园大
道2747号

(72) 发明人 赵亮 赵欣 刘杉

(74) 专利代理机构 北京派特恩知识产权代理有
限公司 11270
专利代理师 王花丽 张颖玲

(51) Int.Cl.
H04N 19/593 (2006.01)
H04B 1/66 (2006.01)
H04N 19/103 (2006.01)
H04N 19/105 (2006.01)
H04N 19/11 (2006.01)
H04N 19/176 (2006.01)

(56) 对比文件
US 2019141318 A1, 2019.05.09
审查员 刘丽媛

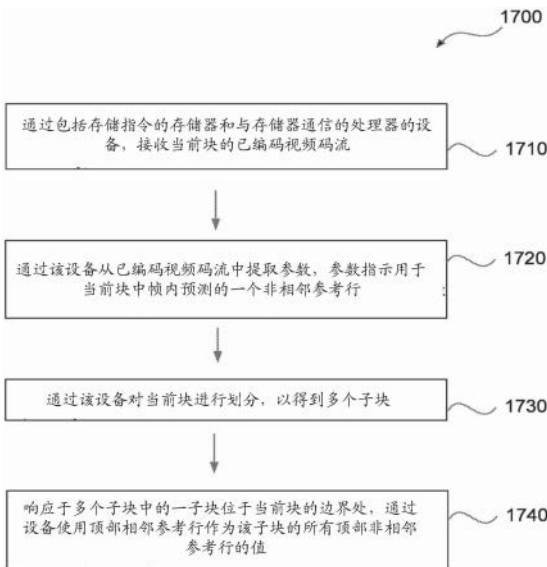
权利要求书1页 说明书25页 附图21页

(54) 发明名称

用于视频解码中的多参考行帧内预测的方法、装置及介质

(57) 摘要

用于视频解码中的多参考行帧内预测的方法、装置和计算机可读存储介质。该方法包括：通过设备接收当前块的已编码视频码流。设备包括存储指令的存储器和与存储器通信的处理器。该方法还包括：通过设备从已编码视频码流中提取参数，参数指示用于当前块中帧内预测的参考行；通过设备对当前块进行划分，以得到多个子块；以及响应于多个子块中的一子块位于当前块的边界处，通过设备使用顶部相邻参考行作为该子块的所有顶部非相邻参考行的值。



1. 一种用于视频解码中的多参考行帧内预测的方法,其特征在于,所述方法包括:
通过设备接收当前块的已编码视频码流,所述设备包括存储指令的存储器和与所述存储器通信的处理器;的设备;
确定对所述当前块应用多参考行选择;
通过所述设备从所述已编码视频码流中提取参数,所述参数指示用于所述当前块中帧内预测的一个非相邻参考行;
通过所述设备对所述当前块进行划分,以得到多个子块;以及
响应于所述多个子块中的一子块位于所述当前块的边界处、且对所述当前块应用了所述多参考行选择,通过所述设备使用与相邻参考行对应的左上值替换与所述子块的非相邻参考行索引对应的非相邻参考行的左上值。
2. 根据权利要求1所述的方法,其特征在于,所述方法进一步包括:
通过所述设备使用所述子块的左侧相邻参考行。
3. 根据权利要求1所述的方法,其特征在于,
所述当前块包括以下至少之一:超级块、最大编码块、编码树块CTB、最大编码单元LCU、具有预定大小的预定块。
4. 根据权利要求1所述的方法,其特征在于,
所述当前块的边界包括以下之一:所述当前块的顶部边界、所述当前块的左侧边界、或者所述当前块的左侧边界和顶部边界。
5. 根据权利要求1所述的方法,其特征在于,所述方法进一步包括:
将来自顶部相邻参考行的样本复制到所有其它顶部非相邻参考行。
6. 根据权利要求5所述的方法,其特征在于,所述方法进一步包括:
通过所述设备对所述子块进行划分,以得到多个变换块。
7. 根据权利要求6所述的方法,其特征在于,所述方法进一步包括:
响应于所述多个变换块中的第一变换块位于所述子块的顶部边界处,通过所述设备使用所述顶部相邻参考行作为所述第一变换块的所有顶部非相邻参考行的值;以及
响应于所述多个变换块中的第二变换块没有位于所述子块的顶部边界处,通过所述设备针对所述第二变换块使用由所述第二变换块的参数指示的参考行。
8. 一种用于视频解码中的多参考行帧内预测的装置,其特征在于,所述装置包括:
存储指令的存储器;和
与所述存储器通信的处理器,其中,当所述处理器执行所述指令时,所述处理器配置成使得所述装置执行根据权利要求1至7中任一项所述的方法。
9. 一种非暂时性计算机可读存储介质,存储有指令,其特征在于,当所述指令由处理器执行时,所述指令配置成使得所述处理器执行根据权利要求1至7中任一项所述的方法。
10. 一种存储视频码流的方法,其特征在于,所述视频码流基于权利要求1至7任一项所述的方法进行解码。

用于视频解码中的多参考行帧内预测的方法、装置及介质

[0001] 相关申请

[0002] 本申请基于并要求于2021年4月16日提交的第63/175,929号美国临时申请和于2022年1月6日提交的第17/569,681号美国非临时申请的优先权权益,这两个申请通过引用整体并入本文中。

技术领域

[0003] 本公开涉及视频编码和/或解码技术,具体地涉及多参考行选择方案的改进的低存储设计和信令。

背景技术

[0004] 本文所提供的背景描述是出于总体上呈现本公开内容的目的。在该背景技术部分以及本说明书的各个方面中所描述的、目前已署名的发明人的工作所进行的程度,并不表明其在本申请提交时有资格作为现有技术,且从未明示或暗示其被承认为本公开内容的现有技术。

[0005] 可使用具有运动补偿的帧间图片预测来执行视频编码和解码。未压缩的数字视频可包括一系列图片,每个图片具有例如为 1920×1080 的亮度样本及相关联的完全或子采样的色度样本的空间大小。该一系列图片可具有例如每秒60幅图片或每秒60帧的固定或可变的图片速率(替代地,称为帧率)。对于流式传输或数据处理,未压缩的视频具有特定比特率要求。例如,具有像素分辨率为 1920×1080 、帧率为60帧/秒、色度子采样为4:2:0、每个像素8位/颜色通道的视频需要接近1.5Gbit/s的带宽。一小时的此类视频需要600GB以上的存储空间。

[0006] 视频编码和解码的一个目的可以是通过压缩来减少未压缩的输入视频信号中的冗余。压缩可有助于减小上述带宽和/或存储空间需求,在某些情况下可减小两个数量级或大于两个数量级。可采用无损压缩和有损压缩以及它们的组合。无损压缩指的是可通过解码过程从已压缩的原始信号中重建原始信号的精确副本的技术。有损压缩指的是以下编码/解码过程:原始视频信息在编码期间没有完全保留,且在解码期间不能完全恢复。当使用有损压缩时,已重建的信号可能与原始信号不同,但是原始信号和已重建的信号之间的失真足够小,以使已重建的信号可用于预期的应用,尽管会损失一些信息。在视频的情况下,在许多应用中广泛采用有损压缩。可容忍的失真量取决于应用。例如,某些消费视频流式应用的用户相比电影或电视广播应用的用户来说可以容忍更高的失真。可选择或调节可通过特定编码算法实现的压缩率,以反映各种失真容忍度:更高的可容忍的失真通常允许编码算法产生更高的损失和更高的压缩率。

[0007] 视频编码器和解码器可利用来自多个宽泛类别的技术和步骤,这些步骤包括例如运动补偿、傅里叶变换、量化和熵编码。

[0008] 视频编解码器技术可包括称为帧内编码的技术。在帧内编码中,在不参考来自先前重建的参考图片的样本或其它数据的情况下表示样本值。在一些视频编解码器中,图片

在空间上细分成样本块。当所有样本块都以帧内模式编码时,该图片可称为帧内图片。帧内图片及其派生物(例如,独立解码器刷新图片)可用于重置解码器状态,因此可用作已编码视频码流和视频会话中的第一张图片,或者用作静止图像。然后,帧内预测之后的块的样本可进行变换,而变换到频域中,且可以在熵编码之前对如此生成的变换系数进行量化。帧内预测表示一种使预变换域中的样本值最小化的技术。在一些情况下,变换之后的DC值越小,且AC系数越小,则在给定的量化步长下表示熵编码之后的块所需的位越少。

[0009] 例如从诸如MPEG-2代编码技术已知的传统帧内编码不使用帧内预测。然而,一些更新的视频压缩技术包括试图基于例如周围样本数据和/或元数据对块进行编码/解码的技术,该周围样本数据和/或元数据在空间相邻的编码和/或解码期间获得并且按照解码顺序排在正在被帧内编码或解码的数据块之前。在下文中这样的技术称为“帧内预测”技术。应注意,至少在一些情况下,帧内预测仅使用来自正在重建的当前图片的参考数据,而不使用来自参考图片的参考数据。

[0010] 帧内预测可以有許多不同形式。当在给定的视频编码技术中可使用不止一种这样的技术时,使用中的技术可称为帧内预测模式。可以在特定编解码器中提供一种或多种帧内预测模式。在一些情况下,模式可具有子模式和/或可与各种参数相关联,且模式/子模式信息和视频块的帧内编码参数可单独地编码或共同地包含在模式码字中。给定的模式、子模式和/或参数组合使用哪个码字,可能会影响通过帧内预测的编码效率增益,因此可能会影响用于将码字转换成码流的熵编码技术。

[0011] H.264引入了一种帧内预测模式,该帧内预测模式在H.265中得到完善,且在诸如联合探索模型(JEM)、下一代视频编码(VVC)和基准集(BMS)等更新的编码技术中进一步得到完善。通常,对于帧内预测,可使用属于已经变成可用的相邻样本值来形成预测值块(predictor block)。例如,特定相邻样本集的可用值可沿着一定方向和/或行复制到预测值块中。对使用方向的参考可以在码流中进行编码,或者其本身可被预测。

[0012] 参考图1A,在右下方描绘了在H.265的33种可能的帧内预测方向(对应于H.265中指定的35个帧内模式的33个角度模式)中指定的9个预测方向的子集。箭头汇聚的点(101)表示正在被预测的样本。箭头表示使用相邻样本预测101处的样本所沿的方向。例如,箭头(102)指示根据在右上方、与水平方向成45度角的一个或多个相邻样本预测样本(101)。类似地,箭头(103)指示根据在样本(101)的左下方、与水平方向成22.5度角的一个或多个相邻样本预测样本(101)。

[0013] 仍然参考图1A,在左上方描绘了 4×4 个样本的正方形块(104)(由粗体虚线指示)。正方形块(104)包括16个样本,每个样本用“S”、其在Y维度上的位置(例如,行索引)和其在X维度上的位置(例如,列索引)来标记。例如,样本S21是在Y维度上(从顶部开始)的第二个样本和在X维度上(从左侧开始)的第一个样本。类似地,样本S44是在Y和X维度上、块(104)中的第四个样本。由于块的大小为 4×4 个样本,因此S44位于右下角。还示出了遵循类似编号方案的示例性参考样本。参考样本用R、其相对于块(104)的Y位置(例如,行索引)和X位置(列索引)来标记。在H.264和H.265中,使用与正在重建的块相邻的预测样本。

[0014] 块104的帧内图片预测可通过根据用信号表示的预测方向从相邻样本复制参考样本值来开始。例如,假设已编码视频码流包括信令,该信令针对该块104指示箭头(102)的预测方向,即,根据在右上方、与水平方向成45度角的一个或多个预测样本来预测样本。在这

种情况下,根据同一个参考样本R05预测样本S41、S32、S23和S14。然后根据参考样本R08预测样本S44。

[0015] 在一些情况下,可例如通过插值来组合多个参考样本的值,以计算参考样本;尤其是当方向不能被45度整除时。

[0016] 随着视频编码技术继续发展,可能的方向的数量增加。在H.264(2003年)中,例如九个不同的方向可用于帧内预测。在H.265(2013年)中,增加到33个方向,以及在本公开时,JEM/VVC/BMS可支持多达65个方向。已进行实验研究来帮助识别最合适的帧内预测方向,且熵编码中的一些技术可用于以小的比特对那些最合适的方向进行编码,对于方向,接受一定的比特代价。此外,有时可以根据已解码的相邻块的帧内预测中所使用的相邻方向来预测方向本身。

[0017] 图1B示出了示意图(180),其描绘了根据JEM的65个帧内预测方向,以说明在随着时间的推移而开发的各種编码技术中,预测方向的数量增加。

[0018] 在已编码视频码流中,将表示帧内预测方向的比特映射到该预测方向的方式,可能因视频编码技术的不同而不同;例如,其范围可以从预测方向简单直接映射到帧内预测模式,变化为预测方向映射到码字,映射到涉及最可能模式的复杂自适应方案,以及类似技术。然而,在所有情况下,可存在某些帧内预测方向,这些方向与某些其它方向相比,在统计上出现在视频内容中的可能性较小。由于视频压缩的目标是减少冗余,因此在一种设计良好的视频编码技术中,相比于更可能的方向,那些不太可能的方向可由更多的位数表示。

[0019] 帧间图片预测或帧间预测可基于运动补偿。在运动补偿中,来自先前重建的图片或其部分(参考图片)的样本数据在沿着由运动矢量(此后称为MV)指示的方向在空间上偏移之后,可用于预测新重建的图片或图片部分(例如,块)。在一些情况下,参考图片可与当前正在重建的图片相同。MV可具有X和Y两个维度,或具有三个维度,第三个维度为正在使用的参考图片的指示(类似于时间维度)。

[0020] 在一些视频压缩技术中,可根据其它MV,例如根据在空间上与正在重建的区域相邻的样本数据的另一区域相关的、且按解码次序处于当前MV之前的其它MV来预测适用于样本数据的某个区域的该当前MV。这样做可通过消除相关MV中的冗余来大大减少对MV进行编码所需的数据总量,从而增加压缩效率。MV预测可有效地工作,例如,由于在对从相机获得的输入视频信号(称为自然视频)进行编码时,存在以下统计可能性:比适用单个MV的区域更大的区域在视频序列中沿着相似的方向移动,因此在某些情况下,可使用从相邻区域的MV导出的相似运动矢量来预测该更大的区域。这使得用于给定区域的实际MV与根据周围MV所预测的MV相似或相同。进而在熵编码之后,该MV可以采用比直接对MV进行编码(而非根据相邻MV来预测该MV)时使用的位数更小的位数来进行表示。在一些情况下,MV预测可以是无损压缩从原始信号(即:样本流)中导出的信号(即:MV)的示例。在其它情况下,例如由于根据多个周围MV计算预测值时出现舍入误差,使得MV预测本身可以是有损的。

[0021] H.265/HEVC(ITU-T H.265建议书,“High Efficiency Video Coding(高效视频编码)”,2016年12月)中描述了各种MV预测机制。在H.265指定的多种MV预测机制中,下面描述的是下文称为“空间合并(spatial merge)”的技术。

[0022] 具体地,参考图2,当前块(201)包括在运动搜索过程期间已由编码器找到的样本,可根据已产生空间偏移的相同大小的先前块来预测该样本。可以从与一个或多个参考图片

相关联的元数据中导出MV,而非直接对该MV进行编码,例如使用与被标记为A0、A1和B0、B1、B2(分别对应202到206)的五个周围样本中的任一样本相关联的MV,(按解码次序)从最近的参考图片中导出该MV。在H.265中,MV预测可使用来自相邻块正在使用的同一参考图片的预测值。

发明内容

[0023] 本公开描述了用于视频编码和/或解码的方法、装置和计算机可读存储介质的各种实施例。

[0024] 根据一方面,本公开的一个实施例提供一种用于视频解码中的多参考行帧内预测的方法。该方法包括:通过设备接收当前块的已编码视频码流。设备包括存储指令的存储器与与存储器通信的处理器。该方法还包括:通过设备从已编码视频码流中提取参数,参数指示用于当前块中帧内预测的一个非相邻参考行;通过设备对当前块进行划分,以得到多个子块;以及响应于多个子块中的一子块位于当前块的边界处,通过设备使用顶部相邻参考行作为该子块的所有顶部非相邻参考行的值。

[0025] 根据另一方面,本公开的一个实施例提供一种用于视频编码和/或解码的装置。该装置包括:存储指令的存储器;和与存储器通信的处理器。当处理器执行指令时,处理器配置成使得该装置执行上述用于视频解码和/或编码的方法。

[0026] 在另一方面,本公开的一个实施例提供一种非暂时性计算机可读介质,非暂时性计算机可读介质存储有指令,当该指令由用于视频解码和/或编码的计算机执行时,该指令使得该计算机执行上述用于视频解码和/或编码的方法。

[0027] 上述和其它方面及其实现方式在附图、说明书和权利要求中更详细地描述。

附图说明

[0028] 根据以下详细描述和附图,所公开的主题的进一步的特征、性质和各种优点将更加明显。

[0029] 图1A示出了帧内预测定向模式的示例性子集的示意性图示。

[0030] 图1B示出了示例性帧内预测方向的图示。

[0031] 图2示出了在一个示例中,用于运动矢量预测的当前块及其周围空间合并候选的示意性图示。

[0032] 图3示出了根据一个示例性实施例的通信系统(300)的简化框图的示意性图示。

[0033] 图4示出了根据一个示例性实施例的通信系统(400)的简化框图的示意性图示。

[0034] 图5示出了根据一个示例性实施例的视频解码器的简化框图的示意性图示。

[0035] 图6示出了根据一个示例性实施例的视频编码器的简化框图的示意性图示。

[0036] 图7示出了根据另一示例性实施例的视频编码器的框图。

[0037] 图8示出了根据另一示例性实施例的视频解码器的框图。

[0038] 图9示出了根据本公开的示例性实施例的编码块划分的方案。

[0039] 图10示出了根据本公开的示例性实施例的编码块划分的另一方案。

[0040] 图11示出了根据本公开的示例性实施例的编码块划分的另一方案。

[0041] 图12示出了根据本公开的示例性实施例的编码块划分的另一方案。

[0042] 图13示出了根据本公开的示例性实施例的将编码块划分成多个变换块的方案以及变换块的编码顺序。

[0043] 图14示出了根据本公开的示例性实施例的将编码块划分成多个变换块的另一方案以及变换块的编码顺序。

[0044] 图15示出了根据本公开的示例性实施例的将编码块划分成多个变换块的另一方案。

[0045] 图16示出了根据本公开的示例性实施例的基于各个参考行的帧内预测方案。

[0046] 图17示出了根据本公开的一个示例性实施例的方法的流程图。

[0047] 图18示出了根据本公开的示例性实施例的基于多个参考行的帧内预测方案。

[0048] 图19示出了根据本公开的示例性实施例的基于多个参考行的帧内预测方案。

[0049] 图20示出了根据本公开的示例性实施例的计算机系统的示意性图示。

具体实施方式

[0050] 现在,将在下文中参考附图详细描述本发明,附图构成本发明的一部分并通过图示的方式示出了实施例的特定示例。然而,请注意,本发明可以以多种不同的形式体现,因此,所涵盖或要求保护的主题旨在被解释为不限于在下文中阐述的任何实施例。还请注意,本发明可体现为方法、设备、组件或系统。因此,本发明的实施例可例如采取硬件、软件、固件或它们的任何组合的形式。

[0051] 在整个说明书和权利要求中,术语除了具有明确叙述的含义之外,还可具有上下文中启示或暗示的细微差别的含义。如本文使用的短语“在一个实施例中”或“在一些实施例中”不一定指的是同一个实施例,且如本文使用的短语“在另一实施例中”或“在其它实施例中”不一定指的是不同的实施例。类似地,如本文使用的短语“在一个实现方式中”或“在一些实现方式中”不一定指的是同一个实现方式,且如本文所使用的短语“在另一实现方式中”或“在其它实现方式中”不一定指的是不同的实现方式。例如,意图在于:所要求保护的主体包括全部或部分示例性实施例/实现方式的组合。

[0052] 通常,术语可以至少部分地从上下文中的用法来理解。例如,如本文使用的诸如“和”、“或”或者“和/或”之类的术语可包括各种含义,这些含义可以至少部分地取决于使用此类术语的上下文。通常,如果“或”用于关联列表(例如A、B或C),则“或”旨在表示A、B和C(这里以包含性意义使用)以及A、B或者C(这里以排他性意义使用)。此外,如本文使用的术语“一个或多个”或“至少一个”至少部分地取决于上下文,可用于以单数意义来描述任何特征、结构或特性,或者可用于以复数意义来描述特征、结构或特性的组合。类似地,诸如“一”、“一个”或“该”之类的术语还可理解为传达单数用法或传达复数用法,而这至少部分地取决于上下文。此外,术语“基于”或“由……确定”可理解为不一定旨在传达一组排他性因素,而是可允许存在不一定明确描述的附加因素,而这至少部分地取决于上下文。

[0053] 图3示出了根据本公开的一个实施例的通信系统(300)的简化框图。通信系统(300)包括多个终端设备,该多个终端设备可通过例如网络(350)彼此通信。例如,通信系统(300)包括通过网络(350)互连的第一终端设备对(310)和(320)。在图3的示例中,第一终端设备对(310)和(320)可执行单向数据传输。例如,终端设备(310)可以对视频数据(例如,由终端设备(310)采集的视频图片流的视频数据)进行编码,以通过网络(350)传输到另一终

端设备 (320)。已编码视频数据可以以一个或多个已编码视频码流的形式传输。终端设备 (320) 可以从网络 (350) 接收已编码视频数据,对已编码视频数据进行解码以恢复视频图片,以及根据所恢复的视频数据显示视频图片。单向数据传输可以在媒体服务等应用中实现。

[0054] 在另一示例中,通信系统 (300) 包括执行已编码视频数据的双向传输的第二终端设备对 (330) 和 (340),该双向传输可例如在视频会议应用期间实现。对于双向数据传输,在一个示例中,终端设备 (330) 和 (340) 中的每个终端设备可以对视频数据 (例如,由终端设备采集的视频图片流的视频数据) 进行编码,以通过网络 (350) 传输到终端设备 (330) 和 (340) 中的另一终端设备。终端设备 (330) 和 (340) 中的每个终端设备还可接收由终端设备 (330) 和 (340) 中的另一终端设备传输的已编码视频数据,且可以对已编码视频数据进行解码以恢复视频图片,以及可根据所恢复的视频数据在可访问的显示设备上显示视频图片。

[0055] 在图3的示例中,终端设备 (310)、终端设备 (320)、终端设备 (330) 和终端设备 (340) 可实现为服务器、个人计算机和智能电话,但是本公开的潜在原理的适用性可不限于此。本公开的实施例可以在台式计算机、膝上型计算机、平板电脑、媒体播放器、可穿戴计算机、专用视频会议设备等中实现。网络 (350) 表示在终端设备 (310)、终端设备 (320)、终端设备 (330) 和终端设备 (340) 之间传送已编码视频数据的任何数量的网络,包括例如有线 (连线的) 和/或无线通信网络。通信网络 (350) 可以在电路交换信道、分组交换信道和/或其它类型的信道中交换数据。代表性的网络包括电信网络、局域网、广域网和/或互联网。出于本讨论的目的,除非在本文中予以明确说明,否则网络 (350) 的架构和拓扑对于本公开的操作来说可能是无关紧要的。

[0056] 作为用于所公开的主题的应用的示例,图4示出了视频编码器和视频解码器在视频流式传输环境中的放置方式。所公开的主题可同等地适用于其它视频应用,包括例如视频会议、数字TV、广播、游戏、虚拟现实、在包括CD、DVD、存储棒等的数字介质上存储压缩视频等等。

[0057] 视频流式传输系统可包括视频采集子系统 (413),采集子系统 (413) 可包括诸如数码相机和视频源 (401),视频源 (401) 用于创建未压缩的视频图片或图像流 (402)。在一个示例中,视频图片流 (402) 包括由视频源401的数码相机记录的样本。相较于已编码视频数据 (404) (或已编码视频码流),被描绘为粗线以强调高数据量的视频图片流 (402) 可由电子设备 (420) 处理,电子设备 (420) 包括耦接到视频源 (401) 的视频编码器 (403)。视频编码器 (403) 可包括硬件、软件或软硬件组合,以实现或实施如下文更详细地描述的所公开的主题的各方面。相较于未压缩的视频图片流 (402),被描绘为细线以强调较低数据量的已编码视频数据 (404) (或已编码视频码流 (404)) 可存储在流式传输服务器 (405) 上以供将来使用,或者直接存储于下游视频设备 (未示出)。一个或多个流式传输客户端子系统,例如图4中的客户端子系统 (406) 和客户端子系统 (408) 可访问流式传输服务器 (405) 以检索已编码视频数据 (404) 的副本 (407) 和副本 (409)。客户端子系统 (406) 可包括例如电子设备 (430) 中的视频解码器 (410)。视频解码器 (410) 对已编码视频数据的传入副本 (407) 进行解码,且产生可以在显示器 (412) (例如,显示屏) 或其它呈现设备 (未描绘) 上呈现的未压缩的输出视频图片流 (411)。视频解码器410可配置成执行本公开所描述的各种功能中的一些或所有功能。在一些流式传输系统中,可根据某些视频编码/压缩标准对已编码视频数据 (404)、

(407) 和 (409) (例如, 视频码流) 进行编码。这些标准的示例包括ITU-T H.265建议书。在一个示例中, 正在开发的视频编码标准非正式地称为下一代视频编码 (VVC)。所公开的主题可用于VVC的上下文和其它视频编码标准中。

[0058] 应注意, 电子设备 (420) 和电子设备 (430) 可包括其它组件 (未示出)。例如, 电子设备 (420) 可包括视频解码器 (未示出), 且电子设备 (430) 还可包括视频编码器 (未示出)。

[0059] 在下文中, 图5示出了根据本公开的任意实施例的视频解码器 (510) 的框图。视频解码器 (510) 可包括在电子设备 (530) 中。电子设备 (530) 可包括接收器 (531) (例如, 接收电路)。视频解码器 (510) 可用于代替图4的示例中的视频解码器 (410)。

[0060] 接收器 (531) 可接收将由视频解码器 (510) 解码的一个或多个已编码视频序列。在同一实施例或另一实施例中, 一次可以对一个已编码视频序列进行解码, 其中每个已编码视频序列的解码独立于其它已编码视频序列。每个视频序列可与多个视频帧或图像相关联。可以从信道 (501) 接收已编码视频序列, 信道 (501) 可以是硬件/软件链路, 其通向存储已编码视频数据的存储设备或者传输已编码视频数据的流式传输源。接收器 (531) 可接收可转发到它们各自的处理电路 (未描绘) 的已编码视频数据和其它数据, 例如已编码音频数据和/或辅助数据流。接收器 (531) 可以将已编码视频序列与其它数据分开。为了防止网络抖动, 缓冲存储器 (515) 可设置在接收器 (531) 和熵解码器/解析器 (520) (此后称为“解析器 (520)”) 之间。在某些应用中, 缓冲存储器 (515) 可实现为视频解码器 (510) 的一部分。在其它应用中, 缓冲存储器 (515) 可位于视频解码器 (510) 的外部且与视频解码器 (510) 分离 (未描绘)。而在另一些其它应用中, 在视频解码器 (510) 的外部可设置缓冲存储器 (未描绘), 其目的是例如防止网络抖动, 且在视频解码器 (510) 的内部可设置另一附加的缓冲存储器 (515) 以例如处理播放定时。当接收器 (531) 从具有足够带宽和可控性的存储/转发设备或从等时同步网络接收数据时, 可能不需要缓冲存储器 (515), 或可以将缓冲存储器做得较小。为了在诸如互联网等业务分组网络上使用, 可能需要足够大小的缓冲存储器 (515), 缓冲存储器 (515) 的大小可相对较大。这种缓冲存储器可实现为具有自适应大小, 且可至少部分地在操作系统或视频解码器 (510) 外部的类似元件 (未描绘) 中实现。

[0061] 视频解码器 (510) 可包括解析器 (520), 以根据已编码视频序列重建符号 (521)。这些符号的类别包括用于管理视频解码器 (510) 的操作的信息, 以及用于控制诸如显示器 (512) (例如, 显示屏) 之类的呈现设备的潜在信息, 该呈现设备可以是或不是电子设备 (530) 的整体部分, 但是可耦接到电子设备 (530), 如图5所示。用于呈现设备的控制信息可以是辅助增强信息 (SEI消息) 或视频可用性信息 (VUI) 参数集片段 (未描绘) 的形式。解析器 (520) 可以对解析器 (520) 所接收的已编码视频序列进行解析/熵解码。已编码视频序列的熵编码可根据视频编码技术或标准进行, 且可遵循各种原理, 包括可变长度编码、霍夫曼 (Huffman) 编码、具有或不具有上下文敏感度的算术编码等。解析器 (520) 可基于对应于子群的至少一个参数, 从已编码视频序列提取用于视频解码器中的像素的子群中的至少一个子群的子群参数集。子群可包括图片群组 (GOP)、图片、图块、切片、宏块、编码单元 (CU)、块、变换单元 (TU)、预测单元 (PU) 等。解析器 (520) 还可以从已编码视频序列提取信息, 例如变换系数 (例如, 傅里叶变换系数), 量化器参数值, 运动矢量等。

[0062] 解析器 (520) 可以对从缓冲存储器 (515) 接收的视频序列执行熵解码/解析操作, 从而创建符号 (521)。

[0063] 取决于已编码视频图片或一部分已编码视频图片(例如:帧间图片和帧内图片,帧间块和帧内块)的类型以及其它因素,符号(521)的重建可涉及多个不同的处理或功能单元。涉及到的单元以及单元所涉及的方式可由解析器(520)通过从已编码视频序列解析的子群控制信息来控制。为了简单起见,未描绘解析器(520)与下文的多个处理或功能单元之间的此类子群控制信息流。

[0064] 除了已提及的功能块之外,视频解码器(510)可以在概念上细分成如下文所描述的多个功能单元。在商业约束下运行的实际实现方式中,这些功能单元中的许多功能单元彼此紧密交互且可至少部分地彼此集成。然而,出于清楚地描述所公开的主题的各种功能的目的,在本公开的下文中采用概念上细分成的多个功能。

[0065] 第一单元可包括缩放器/逆变换单元(551)。缩放器/逆变换单元(551)可以从解析器(520)接收作为符号(521)的量化变换系数以及控制信息,包括指示要使用哪种类型的逆变换、块大小、量化因子/参数、量化缩放矩阵等信息。缩放器/逆变换单元(551)可输出包括样本值的块,样本值可输入到聚合器(555)中。

[0066] 在一些情况下,缩放器/逆变换(551)的输出样本可属于帧内编码块;即,不使用来自先前重建的图片的预测信息,但是可使用来自当前图片的先前重建部分的预测信息的块。此类预测信息可由帧内图片预测单元(552)提供。在一些情况下,帧内图片预测单元(552)可使用已重建且存储在当前图片缓冲器(558)中的周围块信息来生成大小和形状与正在重建的块相同的块。例如,当前图片缓冲器(558)缓冲部分重建的当前图片和/或完全重建的当前图片。在一些实现方式中,聚合器(555)可基于每个样本,将帧内预测单元(552)生成的预测信息添加到由缩放器/逆变换单元(551)提供的输出样本信息中。

[0067] 在其它情况下,缩放器/逆变换单元(551)的输出样本可属于帧间编码和潜在运动补偿的块。在这种情况下,运动补偿预测单元(553)可访问参考图片存储器(557)以提取用于帧间图片预测的样本。在根据属于块的符号(521)对所提取的样本进行运动补偿之后,这些样本可由聚合器(555)添加到缩放器/逆变换单元(551)的输出(单元551的输出可称为残差样本或残差信号),从而生成输出样本信息。运动补偿预测单元(553)从参考图片存储器(557)内的地址提取预测样本可受到运动矢量控制,该运动矢量可以以符号(521)的形式提供给运动补偿预测单元(553)使用,符号(521)可具有例如X分量、Y分量(偏移)和参考图片分量(时间)。运动补偿还可包括在使用子样本精确运动矢量时,从参考图片存储器(557)提取的样本值的内插,运动补偿还可与运动矢量预测机制等相关联。

[0068] 聚合器(555)的输出样本可经受环路滤波器单元(556)中的各种环路滤波技术。视频压缩技术可包括环路内滤波器技术,该环路内滤波器技术受控于包括在已编码视频序列(还称为已编码视频码流)中且可作为来自解析器(520)的符号(521)可用于环路滤波器单元(556)的参数,然而,视频压缩技术还可响应于在对已编码图片或已编码视频序列的先前(按解码次序)部分进行解码期间获得的元信息,以及响应于先前重建且经过环路滤波的样本值。可以以各种顺序包括多种类型的环路滤波器,作为环路滤波器单元556的一部分,如下文将进一步详细描述。

[0069] 环路滤波器单元(556)的输出可以是样本流,该样本流可输出到呈现设备(512)以及存储在参考图片存储器(557)中以用于将来的帧间图片预测。

[0070] 一旦完全重建,某些已编码图片就可用作参考图片以用于将来的帧间图片预测。

例如,一旦对应于当前图片的已编码图片被完全重建,且已编码图片(通过例如解析器(520))被识别为参考图片,则当前图片缓冲器(558)可变成参考图片存储器(557)的一部分,且可以在开始重建后续已编码图片之前重新分配新的当前图片缓冲器。

[0071] 视频解码器(510)可根据诸如ITU-T H.265建议书的标准中采用的预定视频压缩技术执行解码操作。在已编码视频序列遵循视频压缩技术或标准的语法以及视频压缩技术或标准中记录的配置文件的意义上,已编码视频序列可符合所使用的视频压缩技术或标准指定的语法。具体而言,配置文件可以从视频压缩技术或标准中可用的所有工具中选择某些工具,作为在该配置文件下可供使用的仅有工具。为了符合标准,已编码视频序列的复杂度可处于视频压缩技术或标准的层级所限定的范围内。在一些情况下,层级限制最大图片大小、最大帧率、最大重建采样率(以例如每秒兆(mega)个样本为单位进行测量)、最大参考图片大小等。在一些情况下,由层级设定的限制可通过假想参考解码器(HRD)规范和在已编码视频序列中用信号表示的HRD缓冲器管理的元数据来进一步限定。

[0072] 在一些示例性实施例中,接收器(531)可以在接收已编码视频时接收附加(冗余)数据。附加数据可以被包括作为已编码视频序列的一部分。附加数据可由视频解码器(510)使用来对数据进行适当解码和/或更准确地重建原始视频数据。附加数据可采用例如时间、空间或信噪比(SNR)增强层、冗余切片、冗余图片、前向纠错码等形式。

[0073] 图6示出了根据本公开的一个示例性实施例的视频编码器(603)的框图。视频编码器(603)可包括在电子设备(620)中。电子设备(620)可进一步包括传输器(640)(例如,传输电路)。视频编码器(603)可用于代替图4的示例中的视频编码器(403)。

[0074] 视频编码器(603)可以从视频源(601)(并非图6的示例中的电子设备(620)的一部分)接收视频样本,视频源(601)可采集将由视频编码器(603)编码的视频图像。在另一示例中,视频源(601)可实现为电子设备(620)的一部分。

[0075] 视频源(601)可提供将由视频编码器(603)编码的呈数字视频样本流形式的源视频序列,该数字视频样本流可具有任何合适的位深度(例如:8位、10位、12位.....)、任何色彩空间(例如BT.601YCrCb,RGB,XYZ.....)和任何合适的采样结构(例如YCrCb 4:2:0,YCrCb 4:4:4)。在媒体服务系统中,视频源(601)可以是能够存储先前已准备的视频的存储设备。在视频会议系统中,视频源(601)可以是采集本地图像信息作为视频序列的相机。视频数据可作为多个单独的图片或图像来提供,当按顺序观看时,这些图片或图像被赋予运动。图片本身可构建为空间像素阵列,其中取决于所使用的采样结构、色彩空间等,每个像素可包括一个或多个样本。本领域的普通技术人员可容易地理解像素和样本之间的关系。下文侧重于描述样本。

[0076] 根据一些示例性实施例,视频编码器(603)可实时地或在应用所要求的任何其它时间约束下,将源视频序列的图片编码并压缩成已编码视频序列(643)。施行适当的编码速度构成了控制器(650)的一个功能。在一些实施例中,控制器(650)可以在功能上耦接到如下文所描述的其它功能单元且控制所述的其它功能单元。为了简单起见,图中未描绘耦接。由控制器(650)设置的参数可包括速率控制相关参数(图片跳过、量化器、率失真优化技术的 λ 值.....)、图片大小、图片群组(GOP)布局、最大运动矢量搜索范围等。控制器(650)可配置成具有其它合适的功能,这些功能涉及针对某一系统设计优化的视频编码器(603)。

[0077] 在一些示例性实施例中,视频编码器(603)可配置成在编码环路中进行操作。作为

过于简化的描述,在一个示例中,编码环路可包括源编码器(630)(例如,负责基于待编码的输入图片和参考图片来创建符号,例如符号流)和嵌入于视频编码器(603)中的(本地)解码器(633)。解码器(633)重建符号以用类似于(远程)解码器可创建样本数据的方式创建样本数据,即使所嵌入的解码器633处理在不进行熵编码的情况下由源编码器630编码的视频流(因为在所公开的主题中考虑的视频压缩技术中,熵编码中的符号与已编码视频码流之间的任何压缩可以是无损的)。重建的样本流(样本数据)输入到参考图片存储器(634)。由于符号流的解码产生与解码器位置(本地或远程)无关的位精确结果,因此参考图片存储器(634)中的内容在本地编码器和远程编码器之间也是按比特位精确对应的。换句话说,编码器的预测部分“看到”的参考图片样本与解码器将在解码期间使用预测时所“看到”的样本值完全相同。这种参考图片同步性基本原理(以及在例如因信道误差而无法维持同步性的情况下产生的漂移)用于改善编码质量。

[0078] “本地”解码器(633)的操作可与诸如已在上文结合图5详细描述的视频解码器(510)的“远程”解码器相同。然而,另外简要参考图5,由于符号可用且熵编码器(645)和解析器(520)能够无损地将符号编码/解码成已编码视频序列,因此包括缓冲存储器(515)和解析器(520)的视频解码器(510)的熵解码部分可能无法完全在编码器中的本地解码器(633)中实现。

[0079] 此时可以观察到,除了仅可存在于解码器中的解析/熵解码之外的任何解码器技术,还可必定需要以基本上相同的功能形式存在于对应的编码器中。出于此原因,所公开的主题有时可侧重于解码器操作,解码器操作与编码器的解码部分结合。因此,可简化编码器技术的描述,因为编码器技术与全面地描述的解码器技术互逆。在下文中,仅在某些区域或方面,提供编码器的更详细的描述。

[0080] 在操作期间,在一些示例性实现方式中,源编码器(630)可执行运动补偿预测编码,通过参考来自视频序列中被指定为“参考图片”的一个或多个先前已编码图片,该运动补偿预测编码对输入图片进行预测性编码。以这种方式,编码引擎(632)对输入图片的像素块和参考图片的像素块之间的颜色通道中的差异(或残差)进行编码,该参考图片可被选作该输入图片的预测参考。

[0081] 本地视频解码器(633)可基于源编码器(630)创建的符号,对可指定为参考图片的图片的已编码视频数据进行解码。编码引擎(632)的操作有利地可以是有损过程。当已编码视频数据可以在视频解码器(图6未示出)中被解码时,已重建视频序列通常可以是带有一些误差的源视频序列的副本。本地视频解码器(633)复制解码过程,该解码过程可由视频解码器对参考图片执行,且可使已重建参考图片存储在参考图片高速缓存(634)中。以这种方式,视频编码器(603)可以在本地存储已重建参考图片的副本,该副本与将由远端(远程)视频解码器获得的已重建参考图片具有共同内容(不存在传输误差)。

[0082] 预测器(635)可针对编码引擎(632)执行预测搜索。即,对于待编码的新图片,预测器(635)可以在参考图片存储器(634)中搜索可用作新图片的适当预测参考的样本数据(作为候选参考像素块)或某些元数据,例如参考图片运动矢量、块形状等。预测器(635)可基于样本块逐像素块操作,以找到合适的预测参考。在一些情况下,如由预测器(635)获得的搜索结果所确定的,输入图片可具有从参考图片存储器(634)中存储的多个参考图片取得的预测参考。

[0083] 控制器(650)可管理源编码器(630)的编码操作,包括例如设置用于对视频数据进行编码的参数和子群参数。

[0084] 可以在熵编码器(645)中对所有上述功能单元的输出生成熵编码。熵编码器(645)根据诸如霍夫曼编码、可变长度编码、算术编码等的技术来对各种功能单元生成的符号进行无损压缩,从而将该符号转换成已编码视频序列。

[0085] 传输器(640)可缓冲由熵编码器(645)创建的已编码视频序列,从而为通过通信信道(660)进行传输做准备,通信信道(660)可以是通向可存储已编码视频数据的存储设备的硬件/软件链路。传输器(640)可以将来自视频编码器(603)的已编码视频数据与待传输的其它数据合并,其它数据例如是已编码音频数据和/或辅助数据流(未示出来源)。

[0086] 控制器(650)可管理视频编码器(603)的操作。在编码期间,控制器(650)可以向每个已编码图片分配某一已编码图片类型,但这可能影响可应用于相应的图片的编码技术。例如,通常可以将图片分配为以下任一种图片类型:

[0087] 帧内图片(I图片),其可以是不将序列中的任何其它图片用作预测源就可被编码和解码的图片。一些视频编解码器容许不同类型的帧内图片,包括例如独立解码器刷新(“IDR”)图片。本领域的普通技术人员了解I图片的这些变体及其相应的应用和特征。

[0088] 预测性图片(P图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,该帧内预测或帧间预测使用至多一个运动矢量和参考索引来预测每个块的样本值。

[0089] 双向预测性图片(B图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,该帧内预测或帧间预测使用至多两个运动矢量和参考索引来预测每个块的样本值。类似地,多个预测性图片可使用多于两个参考图片和相关联的元数据来用于重建单个块。

[0090] 源图片通常可以在空间上细分成多个样本编码块(例如, 4×4 、 8×8 、 4×8 或 16×16 个样本的块),且逐块进行编码。这些块可参考其它(已编码)块进行预测性编码,其它(已编码)块由应用于块的相应图片的编码分配来确定。例如,I图片的块可进行非预测性编码,或者I图片的块可参考同一图片的已编码块来进行预测性编码(空间预测或帧内预测)。P图片的像素块可参考一个先前编码的参考图片通过空间预测或通过时间预测来进行预测性编码。B图片的块可参考一个或两个先前编码的参考图片通过空间预测或通过时间预测来进行预测性编码。出于其它目的,源图片或中间处理的图片可细分成其它类型的块。编码块和其它类型的块的划分可以或不遵循相同的方式,如下文进一步详细描述。

[0091] 视频编码器(603)可根据诸如ITU-T H.265建议书的预定视频编码技术或标准执行编码操作。在操作中,视频编码器(603)可执行各种压缩操作,包括利用输入视频序列中的时间和空间冗余的预测性编码操作。因此,已编码视频数据可符合所使用的视频编码技术或标准指定的语法。

[0092] 在一些示例性实施例中,传输器(640)可以在传输已编码视频时传输附加数据。源编码器(630)可包括此类数据作为已编码视频序列的一部分。附加数据可包括时间/空间/SNR增强层、诸如冗余图片和切片的其它形式的冗余数据、SEI消息、VUI参数集片段等。

[0093] 所采集的视频可作为呈时间序列的多个源图片(视频图片)。帧内图片预测(通常简化为帧内预测)利用给定图片中的空间相关性,而帧间图片预测则利用图片之间的时间或其它相关性。例如,可以将正在编码/解码的特定图片分成块,正在编码/解码的特定图片

称为当前图片。在当前图片中的块类似于视频中先前已编码且仍被缓冲的参考图片中的参考块时,可通过称为运动矢量的矢量对当前图片中的块进行编码。运动矢量指向参考图片中的参考块,且在使用多个参考图片的情况下,运动矢量可具有识别参考图片的第三个维度。

[0094] 在一些示例性实施例中,双向预测技术可用于帧间图片预测。根据这种双向预测技术,使用两个参考图片,例如按解码次序在视频中的当前图片之前(但是按显示次序可能分别是过去或将来)的第一参考图片和第二参考图片。可通过指向第一参考图片中的第一参考块的第一运动矢量和指向第二参考图片中的第二参考块的第二运动矢量来对当前图片中的块进行编码。可通过第一参考块和第二参考块的组合来联合地预测该块。

[0095] 此外,合并模式技术可用于帧间图片预测以提高编码效率。

[0096] 根据本公开的一些示例性实施例,诸如帧间图片预测和帧内图片预测的预测以块为单位来执行。例如,将视频图片序列中的图片划分成编码树单元(CTU)以用于压缩,图片中的CTU可具有相同大小,例如 64×64 像素、 32×32 像素或 16×16 像素。通常,CTU可包括三个平行的编码树块(CTB):一个亮度CTB和两个色度CTB。可以将每个CTU递归地以四叉树拆分成一个或多个编码单元(CU)。例如,可以将 64×64 像素的CTU拆分成一个 64×64 像素的CU,或4个 32×32 像素的CU。一个或多个 32×32 块中的每一个块可进一步拆分成4个 16×16 像素的CU。在一些示例性实施例中,可以在编码期间分析每个CU以确定多种预测类型之中用于CU的预测类型,例如帧间预测类型或帧内预测类型。根据时间和/或空间可预测性,可以将CU拆分成一个或多个预测单元(PU)。通常,每个PU包括亮度预测块(PB)和两个色度PB。在一个实施例中,编码(编码/解码)中的预测操作以预测块为单位来执行。可以以各种空间图案执行将CU拆分成PU(或不同颜色通道的PB)。例如,亮度或色度PB可包括针对样本的值(例如,亮度值)的矩阵,所述的样本例如是 8×8 像素、 16×16 像素、 8×16 像素、 16×8 像素等。

[0097] 图7示出了根据本公开的另一示例性实施例的视频编码器(703)的图。视频编码器(703)配置成接收视频图片序列中的当前视频图片内的样本值的处理块(例如,预测块),且将处理块编码到作为已编码视频序列的一部分的已编码图片中。在示例中,视频编码器(703)可用于代替图4的示例中的视频编码器(403)。

[0098] 例如,视频编码器(703)接收用于处理块的样本值的矩阵,该处理块例如是 8×8 样本的预测块等。然后,视频编码器(703)使用例如率失真优化(RDO)来确定是否使用帧内模式、帧间模式或双向预测模式来最佳地对处理块进行编码。当确定在帧内模式中对处理块进行编码时,视频编码器(703)可使用帧内预测技术以将处理块编码到已编码图片中;以及当确定在帧间模式或双向预测模式中对处理块进行编码时,视频编码器(703)可分别使用帧间预测或双向预测技术以将处理块编码到已编码图片中。在一些示例性实施例中,合并模式可用作帧间图片预测的子模式,其中,在不借助预测器外部的已编码运动矢量分量的情况下,从一个或多个运动矢量预测器导出运动矢量。在一些其它示例性实施例中,可存在适用于主题块的运动矢量分量。因此,视频编码器(703)可包括未在图7中明确示出的组件,例如用于确定处理块的预测模式的模式决策模块。

[0099] 在图7的示例中,视频编码器(703)包括如图7中的示例性布置所示的耦接在一起的帧间编码器(730)、帧内编码器(722)、残差计算器(723)、开关(726)、残差编码器(724)、

通用控制器 (721) 和熵编码器 (725)。

[0100] 帧间编码器 (730) 配置成接收当前块 (例如, 处理块) 的样本, 将该块与参考图片中的一个或多个参考块 (例如, 按显示次序是先前图片和后来图片中的块) 进行比较, 生成帧间预测信息 (例如, 根据帧间编码技术的冗余信息描述、运动矢量、合并模式信息), 以及基于帧间预测信息使用任何合适的技术来计算帧间预测结果 (例如, 已预测块)。在一些示例中, 参考图片是基于已编码视频信息, 使用嵌入在图6的示例性编码器620中的解码单元633 (如图7的残差解码器728所示, 如下文进一步详细描述) 来解码的已解码参考图片。

[0101] 帧内编码器 (722) 配置成接收当前块 (例如, 处理块) 的样本, 将该块与同一图片中已编码的块进行比较, 以及在变换之后生成量化系数, 且在一些情况下还生成帧内预测信息 (例如, 根据一个或多个帧内编码技术的帧内预测方向信息)。帧内编码器 (722) 可基于帧内预测信息和同一图片中的参考块计算帧内预测结果 (例如, 已预测块)。

[0102] 通用控制器 (721) 可配置成确定通用控制数据, 且基于该通用控制数据控制视频编码器 (703) 的其它组件。在一个示例中, 通用控制器 (721) 确定块的预测模式, 且基于该预测模式将控制信号提供给开关 (726)。例如, 当该预测模式是帧内模式时, 通用控制器 (721) 控制开关 (726) 以选择供残差计算器 (723) 使用的帧内模式结果, 且控制熵编码器 (725) 以选择帧内预测信息并将帧内预测信息包括在码流中; 以及当该块的预测模式是帧间模式时, 通用控制器 (721) 控制开关 (726) 以选择供残差计算器 (723) 使用的帧间预测结果, 且控制熵编码器 (725) 以选择帧间预测信息并将帧间预测信息包括在码流中。

[0103] 残差计算器 (723) 可配置成计算所接收的块与从帧内编码器 (722) 或帧间编码器 (730) 选择的块的预测结果之间的差 (残差数据)。残差编码器 (724) 可配置成对残差数据进行编码以生成变换系数。例如, 残差编码器 (724) 可配置成将残差数据从空间域变换到频域, 以生成变换系数。变换系数随后经受量化处理以得到量化变换系数。在各种示例性实施例中, 视频编码器 (703) 还包括残差解码器 (728)。残差解码器 (728) 配置成执行逆变换, 且生成已解码残差数据。已解码残差数据可适当地由帧内编码器 (722) 和帧间编码器 (730) 使用。例如, 帧间编码器 (730) 可基于已解码残差数据和帧间预测信息生成已解码块, 且帧内编码器 (722) 可基于已解码残差数据和帧内预测信息生成已解码块。适当地处理已解码块以生成已解码图片, 且已解码图片可以在存储器电路 (未示出) 中缓冲并用作参考图片。

[0104] 熵编码器 (725) 可配置成将码流格式化以包括已编码块, 以及执行熵编码。熵编码器 (725) 配置成在码流中包括各种信息。例如, 熵编码器 (725) 可配置成将通用控制数据、所选预测信息 (例如, 帧内预测信息或帧间预测信息)、残差信息和其它合适的信息包括在码流中。当在帧间模式或双向预测模式的合并子模式中对块进行编码时, 可以不存在残差信息。

[0105] 图8示出了根据本公开的另一实施例的示例性视频解码器 (810) 的图。视频解码器 (810) 配置成接收作为已编码视频序列的一部分的已编码图片, 且对已编码图片进行解码以生成已重建图片。在一个示例中, 视频解码器 (810) 可用于代替图4的示例中的视频解码器 (410)。

[0106] 在图8的示例中, 视频解码器 (810) 包括如图8中的示例性布置所示的耦接在一起的熵解码器 (871)、帧间解码器 (880)、残差解码器 (873)、重建模块 (874) 和帧内解码器 (872)。

[0107] 熵解码器 (871) 可配置成根据已编码图片来重建某些符号, 这些符号表示构成已编码图片的语法元素。此类符号可包括例如对块进行编码的模式 (例如, 帧内模式、帧间模式、双向预测模式、合并子模式或另一子模式)、可识别供帧内解码器 (872) 或帧间解码器 (880) 使用以进行预测的某些样本或元数据的预测信息 (例如, 帧内预测信息或帧间预测信息)、呈例如量化变换系数形式的残差信息等。在一个示例中, 当预测模式是帧间或双向预测模式时, 将帧间预测信息提供给帧间解码器 (880); 以及当预测类型是帧内预测类型时, 将帧内预测信息提供给帧内解码器 (872)。残差信息可经受逆量化并提供给残差解码器 (873)。

[0108] 帧间解码器 (880) 可配置成接收帧间预测信息, 且基于该帧间预测信息生成帧间预测结果。

[0109] 帧内解码器 (872) 可配置成接收帧内预测信息, 且基于该帧内预测信息生成预测结果。

[0110] 残差解码器 (873) 可配置成执行逆量化以提取解量化的变换系数, 且处理该解量化的变换系数, 以将残差从频域变换到空间域。残差解码器 (873) 还可利用某些控制信息 (用以包括量化器参数 (QP)), 这些控制信息可由熵解码器 (871) 提供 (未描绘数据路径, 因为这仅仅是低数据量控制信息)。

[0111] 重建模块 (874) 可配置成在空间域中组合由残差解码器 (873) 输出的残差与预测结果 (可由帧间预测模块或帧内预测模块输出, 视情况而定) 以形成已重建块, 已重建块形成已重建图片的一部分, 而已重建图片是已重建视频的一部分。应注意, 还可执行诸如去块操作等其它合适的操作来改善视觉质量。

[0112] 应注意, 可使用任何合适的技术来实现视频编码器 (403)、视频编码器 (603) 和视频编码器 (703) 以及视频解码器 (410)、视频解码器 (510) 和视频解码器 (810)。在一些示例性实施例中, 可使用一个或多个集成电路来实现视频编码器 (403)、视频编码器 (603) 和视频编码器 (703) 以及视频解码器 (410)、视频解码器 (510) 和视频解码器 (810)。在另一实施例中, 可使用执行软件指令的一个或多个处理器来实现视频编码器 (403)、视频编码器 (603) 和视频编码器 (603) 以及视频解码器 (410)、视频解码器 (510) 和视频解码器 (810)。

[0113] 转到编码块划分, 在一些示例性实现方式中, 可应用预定图案。如图9所示, 可使用从第一预定层级 (例如, 64×64 块层级) 开始下探到第二预定层级 (例如, 4×4 层级) 的示例性4路划分树。例如, 基本块可受制于由902、904、906和908指示的四个划分选项, 其中, 可允许标记为R的划分 (partitions) 用于递归划分, 原因是如图9所指示的同一个划分树可以以小规模重复, 直到最小层级 (例如, 4×4 层级)。在一些实现方式中, 可以对图9的划分方案应用附加限制。在图9的实现方式中, 可允许矩形划分 (例如, $1:2/2:1$ 矩形划分), 但是可以不允许矩形划分是递归划分, 而是允许正方形划分是递归的。如果需要的话, 遵循图9递归地划分, 则生成最终编码块集合。这种方案可适用于一个或多个颜色通道。

[0114] 图10示出了另一个示例性预定划分图案, 该预定划分图案允许递归划分以形成划分树。如图10所示, 可预先定义示例性10路划分结构或图案。根块可以从预定层级开始 (例如, 从 128×128 层级或 64×64 层级开始)。图10的示例性划分结构包括各种 $2:1/1:2$ 和 $4:1/1:4$ 矩形划分。在图10的第二行中具有由1002、1004、1006和1008指示的3子划分的划分类型可称为“T型”划分。“T型”划分1002、1004、1006和1008可称为左侧T型、顶部T型、右侧T型和

底部T型。在一些实现方式中,不允许进一步细分图10的矩形划分。可进一步定义编码树深度,以指示从根节点或根块起的拆分深度。例如,对于根节点或根块(例如,对于 128×128 块),编码树深度可设置为0,在遵循图10进一步对根块进行一次拆分之后,编码树深度增加1。在一些实现方式中,仅可允许1010中的所有正方形划分遵循图10的图案递归地划分,成为划分树的下一层级。换句话说,可以不允许递归划分用于具有图案1002、1004、1006和1006的正方形划分。如果需要的话,遵循图10递归地划分,则生成最终编码块集合。这种方案可适用于一个或多个颜色通道。

[0115] 在遵循上述任意划分过程或其它过程分割或划分基本块之后,再次可获得最终划分集合或编码块。这些划分中的每个划分可处于各种划分层级之一。每个划分可称为编码块(CB)。对于上述的各种示例性划分实现方式,每个产生的CB可具有任何允许的大小和划分层级。划分称为编码块,原因是划分可形成单元,可以对单元做出一些基本的编码/解码决策,且可优化、确定编码/解码参数,以及在已编码视频码流中用信号表示编码/解码参数。最终划分中的最高层级表示编码块划分树的深度。编码块可以是亮度编码块或色度编码块。

[0116] 在一些其它示例性实现方式中,四叉树结构可用于将基本亮度块和色度块递归地拆分成编码单元。这种拆分结构可称为编码树单元(CTU),通过使用四叉树结构将CTU拆分成编码单元(CU),以使划分适应基本CTU的各种局部特性。在这样的实现方式中,可以在图片边界处执行隐式四叉树拆分,使得块将保持四叉树拆分,直到大小适合图片边界。使用术语CU以共同地指代亮度编码块和色度编码块(CB)的单元。

[0117] 在一些实现方式中,可进一步划分CB。例如,CB可进一步划分成多个预测块(PB),以便在编码和解码过程期间进行帧内预测或帧间预测。换句话说,CB可进一步分割成不同的子划分,在这些子划分中可做出独立的预测决策/配置。与之并行地,CB可进一步划分成多个变换块(TB),以便描述执行视频数据的变换或逆变换所使用的层级。CB划分成PB和TB的划分方案可以相同,或者可以不同。例如,每个划分方案可基于例如视频数据的各种特性,使用其自己的过程来执行。在一些示例性实现方式中,PB划分方案和TB划分方案可以不相干。在一些其它示例性实现方式中,PB划分方案和TB划分方案以及边界可相互关联。在一些实现方式中,例如,可以在PB划分之后对TB进行划分,尤其是,在遵循编码块的划分确定每个PB之后,则每个PB可进一步划分成一个或多个TB。例如,在一些实现方式中,PB可拆分成一个TB、两个TB、四个TB或其它数量的TB。

[0118] 在一些实现方式中,为了将基本块划分成编码块并进一步划分成预测块和/或变换块,可以不同地处理亮度通道和色度通道。例如,在一些实现方式中,对于亮度通道,可允许将编码块划分成预测块和/或变换块,而对于色度通道,可以不允许如此将编码块划分成预测块和/或变换块。因此,在这样的实现方式中,仅可在编码块层级执行亮度块的变换和/或预测。又例如,亮度通道和色度通道的最小变换块大小可以不同,例如,与色度通道相比,可允许将亮度通道的编码块划分成更小的变换块和/或预测块。对于又一个示例,在亮度通道和色度通道之间,将编码块划分成变换块和/或预测块的最大深度可以不同,例如,与色度通道相比,可允许将亮度通道的编码块划分成更深的变换块和/或预测块。对于特定示例,亮度编码块可划分成多个大小的变换块,这些变换块可由下探多达2个层级的递归划分表示,且可允许诸如正方形、 $2:1/1:2$ 和 $4:1/1:4$ 之类的变换块形状和从 4×4 到 64×64 的变

换块大小。然而,对于色度块,仅可允许对亮度块指定的最大可能的变换块。

[0119] 在一些示例性实现方式中,为了将编码块划分成PB,PB划分的深度、形状和/或其它特性可取决于是否对PB进行帧内编码或帧间编码。

[0120] 可以以各种示例性方案实现将编码块(或预测块)划分成变换块,示例性方案包括但不限于递归或非递归的四叉树拆分和预定图案拆分,且额外考虑位于编码块或预测块的边界处的变换块。通常,产生的变换块可处于不同的拆分层级,可以不具有相同的大小,以及其形状可以不需要为正方形(例如,变换块可以是具有一些允许的大小和宽高比的矩形)。

[0121] 在一些示例性实现方式中,可使用编码划分树方案或结构。用于亮度通道和色度通道的编码划分树方案可以不需要相同。换句话说,亮度通道和色度通道可具有不同的编码树结构。此外,亮度通道和色度通道是否使用相同或不同的编码划分树结构,以及要使用的实际编码划分树结构,可取决于被编码的切片是P切片、B切片还是I切片。例如,对于I切片,色度通道和亮度通道可具有不同的编码划分树结构或编码划分树结构模式,而对于P切片或B切片,亮度通道和色度通道可共享相同的编码划分树方案。当应用不同的编码划分树结构或模式时,可通过一个编码划分树结构将亮度通道划分成CB,且可通过另一编码划分树结构将色度通道划分成色度CB。

[0122] 在下文中描述编码块和变换块划分的特定示例性实现方式。在这种示例性实现方式中,可使用上述递归四叉树拆分将基本编码块拆分成编码块。在每个层级,是否应继续特定划分的进一步四叉树拆分,可通过本地视频数据特性来确定。产生的CB可处于各种大小的各种四叉树拆分层级。可以在CB层级(或CU层级,对于所有三个颜色通道来说)做出是否使用帧间图片(时间)预测或帧内图片(空间)预测对图片区域进行编码的决策。根据PB拆分类型,每个CB可进一步拆分成一个PB、两个PB、四个PB或其它数量的PB。在一个PB内,可应用相同的预测过程,并基于PB将相关信息发送到解码器。在通过应用基于PB拆分类型的预测过程获得残差块之后,可根据与CB的编码树类似的另一四叉树结构来将CB划分成TB。在该特定实现方式中,CB或TB可限于正方形,但不必限于正方形。此外,在该特定示例中,对于帧间预测,PB可以是正方形或矩形,而对于帧内预测,PB只能是正方形。例如,编码块可进一步拆分成四个正方形TB。每个TB可进一步(使用四叉树拆分)递归地拆分成更小的TB,更小的TB称为残差四叉树(RQT)。

[0123] 在下文中描述将基本编码块划分成CB和其它PB和/或TB的另一特定示例。例如,不是使用如图10所示的多划分单元类型,而是可使用具有嵌套多类型树的四叉树,其使用二元和三元拆分分段结构。CB、PB和TB概念的分离(即,将CB划分成PB和/或TB,以及将PB划分成TB)可以放弃,除非当需要CB具有对于最大变换长度来说太大的大小(在这种情况下,可能需要进一步拆分这种CB)时。该示例性拆分方案可设计为支持CB划分形状的更大灵活性,使得预测和变换可以在CB层级执行,而不需要进一步划分。在这种编码树结构中,CB可以是正方形或矩形。具体而言,首先可通过四叉树结构划分编码树块(CTB)。然后,可通过多类型树结构进一步划分四叉树叶节点。在图11中示出了多类型树结构的示例。具体而言,图11的示例性多类型树架构包括四种拆分类型,它们称为垂直二进制拆分(拆分_BT_VER)(1102)、水平二进制拆分(拆分_BT_HOR)(1104)、垂直三元拆分(拆分_TT_VER)(1106)和水平三元拆分(拆分_TT_HOR)(1108)。然后,CB对应于多类型树的叶子。在该示例性实现方式中,除非CB

对于最大变换长度来说太大,否则该分段用于预测和变换处理,而不需要任何进一步划分。这意味着在大多数情况下,CB、PB和TB在具有嵌套多类型树编码块结构的四叉树中具有相同的块大小。当支持的最大变换长度小于CB颜色分量的宽度或高度时,会出现例外情况。

[0124] 在图12中示出了一个CTB的块划分的、具有嵌套多类型树编码块结构的四叉树的一个示例。更详细地说,图12示出了CTB 1200是四叉树,拆分成四个正方形划分1202、1204、1206和1208。对于每个四叉树拆分划分,做出进一步使用图11的多类型树结构进行拆分的决策。在图12的示例中,没有进一步拆分划分1204。划分1202和1208均采用另一四叉树拆分。对于划分1202,第二层级四叉树拆分的左上划分、右上划分、左下划分和右下划分分别采用第三层级四叉树拆分、图11的1104、不拆分和图11的1108。划分1208采用另一四叉树拆分,第二层级四叉树拆分的左上划分、右上划分、左下划分和右下划分分别采用图11的第三层级拆分1106、不拆分、不拆分和图11的1104。1208的第三层级左上划分的两个子划分进一步根据1104和1108来拆分。划分1206采用遵循图11的1102的第二层级拆分图案,拆分成两个划分,这两个划分进一步根据图11的1108和1102在第三层级下进行拆分。根据图11的1104,第四层级拆分进一步应用于其中一个划分。

[0125] 对于上述的特定示例,最大亮度变换大小可以是 64×64 ,支持的最大色度变换大小可以不同于亮度,例如处于 32×32 。当亮度编码块或色度编码块的宽度或高度大于最大变换宽度或高度时,可沿着水平方向和/或垂直方向自动地拆分亮度编码块或色度编码块,以满足沿着该方向的变换大小限制。

[0126] 在特定示例中,为了将基本编码块划分成上述CB,编码树方案可支持亮度和色度具有不同的块树结构的能力。例如,对于P切片和B切片,一个CTU中的亮度CTB和色度CTB可共享相同的编码树结构。例如,对于I切片,亮度和色度可具有不同的编码块树结构。当应用不同的块树模式时,可通过一个编码树结构将亮度CTB划分成亮度CB,以及通过另一编码树结构将色度CTB划分成色度CB。这意味着I切片中的CU可由亮度分量的编码块或两个色度分量的编码块组成,以及P切片或B切片中的CU始终由所有三个颜色分量的编码块组成,除非视频是单色的。

[0127] 在下文中进一步详细描述将编码块或预测块划分成变换块以及变换块的编码顺序的示例性实现方式。在一些示例性实现方式中,变换划分可支持诸如1:1(正方形)、1:2/2:1和1:4/4:1之类的多种形状的变换块,其中变换块的大小的范围例如从 4×4 到 64×64 。在一些实现方式中,如果编码块小于或等于 64×64 ,则变换块划分仅可应用于亮度分量,使得对于色度块,变换块大小与编码块大小相同。否则,如果编码块宽度或高度大于64,则亮度编码块和色度编码块分别可隐式地拆分成多个 $\min(W, 64) \times \min(H, 64)$ 变换块和 $\min(W, 32) \times \min(H, 32)$ 变换块。

[0128] 在一些示例性实现方式中,对于帧内编码块和帧间编码块,编码块可进一步划分成多个变换块,其中划分深度可达预定数量的层级(例如,2个层级)。变换块划分深度和大小可以相关。在下面的表1中示出了从当前深度的变换大小到下一深度的变换大小的示例性映射。

[0129] 表1:变换划分大小设置

当前深度的 变换大小	下一深度的 变换大小
TX_4X4	TX_4X4
TX_8X8	TX_4X4
TX_16X16	TX_8X8
TX_32X32	TX_16X16
TX_64X64	TX_32X32
TX_4X8	TX_4X4
TX_8X4	TX_4X4
TX_8X16	TX_8X8
TX_16X8	TX_8X8
TX_16X32	TX_16X16
TX_32X16	TX_16X16
TX_32X64	TX_32X32
TX_64X32	TX_32X32
TX_4X16	TX_4X8
TX_16X4	TX_8X4
TX_8X32	TX_8X16
TX_32X8	TX_16X8
TX_16X64	TX_16X32
TX_64X16	TX_32X16

[0130] 根据表1的示例性映射,对于1:1正方形块,下一层级变换拆分可创建四个1:1正方形子变换块。变换拆分可例如在 4×4 处停止。因此,当前深度的变换大小 4×4 对应于下一个深度的相同大小 4×4 。在表1的示例中,对于1:2/2:1非正方形块,下一层级变换拆分将创建两个1:1正方形子变换块,而对于1:4/4:1非正方形块,下一层级变换拆分将创建两个1:2/2:1子变换块。

[0131] 在一些示例性实现方式中,对于帧内编码块的亮度分量,可应用附加限制。例如,对于每个变换划分层级,所有子变换块可被限制为大小相等。例如,对于 32×16 编码块,层级1变换拆分会创建两个 16×16 子变换块,层级2变换拆分会创建八个 8×8 子变换块。换句话说,第二层级拆分必须应用于所有第一层级子块,以保持变换单元的大小相等。在图13中示出了遵循表1的帧内编码正方形块的变换块划分的示例,以及由箭头指示的编码顺序。具体而言,1302示出了正方形编码块。在1304中示出了根据表1拆分成4个大小相等的变换块的第一层级拆分,其中编码顺序由箭头指示。在1306中示出了根据表1将所有第一层级的大小相等的块拆分成16个大小相等的变换块的第二层级拆分,其中编码顺序由箭头指示。

[0132] 在一些示例性实现方式中,对于帧间编码块的亮度分量,可以不应用帧内编码的上述限制。例如,在第一层级变换拆分之后,任何一个子变换块可进一步独立地再拆分一个层级。因此,产生的变换块可以大小相同,或者可以大小不同。在图14中示出将帧间编码块拆分成变换块的示例,以及变换块的编码顺序。在图14的示例中,根据表1,帧间编码块1402

拆分成处于两个层级的变换块。在第一层级,帧间编码块拆分成四个大小相等的变换块。然后,四个变换块中只有一个变换块(而非全部变换块)进一步拆分成四个子变换块,从而产生总共7个变换块,这7个变换块具有两种不同的大小,如1404所示。这7个变换块的示例性编码顺序如图14的1404中的箭头所示。

[0134] 在一些示例性实现方式中,对于色度分量,可以对变换块应用一些附加限制。例如,对于色度分量,变换块大小可以与编码块大小一样大,但是不可小于预定大小,例如 8×8 。

[0135] 在一些其它示例性实现方式中,对于宽度(W)或高度(H)大于64的编码块,亮度编码块和色度编码块分别可隐式地拆分成 $\min(W, 64) \times \min(H, 64)$ 变换单元和 $\min(W, 32) \times \min(H, 32)$ 变换单元的倍数个块。

[0136] 图15进一步示出了将编码块或预测块划分成变换块的另一示例性替代方案。如图15所示,作为使用递归变换划分的替代,可根据编码块的变换类型将预定的一组划分类型应用于编码块。在图15所示的特定示例中,可应用6种示例性划分类型之一,以将编码块拆分成各种数量的变换块。这种方案可适用于编码块或预测块。

[0137] 更详细地说,如图15所示,图15的划分方案对任何给定的变换类型提供多达6种划分类型。在该方案中,可基于例如率失真成本对每个编码块或预测块分配变换类型。在一个示例中,可基于编码块或预测块的变换划分类型来确定分配给编码块或预测块的划分类型。特定划分类型可对应于变换块划分大小和模式(或划分类型),如图15所示的4种划分类型所示。可预先定义各种变换类型和各种划分类型之间的对应关系。在下文中示出了示例性对应关系,其中大写字母标签指示可基于率失真成本分配给编码块或预测块的变换类型:

[0138] • PARTITION_NONE(划分_无):分配等于块大小的变换大小。

[0139] • PARTITION_SPLIT(划分_拆分):分配如下变换大小,该变换大小是块大小的宽度的1/2和块大小的高度的1/2。

[0140] • PARTITION_HORZ(划分_HORZ):分配如下变换大小,该变换大小与块大小具有相同宽度且是块大小的高度的1/2。

[0141] • PARTITION_VERT(划分_VERT):分配如下变换大小,该变换大小是块大小的宽度的1/2且与块大小具有相同高度。

[0142] • PARTITION_HORZ4(划分_HORZ4):分配如下变换大小,该变换大小与块大小具有相同宽度且是块大小的高度的1/2。

[0143] • PARTITION_VERT4(划分_VERT4):分配如下变换大小,该变换大小是块大小的宽度的1/4且与块大小具有相同高度。

[0144] 在上述示例中,对于划分的变换块,如图15所示的划分类型均包含统一变换大小。这仅仅是示例,而非限制。在一些其它示例性实现方式中,在特定划分类型(或模式)中,混合变换块大小可用于划分的变换块。

[0145] 返回到帧内预测,在一些示例性实现方式中,对编码块或预测块中的样本的预测可基于一组参考行中的一个参考行。换句话说,不是始终使用最近相邻行(例如,预测块的正上方相邻行或正左侧相邻行,如上面的图1所示的),而是可提供多个参考行作为帧内预测的选择选项。这种帧内预测实现方式可称为多参考行选择(MRLS)。在这些实现方式中,编

码器决定并用信号表示多个参考行中的哪个参考行用于生成帧内预测。在解码器侧,在解析参考行索引之后,可通过根据帧内预测模式(例如定向帧内预测模式、非定向帧内预测模式和其它帧内预测模式)查找指定的参考行来识别已重建参考样本,由此生成当前帧内预测块的帧内预测。在一些实现方式中,参考行索引可以在编码块层级用信号表示,且只能选择多个参考行中的一个参考行来用于一个编码块的帧内预测。在一些示例中,可以一起选择多于一个参考行来进行帧内预测。例如,可(使用或不使用权重)组合、求平均、插值或以任何其它方式处理多于一个参考行,以生成预测。在一些示例性实现方式中,MRLS仅可适用于亮度分量,而不适用于色度分量。

[0146] 在图16中,描绘了4参考行MRLS的示例。如图16的示例所示,可基于4个水平参考行1604、1606、1608和1610中的一个水平参考行以及4个垂直参考行1612、1614、1616和1618中的一个垂直参考行来预测帧内编码块1602。在这些参考行中,1610和1618是直接相邻的参考行。参考行可根据其与编码块相距的距离来进行索引。例如,参考行1610和1618可称为零参考行,而其它参考行可称为非零参考行。具体而言,参考行1608和1616可称为第一参考行;参考行1606和1614可称为第二参考行;以及参考行1604和1612可称为第三参考行。

[0147] 可存在与实现多参考行选择(MRLS)方案相关联的一些问题/难题。在没有多参考行选择的一些实现方式中,只有最近(或邻近)的顶部(或上方)和/或左侧参考行是相关的,且只有来自最近(或邻近)的顶部(或上方)和/或者左侧参考行的样本才需要缓冲/存储,以用于当前块的帧内预测。当MRLS应用于帧内编码块时,4个顶部(或上方)参考行和4个左侧参考行可用于帧内预测。因此,用于存储相邻参考样本的缓冲区需要增加3倍。此外,对于超级块,缓冲区的长度可与位于超级块的边界处的参考行的长度一样长,在一些情况下,可等于硬件解码器中的图片宽度,因此,缓冲区大小增加3倍对硬件解码器来说是个大负担。

[0148] 本公开描述了用于改进视频编码和/或解码中帧内预测的多参考行选择方案的低存储设计和/或信令的各种实施例,从而解决上文讨论的问题/难题中的至少一个问题/难题。

[0149] 在各种实施例中,参考图17,用于视频解码中的多参考行帧内预测的方法1700。方法1700可包括以下步骤中的一部分或全部步骤:步骤1710,通过包括存储指令的存储器和与存储器通信的处理器,接收当前块的已编码视频码流;步骤1720,通过该设备从已编码视频码流中提取参数,参数指示用于当前块中帧内预测的一个非相邻参考行;步骤1730,通过该设备对当前块进行划分,以得到多个子块;和/或步骤1740,响应于多个子块中的一子块位于当前块的边界处,通过该设备使用顶部相邻参考行作为该子块的所有顶部非相邻参考行的值。

[0150] 在一些实现方式中,当前块可称为块,通过划分当前块而获得的多个子块可称为多个编码块。在一些其它实现方式中,步骤1720可包括:通过设备从已编码视频码流中提取参数,参数指示用于当前块中帧内预测的参考行。

[0151] 在本公开的各种实施例中,块(例如但不限于编码块、预测块或变换块)的大小可指的是块的宽度或高度。块的宽度或高度可以是以像素为单位的整数。

[0152] 在本公开的各种实施例中,块(例如但不限于编码块、预测块或变换块)的大小可指的是块的面积大小。块的面积大小可以通过块的宽度乘以块的高度计算的、以像素为单位的整数。

[0153] 在本公开的一些不同的实施例中,块(例如但不限于编码块、预测块或变换块)的大小可指的是块的宽度或高度的最小值、块的高度或宽度的最小值、或者块的宽高比。块的宽度比可通过块的宽度除以高度来计算,或者可通过块的高度除以宽度来计算。

[0154] 在本公开中,参考行索引指示多个参考行之中的参考行。在各种实施例中,块的参考行索引为0,则可指示块的相邻参考行,相邻参考行也是块的最近参考行。例如,参考图16中的块(1602),顶部参考行(1610)是块(1602)的顶部相邻参考行,也是块的顶部最近参考行;左侧参考行(1618)是块(1602)的左侧相邻参考行,也是块的左侧最近参考行。块的参考行索引大于0,则指示块的非相邻参考行,非相邻参考行也是块的非最近参考行。例如,参考图16中的块(1602),参考行索引为1,则可指示顶部参考行(1608)和/或左侧参考行(1616);参考行索引为2,则可指示顶部参考行(1606)和/或左侧参考行(1614);和/或参考行索引为3,则可指示顶部参考行(1604)和/或左侧参考行(1612)。

[0155] 参考步骤1710,设备可以是图5中的电子设备(530)或图8中的视频解码器(810)。在一些实现方式中,设备可以是图6中的编码器(620)中的解码器(633)。在其它实现方式中,设备可以是图5中的电子设备(530)的一部分,图8中的视频解码器(810)的一部分,或者图6中的编码器(620)中的解码器(633)的一部分。已编码视频码流可以是图8中的已编码视频序列,或者图6或图7中的中间编码数据。

[0156] 在一些实现方式中,块可称为超级块。超级块可指的是最大编码块,例如但不限于编码树块(CTB)和/或最大编码单元(LCU)。在一些其它实现方式中,超级块可指的是预定块大小,例如但不限于 32×32 、 64×64 、 128×128 和/或 256×256 。

[0157] 参考步骤1720,设备可以从已编码视频码流中提取参数,参数可用于MRLS并指示块中用于帧内预测的参考行。在一些实现方式中,由参数指示的参考行可以是N个参考行之一,其中N是大于1的整数。在一些其它实现方式中,由参数指示的参考行可以是N个非相邻参考行之一。

[0158] 在一些实现方式中,方法1700可进一步包括:通过设备使用块的左侧相邻参考行。在一些其它实现方式中,左侧相邻参考行可以是块的N个左侧参考行之一。

[0159] 参考步骤1730,设备可以对块进行划分,以得到多个编码块。在一些实现方式中,设备可以对块进行划分,以得到编码块划分树。编码块划分树可包括多个编码块。

[0160] 参考步骤1740,对于来自多个编码块中的一编码块,当该编码块位于块的边界处时,设备可使用顶部相邻参考行作为该编码块的所有顶部非相邻参考行的值。在一些实现方式中,块(例如,超级块)的边界(或多个边界)可指的是仅块的顶部边界,或者仅块的左侧边界,或者块的左侧边界和顶部边界。

[0161] 在各种实施例中,为了减小存储来自多个参考行的样本的存储器大小,当当前帧内编码块位于超级块的边界处时,N个左侧参考行(列)中的样本可用于帧内预测,且最近(或邻近)上方(或顶部)参考行中的样本可用于帧内预测。

[0162] N是大于1的正整数,例如2、3或4。例如,当 $N=3$ 时,对于位于超级块的边界处的编码块,三个左侧参考行和相邻的顶部参考行可用于帧内预测。在一些实现方式中,三个左侧参考行可以是块的三个最近左侧参考行,例如图16中的参考行(1618,1616和1614)。

[0163] 在一些实现方式中,参考行索引可以用信号表示并编码到码流中,无论当前帧内编码块是否位于超级块的边界处。

[0164] 再次参考步骤1740,使用顶部相邻参考行作为编码块的所有顶部非相邻参考行的值的步骤可包括:将来自顶部相邻参考行的样本复制到所有其它顶部非相邻参考行,使得显著地减小存储来自所有其它顶部非相邻参考行的样本的存储器大小。

[0165] 在一些实现方式中,如果当前帧内编码块位于块(例如,超级块)的边界处,且参考行索引指示非零(或非相邻)参考行用于当前编码块的帧内预测,则通过从最近顶部(或上方)参考行进行复制来导出顶部(或上方)非零参考行中的样本。

[0166] 作为一个示例,在图18中,编码块(还称为已编码块或代码块)(1802)位于块(例如,超级块)的顶部边界(1830)和左侧边界(1840)处。如图18所示,超级块的顶部边界(1830)和左侧边界(1840)可由粗线指示。编码块(1802)的最近上方参考行(1810)中的左上样本(A-4、A-3、A-2和/或A-1)和顶部样本(A+0, ..., A+N)复制到上方非零参考行(1808, 1806和/或1804)。在一些实现方式中,可允许多个左侧参考行(1818, 1816, 1814和/或1812)分别存储各自的样本。

[0167] 在上述各种实现方式中,以顶部相邻参考行为例,类似的方案可适用于左侧相邻参考行,其中,为了改进低存储设计,左侧非零参考行中的样本可从最近(或相邻)左侧参考行导出,例如通过复制左侧相邻参考行来导出。

[0168] 在各种实施例中,可选地,方法1700可进一步包括:通过设备对该编码块进行划分,以得到多个变换块。在一些实现方式中,设备可以对该编码块进行划分,以得到包括多个变换块的变换块划分树。在一些实现方式中,可选地,方法1700可包括:响应于多个变换块中的第一变换块位于该编码块的顶部边界处,通过设备使用顶部相邻参考行作为第一变换块的所有顶部非相邻参考行的值;和/或响应于多个变换块中的第二变换块没有位于该编码块的顶部边界处,通过设备针对所述第二变换块使用由第二变换块的参数指示的参考行。在一些实现方式中,由参数指示的参考行可以是第二变换块的N个顶部参考行之一。

[0169] 在一些实现方式中,当前编码块可拆分成多于一个变换块(TB)和/或多于一个变换单元(TU)。当当前编码块位于超级块的边界处且参考行索引指示非零参考行用于当前编码块时,只有最近上方参考行中的样本可用于对位于超级块的边界处的TU进行帧内预测;当当前编码块位于超级块的边界处且参考行索引指示非零参考行用于当前编码块时,上方非零参考行中的样本仍然可用于不位于超级块的边界处的TU。

[0170] 作为一个示例,参考图19,粗实线(1930)指示超级块的顶部边界。多个变换单元(TU1, TU2, TU3和/或TU4)(1904, 1906, 1908和1910)位于一个编码块中。TU1(1904)和TU2(1906)位于超级块的边界(由水平粗实线(1930)指示)处,而TU3(1908)和TU4(1910)不位于超级块的顶部边界处。因此,只有上方最近参考行中的样本可用于TU1和TU2,而非零参考行中的样本仍然可用于TU3和TU4。类似地,在一些其它实现方式中,当编码块内的变换块位于超级块的左侧边界处时,只有左侧最近参考行中的样本可用于变换块。

[0171] 本公开的实施例可单独地使用或者以任何顺序组合。此外,每种方法(或实施例)、编码器和解码器可通过处理电路(例如,一个或多个处理器或者一个或多个集成电路)来实现。在一个示例中,一个或多个处理器执行存储在非暂时性计算机可读介质中的程序。本公开的实施例可分开地应用于多于一个颜色分量,或者可以一起应用于多于一个颜色分量。

[0172] 上述技术可实现为计算机软件,该计算机软件使用计算机可读指令,且物理地存储在一个或多个计算机可读介质中。例如,图20示出了适于实施所公开的主题的某些实施

例的计算机系统(2600)。

[0173] 可使用任何合适的机器代码或计算机语言对计算机软件进行编码,任何合适的机器代码或计算机语言可经受汇编、编译、链接或类似的机制以创建包括指令的代码,指令可由一个或多个计算机中央处理单元(CPU)、图形处理单元(GPU)等直接执行,或者通过解释、微代码执行等执行。

[0174] 指令可以在各种类型的计算机或其组件上执行,计算机或其组件包括例如个人计算机、平板计算机、服务器、智能电话、游戏设备、物联网设备等。

[0175] 图20所示的计算机系统(2600)的组件本质上是示例性的,并不旨在对实施本公开的实施例的计算机软件的用途或功能的范围提出任何限制。组件的配置也不应解释为具有与计算机系统(2600)的示例性实施例中所示的组件中的任何一个组件或组件的组合相关的任何依赖或要求。

[0176] 计算机系统(2600)可包括某些人机接口输入设备。此类人机接口输入设备可响应于一个或多个人类用户通过例如下述的输入:触觉输入(例如:击键、划动,数据手套移动)、音频输入(例如:语音、拍手)、视觉输入(例如:手势)、嗅觉输入(未描绘)。人机接口设备还可用于捕获不一定与人的意识输入直接相关的某些媒介,例如音频(例如:语音、音乐、环境声音)、图像(例如:扫描图像、从静止图像相机获取的拍摄图像)、视频(例如,二维视频、包括立体视频的三维视频)。

[0177] 人机接口输入设备可包括下述中的一项或多项(每种中仅示出一个):键盘(2601)、鼠标(2602)、触控板(2603)、触摸屏(2610)、数据手套(未示出)、操纵杆(2605)、麦克风(2606)、扫描仪(2607)、相机(2608)。

[0178] 计算机系统(2600)还可包括某些人机接口输出设备。此类人机接口输出设备可通过例如触觉输出、声音、光和气味/味道来刺激一个或多个人类用户的感官。此类人机接口输出设备可包括触觉输出设备(例如,触摸屏(2610)的触觉反馈、数据手套(未示出)或操纵杆(2605),但还可以是不作为输入设备的触觉反馈设备)、音频输出设备(例如:扬声器(2609)、耳机(未描绘))、视觉输出设备(例如,包括CRT屏幕、LCD屏幕、等离子屏幕、OLED屏幕的屏幕(2610),每种屏幕具有或没有触摸屏输入功能,每种屏幕具有或没有触觉反馈功能,其中的一些屏幕能够通过诸如立体图像输出之类的装置、虚拟现实眼镜(未描绘)、全息显示器和烟箱(未描绘)以及打印机(未描绘)来输出二维视觉输出或超过三维的输出。

[0179] 计算机系统(2600)还可包括人类可访问存储设备及其关联介质,例如包括具有CD/DVD等介质(2621)的CD/DVD ROM/RW(2620)的光学介质、指状驱动器(2622)、可拆卸硬盘驱动器或固态驱动器(2623)、诸如磁带和软盘之类的传统磁性介质(未描绘)、诸如安全软件狗之类的基于专用ROM/ASIC/PLD的设备(未描绘)等。

[0180] 本领域技术人员还应该理解,结合当前公开的主题所使用的术语“计算机可读介质”不涵盖传输介质、载波或其它暂时性信号。

[0181] 计算机系统(2600)还可包括通向一个或多个通信网络(2655)的接口(2654)。网络可例如是无线网络、有线网络、光网络。网络可进一步是本地网络、广域网络、城域网络、车辆和工业网络、实时网络、延迟容忍网络等。网络的示例包括诸如以太网之类的局域网、无线LAN、包括GSM、3G、4G、5G、LTE等的蜂窝网络、包括有线电视、卫星电视和地面广播电视的电视有线或无线广域数字网络、包括CAN总线的车辆和工业网络等。某些网络通常需要附接

到某些通用数据端口或外围总线 (2649) 的外部网络接口适配器 (例如, 计算机系统 (2600) 的 USB 端口); 如下所述, 其它网络接口通常通过附接到系统总线而集成到计算机系统 (2600) 的内核中 (例如, 连接到 PC 计算机系统 中的以太网接口或连接到智能手机计算机系统 中的蜂窝网络接口)。计算机系统 (2600) 可使用这些网络中的任何网络与其它实体通信。此类通信可以是仅单向接收的 (例如, 广播电视)、仅单向发送的 (例如, 连接到某些 CANBus 设备的 CANBus) 或双向的, 例如, 使用局域网或广域网数字网络连接到其它计算机系统。如上所述, 可以在那些网络和网络接口中的每一个上使用某些协议和协议栈。

[0182] 上述人机接口设备、人机可访问的存储设备和网络接口可附接到计算机系统 (2600) 的内核 (2640)。

[0183] 内核 (2640) 可包括一个或多个中央处理单元 (CPU) (2641)、图形处理单元 (GPU) (2642)、现场可编程门区域 (FPGA) (2643) 形式的专用可编程处理单元、用于某些任务的硬件加速器 (2644)、图形适配器 (2650) 等。这些设备以及只读存储器 (ROM) (2645)、随机存取存储器 (2646)、诸如内部非用户可访问的硬盘驱动器、SSD 等之类的内部大容量存储器 (2647) 可通过系统总线 (2648) 连接。在一些计算机系统中, 可以以一个或多个物理插头的形式访问系统总线 (2648), 以能够通过附加的 CPU、GPU 等进行扩展。外围设备可直接附接到内核的系统总线 (2648) 或通过外围总线 (2649) 附接到内核的系统总线 (2648)。在一个示例中, 屏幕 (2610) 可连接到图形适配器 (2650)。外围总线的架构包括 PCI、USB 等。

[0184] CPU (2641)、GPU (2642)、FPGA (2643) 和加速器 (2644) 可执行某些指令, 这些指令可组合来构成上述计算机代码。该计算机代码可存储在 ROM (2645) 或 RAM (2646) 中。过渡数据还可存储在 RAM (2646) 中, 而永久数据可例如存储在内部大容量存储器 (2647) 中。可通过使用高速缓存来进行通向任何存储设备的快速存储及检索, 该高速缓存可与下述紧密关联: 一个或多个 CPU (2641)、GPU (2642)、大容量存储 (2647)、ROM (2645)、RAM (2646) 等。

[0185] 计算机可读介质可以在其上具有执行各种由计算机实现的操作的计算机代码。介质和计算机代码可以是出于本公开的目的而专门设计和构造的介质和计算机代码, 或者介质和计算机代码可以是计算机软件领域的技术人员公知且可用的类型。

[0186] 作为非限制性示例, 可由于一个或多个处理器 (包括 CPU、GPU、FPGA、加速器等) 执行包含在一种或多种有形的计算机可读介质中的软件而使得具有架构 (2600), 特别是内核 (2640) 的计算机系统提供功能。此类计算机可读介质可以是与如上所介绍的用户可访问的大容量存储相关联的介质, 以及某些非暂时性内核 (2640) 的存储器, 例如内核内部大容量存储器 (2647) 或 ROM (2645)。实施本公开的各个实施例的软件可存储在此类设备中并由内核 (2640) 执行。根据特定需要, 计算机可读介质可包括一个或多个存储设备或芯片。软件可使得内核 (2640), 特别是其中的处理器 (包括 CPU、GPU、FPGA 等) 执行本文所描述的特定过程或特定过程的特定部分, 包括定义存储在 RAM (2646) 中的数据结构以及根据由软件定义的过程来修改此类数据结构。除此之外或作为替代, 可由于硬连线或以其它方式体现在电路 (例如: 加速器 (2644)) 中的逻辑而使得计算机系统提供功能, 该电路可替代软件或与软件一起运行以执行本文描述的特定过程或特定过程的特定部分。在适当的情况下, 提及软件的部分可包含逻辑, 反之亦然。在适当的情况下, 提及计算机可读介质的部分可包括存储用于执行的软件的电路 (例如, 集成电路 (IC))、体现用于执行的逻辑的电路或包括两者。本公开包括硬件和软件的任何合适的组合。

[0187] 虽然参考说明性实施例描述了特定发明,但是该描述并不意味着是限制性的。根据该描述,对本发明的说明性实施例和附加实施例的各种修改对于本领域普通技术人员来说将是显而易见的。本领域技术人员将容易认识到,在不背离本发明的精神和范围的情况下,可以对本文所说明和描述的示例性实施例进行这些和各种其它修改。因此,预计所附的权利要求将涵盖任何此类修改和替代实施例。图示的某些比例可能被夸大,而其它比例可能最小化。因此,公开内容和附图应被视为说明性的,而非限制性的。

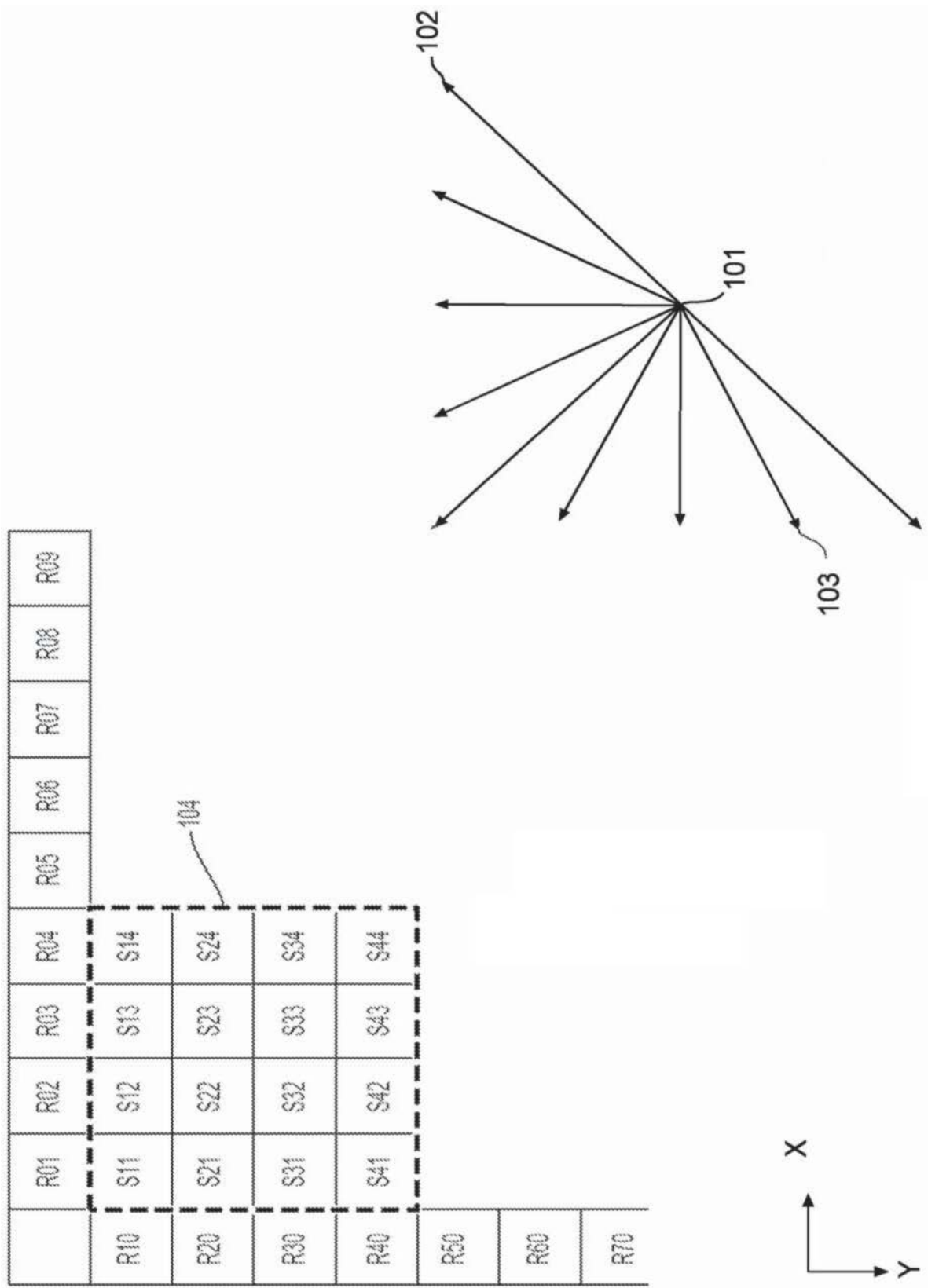


图1A(现有技术)

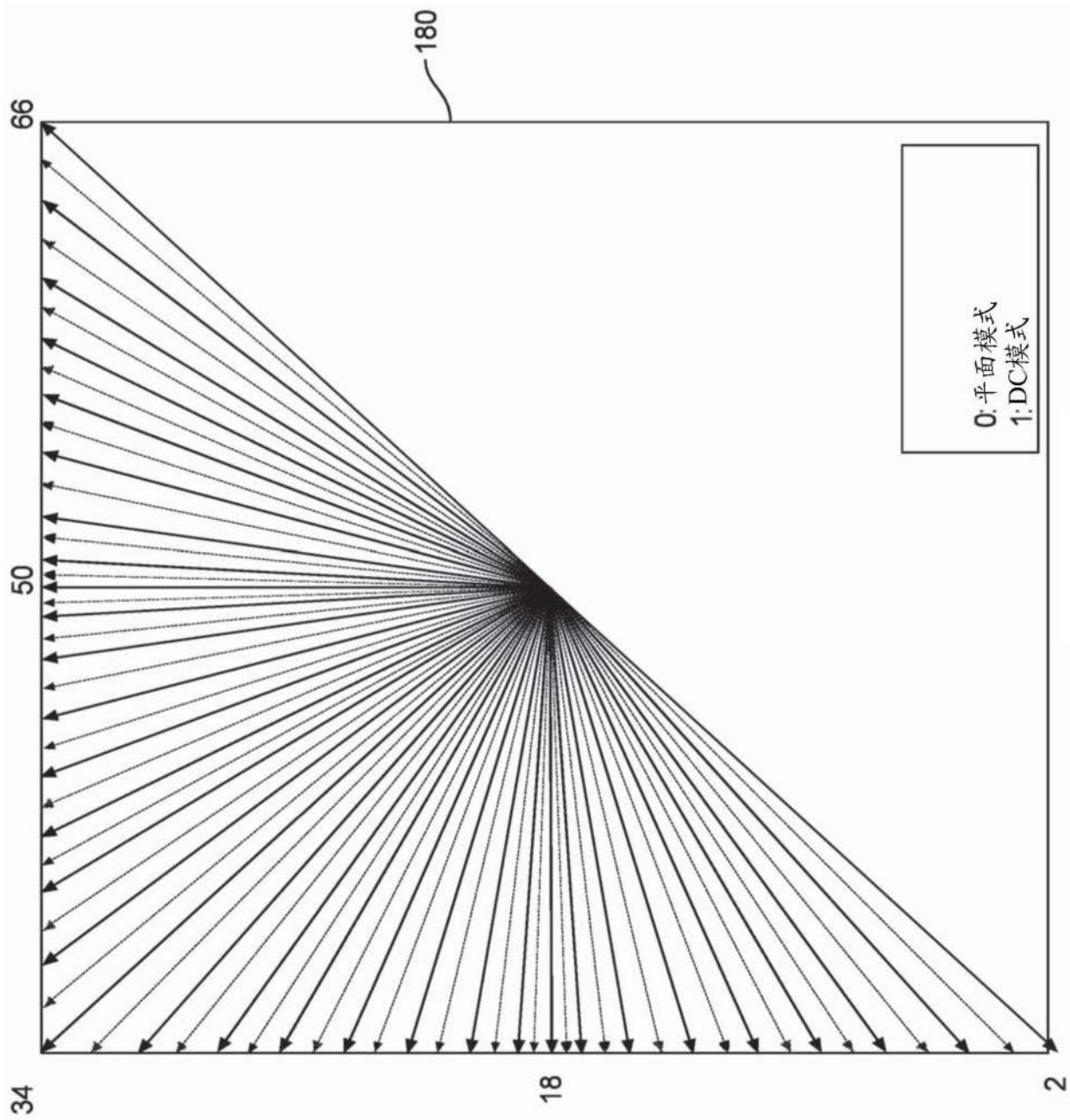


图1B (现有技术)

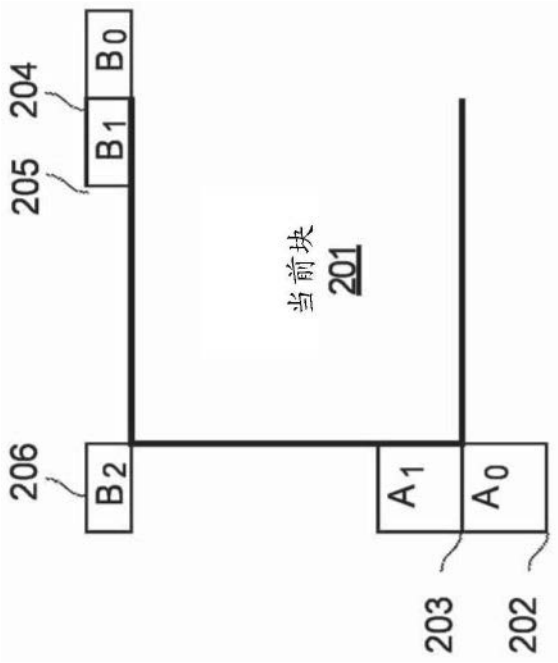


图2 (现有技术)

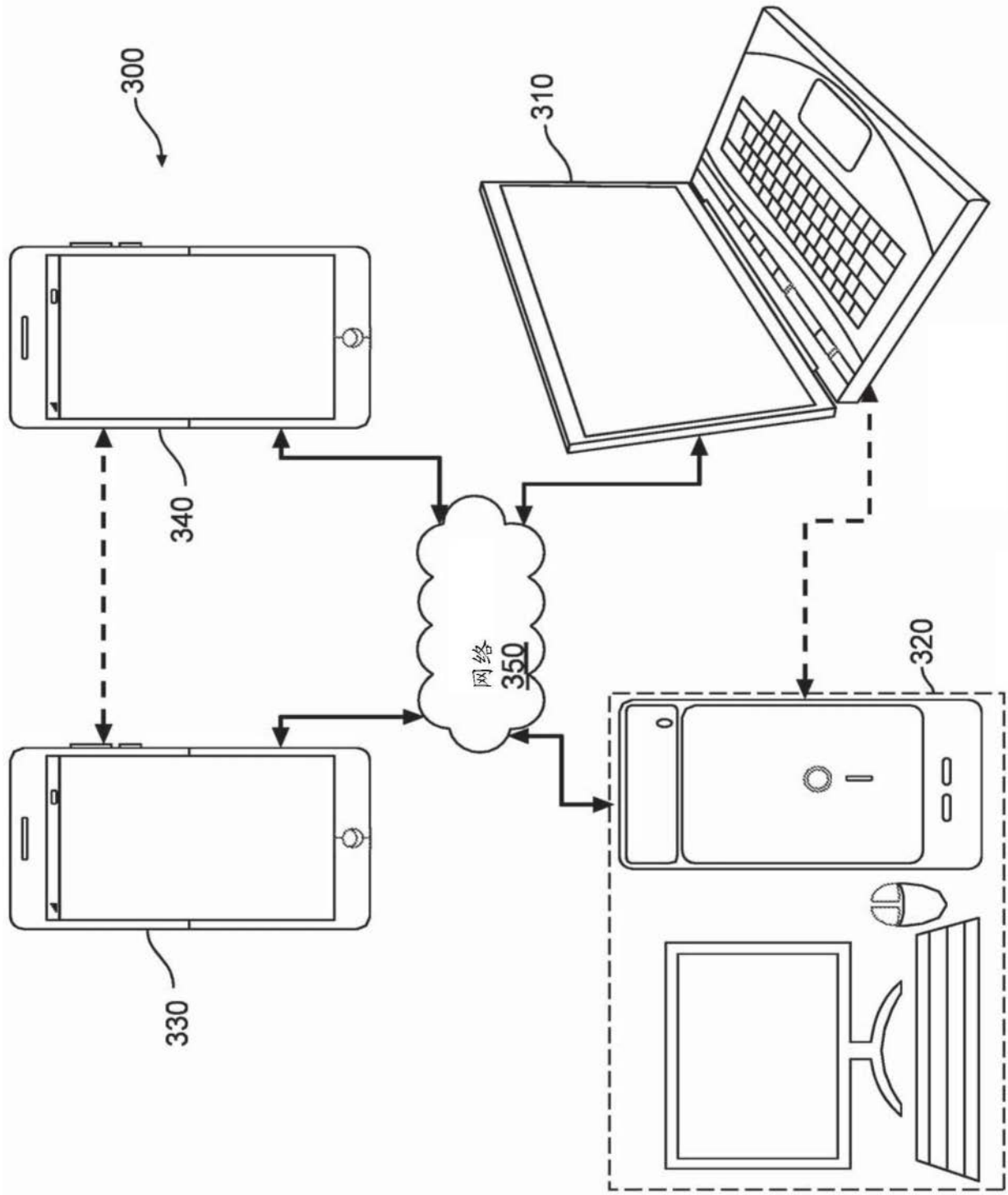


图3

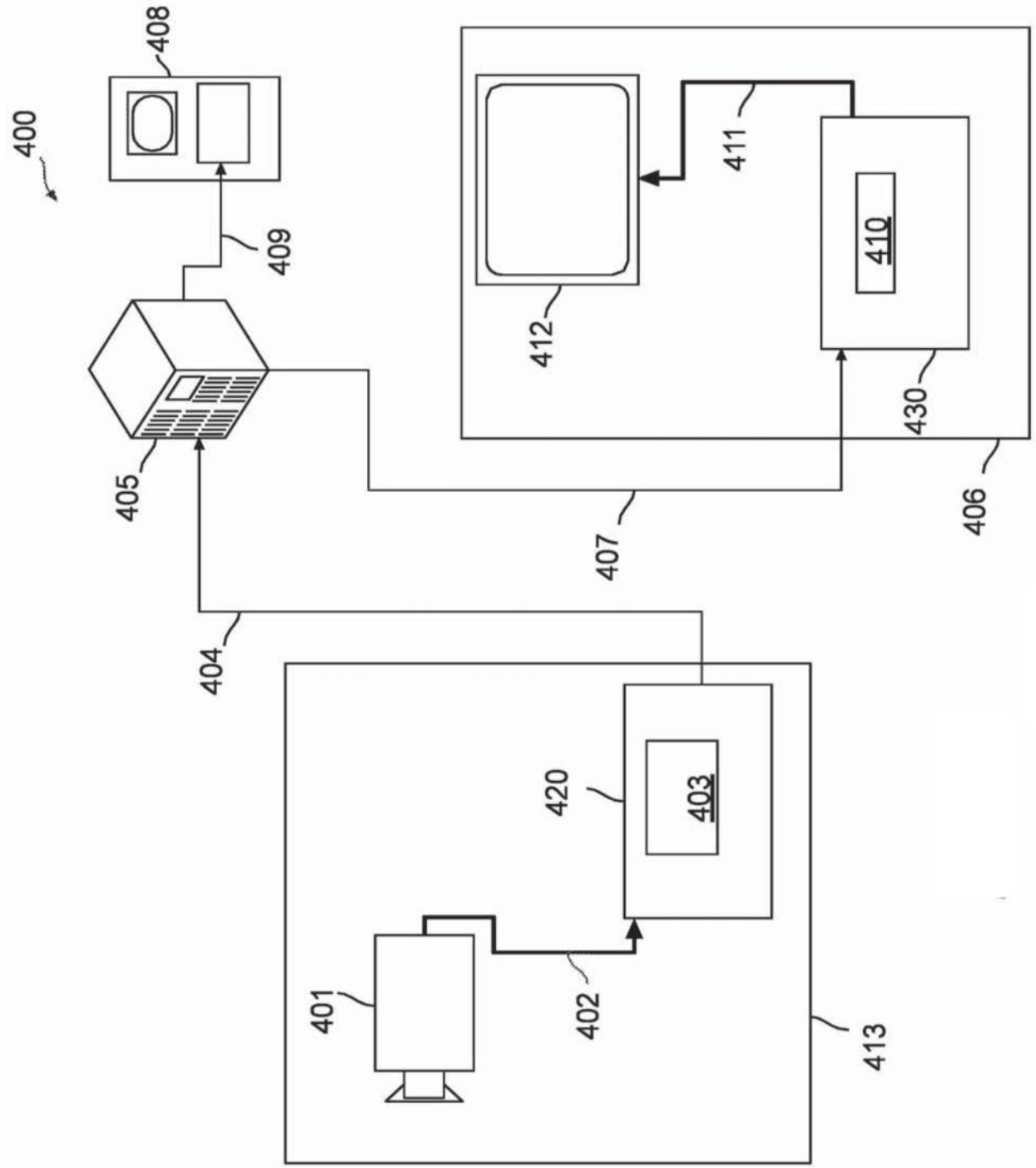


图4

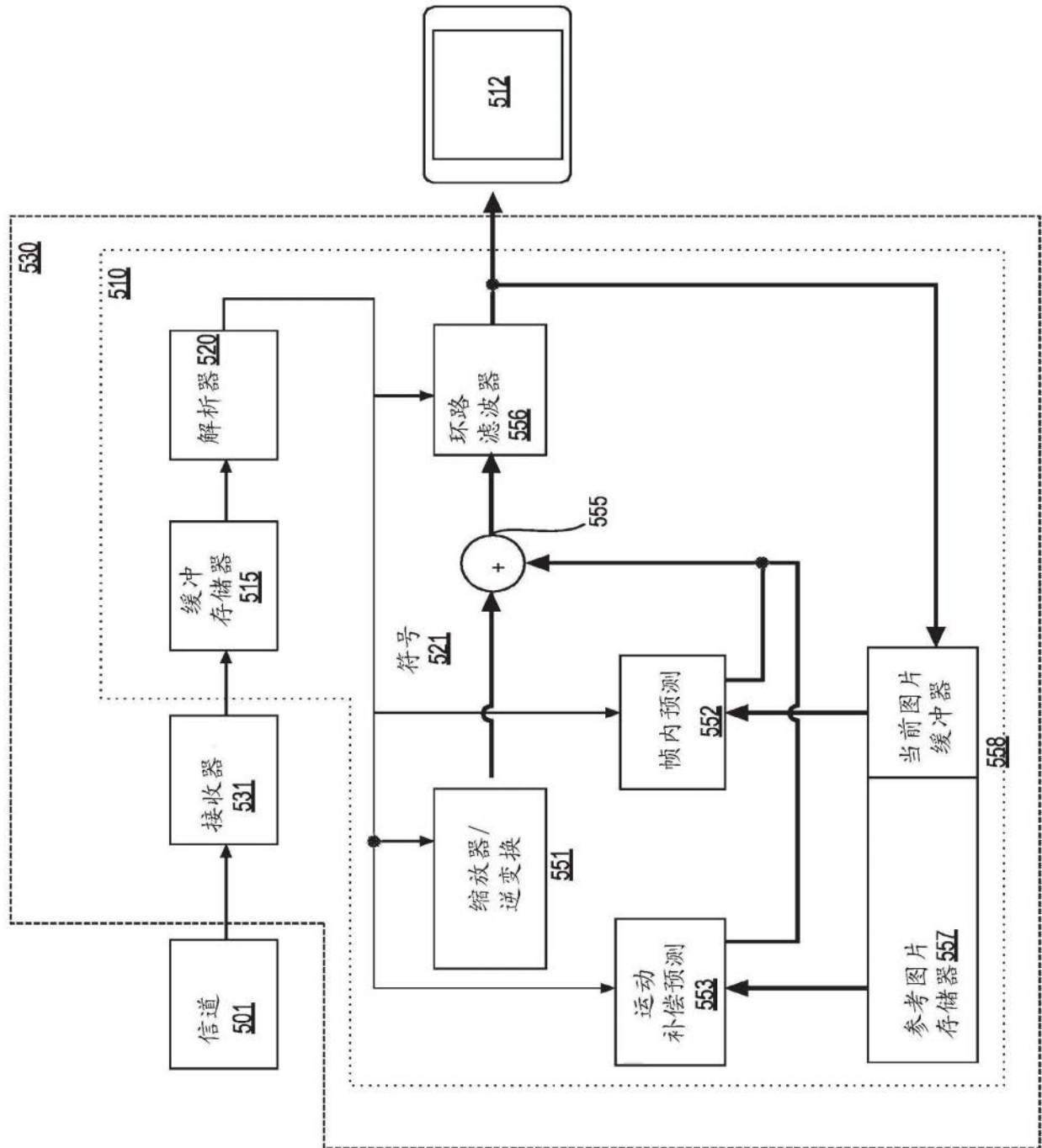


图5

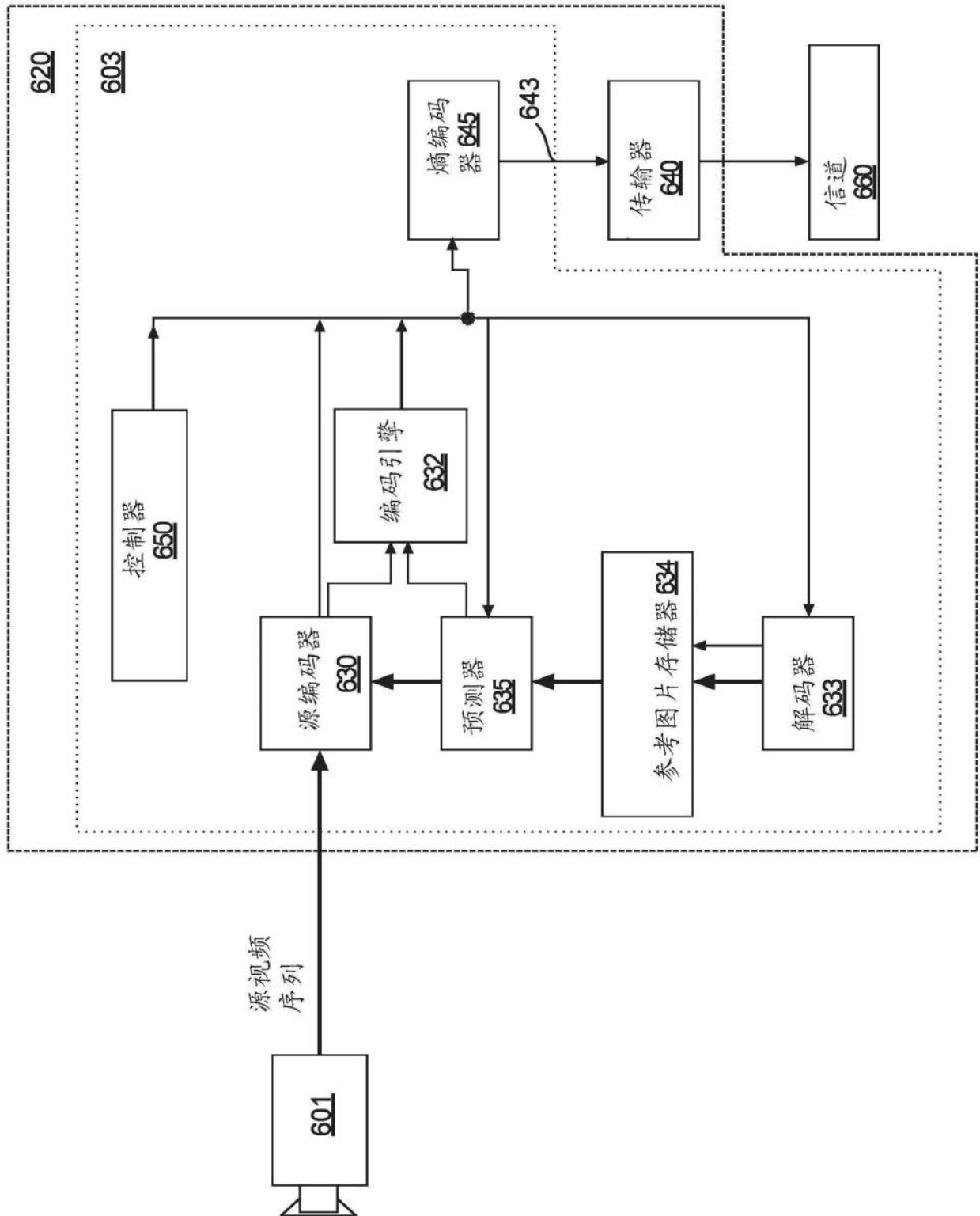


图6

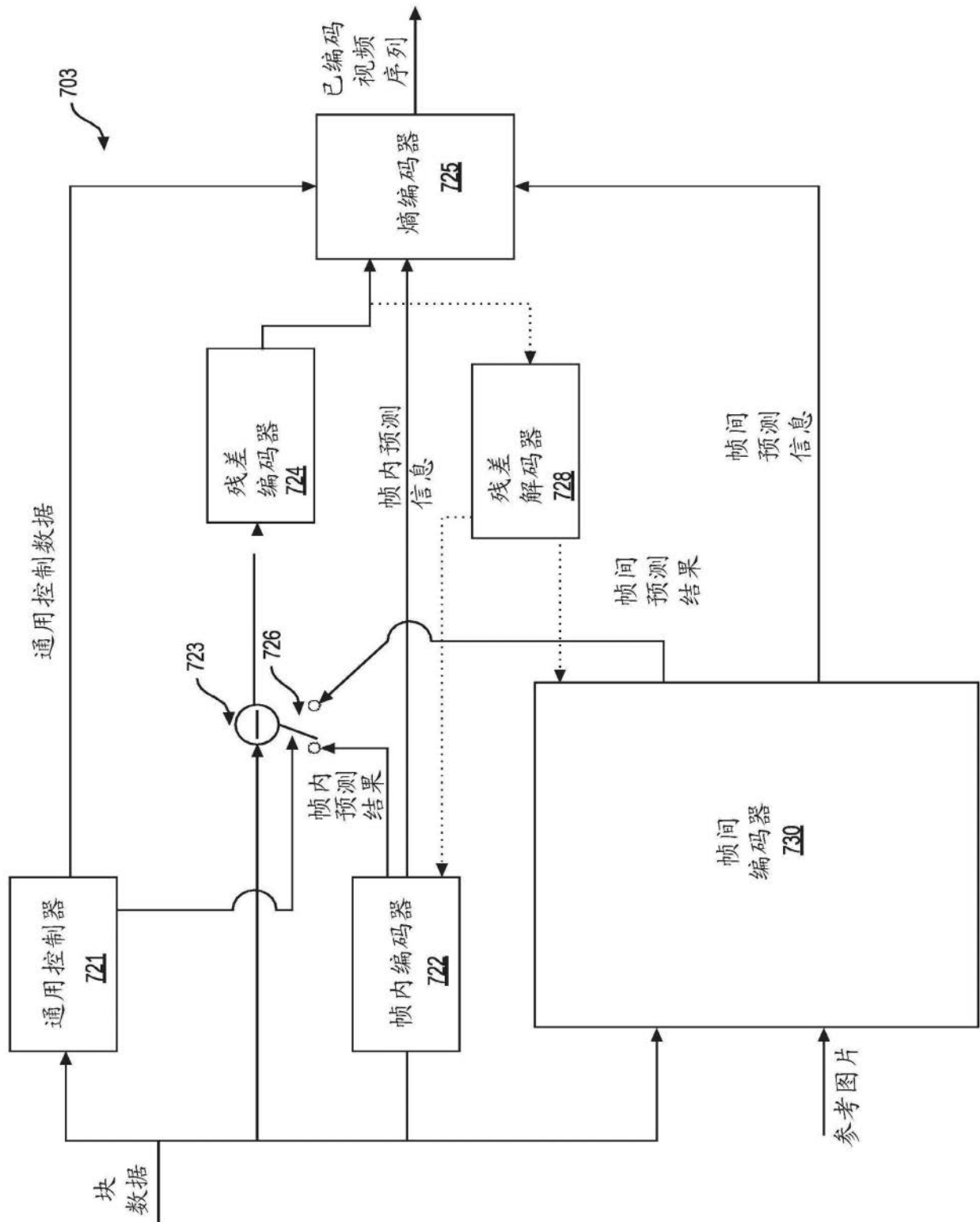


图7

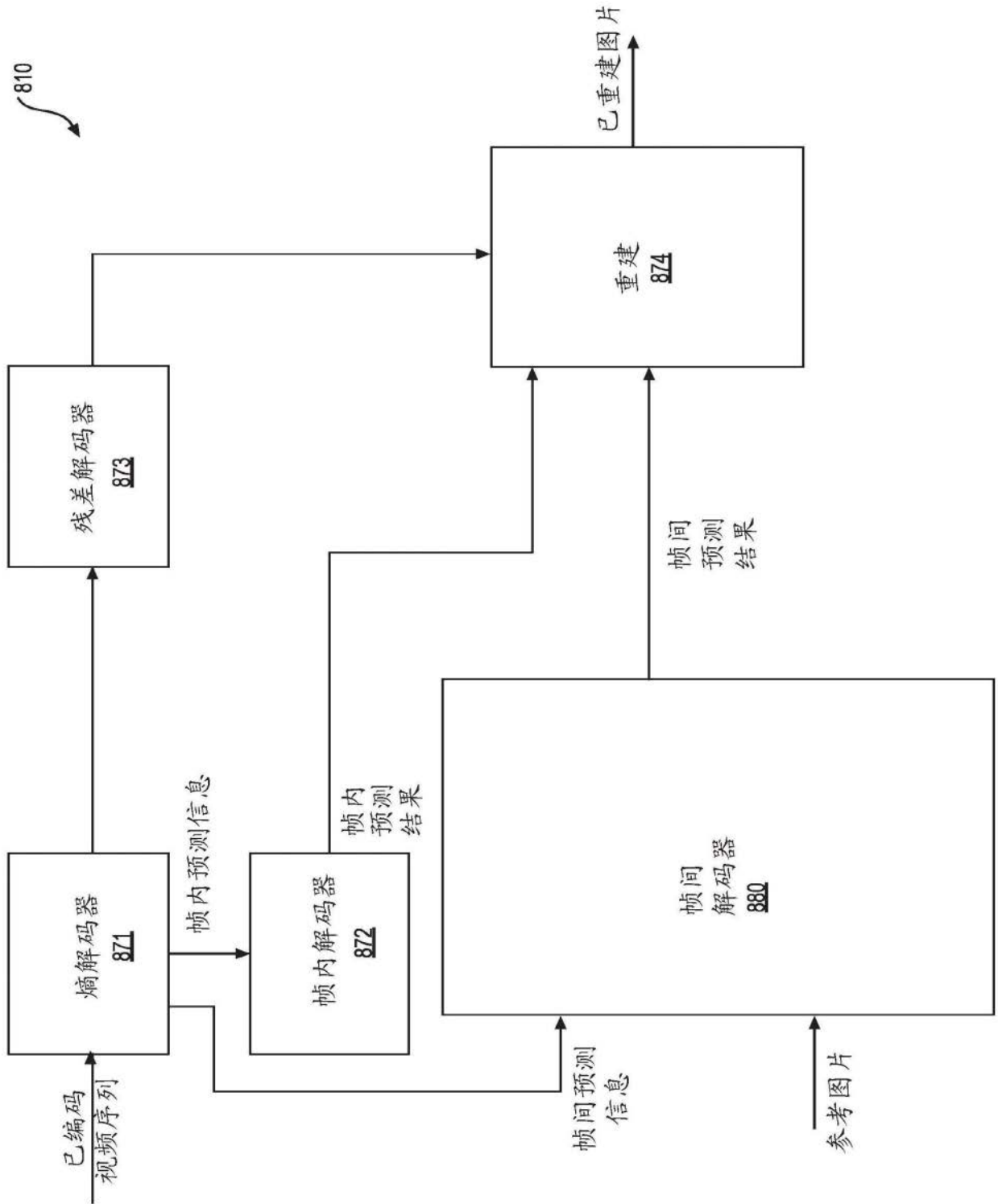


图8

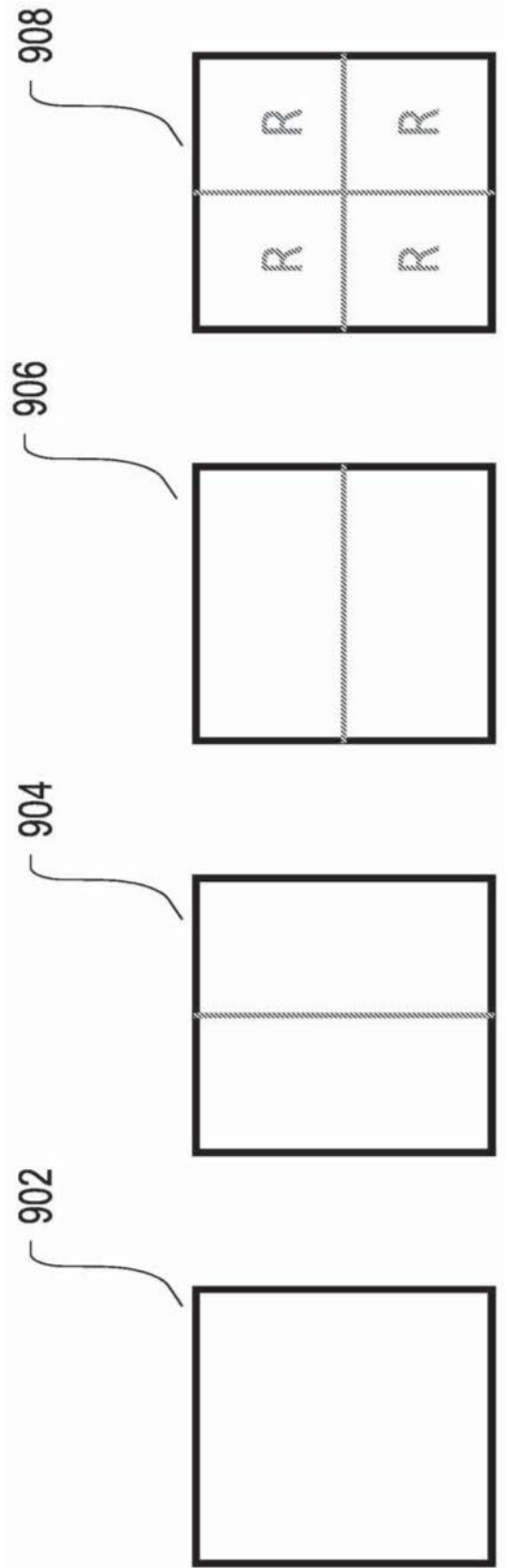


图9

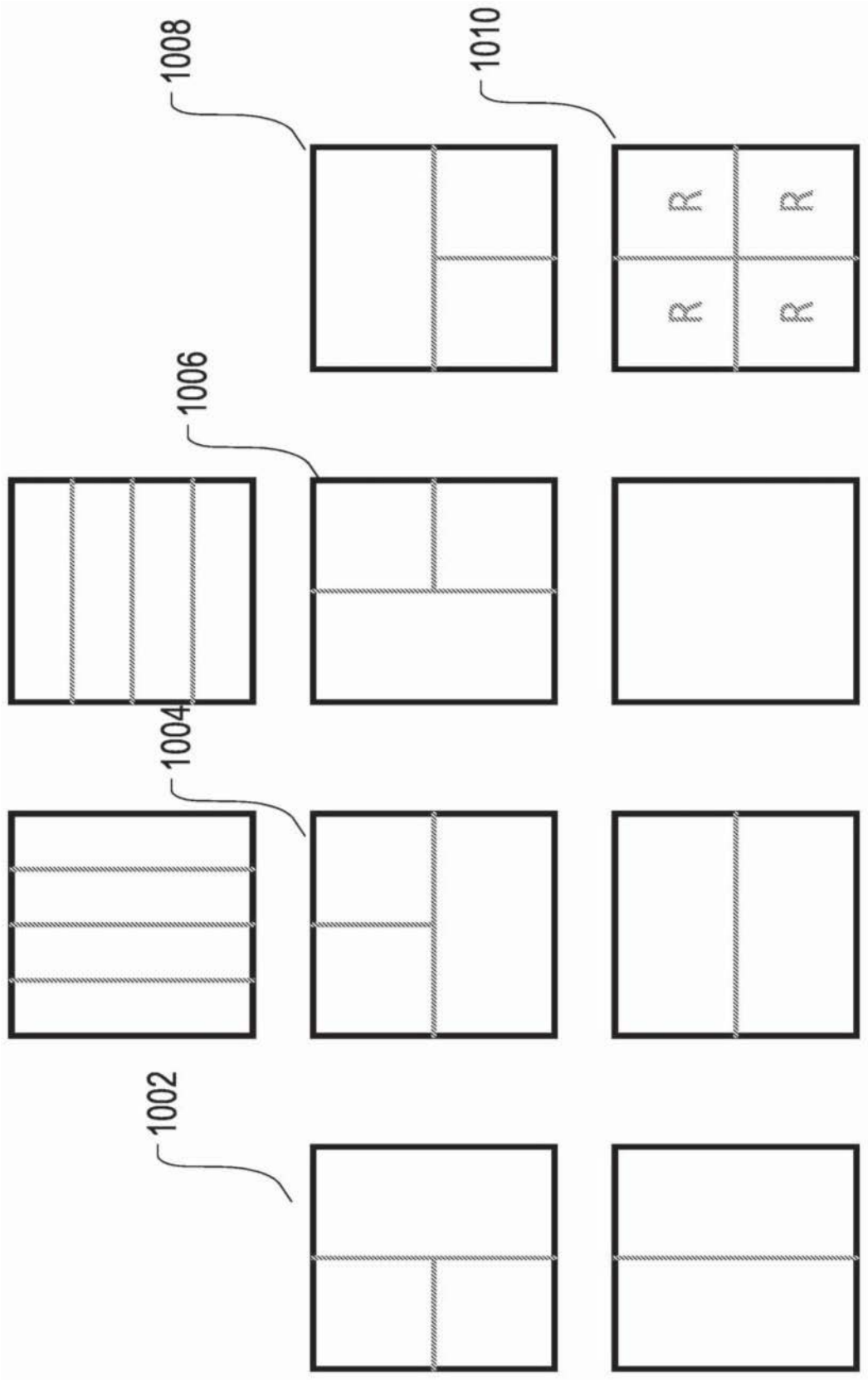


图10

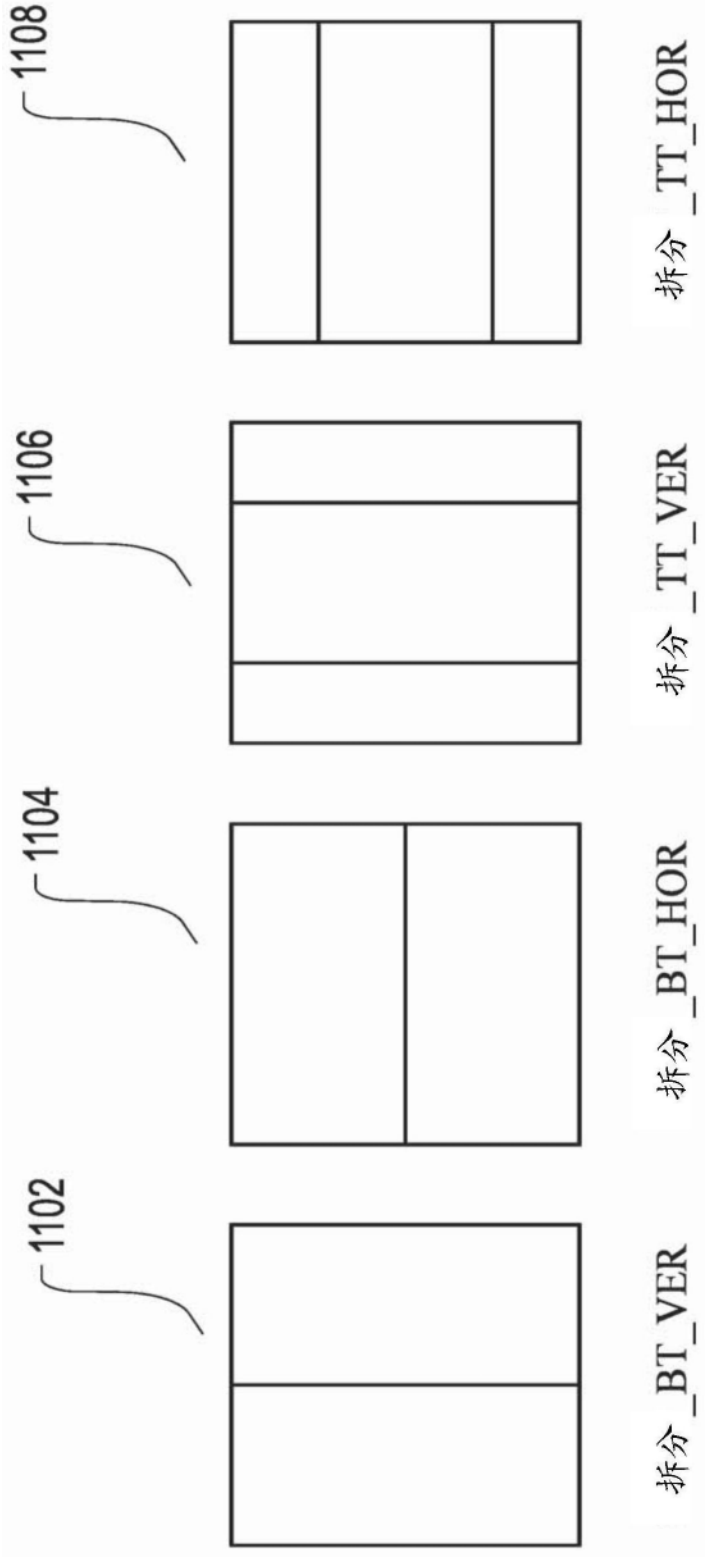


图11

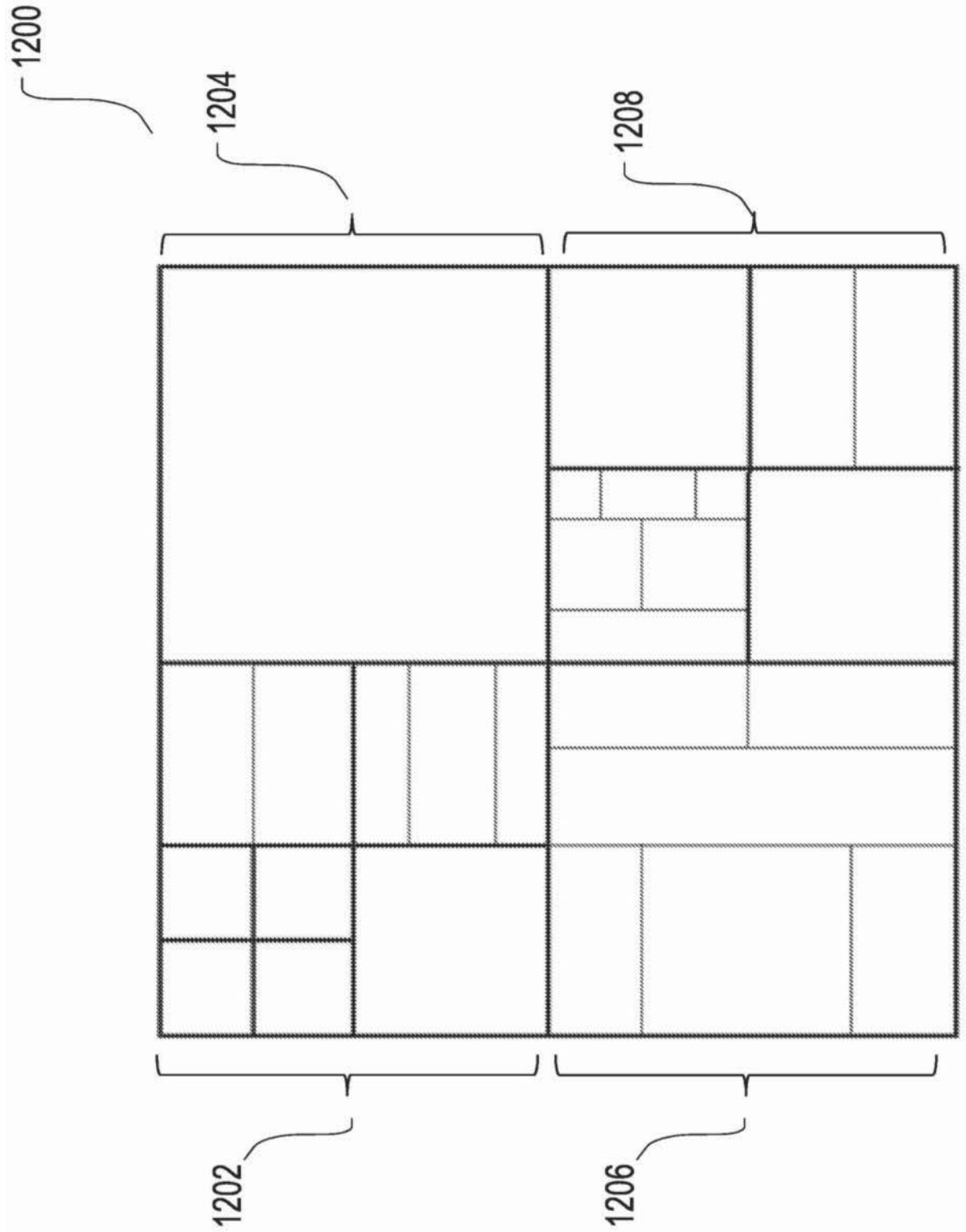


图12

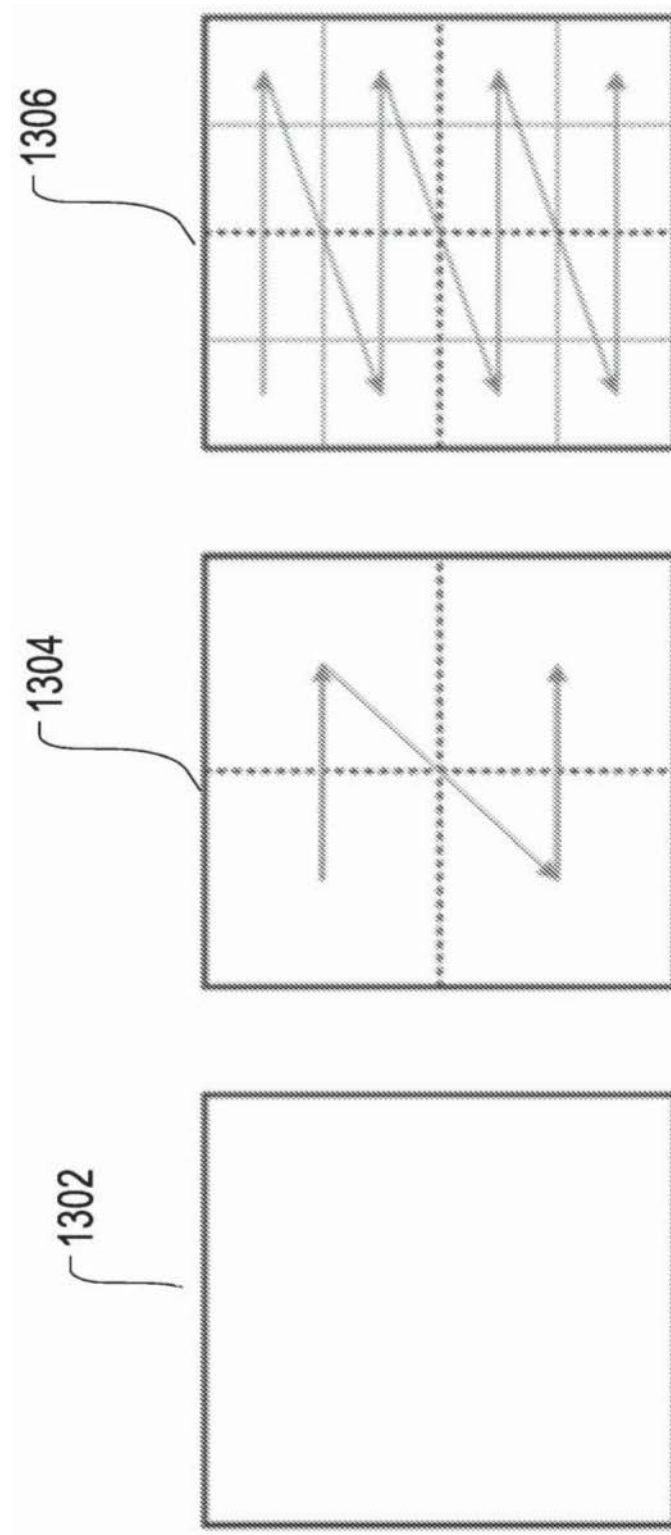


图13

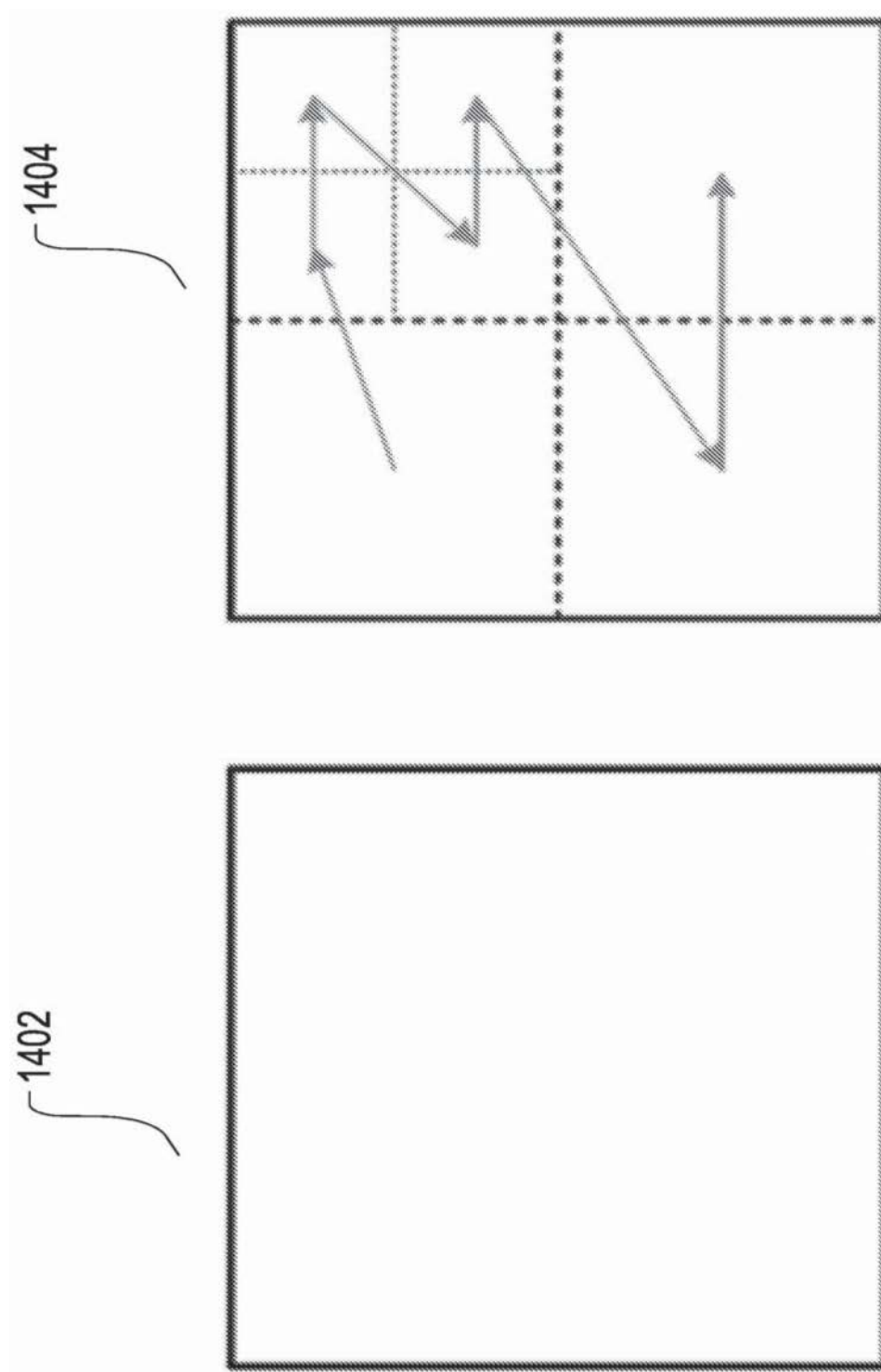


图14

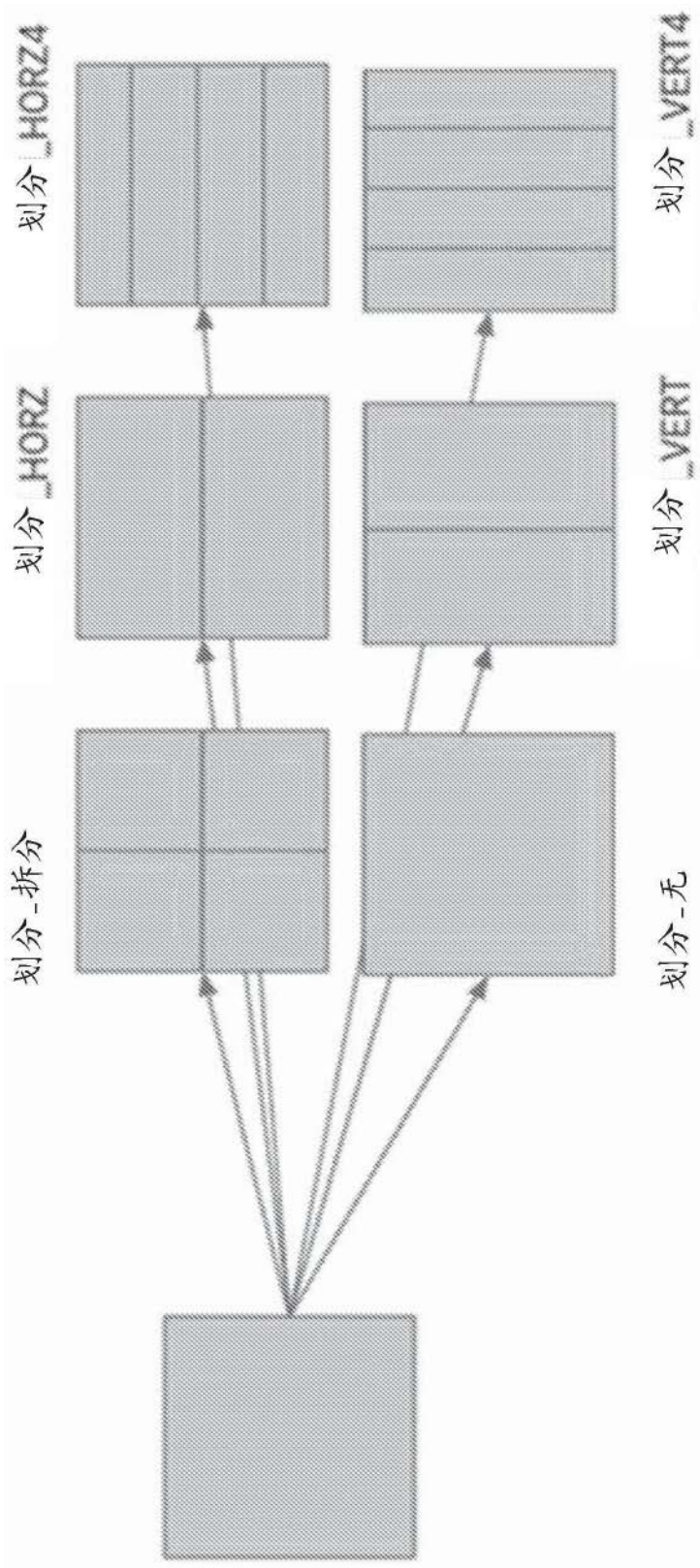


图15

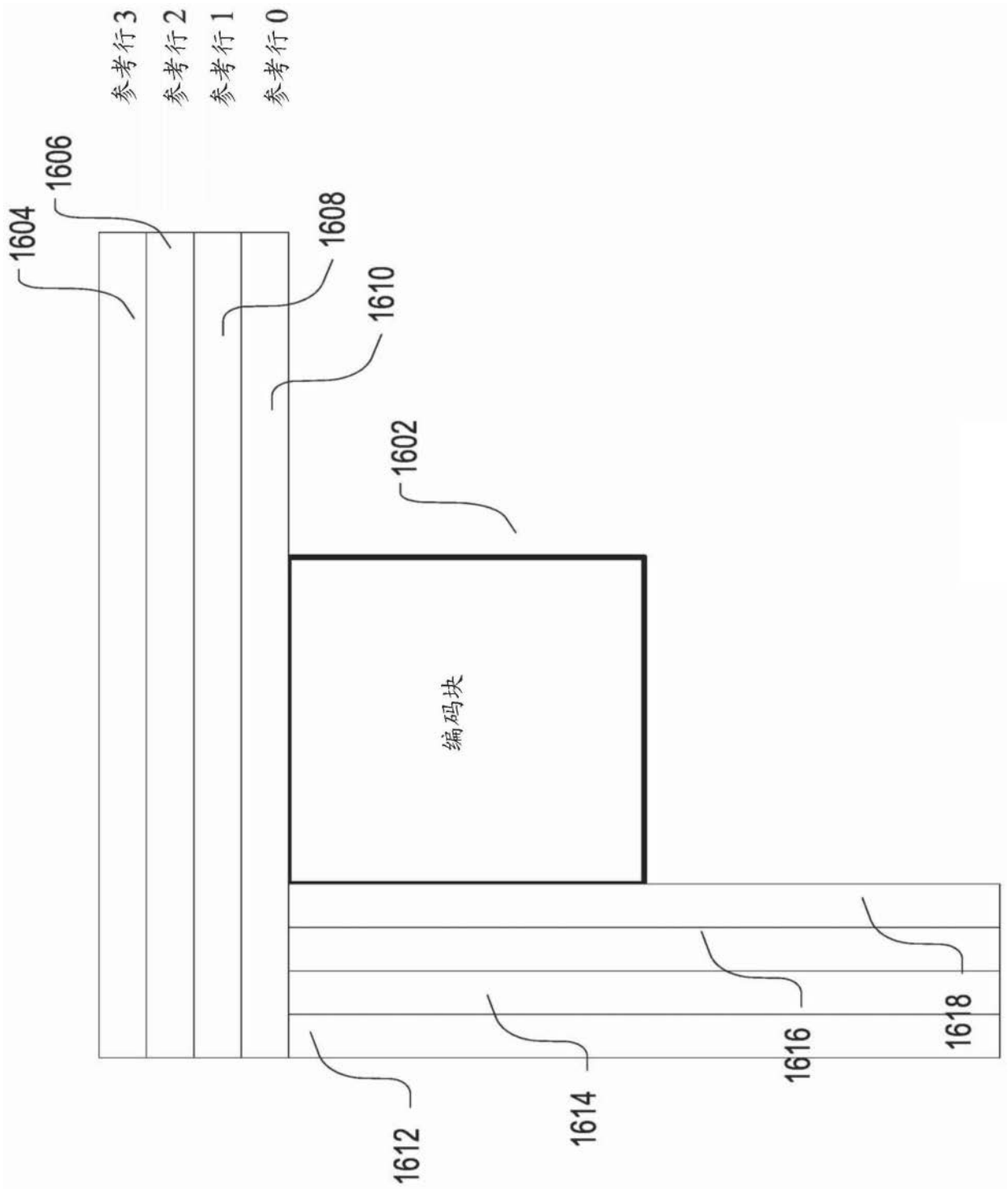


图16

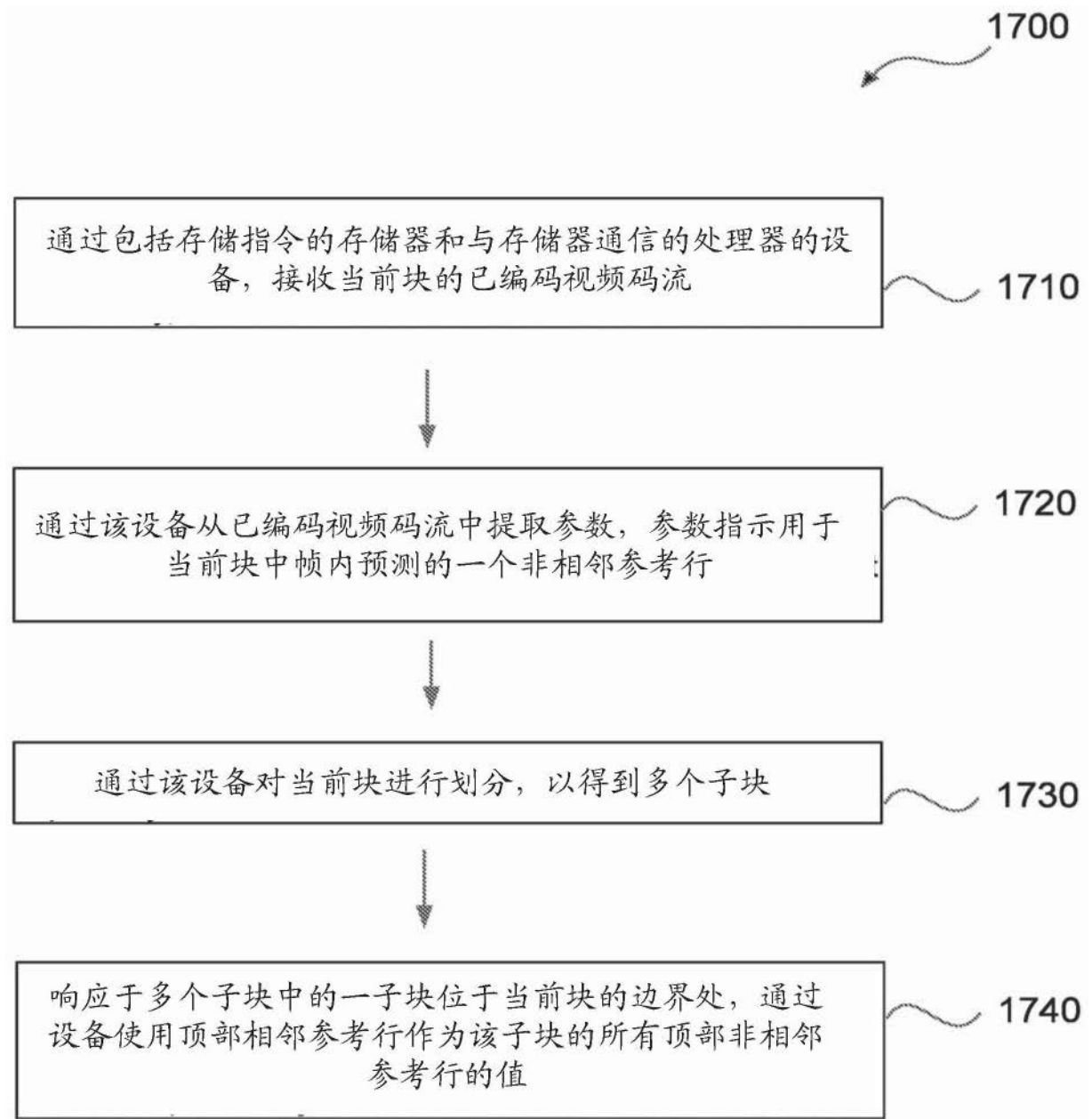


图17

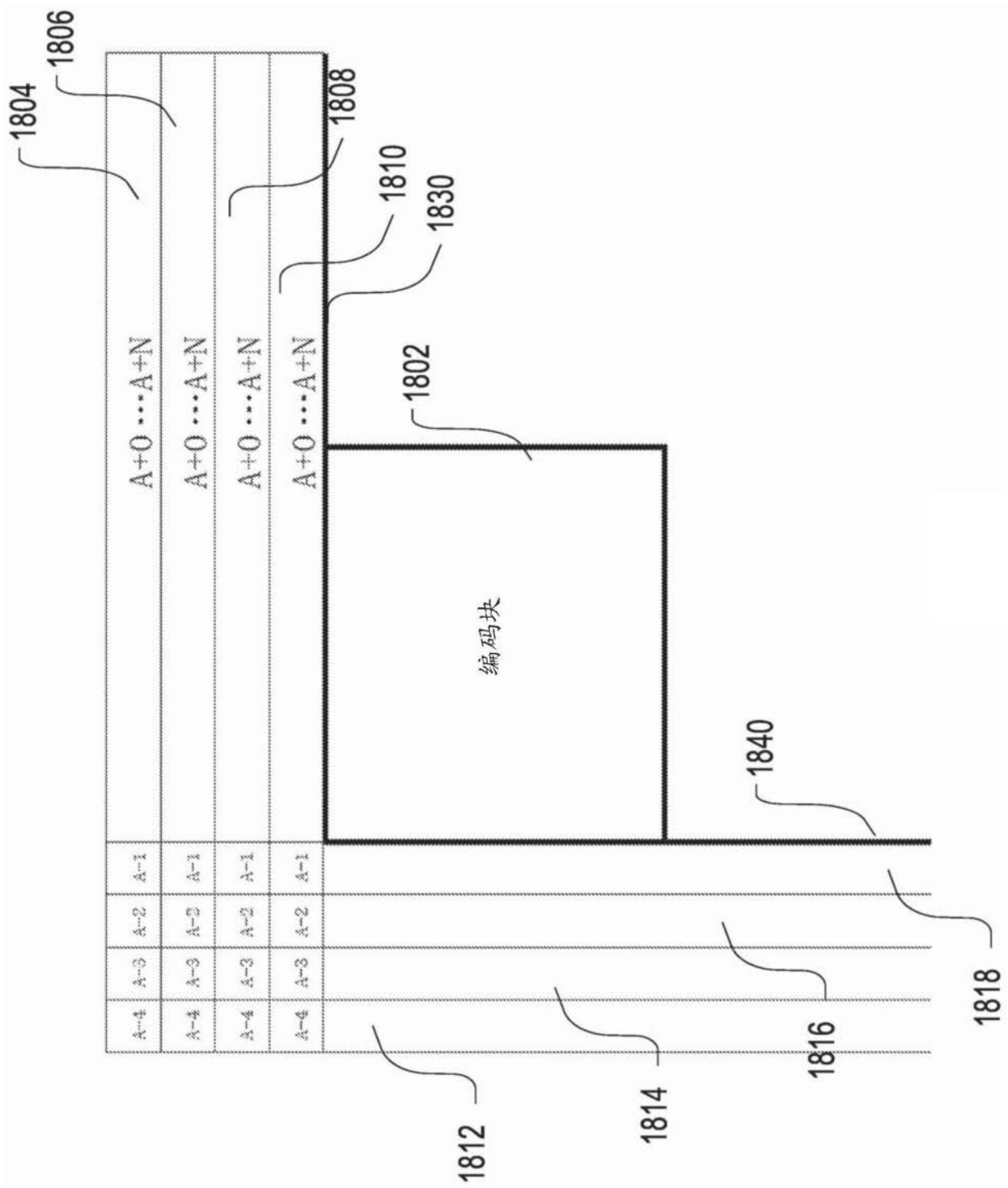


图18

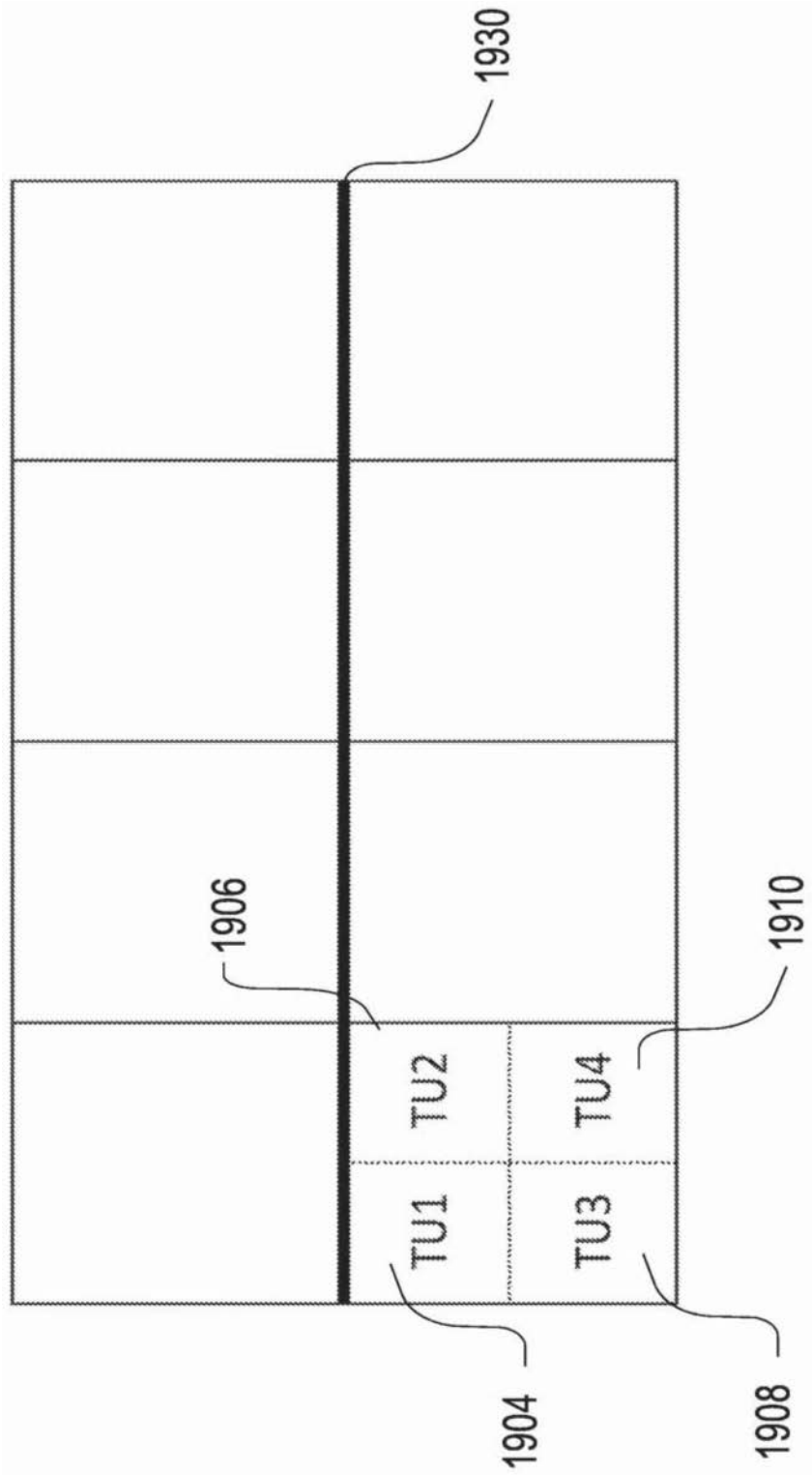


图19

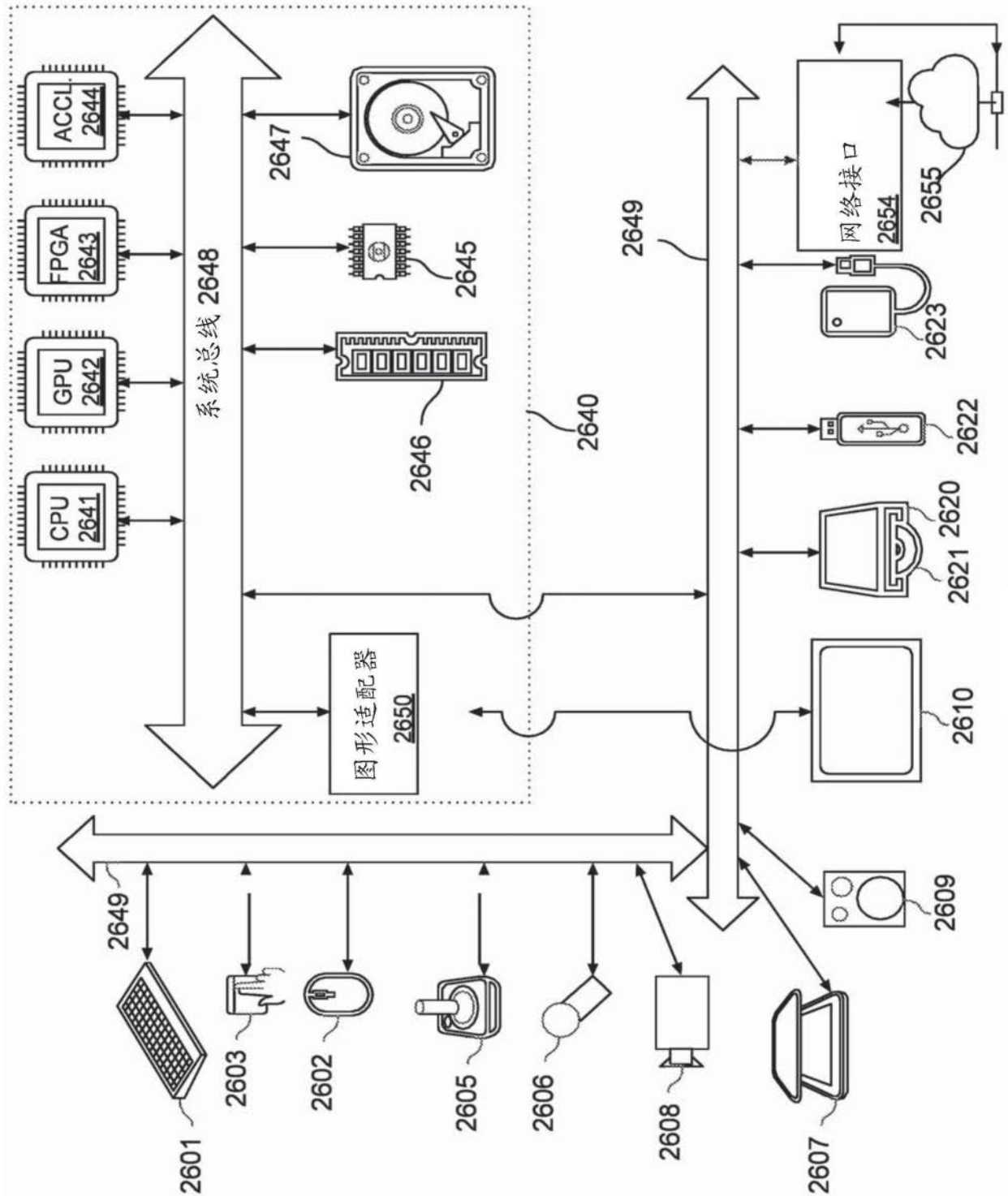


图20