



(12) 发明专利申请

(10) 申请公布号 CN 118575478 A

(43) 申请公布日 2024. 08. 30

(21) 申请号 202380017539.5

(22) 申请日 2023.01.18

(30) 优先权数据

63/300,631 2022.01.18 US

(85) PCT国际申请进入国家阶段日

2024.07.17

(86) PCT国际申请的申请数据

PCT/US2023/011077 2023.01.18

(87) PCT国际申请的公布数据

W02023/141177 EN 2023.07.27

(71) 申请人 北京达佳互联信息技术有限公司

地址 100085 北京市海淀区西二旗西路16
号院12号楼8层101

(72) 发明人 陈漪纹 修晓宇 郭哲玮 朱弘正

陈伟 闫宁 王祥林 于冰

(74) 专利代理机构 北京市汉坤律师事务所

11602

专利代理师 初媛媛 吴丽丽

(51) Int.Cl.

H04N 19/577 (2006.01)

H04N 19/159 (2006.01)

H04N 19/132 (2006.01)

H04N 19/105 (2006.01)

H04N 19/176 (2006.01)

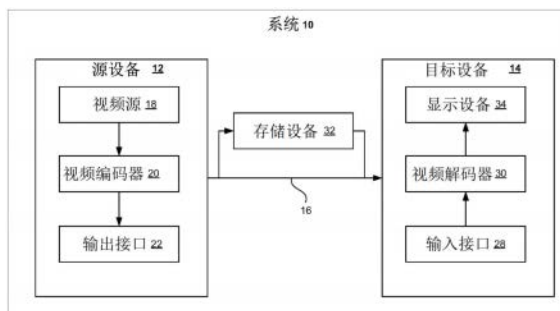
权利要求书3页 说明书46页 附图37页

(54) 发明名称

视频编解码中考虑边界外条件的运动补偿

(57) 摘要

提供了用于视频编解码的方法、装置和非暂态计算机可读存储介质。在一种方法中,解码器推导用于当前编码块的第一参考图片和第二参考图片;所述解码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导第一预测值样点和第二预测值样点;所述解码器确定所述第一预测值样点和所述第二预测值样点是否是边界外的(OOB),并且响应于确定所述第一预测值样点或所述第二预测值样点是OOB,所述解码器用第一替代预测值样点或第二替代预测值样点替换作为OOB的所述第一预测值样点或所述第二预测值样点;并且所述解码器通过对所述第一预测值样点和所述第二预测值样点的样点值求平均值来生成最终预测样点。



1. 一种用于对双向预测编码块进行解码的方法,所述方法包括:
 - 由解码器推导用于当前编码块的第一参考图片;
 - 由所述解码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导第一预测值样点;
 - 由所述解码器确定所述第一预测值样点是否是边界外的(OOB),并且响应于确定所述第一预测值样点是OOB,由所述解码器用第一替代预测值样点替换作为OOB的所述第一预测值样点;
 - 由所述解码器推导用于所述当前编码块的第二参考图片;
 - 由所述解码器使用运动补偿过程基于与所述第二参考图片相关联的运动矢量来从所述第二参考图片中推导第二预测值样点;
 - 由所述解码器确定所述第二预测值样点是否是边界外的(OOB),并且响应于确定所述第二预测值样点是OOB,由所述解码器用第二替代预测值样点替换作为OOB的所述第二预测值样点;以及
 - 通过对所述第一预测值样点和所述第二预测值样点的样点值求平均值来生成最终预测样点。
2. 如权利要求1所述的用于视频解码的方法,其中,用所述第一替代预测值样点替换作为OOB的所述第一预测值样点进一步包括:
 - 将所述第一替代预测值样点设置为所述第二预测值样点。
3. 如权利要求1所述的用于视频解码的方法,其中,用所述第二替代预测值样点替换作为OOB的所述第二预测值样点进一步包括:
 - 将所述第二替代预测值样点设置为所述第一预测值样点。
4. 一种用于对双向预测编码块进行编码的方法,所述方法包括:
 - 由编码器推导用于当前编码块的第一参考图片;
 - 由所述编码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导第一预测值样点;
 - 由所述编码器确定所述第一预测值样点是否是边界外的(OOB);并且响应于确定所述第一预测值样点是OOB,由所述编码器用第一替代预测值样点替换作为OOB的所述预测值样点;
 - 由所述编码器推导用于所述当前编码块的第二参考图片;
 - 由所述编码器使用运动补偿过程基于与所述第二参考图片相关联的运动矢量来从所述第二参考图片中推导第二预测值样点;
 - 由所述编码器确定所述第二预测值样点是否是边界外的(OOB);并且响应于确定所述第二预测值样点是OOB,由所述解码器用第二替代预测值样点替换作为OOB的所述第二预测值样点;以及
 - 通过对所述第一预测值样点和所述第二预测值样点的样点值求平均值来生成最终预测样点。
5. 如权利要求4所述的用于视频编码的方法,其中,用所述第一替代预测值样点替换作为OOB的所述第一预测值样点进一步包括:
 - 将所述第一替代预测值样点设置为所述第二预测值样点。

6. 如权利要求4所述的用于视频编码的方法,其中,用所述第二替代预测值样点替换作为00B的所述第二预测值样点进一步包括:

将所述第二替代预测值样点设置为所述第一预测值样点。

7. 一种用于对单向预测编码块进行解码的方法,所述方法包括:

由解码器推导用于当前编码块的第一参考图片;

由所述解码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导预测值样点;

由所述解码器确定所述预测值样点是否是边界外的(OOB);并且响应于确定所述预测值样点是OOB,由所述解码器用替代预测值样点替换作为OOB的所述预测值样点。

8. 如权利要求7所述的用于对单向预测编码块进行解码的方法,其中,用所述替代预测值样点替换作为OOB的所述预测值样点进一步包括:

基于第二参考图片中的第二预测值样点来推导所述替代预测值样点。

9. 如权利要求8所述的用于对单向预测编码块进行解码的方法,其中,基于所述第二参考图片中的所述第二预测值样点来推导所述替代预测值样点包括:

通过根据当前图片、所述第一参考图片和所述第二参考图片之间的距离缩放所述当前编码块的运动矢量来推导所述第二参考图片中的所述第二预测值样点。

10. 一种用于对单向预测编码块进行编码的方法,所述方法包括:

由编码器推导用于当前编码块的第一参考图片;

由所述编码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导预测值样点;

由所述编码器确定所述预测值样点是否是边界外的(OOB);并且响应于确定所述预测值样点是OOB,由所述编码器用替代预测值样点替换作为OOB的所述预测值样点。

11. 如权利要求10所述的用于视频编码的方法,其中,用所述替代预测值样点替换作为OOB的所述预测值样点进一步包括:

基于第二参考图片中的第二预测值样点来推导所述替代预测值样点。

12. 如权利要求11所述的用于视频编码的方法,其中,基于所述第二参考图片中的所述第二预测值样点来推导所述替代预测值样点:

通过根据当前图片、所述第一参考图片和所述第二参考图片之间的距离缩放所述当前编码块的运动矢量来推导所述第二参考图片中的所述第二预测值样点。

13. 一种装置,包括:

一个或多个处理器;以及

存储器,所述存储器被配置为存储能够由所述一个或多个处理器执行的指令;其中,所述一个或多个处理器在执行所述指令时被配置为执行如权利要求1至12中任一项所述的方法。

14. 一种用于存储计算机可执行指令的非暂态计算机可读存储介质,所述计算机可执行指令在由一个或多个计算机处理器执行时使所述一个或多个计算机处理器接收比特流并基于所述比特流执行如权利要求1至3或7至9中任一项所述的方法。

15. 一种用于存储计算机可执行指令的非暂态计算机可读存储介质,所述计算机可执行指令在由一个或多个计算机处理器执行时使所述一个或多个计算机处理器执行如权利

要求4至6或10至12中任一项所述的方法,以将当前视频块编码成比特流并传输所述比特流。

视频编解码中考虑边界外条件的运动补偿

[0001] 相关申请的交叉引用

[0002] 本申请基于2022年1月18日提交的临时申请号63/300,631并要求其优先权,所述临时申请的全部内容出于所有目的通过引用并入本文。

技术领域

[0003] 本公开涉及视频编解码和压缩。更具体地,本公开涉及用于视频编解码中的帧间预测的方法和装置。

背景技术

[0004] 如数字电视、膝上型计算机或台式计算机、平板计算机、数字相机、数字记录设备、数字媒体播放器、视频游戏机、智能电话、视频电话会议设备、视频流式传输设备等各种电子设备都支持数字视频。电子设备通过通信网络发射和接收或以其他方式传送数字视频数据和/或将数字视频数据存储在存储设备上。由于通信网络的带宽容量有限并且存储设备的存储器资源有限,在将视频数据传送或存储之前,可以使用视频编解码以根据一种或多种视频编解码标准来压缩视频数据。例如,视频编解码标准包括通用视频编解码(VVC)、联合探索测试模型(JEM)、高效视频编解码(HEVC/H.265)、高级视频编解码(AVC/H.264)、运动图片专家组(MPEG)编解码等。视频编解码通常使用预测方法(例如,帧间预测、帧内预测等)进行,所述预测方法利用视频数据中固有的冗余。视频编解码旨在将视频数据压缩成在避免或最小化视频质量降级的同时使用较低比特率的形式。

发明内容

[0005] 本公开的示例提供了用于视频编解码中的帧间预测的方法和装置。

[0006] 根据本公开的第一方面,提供了一种用于对双向预测编码块进行解码的方法。所述方法可以包括:由解码器推导用于当前编码块的第一参考图片;由所述解码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导第一预测值样点;由所述解码器确定所述第一预测值样点是否是边界外的(OOB),并且响应于确定所述第一预测值样点是OOB,由所述解码器用第一替代预测值样点替换作为OOB的所述第一预测值样点;由所述解码器推导用于所述当前编码块的第二参考图片;由所述解码器使用运动补偿过程基于与所述第二参考图片相关联的运动矢量来从所述第二参考图片中推导第二预测值样点;由所述解码器确定所述第二预测值样点是否是边界外的(OOB),并且响应于确定所述第二预测值样点是OOB,由所述解码器用第二替代预测值样点替换作为OOB的所述第二预测值样点;以及通过对所述第一预测值样点和所述第二预测值样点的样点值求平均值来生成最终预测样点。

[0007] 根据本公开的第二方面,提供了一种用于对双向预测编码块进行编码的方法。所述方法可以包括:由编码器推导用于当前编码块的第一参考图片;由所述编码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导第一

预测值样点;由所述编码器确定所述第一预测值样点是否是边界外的(OOB);并且响应于确定所述第一预测值样点是OOB,由所述编码器用第一替代预测值样点替换作为OOB的所述预测值样点;由所述编码器推导用于所述当前编码块的第二参考图片;由所述编码器使用运动补偿过程基于与所述第二参考图片相关联的运动矢量来从所述第二参考图片中推导第二预测值样点;由所述编码器确定所述第二预测值样点是否是边界外的(OOB);并且响应于确定所述第二预测值样点是OOB,由所述解码器用第二替代预测值样点替换作为OOB的所述第二预测值样点;以及通过对所述第一预测值样点和所述第二预测值样点的样点值求平均值来生成最终预测样点。

[0008] 根据本公开的第三方面,提供了一种用于对单向预测编码块进行解码的方法。所述方法可以包括:由解码器推导用于当前编码块的第一参考图片;由所述解码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导预测值样点;由所述解码器确定所述预测值样点是否是边界外的(OOB);并且响应于确定所述预测值样点是OOB,由所述解码器用替代预测值样点替换作为OOB的所述预测值样点。

[0009] 根据本公开的第四方面,提供了一种用于对单向预测编码块进行编码的方法。所述方法可以包括:由编码器推导用于当前编码块的第一参考图片;由所述编码器使用运动补偿过程基于与所述第一参考图片相关联的运动矢量来从所述第一参考图片中推导预测值样点;由所述编码器确定所述预测值样点是否是边界外的(OOB);并且响应于确定所述预测值样点是OOB,由所述编码器用替代预测值样点替换作为OOB的所述预测值样点。

[0010] 应理解,上面的大体描述和下面的详细描述仅是示例性的和解释性的,而不旨在限制本公开。

附图说明

[0011] 结合在说明书中并构成本说明书的一部分的附图图示了与本公开一致的示例,并与所述描述一起用于解释本公开的原理。

[0012] 图1是图示了根据本公开的一些实施方式的用于对视频块进行编码和解码的示例性系统的框图。

[0013] 图2是图示了根据本公开的一些实施方式的示例性视频编码器的框图。

[0014] 图3是图示了根据本公开的一些实施方式的示例性视频解码器的框图。

[0015] 图4A至图4E是图示了根据本公开的一些实施方式的如何将帧递归地分区成具有不同尺寸和形状的视频块的框图。

[0016] 图5图示了根据本公开的一些实施方式的空间合并候选的位置的示例。

[0017] 图6图示了根据本公开的一些实施方式的被考虑用于对空间合并候选进行冗余检查的候选对的示例。

[0018] 图7图示了根据本公开的一些实施方式的用于时间合并候选的示例运动矢量缩放。

[0019] 图8图示了根据本公开的一些实施方式的时间合并候选的候选位置。

[0020] 图9A图示了根据本公开的一些实施方式的MMVD搜索点的示例。

[0021] 图9B图示了根据本公开的一些实施方式的MMVD搜索点的示例。

[0022] 图10A图示了根据本公开的一些实施方式的基于控制点的仿射运动模型的示例。

- [0023] 图10B图示了根据本公开的一些实施方式的基于控制点的仿射运动模型的示例。
- [0024] 图11图示了根据本公开的一些实施方式的每个子块的仿射MVF的示例。
- [0025] 图12图示了根据本公开的一些实施方式的继承的仿射运动预测值的位置。
- [0026] 图13图示了根据本公开的一些实施方式的用于构建的仿射合并模式的候选位置的定位。
- [0027] 图14图示了根据本公开的一些实施方式的子块MV和像素的示例。
- [0028] 图15A图示了根据本公开的一些实施方式的由ATVMP使用的空间邻近块的示例。
- [0029] 图15B图示了根据本公开的一些实施方式的SbTMVP过程的示例。
- [0030] 图16图示了根据本公开的一些实施方式的在BDOF中使用的扩展的CU区域的示例。
- [0031] 图17图示了根据本公开的一些实施方式的解码侧运动矢量细化的示例。
- [0032] 图18图示了根据本公开的一些实施方式的按相同角度分组的GPM分割的示例。
- [0033] 图19图示了根据本公开的一些实施方式的用于几何分区模式的单向预测MV选择的示例。
- [0034] 图20图示了根据本公开的一些实施方式的在CIIP权重推导中使用的顶部邻近块和左侧邻近块的示例。
- [0035] 图21图示了根据本公开的一些实施方式的用于推导空间合并候选的空间邻近块的示例。
- [0036] 图22图示了根据本公开的一些实施方式对初始MV周围的搜索区域执行的模板匹配的示例。
- [0037] 图23图示了根据本公开的一些实施方式的搜索区域中的菱形区域的示例。
- [0038] 图24图示了根据本公开的一些实施方式的在半像素相位下的内插滤波器和VVC内插滤波器的频率响应。
- [0039] 图25图示了根据本公开的一些实施方式的模板以及在参考图片中模板的参考样点的示例。
- [0040] 图26图示了根据本公开的一些实施方式的对于具有子块运动的块使用当前块的子块的运动信息的模板和模板的参考样点的示例。
- [0041] 图27图示了根据本公开的一些实施方式的用于运动补偿的已填充参考图片的示例。
- [0042] 图28图示了根据本公开的一些实施方式的分数内插的示例。
- [0043] 图29是图示了根据本公开的一些实施方式的与用户界面耦接的计算环境的框图。
- [0044] 图30是示出了根据本公开的一些实施方式的视频编解码过程的框图。
- [0045] 图32图示了根据本公开的一些实施方式的被划分为CTU的图片的示例。
- [0046] 图32图示了根据本公开的一些实施方式的被分区为瓦片和光栅扫描条带的图片的示例。
- [0047] 图33图示了根据本公开的一些实施方式的被分区为瓦片和矩形条带的图片的示例。
- [0048] 图34图示了根据本公开的一些实施方式的被分区为4个瓦片和4个矩形条带的图片的示例。
- [0049] 图35图示了根据本公开的一些实施方式的被分区为28个子图片的图片的示例。

[0050] 图36图示了根据本公开的一些实施方式的VVC中的水平环绕运动补偿的示例。

[0051] 图37是示出了根据本公开的一些实施方式的视频编解码方法的流程图。

[0052] 图38是示出了根据本公开的一些实施方式的用于对双向预测编码块进行解码的方法的流程图。

[0053] 图39是示出了根据本公开的一些实施方式的用于对双向预测编码块进行编码的方法的流程图。

[0054] 图40是示出了根据本公开的一些实施方式的用于对单向预测编码块进行解码的方法的流程图。

[0055] 图41是示出了根据本公开的一些实施方式的用于对单向预测编码块进行编码的方法的流程图。

具体实施方式

[0056] 现在将详细参考示例实施例,附图中图示了这些实施例的示例。以下描述均参考附图,在附图中,除非另有说明,否则不同附图中的相同参考标记表示相同或相似的要素。以下示例实施例描述中阐述的实施方式并不表示与本公开一致的所有实施方式。而是,它们仅仅是与所附权利要求中叙述的与本公开相关的方面一致的装置和方法的示例。

[0057] 本公开中使用的术语仅出于描述特定实施例的目的,而不旨在限制本公开。如在本公开和所附权利要求中使用的,单数形式“一个(a)”、“一种(an)”和“所述(the)”旨在也包括复数形式,除非上下文另有明确指示。还应当理解,本文中使用的术语“和/或”旨在表示并包括相关联列举项目中的一个或多个项目的任何或所有可能组合。

[0058] 应当理解,尽管本文可以使用术语“第一”、“第二”、“第三”等来描述各种信息,但是这些信息不应受这些术语的限制。这些术语仅仅是用来将一类信息与另一类信息进行区分。例如,在不脱离本公开的范围的情况下,第一信息也可以被称为第二信息;并且类似地,第二信息也可以被称为第一信息。如本文所使用的,根据上下文,术语“如果”可以被理解为意指“当……时”或“在……时”或“响应于判断”。

[0059] 可以使用各种视频编解码技术来压缩视频数据。视频编解码是根据一个或多个视频编解码标准来执行的。例如,如今,一些众所周知的视频编解码标准包括通用视频编解码(VVC)、高效视频编解码(HEVC,也被称为H.265或MPEG-H第2部分)和高级视频编解码(AVC,也被称为H.264或MPEG-4第10部分),所述视频编解码标准由ISO/IEC MPEG和ITU-T VCEG联合开发。A0媒体视频1(AV1)由开放媒体联盟(AOM)开发作为其先前标准VP9的后续标准。音视频编解码(AVS)(其是指数字音频和数字视频压缩标准)是中国数字音视频编解码技术标准工作组(Audio and Video Coding Standard Workgroup of China)开发的另一个视频压缩系列标准。大多数现有视频编解码标准建立在著名的混合视频编解码框架上,即,使用基于块的预测方法(例如,帧间预测、帧内预测)来减少视频图像或序列中存在的冗余,并使用变换编解码来压缩预测误差的能量。视频编解码技术的一个重要目标在于将视频数据压缩成在避免或最小化视频质量降级的同时使用较低比特率的形式。

[0060] 第一代AVS标准包括中国国家标准“信息技术高级音视频编解码第2部分:视频”(被称为AVS1)和“信息技术高级音视频编解码第16部分:广播电视视频”(被称为AVS+)。与MPEG-2标准相比,第一代AVS标准可以在相同的感知质量下提供大约50%的比特率节省。

AVS1标准视频部分于2006年2月作为中国国家标准颁布。第二代AVS标准包括中国国家标准“信息技术高效多媒体编解码”(被称为AVS2)系列,其主要针对额外HD TV节目的传输。AVS2的编解码效率是AVS+的编解码效率的两倍。2016年5月,AVS2被颁布为中国国家标准。同时,AVS2标准视频部分由电气和电子工程师协会(IEEE)作为一项国际应用标准提交。AVS3标准是针对UHD视频应用的新一代视频编解码标准,旨在超越最新国际标准HEVC的编解码效率。2019年3月,在第68届AVS会议上,AVS3-P2基线已经完成,其提供了超过HEVC标准大约30%的比特率节省。目前,存在一种被称为高性能模型(HPM)的参考软件,由AVS工作组维护以展示AVS3标准的参考实施方式。

[0061] 图1是图示了根据本公开的一些实施方式的用于并行地对视频块进行编码和解码的示例性系统10的框图。如图1所示,系统10包括源设备12,所述源设备生成并编码要由目标设备14在稍后时间解码的视频数据。源设备12和目标设备14可以包括多种电子设备中的任一种,所述多种电子设备包括台式计算机或膝上型计算机、平板计算机、智能电话、机顶盒、数字电视、相机、显示设备、数字媒体播放器、视频游戏机、视频流式传输设备等。在一些实施方式中,源设备12和目标设备14配备有无线通信能力。

[0062] 在一些实施方式中,目标设备14可以经由链路16接收要解码的已编码视频数据。链路16可以包括能够将已编码视频数据从源设备12移到目标设备14的任何类型的通信介质或设备。在一个示例中,链路16可以包括用于使源设备12能够实时地将已编码视频数据直接传输到目标设备14的通信介质。已编码视频数据可以根据如无线通信协议等通信标准来调制并传输到目标设备14。通信媒体可以包括任何无线或有线通信媒体,如射频(RF)频谱或一条或多条物理传输线。通信介质可以形成基于分组的网络(如局域网、广域网、或全球网(如互联网))的一部分。通信介质可以包括路由器、交换机、基站、或可以用于促进从源设备12到目标设备14的通信的任何其他装置。

[0063] 在一些其他实施方式中,已编码视频数据可以从输出接口22传输到存储设备32。随后,存储设备32中的已编码视频数据可以由目标设备14经由输入接口28访问。存储设备32可以包括各种分布式或本地访问的数据存储介质中的任一种,如硬盘驱动器、蓝光光盘、数字通用盘(DVD)、光碟只读存储器(CD-ROM)、闪速存储器、易失性存储器或非易失性存储器、或用于存储已编码视频数据的任何其他合适的数字存储介质。在进一步的示例中,存储设备32可以对应于可以保持由源设备12生成的已编码视频数据的文件服务器或另一个中间存储设备。目标设备14可以经由流传输或下载从存储设备32访问所存储的视频数据。文件服务器可以是能够存储已编码视频数据并且将已编码视频数据传输到目标设备14的任何类型的计算机。示例性文件服务器包括web服务器(例如,用于网站)、文件传输协议(FTP)服务器、网络附加存储(NAS)设备、或本地磁盘驱动器。目标设备14可以通过任何标准数据连接来访问已编码视频数据,所述连接包括适于访问存储在文件服务器上的已编码视频数据的无线信道(例如,无线保真(Wi-Fi)连接)、有线连接(例如,数字用户线(DSL)、电缆调制解调器等)、或两者的组合。从存储设备32传输已编码视频数据可以是流式传输、下载传输、或两者的组合。

[0064] 如图1中所示,源设备12包括视频源18、视频编码器20和输出接口22。视频源18可以包括如视频捕获设备等源,例如摄像机、包含先前捕获的视频的视频档案、用于从视频内容提供方接收视频的视频馈送接口和/或用于生成计算机图形数据作为源视频的计算机图

形系统或这些源的组合。作为一个示例,如果视频源18是安全监控系统的摄像机,则源设备12和目标设备14可以形成拍照电话或视频电话。然而,本申请中描述的实施方式通常可以适用于视频编解码并且可以应用于无线和/或有线应用。

[0065] 捕获的、预先捕获的、或计算机生成的视频可以由视频编码器20进行编码。已编码视频数据可以经由源设备12的输出接口22直接传输到目标设备14。已编码视频数据也可以(或替代性地)存储到存储设备32上,以供目标设备14或其他设备以后访问,以进行解码和/或回放。输出接口22可以进一步包括调制解调器和/或发射器。

[0066] 目标设备14包括输入接口28、视频解码器30、以及显示设备34。输入接口28可以包括接收器和/或调制解调器,并且通过链路16接收已编码视频数据。通过链路16传送的或提供在存储设备32上的已编码视频数据可以包括由视频编码器20生成的各种语法元素,以供视频解码器30用于解码视频数据。这种语法元素可以被包括于在通信介质上传输的、存储在存储介质上的、或存储在文件服务器上的已编码视频数据内。

[0067] 在一些实施方式中,目标设备14可以包括显示设备34,所述显示设备可以是集成显示设备和被配置为与目标设备14通信的外部显示设备。显示设备34向用户显示经解码视频数据并且可以包括各种显示设备中的任一种,如液晶显示器(LCD)、等离子显示器、有机发光二极管(OLED)显示器、或另一种类型的显示设备。

[0068] 视频编码器20和视频解码器30可以根据专有或行业标准(如VVC、HEVC、MPEG-4第10部分、AVC、或这种标准的扩展)进行操作。应当理解,本申请不限于特定的视频编码/解码标准,并且可以适用于其他视频编码/解码标准。通常设想,源设备12的视频编码器20可以被配置为根据这些当前或未来标准中的任一种来对视频数据进行编码。类似地,通常还设想,目标设备14的视频解码器30可以被配置为根据这些当前或未来标准中的任一种来对视频数据进行解码。

[0069] 视频编码器20和视频解码器30各自可以实施为各种合适的编码器和/或解码器电路中的任一种,如一个或多个微处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、离散逻辑、软件、硬件、固件或其任何组合。当部分地在软件中实施时,电子设备可以将用于软件的指令存储在合适的非暂态计算机可读介质中并且使用一个或多个处理器在硬件中执行指令以执行本公开中公开的视频编码/解码操作。视频编码器20和视频解码器30中的每一个可以包括在一个或多个编码器或解码器中,所述一个或多个编码器或解码器中的任一个可以集成为相应设备中的组合编码器/解码器(CODEC)的一部分。

[0070] 图2是图示了根据本申请中描述的一些实施方式的示例性视频编码器20的框图。视频编码器20可以执行对视频帧内的视频块的帧内预测编解码和帧间预测编解码。帧内预测编解码依赖于空间预测来减少或去除给定视频帧或图片内的视频数据的空间冗余。帧间预测编解码依赖于时间预测以减少或去除视频序列的相邻视频帧或图片内的视频数据的时间冗余。应该注意的是,在视频编解码领域,术语“帧”可以用作术语“图像”或“图片”的同义词。

[0071] 如图2中所示,视频编码器20包括视频数据存储单元40、预测处理单元41、已解码图片缓冲器(DPB)64、加法器50、变换处理单元52、量化单元54、以及熵编码单元56。预测处理单元41进一步包括运动估计单元42、运动补偿单元44、分区单元45、帧内预测处理单元46和

帧内块复制(BC)单元48。在一些实施方式中,视频编码器20还包括用于视频块重建的逆量化单元58、逆变换处理单元60、以及加法器62。比如去块滤波器等环路滤波器63可以位于加法器62与DPB 64之间,以对块边界进行滤波,以从重建的视频中去除块效应伪像。除了去块滤波器之外,还可以使用另一个环路滤波器(如样点自适应偏移(SAO)滤波器和/或自适应环路滤波器(ALF))对加法器62的输出进行滤波。在一些示例中,可以省略环路滤波器,并且已解码视频块可以由加法器62直接提供给DPB 64。视频编码器20可以采用固定或可编程硬件单元的形式,或者可以在所图示的固定或可编程硬件单元中的一个或多个之中进行划分。

[0072] 视频数据存储单元40可以存储要由视频编码器20的部件编码的视频数据。例如,可以从如图1所示的视频源18获得视频数据存储单元40中的视频数据。DPB 64是存储参考视频数据(例如,参考帧或图片)以用于由视频编码器20对视频数据进行编码(例如,在帧内预测编解码模式或帧间预测编解码模式下)的缓冲器。视频数据存储单元40和DPB 64可以由多种存储器设备中的任何一种形成。在各个示例中,视频数据存储单元40可以与视频编码器20的其他部件一起在片上,或者相对于那些部件在片外。

[0073] 如图2中所示,在接收到视频数据之后,预测处理单元41内的分区单元45将视频数据分区为视频块。该分区还可以包括根据预定义的分割结构(如与视频数据相关联的四叉树(QT)结构)将视频帧分区为条带、瓦片(例如,视频块集)、或其他更大的编解码单元(CU)。视频帧是或可以被看作是具有样点值的二维样点阵列或矩阵。阵列中的样点也可以称为像素(pixel或pel)。阵列或图片的水平和垂直方向(或轴)上的许多样点定义了视频帧的尺寸和/或分辨率。例如,可以通过使用QT分区将视频帧划分为多个视频块。视频块再次是或可以被看作是具有样点值的二维样点阵列或矩阵,但是其维度小于视频帧。视频块的水平和垂直方向(或轴)上的许多样点定义了视频块的尺寸。可以通过例如迭代地使用QT分区、二叉树(BT)分区或三叉树(TT)分区或其任何组合将视频块进一步划分为一个或多个块分区或子块(这些子块可以再次形成块)。应该注意的是,本文使用的术语“块”或“视频块”可以是帧或图片的一部分,特别是矩形(正方形或非正方形)部分。例如,参照HEVC和VVC,块或视频块可以是或对应于编码树单元(CTU)、CU、预测单元(PU)或变换单元(TU),和/或可以是或对应于对应的块(例如,编码树块(CTB)、编码块(CB)、预测块(PB)或变换块(TB))和/或对应于子块。

[0074] 预测处理单元41可以基于误差结果(例如,编码率和失真水平)为当前视频块选择多个可能的预测编码模式之一,如多个帧内预测编码模式之一或多个帧间预测编码模式之一。预测处理单元41可以将所得的帧内预测编码块或帧间预测编码块提供给加法器50以生成残差块,并且提供给加法器62以重建已编码块以随后用作参考帧的一部分。预测处理单元41还将如运动矢量、帧内模式指示符、分区信息、以及其他这种语法信息等语法元素提供给熵编码单元56。

[0075] 为了为当前视频块选择适当的帧内预测编码模式,预测处理单元41内的帧内预测处理单元46可以相对于与要编码的当前块相同的帧中的一个或多个邻近块执行对当前视频块的帧内预测编码,以提供空间预测。预测处理单元41内的运动估计单元42和运动补偿单元44相对于一个或多个参考帧中的一个或多个预测块执行对当前视频块的帧间预测编码,以提供时间预测。视频编码器20可以执行多个编解码通道,例如,以便为视频数据的每

个块选择适当的编解码模式。

[0076] 在一些实施方式中,运动估计单元42根据视频帧序列内的预定模式通过生成运动矢量来确定当前视频帧的帧间预测模式,所述运动矢量指示当前视频帧内的视频块相对于参考视频帧内的预测块的位移。由运动估计单元42执行的运动估计是生成运动矢量的过程,所述过程估计了视频块的运动。运动矢量例如可以指示当前视频帧或图片内的视频块相对于参考帧内的预测块的位移,所述预测块是相对于在当前帧内编解码的当前块的。预定模式可以将序列中的视频帧指定为P帧或B帧。帧内BC单元48可以以与由运动估计单元42确定运动矢量以进行帧间预测的方式类似的方式确定用于进行帧内BC编码的矢量(例如,块矢量),或者可以利用运动估计单元42来确定块矢量。

[0077] 视频块的预测块可以是或可以对应于参考帧的块或参考块,所述参考帧被认为在像素差值方面与要编码的视频块紧密匹配,所述像素差值可以由绝对差值和(SAD)、平方差值和(SSD)或其他差值度量来确定。在一些实施方式中,视频编码器20可以计算存储在DPB 64中的参考帧的子整数像素位置的值。例如,视频编码器20可以插入参考帧的四分之一像素位置、八分之一像素位置、或其他分数像素位置的值。因此,运动估计单元42可以相对于全像素位置和分数像素位置执行运动搜索并且以分数像素精度输出运动矢量。

[0078] 运动估计单元42通过将视频块的位置与从第一参考帧列表(列表0)或第二参考帧列表(列表1)中选择的参考帧的预测块的位置进行比较来计算帧间预测编码帧中的视频块的运动矢量,所述列表中的每一个标识存储在DPB 64中的一个或多个参考帧。运动估计单元42将经计算的运动矢量发送到运动补偿单元44,并且然后发送到熵编码单元56。

[0079] 由运动补偿单元44执行的运动补偿可以涉及基于由运动估计单元42确定的运动矢量获取或生成预测块。在接收到当前视频块的运动矢量后,运动补偿单元44可以在参考帧列表中的一个中定位运动矢量所指向的预测块,从DPB 64取得预测块并且将预测块转发到加法器50。然后,加法器50通过从被编码的当前视频块的像素值中减去由运动补偿单元44提供的预测块的像素值来形成具有像素差值的残差视频块。形成残差视频块的像素差值可以包括亮度差分量或色度差分量或两者。运动补偿单元44还可以生成与视频帧的视频块相关联的语法元素,以供视频解码器30在对视频帧的视频块进行解码时使用。语法元素可以包括例如定义用于标识预测块的运动矢量的语法元素、指示预测模式的任何标志、或本文描述的任何其他语法信息。注意,运动估计单元42和运动补偿单元44可以是高度集成的,但是出于概念性目的而分别图示。

[0080] 在一些实施方式中,帧内BC单元48可以以与上文结合运动估计单元42和运动补偿单元44所描述的方式类似的方式来生成矢量并且获取预测块,但是其中预测块与被编码的当前块处于同一帧中,并且其中相对于运动矢量,所述矢量被称为块矢量。具体地,帧内BC单元48可以确定帧内预测模式以用于对当前块进行编码。在一些示例中,帧内BC单元48可以例如在单独的编码通道期间使用各种帧内预测模式对当前块进行编码,并且通过率失真分析来测试其性能。接下来,帧内BC单元48可以在各种经测试帧内预测模式中选择适当的帧内预测模式来使用并相应地生成帧内模式指示符。例如,帧内BC单元48可以使用针对各种测试的帧内预测模式的率失真分析来计算率失真值并且在测试的模式中选择具有最佳率失真特性的帧内预测模式作为要使用的适当的帧内预测模式。率失真分析通常确定已编码块与原始的未编码块(被编码以产生已编码块)之间的失真(或误差)量以及用于产生已

编码块的比特率(即,比特数)。帧内BC单元48可以根据各个已编码块的失真和速率来计算比值,以确定哪个帧内预测模式展现出块的最佳率失真值。

[0081] 在其他示例中,帧内BC单元48可以全部或部分地使用运动估计单元42和运动补偿单元44,以根据本文描述的实施方式执行用于帧内BC预测的这种功能。在任一种情况下,对于帧内块复制,预测块可以是就像像素差而言被认为与要编码的块紧密匹配的块,所述像素差可以由SAD、SSD或其他差值度量来确定,并且预测块的识别可以包括计算子整数像素位置的值。

[0082] 无论预测块是根据帧内预测来自同一帧还是根据帧间预测来自不同帧,视频编码器20都可以通过从被编解码的当前视频块的像素值中减去预测块的像素值从而形成像素差值来形成残差视频块。形成残差视频块的像素差值可以包括亮度分量差和色度分量差。

[0083] 如上文描述的,帧内预测处理单元46可以对当前视频块进行帧内预测,作为由运动估计单元42和运动补偿单元44执行的帧间预测、或者由帧内BC单元48执行的帧内块复制预测的替代方案。具体地,帧内预测处理单元46可以确定帧内预测模式以用于对当前块进行编码。为此,帧内预测处理单元46可以例如在单独的编码通道期间使用各种帧内预测模式对当前块进行编码,并且帧内预测处理单元46(或在一些示例中为模式选择单元)可以从经测试帧内预测模式中选择适当的帧内预测模式来使用。帧内预测处理单元46可以将指示块的所选帧内预测模式的信息提供给熵编码单元56。熵编码单元56可以对指示比特流中的所选帧内预测模式的信息进行编码。

[0084] 在预测处理单元41经由帧间预测或帧内预测确定当前视频块的预测块之后,加法器50通过从当前视频块中减去预测块来形成残差视频块。残差块中的残差视频数据可以被包括在一个或多个TU中,并且被提供给变换处理单元52。变换处理单元52使用如离散余弦变换(DCT)或概念上类似的变换等变换将残差视频数据变换为残差变换系数。

[0085] 变换处理单元52可以将所得的变换系数发送到量化单元54。量化单元54对变换系数进行量化以进一步降低比特率。量化过程还可以减小与系数中的一些或所有系数相关联的位深度。量化程度可以通过调整量化参数来修改。在一些示例中,量化单元54然后可以执行对包括经量化的变换系数的矩阵的扫描。替代性地,熵编码单元56可以执行扫描。

[0086] 量化之后,熵编码单元56使用例如上下文自适应可变长度编解码(CAVLC)、上下文自适应二进制算术编解码(CABAC)、基于语法的上下文自适应二进制算术编解码(SBAC)、概率区间分区熵(PIPE)编解码或其他熵编码方法或技术将量化变换系数熵编码为视频比特流。然后可以将已编码比特流传输到如图1所示的视频解码器30,或将其存档到如图1所示的存储设备32中,以供以后传输到视频解码器30或由所述视频解码器取得。熵编码单元56还可以对被编码的当前视频帧的运动矢量和其他语法元素进行熵编码。

[0087] 逆量化单元58和逆变换处理单元60分别应用逆量化和逆变换以在像素域中重建残差视频块,以生成用于预测其他视频块的参考块。如上所述,运动补偿单元44可以从DPB 64中存储的帧的一个或多个参考块中生成经运动补偿的预测块。运动补偿单元44还可以将一个或多个内插滤波器应用于预测块以计算用于运动估计中的子整数像素值。

[0088] 加法器62将重建的残差块添加到由运动补偿单元44产生的经运动补偿的预测块,以产生参考块用于存储在DPB 64中。参考块然后可以由帧内BC单元48、运动估计单元42和运动补偿单元44用作预测块,以对后续视频帧中的另一个视频块进行帧间预测。

[0089] 图3是图示了根据本申请的一些实施方式的示例性视频解码器30的框图。视频解码器30包括视频数据存储单元79、熵解码单元80、预测处理单元81、逆量化单元86、逆变换处理单元88、加法器90、以及DPB 92。预测处理单元81进一步包括运动补偿单元82、帧内预测单元84和帧内BC单元85。视频解码器30可以执行与上文结合图2关于视频编码器20所描述的编码过程相反的解码过程。例如,运动补偿单元82可以基于从熵解码单元80接收到的运动矢量生成预测数据,而帧内预测单元84可以基于从熵解码单元80接收到的帧内预测模式指示符生成预测数据。

[0090] 在一些示例中,视频解码器30的单元可以被指派执行本申请的实施方式。同样,在一些示例中,本公开的实施方式可以在视频解码器30的一个或多个单元之间进行划分。例如,帧内BC单元85可以单独或与视频解码器30的其他单元(如运动补偿单元82、帧内预测单元84、以及熵解码单元80)组合地执行本申请的实施方式。在一些示例中,视频解码器30可以不包括帧内BC单元85,并且帧内BC单元85的功能可以由预测处理单元81的其他部件(如运动补偿单元82)执行。

[0091] 视频数据存储单元79可以存储要由视频解码器30的其他部件解码的视频数据,如已编码视频比特流。例如,可以经由对视频数据进行有线或无线网络传送或者通过访问物理数据存储介质(例如,闪速驱动器或硬盘)从存储设备32、本地视频源(如相机)获得存储在视频数据存储单元79中的视频数据。视频数据存储单元79可以包括存储来自已编码视频比特流的已编码视频数据的编码图片缓冲器(CPB)。视频解码器30的DPB 92存储参考视频数据,以用于由视频解码器30对视频数据进行解码(例如,在帧内预测编解码模式或帧间预测编解码模式下)。视频数据存储单元79和DPB 92可以由多种存储器设备中的任一种形成,如动态随机存取存储器(DRAM),包括同步DRAM(SDRAM)、磁阻式RAM(MRAM)、电阻式RAM(RRAM)、或其他类型的存储器设备。为了说明目的,在图3中将视频数据存储单元79和DPB 92描绘为视频解码器30的两个不同部件。但是对于本领域技术人员将显而易见的是,视频数据存储单元79和DPB 92可以由相同的存储器设备或单独的存储器设备提供。在一些示例中,视频数据存储单元79可以与视频解码器30的其他部件一起在片上,或者相对于那些部件在片外。

[0092] 在解码过程期间,视频解码器30接收表示已编码视频帧的视频块的已编码视频比特流和相关联的语法元素。视频解码器30可以在视频帧级和/或视频块级接收语法元素。视频解码器30的熵解码单元80对比特流进行熵解码以生成经量化的系数、运动矢量或帧内预测模式指示符、以及其他语法元素。熵解码单元80然后将运动矢量或帧内预测模式指示符和其他语法元素转发到预测处理单元81。

[0093] 当视频帧被编码为帧内预测编码(I)帧或用于其他类型的帧中的帧内编码预测块时,预测处理单元81的帧内预测单元84可以基于信号传输的帧内预测模式和来自当前帧的先前经解码块的参考数据来生成当前视频帧的视频块的预测数据。

[0094] 当视频帧被编解码为帧间预测编解码(即,B或P)帧时,预测处理单元81的运动补偿单元82基于从熵解码单元80接收到的运动矢量和其他语法元素产生当前视频帧的视频块的一个或多个预测块。每个预测块可以从参考帧列表之一内的参考帧产生。视频解码器30可以基于存储在DPB 92中的参考帧使用默认构建技术构建参考帧列表:列表0和列表1。

[0095] 在一些示例中,当根据本文描述的帧内BC模式对视频块进行编解码时,预测处理单元81的帧内BC单元85基于从熵解码单元80接收到的块矢量和和其他语法元素,为当前视频

块产生预测块。预测块可以处于与由视频编码器20定义的当前视频块相同的图片的重建区域内。

[0096] 运动补偿单元82和/或帧内BC单元85通过解析运动矢量和其他语法元素来确定当前视频帧的视频块的预测信息,并且然后使用预测信息来产生被解码的当前视频块的预测块。例如,运动补偿单元82使用接收到的语法元素中的一些来确定用于对视频帧的视频块进行编码的预测模式(例如,帧内预测或帧间预测)、帧间预测帧类型(例如,B或P)、帧的参考帧列表中的一个或多个参考帧列表的构建信息、帧的每个帧间预测已编码视频块的运动矢量、帧的每个帧间预测编码视频块的帧间预测状态、以及用于对当前视频帧中的视频块进行解码的其他信息。

[0097] 类似地,帧内BC单元85可以使用接收到的语法元素中的一些(例如,标志)来确定当前视频块是使用以下各项预测的:帧内BC模式、关于帧的视频块处于重建的区域内并且应存储在DPB 92中的构建信息、帧的每个帧内BC预测视频块的块矢量、帧的每个帧内BC预测视频块的帧内BC预测状态、以及用于对当前视频帧中的视频块进行解码的其他信息。

[0098] 运动补偿单元82还可以如由视频编码器20在对视频块进行编码期间使用的那样使用内插滤波器来执行内插以计算参考块的子整数像素的内插值。在这种情况下,运动补偿单元82可以从接收到的语法元素确定由视频编码器20使用的内插滤波器并且使用内插滤波器来产生预测块。

[0099] 逆量化单元86使用由视频编码器20针对视频帧中的每个视频块计算的用于确定量化程度的相同的量化参数,对在比特流中提供的并且由熵解码单元80进行熵解码的经量化的变换系数进行逆量化。逆变换处理单元88将逆变换(例如,逆DCT、逆整数变换、或概念上类似的逆变换过程)应用于变换系数,以便在像素域中重建残差块。

[0100] 在运动补偿单元82或帧内BC单元85基于矢量和和其他语法元素生成当前视频块的预测块之后,加法器90通过对来自逆变换处理单元88的残差块以及由运动补偿单元82和帧内BC单元85生成的对应预测块求和来重建当前视频块的经解码视频块。环路滤波器91(如去块滤波器、SAO滤波器和/或ALF)可以定位于加法器90与DPB 92之间,以进一步处理经解码视频块。在一些示例中,可以省略环路滤波器91,已解码视频块可以由加法器90直接提供给DPB 92。然后将给定帧中的已解码视频块存储在DPB 92中,所述DPB存储用于对接下来的视频块进行后续运动补偿的参考帧。DPB 92或与DPB 92分开的存储器设备还可以存储已解码视频以供稍后呈现在如图1的显示设备34等显示设备上。

[0101] 在典型的视频编解码过程中,视频序列通常包括经排序的帧或图片集。每个帧可以包括三个样点阵列,分别表示为SL、SCb和SCr。SL是亮度样点的二维阵列。SCb是Cb色度样点的二维阵列。SCr是Cr色度样点的二维阵列。在其他实例中,帧可以是单色的,并且因此仅包括亮度样点的一个二维阵列。

[0102] 如图4A所示,视频编码器20(或更具体地,分区单元45)通过首先将帧分区为一组CTU来生成帧的已编码表示。视频帧可以包括从左到右以及从上到下以光栅扫描顺序连续排序的整数个CTU。每个CTU是最大的逻辑编码单元,并且由视频编码器20在序列参数集中用信号传输CTU的宽度和高度,使得视频序列中的所有CTU具有相同的尺寸,即 128×128 、 64×64 、 32×32 和 16×16 中的一个。但是应当注意,本申请不必限于特定的尺寸。如图4B所示,每个CTU可以包括亮度样点的一个CTB、色度样点的两个对应的编码树块以及用于对编码树

块的样点进行编解码的语法元素。语法元素描述像素的编解码块的不同类型的单元的属性以及可以如何在视频解码器30处重建视频序列,所述语法元素包括帧间预测或帧内预测、帧内预测模式、运动矢量和其他参数。在单色图片或具有三个单独的色彩平面的图片中,CTU可以包括单个编码树块和用于对编码树块的样点进行编解码的语法元素。编码树块可以是 $N \times N$ 样点块。

[0103] 为了实现更好的性能,视频编码器20可以对CTU的编码树块递归地执行树分区(如二叉树分区、三叉树分区、四叉树分区或其组合),并且将CTU划分为较小的CU。如图4C描绘的,首先将 64×64 CTU 400划分为四个较小的CU,每个CU的块尺寸为 32×32 。在四个较小的CU中,CU 410和CU 420按块尺寸各自划分为四个 16×16 的CU。两个 16×16 CU 430和440按块尺寸各自进一步划分为四个 8×8 的CU。图4D描绘了图示了如图4C中所描绘的CTU 400的分区过程的最终结果的四叉树数据结构,四叉树的每个叶节点对应于相应尺寸在 32×32 至 8×8 范围内的一个CU。类似于图4B描绘的CTU,每个CU可以包括亮度样点的CB和相同尺寸的帧的色度样点的两个对应的编码块,以及用于对编码块的样点进行编解码的语法元素。在单色图片或具有三个单独的色彩平面的图片中,CU可以包括单个编码块和用于对编码块的样点进行编解码的语法结构。应当注意,图4C和图4D中描绘的四叉树分区仅用于说明目的,并且可以将一个CTU分割成多个CU以适应基于四叉树/三叉树/二叉树分区的不同的局部特性。在多类型树结构中,一个CTU被四叉树结构分区,并且每个四叉树叶CU可以进一步被二叉树结构或三叉树结构分区。如图4E所示,具有宽度W和高度H的编码块存在五种可能的分区类型,即,四叉分区、水平二叉分区、垂直二叉分区、水平三叉分区和垂直三叉分区。

[0104] 在一些实施方式中,视频编码器20可以进一步将CU的编码块分区为一个或多个 $M \times N$ PB。PB是应用相同预测(帧间或帧内)的矩形(正方形或非正方形)样点块。CU的PU可以包括亮度样点的PB、色度样点的两个对应的PB以及用于对PB进行预测的语法元素。在单色图片或具有三个单独的色彩平面的图片中,PU可以包括单个PB和用于对PB进行预测的语法结构。视频编码器20可以生成CU的每个PU的亮度、Cb和Cr PB的预测亮度、Cb和Cr块。

[0105] 视频编码器20可以使用帧内预测或帧间预测来生成PU的预测块。如果视频编码器20使用帧内预测来生成PU的预测块,则视频编码器20可以基于与PU相关联的帧的已解码样点来生成PU的预测块。如果视频编码器20使用帧间预测来生成PU的预测块,则视频编码器20可以基于除与PU相关联的帧之外的一个或多个帧的已解码样点来生成PU的预测块。

[0106] 在视频编码器20生成CU的一个或多个PU的预测亮度、Cb和Cr块之后,视频编码器20可以通过从其原始亮度编码块中减去CU的预测亮度块来生成CU的亮度残差块,使得CU的亮度残差块中的每个样点指示CU的预测亮度块之一中的亮度样点与CU的原始亮度编码块中的对应样点之间的差。类似地,视频编码器20可以分别生成CU的Cb残差块和Cr残差块,使得CU的Cb残差块中的每个样点指示CU的预测Cb块之一中的Cb样点与CU的原始Cb编码块中的对应样点之间的差,并且CU的Cr残差块中的每个样点可以指示CU的预测Cr块之一中的Cr样点与CU的原始Cr编码块中的对应样点之间的差。

[0107] 此外,如图4C所图示的,视频编码器20可以使用四叉树分区来将CU的亮度、Cb和Cr残差块分别分解为一个或多个亮度、Cb和Cr变换块。变换块是样点的应用相同变换的矩形(正方形或非正方形)块。CU的TU可以包括亮度样点的变换块、色度样点的两个对应的变换块以及用于对变换块样点进行变换的语法元素。因此,CU的每个TU可以与亮度变换块、Cb变

换块和Cr变换块相关联。在一些示例中,与TU相关联的亮度变换块可以是CU的亮度残差块的子块。Cb变换块可以是CU的Cb残差块的子块。Cr变换块可以是CU的Cr残差块的子块。在单色图片或具有三个单独的色彩平面的图片中,TU可以包括单个变换块和用于对变换块的样点进行变换的语法结构。

[0108] 视频编码器20可以将一个或多个变换应用于TU的亮度变换块以生成TU的亮度系数块。系数块可以是变换系数的二维阵列。变换系数可以是标量。视频编码器20可以将一个或多个变换应用于TU的Cb变换块以生成TU的Cb系数块。视频编码器20可以将一个或多个变换应用于TU的Cr变换块以生成TU的Cr系数块。

[0109] 在生成系数块(例如,亮度系数块、Cb系数块或Cr系数块)之后,视频编码器20可以对系数块进行量化。量化通常是指对变换系数进行量化以可能减少用于表示变换系数的数据量从而提供进一步压缩的过程。在视频编码器20对系数块进行量化之后,视频编码器20可以对指示量化变换系数的语法元素进行熵编码。例如,视频编码器20可以对指示量化变换系数的语法元素执行CABAC。最终,视频编码器20可以输出包括形成编解码帧和相关联数据的表示的比特序列的比特流,所述比特流被保存在存储设备32中或被传输到目标设备14。

[0110] 在接收到由视频编码器20生成的比特流之后,视频解码器30可以解析所述比特流以从所述比特流中获得语法元素。视频解码器30可以至少部分地基于从比特流获得的语法元素来重建视频数据的帧。重建视频数据的过程通常与由视频编码器20执行的编码过程是相反的。例如,视频解码器30可以对与当前CU的TU相关联的系数块执行逆变换以重建与当前CU的TU相关联的残差块。视频解码器30还通过将当前CU的PU的预测块的样点添加到当前CU的TU的变换块的对应样点来重建当前CU的编码块。在重建帧的每个CU的编码块之后,视频解码器30可以重建帧。

[0111] 如上所述,视频编解码主要使用两种模式,即,帧内预测(intra-frame prediction)(或帧内预测(intra-prediction))和帧间预测(inter-frame prediction)(或帧间预测(inter-prediction))来实现视频压缩。应注意的是,IBC可以被认为是帧内预测或第三模式。在这两种模式之间,由于使用用于从参考视频块中预测当前视频块的运动矢量,因此帧间预测比帧内预测对编码效率的贡献更大。

[0112] 但是随着不断改进的视频数据捕获技术和用于保留视频数据中的细节的更精细的视频块尺寸,表示当前帧的运动矢量所需的数据量也大幅度增加。克服这个挑战的一种方式受益于以下事实:不仅空间域和时间域中的一组邻近CU具有用于预测目的类似视频数据,而且这些邻近CU之间的运动矢量也类似。因此,可以通过探索CU的空间和时间相关性,将空间上邻近的CU和/或时间上并置的CU的运动信息用作当前CU的运动信息(例如,运动矢量)的近似值,所述近似值也被称为当前CU的“运动矢量预测值”(MVP)。

[0113] 代替将由如上文结合图2描述的运动估计单元42确定的当前CU的实际运动矢量编码为视频比特流,从当前CU的实际运动矢量中减去当前CU的运动矢量预测值,以产生当前CU的运动矢量差(MVD)。这样做,不需要将由运动估计单元42针对帧的每个CU确定的运动矢量编码为视频比特流,并且可以显著减少用于表示视频比特流中的运动信息的数据量。

[0114] 像在编码块的帧间预测期间在参考帧中选择预测块的过程一样,需要由视频编码器20和视频解码器30两者采用一组规则以用于使用与当前CU的在空间上邻近的CU和/或在

时间上并置的CU相关联的那些潜在候选运动矢量来构建当前CU的运动矢量候选列表(也称为“合并列表”),并且然后从运动矢量候选列表选择一个成员作为当前CU的运动矢量预测值。这样做,不需要将运动矢量候选列表本身从视频编码器20传输到视频解码器30,并且运动矢量候选列表内的所选运动矢量预测值的索引足以使视频编码器20和视频解码器30使用运动矢量候选列表内相同的运动矢量预测值来对当前CU进行编码和解码。

[0115] 对于每个帧间预测的CU,运动参数由以下各项构成:运动矢量、参考图片索引、和参考图片列表使用索引、以及用于帧间预测的样点生成的VVC的新编解码特征所需的附加信息。运动参数可以以显式或隐式的方式用信号传输。当用跳过模式对CU进行编解码时,所述CU与一个PU相关联,并且不具有显著的残差系数,不具有已编解码运动矢量 Δ 或参考图片索引。指定合并模式,从而从邻近CU(包括空间和时间候选)和VVC中引入的附加调度获得当前CU的运动参数。可以将合并模式应用于任何帧间预测的CU,而不仅仅是用于跳过模式。合并模式的替代方案是运动参数的显式传输,其中,运动矢量、每个参考图片列表的对应参考图片索引和参考图片列表使用标志以及其他所需信息按每个CU显式地用信号传输。

[0116] 此外,ITU-T VCEG(Q6/16)和ISO/IEC MPEG(JTC 1/SC 29/WG 11)正在研究未来视频编解码技术标准化的潜在需求,所述技术的压缩能力大大超过当前VVC标准的压缩能力。这种未来的标准化行动可以采取VVC的(多种)附加扩展或全新标准的形式。被称为联合视频探索小组(JVET)的小组正在联合协作努力中共同开展这项探索活动评估其专家在此领域中提出的压缩技术设计。第一个探索实验(EE)是在2021年1月6日至15日期间的JVET会议上建立的,并且该探索软件模型被命名为增强压缩模型(ECM),并且ECM版本2(ECM2)于2021年8月发布。ECM2中新开发的帧间预测方案将在以下章节中进行详细说明。

[0117] 以下章节提供了关于VVC和正在开发的ECM模型中指定的那些帧间预测方法的细节。

[0118] VVC中的扩展合并预测

[0119] 在VVC中,合并候选列表是通过按顺序包括以下五种类型的候选来构建的:

[0120] 1).来自空间相邻CU的空间MVP

[0121] 2).来自同位CU的时间MVP

[0122] 3).来自FIFO表的基于历史的MVP

[0123] 4).成对平均MVP

[0124] 5).零MV。

[0125] 在序列参数集标头中用信号传输合并列表的尺寸,并且合并列表的最大允许尺寸为6。对于合并模式下的每个CU编解码,使用截断的一元二值化(TU)对最佳合并候选的索引进行编码。合并索引的第一二进制位利用上下文进行编解码,并且其他二进制位使用旁路编解码。

[0126] 本节提供了每类合并候选的推导过程。正如在HEVC中所做的那样,VVC还支持在一定尺寸的区域对所有CU的合并候选列表进行并行推导。

[0127] VVC中的空间候选推导

[0128] 除了前两个合并候选的位置被交换之外,VVC中空间合并候选的推导与HEVC中的推导相同。在位于图5中描绘的位置的候选中选择最多四个合并候选。推导顺序为 B_0 、 A_0 、 B_1 、 A_1 和 B_2 。只有当位置 B_0 、 A_0 、 B_1 、 A_1 中的一个或多于一个CU不可用(例如,因为它属于另一个条

带或瓦片)或被帧内编码时,才考虑位置 B_2 。在添加了位置 A_1 处的候选之后,对剩余候选的添加进行冗余检查,这确保了将具有相同运动信息的候选从列表中排除,从而提高了编解码效率。为了降低计算复杂度,并非所有可能的候选对都被考虑在所提及的冗余检查中。相反,仅考虑与图6中的箭头链接的对,并且仅在用于冗余检查的对应候选不具有相同运动信息的情况下才将候选添加到列表中。

[0129] VVC中的时间候选推导

[0130] 在该步骤中,只有一个候选被添加到列表中。特别地,在该时间合并候选的推导中,基于属于同位参考图片的同位CU来推导缩放运动矢量。用于推导同位CU的参考图片列表和参考索引在条带标头中显式地用信号传输。如图7中的虚线所示,获得时间合并候选的缩放运动矢量,所述缩放运动矢量是使用POC距离 t_b 和 t_d 从同位CU的运动矢量缩放的,其中, t_b 定义为当前图片的参考图片与当前图片之间的POC差,并且 t_d 定义为同位图片的参考图片与同位图片之间的POC差。时间合并候选的参考图片索引被设置为等于零。

[0131] 如图8所描绘的,在候选 C_0 与 C_1 之间选择时间候选的位置。如果位置 C_0 处的CU不可用,是帧内编码的或在CTU的当前行之外,则使用位置 C_1 。否则,位置 C_0 用于推导时间合并候选。

[0132] VVC中的基于历史的合并候选推导

[0133] 将基于历史的MVP (HMVP) 合并候选添加到合并列表中处于空间MVP和TMVP之后。在这种方法中,将先前编码块的运动信息存储在表中,并用作当前CU的MVP。在编码/解码过程期间保持具有多个HMVP候选的表。当遇到新的CTU行时,将所述表重置(清空)。每当存在非子块帧间编码CU时,将相关联的运动信息作为新的HMVP候选添加到表的最后一个条目。

[0134] 将HMVP表尺寸 S 设置为6,这表示至多可以向表中添加5个基于历史的MVP (HMVP) 候选。当将新的运动候选插入到表中时,使用受约束的先进先出(FIFO)规则,其中,首先应用冗余检查来查找表中是否存在相同的HMVP。如果被查找到,则从表中移除相同的HMVP,并且然后向前移动所有HMVP候选,并将相同的HMVP插入到表的最后一个条目。

[0135] HMVP候选可以用于合并候选列表构建过程。按顺序检查表中最新的几个HMVP候选,并将其插入候选列表中处于TMVP候选之后。对HMVP候选应用冗余检查以获得空间或时间合并候选。

[0136] 为了减少冗余检查操作的数量,引入了以下简化:

[0137] 1. 表中的最后两个条目分别相对于 A_1 和 B_1 空间候选进行冗余检查。

[0138] 2. 一旦可用合并候选的总数达到最大允许合并候选减1,则终止来自HMVP的合并候选列表构建过程。

[0139] VVC中的成对平均合并候选推导

[0140] 通过使用前两个合并候选对现有合并候选列表中的预定义候选对进行平均,生成成对平均候选。相应地,第一合并候选被定义为 p_0Cand 并且第二合并候选可以被定义为 p_1Cand 。对于每个参考列表,根据 p_0Cand 和 p_1Cand 的运动矢量的可用性单独地计算平均运动矢量。如果在一个列表中两个运动矢量都可用,则即使在这两个运动矢量指向不同的参考图片时,也会对这两个运动矢量进行平均,并且其参考图片被设置为 p_0Cand 中的一个;如果只有一个运动矢量可用,则直接使用所述运动矢量;如果没有可用的运动矢量,则使该列表保持无效。此外,如果 p_0Cand 和 p_1Cand 的半像素内插滤波器索引不同,则将其设置为0。

[0141] 当添加成对平均合并候选之后合并列表未滿时,最后插入零MVP,直到遇到最大合并候选数量。

[0142] VVC中的高精度(1/16像素)运动补偿和运动矢量存储

[0143] VVC将MV精度提高到1/16亮度样点,以提高慢动作视频的预测效率。这种更高的运动准确性对于具有局部变化和非平移运动的视频内容特别有帮助,如在仿射模式的情况下。对于更高MV准确性的分数位置样点生成,将HEVC的8抽头亮度内插滤波器和4抽头色度内插滤波器针对亮度扩展到16个相位并且针对色度扩展到32个相位。该扩展滤波器集应用于除仿射模式下的CU之外的帧间编码CU的MC过程中。对于仿射模式,使用一组具有16个相位的6抽头亮度内插滤波器来降低计算复杂度以及节省存储器带宽。

[0144] 在VVC中,非仿射CU的显式地用信号传输的运动矢量的最高精度是四分之一亮度样点。在如仿射模式等一些帧间预测模式下,可以以1/16亮度样点精度用信号传输运动矢量。在具有隐式地推断的MV的所有帧间编码CU中,MV是以1/16亮度样点精度推导的,并且运动补偿预测是以1/16样点精度执行的。在内部运动场存储方面,所有运动矢量都以1/16亮度样点精度存储。

[0145] 对于由TMVP和SbTMVP使用的时间运动场存储,与HEVC中的 16×16 尺寸粒度相比,运动场压缩是以 8×8 尺寸粒度执行的。

[0146] VVC中的利用MVD的合并模式(MMVD)

[0147] 除了其中将隐式推导的运动信息直接用于当前CU的预测样点生成的合并模式之外,VVC中还引入了利用运动矢量差的合并模式(MMVD)。在发送常规合并标志之后立即用信号传输MMVD标志以指定MMVD模式是否用于CU。

[0148] 在MMVD下,在选择合并候选后,通过用信号传输的MVD信息对所选合并候选进一步细化。进一步的信息包括合并候选标志、用于指定运动幅值的索引和用于指示运动方向的索引。在MMVD模式下,为合并列表中的前两个候选选择一个用作MV基础。用信号传输mmvd候选标志以指定在第一合并候选与第二合并候选之间使用哪一个。

[0149] 距离索引指定运动幅值信息并且指示从起点开始的预定义偏移。如图9A和图9B所示,偏移被添加到起始MV的水平分量或垂直分量。表1中指定了距离索引与预定义偏移的关系。

[0150] 表1:距离索引与预定义偏移的关系

距离 IDX	0	1	2	3	4	5	6	7
偏移(以亮度样点为单位)	1/4	1/2	1	2	4	8	16	32

[0152] 方向索引表示MVD相对于起点的方向。方向索引可以表示如表2所示的这四个方向。应当注意,MVD符号的含义可以根据起始MV的信息而变化。当起始MV是单向预测MV或双向预测MV且两个列表都指向当前图片的同一侧时(即两个参考的POC都大于当前图片的POC,或都小于当前图片的POC),表2中的符号指定添加到起始MV的MV偏移的符号。当起始MV是双向预测MV且两个MV指向当前图片的不同侧(即一个参考的POC大于当前图片的POC并且另一个参考的POC小于当前图片的POC)并且列表0中的POC的差大于列表1中的差时,表2中

的符号指定添加到起始MV的列表0MV分量的MV偏移的符号,并且列表1MV的符号具有相反值。否则,如果列表1中的POC的差大于列表0中的差,则表2中的符号指定添加到起始MV的列表1MV分量的MV偏移的符号,并且列表0MV的符号具有相反值。

[0153] MVD根据在每个方向上POC的差进行缩放。如果两个列表中POC的差相同,则不需要缩放。否则,如果列表0中的POC的差大于列表1中的差,则通过将L0的POC差定义为td并且将L1的POC差定义为tb来缩放列表1的MVD,如图7中所描述的。如果L1的POC差大于L0的POC差,则以相同的方式缩放列表0的MVD。如果起始MV是单向预测的,则将MVD添加到可用MV。

[0154] 表2:由方向索引指定的MV偏移的符号

方向IDX	00	01	10	11
x轴	+	-	不适用	不适用
y轴	不适用	不适用	+	-

[0156] VVC中的对称MVD编解码

[0157] 在VVC中,除了标准的单向预测和双向预测模式MVD信令外,还应用了用于双向预测MVD信令的对称MVD模式。在对称MVD模式下,包括列表0和列表1两者的参考图片索引以及列表1的MVD的运动信息不是用信号传输的,而是推导的。

[0158] 对称MVD模式的解码过程如下:

[0159] 1. 在条带级下,变量BiDirPredFlag、RefIdxSymL0和RefIdxSymL1推导如下:

[0160] -如果mvd_l1_zero_flag为1,则将BiDirPredFlag设置为等于0。

[0161] -否则,如果列表0中的最近参考图片和列表1中的最近参考图片形成前向和后向参考图片对或后向和前向参考图片对,则将BiDirPredFlag设置为1,并且列表0和列表1参考图片都是短期参考图片。否则,将BiDirPredFlag设置为0。

[0162] 2. 在CU级下,如果CU是双向预测编解码的并且BiDirPredFlag等于1,则显式地用信号传输指示是否使用对称模式的对称模式标志。

[0163] 当对称模式标志为真时,仅显式地用信号传输mvp_10_flag、mvp_11_flag和MVD0。将列表0和列表1的参考索引设置为分别等于所述参考图片对。将MVD1设置为等于(-MVD0)。最终的运动矢量如以下公式所示。

$$[0164] \quad \begin{cases} (m vx_0, m vy_0) = (m vpx_0 + m vdx_0, m vpy_0 + m vdy_0) \\ (m vx_1, m vy_1) = (m vpx_1 - m vdx_0, m vpy_1 - m vdy_0) \end{cases} \quad (1)$$

[0165] 在编码器中,对称MVD运动估计从初始MV评估开始。一组初始MV候选包括从单向预测搜索获得的MV、从双向预测搜索获得的MV和来自AMVP列表的MV。选择具有最低率失真成本的一个MV作为对称MVD运动搜索的初始MV。

[0166] VVC中的仿射运动补偿预测

[0167] 在HEVC中,仅将平移运动模型应用于运动补偿预测(MCP)。然而在现实世界中,存在很多种运动,例如,放大/缩小、旋转、透视运动和其他不规则运动。在VVC中,应用了基于块的仿射变换运动补偿预测。如图10A和图10B所示,块的仿射运动场由如图10A所示的两个控制点运动矢量(4参数)或如图10B所示的三个控制点运动矢量(6参数)的运动信息来描述。

[0168] 对于4参数仿射运动模型,推导块中样点位置(x,y)处的运动矢量为:

$$[0169] \quad \begin{cases} mv_x = \frac{mv_{1x} - mv_{0x}}{W} x + \frac{mv_{0y} - mv_{1y}}{W} y + mv_{0x} \\ mv_y = \frac{mv_{1y} - mv_{0y}}{W} x + \frac{mv_{1x} - mv_{0x}}{W} y + mv_{0y} \end{cases} \quad (2)$$

[0170] 对于6参数仿射运动模型,推导块中样点位置(x,y)处的运动矢量为:

$$[0171] \quad \begin{cases} mv_x = \frac{mv_{1x} - mv_{0x}}{W} x + \frac{mv_{2x} - mv_{0x}}{H} y + mv_{0x} \\ mv_y = \frac{mv_{1y} - mv_{0y}}{W} x + \frac{mv_{2y} - mv_{0y}}{H} y + mv_{0y} \end{cases} \quad (3)$$

[0172] 其中,(mv_{0x}, mv_{0y})是左上角控制点的运动矢量,(mv_{1x}, mv_{1y})是右上角控制点的运动矢量,并且(mv_{2x}, mv_{2y})是左下角控制点的运动矢量。

[0173] 为了简化运动补偿预测,应用了基于块的仿射变换预测。如图11所示,为了推导每个 4×4 亮度子块的运动矢量,根据上文的方程计算每个子块的中心样点的运动矢量,并且四舍五入到1/16分数精度。然后,应用运动补偿内插滤波器以使用推导的运动矢量生成每个子块的预测。也将色度分量的子块尺寸设置为 4×4 。 4×4 色度子块的MV被计算为同位 8×8 亮度区域中的左上和右下亮度子块的MV的平均值。

[0174] 与平移运动帧间预测类似,也存在两种仿射运动帧间预测模式:仿射合并模式和仿射AMVP模式。

[0175] 仿射合并预测

[0176] AF_MERGE模式可以应用于宽度和高度两者都大于或等于8的CU。在该模式下,基于空间邻近CU的运动信息生成当前CU的CPMV。至多可以有五个CPMVP候选,并且用信号传输索引来指示要用于当前CU的一个CPMVP候选。以下三种类型的CPVM候选用于形成仿射合并候选列表:

[0177] -从相邻CU的CPMV外推出的继承的仿射合并候选

[0178] -使用相邻CU的平移MV推导的构建的仿射合并候选CPMVP

[0179] -零MV

[0180] 在VVC中,存在最多两个从邻近块的仿射运动模型推导的继承的仿射候选,一个来自左侧邻近CU并且一个来自上方邻近CU。图12中示出了候选块。对于左边的预测值,扫描顺序是A0->A1,对于上面的预测值,扫描顺序是B0->B1->B2。仅选择每一侧的第一个继承的候选。在两个继承的候选之间不执行剪除检查。当邻近仿射CU被识别时,其控制点运动矢量用于在当前CU的仿射合并列表中推导CPMVP候选。如果相邻左下块A以仿射模式编解码,则获得包含块A的CU的左上角、右上角和左下角的运动矢量 v_2 、 v_3 和 v_4 。当使用4参数仿射模型对块A进行编解码时,根据 v_2 和 v_3 计算当前CU的两个CPMV。在使用6参数仿射模型对块A进行编解码的情况下,根据 v_2 、 v_3 和 v_4 计算当前CU的三个CPMV。

[0181] 构建的仿射候选意指候选是通过组合每个控制点的邻近平移运动信息构建的。控制点的运动信息是从图13所示的指定空间邻近块和时间邻近块中推导的。CPMV $_k$ ($k=1, 2, 3, 4$)表示第k个控制点。对于CPMV $_1$,检查B2->B3->A2块,并且使用第一可用块的MV。对于CPMV $_2$,检查B1->B0块,并且对于CPMV $_3$,检查A1->A0块。如果TMVP可用,则将其用作CPMV $_4$ 。

[0182] 在获得四个控制点的MV后,基于这些运动信息构建仿射合并候选。以下控制点MV的组合用于依次构建:

[0183] $\{CPMV_1, CPMV_2, CPMV_3\}$ 、 $\{CPMV_1, CPMV_2, CPMV_4\}$ 、 $\{CPMV_1, CPMV_3, CPMV_4\}$ 、 $\{CPMV_2, CPMV_3,$

$\{CPMV_4\}$ 、 $\{CPMV_1, CPMV_2\}$ 、 $\{CPMV_1, CPMV_3\}$

[0184] 3个CPMV的组合构建6参数仿射合并候选,并且2个CPMV的组合构建4参数仿射合并候选。为了避免运动缩放过程,如果控制点的参考索引不同,则丢弃相关的控制点MV组合。

[0185] 在检查了继承的仿射合并候选和构建的仿射合并候选后,如果列表仍不满,则将零MV插入到列表的最后。

[0186] VVC中的仿射AMVP预测

[0187] 仿射AMVP模式可以应用于宽度和高度两者都大于或等于16的CU。在比特流中用信号传输CU级的仿射标志以指示是否使用仿射AMVP模式,并且然后用信号传输另一个标志以指示是4参数仿射还是6参数仿射。在该模式下,在比特流中用信号传输当前CU的CPMV与其预测值CPMVP的差。仿射AMVP候选列表尺寸为2,并且其是通过按顺序使用以下四种类型的CPVM候选来生成的:

[0188] -从相邻CU的CPMV外推出的继承的仿射AMVP候选

[0189] -使用相邻CU的平移MV推导的构建的仿射AMVP候选CPMVP

[0190] -来自邻近CU的平移MV

[0191] -零MV

[0192] 继承的仿射AMVP候选的检查顺序与继承的仿射合并候选的检查顺序相同。唯一的区别在于,对于AMVP候选,仅考虑具有与当前块中的参考图片相同的参考图片的仿射CU。当将继承的仿射运动预测值插入到候选列表中时,不应用剪除过程。

[0193] 构建的AMVP候选是从图13所示的指定空间邻近块中推导的。使用与在仿射合并候选构建中的检查顺序相同的检查顺序。另外,还检查邻近块的参考图片索引。使用检查顺序中的第一块,所述第一块被帧间编解码并且具有与当前CU中的参考图片相同的参考图片。仅存在一个,当使用4参数仿射模式对当前CU进行编解码并且 mv_0 和 mv_1 都可用时,将它们作为一个候选添加在仿射AMVP列表中。当使用6参数仿射模式对当前CU进行编解码并且所有三个CPMV都可用时,将它们作为一个候选添加在仿射AMVP列表中。否则,将构建的AMVP候选将设置为不可用。

[0194] 如果在插入有效的继承的仿射AMVP候选和构建的AMVP候选后,仿射AMVP列表候选的数量仍然小于2,则将依次添加 mv_0 、 mv_1 和 mv_2 作为平移MV,以预测当前CU(如果可用)的所有控制点MV。最后,如果仿射AMVP列表仍然未满足,则使用零MV来填充所述列表。

[0195] VVC中用于仿射模式的利用光流的预测细化

[0196] 与基于像素的运动补偿相比,基于子块的仿射运动补偿可以节省存储器访问带宽并降低计算复杂度,但代价是预测准确性损失。为了实现更精细的运动补偿粒度,使用利用光流的预测细化(PROF)来在不增加用于运动补偿的存储器访问带宽的情况下细化基于子块的仿射运动补偿预测。在VVC中,在执行基于子块的仿射运动补偿之后,通过添加由光流方程推导的差来细化亮度预测样点。将PROF描述为以下四个步骤:

[0197] 步骤1) 执行基于子块的仿射运动补偿以生成子块预测 $I(i, j)$ 。

[0198] 步骤2) 使用3抽头滤波器 $[-1, 0, 1]$ 在每个样点位置处计算子块预测的空间梯度 $g_x(i, j)$ 和 $g_y(i, j)$ 。梯度计算与BDOF中的梯度计算完全相同。

[0199] $g_x(i, j) = (I(i+1, j) \gg \text{shift1}) - (I(i-1, j) \gg \text{shift1})$ (4)

[0200] $g_y(i, j) = (I(i, j+1) \gg \text{shift1}) - (I(i, j-1) \gg \text{shift1})$ (5)

[0201] shift1用于控制梯度的精度。将子块(即, 4×4) 预测在每一侧上扩展一个样点用于梯度计算。为了避免额外的存储器带宽和额外的内插计算, 从参考图片中的最近整数像素位置复制扩展边界上的那些扩展样点。

[0202] 步骤3) 通过以下光流方程计算亮度预测细化。

$$[0203] \quad \Delta I(i, j) = g_x(i, j) * \Delta v_x(i, j) + g_y(i, j) * \Delta v_y(i, j) \quad (6)$$

[0204] 其中, $\Delta v(i, j)$ 是针对样点位置 (i, j) 计算的样点MV(由 $v(i, j)$ 表示) 与样点 (i, j) 所属子块的子块MV之间的差, 如图14所示。 $\Delta v(i, j)$ 以 $1/32$ 亮度样点精度为单位进行量化。

[0205] 因为仿射模型参数和相对于子块中心的样点位置不会逐子块变化, 所以可以针对第一个子块计算 $\Delta v(i, j)$, 并且将其重复用于相同CU中的其他子块。设 $dx(i, j)$ 和 $dy(i, j)$ 为从样点位置 (i, j) 到子块 (x_{SB}, y_{SB}) 的中心的水平和垂直偏移, 可以通过以下方程推导 $\Delta v(x, y)$,

$$\begin{cases} dx(i, j) = i - x_{SB} \\ dy(i, j) = j - y_{SB} \end{cases} \quad (7)$$

$$[0206] \quad \begin{cases} \Delta v_x(i, j) = C * dx(i, j) + D * dy(i, j) \\ \Delta v_y(i, j) = E * dx(i, j) + F * dy(i, j) \end{cases} \quad (8)$$

[0207] 为了保持准确性, 将子块 (x_{SB}, y_{SB}) 的输入计算为 $((W_{SB}-1)/2, (H_{SB}-1)/2)$, 其中, W_{SB} 和 H_{SB} 分别为子块宽度和高度。

[0208] 对于4参数仿射模型,

$$[0209] \quad \begin{cases} C = F = \frac{v_{1x} - v_{0x}}{w} \\ E = -D = \frac{v_{1y} - v_{0y}}{w} \end{cases} \quad (9)$$

[0210] 对于6参数仿射模型,

$$[0211] \quad \begin{cases} C = \frac{v_{1x} - v_{0x}}{w} \\ D = \frac{v_{2x} - v_{0x}}{h} \\ E = \frac{v_{1y} - v_{0y}}{w} \\ F = \frac{v_{2y} - v_{0y}}{h} \end{cases} \quad (10)$$

[0212] 其中, (v_{0x}, v_{0y}) 、 (v_{1x}, v_{1y}) 和 (v_{2x}, v_{2y}) 是左上、右上和左下控制点运动矢量, w 和 h 是CU的宽度和高度。

[0213] 步骤4) 最后, 将亮度预测细化 $\Delta I(i, j)$ 添加到子块预测 $I(i, j)$ 。最终预测 I' 生成如下方程。

$$[0214] \quad I'(i, j) = I(i, j) + \Delta I(i, j)$$

[0215] 对于仿射编码CU, 在以下两种情况下不应用PROF: 1) 所有控制点MV相同, 这表明CU仅具有平移运动; 2) 仿射运动参数大于指定的极限, 因为基于子块的仿射MC被降级为基于CU的MC以避免大的存储器访问带宽要求。

[0216] 应用快速编码方法来降低利用PROF的仿射运动估计的编码复杂度。在以下两种情况下, 在仿射运动估计阶段不应用PROF: a) 如果该CU不是根块并且其父块未选择仿射模式作为其最佳模式, 则由于当前CU选择仿射模式作为最佳模式的可能性低, 所以不应用PROF; b) 如果四个仿射参数(C、D、E、F)的幅值都小于预定义阈值并且当前图片不是低延迟图片,

则由于在这种情况下由PROF引入的改进很小,所以不应用PROF。通过这种方式,可以加速利用PROF的仿射运动估计。

[0217] VVC中的基于子块的时间运动矢量预测 (SbTMVP)

[0218] VVC支持基于子块的时间运动矢量预测 (SbTMVP) 方法。与HEVC中的时间运动矢量预测 (TMVP) 类似,SbTMVP使用同位图片中的运动场来改进当前图片中CU的运动矢量预测和合并模式。将TMVP所使用的相同同位图片用于SbTMVP。SbTMVP与TMVP的不同之处在于以下两个主要方面:

[0219] -TMVP预测CU级的运动,但SbTMVP预测子CU级的运动;

[0220] -TMVP从同位图片中的同位块获取时间运动矢量(同位块是相对于当前CU的右下或中心块),而SbTMVP在从同位图片获取时间运动信息之前应用运动移位,其中,运动移位是从当前CU的空间邻近块之一的运动矢量获得的。

[0221] 图15A和图15B中图示了SbTMVP过程。SbTMVP在两个步骤中预测当前CU内的子CU的运动矢量。在第一步骤中,检查图15A中的空间邻近块A1。如果A1具有使用同位图片作为其参考图片的运动矢量,就选择该运动矢量作为要应用的运动移位。如果没有识别出这种运动,则将运动移位设置为(0,0)。

[0222] 在第二步骤中,如图15B所示,应用在第1步骤中识别的运动移位(即,添加到当前块的坐标)以从同位图片中获得子CU级运动信息(运动矢量和参考索引)。图15B中的示例假设运动移位设置为阻止A1的运动。然后,对于每个子CU,使用同位图片中其对应块(覆盖中心样点的最小运动网格)的运动信息来推导所述子CU的运动信息。在识别出同位子CU的运动信息后,按照类似于HEVC的TMVP过程的方式将所述运动信息转换为当前子CU的运动矢量和参考索引,其中,应用时间运动缩放以将时间运动矢量的参考图片与当前CU的参考图片对齐。

[0223] 在VVC中,包含SbTMVP候选和仿射合并候选的组合的基于子块的合并列表用于基于子块的合并模式的信号传输。SbTMVP模式由序列参数集 (SPS) 标志启用/禁用。如果启用了SbTMVP模式,则添加SbTMVP预测值作为基于子块的合并候选列表的第一个条目,并且然后是仿射合并候选。在SPS中用信号传输基于子块的合并列表的尺寸,并且在VVC中基于子块的合并列表的最大允许尺寸为5。

[0224] SbTMVP中使用的子CU尺寸固定为 8×8 ,并且与仿射合并模式类似,SbTMVP模式仅适用于宽度和高度两者都大于或等于8的CU。

[0225] 附加SbTMVP合并候选的编码逻辑与其他合并候选的编码逻辑相同,即,对于P或B条带中的每个CU,执行附加RD检查以决定是否使用SbTMVP候选。

[0226] VVC中的自适应运动矢量分辨率 (AMVR)

[0227] 在HEVC中,当条带标头中use_integer_mv_flag等于0时,以四分之一亮度样点为单位用信号传输运动矢量差 (MVD) (CU的运动矢量与预测运动矢量之间)。在VVC中,引入了CU级自适应运动矢量分辨率 (AMVR) 方案。AMVR允许以不同的精度对CU的MVD进行编解码。根据当前CU的模式(标准AMVP模式或仿射AMVP模式),可以如下自适应地选择当前CU的MVD:

[0228] -标准AMVP模式:四分之一亮度样点、半亮度样点、整数亮度样点或四亮度样点。

[0229] -仿射AMVP模式:四分之一亮度样点、整数亮度样点或1/16亮度样点。

[0230] 如果当前CU具有至少一个非零MVD分量,则有条件地用信号传输CU级MVD分辨率指

示。如果所有MVD分量(也就是说,参考列表L0和参考列表L1的水平MVD和垂直MVD两者)都为零,则推断出四分之一亮度样点MVD分辨率。

[0231] 对于具有至少一个非零MVD分量的CU,用信号传输第一标志,以指示是否针对CU使用了四分之一亮度样点MVD精度。如果第一标志为0,则不需要进一步的信号传输,并且将四分之一亮度样点MVD精度用于当前CU。否则,用信号传输第二标志以指示将半亮度样点或其他MVD精度(整数或四亮度样点)用于标准AMVP CU。在半亮度样点的情况下,针对半亮度样点位置使用6抽头内插滤波器,而不是默认的8抽头内插滤波器。否则,用信号传输第三标志,以指示是将整数亮度样点还是四亮度样点MVD精度用于标准AMVP CU。在仿射AMVP CU的情况下,使用第二标志来指示是使用整数亮度样点还是1/16亮度样点MVD精度。为了确保重建的MV具有预期精度(四分之一亮度样点、半亮度样点、整数亮度样点或四亮度样点),在与MVD一起添加之前,可以将CU的运动矢量预测值舍入到与MVD的精度相同的精度。将运动矢量预测值向零舍入(即,将负运动矢量预测值向正无穷大舍入,并且将正运动矢量预测值向负无穷大舍入)。

[0232] 编码器使用RD检查确定当前CU的运动矢量分辨率。为了避免总是对每个MVD分辨率执行四次CU级RD检查,在VTM14中,仅有条件地调用除四分之一亮度样点以外的MVD精度的RD检查。对于标准AVMP模式,首先计算四分之一亮度样点MVD精度和整数亮度样点MV精度的RD成本。然后,将整数亮度样点MVD精度的RD成本与四分之一亮度样点MVD精度的RD成本进行比较,以决定是否需要进一步检查四亮度样点MVD精度的RD成本。当四分之一亮度样点MVD精度的RD成本远小于整数亮度样点MVD精度的RD成本时,跳过四个亮度样点MVD精度的RD检查。然后,如果整数亮度样点MVD精度的RD成本显著大于先前测试的MVD精度的最佳RD成本,则跳过半亮度样点MVD精度的检查。对于仿射AMVP模式,如果在检查了仿射合并/跳过模式、合并/跳过模式、四分之一亮度样点MVD精度标准AMVP模式和四分之一亮度样点MVD精度仿射AMVP模式的率失真成本之后未选择仿射帧间模式,则不检查1/16亮度样点MV精度和1像素MV精度仿射帧间模式。此外,在1/16亮度样点和四分之一亮度样点MV精度仿射帧间模式下,使用在四分之一亮度样点MV精度仿射帧间模式下获得的仿射参数作为开始搜索点。

[0233] VVC中的具有CU级权重的双向预测(BCW)

[0234] 在HEVC中,通过对从两个不同的参考图片和/或使用两个不同的运动矢量获得的两个预测信号求平均来生成双向预测信号。在VVC中,双向预测模式超越了简单的平均,允许对这两个预测信号进行加权平均。

[0235]
$$P_{bi-pred} = ((8-w)*P_0 + w*P_1 + 4) >> 3 \quad (11)$$

[0236] 在加权平均双向预测中允许五个权重, $w \in \{-2, 3, 4, 5, 10\}$ 。对于每个双向预测CU,以以下两种方式之一确定权重w: 1) 对于非合并CU,在运动矢量差之后用信号传输权重索引; 2) 对于合并CU,基于合并候选索引从相邻块推断权重索引。BCW仅适用于具有256个或更多亮度样点的CU(即, CU宽度乘以CU高度大于或等于256)。对于低延迟图片,使用所有5个权重。对于非低延迟图片,仅使用3个权重 ($w \in \{3, 4, 5\}$)。

[0237] -在编码器处,应用快速搜索算法来查找权重索引,而不会显著增加编码器的复杂性。这些算法总结如下。有关更多详细信息,请参阅VTM软件和文档JVET-L0646。当与AMVR组合时,如果当前图片是低延迟图片,则仅针对1像素和4像素运动矢量精度有条件地检查不相等的权重。

[0238] -当与仿射组合时,当且仅当仿射模式被选择为当前最佳模式时,才可以对不相等的权重执行仿射ME。

[0239] -当双向预测中的两个参考图片相同时,仅有条件地检查不相等的权重。

[0240] -当满足某些条件时,不搜索不相等的权重,这取决于当前图片与其参考图片之间的POC距离、编码QP和时间级。

[0241] BCW权重索引使用一个上下文编码的二进制位和随后的旁路编码的二进制位进行编解码。第一上下文编码的二进制位指示是否使用相等的权重;并且如果使用不相等的权重,则使用旁路编码用信号传输附加二进制位以指示使用哪个不相等的权重。

[0242] 加权预测(WP)是由H.264/AVC和HEVC标准支持以在衰减的情况下对视频内容进行高效编解码的编解码工具。VVC标准中还增加了对WP的支持。WP允许针对参考图片列表L0和L1中的每一个中的每个参考图片用信号传输加权参数(权重和偏移)。然后,在运动补偿期间,应用(多个)对应参考图片的(多个)权重和(多个)偏移。WP和BCW是为不同类型的视频内容而设计的。为了避免WP与BCW之间的交互(这可能使VVC解码器设计复杂化),如果CU使用WP,则不用信号传输BCW权重索引,并且w被推断为4(即,应用相等的权重)。对于合并CU,基于合并候选索引从相邻块推断权重索引。这可以应用于标准合并模式和继承的仿射合并模式两者。对于构建的仿射合并模式,基于至多3个块的运动信息来构建仿射运动信息。简单地将使用所构建的仿射合并模式的CU的BCW索引设置为等于第一控制点MV的BCW索引。

[0243] 在VVC中,CIIP和BCW不能联合应用于CU。当使用CIIP模式对CU进行编解码时,将当前CU的BCW索引设置为2,例如,相等权重。

[0244] VVC中的双向光流(BDOF)

[0245] 双向光流(BDOF)工具被包括在VVC中。先前称为BIO的BDOF被包括在JEM中。与JEM版本相比,VVC中的BDOF是更简单的版本,其需要的计算少得多,尤其是在乘法次数和乘法器大小方面。

[0246] BDOF用于细化 4×4 子块级的CU的双向预测信号。如果BDOF满足以下所有条件,则将BDOF应用于CU:

[0247] -使用“真实”双向预测模式对CU进行编解码,即,两个参考图片中的一个按显示顺序在当前图片之前并且另一个按显示顺序在当前图片之后

[0248] -两个参考图片到当前图片的距离(即,POC差)相同

[0249] -两个参考图片均为短期参考图片。

[0250] -CU未使用仿射模式或SbTMVP合并模式进行编解码

[0251] -CU具有超过64个亮度样点

[0252] -CU高度和CU宽度均大于或等于8个亮度样点

[0253] -BCW权重索引指示相等的权重

[0254] -未针对当前CU启用WP

[0255] -未针对当前CU使用CIIP模式

[0256] BDOF仅适用于亮度分量。正如其名称所指示,BDOF模式基于光流概念,所述概念假设对象的运动是平滑的。对于每个 4×4 子块,通过最小化L0和L1预测样点之间的差来计算运动细化 (v_x, v_y) 。然后使用运动细化来调整 4×4 子块中的双向预测样点值。将以下步骤应用于BDOF过程。

[0257] 首先,通过直接计算两个邻近样点之间的差来计算两个预测信号的水平垂直梯度 $\frac{\partial I^{(k)}}{\partial x}(i,j)$ 和 $\frac{\partial I^{(k)}}{\partial y}(i,j)$ ($k=0,1$),即,

$$\begin{aligned} \frac{\partial I^{(k)}}{\partial x}(i,j) &= ((I^{(k)}(i+1,j) \gg \text{shift1}) - (I^{(k)}(i-1,j) \gg \text{shift1})) \\ \frac{\partial I^{(k)}}{\partial y}(i,j) &= ((I^{(k)}(i,j+1) \gg \text{shift1}) - (I^{(k)}(i,j-1) \gg \text{shift1})) \end{aligned} \quad (12)$$

[0259] 其中, $I^{(k)}(i,j)$ 是列表 $k, k=0,1$ 中预测信号的坐标 (i,j) 处的样点值,并且基于亮度位深度 bitDepth 计算 shift1 ,如 $\text{shift1}=\max(6, \text{bitDepth}-6)$ 。

[0260] 然后,计算梯度的自相关性和互相关性 S_1, S_2, S_3, S_5 和 S_6 如下

$$\begin{aligned} S_1 &= \sum_{(i,j) \in \Omega} \text{Abs}(\psi_x(i,j)), \quad S_3 = \sum_{(i,j) \in \Omega} \theta(i,j) \cdot \text{Sign}(\psi_x(i,j)) \\ S_2 &= \sum_{(i,j) \in \Omega} \psi_x(i,j) \cdot \text{Sign}(\psi_y(i,j)) \end{aligned} \quad (13)$$

$$S_5 = \sum_{(i,j) \in \Omega} \text{Abs}(\psi_y(i,j)), \quad S_6 = \sum_{(i,j) \in \Omega} \theta(i,j) \cdot \text{Sign}(\psi_y(i,j))$$

[0262] 其中,

$$\begin{aligned} \psi_x(i,j) &= \left(\frac{\partial I^{(1)}}{\partial x}(i,j) + \frac{\partial I^{(0)}}{\partial x}(i,j) \right) \gg n_a \\ \psi_y(i,j) &= \left(\frac{\partial I^{(1)}}{\partial y}(i,j) + \frac{\partial I^{(0)}}{\partial y}(i,j) \right) \gg n_a \\ \theta(i,j) &= (I^{(1)}(i,j) \gg n_b) - (I^{(0)}(i,j) \gg n_b) \end{aligned} \quad (14)$$

[0264] 其中, Ω 是 4×4 子块周围的 6×6 窗口,并且 n_a 和 n_b 的值分别设置为等于 $\min(1, \text{bitDepth}-11)$ 和 $\min(4, \text{bitDepth}-8)$ 。

[0265] 然后使用互相关性项和自相关性项、使用以下公式来推导运动细化 (v_x, v_y) :

$$\begin{aligned} v_x &= S_1 > 0? \text{clip3} \left(-\text{th}'_{BIO}, \text{th}'_{BIO}, -((S_3 \cdot 2^{n_b-n_a}) \gg \lfloor \log_2 S_1 \rfloor) \right) : 0 \\ v_y &= S_5 > 0? \text{clip3} \left(-\text{th}'_{BIO}, \text{th}'_{BIO}, - \left((S_6 \cdot 2^{n_b-n_a} \right. \right. \\ &\quad \left. \left. - ((v_x S_{2,m}) \ll n_{S_2} + v_x S_{2,s}) / 2 \right) \gg \lfloor \log_2 S_5 \rfloor \right) : 0 \end{aligned} \quad (15)$$

[0267] 其中 $S_{2,m} = S_2 \gg n_{S_2}$, $S_{2,s} = S_2 \& (2^{n_{S_2}} - 1)$, $\text{th}'_{BIO} = 2^{\max(5, \text{BD}-7)}$ 。 $\lfloor \cdot \rfloor$ 是地板函数,并且 $n_{S_2} = 12$ 。

[0268] 基于运动细化和梯度,针对 4×4 子块中的每个样点计算以下调整:

$$b(x,y) = \text{rnd} \left(\left(v_x \left(\frac{\partial I^{(1)}(x,y)}{\partial x} - \frac{\partial I^{(0)}(x,y)}{\partial x} \right) + v_y \left(\frac{\partial I^{(1)}(x,y)}{\partial y} - \frac{\partial I^{(0)}(x,y)}{\partial y} \right) + 1 \right) / 2 \right) \quad (16)$$

[0270] 最后,通过如下调整双向预测样点来计算CU的BDOF样点:

[0271] $\text{pred}_{\text{BDOF}}(x, y) = (I^{(0)}(x, y) + I^{(1)}(x, y) + b(x, y) + o_{\text{offst}}) \gg \text{shift} \quad (17)$

[0272] 这些值被选择成使得BDOF过程中的乘法器不超过15位,并且BDOF过程的中间参数的最大位宽保持在32位以内。

[0273] 为了推导梯度值,需要生成当前CU边界外、列表k(k=0,1)中的一些预测样点 $I^{(k)}(i, j)$ 。如图16中所描绘的,VVC中的BDOF在CU的边界周围使用一个扩展的行/列。为了控制生成边界外预测样点的计算复杂度,通过直接取附近整数位置处的参考样点(使用坐标上的 $\text{floor}()$ 运算)而不进行内插来生成扩展区域(白色位置)中的预测样点,并且使用标准的8抽头运动补偿内插滤波器来生成CU(灰色位置)内的预测样点。这些扩展样点值仅用于梯度计算。对于BDOF过程中的剩余步骤,如果需要CU边界外的任何样点和梯度值,则从其最近的邻近块对其进行填充(即重复)。

[0274] 当CU的宽度和/或高度大于16个亮度样点时,可以将CU分割为宽度和/或高度等于16个亮度样点的子块,并且在BDOF过程中将子块边界视为CU边界。将BDOF过程的最大单元尺寸限制为 16×16 。对于每个子块,可以跳过BDOF过程。当初始L0和L1预测样点之间的SAD小于阈值时,不将BDOF过程应用于子块。将阈值设置为等于 $(8 * W * (H >> 1))$,其中,W表示子块宽度,并且H表示子块高度。为了避免SAD计算的额外复杂性,这里重新使用在DVMR过程中计算的初始L0和L1预测样点之间的SAD。

[0275] 如果针对当前块启用BCW,即,BCW权重索引指示不相等的权重,则禁用双向光流。类似地,如果针对当前块启用WP,即,对于两个参考图片中的任一个,luma_weight_lx_flag为1,则也禁用BDOF。当以对称MVD模式或CIIP模式对CU进行编解码时,也禁用BDOF。

[0276] VVC中的解码器侧运动矢量细化(DMVR)

[0277] 为了提高合并模式的MV的准确性,在VVC中应用基于双边匹配(BM)的解码器侧运动矢量细化。在双向预测操作中,在参考图片列表L0和参考图片列表L1中的初始MV周围搜索细化的MV。BM方法计算参考图片列表L0和列表L1中的两个候选块之间的失真。如图17中图示的,基于初始MV周围的每个MV候选计算红色块之间的SAD。具有最低SAD的MV候选成为细化的MV,并且用于生成双向预测信号。

[0278] 在VVC中,DMVR的应用受到限制,并且仅适用于用以下模式和特征进行编解码的CU:

[0279] -具有双向预测MV的CU级合并模式

[0280] -相对于当前图片,一个参考图片在过去并且另一个参考图片在未来

[0281] -两个参考图片到当前图片的距离(即,POC差)相同

[0282] -两个参考图片均为短期参考图片

[0283] -CU具有超过64个亮度样点

[0284] -CU高度和CU宽度均大于或等于8个亮度样点

[0285] -BCW权重索引指示相等的权重

[0286] -未针对当前块启用WP

[0287] -未针对当前块使用CIIP模式

[0288] 通过DMVR过程推导的细化的MV用于生成帧间预测样点,并且还在用于未来图片编解码的时间运动矢量预测中使用。而原始MV在去块过程中使用,并且还在用于未来CU编解码的空间运动矢量预测中使用。

[0289] DMVR的附加特征在以下子条款中提及。

[0290] 搜索方案

[0291] 在DMVR中,搜索点围绕初始MV,并且MV偏移遵循MV差值镜像规则。换句话说,由候选MV对(MV0,MV1)表示的、由DMVR检查的任何点都遵循以下两个方程:

$$[0292] \quad MV0' = MV0 + MV_offset \quad (18)$$

$$[0293] \quad MV1' = MV1 - MV_offset \quad (19)$$

[0294] 其中,MV_offset表示参考图片之一中初始MV与细化的MV之间的细化偏移。细化搜索范围是来自初始MV的两个整数亮度样点。搜索包括整数样点偏移搜索阶段和分数样点细化阶段。

[0295] 将25点完全搜索应用于整数样点偏移搜索。首先计算初始MV对的SAD。如果初始MV对的SAD小于阈值,则终止DMVR的整数样点阶段。否则,计算并按光栅扫描顺序检查剩余24个点的SAD。选择SAD最小的点作为整数样点偏移搜索阶段的输出。为了降低DMVR细化的不确定性的罚分,提出了在DMVR过程期间支持原始MV。初始MV候选所参考的参考块之间的SAD减少了SAD值的1/4。

[0296] 整数样点搜索之后是分数样点细化。为了降低计算复杂度,分数样点细化是通过使用参数误差表面方程而不是使用SAD比较进行附加搜索推导的。基于整数样点搜索阶段的输出有条件地调用分数样点细化。当整数样点搜索阶段在第一次迭代搜索或第二次迭代搜索中以中心具有最小SAD而终止时,进一步应用分数样点细化。

[0297] 在基于参数误差表面的子像素偏移估计中,中心位置成本和和距离中心的四个邻近位置处的成本用于拟合以下形式的2-D抛物线误差表面方程:

$$[0298] \quad E(x,y) = A(x-x_{min})^2 + B(y-y_{min})^2 + C \quad (20)$$

[0299] 其中,(x_{min}, y_{min})与具有最低成本的分数位置相对应,并且C与最小成本值相对应。通过使用五个搜索点的成本值求解上述方程,(x_{min}, y_{min})计算为:

$$[0300] \quad x_{min} = (E(-1,0) - E(1,0)) / (2(E(-1,0) + E(1,0) - 2E(0,0))) \quad (21)$$

$$[0301] \quad y_{min} = (E(0,-1) - E(0,1)) / (2(E(0,-1) + E(0,1) - 2E(0,0))) \quad (22)$$

[0302] x_{min} 和 y_{min} 的值自动限制在-8与8之间,因为所有成本值都是正的并且最小值为E(0,0)。这对应于在VVC中具有1/16像素MV准确度的半响声偏移。计算出的分数(x_{min}, y_{min})被添加到整数距离细化MV以获得子像素准确度细化 ΔMV 。

[0303] 双线性内插和样点填充

[0304] 在VVC中,MV的分辨率是1/16亮度样点。使用8抽头内插滤波器来插入分数位置处的样点。在DMVR中,搜索点围绕具有整数样点偏移的初始分数像素MV,因此需要对这些分数位置的样点进行内插以进行DMVR搜索过程。为了降低计算复杂度,在DMVR中的搜索过程中使用双线性内插滤波器来生成分数样点。由使用双线性滤波器引起的另一个重要影响是,在2样点搜索范围内,DMVR不能比标准运动补偿过程访问更多参考样点。在通过DMVR搜索过程获得细化的MV之后,应用标准8抽头内插滤波器来生成最终预测。为了不相对于标准MC过程访问更多参考样点,将从那些可用样点中填充基于原始MV的内插过程不需要但基于细化MV的内插过程需要的样点。

[0305] VVC中的几何分区模式(GPM)

[0306] 在VVC中,支持几何分区模式用于帧间预测。使用CU级标志用信号传输几何分区模

式作为一种合并模式,其他合并模式包括常规合并模式、MMVD模式、CIIP模式和子块合并模式。对于每个可能的CU尺寸 $w \times h = 2^m \times 2^n$ (其中, $m, n \in \{3 \cdots 6\}$,不包括 8×64 和 64×8),几何分区模式总共支持64个分区。

[0307] 当使用该模式时,通过几何定位的直线将CU分割成两个部分(图18)。分割线的位置是根据特定分区的角度和偏移参数在数学上推导的。使用CU中的几何分区的每个部分自身的运动对其进行帧间预测;每个分区仅允许单向预测,即,每个部分具有一个运动矢量和一个参考索引。应用单向预测运动约束以确保与传统的双向预测一样,每个CU仅需要两种运动补偿预测。使用3.4.11.1中描述的过程推导每个分区的单向预测运动。

[0308] 如果针对当前CU使用几何分区模式,则进一步用信号传输指示几何分区的分区模式(角度和偏移)的几何分区索引和两个合并索引(每个分区一个)。在SPS中显式地用信号传输最大GPM候选尺寸的数目,并且所述数目指定用于GPM合并索引的语法二值化。在预测了几何分区的每个部分之后,使用如3.4.11.2中的具有自适应权重的混合处理来调整沿几何分区边缘的样点值。这是整个CU的预测信号,并且变换和量化过程可以像在其他预测模式中一样应用于整个CU。最后,如在3.4.11.3中那样存储使用几何分区模式预测的CU的运动场。

[0309] 单向预测候选列表构建

[0310] 直接从根据3.4.1中的扩展合并预测过程构建的合并候选列表推导单向预测候选列表。将 n 表示为几何单向预测候选列表中的单向预测运动的索引。使用第 n 个扩展合并候选的LX运动矢量(X 等于 n 的奇偶性)作为几何分区模式的第 n 个单向预测运动矢量。这些运动矢量在图19中用“x”标记。在第 n 个扩展合并候选的对应LX运动矢量不存在的情况下,使用同一候选的 $L(1-X)$ 运动矢量作为几何分区模式的单向预测运动矢量。

[0311] VVC中的组合帧间和帧内预测(CIIP)

[0312] 在VVC中,当以合并模式对CU编解码时,如果CU包含至少64个亮度样点(即,CU宽度乘以CU高度等于或大于64)并且如果CU宽度和CU高度都小于128个亮度样点,则用信号传输用于指示是否将组合帧间/帧内预测(CIIP)模式应用于当前CU的附加标志。正如其名称所指示,CIIP预测将帧间预测信号与帧内预测信号相组合。使用应用于常规合并模式的相同帧间预测过程推导CIIP模式下的帧间预测信号 P_{inter} ;并且在利用平面模式的常规帧内预测过程之后推导帧内预测信号 P_{intra} 。然后,使用加权平均来组合帧内预测信号和帧间预测信号,其中,根据(图20中描绘的)顶部相邻块和左侧相邻块的编解码模式来计算权重值如下:

[0313] -如果顶部邻近块可用且已进行帧内编码,则将 $isIntraTop$ 设置为1,否则将 $isIntraTop$ 设置为0;

[0314] -如果左侧邻近块可用且已进行帧内编码,则将 $isIntraLeft$ 设置为1,否则将 $isIntraLeft$ 设置为0;

[0315] -如果 $(isIntraLeft + isIntraTop)$ 等于2,则将 wt 设置为3;

[0316] -否则,如果 $(isIntraLeft + isIntraTop)$ 等于1,则将 wt 设置为2;

[0317] -否则,将 wt 设置为1。

[0318] CIIP预测形成如下:

$$P_{CIIP} = ((4-wt) * P_{inter} + wt * P_{intra} + 2) >> 2 \quad (23)$$

[0320] VVC中的帧内块复制(IBC)

[0321] 帧内块复制(IBC)是在SCC上的HEVC扩展中采用的工具。众所周知,其显著提高了屏幕内容材料的编解码效率。由于IBC模式被实施为块级编解码模式,因此在编码器处执行块匹配(BM),以找到每个CU的最佳块矢量(或运动矢量)。在此,块矢量用于指示从当前块到参考块的位移,所述参考块已经在当前图片内重建。IBC编解码CU的亮度块矢量具有整数精度。色度块矢量也舍入到整数精度。当与AMVR组合时,IBC模式可以在1像素运动矢量精度与4像素运动矢量精度之间切换。IBC编解码CU被视为不同于帧内预测模式或帧间预测模式的第三预测模式。IBC模式适用于宽度和高度均小于或等于64个亮度样点的CU。

[0322] 在编码器侧,对IBC执行基于散列的运动估计。编码器对宽度或高度不大于16个亮度样点的块执行RD检查。对于非合并模式,首先使用基于散列的搜索来执行块矢量搜索。如果散列搜索没有返回有效的候选,则执行基于块匹配的本地搜索。

[0323] 在基于散列的搜索中,将当前块与参考块之间的散列键匹配(32位CRC)扩展到所有允许的块尺寸。当前图片中每个位置的散列键计算是基于 4×4 子块。对于较大尺寸的当前块,当所有 4×4 子块的所有散列键与对应参考位置中的散列键匹配时,确定散列键与参考块的散列键相匹配。如果发现多个参考块的散列键与当前块的散列键匹配,则计算每个匹配的参考的块矢量成本,并选择具有最小成本的参考。

[0324] 在块匹配搜索中,将搜索范围设置为覆盖先前CTU和当前CTU两者。在CU级下,使用标志用信号传输IBC模式,并且其可以用信号传输为如下的IBC AMVP模式或IBC跳过/合并模式:

[0325] IBC跳过/合并模式:合并候选索引用于指示来自邻近候选IBC编解码块的列表中的哪些块矢量用于预测当前块。合并列表由空间候选、HMVP候选和成对候选构成。

[0326] IBC AMVP模式:以与运动矢量差相同的方式对块矢量差进行编解码。块矢量预测方法使用两个候选作为预测值,一个来自左侧邻近块,并且一个来自上方邻近块(如果IBC编解码的话)。当任一个邻近块不可用时,默认的块矢量可以用作预测值。用信号传输标志以指示块矢量预测值索引。

[0327] IBC参考区域

[0328] 为了减少存储器消耗和解码器复杂度,VVC中的IBC只允许包括当前CTU的区域和左侧CTU的一定区域的预定义区域的重建部分。根据当前编码CU位置在当前CTU内的定位,以下各项适用:

[0329] 如果当前块落入当前CTU的左上 64×64 块中,则除了当前CTU中已经重建的样点外,使用CPR模式,所述当前块还可以参考左侧CTU的右下 64×64 块中的参考样点。使用CPR模式,当前块还可以参考左侧CTU的左下 64×64 块中的参考样点和左侧CTU的右上 64×64 块中的参考样点。

[0330] 如果当前块落入当前CTU的右上 64×64 块中,则除了当前CTU中已经重建的样点之外,在相对于当前CTU尚未重建亮度位置(0,64)的情况下,使用CPR模式,当前块还可以参考左侧CTU的左下 64×64 块和右下 64×64 块中的参考样点;否则,当前块还可以参考左侧CTU的右下 64×64 块中的参考样点。

[0331] 如果当前块落入当前CTU的左下 64×64 块中,则除了当前CTU中已经重建的样点之外,在相对于当前CTU尚未重建亮度位置(64,0)的情况下,使用CPR模式,当前块还可以参考左侧CTU的右上 64×64 块和右下 64×64 块中的参考样点。否则,使用CPR模式,当前块还可以

参考左侧CTU的右下 64×64 块中的参考样点。

[0332] 如果当前块落入当前CTU的右下 64×64 块中,则使用CPR模式,所述当前块只能参考当前CTU中已经重建的样点。

[0333] 该限制允许IBC模式使用硬件实施方式的本地片上存储器来实施。

[0334] ECM中的局部照明补偿(LIC)

[0335] LIC是用于将当前块与其预测块之间的局部照明变化建模为当前块模板与参考块模板之间的局部照度变化的函数的帧间预测技术。所述函数的参数可以用比例 α 和偏移 β 表示,其形成了用于补偿照明变化的线性方程(即, $\alpha * p[x] + \beta$),其中, $p[x]$ 是在参考图片上的位置 x 处的MV指向的参考样点。由于 α 和 β 可以基于当前块模板和参考块模板来推导,因此除了针对AMVP模式用信号传输指示使用LIC的LIC标志之外,它们不需要信令开销。

[0336] 通过以下修改将JVET-00066中提出的局部照明补偿用于单向预测帧间CU。

[0337] -帧内邻近块样点可以用于LIC参数推导;

[0338] -针对具有少于32个亮度样点的块禁用LIC;

[0339] -对于非子块模式和仿射模式两者,LIC参数推导是基于与当前CU相对应的模板块样点执行的,而不是基于与第一个左上 16×16 单元相对应的部分模板块样点执行的;

[0340] -通过在不将其舍入到整数像素精度的情况下使用MC和块MV来生成参考块模板的样点。

[0341] ECM中的非相邻空间候选

[0342] 将如JVET-L0399中的非相邻空间合并候选插入到常规合并候选列表中的TMVP之后。在图21中示出了空间合并候选的模式。非相邻空间候选与当前编码块之间的距离是基于当前编码块的宽度和高度。未应用行缓冲器限制。

[0343] ECM中的模板匹配(TM)

[0344] 模板匹配(TM)是用于通过找到当前图片中的模板(即,当前CU的顶部相邻块和/或左侧相邻块)与参考图片中的块(即,与模板尺寸相同)之间的最接近匹配来细化当前CU的运动信息的解码器侧MV推导方法。如图22所图示的,在 $[-8, +8]$ 像素搜索范围内围绕当前CU的初始运动搜索更佳的MV。在进行以下修改的情况下使用JVET-J0021中的模板匹配方法:基于AMVR模式确定搜索步长,并且可以在合并模式下与双边匹配过程级联TM。

[0345] 在AMVP模式下,基于模板匹配误差来确定MVP候选,以选择实现当前块模板与参考块模板之间的最小差异的一个,然后仅对该特定MVP候选执行TM以进行MV细化。TM通过使用迭代菱形搜索在 $[-8, +8]$ 像素搜索范围内从全像素MVD精度(或对于4像素AMVR模式为4像素)开始来细化该MVP候选。可以通过使用具有全像素MVD精度(或对于4像素AMVR模式为4像素)、随后按顺序是半像素和四分之一像素精度的交叉搜索根据AMVR模式来进一步细化AMVP候选,如表3中所指定的。该搜索过程确保MVP候选在TM过程之后仍然保持如AMVR模式所指示的相同MV精度。

[0346] 表3:AMVR的搜索模式以及利用AMVR的合并模式。

搜索模式	AMVR 模式				合并模式	
	4 像素	全像素	半像素	四分之一像素	AltIF=0	AltIF=1
4 像素菱形	v					
4 像素交叉	v					
[0347] 全像素菱形		v	v	v	v	v
全像素交叉		v	v	v	v	v
半像素交叉			v	v	v	v
四分之一像素交叉				v	v	
1/8 像素交叉					v	

[0348] 在合并模式下,对由合并索引指示的合并候选应用类似的搜索方法。如表3所示,取决于是否根据合并的运动信息使用替代内插滤波器(当AMVR采用半像素模式时使用),TM可以一直执行到1/8像素MVD精度或者跳过超过半像素MVD精度的那些。此外,当启用TM模式时,模板匹配可以作为基于块和基于子块的双边匹配(BM)方法之间的独立过程或额外的MV细化过程来工作,这取决于BM是否可以根据其启用条件检查来启用。

[0349] ECM中的多遍解码器侧运动矢量细化

[0350] 应用多遍解码器侧运动矢量细化。在第一遍中,将双边匹配(BM)应用于编码块。在第二遍中,将BM应用于编码块内的每个 16×16 子块。在第三遍中,通过应用双向光流(BDOF)来细化每个 8×8 子块中的MV。将细化的MV存储用于空间和时间运动矢量预测。

[0351] 第一遍-基于块的双边匹配MV细化

[0352] 在第一遍中,通过将BM应用于编码块来推导细化的MV。与解码器侧运动矢量细化(DMVR)类似,在双向预测操作中,在参考图片列表L0和L1中的两个初始MV(MV0和MV1)周围搜索细化的MV。基于L0和L1中的两个参考块之间的最小双边匹配成本来围绕初始MV推导细化的MV(MV0_pass1和MV1_pass1)。

[0353] BM执行局部搜索以推导整数样点精度intDeltaMV。局部搜索应用 3×3 正方形搜索模式,以在水平方向上的搜索范围 $[-sHor, sHor]$ 和垂直方向上的搜索范围 $[-sVer, sVer]$ 中循环遍历,其中,sHor和sVer的值由块尺寸确定,并且sHor和sVer的最大值为8。

[0354] 双边匹配成本计算为: $bilCost = mvDistanceCost + sadCost$ 。当块尺寸 $cbW * cbH$ 大于64时,应用MRSAD成本函数来去除参考块之间失真的DC效应。当 3×3 搜索模式的中心点处的bilCost具有最小成本时,终止intDeltaMV局部搜索。否则,当前最小成本搜索点成为 3×3 搜索模式的新中心点,并继续搜索最小成本,直到其到达搜索范围的末端。

[0355] 进一步应用现有的分数样点细化来推导最终的deltaMV。然后,将第一遍后的细化MV推导为:

[0356] $-MV0_pass1 = MV0 + deltaMV$

[0357] $-MV1_pass1 = MV1 - deltaMV$

[0358] 第二遍-基于子块的双边匹配MV细化

[0359] 在第二遍中,通过将BM应用于 16×16 网格子块来推导细化的MV。对于每个子块,在

参考图片列表L0和L1中的在第一遍上获得的两个MV (MV0_pass1和MV1_pass1) 周围搜索细化的MV。基于L0和L1中的两个参考子块之间的最小双边匹配成本来推导细化的MV (MV0_pass2(sbIdx2) 和MV1_pass2(sbIdx2))。

[0360] 对于每个子块, BM执行完全搜索以推导整数样点精度intDeltaMV。完全搜索在水平方向上具有搜索范围[-sHor, sHor]并且在垂直方向上具有搜索范围[-sVer, sVer], 其中, sHor和sVer的值由块尺寸确定, 并且sHor和sVer的最大值为8。

[0361] 通过将成本因子应用于两个参考子块之间的SATD成本来计算双边匹配成本如下: $bilCost = satdCost * costFactor$ 。将搜索区域 $(2*sHor+1) * (2*sVer+1)$ 划分为图23所示的至多5个菱形搜索区域。每个搜索区域被分配了costFactor, 所述costFactor由每个搜索点与起始MV之间的距离(intDeltaMV)确定, 并且每个菱形区域按照从搜索区域的中心开始的顺序进行处理。在每个区域中, 搜索点按照从区域的左上角开始一直到右下角的光栅扫描顺序进行处理。在当前搜索区域内的最小bilCost小于阈值(所述阈值等于sbW*sbH)时, 终止整数像素完全搜索, 否则, 整数像素完全搜索继续到下一个搜索区域, 直到检查了所有搜索点为止。

[0362] 进一步应用现有的VVC DMVR分数样点细化来推导最终的deltaMV(sbIdx2)。然后, 将第二遍时的细化MV推导为:

[0363] $-MV0_pass2(sbIdx2) = MV0_pass1 + deltaMV(sbIdx2)$

[0364] $-MV1_pass2(sbIdx2) = MV1_pass1 - deltaMV(sbIdx2)$

[0365] 第三遍-基于子块的双向光流MV细化

[0366] 在第三遍中, 通过将BDOF应用于 8×8 网格子块来推导细化的MV。对于每个 8×8 子块, 应用BDOF细化来推导缩放的Vx和Vy, 而不从第二遍的父子块的细化MV开始进行剪裁。将推导的bioMv(Vx, Vy)舍入到1/16样点精度, 并在-32与32之间进行剪裁。

[0367] 将第三遍时的细化MV (MV0_pass3(sbIdx3) 和MV1_pass3(sbIdx3)) 推导为:

[0368] $-MV0_pass3(sbIdx3) = MV0_pass2(sbIdx2) + bioMv$

[0369] $-MV1_pass3(sbIdx3) = MV0_pass2(sbIdx2) - bioMv$

[0370] ECM中的OBMC

[0371] 当应用OBMC时, 如JVET-L0101所述, 使用邻近块的运动信息利用加权预测来细化CU的顶部边界像素和左侧边界像素。

[0372] 不应用OBMC的条件如下:

[0373] -当在SPS级下禁用OBMC时

[0374] -当当前块具有帧内模式或IBC模式时

[0375] -当当前块应用LIC时

[0376] -当当前亮度块面积小于或等于32时

[0377] 通过使用邻近子块的运动信息将相同的混合应用于顶部子块边界像素、左侧子块边界像素、底部子块边界像素和右侧子块边界像素来执行子块边界OBMC。其是为以下基于子块的编解码工具启用的:

[0378] -仿射AMVP模式;

[0379] -仿射合并模式和基于子块的时间运动矢量预测(SbTMVP);

[0380] -基于子块的双边匹配。

[0381] ECM中的基于样点的BDof

[0382] 在基于样点的BDof中,运动细化 (V_x, V_y) 是按样点执行的,而不是以块为基础推导的。

[0383] 将编码块划分为 8×8 子块。对于每个子块,通过对照阈值检查两个参考子块之间的SAD来确定是否应用BDof。如果决定将BDof应用于子块,则对于子块中的每个样点,使用滑动 5×5 窗口,并将现有的BDof过程应用于每个滑动窗口,以推导 V_x 和 V_y 。应用推导的运动细化 (V_x, V_y) 来调整窗口的中心样点的双向预测样点值。

[0384] ECM中的内插

[0385] 将在VVC中使用的8抽头内插滤波器用12抽头滤波器替换。内插滤波器是从sinc函数推导的并被余弦窗函数裁剪,所述sinc函数的频率响应在奈奎斯特频率下被截断。表4给出了所有16个相位的滤波器系数。图24将内插滤波器与VVC内插滤波器(全部在半像素相位下)的频率响应进行比较。

[0386] 表4:12抽头内插滤波器的滤波器系数

	1/16	-1	2	-3	6	-14	254	16	-7	4	-2	1	0
	2/16	-1	3	-7	12	-26	249	35	-15	8	-4	2	0
	3/16	-2	5	-9	17	-36	241	54	-22	12	-6	3	-1
	4/16	-2	5	-11	21	-43	230	75	-29	15	-8	4	-1
	5/16	-2	6	-13	24	-48	216	97	-36	19	-10	4	-1
	6/16	-2	7	-14	25	-51	200	119	-42	22	-12	5	-1
	7/16	-2	7	-14	26	-51	181	140	-46	24	-13	6	-2
[0387]	8/16	-2	6	-13	25	-50	162	162	-50	25	-13	6	-2
	9/16	-2	6	-13	24	-46	140	181	-51	26	-14	7	-2
	10/16	-1	5	-12	22	-42	119	200	-51	25	-14	7	-2
	11/16	-1	4	-10	19	-36	97	216	-48	24	-13	6	-2
	12/16	-1	4	-8	15	-29	75	230	-43	21	-11	5	-2
	13/16	-1	3	-6	12	-22	54	241	-36	17	-9	5	-2
	14/16	0	2	-4	8	-15	35	249	-26	12	-7	3	-1
	15/16	0	1	-2	4	-7	16	254	-14	6	-3	2	-1

[0388] ECM中的多假设预测 (MHP)

[0389] 在多假设帧间预测模式 (JVET-M0425) 下,除了传统的双向预测信号之外,还用信号传输一个或多个附加的运动补偿预测信号。通过逐样点加权叠加来获得所得的总体预测信号。利用双向预测信号 p_{bi} 和第一附加帧间预测信号/假设 h_3 ,如下获得所得的预测信号 p_3 :

$$p_3 = (1-\alpha) p_{bi} + \alpha h_3$$

[0391] 加权因子 α 由新语法元素add_hyp_weight_idx根据以下映射指定:

[0392]	add_hyp_weight_idx	α
	0	1/4

1	-1/8
---	------

[0393] 与上述类似,可以使用多于一个附加预测信号。将所得的总体预测信号与每个附加预测信号迭代地累加。

$$[0394] \quad p_{n+1} = (1 - \alpha_{n+1}) p_n + \alpha_{n+1} h_{n+1}$$

[0395] 所得的总体预测信号被获得为最后的 p_n (即,具有最大索引 p_n 的 n)。在该EE内,可以使用至多两个附加预测信号 (即, n 被限制为2)。

[0396] 每个附加预测假设的运动参数可以通过指定参考索引、运动矢量预测值索引和运动矢量差来显式地用信号传输,或者可以通过指定合并索引来隐式地用信号传输。单独的多假设合并标志可区分这两种信号传输模式。

[0397] 对于帧间AMVP模式,仅当在双向预测模式下选择BCW中的不相等权重时,才应用MHP。

[0398] MHP和BDOP的组合是可能的,然而BDOP仅应用于预测信号的双向预测信号部分 (即,普通的前两个假设)。

[0399] ECM中利用模板匹配对合并候选进行的自适应重新排序 (ARMC-TM)

[0400] 利用模板匹配 (TM) 对合并候选进行自适应重新排序。将重新排序方法应用于常规合并模式、模板匹配 (TM) 合并模式和仿射合并模式 (不包括SbTMVP候选)。对于TM合并模式,在细化过程之前对合并候选进行重新排序。

[0401] 在构建合并候选列表后,将合并候选划分为若干个子组。对于常规合并模式和TM合并模式,将子组大小设置为5。对于仿射合并模式,将子组大小设置为3。基于模板匹配、根据成本值以升序的方式对每个子组中的合并候选进行重新排序。为了简化,不重新排序最后一个子组而不是第一个子组中的合并候选。

[0402] 通过当前块的模板的样点与其对应的参考样点之间的绝对差值和 (SAD) 来测量合并候选的模板匹配成本。所述模板包括与当前块邻近的一组重建样点。通过合并候选的运动信息来定位模板的参考样点。

[0403] 当合并候选利用双向预测时,还通过双向预测生成合并候选的模板的参考样点,如图25所示。

[0404] 对于子块尺寸等于 $W_{sub} \times H_{sub}$ 的基于子块的合并候选,上方模板包括若干个尺寸为 $W_{sub} \times 1$ 的子模板,并且左侧模板包括若干个尺寸为 $1 \times H_{sub}$ 的子模板。如图26所示,当前块的第一行和第一列中的子块的运动信息用于推导每个子模板的参考样点。

[0405] ECM中利用合并运动矢量差 (MMVD) 的几何分区模式 (GPM)

[0406] 通过在现有的GPM单向MV的顶部上应用运动矢量细化来扩展VVC中的GPM。首先用信号传输用于GPM CU的标志,以指定是否使用该模式。如果使用所述模式,则GPM CU的每个几何分区可以进一步决定是否用信号传输MVD。如果针对几何分区用信号传输MVD,则在选择GPM合并候选之后,通过用信号传输的MVD信息来进一步细化分区的运动。所有其他程序与GPM中的程序保持相同。

[0407] 与在MMVD中类似,将MVD作为一对距离和方向用信号传输。在利用MMVD的GPM (GPM-MMVD) 中涉及九个候选距离 (1/4像素、1/2像素、1像素、2像素、3像素、4像素、6像素、8像素、16像素) 和八个候选方向 (四个水平/垂直方向和四个对角线方向)。另外,当 $pic_fpel_mmvd_enabled_flag$ 等于1时,将MVD如在MMVD中那样左移2位。

[0408] ECM中利用模板匹配(TM)的几何分区模式(GPM)

[0409] 将模板匹配应用于GPM。当针对CU启用GPM模式时,用信号传输CU级标志以指示TM是否被应用于两个几何分区。使用TM细化每个几何分区的运动信息。当选择TM时,根据如表5所示的分区角度使用左侧邻近样点、上方邻近样点、或左侧邻近样点和上方邻近样点来构建模板。然后,通过使用禁用半像素内插滤波器的合并模式的相同搜索模式最小化当前模板与参考图片中的模板之间的差异来细化运动。

[0410] 表5:用于第1几何分区和第2几何分区的模板,其中,A表示使用上方样点,L表示使用左侧样点,并且L+A表示使用左侧样点和上方样点两者。

分区角度	0	2	3	4	5	8	11	12	13	14
第1分区	A	A	A	A	L+A	L+A	L+A	L+A	A	A
第2分区	L+A	L+A	L+A	L	L	L	L	L+A	L+A	L+A
分区角度	16	18	19	20	21	24	27	28	29	30
第1分区	A	A	A	A	L+A	L+A	L+A	L+A	A	A
第2分区	L+A	L+A	L+A	L	L	L	L	L+A	L+A	L+A

[0411] GPM候选列表构建如下:

[0413] 1. 直接从常规合并候选列表中推导交错的列表0MV候选和列表1MV候选,其中,列表0MV候选的优先级高于列表1MV候选。应用其中自适应阈值基于当前CU尺寸的剪除方法来去除冗余MV候选。

[0414] 2. 直接从常规合并候选列表中进一步推导交错的列表1MV候选和列表0MV候选,其中,列表1MV候选的优先级高于列表0MV候选。也应用相同的利用自适应阈值的剪除方法来去除冗余MV候选。

[0415] 3. 填充零MV候选,直到GPM候选列表满为止。

[0416] 仅针对一个GPM CU启用GPM-MMVD和GPM-TM。这是通过首先用信号传输GPM-MMVD语法来实现的。当两个GPM-MMVD控制标志都等于假时(即,针对两个GPM分区禁用GPM-MMVD),用信号传输GPM-TM标志以指示模板匹配是否被应用于两个GPM分区。否则(至少一个GPM-MMVD标志等于真),GPM-TM标志的值被推断为假。

[0417] 将图片分区为CTU

[0418] 将图片划分为一系列编码树单元(CTU)。CTU的概念与HEVC[6][7]的概念相同。对于具有三个样点阵列的图片,CTU由N×N亮度样点块连同两个对应的色度样点块构成。图31示出了划分为CTU的图片的示例。

[0419] CTU中允许的最大亮度块尺寸指定为128×128(尽管亮度变换块的最大尺寸为64×64)。

[0420] 将图片分区为子图片、条带、瓦片

[0421] 将图片划分为一个或多个瓦片行和一个或多个瓦片列。瓦片是覆盖图片的矩形区域的一系列CTU。

[0422] 条带由整数数量的完整瓦片或图片中瓦片内整数数量的连续完整CTU行构成。

[0423] 支持两种条带模式,即光栅扫描条带模式和矩形条带模式。在光栅扫描条带模式

中,条带包含图片的瓦片光栅扫描中的一系列完整瓦片。在矩形条带模式中,条带包含共同形成图片的矩形区域的多个完整瓦片或共同形成图片的矩形区域的一个瓦片的多个连续完整CTU行。矩形条带内的瓦片在对应于所述条带的矩形区域内按瓦片光栅扫描顺序进行扫描。

[0424] 子图片包含共同覆盖图片的矩形区域的一个或多个条带。

[0425] 图32示出了图片的光栅扫描条带分区的示例,其中,图片被划分为12个瓦片和3个光栅扫描条带。

[0426] 图33示出了图片的矩形条带分区的示例,其中,图片被划分为24个瓦片(6个瓦片列和4个瓦片行)和9个矩形条带。

[0427] 图34示出了被分区为瓦片和矩形条带的图片的示例,其中,图片被划分为4个瓦片(2个瓦片列和2个瓦片行)和4个矩形条带。

[0428] 图35示出了图片的子图片分区的示例,其中,图片被分区为18个瓦片,左侧的12个瓦片各自覆盖4乘4个CTU的一个条带,并且右侧的6个瓦片各自覆盖2乘2个CTU的垂直堆叠的2个条带,从而总共产生24个条带和24个不同尺寸的子图片(每个条带是子图片)。

[0429] 360度视频编解码工具

[0430] 水平环绕运动补偿

[0431] VTM10中的水平环绕运动补偿是被设计成提高等矩形(ERP)投影格式[5]的重建的360度视频的视觉质量的360特定性编解码工具。在传统的运动补偿中,当运动矢量参考超出参考图片的图片边界的样点时,应用重复填充以通过从对应图片边界上的那些最近的邻近块复制来推导界外样点的值。对于360度视频,这种重复填充的方法是不合适的,并且可能在重建的视口视频中引起被称为“接缝伪影”的视觉伪影。因为360度视频是在球体上捕获的并且固有地没有“边界”,所以在投影域中的参考图片的边界之外的参考样点总是可以从球面域中的邻近样点中获得。对于一般的投影格式,可能难以推导球面域中对应的邻近样点,因为其涉及2D到3D和3D到2D坐标转换[5]以及分数样点位置的样点内插。对于ERP投影格式的左边界和右边界,这个问题要简单得多,因为可以从右图片边界内的样点中获得左图片边界外的球面邻近块,反之亦然。鉴于ERP投影格式的广泛使用和相对易于实施,在VVC中采用水平环绕运动补偿,以提高以ERP投影格式编解码的360视频的视觉质量。

[0432] 水平环绕运动补偿过程如图36所示。当参考块的一部分在投影域中的参考图片的左(或右)边界之外时,“边界外”部分不是重复填充的,而是从位于参考图片内的对应球面邻近块向投影域中的右(或左)边界截取的。重复填充仅用于图片的顶部边界和底部边界。如图36中所描绘的,可以将水平环绕运动补偿与360度视频编解码中经常使用的非规范填充方法组合(参见[5]中的已填充ERP)。在VVC中,这是通过用信号传输高级语法元素来指示环绕偏移来实现的,所述环绕偏移应设置为填充之前的ERP图片宽度;该语法用于相应地调整水平环绕的位置。该语法不受左图片边界和右图片边界上特定填充量的影响,并且因此自然支持ERP图片的不对称填充(即,当左填充和右填充不同时)。当参考样点在参考图片的左边界和右边界之外时,水平环绕运动补偿为运动补偿提供更有意义的信息。在360视频CTC[4]下,该工具不仅在率失真性能方面提高了压缩性能,而且在减少接缝伪影和提高重建的360度视频的主观质量方面也提高了压缩性能。水平环绕运动补偿也可以用于在水平方向上具有恒定采样密度的其他单面投影格式,如360Lib[5]中的经调整等面积投影。

[0433] 表6:VVC中的亮度整数样点获取过程。

8.5.6.3.3 亮度整数样点获取过程

该过程的输入是:

- 以全样点单位表示的亮度位置 $(xInt_L, yInt_L)$,
- 亮度参考样点阵列 $refPicLX_L$,
- 指示是否启用水平环绕运动补偿的变量 $refWraparoundEnabledFlag$ 。

该过程的输出是预测的亮度样点值 $predSampleLX_L$

将可变移位设置为等于 $Max(2, 14 - BitDepth)$ 。

将变量 $picW$ 设置为等于参考图片 $refPicLX$ 的 $pps_pic_width_in_luma_samples$, 并且将变量 $picH$ 设置为等于参考图片 $refPicLX$ 的 $pps_pic_height_in_luma_samples$ 。

将以全样点单位表示的亮度位置 $(xInt, yInt)$ 推导如下:

[0434] - 如果 $sps_subpic_treated_as_pic_flag[CurrSubpicIdx]$ 等于1, 并且参考图片 $refPicLX$ 的 $sps_num_subpics_minus1$ 大于0, 则以下适用:

$$xInt = Clip3(SubpicLeftBoundaryPos, SubpicRightBoundaryPos, refWraparoundEnabledFlag ? ClipH((PpsRefWraparoundOffset) * MinCbSizeY, picW, xInt_L) : xInt_L) \quad (928)$$

$$yInt = Clip3(SubpicTopBoundaryPos, SubpicBotBoundaryPos, yInt_L) \quad (929)$$

- 否则 ($sps_subpic_treated_as_pic_flag[CurrSubpicIdx]$ 等于0, 或者参考图片 $refPicLX$ 的 $sps_num_subpics_minus1$ 等于0), 以下适用:

$$xInt = Clip3(0, picW - 1, refWraparoundEnabledFlag ? ClipH((PpsRefWraparoundOffset) * MinCbSizeY, picW, xInt_L) : xInt_L) \quad (930)$$

$$yInt = Clip3(0, picH - 1, yInt_L) \quad (931)$$

将预测亮度样点值 $predSampleLX_L$ 推导如下:

$$predSampleLX_L = refPicLX_L[xInt][yInt] \ll shift \quad (932)$$

[0435] 在现有的视频编解码器(如HEVC和VVC)中,帧间模式编码CU(以下称为帧间CU)的参考块可能部分或全部位于参考图片之外。这是因为重复填充过程被应用于参考图片以生成围绕每个参考图片的参考像素。

[0436] 在VVC规范中,填充过程是通过改变整数参考样点获取过程来实施的(如表6中所图示)。每当要获取的整数参考样点位于参考图片之外时,都会替代地使用参考图片内最近的整数参考样点。

[0437] 由于参考图片已填充,如图27中所图示的,帧间CU的参考块部分地或全部地位于参考图片之外是有效的。在图27中,执行双向运动补偿以生成当前块的帧间预测块。在该示例中,列表0参考块部分地是边界外的(OOB),而列表1参考块完全在参考图片内。

[0438] 当对帧间块进行双向预测或甚至多向预测时,最终预测值只是两个或多个运动补偿预测块的平均值;当进一步应用BWC时,将不同的权重分别应用于列表0和列表1MC预测值,以生成最终预测值。然而,运动补偿块的边界外(OOB)部分的预测效果较差,因为OOB部

分只是由参考图片内的边界像素生成的重复模式。然而,在现有的视频编解码器中,在帧间预测中没有特别考虑MC块的有效性较低的00B部分。类似地,在VVC中,也可以使用环绕运动补偿。在环绕MC的方案中可以观察到相同的问题。

[0439] 在本公开中,提出了用于通过考虑MC块的有效性较低的00B部分来改进帧间预测的几种方法。此外,“块”用于描述本公开的概念,并且“块”可以容易地被现有编解码器中使用的任何具体定义所替代。例如,“块”可以是现有编解码器中的编码单元(CU)、预测单元(PU)、变换单元(TU)、编码树单元(CTU)、编码块(CB)、预测块(PB)、变换块(TB)、编码树块(CTB)、子CU、子PU或任何其他定义明确的术语。

[0440] 可以单独或组合应用以下方法。

[0441] 在根据本公开的一个实施例中,当组合由运动补偿过程生成的两个或多于两个预测块时,将附加权重应用于预测值;附加权重是基于预测值样点是否是00B来推导的。基本概念是,当执行两个预测值块的加权平均时,由于00B预测值样点的有效性较低,因此给予所述00B预测值样点的权重较小。

[0442] 在一种方案中,通过使用以下方程对预测值进行加权平均来生成双向帧间编码块的最终预测值样点:

$$[0443] \quad P_{i,j}^{final} = (w_{i,j}^{L0} P_{i,j}^{L0} + w_{i,j}^{L1} P_{i,j}^{L1} + o_{offset}) \gg shift$$

[0444] 其中, $P_{i,j}^{L0}$ 和 $P_{i,j}^{L1}$ 分别是根据运动补偿过程从列表0和列表1参考图片中推导的预测值样点; $w_{i,j}^{L0}$ 和 $w_{i,j}^{L1}$ 是与由00B条件推导的对应预测值相关联的权重; $shift$ 是平均因子,并且当其是两个预测值的平均值时,其被设置为1, o_{offset} 是舍入偏移,其通常被设置为 $1 \ll (shift-1)$ 。

[0445] 如图28中所图示的,通过从参考图片执行运动补偿来推导块的预测值样点。当前块的运动矢量用于定位参考块。应注意,当参考块位于整数参考样点之间的分数位置处时,进一步应用分数样点内插过程来推导预测样点。如VVC中的子条款8.5.6.3.2“亮度样点内插滤波过程”中所说明的,分数样点内插首先执行访问垂直邻近8个整数样点的垂直一维内插,然后执行访问水平邻近8个垂直内插样点的水平一维内插。在ECM中,进一步将在VVC中使用的8抽头内插滤波器用12抽头滤波器替换。

[0446] 基于如上所述的运动补偿方案,当用于(通过内插过程)生成预测值样点的整数参考样点中的至少一个位于参考图片之外时,预测值样点 $P_{i,j}^{Lx}$ 被定义为00B。

[0447] 可以使用以下方法推导权重。

[0448] 如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0449] 则将 $w_{i,j}^{L0}$ 设置为0

[0450] 将 $w_{i,j}^{L1}$ 设置为2

[0451] (其与 $P_{i,j}^{final} = P_{i,j}^{L1}$ 相同)

[0452] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0453] 则将 $w_{i,j}^{L0}$ 设置为2

[0454] 将 $w_{i,j}^{L1}$ 设置为0

[0455] (其与 $P_{i,j}^{final} = P_{i,j}^{L0}$ 相同)

[0456] 否则,如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0457] 则将 $w_{i,j}^{L0}$ 设置为1

[0458] 将 $w_{i,j}^{L1}$ 设置为1

[0459] (其与 $(P_{i,j}^{final} = (P_{i,j}^{L0} + P_{i,j}^{L1} + 1) \gg 1)$ 相同)

[0460] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0461] 则将 $w_{i,j}^{L0}$ 设置为1

[0462] 将 $w_{i,j}^{L1}$ 设置为1

[0463] (其与 $(P_{i,j}^{final} = (P_{i,j}^{L0} + P_{i,j}^{L1} + 1) \gg 1)$ 相同)

[0464] 图30示出了根据本公开的编解码方法的一个示例。在步骤3001中,解码器推导用于当前编码块的第一参考图片和第二参考图片。在步骤3002中,解码器使用运动补偿过程基于与第一参考图片相关联的第一运动矢量来从第一参考图片中推导第一预测值样点。在步骤3003中,解码器使用运动补偿过程基于与第二参考图片相关联的第二运动矢量来从第二参考图片中推导第二预测值样点。在步骤3004中,解码器基于第一预测值样点或第二预测值样点中的至少一个和边界外(00B)条件来获得当前编码块中的最终预测值样点。示例方法也可以由编码器来执行。

[0465] 在本公开的另一个实施例中,当块被编码为通过BCW加权平均将由运动补偿过程生成的两个预测块组合的BCW模式时,将附加权重应用于预测值;附加权重是基于预测值是否是00B来推导的。在一种方案中,通过使用以下方程对预测值进行加权平均来生成BCW编码块的最终预测值样点:

$$[0466] \quad P_{i,j}^{final} = \left((8 - w) * w_{i,j}^{L0} P_{i,j}^{L0} + w * w_{i,j}^{L1} P_{i,j}^{L1} + o_{offset} \right) \gg shift$$

[0467] 其中, w 是BCW权重, $P_{i,j}^{L0}$ 和 $P_{i,j}^{L1}$ 分别是根据运动补偿过程从列表0和列表1参考图片中推导的预测值样点; $w_{i,j}^{L0}$ 和 $w_{i,j}^{L1}$ 是与由00B条件推导的对应预测值相关联的权重; $shift$ 是平均因子并且被设置为3以归一化BCW权重, o_{offset} 是舍入偏移,其通常被设置为 $1 \ll (shift - 1)$ 。

[0468] 如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0469] 则将 $w_{i,j}^{L0}$ 设置为0

[0470] 将 $w_{i,j}^{L1}$ 设置为 $8/w$

[0471] (其与 $P_{i,j}^{final} = P_{i,j}^{L1}$ 相同)

[0472] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0473] 则将 $w_{i,j}^{L0}$ 设置为 $8/(8-w)$

[0474] 将 $w_{i,j}^{L1}$ 设置为0

[0475] (其与 $P_{i,j}^{final} = P_{i,j}^{L0}$ 相同)

[0476] 否则,如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0477] 则将 $w_{i,j}^{L0}$ 设置为1

[0478] 将 $w_{i,j}^{L1}$ 设置为1

[0479] (其与 $P_{i,j}^{final} = ((8-w) * P_{i,j}^{L0} + w * P_{i,j}^{L1} + 4) \gg 3$ 相同)

[0480] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0481] 则将 $w_{i,j}^{L0}$ 设置为1

[0482] 将 $w_{i,j}^{L1}$ 设置为1

[0483] (其与 $(P_{i,j}^{final} = ((8-w) * P_{i,j}^{L0} + w * P_{i,j}^{L1} + 4) \gg 3)$ 相同)

[0484] 在本公开的另一个实施例中,当针对块启用BDOF时,将附加权重应用于预测值;附加权重是基于预测值是否是00B来推导的。在一种方案中,通过使用以下方程对预测值进行加权平均来生成启用BDOF的块的最终预测值样点:

[0485]
$$P_{i,j}^{final} = ((w_{i,j}^{L0} P_{i,j}^{L0} + w_{i,j}^{L1} P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j}$$

[0486] 其中, $B_{i,j}$ 是每个预测值样点的BDOF偏移, $P_{i,j}^{L0}$ 和 $P_{i,j}^{L1}$ 分别是根据运动补偿过程从列表0和列表1参考图片中推导的预测值样点; $w_{i,j}^{L0}$ 和 $w_{i,j}^{L1}$ 是与由00B条件推导的对应预测值相关联的权重;shift是平均因子,并且当其是两个预测值的平均值时,其被设置为1, o_{offset} 是舍入偏移,其通常被设置为 $1 \ll (\text{shift} - 1)$ 。

[0487] 如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0488] 则将 $w_{i,j}^{L0}$ 设置为0

[0489] 将 $w_{i,j}^{L1}$ 设置为 $1 \ll \text{shift}$

[0490] (其与 $P_{i,j}^{final} = P_{i,j}^{L1} + B_{i,j}$ 相同)

[0491] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0492] 则将 $w_{i,j}^{L0}$ 设置为 $1 \ll \text{shift}$

[0493] 将 $w_{i,j}^{L1}$ 设置为0

[0494] (其与 $P_{i,j}^{final} = P_{i,j}^{L0} + B_{i,j}$ 相同)

[0495] 否则,如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0496] 则将 $w_{i,j}^{L0}$ 设置为1

[0497] 将 $w_{i,j}^{L1}$ 设置为1

[0498] (其与 $P_{i,j}^{final} = ((P_{i,j}^{L0} + P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j}$ 相同)

[0499] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0500] 则将 $w_{i,j}^{L0}$ 设置为1

[0501] 将 $w_{i,j}^{L1}$ 设置为1

[0502] (其与 $(P_{i,j}^{final} = ((P_{i,j}^{L0} + P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j})$ 相同)

[0503] 在本公开的另一个实施例中,当针对块启用BDOF时,将附加权重应用于预测值;附加权重是基于预测值是否是00B来推导的。在一种方案中,通过使用以下方程对预测值进行加权平均来生成启用BDOF的块的最终预测值样点:

[0504]
$$P_{i,j}^{final} = ((w_{i,j}^{L0} P_{i,j}^{L0} + w_{i,j}^{L1} P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j}$$

[0505] 其中, $B_{i,j}$ 是每个预测值样点的BDOF偏移, $P_{i,j}^{L0}$ 和 $P_{i,j}^{L1}$ 分别是根据运动补偿过程从列表0和列表1参考图片中推导的预测值样点; $w_{i,j}^{L0}$ 和 $w_{i,j}^{L1}$ 是与由00B条件推导的对应预测值相关联的权重;shift是平均因子,并且当其是两个预测值的平均值时,其被设置为1, o_{offset} 是舍入偏移,其通常被设置为 $1 \ll (\text{shift} - 1)$ 。

[0506] 如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0507] 则将 $w_{i,j}^{L0}$ 设置为0

[0508] 将 $w_{i,j}^{L1}$ 设置为 $1 \ll \text{shift}$

[0509] 将 $B_{i,j}$ 设置为0

[0510] (其与 $P_{i,j}^{final} = P_{i,j}^{L1}$ 相同)

[0511] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0512] 则将 $w_{i,j}^{L0}$ 设置为 $1 \ll \text{shift}$

[0513] 将 $w_{i,j}^{L1}$ 设置为0

[0514] 将 $B_{i,j}$ 设置为0

[0515] (其与 $P_{i,j}^{final} = P_{i,j}^{L0}$ 相同)

[0516] 否则,如果 $P_{i,j}^{L0}$ 被确定为00B并且 $P_{i,j}^{L1}$ 被确定为00B

[0517] 则将 $w_{i,j}^{L0}$ 设置为1

[0518] 将 $w_{i,j}^{L1}$ 设置为1

[0519] (其与 $P_{i,j}^{final} = ((P_{i,j}^{L0} + P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j}$ 相同)

[0520] 否则,如果 $P_{i,j}^{L0}$ 被确定为非00B并且 $P_{i,j}^{L1}$ 被确定为非00B

[0521] 则将 $w_{i,j}^{L0}$ 设置为1

[0522] 将 $w_{i,j}^{L1}$ 设置为1

[0523] (其与 $(P_{i,j}^{final} = ((P_{i,j}^{L0} + P_{i,j}^{L1} + o_{offset}) \gg shift) + B_{i,j})$ 相同)

[0524] 基于如上所述的运动补偿方案,当用于(通过内插过程)生成预测值样点的整数参考样点中的至少一个位于参考图片之外时,预测值样点被定义为00B。

[0525] 然而,有时只有少数00B参考整数样点用于通过内插过程生成预测值样点,在这种情况下,该预测值样点仍然可以提供高效的预测(例如,8个参考整数样点中只有2个是00B)。因此,为了提供00B确定的容限,提出了不同的方案来确定预测值样点是否是00B。

[0526] 在本公开的另一个实施例中,当垂直最近的整数参考样点中的至少一个是00B或水平最近的整数参考样点中的至少一个是00B时,将预测值样点确定为00B。例如,如图28中所图示的,预测值 $P_{0,0}^{Lx}$ 的水平最近的整数参考样点是样点d和e。

[0527] 在本公开的另一个实施例中,当用于生成预测值样点的整数样点中的至少N个是00B时,将预测值样点确定为00B,其中,N是任何整数。

[0528] 在本公开的另一个实施例中,当执行两个或多个预测值块的加权平均时,给予使用00B整数参考样点进行内插的预测值样点较小的权重。此外,权重与00B整数参考样点的数量成反比。

[0529] 在本公开的另一个实施例中,当执行两个或多个预测值块的加权平均时,给予使用00B整数参考样点进行内插的预测值样点权重。在比特流中以不同级(如序列级(序列参数集)、图片级(图片参数集)、条带级(条带标头)或块级)用信号传输权重。

[0530] 然而,在本公开中,为了提供灵活的00B确定,提出了以下方案来确定预测值样点是否是00B。在一个或多个实施例中,根据预测值样点是否位于参考图片之外以及预测值样点的分数位置,将预测值样点确定为00B。具体地说,将预测像素在当前块内的位置表示为 $Pos_{i,j}$;将水平位置索引和垂直位置索引进一步表示为 $Pos_{x_{i,j}}$ 和 $Pos_{y_{i,j}}$;将当前块的列表0MV和列表1MV分别表示为 $Mv_{i,j}^{L0}$ 和 $Mv_{i,j}^{L1}$;将MV的水平分量和垂直分量进一步表示为 $Mv_{x_{i,j}}^{Lx}$ 和 $Mv_{y_{i,j}}^{Lx}$ (Lx 可以是分别表示列表0和列表1的L0或L1)。由于在许多现有的视频编解码标准中支持分数MV,如VVC中使用的1/16样点MV(也可以表示为1/16像素MV)、HEVC中使用的1/4样点MV, $Mv_{i,j}^{L0}$ 和 $Mv_{i,j}^{L1}$ 以分数样点(例如1/16样点)为单位表示, $Pos'_{x_{i,j}}$ 和 $Pos'_{y_{i,j}}$ 是当前块内预测像素位置的以分数样点为单位表示的位置索引。 $Pos'_{LeftBdry}$ 、 $Pos'_{RightBdry}$ 、 $Pos'_{TopBdry}$ 、 $Pos'_{BottomBdry}$ 是以分数样点为单位的图片边界的位置索引。

[0531] 请注意,在本公开中,假设了位置索引从图片的左上开始,例如,图片的左上样点的位置索引表示为(0,0);并且位置索引从上到下和从左到右增加。在一种示例性方案中,使用以下条件检查将一个预测值样点确定为00B:

[0532] 在一种示例性方案中,使用以下条件检查将一个预测值样点确定为00B(M、N、O、P是任意整数):

[0533] 当以下条件中的至少一个成立时,将预测值样点 $P_{i,j}^{Lx}$ 视为00B:

$$(Pos_{x'_{i,j}} + Mv_{x_{i,j}}^{Lx}) > (Pos'_{RightBdry} + M),$$

[0534] $(Pos_{x'_{i,j}} + Mv_{x_{i,j}}^{Lx}) < (Pos_{LeftBdry} - N),$

$$(Pos_{y'_{i,j}} + Mv_{y_{i,j}}^{Ly}) > (Pos'_{BottomBdry} + O),$$

[0535] 或 $(Pos_{y'_{i,j}} + Mv_{y_{i,j}}^{Ly}) < (Pos_{TopBdry} - P)$

[0536] 否则,当上述条件均不成立时,将预测值样点 $P_{i,j}^{Lx}$ 视为非00B。

[0537] 图37图示了根据本公开的一个示例的视频编解码过程的流程图。在步骤3701中,编码器或解码器可以推导用于当前编码块的参考图片。在步骤3702中,编码器或解码器可以使用运动补偿过程基于与参考图片相关联的运动矢量来从参考图片中推导预测值样点。在步骤3703中,编码器或解码器可以确定预测值样点是否位于参考图片之外一定外边距处。在步骤3704中,编码器或解码器可以响应于确定预测值样点位于参考图片之外一定外边距处而确定预测值样点是00B。

[0538] 在又一个实施例中,可以基于一组预测值样点来进行预测值样点的00B确定。在一个示例中,根据先前描述的00B确定过程(但不限于所提出的方案), $M \times N$ 个预测值样点的块(其中, M 是块宽度,并且 N 是块高度)共享相同的00B状态(确定为00B或非00B)。

[0539] 考虑00B条件的DMVR

[0540] 为了提高合并模式的MV的准确性,在VVC中应用基于双边匹配(BM)的解码器侧运动矢量细化。在双向预测操作中,在参考图片列表L0和参考图片列表L1中的初始MV周围搜索细化的MV。BM方法计算参考图片列表L0和列表L1中的两个候选块之间的失真。如图17中图示的,基于初始MV周围的每个MV候选计算红色块之间的SAD。具有最低SAD的MV候选成为细化的MV,并且用于生成双向预测信号。在另一个实施例中,当参考图片列表L0或列表L1中的候选块的样点被确定为00B时,所述样点和其他对应候选块的其相关样点不用于计算匹配成本(例如,绝对样点差值和或样点差值平方和)。

[0541] 考虑00B条件的模板匹配

[0542] 模板匹配(TM)是用于通过找到当前图片中的模板(即,当前CU的顶部相邻块和/或左侧相邻块)与参考图片中的参考块(即,与模板尺寸相同)之间的最接近匹配来细化当前CU的运动信息的解码器侧MV推导方法。在另一个实施例中,当参考图片中的参考块的样点被确定为00B时,当前图片中模板的所述样点和其相关样点不用于计算匹配成本(例如,绝对样点差值和或样点差值平方和)。

[0543] 考虑00B条件的MHP

[0544] 在多假设帧间预测模式(JVET-M0425)下,除了传统的双向预测信号之外,还用信号传输一个或多个附加的运动补偿预测信号。通过逐样点加权叠加来获得所得的总体预测信号。利用双向预测信号 p_{bi} 和第一附加帧间预测信号/假设 h_3 , 如下获得所得的预测信号

p_3 :

[0545]
$$p_3 = (1 - \alpha) p_{bi} + \alpha h_3$$

[0546] 在另一个实施例中,当附加运动补偿预测的样点被确定为00B时,所述样点不用于生成MHP模式的所得总体预测信号。

[0547] 考虑00B条件的LIC

[0548] LIC是用于将当前块与其预测块之间的局部照明变化建模为函数的帧间预测技术,所述函数从当前块模板与当前块模板的参考块之间的像素推导。所述函数的参数可以用比例 α 和偏移 β 表示,其形成了用于补偿照明变化的线性方程(即, $\alpha * p[x] + \beta$),其中, $p[x]$ 是在参考图片上的位置 x 处的MV指向的参考样点。由于 α 和 β 可以基于当前块模板和当前块模板的参考块来推导,因此除了用信号传输指示使用LIC的LIC标志之外,它们不需要信令开销。在另一个实施例中,在当前块模板的参考块的样点被确定为OOB时,当前块模板的所述样点和其相关样点不用于推导LIC参数。

[0549] 仿射PROF

[0550] 在另一个实施例中,当执行PROF时,当用于推导水平/垂直梯度值的样点包含OOB参考样点时,PROF偏移被设置为零。

[0551] 在另一个实施例中,当执行PROF时,当用于推导水平/垂直梯度值的样点包含OOB参考样点时,水平/垂直梯度值被设置为零。

[0552] 考虑OOB条件的单向预测

[0553] 单向预测方法类似于使用单个运动矢量预测的帧间预测块。在本公开的一个或多个示例中,对于单向预测块,使用替代预测值来替换OOB预测值样点。在一种方案中,通过使用在VVC或HEVC中的时间MV预测(TMVP)的推导中使用的类似缩放方法、根据当前图片、参考图片和目标参考图片之间的时间距离缩放当前块的mv而指向的其他参考图片中的预测值来推导替代预测值。

[0554] 如图38所示,根据本公开的一个示例,提供了一种用于对双向预测编码块进行解码的方法。在步骤3801中,解码器推导用于当前编码块的第一参考图片。在步骤3802中,解码器使用运动补偿过程基于与第一参考图片相关联的运动矢量来从第一参考图片中推导第一预测值样点。在步骤3803中,解码器确定第一预测值样点是否是边界外的(OOB),并且响应于确定第一预测值样点是OOB,解码器用第一替代预测值样点替换作为OOB的第一预测值样点。在步骤3804中,解码器推导用于当前编码块的第二参考图片。在步骤3805中,解码器使用运动补偿过程基于与第二参考图片相关联的运动矢量来从第二参考图片中推导第二预测值样点。在步骤3806中,解码器确定第二预测值样点是否是边界外的(OOB),并且响应于确定第二预测值样点是OOB,解码器用第二替代预测值样点替换作为OOB的第二预测值样点。在步骤3807中,解码器通过对第一预测值样点和第二预测值样点的样点值求平均值来生成最终预测样点。

[0555] 图39是图示了根据本公开的一个示例的用于对双向预测编码块进行编码的方法的框图。在步骤3901中,编码器推导用于当前编码块的第一参考图片。在步骤3902中,编码器使用运动补偿过程基于与第一参考图片相关联的运动矢量来从第一参考图片中推导第一预测值样点。在步骤3903中,编码器确定第一预测值样点是否是边界外的(OOB),并且响应于确定第一预测值样点是OOB,编码器用第一替代预测值样点替换作为OOB的第一预测值样点。在步骤3904中,编码器推导用于当前编码块的第二参考图片。在步骤3905中,编码器使用运动补偿过程基于与第二参考图片相关联的运动矢量来从第二参考图片中推导第二预测值样点。在步骤3906中,编码器确定第二预测值样点是否是边界外的(OOB),并且响应于确定第二预测值样点是OOB,编码器用第二替代预测值样点替换作为OOB的第二预测值样点。在步骤3907中,编码器通过对第一预测值样点和第二预测值样点的样点值求平均值来

生成最终预测样点。

[0556] 如图40所示,根据本公开的一个示例,提供了一种用于对单向预测编码块进行解码的方法。在步骤4001中,解码器推导用于当前编码块的第一参考图片。在步骤4002中,解码器使用运动补偿过程基于与第一参考图片相关联的运动矢量来从第一参考图片中推导第一预测值样点。在步骤4003中,解码器确定第一预测值样点是否是边界外的(OOB),并且响应于确定第一预测值样点是OOB,解码器用替代预测值样点替换作为OOB的第一预测值样点。解码器可以基于第二参考图片中的第二预测值样点来推导替代预测值样点。例如,解码器可以通过根据当前图片、第一参考图片和第二参考图片之间的距离缩放当前编码块的运动矢量来推导第二参考图片中的第二预测值样点。

[0557] 图41是图示了根据本公开的一个示例的用于对单向预测编码块进行编码的方法的框图。在步骤4101中,编码器推导用于当前编码块的第一参考图片。在步骤4102中,编码器使用运动补偿过程基于与第一参考图片相关联的运动矢量来从第一参考图片中推导第一预测值样点。在步骤4103中,编码器确定第一预测值样点是否是边界外的(OOB),并且响应于确定第一预测值样点是OOB,编码器用替代预测值样点替换作为OOB的第一预测值样点。

[0558] 所提出的方法的实施方式

[0559] 在实现所提出的方法的相同动机的替代方式中,在第十四实施例中,一个参考列表LX(X可以是0或1)中的OOB预测值可以被另一参考列表L(1-X)中的对应的非OOB预测值替换。在替换之后,双向MC的原始平均过程可以保持不变。下面给出一个示例以具有与第一实施例中所示的结果相同的结果。

[0560] 如果 $P_{i,j}^{L0}$ 被确定为OOB并且 $P_{i,j}^{L1}$ 被确定为非OOB

[0561] 则其与 $(P_{i,j}^{final} = (P_{i,j}^{L1} + P_{i,j}^{L1} + 1) \gg 1)$ 相同

[0562] (并且其还与 $P_{i,j}^{final} = P_{i,j}^{L1}$ 相同)

[0563] 否则,如果 $P_{i,j}^{L0}$ 被确定为非OOB并且 $P_{i,j}^{L1}$ 被确定为OOB

[0564] 则其与 $(P_{i,j}^{final} = (P_{i,j}^{L0} + P_{i,j}^{L0} + 1) \gg 1)$ 相同

[0565] (其与 $P_{i,j}^{final} = P_{i,j}^{L0}$ 相同)

[0566] 否则,如果 $P_{i,j}^{L0}$ 被确定为OOB并且 $P_{i,j}^{L1}$ 被确定为OOB

[0567] 则其与 $(P_{i,j}^{final} = (P_{i,j}^{L1} + P_{i,j}^{L1} + 1) \gg 1)$ 相同

[0568] 否则,如果 $P_{i,j}^{L0}$ 被确定为非OOB并且 $P_{i,j}^{L1}$ 被确定为非OOB

[0569] 则其与 $(P_{i,j}^{final} = (P_{i,j}^{L1} + P_{i,j}^{L1} + 1) \gg 1)$ 相同

[0570] 高级语法信令

[0571] 在另一个实施例中,可以在比特流中用信号传输一个或多个语法元素,以指示考虑OOB条件的所提出的MC的启用/禁用。此外,可以在比特流中用信号传输一个或多个语法元素,以指示在考虑OOB的MC中使用的参数或阈值。可以以序列级(例如,序列参数集)、图片级(例如,图片参数集)、条带级(例如,条带标头)、CTU/CTB级、CU/CB级、PU/PB级用信号传输

所提出的语法元素。此外,启用/禁用控制可以采用分级方案。例如,以序列级用信号传输一个语法元素以指示所提出的OOB方案的启用/禁用。当所述语法元素指示禁用OOB方案时,在比特流中不再用信号传输语法元素;当所述语法元素指示启用OOB方案时,可以以较低级用信号传输更多的语法元素以指示在较低级下启用/禁用OOB。例如,当序列级语法元素指示启用OOB时,用信号传输一个条带级语法元素以指示在条带级下启用/禁用OOB方案。

[0572] 在现有视频编解码标准的运动补偿(MC)方案中,当由运动矢量指示的参考样点位于预定义边界之外时,使用边界样点填充或水平/垂直环绕来启用使用位于边界之外的参考样点的MC过程。预定义边界包括但不限于编码图片边界、子图片边界、条带边界和/或瓦片边界。在另一个实施例中,当可以基于位于如上所述的边界之外的参考样点执行运动补偿时,所提出的MC方案可以应用于任何方案。还应注意,所提出的方案可以在使用参考图片重采样(RPR)时应用。

[0573] 应注意,所提出的考虑OOB条件的运动补偿(MC)方案不限于应用于所提出的实施例中的编解码方法,而是还可以如先前章节中说明的那样应用于执行MC以生成预测值样点的所有帧间工具,如子块MC(如子块TMVP、仿射模式)、OBMC、IBC、SMVD、DMVR等。

[0574] 可以使用包括一个或多个电路的装置来实施上述方法,该一个或多个电路包括专用集成电路(ASIC)、数字信号处理器(DSP)、数字信号处理设备(DSPD)、可编程逻辑设备(PLD)、现场可编程门阵列(FPGA)、控制器、微控制器、微处理器或其他电子部件。所述装置可以将这些电路与用于执行上述方法的其他硬件或软件部件结合使用。上文公开的每个模块、子模块、单元、或子单元都可以使用这一个或多个电路来至少部分地实施。

[0575] 图29示出了与用户界面1650耦接的计算环境1610。计算环境1610可以是数据处理服务器的一部分。计算环境1610包括处理器1620、存储器1630和输入/输出(I/O)接口1640。

[0576] 处理器1620通常控制计算环境1610的整体操作,如与显示、数据获取、数据通信以及图像处理相关联的操作。处理器1620可以包括执行指令的一个或多个处理器,以执行上文描述的方法中的所有或一些步骤。此外,处理器1620可以包括促进处理器1620与其他部件之间的交互的一个或多个模块。处理器可以是中央处理单元(CPU)、微处理器、单片机、图形处理单元(GPU)等。

[0577] 存储器1630被配置为存储各种类型的数据,以支持计算环境1610的操作。存储器1630可以包括预定软件1632。这种数据的示例包括用于在计算环境1610上操作的任何应用程序或方法的指令、视频数据集、图像数据等。存储器1630可以通过使用任何类型的易失性或非易失性存储器设备或其组合来实施,如静态随机存取存储器(SRAM)、电可擦可编程只读存储器(EEPROM)、可擦可编程只读存储器(EPROM)、可编程只读存储器(PROM)、只读存储器(ROM)、磁存储器、闪存存储器、磁盘或光盘。

[0578] I/O接口1640提供处理器1620与外围接口模块(如键盘、点击轮、按钮等)之间的接口。按钮可以包括但不限于主页按钮、开始扫描按钮和停止扫描按钮。I/O接口1640可以与编码器和解码器耦接。

[0579] 在实施例中,还提供了一种非暂态计算机可读存储介质,所述非暂态计算机可读存储介质例如在存储器1630中包括多个程序,所述多个程序可由计算环境1610中的处理器1620执行,用于执行上述方法。在一个示例中,所述多个程序可以由计算环境1610中的处理器1620执行以接收(例如,从图2中的视频编码器20)包括已编码视频信息(例如,表示已编

码视频帧的视频块和/或相关联的一个或多个语法元素等)的比特流或数据流,并且还可以由计算环境1610中的处理器1620执行以根据接收到的比特流或数据流执行上述解码方法。在另一个示例中,所述多个程序可以由计算环境1610中的处理器1620执行以执行上述编码方法来将视频信息(例如,表示视频帧的视频块和/或相关联的一个或多个语法元素等)编码成比特流或数据流,并且还可以由计算环境1610中的处理器1620执行以传输比特流或数据流(例如,传输到图3中的视频解码器30)。替代性地,非暂态计算机可读存储介质可以在其中存储有比特流或数据流,所述比特流或数据流包括由编码器(例如,图2中的视频编码器20)使用例如上文描述的编码方法生成的已编码视频信息(例如,表示已编码视频帧的视频块和/或相关联的一个或多个语法元素等),以供解码器(例如,图3中的视频解码器30)对视频数据进行解码。非暂态计算机可读存储介质可以是例如ROM、随机存取存储器(RAM)、CD-ROM、磁带、软盘、光学数据存储设备等。

[0580] 在实施例中,还提供了一种计算设备,所述计算设备包括:一个或多个处理器(例如,处理器1620);以及非暂态计算机可读存储介质或存储器1630,所述非暂态计算机可读存储介质或存储器中存储有可由所述一个或多个处理器执行的多个程序,其中,所述一个或多个处理器被配置为在执行所述多个程序时执行上述方法。

[0581] 在实施例中,还提供了一种计算机程序产品,所述计算机程序产品例如在存储器1630中包括多个程序,所述多个程序可由计算环境1610中的处理器1620执行,用于执行上述方法。例如,计算机程序产品可以包括非暂态计算机可读存储介质。

[0582] 在实施例中,计算环境1610可以用一个或多个ASIC、DSP、数字信号处理设备(DSPD)、可编程逻辑设备(PLD)、FPGA、GPU、控制器、微控制器、微处理器或其他电子部件实施,用于执行上述方法。

[0583] 已经出于说明目的呈现了对本公开的描述,并且所述描述不旨在是穷举的、或限于本公开。受益于前述描述和相关联附图中呈现的教导,许多修改、变体和替代性实施方式对于本领域普通技术人员将是显而易见的。

[0584] 除非另有特别说明,否则根据本公开的方法的步骤顺序仅旨在是说明性的,并且根据本公开的方法的步骤不限于上述具体描述的顺序,而是可以根据实际条件进行改变。另外,根据本公开的方法的至少一个步骤可以根据实际需要进行调整、组合或删除。

[0585] 选择并描述示例以便解释本公开的原理,并且使本领域其他技术人员能够理解本公开的各种实施方式并且最好地利用基本原理以及具有适合于预期的特定用途的各种修改的各种实施方式。因此,应当理解,本公开的范围不应受限于所公开的实施方式的具体示例,并且修改和其他实施方式旨在包括在本公开的范围內。

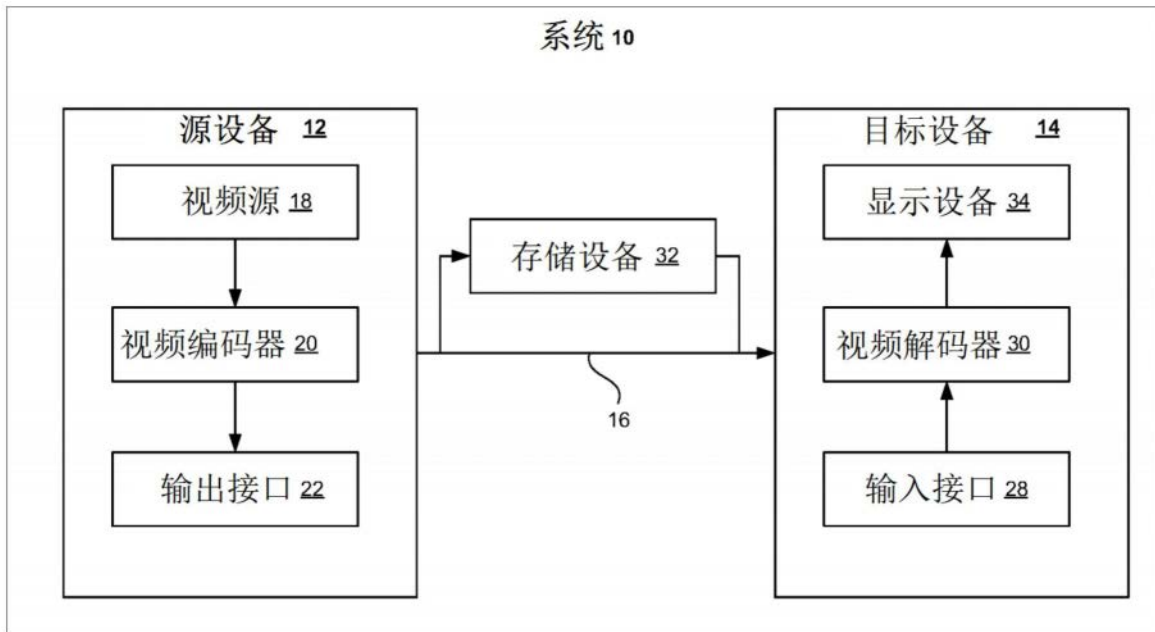


图1

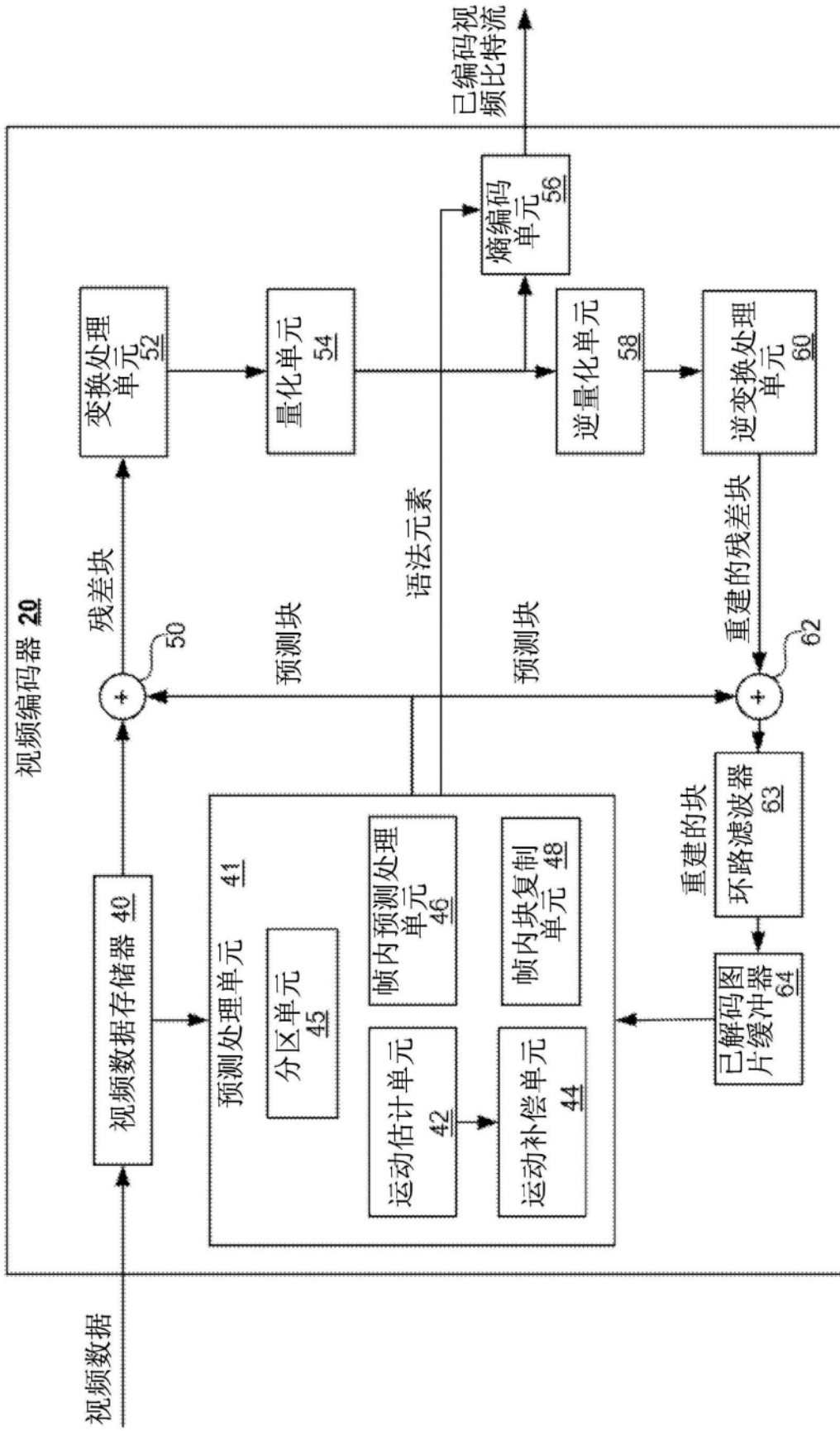


图2

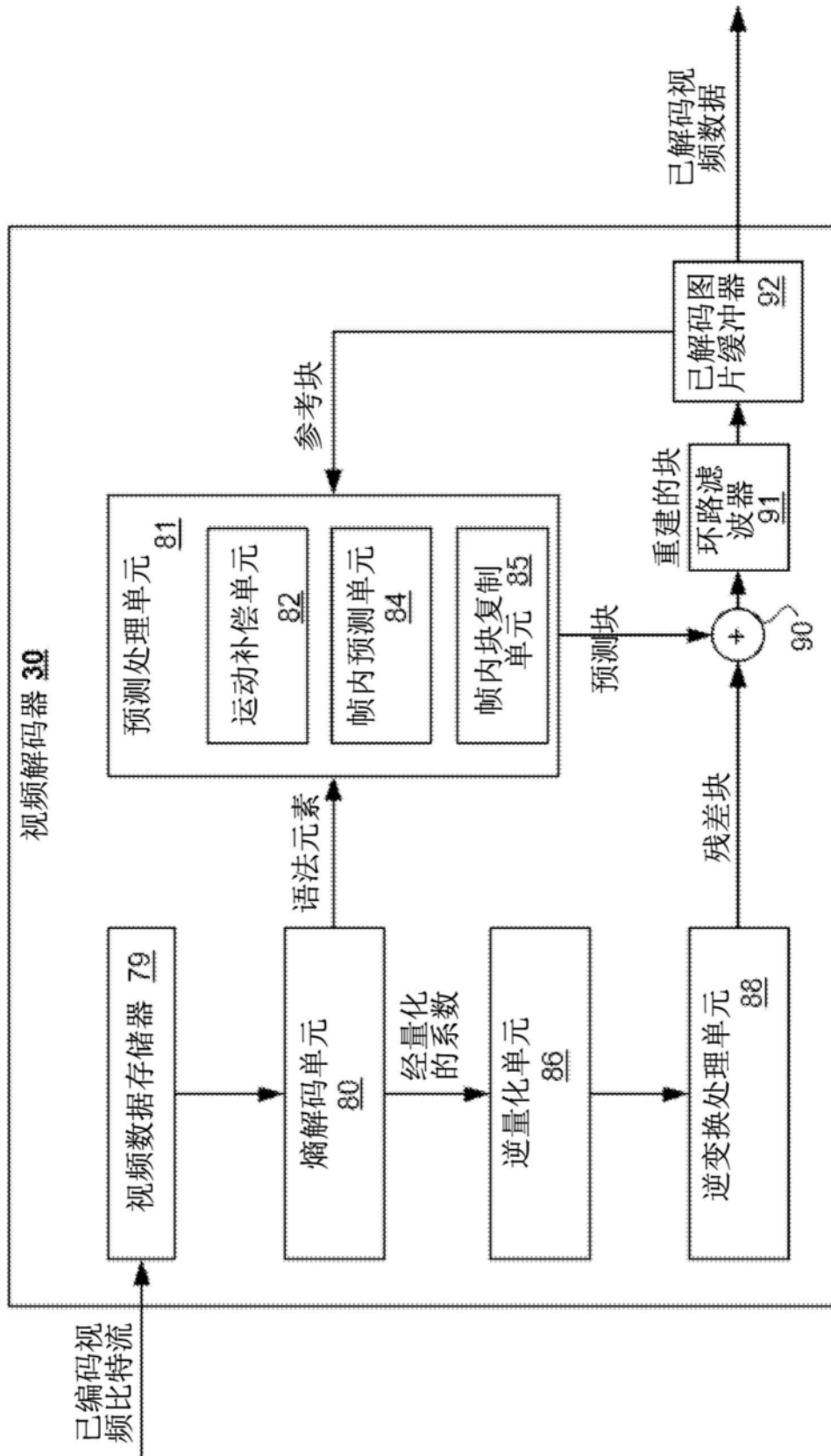


图3

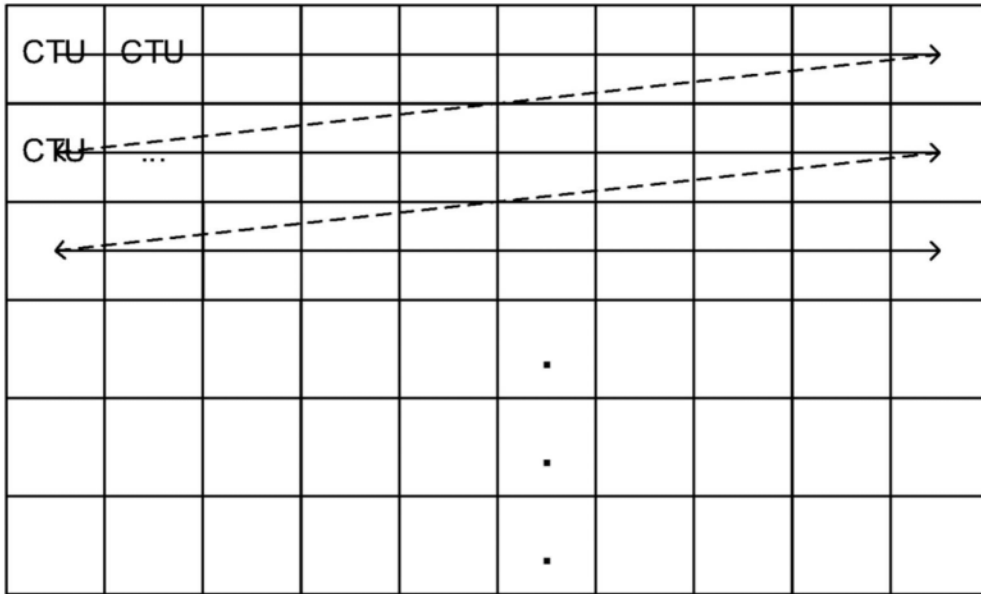


图4A

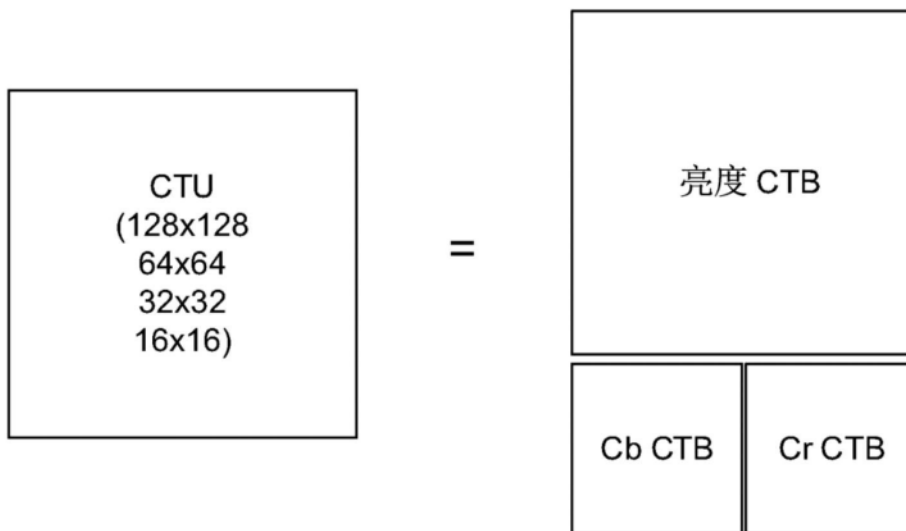


图4B

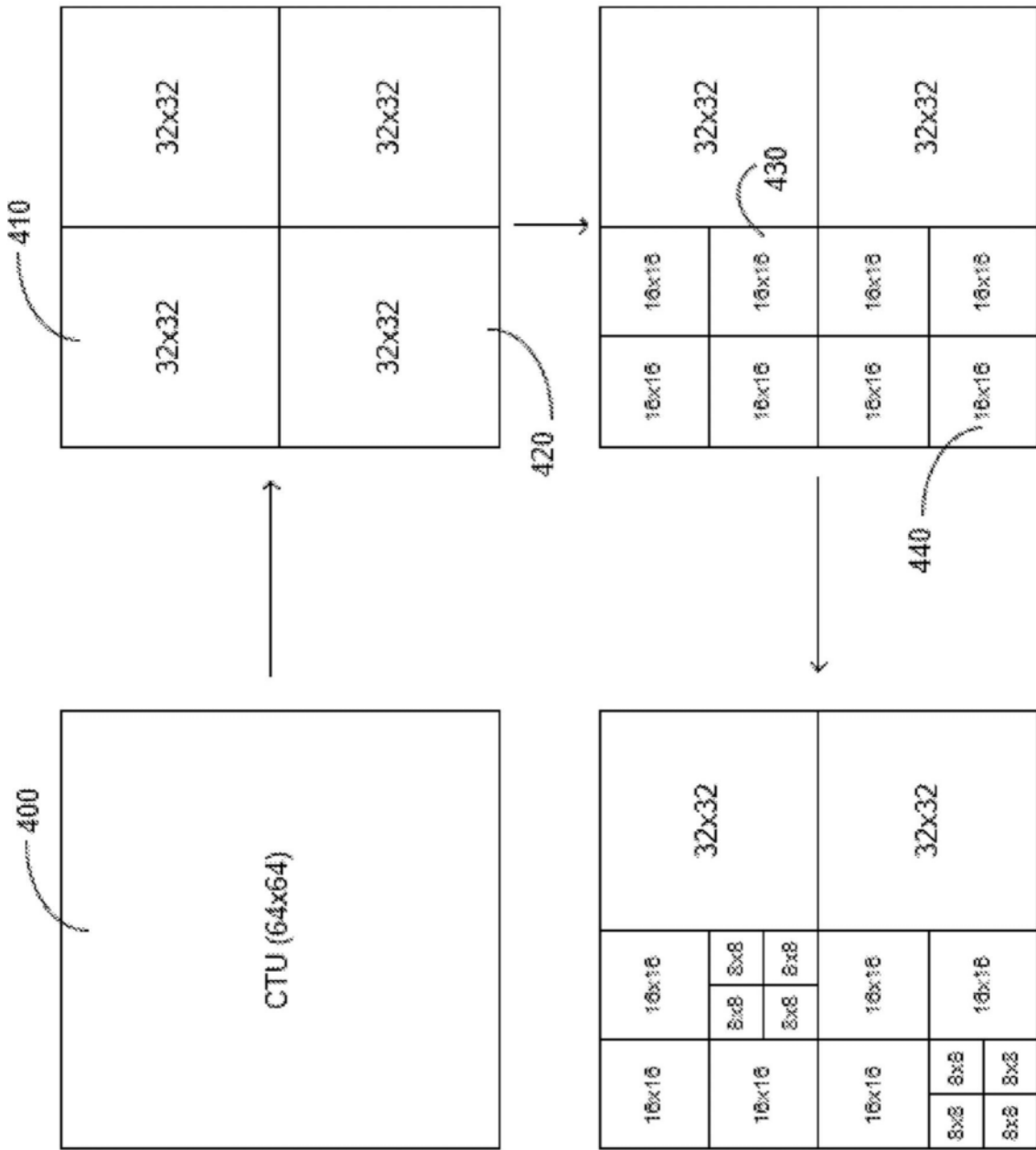


图4C

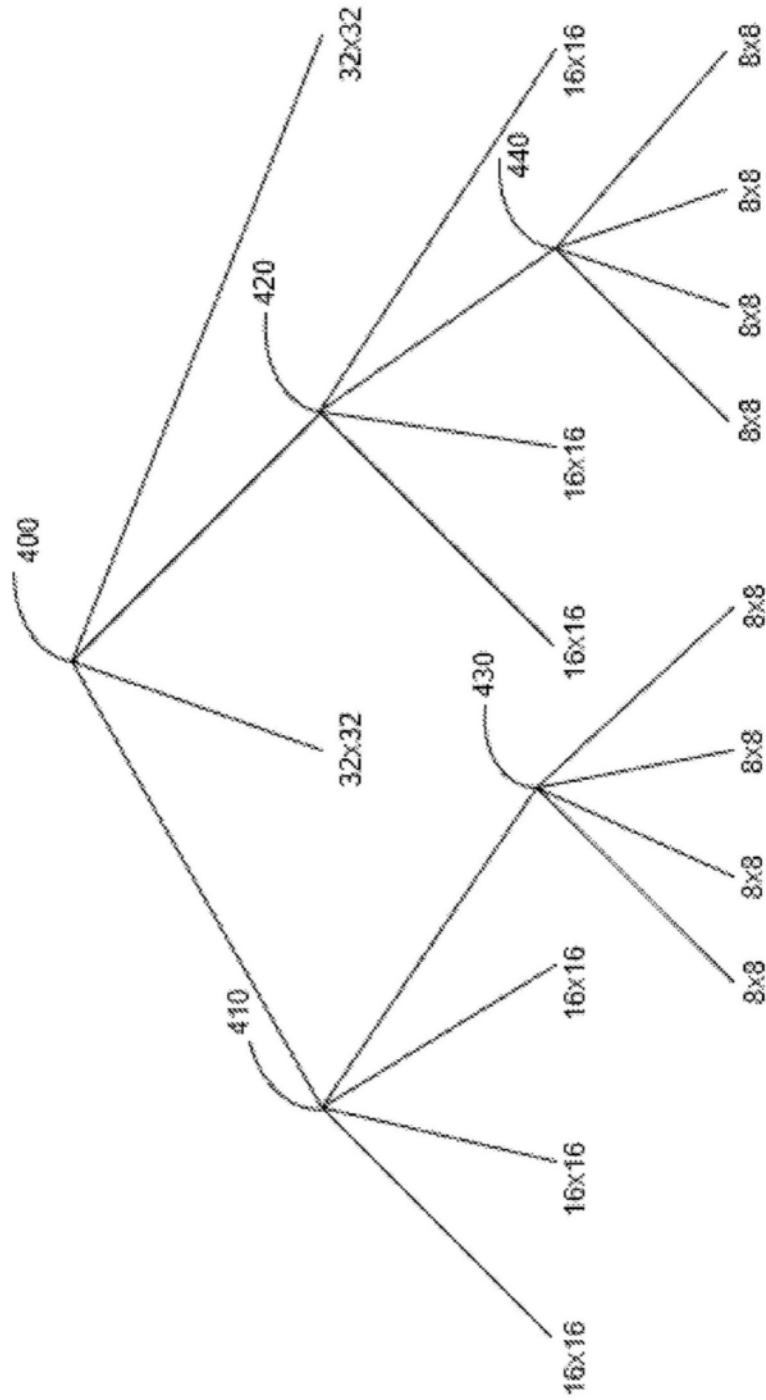


图4D

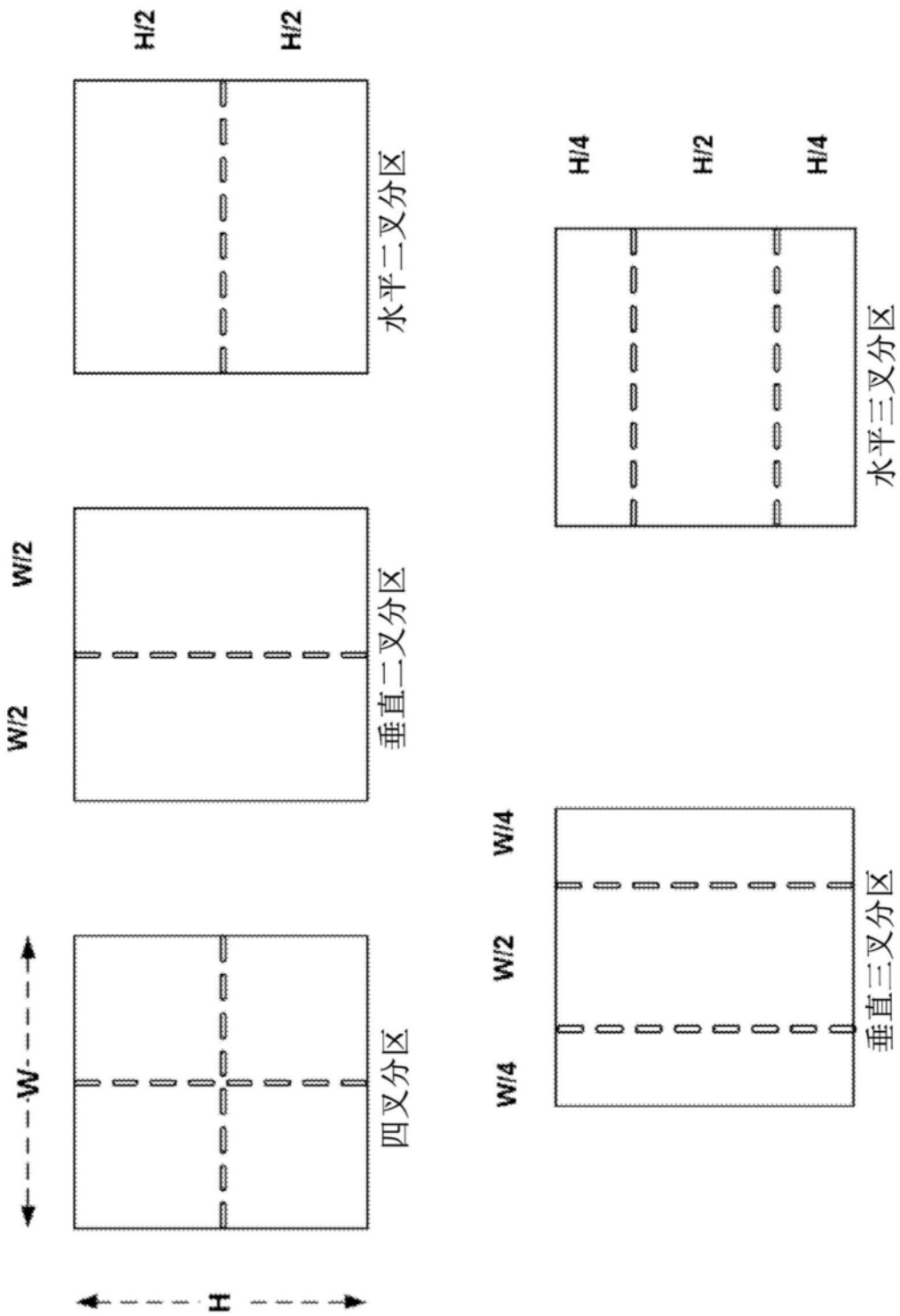


图4E

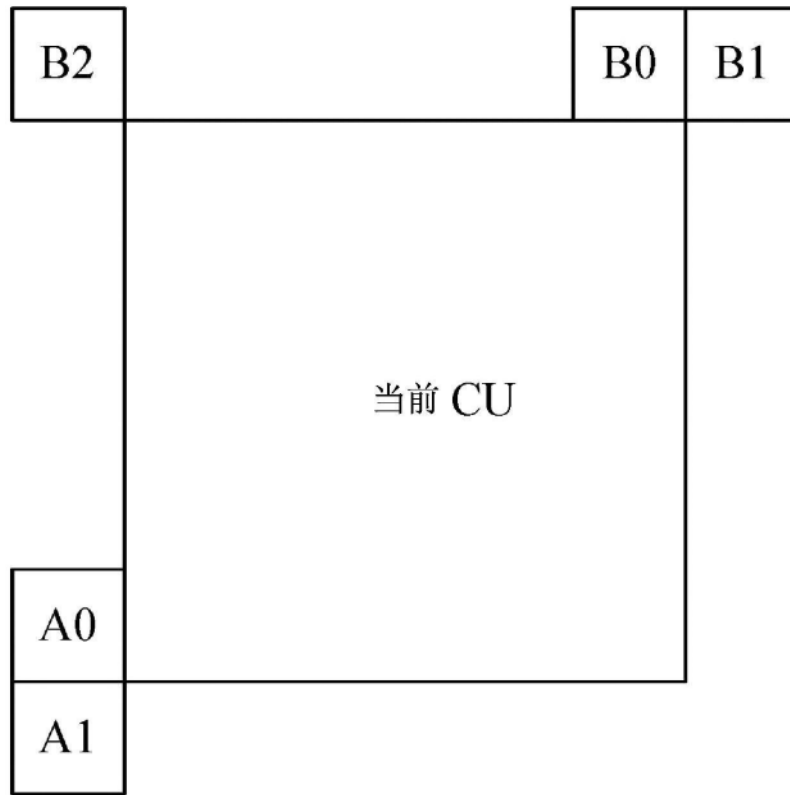


图5

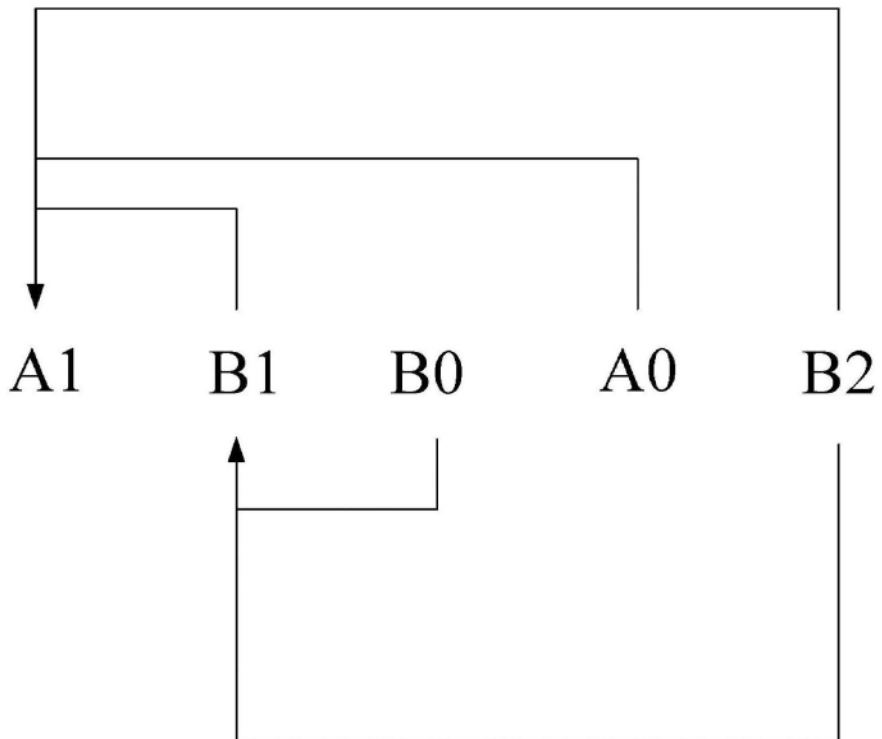


图6

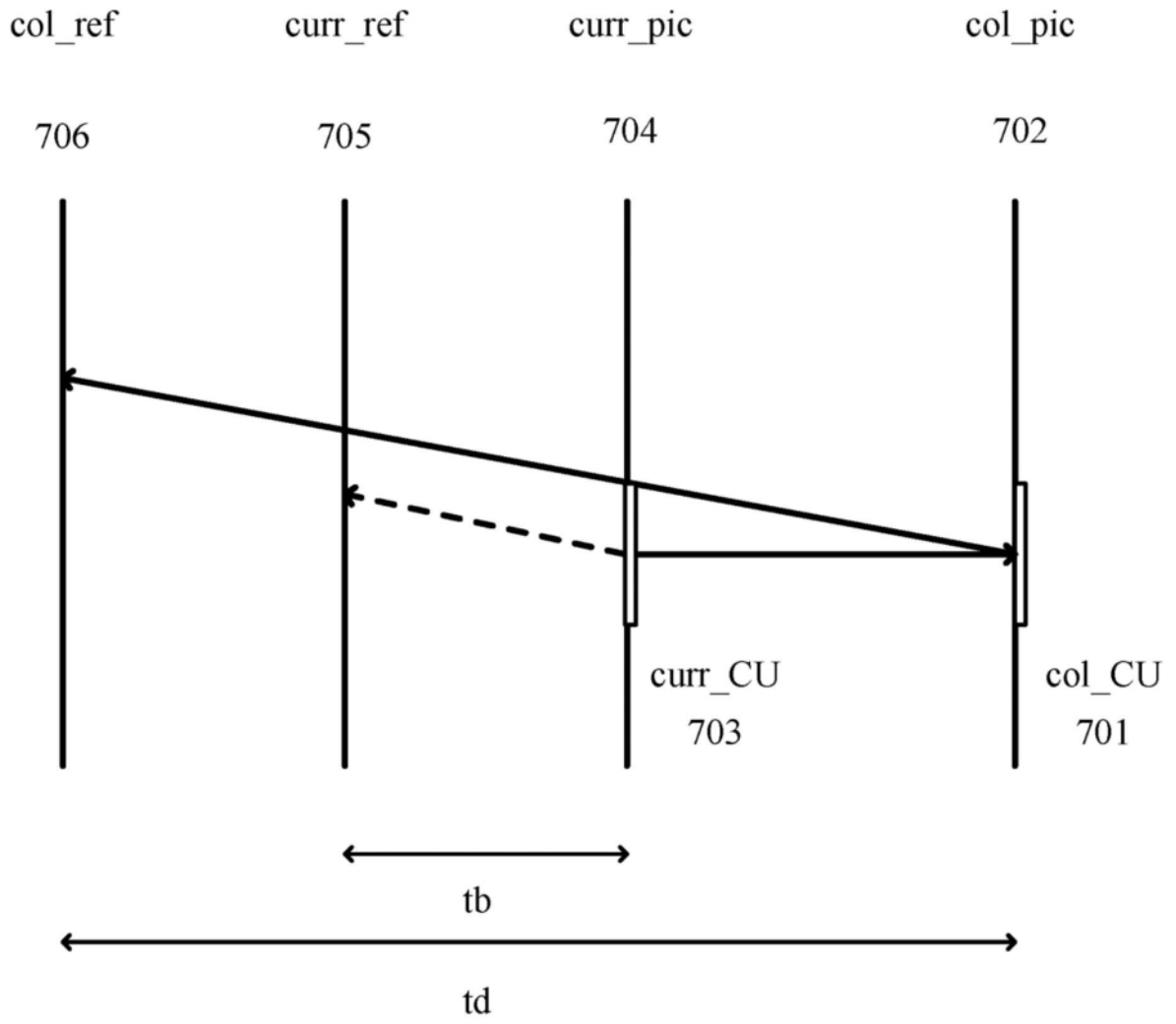


图7

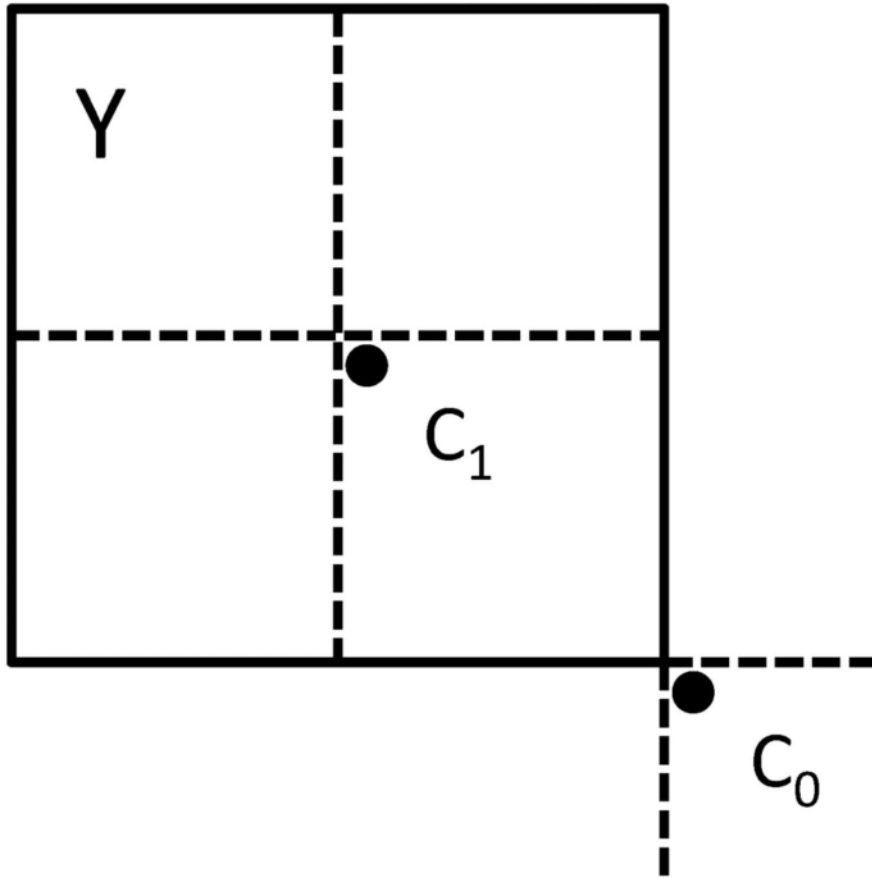


图8

L0 参考图片

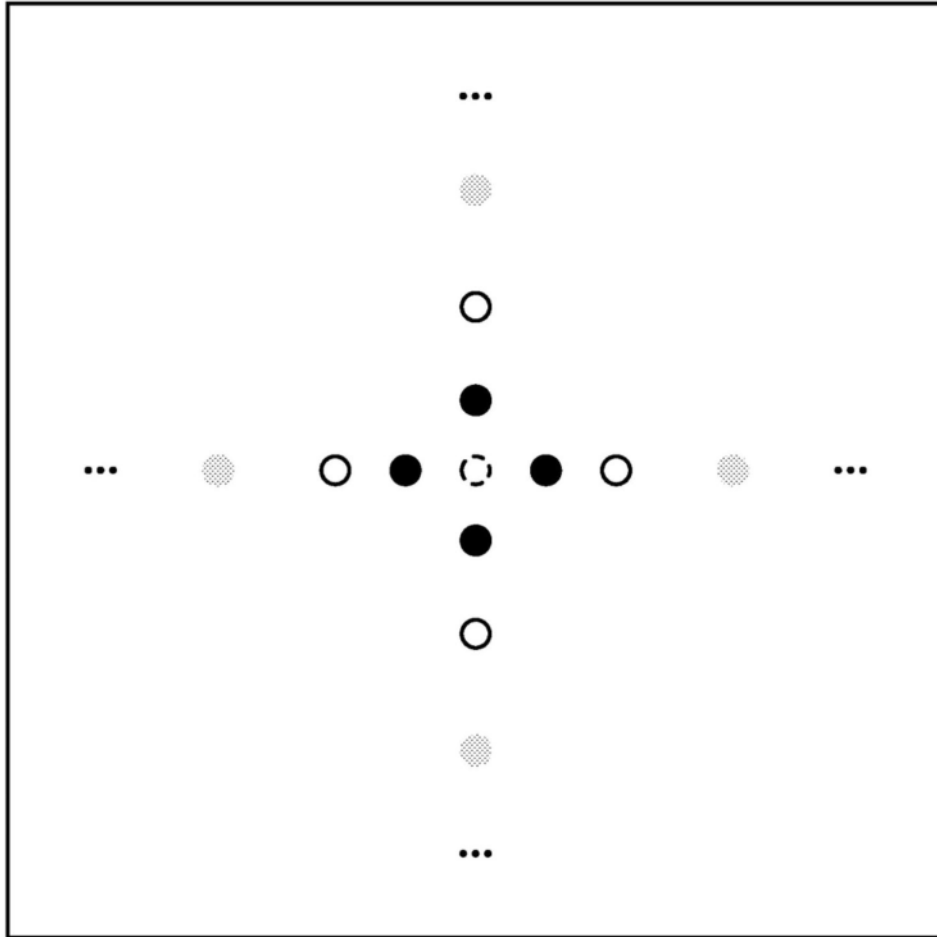


图9A

L1 参考图片

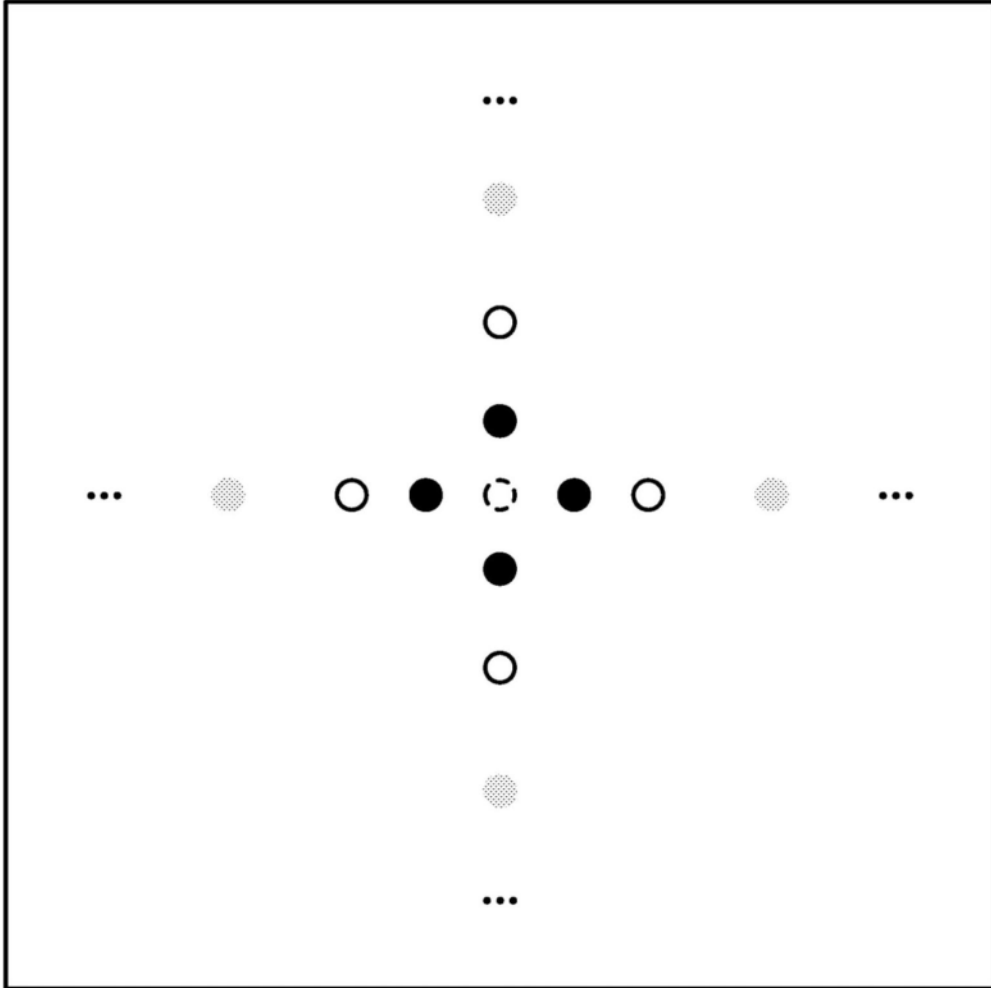


图9B

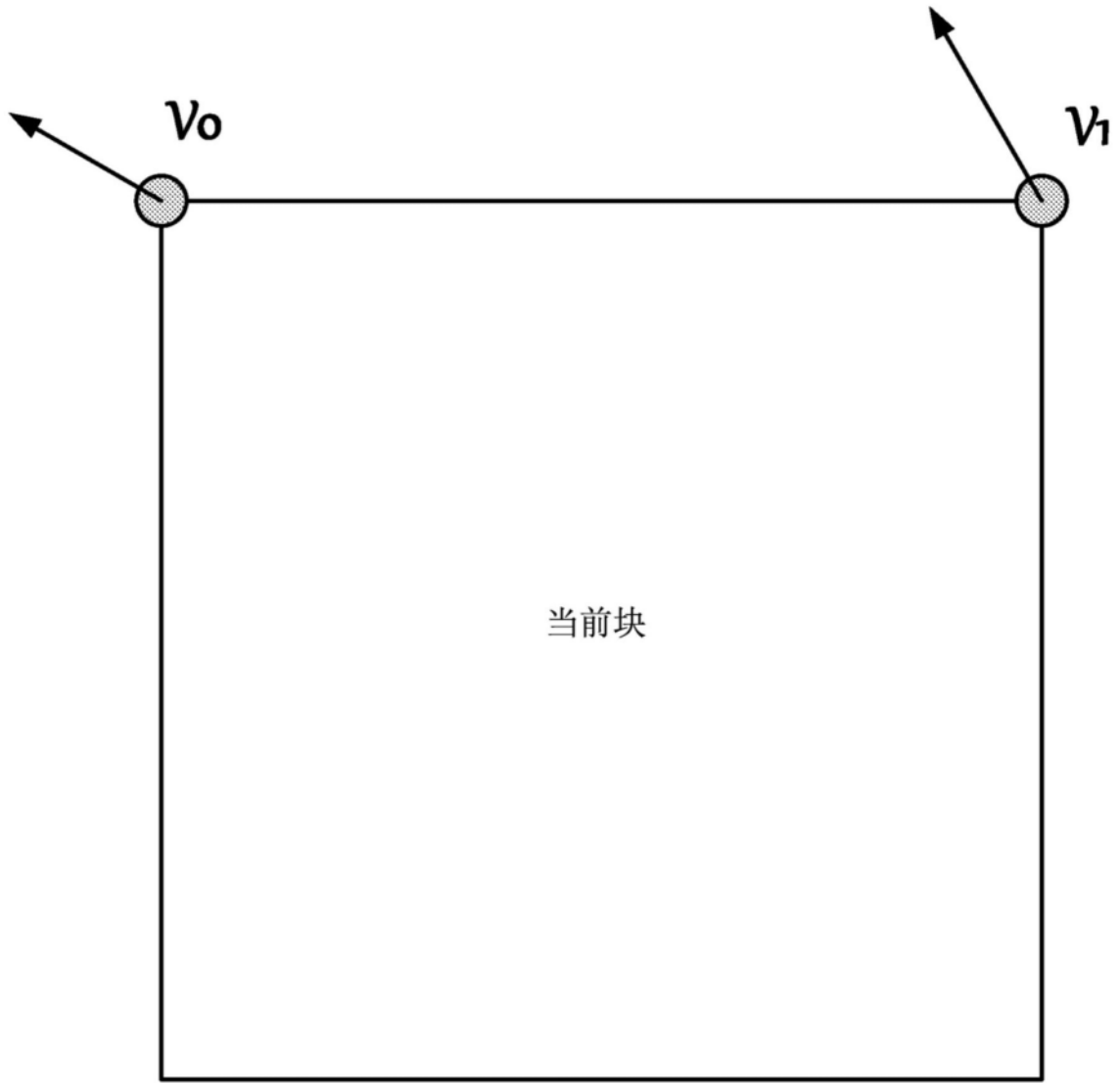


图10A

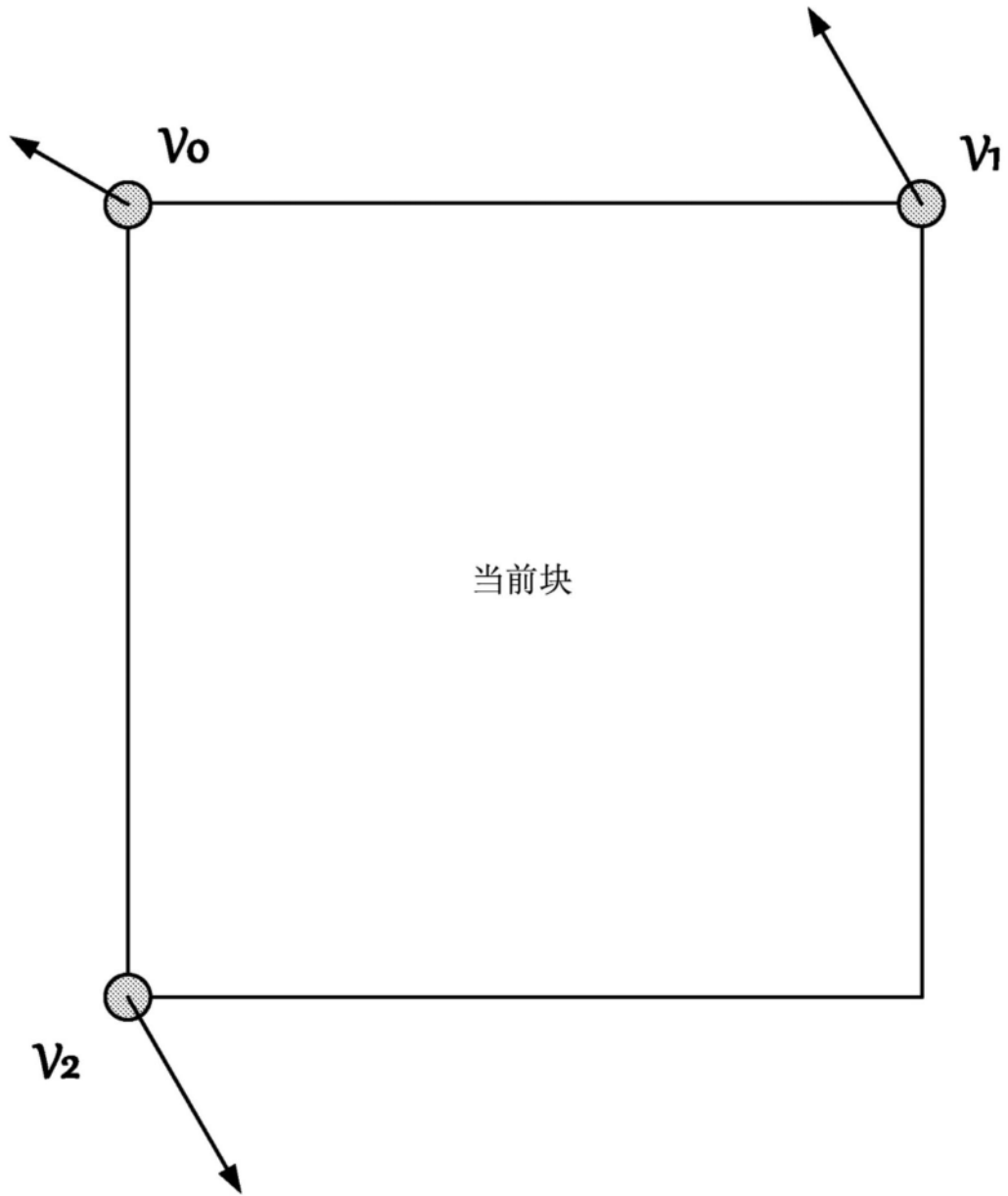


图10B

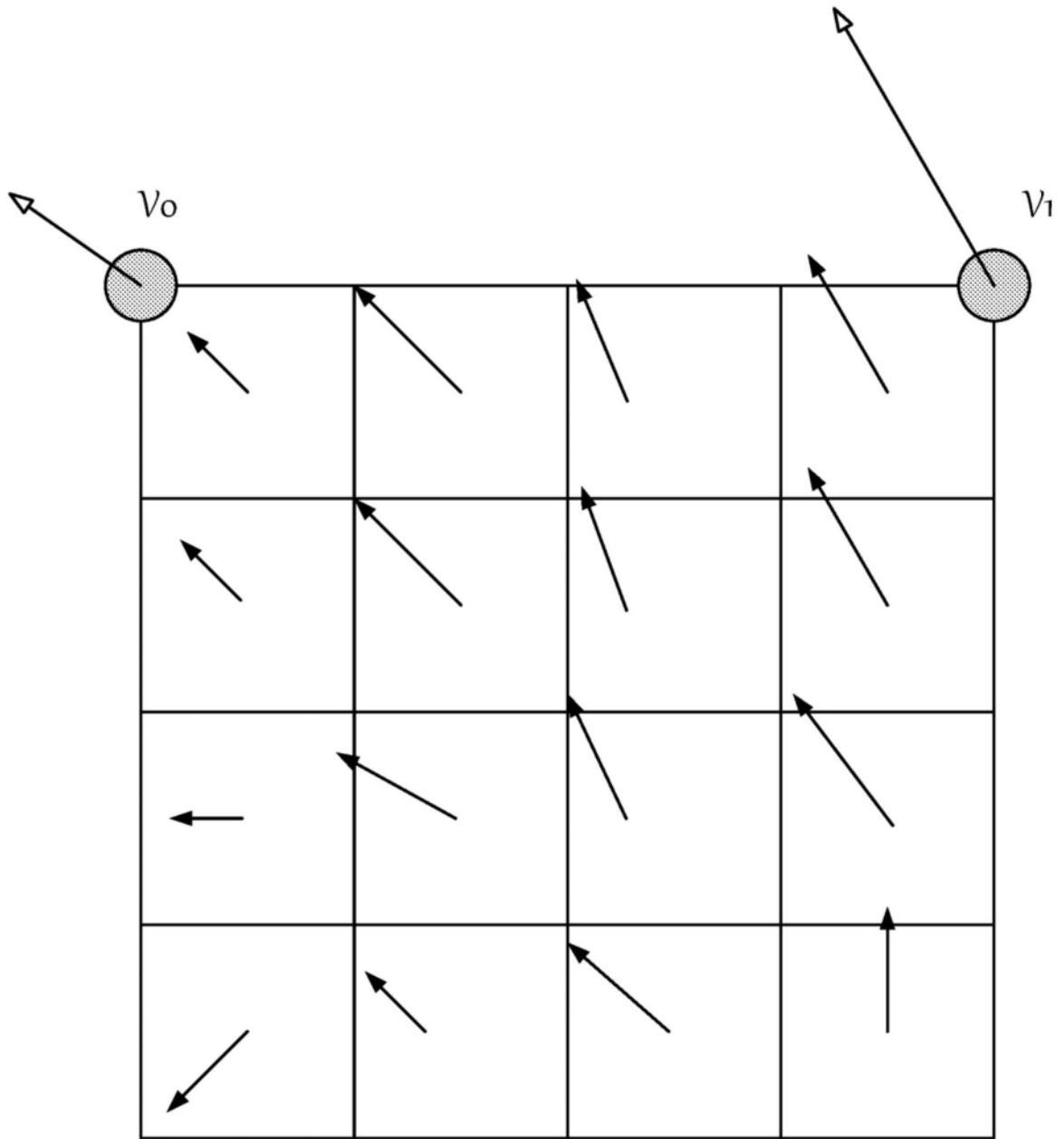


图11

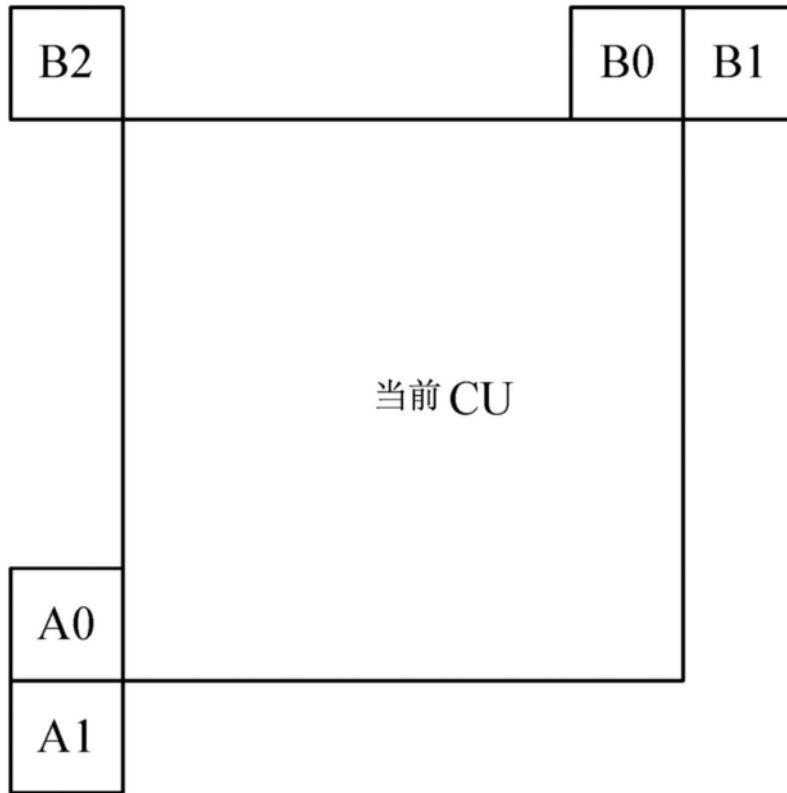


图12

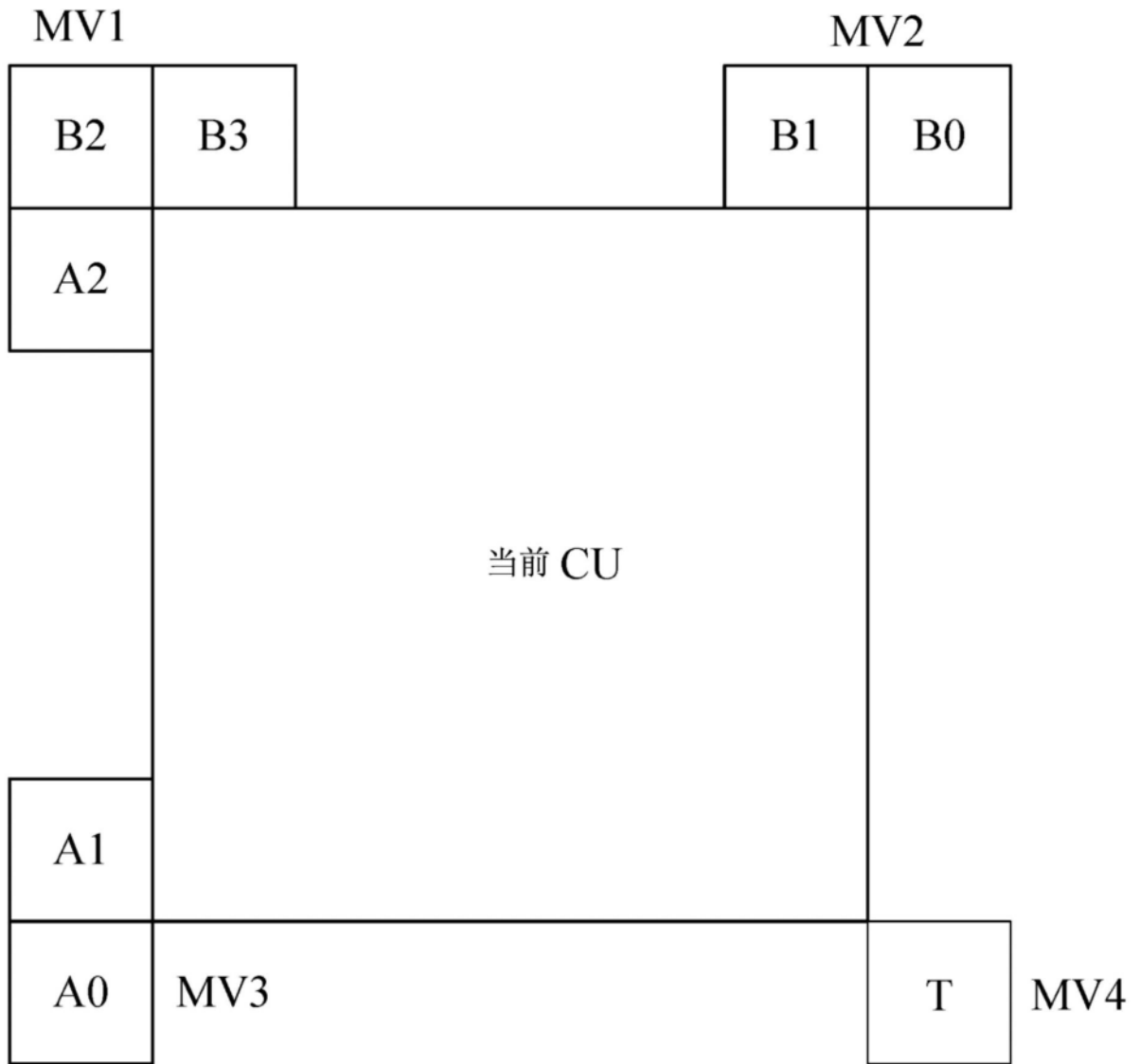


图13

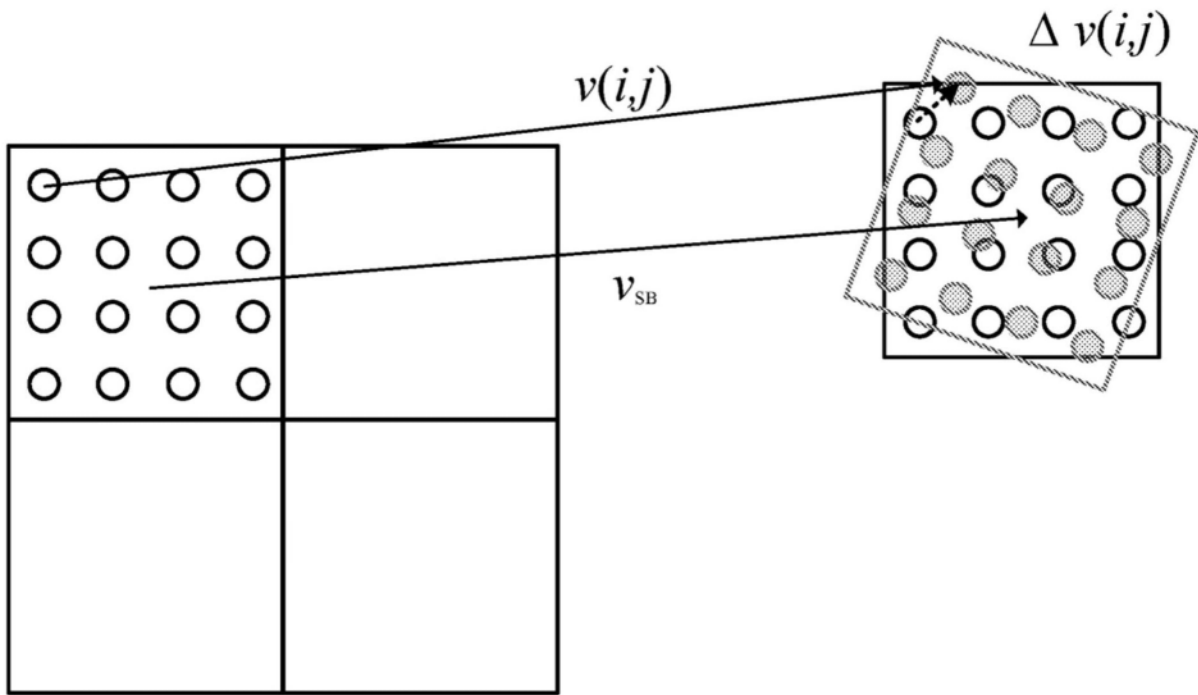


图14

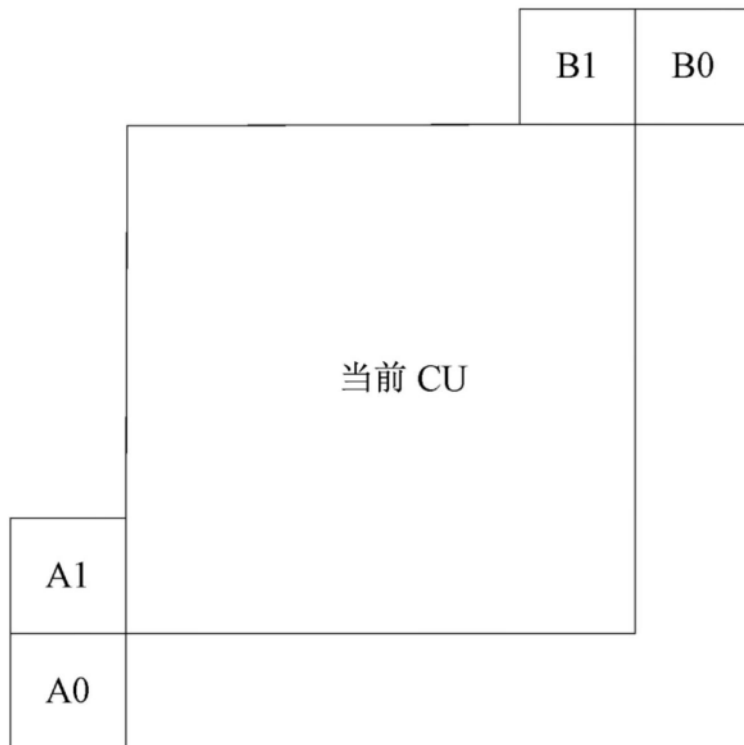


图15A

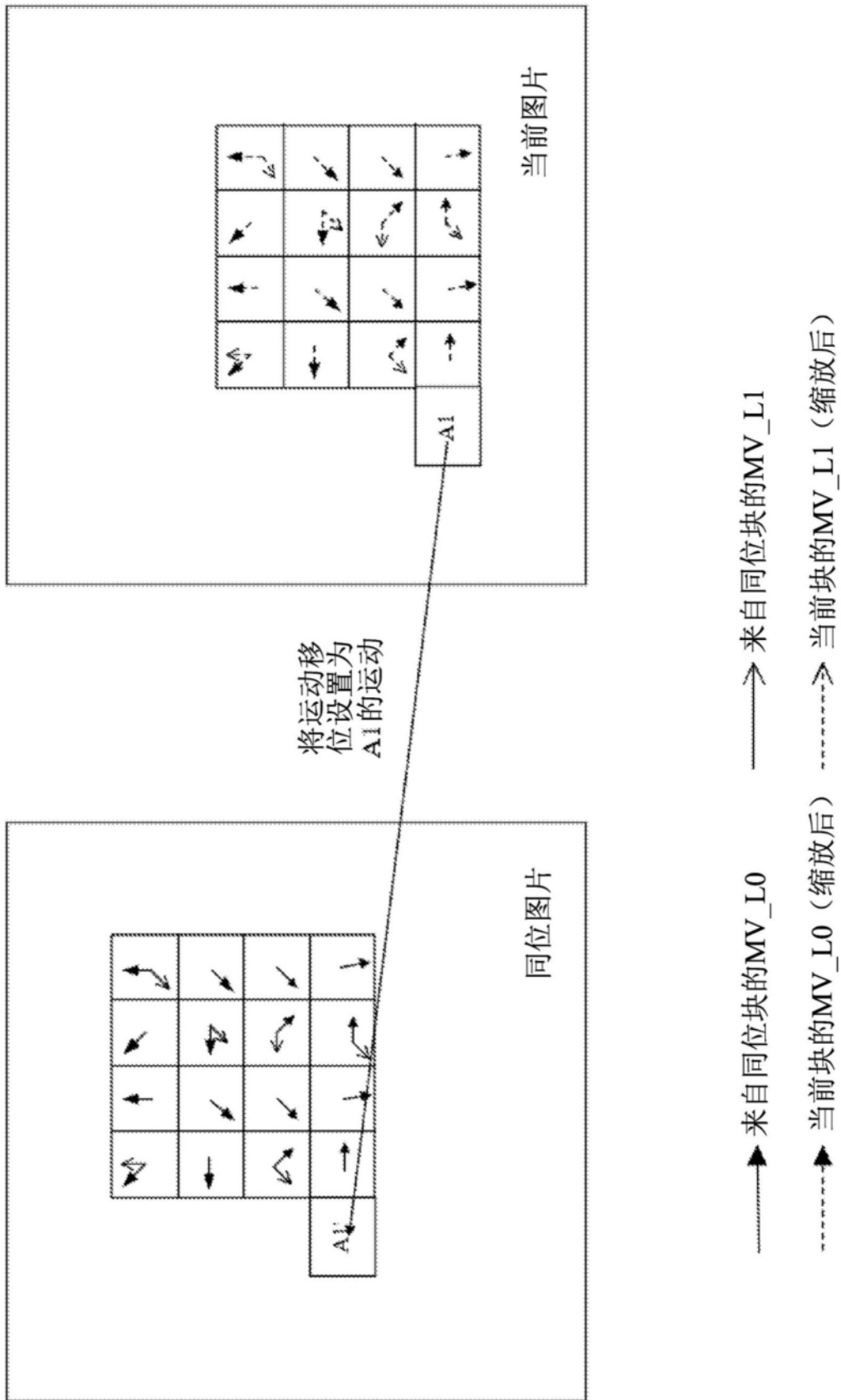


图15B

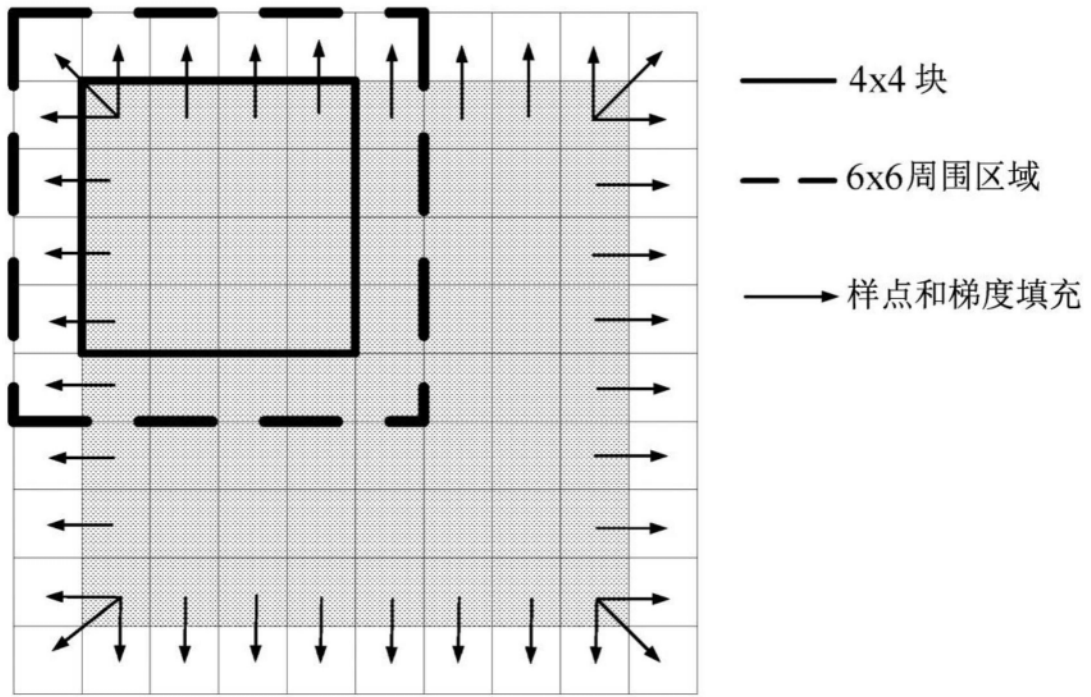


图16

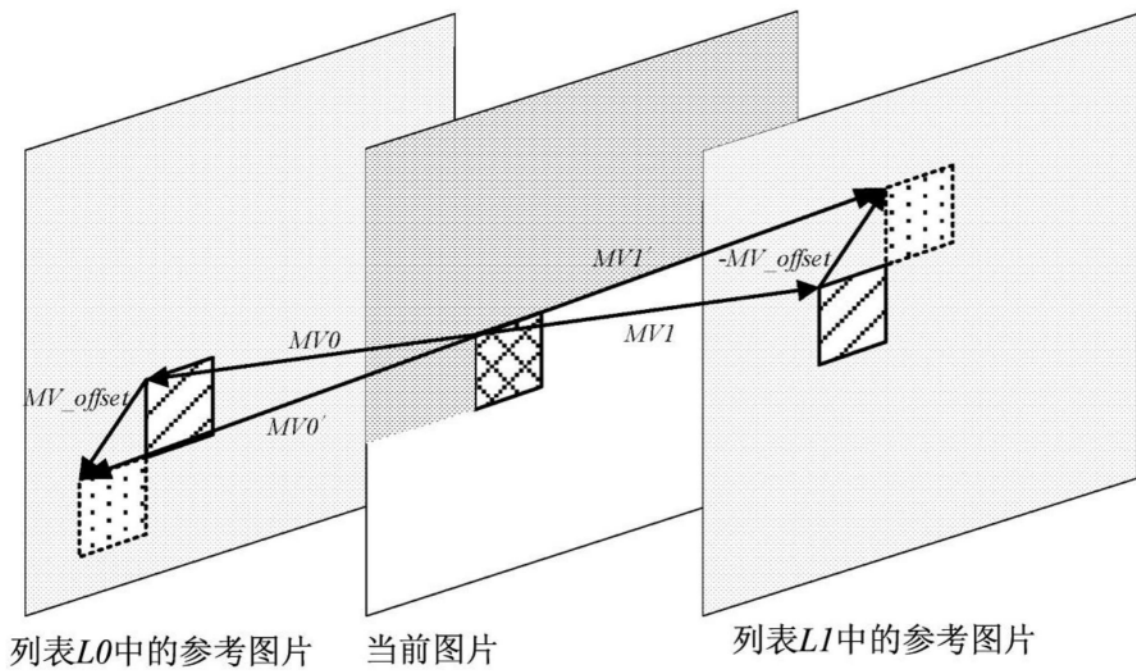


图17

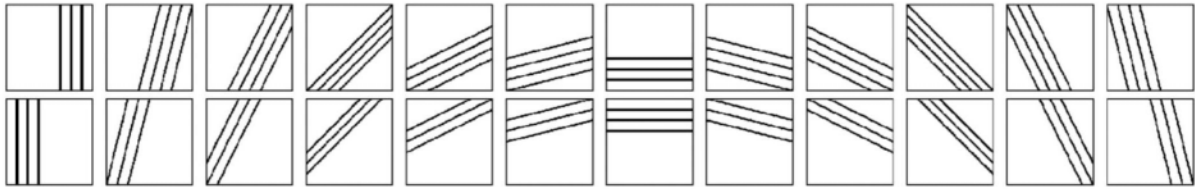


图18

	L0 MV	L1 MV
合并索引		
0	×	
1		×
2	×	
3		×
4	×	

图19

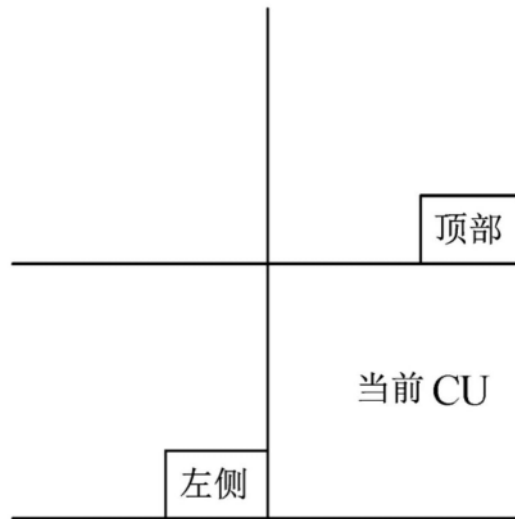


图20

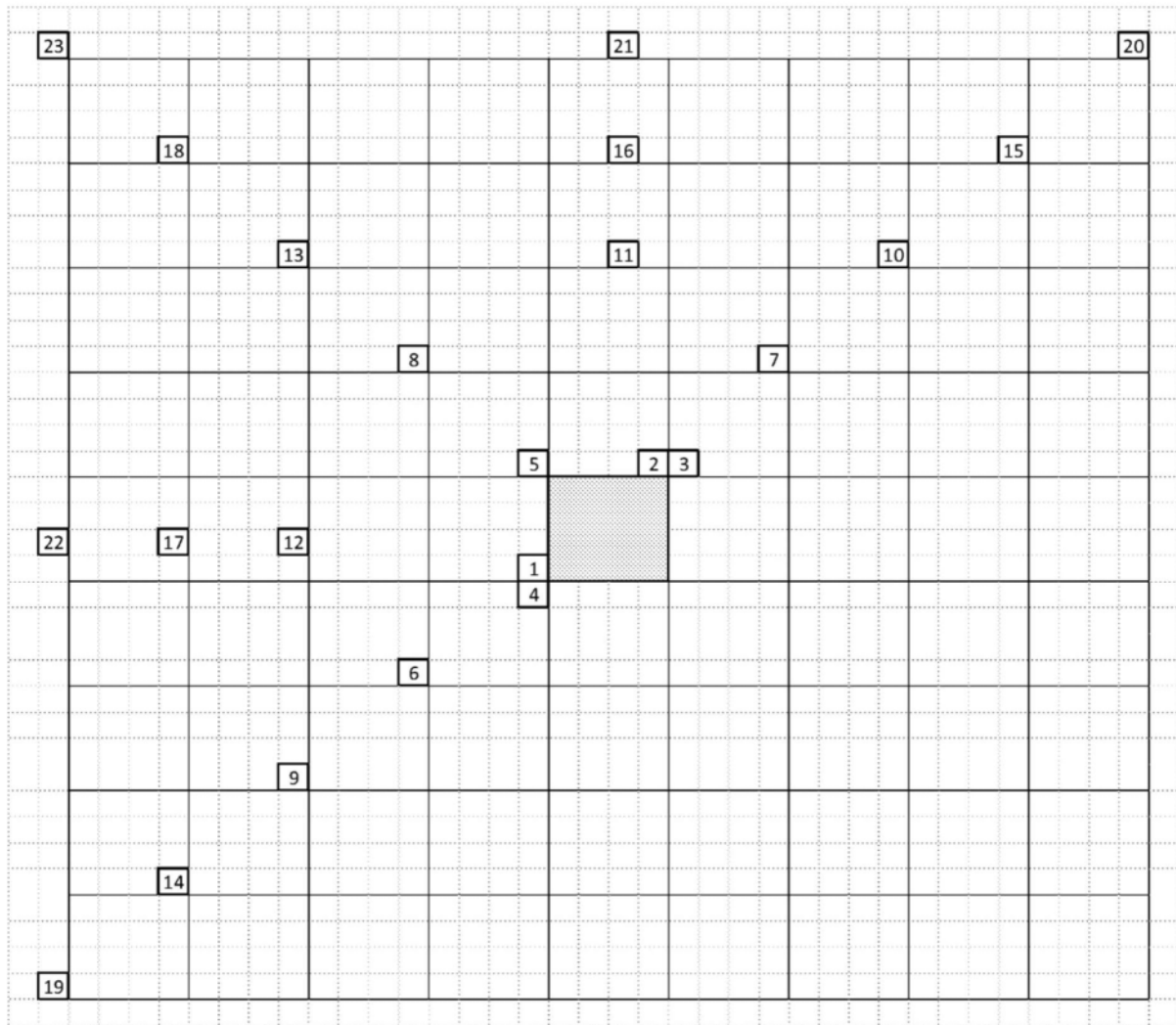


图21

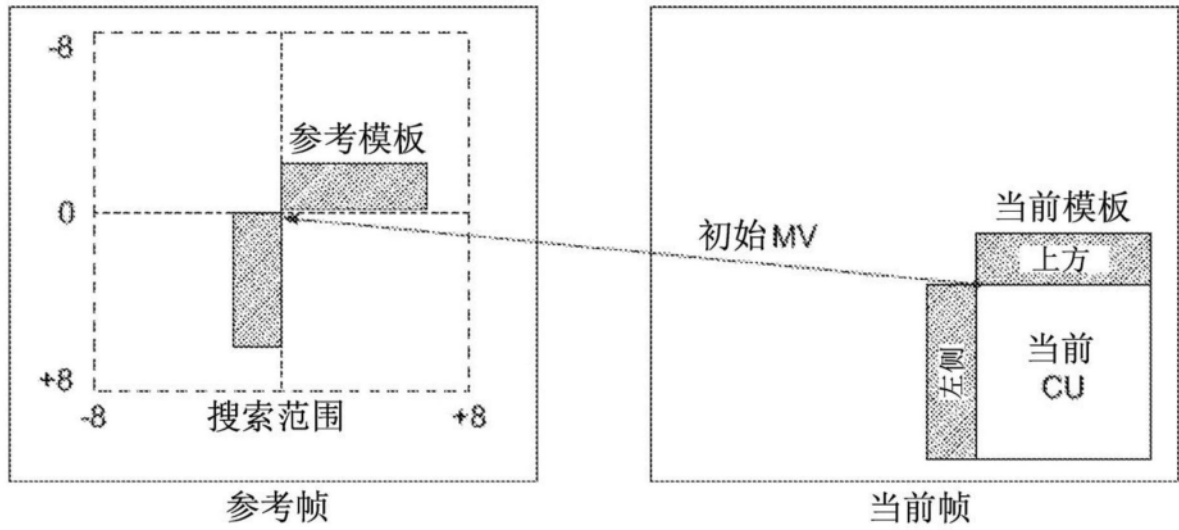


图22

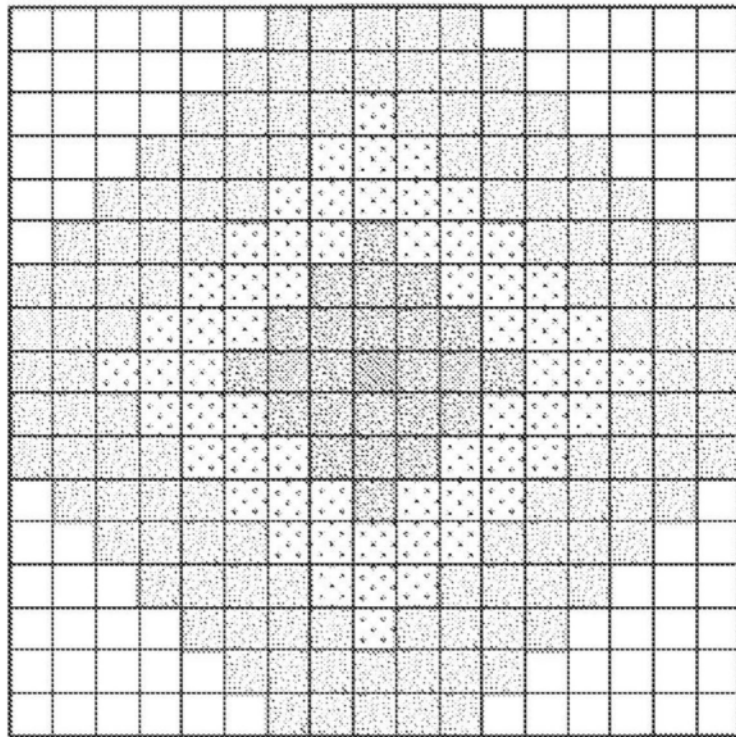


图23

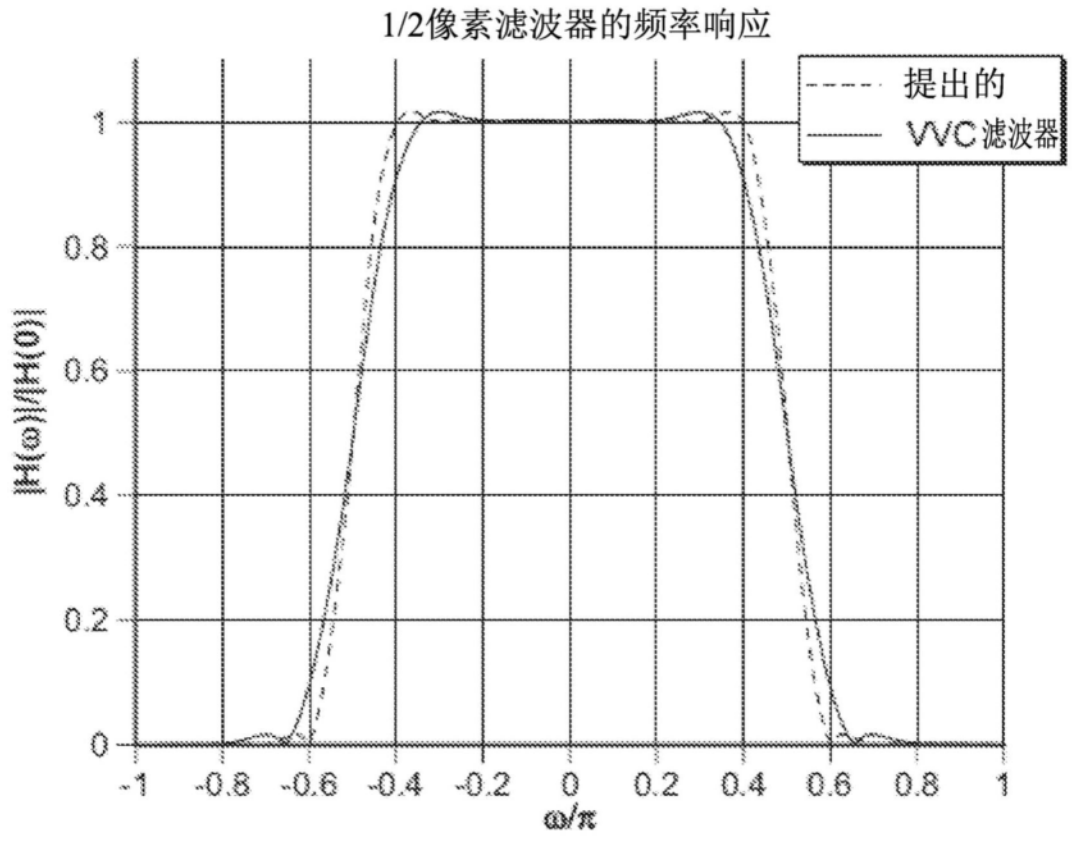


图24

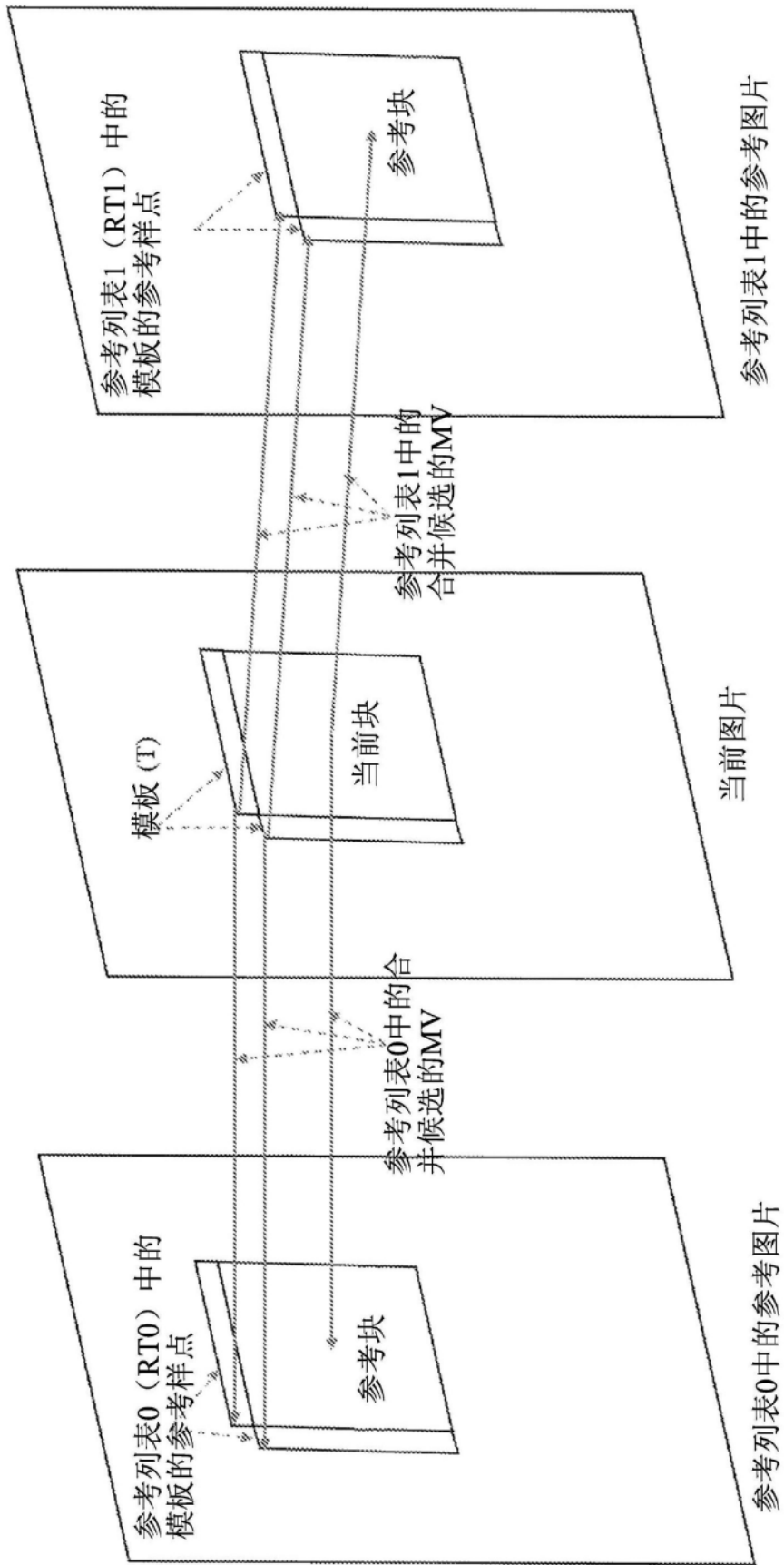


图25

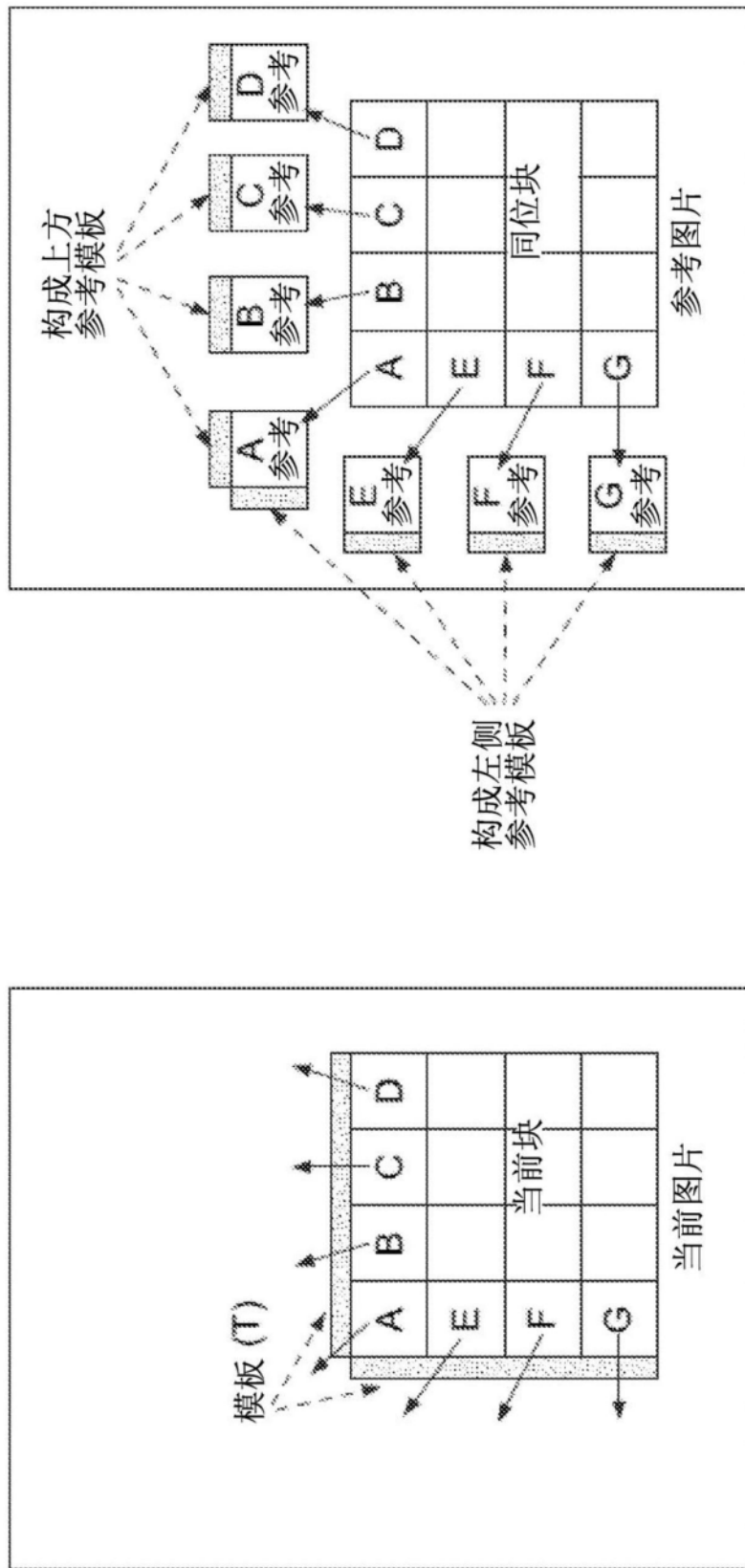


图26

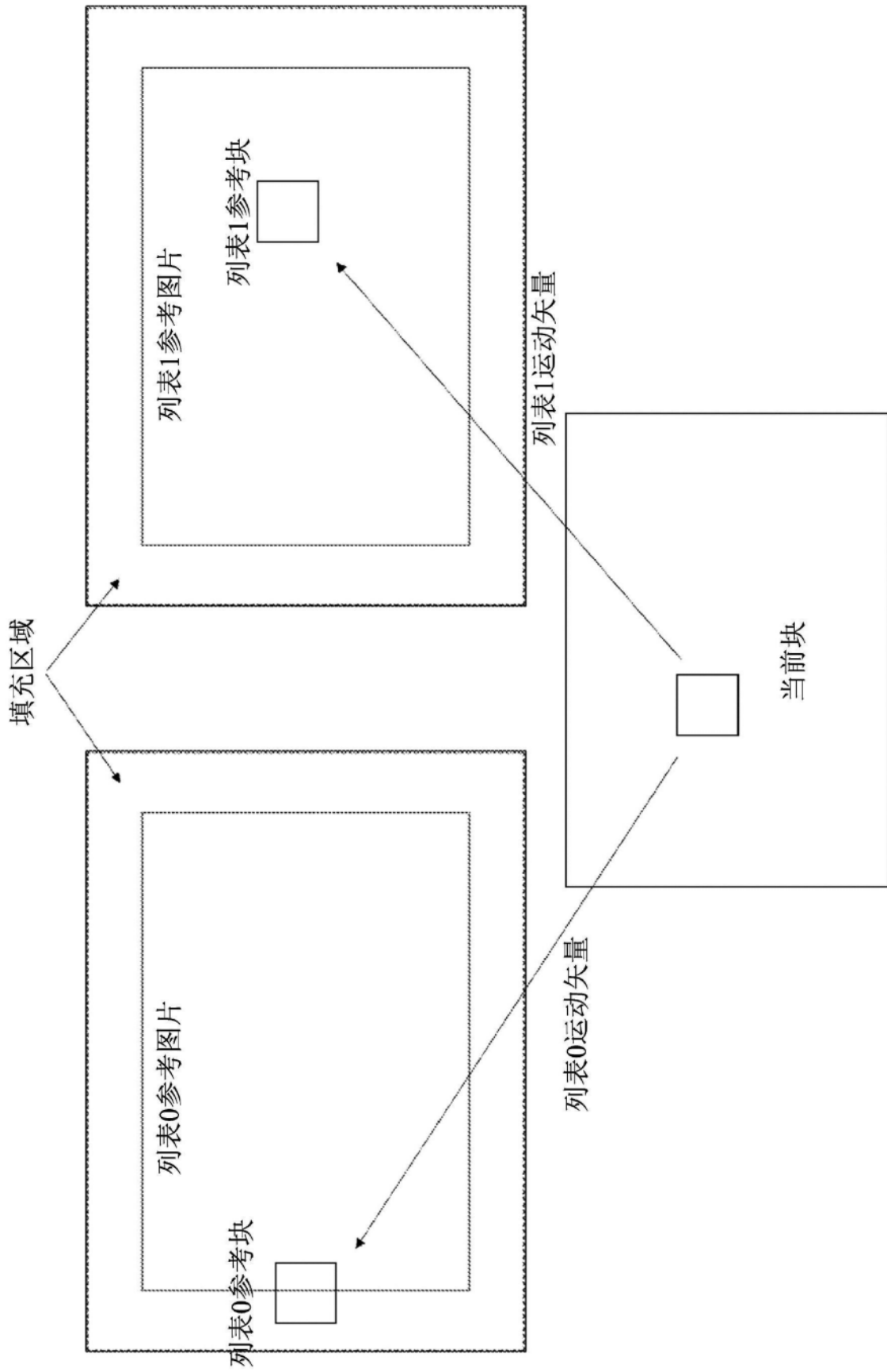


图27

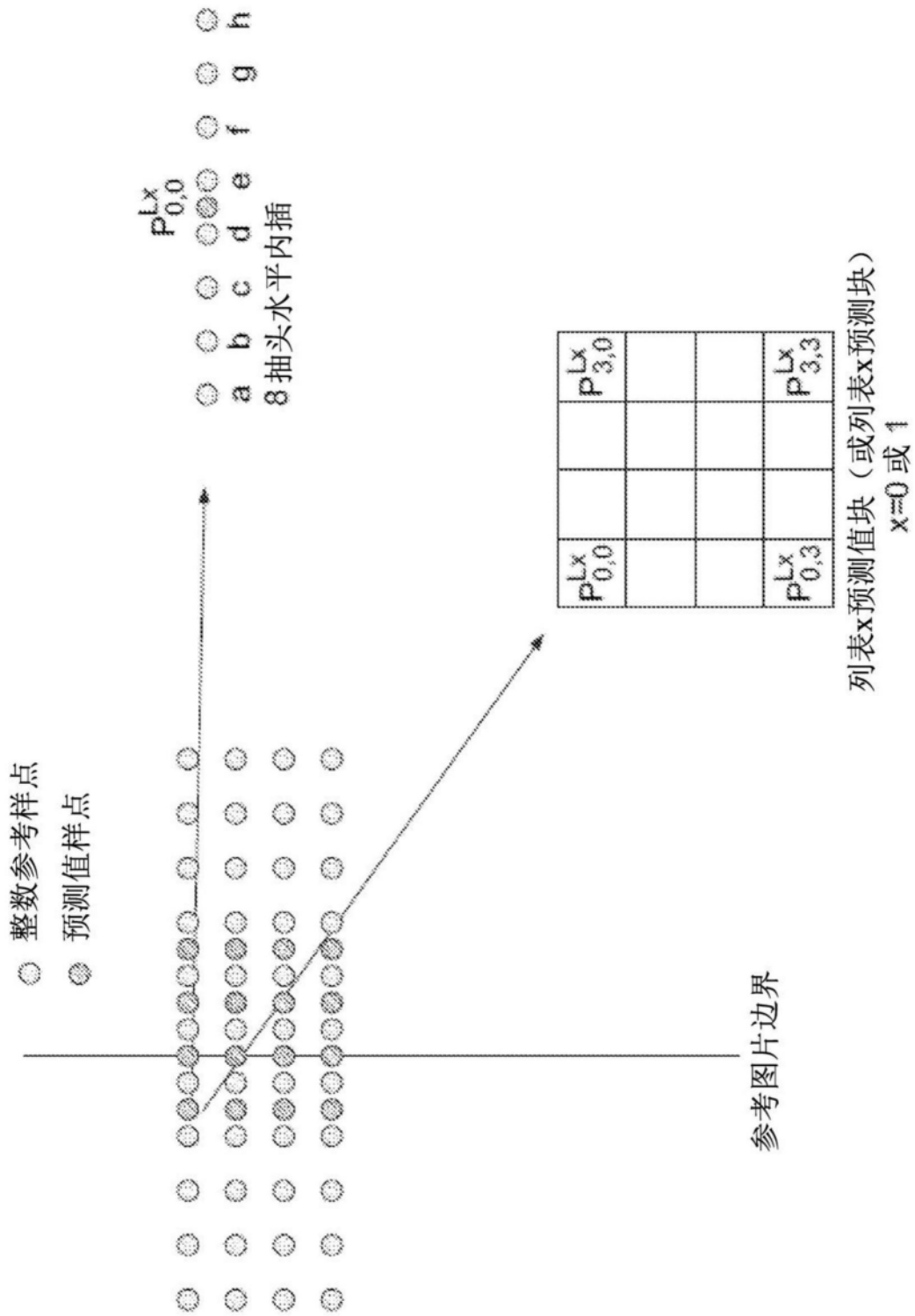


图28

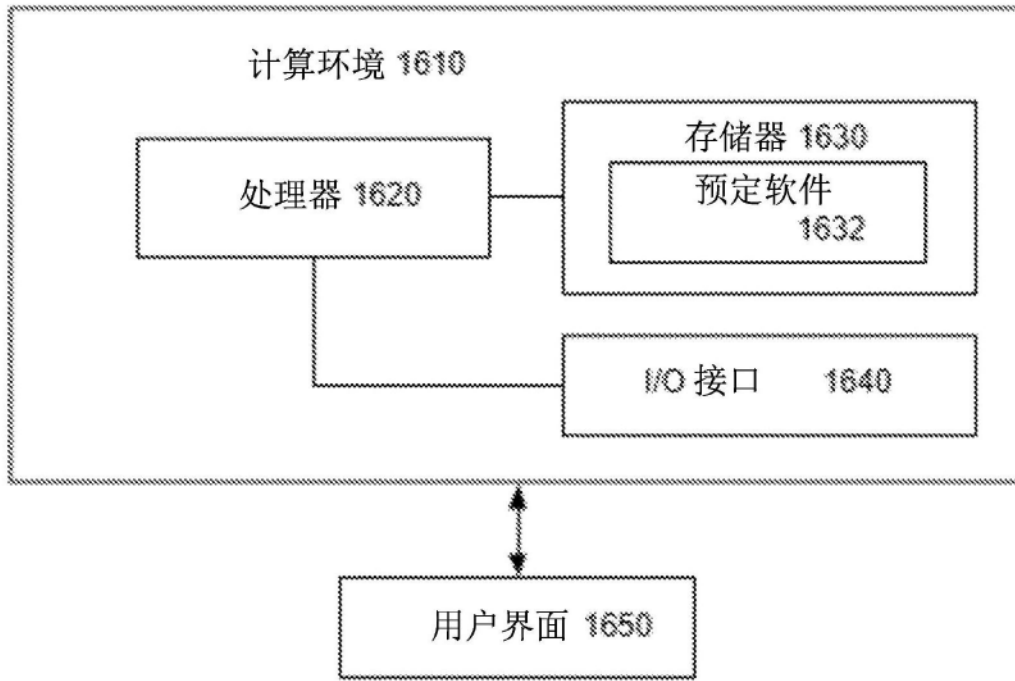


图29

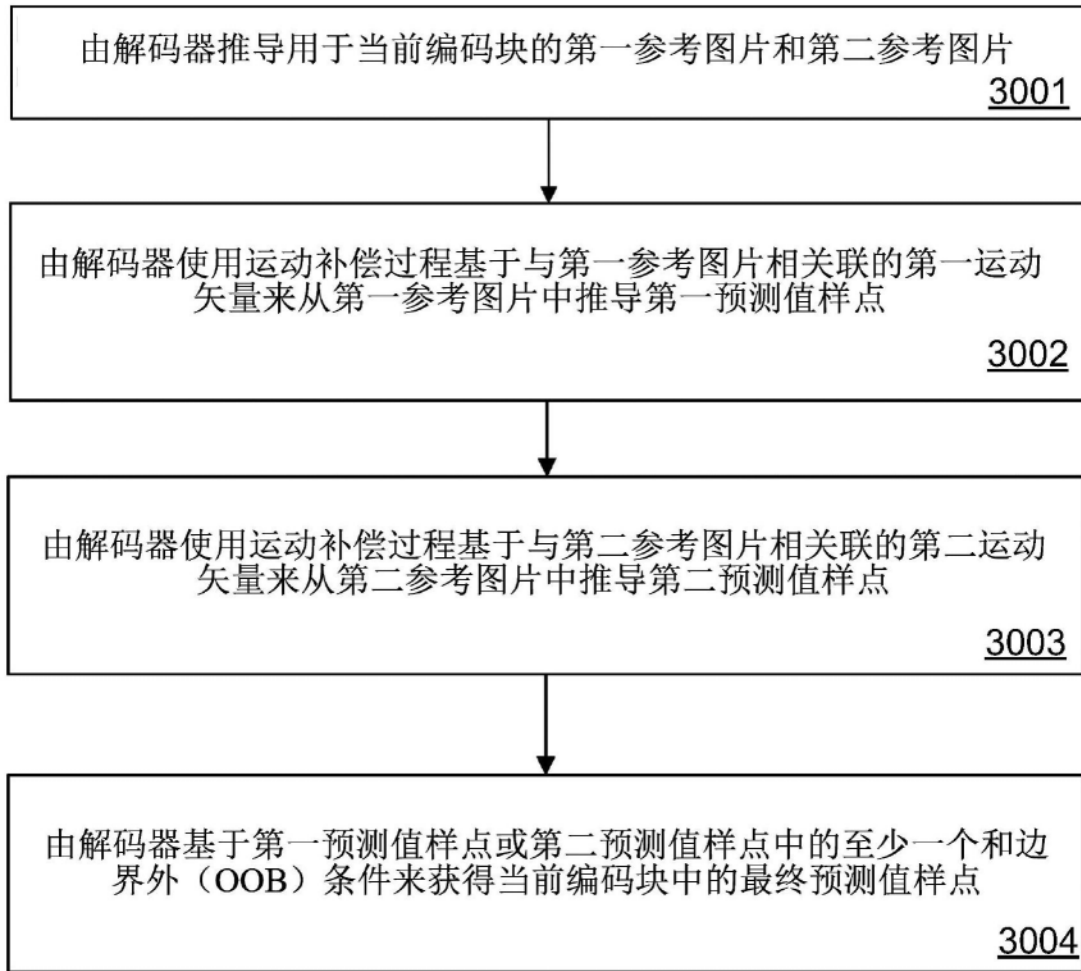


图30

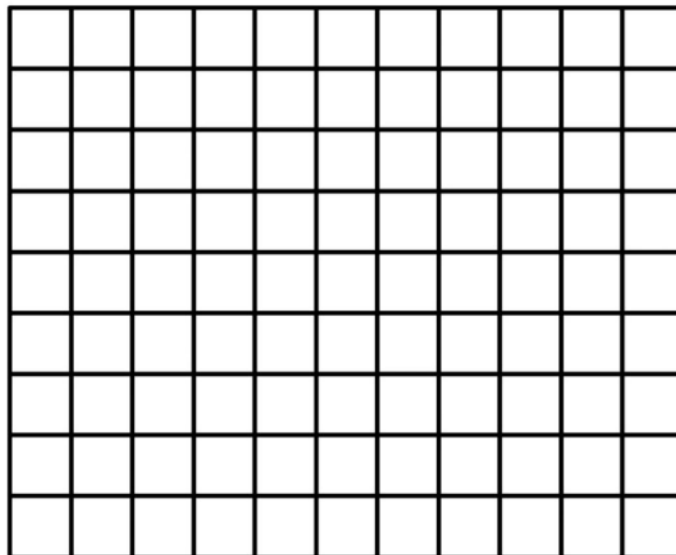


图31

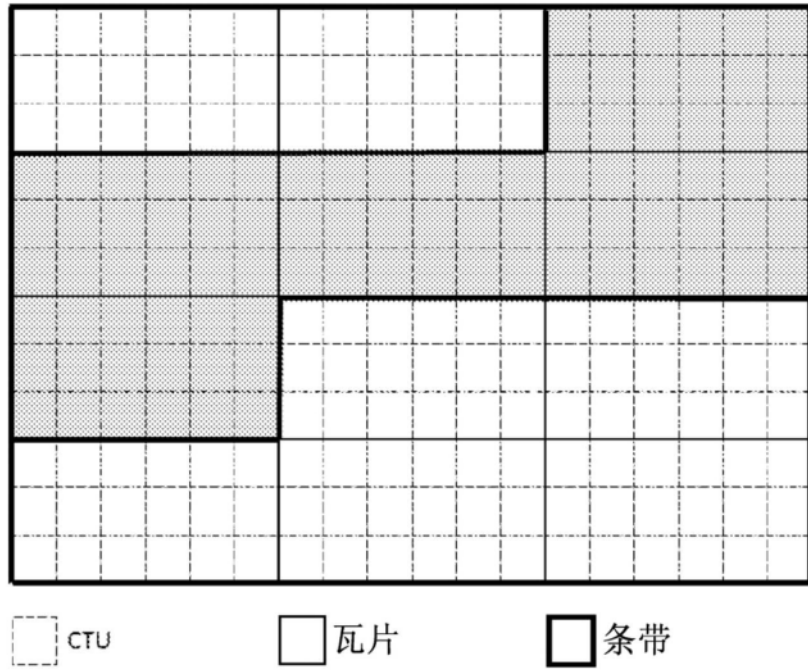


图32

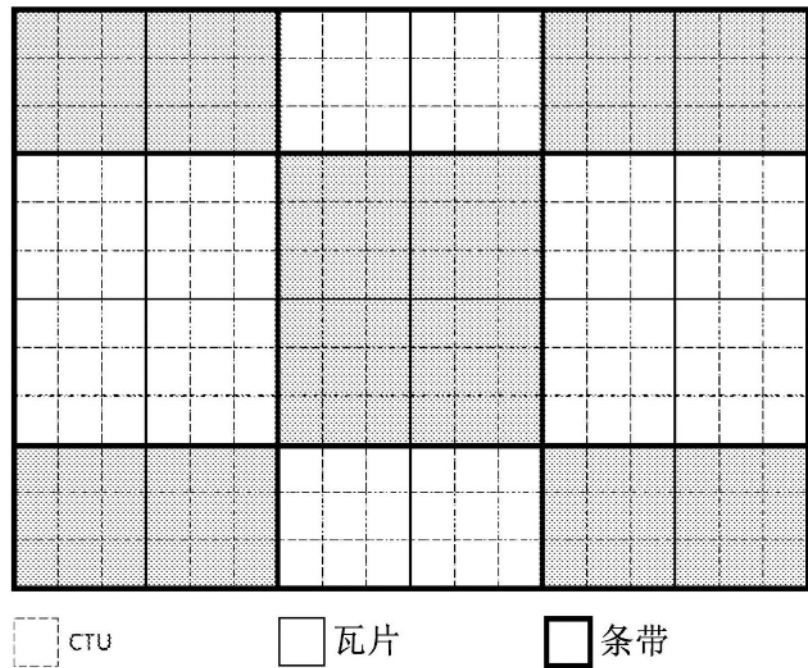


图33

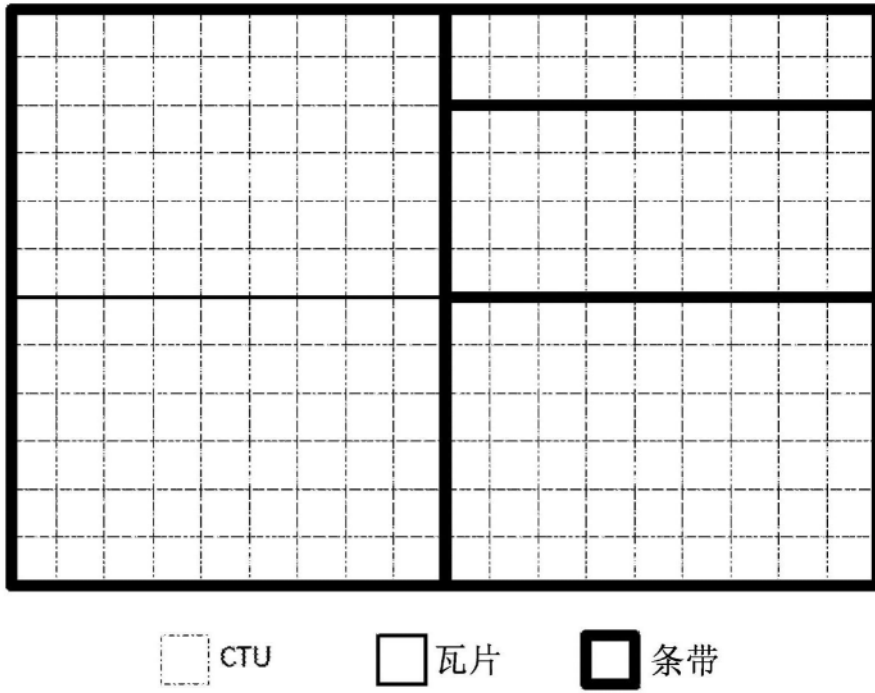


图34

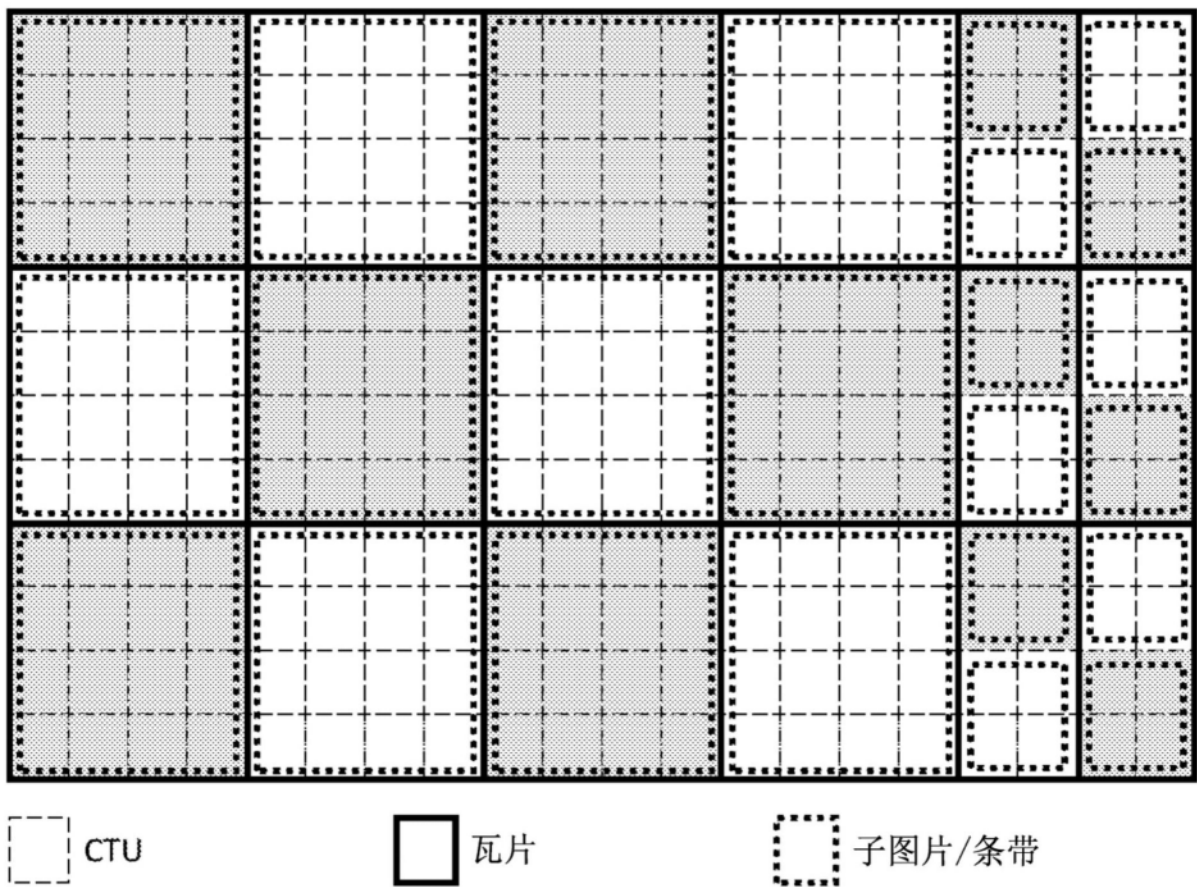


图35

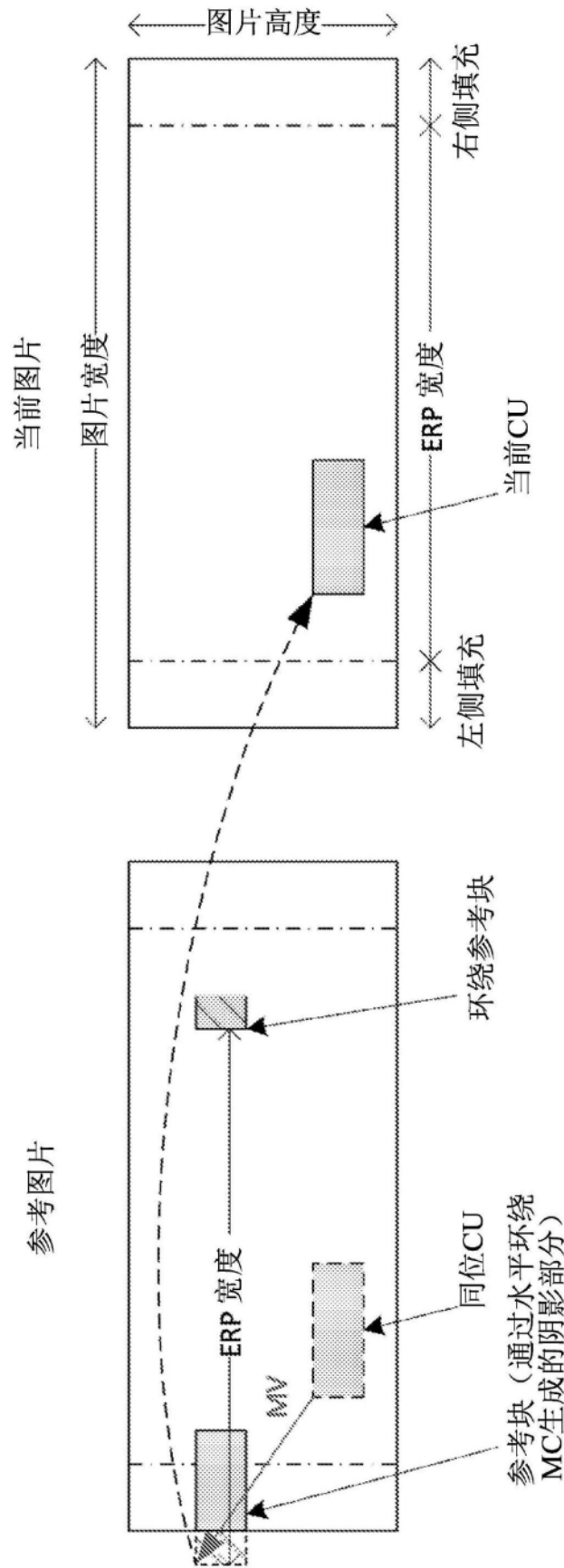


图36

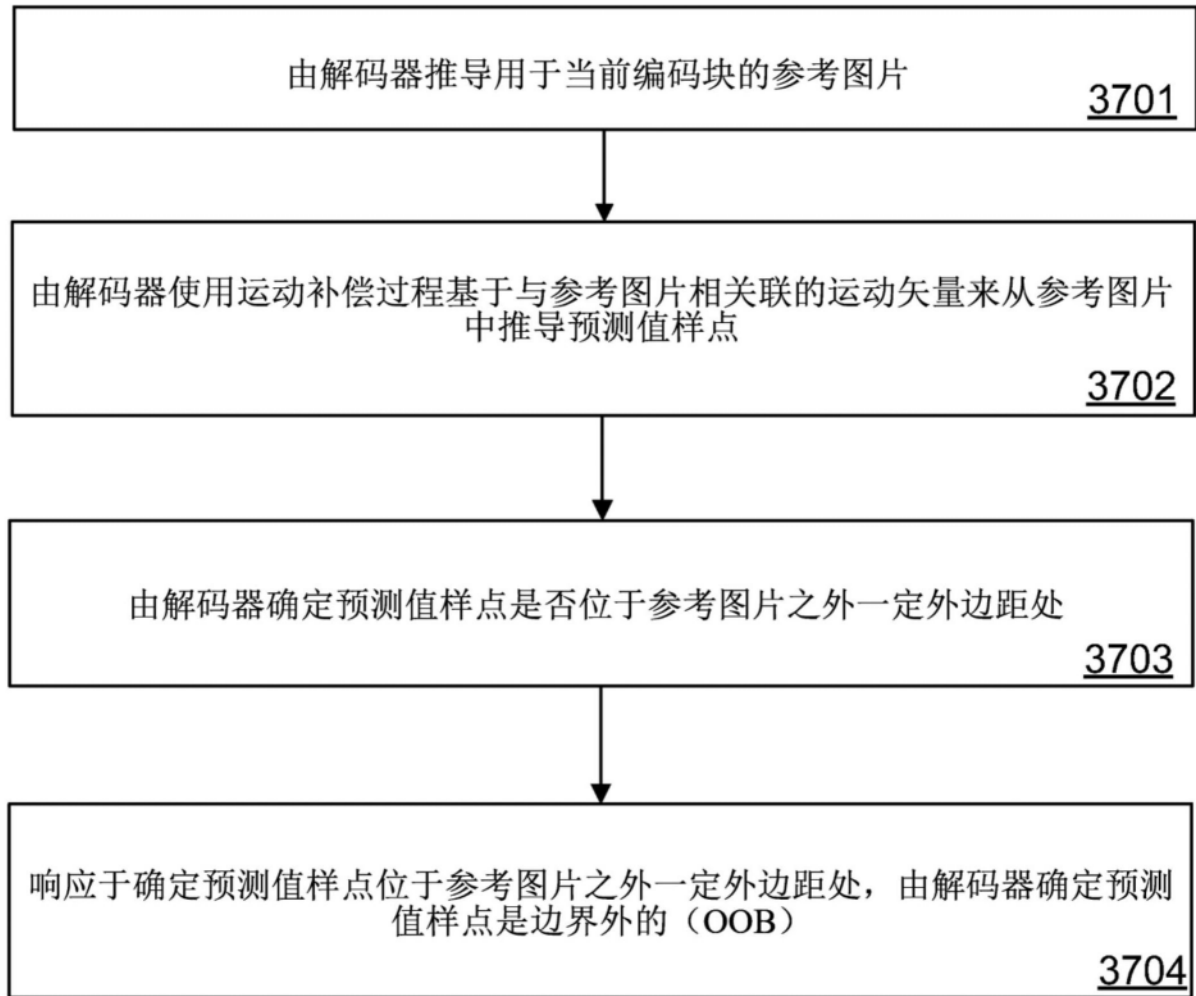


图37

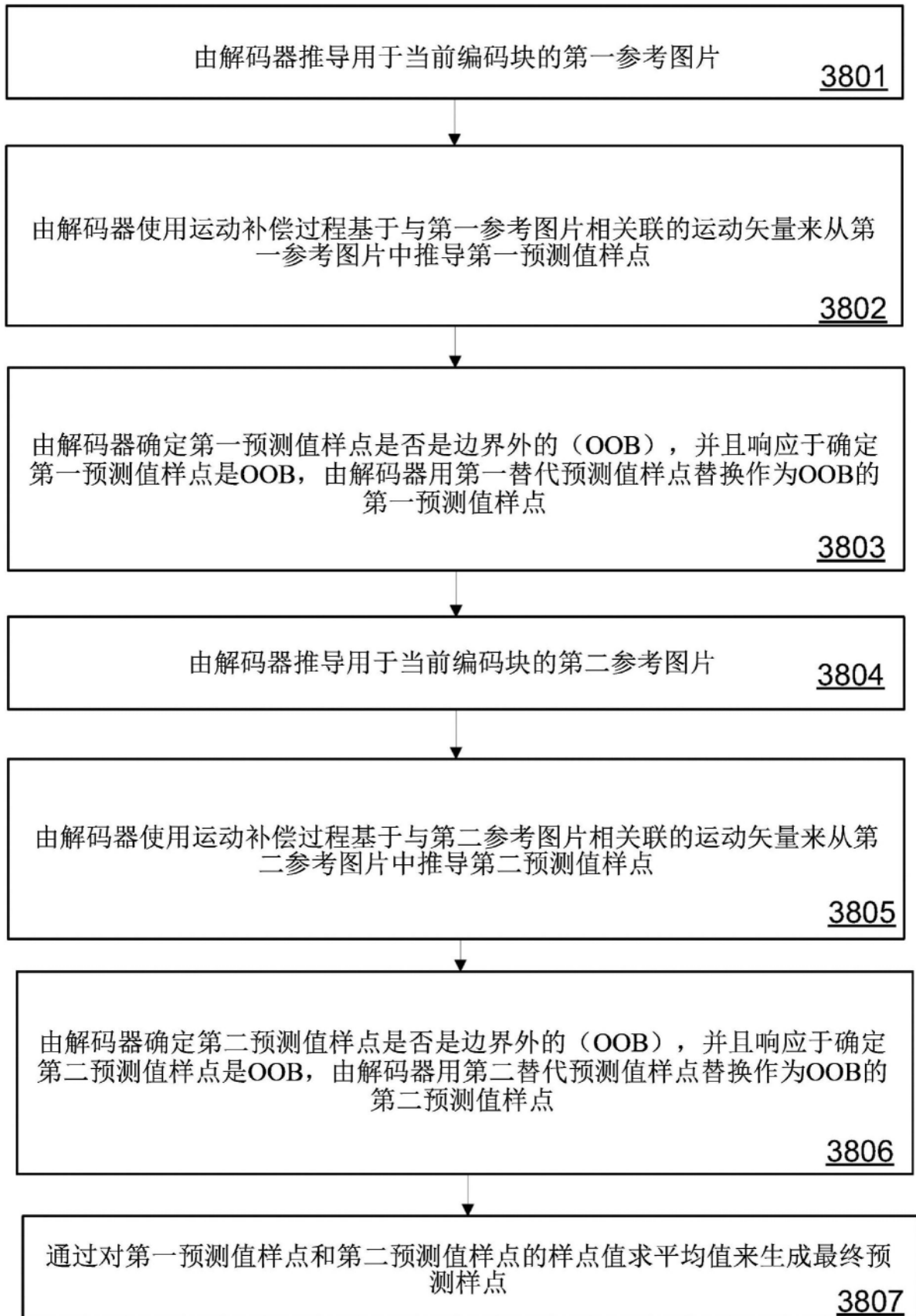


图38

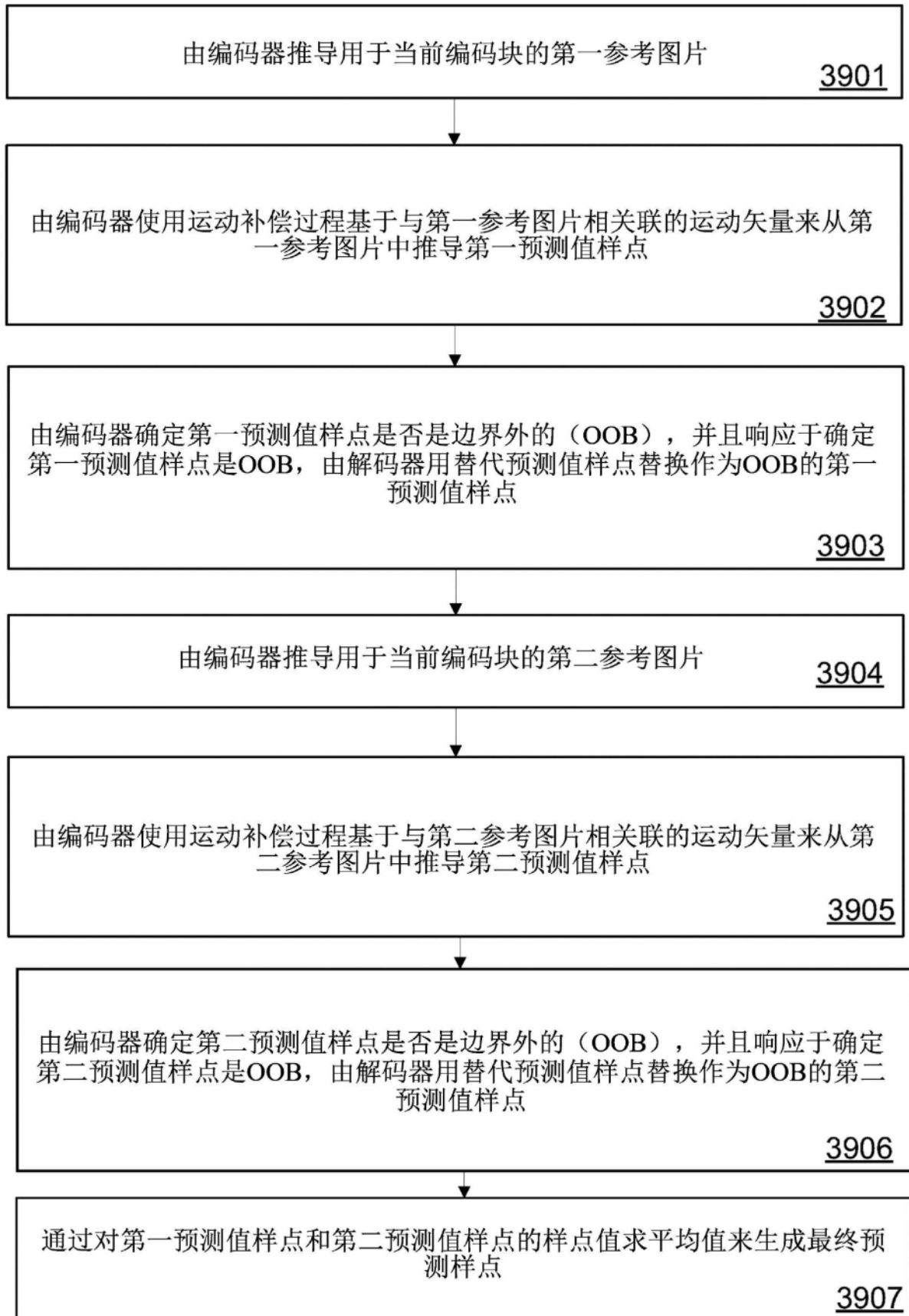


图39

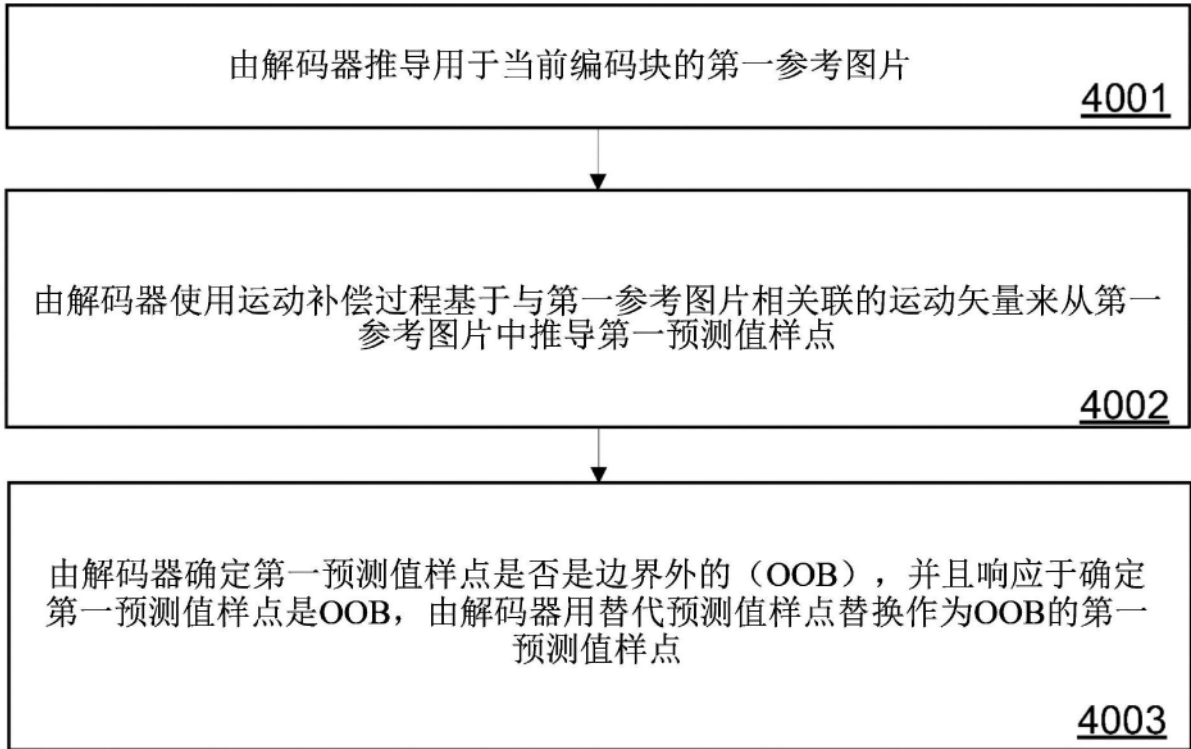


图40

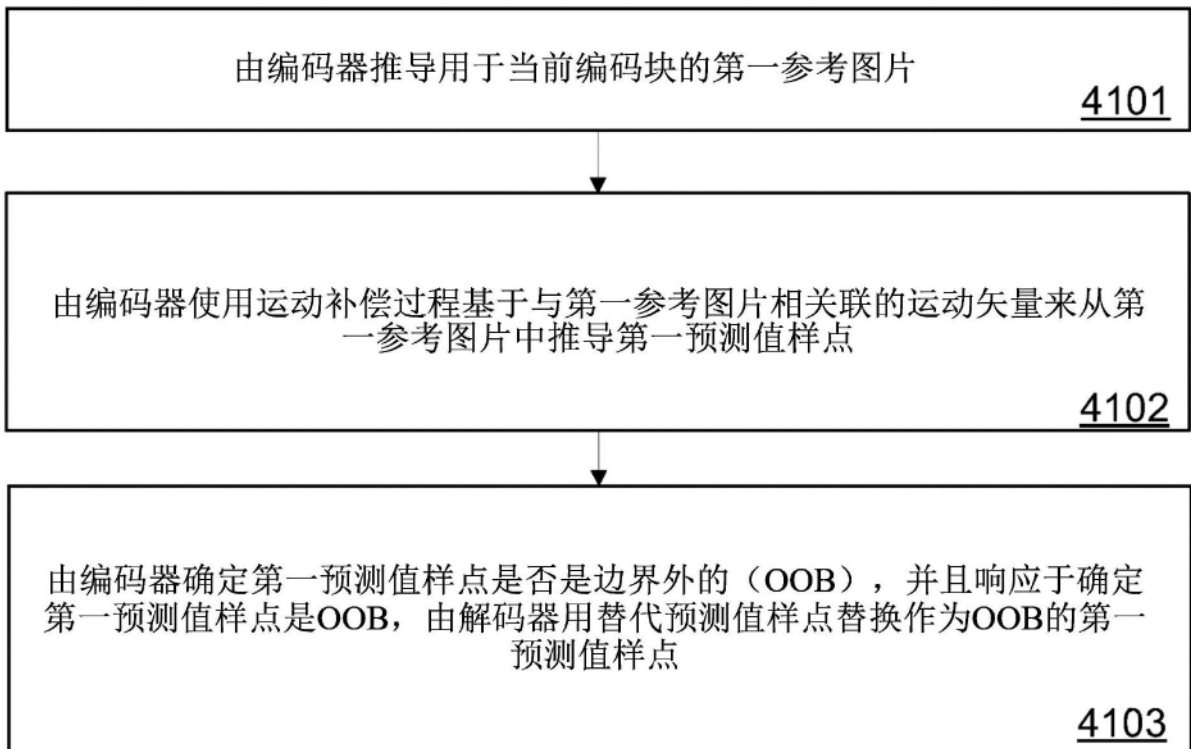


图41