



(12) 发明专利申请

(10) 申请公布号 CN 119967167 A

(43) 申请公布日 2025. 05. 09

(21) 申请号 202510248253.9

H04N 19/176 (2014.01)

(22) 申请日 2020.06.15

(30) 优先权数据

62/861,982 2019.06.14 US

(62) 分案原申请数据

202080047873.1 2020.06.15

(71) 申请人 LG 电子株式会社

地址 韩国

(72) 发明人 朴奈利 南廷学 张炯文

(74) 专利代理机构 中原信达知识产权代理有限
责任公司 11219

专利代理师 张伟峰 夏凯

(51) Int. Cl.

H04N 19/109 (2014.01)

H04N 19/513 (2014.01)

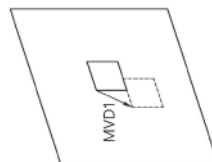
权利要求书3页 说明书60页 附图17页

(54) 发明名称

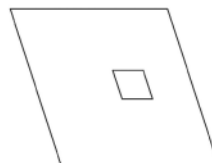
使用运动矢量差进行图像编译的方法和装置

(57) 摘要

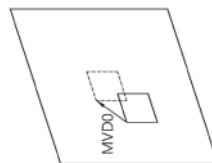
本发明涉及使用运动矢量差进行图像编译的方法和装置。根据本文档的实施例,可针对图像/视频编译执行预测过程,并且根据帧间预测,所述预测过程可以包括对称运动矢量差(SMVD)和合并模式运动矢量差(MMVD)。可以基于当前图片的参考图片执行帧间预测,并且可针对帧间预测考虑参考图片的类型(例如,长期参考图片、短期参考图片等)。因此,可以提高预测过程中的性能和编码效率。



列表1参考图片



当前图片



列表0参考图片

1. 一种用于图像解码的解码设备,所述解码设备包括:

存储器;以及

至少一个处理器,所述至少一个处理器被连接到所述存储器,所述至少一个处理器被配置为:

通过比特流获得包括预测相关信息、关于运动矢量预测子MVP索引的信息,和关于运动矢量差MVD的信息的图像信息,其中,所述预测相关信息包括关于对称运动矢量差SMVD的信息;

基于所述预测相关信息来推导用于当前块的帧间预测模式;

推导用于所述当前块的参考图片列表,所述参考图片列表包括参考图片列表L0和参考图片列表L1;

基于所述关于SMVD的信息推导用于L0的参考图片索引和用于L1的参考图片索引;

基于所述当前块的邻近块来构造MVP候选列表,所述MVP候选列表包括用于L0的MVP候选列表和用于L1的MVP候选列表;

从所述MVP候选列表中由MVP索引指示的MVP候选中,推导用于所述当前块的MVP,包括用于L0的MVP和用于L1的MVP;

基于用于L0的MVD和所述用于L0的MVP来推导用于L0的运动矢量,并且基于用于L1的MVD和所述用于L1的MVP来推导用于L1的运动矢量;以及

基于包括所述用于L0的参考图片索引和所述用于L0的运动矢量的用于L0的运动信息,以及包括所述用于L1的参考图片索引和所述用于L1的运动矢量的用于L1的运动信息,来生成用于所述当前块的预测样本,

其中,所述用于L0的MVD从所述关于MVD的信息被推导出,并且所述用于L1的MVD从所述用于L0的MVD被推导出,

其中,基于所述关于SMVD的信息的值为1,当以下所有条件均为真时,第一参考图片列表中的第一参考图片的参考图片索引被推导为用于所述第一参考图片列表的所述参考图片索引:

所述第一参考图片是短期参考图片,

包括所述当前块的当前图片的图片顺序计数POC大于所述第一参考图片的POC,以及

所述第一参考图片的POC比所述第一参考图片列表中的另一参考图片的POC更接近于所述当前图片,并且

当以下所有条件均为真时,第二参考图片列表中的第二参考图片的参考图片索引被推导为用于所述第二参考图片列表的所述参考图片索引:

所述第二参考图片是短期参考图片,

所述当前图片的POC小于所述第二参考图片的POC,以及

所述第二参考图片的POC比所述第二参考图片列表中的另一参考图片的POC更接近于所述当前图片。

2. 一种用于图像编码的编码设备,所述编码设备包括:

存储器;以及

至少一个处理器,所述至少一个处理器被连接到所述存储器,所述至少一个处理器被配置为:

推导用于当前块的帧间预测模式；

推导用于所述当前块的参考图片列表,所述参考图片列表包括参考图片列表L0和参考图片列表L1;

基于对称运动矢量差SMVD推导用于L0的参考图片索引和用于L1的参考图片索引;

基于所述当前块的邻近块来构造运动矢量预测子MVP候选列表,所述MVP候选列表包括用于L0的MVP候选列表和用于L1的MVP候选列表;

基于所述MVP候选列表来推导用于所述当前块的MVP,包括用于L0的MVP和用于L1的MVP;

生成与所述MVP候选列表中包括的用于所述当前块的MVP相关的关于MVP索引的信息;

推导包括用于L0的运动矢量和所述用于L0的参考图像索引的用于L0的运动信息,以及推导包括用于L1的运动矢量和所述用于L1的参考图像索引的用于L1的运动信息;

基于所述用于L0的MVP和所述用于L0的运动矢量,推导用于L0的运动矢量差MVD;

生成与所述帧间预测模式相关的预测相关信息,并且基于所述用于L0的MVD生成关于MVD的信息;

基于所述用于L0的运动信息和所述用于L1的运动信息来生成用于所述当前块的预测样本;

基于所述预测样本来生成残差信息;以及

对包括所述关于MVP索引的信息、所述预测相关信息、所述关于MVD的信息和所述残差信息的图像信息进行编码,

其中,所述预测相关信息包括关于SMVD的信息,

其中,基于所述关于SMVD的信息的值为1,当以下所有条件均为真时,第一参考图片列表中的第一参考图片的参考图片索引被推导为用于所述第一参考图片列表的所述参考图片索引:

所述第一参考图片是短期参考图片,

包括所述当前块的当前图片的图片顺序计数POC大于所述第一参考图片的POC,以及

所述第一参考图片的POC比所述第一参考图片列表中的另一参考图片的POC更接近于所述当前图片,并且

当以下所有条件均为真时,第二参考图片列表中的第二参考图片的参考图片索引被推导为用于所述第二参考图片列表的所述参考图片索引:

所述第二参考图片是短期参考图片,

所述当前图片的POC小于所述第二参考图片的POC,以及

所述第二参考图片的POC比所述第二参考图片列表中的另一参考图片的POC更接近于所述当前图片。

3. 一种用于发送用于图像的数据的设备,所述设备包括:

至少一个处理器,所述至少一个处理器被配置为获取用于图像的比特流,其中,所述比特流基于以下生成:推导用于当前块的帧间预测模式;推导用于所述当前块的参考图片列表,所述参考图片列表包括参考图片列表L0和参考图片列表L1;基于对称运动矢量差SMVD推导用于L0的参考图片索引和用于L1的参考图片索引;基于所述当前块的邻近块来构造运动矢量预测子MVP候选列表,所述MVP候选列表包括用于L0的MVP候选列表和用于L1的MVP候

选列表;基于所述MVP候选列表来推导用于所述当前块的MVP,包括用于L0的MVP和用于L1的MVP;生成与所述MVP候选列表中包括的用于所述当前块的MVP相关的关于MVP索引的信息;推导包括用于L0的运动矢量和所述用于L0的参考图像索引的用于L0的运动信息,以及推导包括用于L1的运动矢量和所述用于L1的参考图像索引的用于L1的运动信息;基于所述用于L0的MVP和所述用于L0的运动矢量,推导用于L0的运动矢量差MVD;生成与所述帧间预测模式相关的预测相关信息,并且基于所述用于L0的MVD生成关于MVD的信息;基于所述用于L0的运动信息和所述用于L1的运动信息来生成用于所述当前块的预测样本;基于所述预测样本来生成残差信息;以及对包括所述关于MVP索引的信息、所述预测相关信息、所述关于MVD的信息和所述残差信息的图像信息进行编码;以及

发送器,所述发送器被配置为发送包括所述比特流的数据,

其中,所述预测相关信息包括关于SMVD的信息,

其中,基于所述关于SMVD的信息的值为1,当以下所有条件均为真时,第一参考图片列表中的第一参考图片的参考图片索引被推导为用于所述第一参考图片列表的所述参考图片索引:

所述第一参考图片是短期参考图片,

包括所述当前块的当前图片的图片顺序计数POC大于所述第一参考图片的POC,以及

所述第一参考图片的POC比所述第一参考图片列表中的另一参考图片的POC更接近于所述当前图片,并且

当以下所有条件均为真时,第二参考图片列表中的第二参考图片的参考图片索引被推导为用于所述第二参考图片列表的所述参考图片索引:

所述第二参考图片是短期参考图片,

所述当前图片的POC小于所述第二参考图片的POC,以及

所述第二参考图片的POC比所述第二参考图片列表中的另一参考图片的POC更接近于所述当前图片。

使用运动矢量差进行图像编译的方法和装置

[0001] 本申请是2021年12月29日提交的国际申请日为2020年6月15日的申请号为202080047873.1 (202080047873.1)的,发明名称为“使用运动矢量差进行图像编译的方法和装置”的专利申请的分案申请。

技术领域

[0002] 本文档涉及一种使用运动矢量差进行图像编译的方法和装置。

背景技术

[0003] 最近,在各种领域中对诸如4K或8K超高清(UHD)图像/视频的高分辨率、高质量图像/视频的需求不断增加。随着图像/视频分辨率或质量变得更高,与传统图像/视频数据相比发送相对更多的信息或比特。因此,如果图像/视频数据经由诸如现有有线/无线宽带线路的介质发送或被存储在传统存储介质中,则传输和存储的成本容易增加。

[0004] 此外,对虚拟现实(VR)和人工现实(AR)内容以及诸如全息图的沉浸式媒体的兴趣和需求日益增长;并且表现出与现实图像/视频不同的图像/视频特性的图像/视频(例如,游戏图像/视频)的广播也日益增长。

[0005] 因此,需要高度高效的图像/视频压缩技术来有效地压缩并发送、存储或播放如上所述显示出各种特性的高分辨率、高质量图像/视频。

[0006] 具体地,图像/视频编译中的帧间预测可以使用运动矢量差。关于过程,讨论了基于参考图片类型(例如,短期或长期参考图片)推导运动矢量差。

发明内容

[0007] 技术方案

[0008] 根据本文档的实施例,提供了一种用于提高图像/视频编译效率的方法和装置。

[0009] 根据本文档的实施例,提供了一种用于在图像/视频编译系统中有效地执行帧间预测的方法和装置。

[0010] 根据本文档的实施例,提供了一种用于用信号发送关于用于帧间预测的运动矢量差的信息的方法和装置。

[0011] 根据本文档的实施例,当对当前块应用双预测时,提供了用于用信号发送关于L0运动矢量差和L1运动矢量差的信息的方法和装置。

[0012] 根据本文的实施例,提供了一种用于用信号发送SMVD标志的方法和装置。

[0013] 根据本文档的实施例,可以使用用于推导对称运动矢量差的特定参考图片类型。

[0014] 根据本文档的实施例,可以执行用于使用短期参考图片(标记为短期参考)来推导SMVD参考索引的过程。

[0015] 根据本文档的实施例,提供了由解码设备执行的视频/图像解码方法。

[0016] 根据本文档的实施例,提供了用于执行视频/图像解码的解码设备。

[0017] 根据本文档的实施例,提供了由编码设备执行的视频/图像编码方法。

- [0018] 根据本文档的实施例,提供了用于执行视频/图像编码的编码设备。
- [0019] 根据本文档的一个实施例,提供了一种计算机可读数字存储介质,其中存储根据本文档的至少一个实施例中公开的视频/图像编码方法生成的编码视频/图像信息。
- [0020] 根据本文档的实施例,提供了一种计算机可读数字存储介质,其中存储使得通过解码设备执行在本文档的至少一个实施例中公开的视频/图像解码方法的编码信息或编码视频/图像信息。
- [0021] 有益效果
- [0022] 根据本公开,可以提高整体图像/视频压缩效率。
- [0023] 根据本公开,可以有效地执行关于运动矢量差的信息的信令。
- [0024] 根据本公开,当对当前块应用双预测时,可以有效地推导L1运动矢量差。
- [0025] 根据本公开,可以基于参考图片的类型来用信号发送用于推导L1运动矢量差的信息,因此可以降低编译系统的复杂性。
- [0026] 根据本公开的一个实施例,可以使用用于推导SMVD的参考图片索引的特定参考图片类型来执行有效帧间预测。
- [0027] 通过本公开的特定实施例获得的效果不限于上述那些效果。例如,可以获得相关领域的普通技术人员能够理解或从本公开中推导的各种其他技术效果。因此,本公开的特定效果不限于本文档中明确公开的那些,并且可以包括可以从本公开的技术特征中理解或推导的各种其他效果。

附图说明

- [0028] 图1图示本文档的实施例可以应用于的视频/图像编译系统的示例。
- [0029] 图2是示意性地图示本文档的实施例可以应用于的视频/图像编码设备的配置的图。
- [0030] 图3是图示本文档的实施例可以应用于的视频/图像解码设备的配置的示意图。
- [0031] 图4示出基于帧间预测的视频/图像编码方法的示例。
- [0032] 图5示出基于帧间预测的视频/图像解码方法的示例。
- [0033] 图6示例性地示出帧间预测过程。
- [0034] 图7是用于描述对称运动矢量差(SMVD)的图。
- [0035] 图8是用于描述在帧间预测中推导运动矢量的方法的图。
- [0036] 图9至图13示出根据本文档的实施例的MMVD的MVD推导方法。
- [0037] 图14和图15图示根据本公开的实施例的视频/图像编码方法和相关组件的一个示例。
- [0038] 图16和图17图示根据本公开的实施例的视频/图像解码方法和相关组件的一个示例。
- [0039] 图18图示本文档的实施例可以应用于的内容流系统的示例。

具体实施方式

- [0040] 本文档可以以各种方式修改,并且其特定实施例将在附图中描述并且示出。然而,实施例并不旨限制本文档。以下描述中使用的术语仅用于描述特定实施例,而不是旨在限

制本文档。单数的表达式包括复数的表达式,只要它不被清楚地不同地理解。诸如“包括”或“具有”这样的术语应该被理解为指示存在以下描述中使用的特性、数字、步骤、操作、元件、组件或其组合,而没有排除存在或添加一个或多个不同的特性、数字、步骤、操作、元件、组件或其组合的可能性。

[0041] 同时,为了便于关于不同特征功能的描述,独立地示出本文档中描述的附图中的每个配置,并且不意味着每个配置被实现为单独的硬件或单独的软件。例如,可以组合每个组件中的两个或更多个组件以形成一个组件,或者可以将一个组件划分为多个组件。其中每个组件被集成和/或分离的实施例也包括在本文档的公开的范围內。

[0042] 在下文中,将参考附图详细地描述本实施例的示例。此外,在附图中,相同的附图标记用于指示相同的元件,并且可省略了相同元件的重复描述。

[0043] 图1图示可以应用本文档的实施例的视频/图像编译系统的示例。

[0044] 参考图1,视频/图像编译系统可以包括第一装置(源装置)和第二装置(接收装置)。源装置可以经由数字存储介质或网络以文件或流的形式将编码后的视频/图像信息或数据传送到接收装置。

[0045] 源装置可以包括视频源、编码设备和发送器。接收装置可以包括接收器、解码设备和渲染器。编码设备可以被称为视频/图像编码设备,并且解码设备可以被称为视频/图像解码设备。发送器可以被包括在编码设备中。接收器可以被包括在解码设备中。渲染器可以包括显示器,并且显示器可以被配置为单独的装置或外部组件。

[0046] 视频源可以通过捕获、合成或生成视频/图像的处理来获得视频/图像。视频源可以包括视频/图像捕获装置和/或视频/图像生成装置。视频/图像捕获装置可以包括例如一个或更多个相机、包括先前捕获的视频/图像的视频/图像档案等。视频/图像生成装置可以包括例如计算机、平板计算机和智能电话,并且可以(电子地)生成视频/图像。例如,可以通过计算机等生成虚拟视频/图像。在这种情况下,视频/图像捕获过程可以被生成相关数据的过程取代。

[0047] 编码设备可以对输入视频/图像进行编码。编码设备可以执行诸如预测、变换和量化这样的一系列过程用于压缩和编译效率。编码后的数据(编码后的视频/图像信息)可以以比特流的形式输出。

[0048] 发送器可以通过数字存储介质或网络以文件或流的形式将以比特流的形式输出的编码后的图像/视频信息或数据发送到接收装置的接收器。数字存储介质可以包括诸如USB、SD、CD、DVD、蓝光、HDD、SSD等这样的各种存储介质。发送器可以包括用于通过预定文件格式生成媒体文件的元件,并且可以包括用于通过广播/通信网络进行传输的元件。接收器可以接收/提取比特流,并且将接收的比特流发送到解码设备。

[0049] 解码设备可以通过执行与编码设备的操作对应的诸如解量化、逆变换、预测等这样的一系列过程来解码视频/图像。

[0050] 渲染器可以渲染解码后的视频/图像。可以通过显示器显示渲染后的视频/图像。

[0051] 本文档涉及视频/图像编译。例如,本文档中公开的方法/实施例可以应用于通用视频编码(VVC)、基本视频编码(EVC)标准、AOMedia Video 1(AV1)标准、第二代音频视频编码标准(AVS2)或下一代视频/图像编码标准(例如,H.267或H.268等)中公开的方法。

[0052] 本文档提出视频/图像编译的各种实施例,并且除非相反地提及,否则这些实施例

可以被彼此组合地执行。

[0053] 在本文档中,视频可指随着时间的一系列图像。图片通常是指表示特定时间帧处的一个图像的单元,并且切片(slice)/拼块(tile)是指在编译方面构成图片的一部分的单元。切片/拼块可以包括一个或多个编译树单元(CTU)。一个图片可以由一个或多个切片/拼块组成。一个图片可以由一个或多个拼块组组成。一个拼块组可以包括一个或多个拼块。图块(brick)可以表示图片中的拼块内的CTU行的矩形区域。拼块可以被分区成多个图块,每个图块可以利用拼块内的一个或多个CTU行来构建。未被分区成多个图块的拼块也可以被称为图块。图块扫描可以表示将图片分区的CTU的特定顺序排序,其中CTU可以在图块内以CTU光栅扫描排序,并且拼块内的图块可以以拼块的图块的光栅扫描连续地排序,并且图片中的拼块可以以图片的拼块的光栅扫描连续地排序。拼块是图片内特定拼块列和特定拼块行内的CTU的矩形区域。拼块列是CTU的矩形区域,其具有等于图片的高度的高度和由图片参数集中的语法元素指定的宽度。拼块行是CTU的矩形区域,其具有由图片参数集中的语法元素指定的高度和等于图片的宽度的宽度。拼块扫描是将图片分区的CTU的特定顺序排序,其中CTU在拼块中以CTU光栅扫描连续地排序而图片中的拼块以图片的拼块的光栅扫描连续地排序。切片包括图片的整数个图块,其可以排他性地包括在单个NAL单元中。切片可以由多个完整拼块组成,或者仅由一个拼块的连续完整图块序列组成。在本文档中,可以使用拼块组和切片来代替彼此。例如,在本文档中,拼块组/拼块组报头可以称为切片/切片报头。

[0054] 同时,一个图片可以划分成两个或更多个子图片。子图片可以是图片内一个或多个切片的矩形区域。

[0055] 像素或像元(pixel)可以意指构成一个图片(或图像)的最小单元。另外,“样本”可以用作与像素对应的术语。样本通常可以表示像素或像素值,并且可以仅表示亮度分量的像素/像素值或仅表示色度分量的像素/像素值。

[0056] 单元可以表示图像处理的基本单位。单元可以包括图片的特定区域和与该区域有关的信息中的至少一个。一个单元可以包括一个亮度块和两个色度(例如,cb、cr)块。在一些情况下,单元可以与诸如块或区域的术语互换使用。在一般情况下, $M \times N$ 块可以包括M列和N行的样本(或样本阵列)或变换系数的集合(或阵列)。可选地,样本可以意指空间域中的像素值,并且当这种像素值被变换到频域时,它可以意指频域中的变换系数。

[0057] 在本文档中,“A或B”可以意指“仅A”、“仅B”或“A和B这二者”。换句话说,在本文档中,“A或B”可以被解释为“A和/或B”。例如,在本文档中,“A、B或C(A、B或C)”意指“仅A”、“仅B”、“仅C”或“A、B和C的任何组合”。

[0058] 本文档中使用的斜线“/”或逗号(,)可以意指“和/或”。例如,“A/B”可以意指“A和/或B”。因此,“A/B”可以意指“仅A”、“仅B”或“A和B这二者”。例如,“A、B、C”可以意指“A、B或C”。

[0059] 在本文档中,“A和B中的至少一个”可以意指“仅A”、“仅B”或“A和B这二者”。此外,在本文档中,表达“A或B中的至少一个”或“A和/或B中的至少一个”可以被解释为“A和B中的至少一个”。

[0060] 此外,在本文档中,“A、B和C中的至少一个”可以意指“仅A”、“仅B”、“仅C”或“A、B和C的任何组合”。此外,“A、B或C中的至少一个”或“A、B和/或C中的至少一个”可以意指“A、B和

C中的至少一个”。

[0061] 此外,本文档中使用的括号可以意指“例如”。具体地,当指示“预测(帧内预测)”时,可以将“帧内预测”作为“预测”的示例提出。换句话说,本文档中的“预测”不限于“帧内预测”,并且可以将“帧内预测”作为“预测”的示例提出。此外,即使在指示“预测(即,帧内预测)”,也可以将“帧内预测”作为“预测”的示例提出。

[0062] 在本文档的一个附图中单独描述的技术特征可以单独地或同时地实现。

[0063] 图2示出可以应用本文档的实施例的视频/图像编码设备的结构。在下文中,所谓的视频编码设备可以包括图像编码设备。

[0064] 参考图2,编码设备200包括图像分区器210、预测器220、残差处理器230和熵编码器240、加法器250、滤波器260和存储器270。预测器220可以包括帧间预测器221和帧内预测器222。残差处理器230可以包括变换器232、量化器233、解量化器234和逆变换器235。残差处理器230还可以包括减法器231。加法器250可以被称为重构器或重构块生成器。根据实施例,图像分区器210、预测器220、残差处理器230、熵编码器240、加法器250和滤波器260可以由至少一个硬件组件(例如,编码器芯片组或处理器)配置。另外,存储器270可以包括解码图片缓冲器(DPB),或者可以由数字存储介质配置。硬件组件还可以包括存储器270作为内部/外部组件。

[0065] 图像分区器210可以将输入到编码设备200的输入图像(或者图片或帧)分区成一个或多个处理单元。例如,处理单元可以被称为编译单元(CU)。在这种情况下,编译单元可以根据四叉树二叉树三叉树(QTBT)结构从编译树单元(CTU)或最大编译单元(LCU)递归地分区。例如,一个编译单元可以基于四叉树结构、二叉树结构和/或三元结构被分区成深度更深的多个编译单元。在这种情况下,例如,可以首先应用四叉树结构,稍后可以应用二叉树结构和/或三元结构。另选地,可以首先应用二叉树结构。可以基于不再分区的最终编译单元来执行根据本文档的编译过程。在这种情况下,根据图像特性基于编译效率,最大编译单元可以用作最终编译单元,或者如果需要,编译单元可以被递归地分区成深度更深的编译单元并且具有最优大小的编译单元可以用作最终编译单元。这里,编译过程可以包括预测、变换和重构的过程(将稍后描述)。作为另一示例,处理单元还可以包括预测单元(PU)或变换单元(TU)。在这种情况下,预测单元和变换单元可以从上述最终编译单元分割或分区。预测单元可以是样本预测的单元,变换单元可以是用于推导变换系数的单元和/或用于从变换系数推导残差信号的单元。

[0066] 在一些情况下,单元可以与诸如块或区域的术语互换使用。在一般情况下, $M \times N$ 块可以表示由M列和N行组成的样本或变换系数的集合。样本通常可以表示像素或像素值,可以仅表示亮度分量的像素/像素值或者仅表示色度分量的像素/像素值。样本可以用作与像素或像元(pe1)的一个图片(或图像)对应的术语。

[0067] 在编码设备200中,从输入图像信号(原始块、原始样本阵列)减去从帧间预测器221或帧内预测器222输出的预测信号(预测块、预测样本阵列)以生成残差信号(残差块、残差样本阵列),并且所生成的残差信号被发送到变换器232。在这种情况下,如所示,在编码设备200中从输入图像信号(原始块、原始样本阵列)减去预测信号(预测块、预测样本阵列)的单元可以被称为减法器231。预测器可以对要处理的块(以下,称为当前块)执行预测并且生成包括当前块的预测样本的预测块。预测器可以以当前块或CU为单位确定应用帧内预测

还是帧间预测。如在各个预测模式的描述中稍后描述的,预测器可以生成与预测有关的各种信息(例如,预测模式信息)并将所生成的信息发送到熵编码器240。关于预测的信息可在熵编码器240中编码并以比特流的形式输出。

[0068] 帧内预测器222可以参考当前图片中的样本来预测当前块。根据预测模式,所参考的样本可以位于当前块附近或者可以隔开。在帧内预测中,预测模式可以包括多个非定向模式和多个定向模式。例如,非定向模式可以包括DC模式和平面模式。例如,根据预测方向的详细程度,定向模式可以包括33个定向预测模式或65个定向预测模式。然而,这仅是示例,可以根据设置使用更多或更少的定向预测模式。帧内预测器222可以使用应用于邻近块的预测模式来确定应用于当前块的预测模式。

[0069] 帧间预测器221可以基于参考图片上由运动矢量指定的参考块(参考样本阵列)来推导当前块的预测块。这里,为了减少在帧间预测模式下发送的运动信息量,可以基于邻近块与当前块之间的运动信息的相关性以块、子块或样本为单位预测运动信息。运动信息可以包括运动矢量和参考图片索引。运动信息还可以包括帧间预测方向(L0预测、L1预测、Bi预测等)信息。在帧间预测的情况下,邻近块可以包括存在于当前图片中的空间邻近块和存在于参考图片中的时间邻近块。包括参考块的参考图片和包括时间邻近块的参考图片可相同或不同。时间邻近块可以被称为并置参考块、并置CU(colCU)等,并且包括时间邻近块的参考图片可以被称为并置图片(colPic)。例如,帧间预测器221可以基于邻近块来配置运动信息候选列表并且生成指示哪一候选用于推导当前块的运动矢量和/或参考图片索引的信息。可以基于各种预测模式执行帧间预测。例如,在跳过模式和合并模式的情况下,帧间预测器221可以使用邻近块的运动信息作为当前块的运动信息。在跳过模式下,与合并模式不同,可以不发送残差信号。在运动矢量预测(MVP)模式的情况下,邻近块的运动矢量可以用作运动矢量预测子(predictor),并且可以通过用信号发送运动矢量差来指示当前块的运动矢量。

[0070] 预测器220可以基于下面描述的各种预测方法来生成预测信号。例如,预测器可不仅应用帧内预测或帧间预测来预测一个块,而且同时应用帧内预测和帧间预测二者。这可以被称为组合帧间和帧内预测(CIIP)。另外,预测器可以基于帧内块复制(IBC)预测模式或调色板模式来预测块。IBC预测模式或调色板模式可以用于游戏等的图像/视频编码,例如屏幕内容编译(SCC)。IBC基本上在当前图片中执行预测,但是可以与帧间预测相似地执行,使得在当前图片中推导参考块。即,IBC可以使用本文档中描述的至少一个帧间预测技术。调色板模式可以被视为帧内编译或帧内预测的示例。当应用调色板模式时,可以基于关于调色板表和调色板索引的信息用信号发送图片内的样本值。

[0071] 由预测器(包括帧间预测器221和/或帧内预测器222)生成的预测信号可以用于生成重构信号或生成残差信号。变换器232可以通过对残差信号应用变换技术来生成变换系数。例如,变换技术可以包括离散余弦变换(DCT)、离散正弦变换(DST)、karhunen-loève变换(KLT)、基于图形的变换(GBT)或条件非线性变换(CNT)中的至少一个。这里,当像素之间的关系信息由图形表示时,GBT意指从图形获得的变换。CNT是指基于使用所有先前重构的像素生成的预测信号生成的变换。另外,变换处理可以应用于具有相同大小的正方形像素块或者可以应用于正方形以外的具有可变大小的块。

[0072] 量化器233可以将变换系数量化并将它们发送到熵编码器240,并且熵编码器240

可以对量化的信号(关于量化变换系数的信息)进行编码并输出比特流。关于量化变换系数的信息可以被称为残差信息。量化器233可以基于系数扫描顺序将块类型量化变换系数重排为一维矢量形式,并且基于一维矢量形式的量化变换系数来生成关于量化变换系数的信息。可以生成关于变换系数的信息。熵编码器240可以执行例如指数Golomb、上下文自适应可变长度编码(CAVLC)、上下文自适应二进制算术编码(CABAC)等的各种编码方法。熵编码器240可以对量化变换系数以外的视频/图像重构所需的信息(例如,语法元素的值等)一起或单独地进行编码。编码的信息(例如,编码的视频/图像信息)可以按比特流的形式以NAL(网络抽象层)为单位发送或存储。视频/图像信息还可以包括关于各种参数集的信息,例如自适应参数集(APS)、图片参数集(PPS)、序列参数集(SPS)或视频参数集(VPS)。另外,视频/图像信息还可以包括一般约束信息。在本文档中,从编码设备发送/用信号发送给解码设备的信息和/或语法元素可以被包括在视频/图片信息中。视频/图像信息可以通过上述编码过程编码并被包括在比特流中。比特流可以经由网络发送或者可以被存储在数字存储介质中。网络可以包括广播网络和/或通信网络,并且数字存储介质可以包括诸如USB、SD、CD、DVD、蓝光、HDD、SSD等的各种存储介质。发送从熵编码器240输出的信号的发送器(未示出)和/或存储该信号的存储单元(未示出)可以被包括作为编码设备200的内部/外部元件,并且另选地,发送器可以被包括在熵编码器240中。

[0073] 从量化器233输出的量化变换系数可以用于生成预测信号。例如,可以通过经由解量化器234和逆变换器235对量化变换系数应用解量化和逆变换来重构残差信号(残差块或残差样本)。加法器250将重构的残差信号与从帧间预测器221或帧内预测器222输出的预测信号相加以生成重构信号(重构图片、重构块、重构样本阵列)。如果要处理的块不存在残差(例如,应用跳过模式的情况),则预测块可以用作重构块。加法器250可以被称为重构器或重构块生成器。如下所述,所生成的重构信号可以用于当前图片中要处理的下一块的帧内预测并且可以通过滤波用于下一图片的帧间预测。

[0074] 此外,可在图片编码和/或重构期间应用亮度映射与色度缩放(LMCS)。

[0075] 滤波器260可以通过对重构信号应用滤波来改进主观/客观图像质量。例如,滤波器260可以通过对重构图片应用各种滤波方法来生成修改的重构图片并将修改的重构图片存储在存储器270(具体地,存储器270的DPB)中。例如,各种滤波方法可以包括去块滤波、样本自适应偏移、自适应环路滤波器、双边滤波器等。滤波器260可以生成与滤波有关的各种信息并且将所生成的信息发送到熵编码器240,如在各个滤波方法的描述中稍后描述的。与滤波有关的信息可以由熵编码器240编码并以比特流的形式输出。

[0076] 发送到存储器270的修改的重构图片可以用作帧间预测器221中的参考图片。当通过编码设备应用帧间预测时,可以避免编码设备200与解码设备之间的预测失配并且编码效率可改进。

[0077] 存储器270的DPB可以存储用作帧间预测器221中的参考图片的修改的重构图片。存储器270可以存储推导(或编码)当前图片中的运动信息的块的运动信息和/或图片中已经重构的块的运动信息。所存储的运动信息可以被发送到帧间预测器221并用作空间邻近块的运动信息或时间邻近块的运动信息。存储器270可以存储当前图片中的重构块的重构样本并且可以将重构样本传送至帧内预测器222。

[0078] 图3示出可以应用本说明书的实施例的视频/图像解码设备的配置。

[0079] 参考图3,解码设备300可以包括熵解码器310、残差处理器320、预测器330、加法器340、滤波器350、存储器360。预测器330可以包括帧间预测器332和帧内预测器331。残差处理器320可以包括解量化器321和逆变换器322。根据实施例,熵解码器310、残差处理器320、预测器330、加法器340和滤波器350可以由硬件组件(例如,解码器芯片组或处理器)配置。另外,存储器360可以包括解码图片缓冲器(DPB)或者可以由数字存储介质配置。硬件组件还可以包括存储器360作为内部/外部组件。

[0080] 当输入包括视频/图像信息的比特流时,解码设备300可以重构与在图2的编码设备中处理视频/图像信息的处理对应的图像。例如,解码设备300可以基于从比特流获得的块分区相关信息来推导单元/块。解码设备300可以使用编码设备中应用的处理器来执行解码。因此,例如,解码的处理器可以是编译单元,并且编译单元可以根据二叉树结构、二叉树结构和/或三叉树结构从编码树单元或最大编译单元分区。可从编译单元推导一个或更多个变换单元。通过解码设备300解码和输出的重构的图像信号可以通过再现设备再现。

[0081] 解码设备300可接收从图2的编码设备以比特流的形式输出的信号,并且所接收的信号可以通过熵解码器310解码。例如,熵解码器310可以解析比特流以推导图像重构(或图片重构)所需的信息(例如,视频/图像信息)。视频/图像信息还可以包括关于各种参数集的信息,例如自适应参数集(APS)、图片参数集(PPS)、序列参数集(SPS)或视频参数集(VPS)。另外,视频/图像信息还可以包括一般约束信息。解码设备还可以基于关于参数集的信息和/或一般约束信息将图片解码。本文档中稍后描述的用信号发送/接收的信息和/或语法元素可以通过解码过程解码并从比特流获得。例如,熵解码器310基于诸如指数Golomb编码、CAVLC或CABAC的编码方法对比特流中的信息进行解码,并且输出图像重构所需的语法元素和残差的变换系数的量化值。更具体地,CABAC熵解码方法可接收与比特流中的各个语法元素对应的bin(二进制位),使用解码目标语法元素信息、解码目标块的解码信息或在先前阶段中解码的符号/bin的信息来确定上下文模型,并且通过根据所确定的上下文模型预测bin出现的概率对bin执行算术解码,并且生成与各个语法元素的值对应的符号。在这种情况下,CABAC熵解码方法可以在确定上下文模型之后通过将解码的符号/bin的信息用于下一符号/bin的上下文模型来更新上下文模型。熵解码器310所解码的信息当中与预测有关的信息可以被提供给预测器(帧间预测器332和帧内预测器331),并且在熵解码器310中执行了熵解码的残差值(即,量化变换系数和相关参数信息)可以被输入到残差处理器320。残差处理器320可以推导残差信号(残差块、残差样本、残差样本阵列)。另外,熵解码器310所解码的信息当中关于滤波的信息可以被提供给滤波器350。此外,用于接收从编码设备输出的信号的接收器(未示出)还可以被配置成解码设备300的内部/外部元件,或者接收器可以是熵解码器310的组件。此外,根据本文档的解码设备可以被称为视频/图像/图片解码设备,并且解码设备可以被分类为信息解码器(视频/图像/图片信息解码器)和样本解码器(视频/图像/图片样本解码器)。信息解码器可以包括熵解码器310,并且样本解码器可以包括解量化器321、逆变换器322、加法器340、滤波器350、存储器360、帧间预测器332和帧内预测器331中的至少一个。

[0082] 解量化器321可以将量化变换系数解量化并输出变换系数。解量化器321可以按二维块形式重排量化变换系数。在这种情况下,可以基于在编码设备中执行的系数扫描顺序来执行重排。解量化器321可以使用量化参数(例如,量化步长信息)对量化变换系数执行解

量化并且获得变换系数。

[0083] 逆变换器322对变换系数逆变换以获得残差信号(残差块、残差样本阵列)。

[0084] 预测器可以对当前块执行预测并生成包括当前块的预测样本的预测块。预测器可以基于从熵解码器310输出的关于预测的信息来确定对当前块应用帧内预测还是帧间预测并且可以确定特定帧内/帧间预测模式。

[0085] 预测器330可以基于下述各种预测方法来生成预测信号。例如,预测器不仅可应用帧内预测或帧间预测来预测一个块,而且可以同时应用帧内预测和帧间预测。这可以被称为组合帧间和帧内预测(CIIP)。另外,预测器可以基于帧内块复制(IBC)预测模式或调色板模式来预测块。IBC预测模式或调色板模式可以用于游戏等的内容图像/视频编码,例如屏幕内容编译(SCC)。IBC基本上中当前图片中执行预测,但是可以与帧间预测相似地执行,使得在当前图片中推导参考块。即,IBC可以使用本文中描述的至少一种帧间预测技术。调色板模式可以被视为帧内编译或帧内预测的示例。当应用调色板模式时,可以基于关于调色板表和调色板索引的信息用信号发送图片内的样本值。

[0086] 帧内预测器331可以参考当前图片中的样本来预测当前块。根据预测模式,所参考的样本可以位于当前块附近或者可以隔开。在帧内预测中,预测模式可以包括多个非定向模式和多个定向模式。帧内预测器331可以使用应用于邻近块的预测模式来确定应用于当前块的预测模式。

[0087] 帧间预测器332可以基于参考图片上由运动矢量指定的参考块(参考样本阵列)来推导当前块的预测块。在这种情况下,为了减少在帧间预测模式下发送的运动信息量,可以基于邻近块与当前块之间的运动信息的相关性以块、子块或样本为单位预测运动信息。运动信息可以包括运动矢量和参考图片索引。运动信息还可以包括帧间预测方向(L0预测、L1预测、Bi预测等)信息。在帧间预测的情况下,邻近块可以包括存在于当前图片中的空间邻近块和存在于参考图片中的时间邻近块。例如,帧间预测器332可以基于邻近块来配置运动信息候选列表并且基于所接收的候选选择信息来推导当前块的运动矢量和/或参考图片索引。可以基于各种预测模式来执行帧间预测,并且关于预测的信息可以包括指示当前块的帧间预测模式的信息。

[0088] 加法器340可以通过将所获得的残差信号与从预测器(包括帧间预测器332和/或帧内预测器331)输出的预测信号(预测块、预测样本阵列)相加来生成重构信号(重构图片、重构块、重构样本阵列)。如果要处理的块不存在残差,例如当应用跳过模式时,预测块可以用作重构块。

[0089] 加法器340可以被称为重构器或重构块生成器。所生成的重构信号可以用于当前图片中要处理的下一块的帧内预测,可以如下所述通过滤波输出,或者可以用于下一图片的帧间预测。

[0090] 此外,可以在图片解码处理中应用亮度映射与色度缩放(LMCS)。

[0091] 滤波器350可以通过对重构信号应用滤波来改进主观/客观图像质量。例如,滤波器350可以通过对重构图片应用各种滤波方法来生成修改的重构图片并且将修改的重构图片存储在存储器360(具体地,存储器360的DPB)中。例如,各种滤波方法可以包括去块滤波、样本自适应偏移、自适应环路滤波器、双边滤波器等。

[0092] 存储在存储器360的DPB中的(修改的)重构图片可以用作帧间预测器332中的参考

图片。存储器360可以存储推导(或解码)当前图片中的运动信息的块的运动信息和/或图片中已经重构的块的运动信息。所存储的运动信息可以被发送到帧间预测器332以用作空间邻近块的运动信息或时间邻近块的运动信息。存储器360可以存储当前图片中的重构块的重构样本并将重构样本传送至帧内预测器331。

[0093] 在本文档中,在编码设备200的滤波器260、帧间预测器221和帧内预测器222中描述的实施例可以与解码设备300的滤波器350、帧间预测器332和帧内预测器331相同或分别与之对应地应用。这也可适用于帧间预测器332和帧内预测器331。

[0094] 如上所述,在执行视频编译时,执行预测以增加压缩效率。通过此,可以生成包括用于当前块(要被编译的块)的预测样本的预测块。这里,预测块包括空间域(或像素域)中的预测样本。预测块在编码装置和解码装置中相同地推导,并且编码装置解码关于原始块与预测块之间的残差的信息(残差信息),而非原始块的原始样本值本身。通过用信号发送给装置,可以增加图像编译效率。解码设备可以基于残差信息来推导包括残差样本的残差块,通过对残差块和预测块求和来生成包括重构样本的重构块,并且生成包括重构块的重构图片。

[0095] 残差信息可以通过变换过程和量化过程来生成。例如,编码设备可以推导原始块与预测块之间的残差块,并且对包括在残差块中的残差样本(残差样本阵列)执行变换过程来推导变换系数,并且然后,通过执行变换系数的量化过程来推导量化变换系数,以将残差相关信息(通过比特流)用信号发送给解码设备。这里,残差信息可以包括位置信息、变换技术、变换核和量化参数、量化变换系数的值信息等。解码设备可以基于残差信息来执行解量化/逆变换过程并推导残差样本(或残差块)。解码设备可以基于预测块和残差块来生成重构图片。编码设备还可以对供后续图片的帧间预测参考的量化变换系数进行解量化/逆变换以推导残差块,并且基于其来生成重构图片。

[0096] 在本文档中,可以省略量化/解量化和/或变换/逆变换中的至少一个。当省略量化/解量化时,量化变换系数可以称为变换系数。当省略变换/逆变换时,变换系数可以被称为系数或残差系数,或者为了表达的统一性仍然可以被称为变换系数。

[0097] 在本文档中,量化变换系数和变换系数可以分别称为变换系数和缩放变换系数。在这种情况下,残差信息可以包括关于变换系数的信息,并且关于变换系数的信息可以通过残差编译语法用信号发送。可以基于残差信息(或关于变换系数的信息)来推导变换系数,并且可以通过对变换系数进行逆变换(缩放)来推导缩放变换系数。残差样本可以基于缩放变换系数的逆变换(变换)来推导。这也可以在本文档的其他部分中应用/表示。

[0098] 帧内预测可以指基于当前块所属于的图片(在下文中,称为当前图片)中的参考样本生成用于当前块的预测样本的预测。当对当前块应用帧内预测时,可以推导要用于当前块的帧内预测的邻近参考样本。当前块的邻近参考样本可以包括与大小为 $nW \times nH$ 的当前块的左边界相邻的样本和与左下相邻的总共 $2 \times nH$ 个样本、与当前块的上边界相邻的样本和与右上相邻的总共 $2 \times nW$ 个样本、以及与当前块的左上相邻的一个样本。可替代地,当前块的邻近参考样本可以包括多个上邻近样本和多个左邻近样本。此外,当前块的邻近参考样本可以包括与大小为 $nW \times nH$ 的当前块的右边界相邻的总共 nH 个样本、与当前块的下边界相邻的总共 nW 个样本以及当前块的右下相邻(与右下相邻)的一个样本。

[0099] 然而,当前块的邻近参考样本中的一些尚未被解码或可能不可用。在这种情况下,

解码器可以通过用可用样本替换不可用样本来配置将用于预测的邻近参考样本。可替代地,可以通过可用样本的插值来配置将用于预测的邻近参考样本。

[0100] 在推导邻近参考样本时,(i)可以基于当前块的邻近参考样本的平均值或插值来推导预测样本,并且(ii)可以基于当前块的周边参考样本当中的针对预测样本的特定(预测)方向上存在的参考样本来推导预测样本。(i)的情况可以被称为是非定向模式或非角度模式,并且(ii)的情况可以被称为定向模式或角度模式。

[0101] 另外,也可以通过邻近参考样本当中的基于当前块的预测样本的位于当前块的帧内预测模式的预测方向的第一邻近样本和位于与预测方向相反的方向的第二邻近样本的插值来生成预测样本。上述情况可以被称为线性插值帧内预测(LIP)。此外,可以使用线性模型基于亮度样本来生成色度预测样本。这种情况可以被称为LM模式。

[0102] 此外,可以基于经滤波的邻近参考样本来推导当前块的临时预测样本,并且还可以通过根据帧内预测模式在现有邻近参考样本(即,未经滤波的邻近参考样本)当中推导的至少一个参考样本与临时预测样本进行加权求和来推导当前块的预测样本。前述情况可以被称作位置相关的帧内预测(PDPC)。

[0103] 此外,可以通过在当前块的邻近多参考样本线当中选择具有最高预测准确度的参考样本线来通过使用位于对应线中的预测方向上的参考样本来推导预测样本,并且然后可以向解码设备指示(用信号发送)本文使用的参考样本线,由此执行帧内预测编码。上述情况可以被称为多参考线(MRL)帧内预测或基于MRL的帧内预测。

[0104] 此外,可以通过将当前块划分成垂直或水平子分区而基于相同帧内预测模式执行帧内预测,并且可以以子分区为单位来推导并且使用邻近参考样本。也就是说,在此情况下,用于当前块的帧内预测模式同等地应用于子分区,并且可以通过以子分区为单位来推导并且使用邻近参考样本来在一些情况下改善帧内预测性能。这种预测方法可以被称为帧内子分区(ISP)帧内预测或基于ISP的帧内预测。

[0105] 上述帧内预测方法可以被称为与帧内预测模式分开的帧内预测类型。帧内预测类型可称为诸如帧内预测技术或附加帧内预测模式之类的各种术语。例如,帧内预测类型(或附加帧内预测模式等)可以包括上述LIP、PDPC、MRL及ISP中的至少一个。除了诸如LIP、PDPC、MRL及ISP之类的特定帧内预测类型的一般帧内预测方法可以被称作正常帧内预测类型。当不应用特定帧内预测类型的情况下,通常可以应用正常帧内预测类型,并且可以基于上述帧内预测模式执行预测。同时,也可以对根据需要推导的预测样本执行后滤波。

[0106] 具体地,帧内预测过程可以包括帧内预测模式/类型确定步骤、邻近参考样本推导步骤和基于帧内预测模式/类型的预测样本推导步骤。此外,也可以对根据需要推导的预测样本执行后滤波步骤。

[0107] 当应用帧内预测时,可以使用邻近块的帧内预测模式来确定应用于当前块的帧内预测模式。例如,解码设备可以基于接收到的MPM索引,在MPM(最可能模式)列表中选择MPM候选之一,该MPM(最可能模式)列表是基于当前块的邻近块(例如,左和/或上邻近块)的帧内预测模式推导的,并且可以基于剩余的帧内预测模式信息来选择未包括在MPM候选中的剩余帧内预测模式(和平面模式)之一。MPM列表可以被配置成包括或不包括平面模式作为候选。例如,如果MPM列表包括平面模式作为候选,MPM列表可以具有6个候选。如果MPM列表不包括平面模式作为候选,MPM列表可以具有5个候选。当MPM列表不包括平面模式作为候选

时,可以用信号发送指示当前块的帧内预测模式是否不是平面模式的非平面标志(例如, `intra_luma_not_planar_flag`)。例如,可以首先用信号发送MPM标志,并且当MPM标志的值为1时可以用信号发送MPM索引和非平面标志。此外,当非平面标志的值为1时可以用信号发送MPM索引。这里,配置MPM列表不包括平面模式作为候选的原因是由于平面模式始终被视为MPM因此通过首先用信号发送标志(非平面标志)来首先检查MPM是否是平面模式,而不是说平面模式不是MPM。

[0108] 例如,可以基于MPM标志(例如, `intra_luma_mpm_flag`)指示应用于当前块的帧内预测模式是在MPM候选(和平面模式)之中还是在剩余模式之中。MPM标志值1可以指示用于当前块的帧内预测模式在MPM候选(和平面模式)内,而MPM标志值0可以指示用于当前块的帧内预测模式不在MPM候选(和平面模式)中。非平面标志(例如, `intra_luma_not_planar_flag`)值0可以指示用于当前块的帧内预测模式是平面模式,并且非平面标志值1可以指示用于当前块的帧内预测模式不是平面模式。MPM索引可以以 `mpm_idx` 或 `intra_luma_mpm_idx` 语法元素的形式被用信号发送,并且剩余的帧内预测模式信息可以以 `rem_intra_luma_pred_mode` 或 `intra_luma_mpm_remainder` 语法元素的形式被用信号发送。例如,剩余帧内预测模式信息可以通过按照预测模式编号的顺序索引来指示所有帧内预测模式当中未包括在MPM候选(和平面模式)中的剩余帧内预测模式之一。帧内预测模式可以是亮度分量(样本)的帧内预测模式。在下文中,帧内预测模式信息可以包括以下中的至少一个:MPM标志(例如, `intra_luma_mpm_flag`)、非平面标志(例如, `intra_luma_not_planar_flag`)、MPM索引(例如, `mpm_idx` 或 `intra_luma_mpm_idx`)和剩余帧内预测模式信息(`rem_intra_luma_pred_mode` 或 `intra_luma_mpm_remainder`)。在本文档中,MPM列表可以通过诸如MPM候选列表和 `candModeList` 的多种术语来指代。当MIP应用于当前块时,可以用信号发送用于MIP的单独的mpm标志(例如, `intra_mip_mpm_flag`)、mpm索引(例如, `intra_mip_mpm_idx`)和剩余的帧内预测模式信息(例如, `intra_mip_mpm_remainder`),并且不用信号发送非平面标志。

[0109] 换言之,通常,当对图像执行块分割时,待编译的当前块和邻近块具有相似的图像特性。因此,当前块和邻近块具有相同或相似的帧内预测模式的可能性很高。因此,编码器可以使用邻近块的帧内预测模式以对当前块的帧内预测模式进行编码。

[0110] 例如,编码器/解码器可以配置用于当前块的最可能模式(MPM)的列表。MPM列表可以被称为MPM候选列表。这里,MPM可以指的是用于在帧内预测模式编译中考虑当前块和邻近块之间的相似性来改进编译效率的模式。如上所述,MPM列表可以被配置为包括平面模式,或者可以被配置为排除平面模式。例如,当MPM列表包括平面模式时,MPM列表中的候选数目可以是6个。并且,如果MPM列表不包括平面模式,则MPM列表中的候选数量可以是5个。

[0111] 编码器/解码器可以配置包括5或6个MPM的MPM列表。

[0112] 为了配置MPM列表,可以考虑三种模式:默认帧内模式、邻近帧内模式和推导的帧内模式。

[0113] 对于邻近帧内模式,可以考虑两个邻近块,即左邻近块和上邻近块。

[0114] 如上所述,如果MPM列表被配置为不包括平面模式,则平面模式被从列表中排除,并且MPM列表的候选数量可以被设置为5个。

[0115] 此外,帧内预测模式当中的非定向模式(或非角度模式)可以包括基于当前块的邻

近参考样本的平均值的DC模式或基于插值的平面模式。

[0116] 当应用帧间预测时,编码设备/解码设备的预测器可以通过以块为单位执行帧间预测来推导预测样本。帧间预测可以是以依赖于除当前图片之外的图片的数据元素(例如,样本值或运动信息)的方式推导的预测。当对当前块应用帧间预测时,可以基于由参考图片索引指示的参考图片上的由运动矢量指定的参考块(参考样本阵列)推导用于当前块的预测块(预测样本阵列)。在此,为了减少在帧间预测模式中发送的运动信息量,可以基于邻近块和当前块之间的运动信息的相关性以块、子块或样本为单位来预测当前块的运动信息。运动信息可以包括运动矢量和参考图片索引。运动信息可以进一步包括帧间预测类型(L0预测、L1预测、Bi预测等)信息。在帧间预测的情况下,邻近块可以包括存在于当前图片中的空间邻近块和存在于参考图片中的时间邻近块。包括参考块的参考图片和包括时间邻近块的参考图片可以彼此相同或彼此不同。时间邻近块可以被称为并置参考块、并置CU(co1CU)等,并且包括时间邻近块的参考图片可以被称为并置图片(co1Pic)。例如,可以基于当前块的邻近块来配置运动信息候选列表,并且可以用信号发送指示选择(使用)哪个候选的标志或索引信息以便推导当前块的运动矢量和/或参考图片索引。可以基于各种预测模式来执行帧间预测。例如,在跳过模式和合并模式的情况下,当前块的运动信息可以与邻近块的运动信息相同。在跳过模式的情况下,与合并模式不同,可以不发送残差信号。在运动矢量预测(MVP)模式的情况下,所选择的邻近块的运动矢量可以用作运动矢量预测子并且可以用信号发送当前块的运动矢量。在这种情况下,可以使用运动矢量预测子和运动矢量差的和来推导当前块的运动矢量。

[0117] 运动信息可以包括根据帧间预测类型(L0预测、L1预测、Bi预测等)的L0运动信息和/或L1运动信息。L0方向的运动矢量可以被称为L0运动矢量或MVL0,而L1方向的运动矢量可以被称为L1运动矢量或MVL1。基于L0运动矢量的预测可以称为L0预测,基于L1运动矢量的预测可以称为L1预测,并且基于L0运动矢量和L1运动矢量这两者的预测可以称为双预测。这里,L0运动矢量可以指示与参考图片列表L0相关联的运动矢量,并且L1运动矢量可以指示与参考图片列表L1相关联的运动矢量。参考图片列表L0可以包括按照输出顺序在当前图片之前的图片作为参考图片,并且参考图片列表L1可以包括按照输出顺序在当前图片之后的图片。先前图片可以被称为前向(参考)图片并且后续图片可以被称为反向(参考)图片。参考图片列表L0可以进一步包括按照输出顺序在当前图片之后的图片作为参考图片。在这种情况下,可以首先在参考图片列表L0中对先前图片进行索引,并且然后可以对后续图片进行索引。参考图片列表L1可以进一步包括按照输出顺序在当前图片之前的图片作为参考图片。在这种情况下,可以首先在参考图片列表L1中对后续图片进行索引,并且然后可以对先前图片进行索引。输出顺序可以对应于图片顺序计数(POC)顺序。

[0118] 基于帧间预测的视频/图像编码过程可以包括例如以下内容。

[0119] 图4示出基于帧间预测的视频/图像编码方法的示例。

[0120] 编码设备对当前块执行帧间预测(S400)。编码设备可以推导当前块的帧间预测模式和运动信息,并且生成当前块的预测样本。这里,确定帧间预测模式、推导运动信息和生成预测样本的过程可以同时执行,或者任一个过程可以在其它过程之前执行。例如,编码设备的帧间预测器可以包括预测模式确定器、运动信息推导器和预测样本推导器,并且预测模式确定器可以确定当前块的预测模式,运动信息推导器可推导当前块的运动信息,并且

预测样本推导器可推导当前块的预测样本。例如,编码设备的帧间预测器可以通过运动估计在参考图片的特定区域(搜索区域)中搜索与当前块相似的块,并且推导与当前块的差最小或者特定参考或以下的参考块。基于此,可以推导指示参考块所在的参考图片的参考图片索引,并且可以基于参考块与当前块之间的位置差来推导运动矢量。编码设备可以在各种预测模式当中确定应用于当前块的模式。编码设备可以比较各种预测模式的RD成本并且为当前块确定最优预测模式。

[0121] 例如,如果对当前块应用跳过模式或合并模式,则编码设备可以配置稍后描述的合并候选列表,并且推导包括在合并候选列表中的合并候选所指示的参考块当中与当前块的差最小或者特定参考或以下的参考块。在这种情况下,可以选择与所推导的参考块相关联的合并候选,并且可以生成指示所选合并候选的合并索引信息并用信号发送给解码设备。可以使用所选合并候选的运动信息来推导当前块的运动信息。

[0122] 作为另一示例,如果对当前块应用(A)MVP模式,则编码设备可配置稍后描述的(A)MVP候选列表,并且使用包括在(A)MVP候选列表中的运动矢量预测子(mvp)候选当中所选mvp候选的运动矢量作为当前块的mvp。在这种情况下,例如,指示通过上述运动估计推导的参考块的运动矢量可以用作当前块的运动矢量,并且mvp候选当中具有与当前块的运动矢量的差最小的运动矢量的mvp候选可以是所选mvp候选。可推导运动矢量差(MVD),其是通过从当前块的运动矢量减去mvp而获得的差。在这种情况下,可以将关于MVD的信息用信号发送给解码设备。另外,如果应用(A)MVP模式,则参考图片索引的值可以被配置为参考图片索引信息并单独地用信号发送给解码设备。

[0123] 编码设备可以基于预测样本来推导残差样本(S410)。编码设备可以通过将当前块的原始样本与预测样本进行比较来推导残差样本。

[0124] 编码设备对包括预测信息和残差信息的图像信息进行编码(S420)。编码设备可按比特流的形式输出编码图像信息。预测信息可以包括关于预测模式信息(例如,跳过标志、合并标志、模式索引等)和运动信息的信息作为与预测过程有关的信息。关于运动信息的信息可以包括作为用于推导运动矢量的信息的候选选择信息(例如,合并索引、mvp标志或mvp索引)。另外,关于运动信息的信息可以包括上述关于MVD的信息和/或参考图片索引信息。另外,关于运动信息的信息可以包括指示是否应用L0预测、L1预测或bi预测的信息。残差信息是关于残差样本的信息。残差信息可以包括关于残差样本的量化变换系数的信息。

[0125] 输出比特流可以被存储在(数字)存储介质中并传送至解码设备,或者也可以经由网络传送至解码设备。

[0126] 此外,如上所述,编码设备可以基于参考样本和残差样本来生成重构图片(包括重构样本和重构块)。这是为了由编码设备推导解码设备所推导的相同预测结果,因此可以增加编译效率。因此,编码设备可以将重构图片(或重构样本、重构块)存储在存储器中,并且将其用作帧间预测的参考图片。如上所述,还可以对重构图片应用环内滤波过程等。

[0127] 基于帧间预测的视频/图像解码过程可以包括例如以下内容。

[0128] 图5示出基于帧间预测的视频/图像解码方法的示例。

[0129] 参考图5,解码设备可以执行与编码设备所执行的操作对应的操作。解码设备可以基于所接收的预测信息对当前块执行预测并推导预测样本。

[0130] 具体地,解码设备可以基于所接收的预测信息来确定当前块的预测模式(S500)。

解码设备可以基于预测信息中的预测模式信息来确定应用于当前块的帧间预测模式。

[0131] 例如,解码设备可以基于合并标志来确定是否对当前块应用合并模式或者确定(A) MVP模式。另选地,解码设备可以基于模式索引来选择各种帧间预测模式候选之一。帧间预测模式候选可以包括跳过模式、合并模式和/或(A) MVP模式,或者可以包括稍后描述的各种帧间预测模式。

[0132] 解码设备基于所确定的帧间预测模式来推导当前块的运动信息(S510)。例如,如果对当前块应用跳过模式或合并模式,则解码设备可以配置稍后描述的合并候选列表,并且在包括在合并候选列表中的合并候选当中选择一个合并候选。可以基于上述选择信息(合并索引)执行选择。可以使用所选合并候选的运动信息来推导当前块的运动信息。所选合并候选的运动信息可以用作当前块的运动信息。

[0133] 作为另一示例,如果对当前块应用(A) MVP模式,则解码设备可以配置稍后描述的(A) MVP候选列表,并且使用包括在(A) MVP候选列表中的运动矢量预测子(mvp)候选当中所选mvp候选的运动矢量,作为当前块的mvp。可以基于上述选择信息(mvp标志或mvp索引)执行选择。在这种情况下,可以基于关于MVD的信息来推导当前块的MVD,并且可以基于当前块的mvp和MVD来推导当前块的运动矢量。另外,可以基于参考图片索引信息来推导当前块的参考图片索引。当前块上参考图片列表中的参考图片索引所指示的图片可以被推导为当前块的帧间预测所参考的参考图片。

[0134] 此外,如稍后描述的,可在不配置候选列表的情况下推导当前块的运动信息,并且在在这种情况下,可以根据稍后描述的预测模式中公开的过程来推导当前块的运动信息。在这种情况下,可省略上述候选列表的配置。

[0135] 解码设备可以基于当前块的运动信息来生成当前块的预测样本(S520)。在这种情况下,解码设备可以基于当前块的参考图片索引来推导参考图片,并且使用参考图片上当前块的运动矢量所指示的参考块的样本来推导当前块的预测样本。在这种情况下,如稍后描述的,在一些情况下还可以执行对当前块的所有或一些预测样本的预测样本滤波过程。

[0136] 例如,解码设备的帧间预测器可以包括预测模式确定器、运动信息推导器和预测样本推导器,并且预测模式确定器可以基于所接收的预测模式信息来确定当前块的预测模式,运动信息推导器可以基于关于所接收的运动信息的信息来推导当前块的运动信息(例如运动矢量和/或参考图片索引),预测样本推导器可以推导当前块的预测样本。

[0137] 解码设备基于所接收的残差信息来生成当前块的残差样本(S530)。解码设备可以基于预测样本和残差样本来生成当前块的重构样本,并且基于其来生成重构图片(S540)。此后,如上所述还可以对重构图片应用环内滤波过程等。

[0138] 图6示例性地示出帧间预测过程。

[0139] 参考图6,如上所述,帧间预测过程可以包括:确定帧间预测模式,根据所确定的预测模式来推导运动信息,并且基于所推导的运动信息来执行预测(生成预测样本)。帧间预测过程可以由如上所述的编码设备和解码设备执行。本文档中的编译设备可以包括编译设备和/或解码设备。

[0140] 参考图6,编译设备为当前块确定帧间预测模式(S600)。各种帧间预测模式可以用于预测图片中的当前块。例如,可以使用诸如合并模式、跳过模式、运动矢量预测(MVP)模式、仿射模式、子块合并模式以及与MVD合并(MMVD)模式的各种模式。另外或作为附带模式

代替这些模式,可以使用解码器侧运动矢量细化(DMVR)模式、自适应运动矢量分辨率(AMVR)模式、具有CU级别权重的双预测(BCW)、双向光流(BDOF)等。仿射模式可以被称为仿射运动预测模式。MVP模式可以被称为高级运动矢量预测(AMVP)模式。在本文档中,一些模式和/或通过一些模式推导的运动信息候选也可以被包括作为另一模式下的运动信息相关候选之一。例如,HMVP候选可以在合并/跳过模式下被添加作为合并候选,或者可以在MVP模式下被添加作为MVP候选。如果在合并模式或跳过模式下HMVP候选用作运动信息候选,则HMVP候选可以被称为HMVP合并候选。

[0141] 指示当前块的帧间预测模式的预测模式信息可以从编码设备用信号发送到解码设备。预测模式信息可以被包括在比特流中并且由解码设备接收。预测模式信息可以包括指示多个候选模式之一的索引信息。可替代地,可以通过标志信息的分层信令来指示帧间预测模式。在这种情况下,预测模式信息可以包括一个或多个标志。例如,可以通过用信号发送跳过标志来指示是否应用跳过模式,并且如果不应用跳过模式,可以通过用信号发送合并标志来指示是否应用合并模式,以及指示应用MVP模式或者可以进一步用信号发送用于附加分类的标志。仿射模式可以作为独立模式被用信号发送或作为关于合并模式或MVP模式的从属模式被用信号发送。例如,仿射模式可以包括仿射合并模式和仿射MVP模式。

[0142] 同时,指示在当前块(当前编译单元)中是否使用上述list0(L0)预测、list1(L1)预测或双预测的信息可以在当前块中用信号发送。该信息可以称为运动预测方向信息、帧间预测方向信息或帧间预测指示信息,并且可以以例如inter_pred_idc语法元素的形式配置/编码/用信号发送。即,inter_pred_idc语法元素可以指示前述list0(L0)预测、list1(L1)预测或双预测是否用于当前块(当前编译单元)。在本文档中,为了便于描述,可以将由inter_pred_idc语法元素指示的帧间预测类型(L0预测、L1预测或BI预测)指示为运动预测方向。L0预测可以表示为pred_L0,L1预测表示为pred_L1,双预测表示为pred_BI。例如,可以根据inter_pred_idc语法元素的值来指示以下预测类型。

[0143] [表1]

inter_pred_idc 的值	inter_pred_idc 的名称	
	(cbWidth + cbHeight) != 8	(cbWidth + cbHeight) == 8
0	PRED_L0	PRED_L0
1	PRED_L1	PRED_L1
2	PRED_BI	-

[0145] 如上所述,一个图片可以包括一个或多个切片。切片可具有包括帧内(I)切片、预测(P)切片和双预测(B)切片的切片类型中的一个。切片类型可以基于切片类型信息来指示。针对I切片中的块,帧间预测可以不用于预测,并且可以仅使用帧内预测。当然,即使在这种情况下,也可以中不进行预测的情况下对原始样本值进行编译并且用信号发送。对于P切片中的块,可以使用帧内预测或帧间预测,并且在使用帧间预测的情况下,可以仅使用单向预测。同时,对于B切片中的块,可以使用帧内预测或帧间预测,并且在使用帧间预测的情况下,可以使用直至双预测。

[0146] L0和L1可以包括在当前图片之前先前编码/解码的参考图像。例如,L0可以包括按

POC顺序在当前图片之前和/或之后的参考图片,且L1可以包括按POC顺序在当前图片之后和/或之前的参考图片。在此情况下,L0可以被指配相对于先前参考图片按POC顺序比当前参考图片更低的参考图片索引,并且L1可以被指配相对于先前参考图片按POC顺序比当前参考图片更低的参考图片索引。在B切片的情况下,可以应用双预测,并且在此情况下,可以应用单向双预测,或可以应用双向双预测。双向双预测可以被称为真双预测。

[0147] 下面的表示出根据本文档的实施例的编译单元的语法。

[0148] [表2]

	coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	描述符
	if(slice_type != I sps_ibc_enabled_flag) {	
	if(treeType != DUAL_TREE_CHROMA && !(cbWidth == 4 && cbHeight == 4 && !sps_ibc_enabled_flag))	
	cu_skip_flag[x0][y0]	ae(v)
	if(cu_skip_flag[x0][y0] == 0 && slice_type != I && !(cbWidth == 4 && cbHeight == 4))	
	pred_mode_flag	ae(v)
	if(((slice_type == I && cu_skip_flag[x0][y0] == 0) (slice_type != I && (CuPredMode[x0][y0] != MODE_INTRA (cbWidth == 4 && cbHeight == 4 && cu_skip_flag[x0][y0] == 0)))) && sps_ibc_enabled_flag && (cbWidth != 128 cbHeight != 128))	
	pred_mode_ibc_flag	ae(v)
	}	
	if(CuPredMode[x0][y0] == MODE_INTRA) {	
	if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
	pcm_flag[x0][y0]	ae(v)
	if(pcm_flag[x0][y0]) {	
	while(!byte_aligned())	
	pcm_alignment_zero_bit	f(1)
[0149]	pcm_sample(cbWidth, cbHeight, treeType)	
	} else {	
	if(treeType == SINGLE_TREE treeType == DUAL_TREE_LUMA) {	
	if(cbWidth <= 32 && cbHeight <= 32)	
	intra_bdpcm_flag[x0][y0]	ae(v)
	if(intra_bdpcm_flag[x0][y0])	
	intra_bdpcm_dir_flag[x0][y0]	ae(v)
	else {	
	if(sps_mip_enabled_flag && (Abs(Log2(cbWidth) - Log2(cbHeight)) <= 2) && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
	intra_mip_flag[x0][y0]	ae(v)
	if(intra_mip_flag[x0][y0]) {	
	intra_mip_mpm_flag[x0][y0]	ae(v)
	if(intra_mip_mpm_flag[x0][y0])	
	intra_mip_mpm_idx[x0][y0]	ae(v)
	else	
	intra_mip_mpm_remainder[x0][y0]	ae(v)
	} else {	
	if(sps_mrl_enabled_flag && ((y0 % CtbSizeY) > 0))	
	intra_luma_ref_idx[x0][y0]	ae(v)

[0150]

if (sps_isp_enabled_flag && intra_luma_ref_idx[x0][y0] == 0 && (cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY) && (cbWidth * cbHeight > MinTbSizeY * MinTbSizeY))	
intra_subpartitions_mode_flag[x0][y0]	ae(v)
if(intra_subpartitions_mode_flag[x0][y0] == 1 && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
intra_subpartitions_split_flag[x0][y0]	ac(v)
if(intra_luma_ref_idx[x0][y0] == 0 && intra_subpartitions_mode_flag[x0][y0] == 0)	
intra_luma_mpm_flag[x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0]) {	
if(intra_luma_ref_idx[x0][y0] == 0)	
intra_luma_not_planar_flag[x0][y0]	ac(v)
if(intra_luma_not_planar_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ac(v)
} else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
}	
}	
if(treeType == SINGLE_TREE treeType == DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ac(v)
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
if(cu_skip_flag[x0][y0] == 0)	
general_merge_flag[x0][y0]	ae(v)
if(general_merge_flag[x0][y0]) {	
merge_data(x0, y0, cbWidth, cbHeight)	
} else if (CuPredMode[x0][y0] == MODE_IBC) {	
mvd_coding(x0, y0, 0, 0)	
mvp_l0_flag[x0][y0]	ae(v)
if(sps_amvr_enabled_flag && (MvdL0[x0][y0][0] != 0 MvdL0[x0][y0][1] != 0)) {	
amvr_precision_flag[x0][y0]	ae(v)
}	
} else {	
if(slice_type == B)	
inter_pred_idc[x0][y0]	ae(v)
if(sps_affine_enabled_flag && cbWidth >= 16 && cbHeight >= 16) {	
inter_affine_flag[x0][y0]	ae(v)
if(sps_affine_type_flag && inter_affine_flag[x0][y0])	
cu_affine_type_flag[x0][y0]	ae(v)
}	
if(sps_symvd_enabled_flag && inter_pred_idc[x0][y0] == PRED_BI && !inter_affine_flag[x0][y0] && RefIdxSymL0 > -1 && RefIdxSymL1 > -1)	
sym_mvd_flag[x0][y0]	ae(v)
if(inter_pred_idc[x0][y0] != PRED_L1) {	
if(NumRefIdxActive[0] > 1 && !sym_mvd_flag[x0][y0])	

[0151]

ref_idx_10 [x0][y0]	ac(v)
mvd_coding(x0, y0, 0, 0)	
if(MotionModelIdx[x0][y0] > 0)	
mvd_coding(x0, y0, 0, 1)	
if(MotionModelIdx[x0][y0] > 1)	
mvd_coding(x0, y0, 0, 2)	
mvp_10_flag [x0][y0]	ac(v)
} else {	
MvdL0[x0][y0][0] = 0	
MvdL0[x0][y0][1] = 0	
}	
if(inter_pred_idc[x0][y0] != PRED_L0) {	
if(NumRefIdxActive[1] > 1 && !sym_mvd_flag[x0][y0])	
ref_idx_11 [x0][y0]	ac(v)
if(mvd_11_zero_flag && inter_pred_idc[x0][y0] == PRED_BI) {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
MvdCpL1[x0][y0][0][0] = 0	
MvdCpL1[x0][y0][0][1] = 0	
MvdCpL1[x0][y0][1][0] = 0	
MvdCpL1[x0][y0][1][1] = 0	
MvdCpL1[x0][y0][2][0] = 0	
MvdCpL1[x0][y0][2][1] = 0	
} else {	
if(sym_mvd_flag[x0][y0]) {	
MvdL1[x0][y0][0] = -MvdL0[x0][y0][0]	
MvdL1[x0][y0][1] = -MvdL0[x0][y0][1]	
} else	
mvd_coding(x0, y0, 1, 0)	
if(MotionModelIdx[x0][y0] > 0)	
mvd_coding(x0, y0, 1, 1)	
if(MotionModelIdx[x0][y0] > 1)	
mvd_coding(x0, y0, 1, 2)	
mvp_11_flag [x0][y0]	ac(v)
}	
} else {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
}	

[0152]

<pre> if((sps_amvr_enabled_flag && inter_affine_flag[x0][y0] == 0 && (MvdL0[x0][y0][0] != 0 MvdL0[x0][y0][1] != 0 MvdL1[x0][y0][0] != 0 MvdL1[x0][y0][1] != 0)) (sps_affine_amvr_enabled_flag && inter_affine_flag[x0][y0] == 1 && (MvdCpL0[x0][y0][0][0] != 0 MvdCpL0[x0][y0][0][1] != 0 MvdCpL1[x0][y0][0][0] != 0 MvdCpL1[x0][y0][0][1] != 0 MvdCpL0[x0][y0][1][0] != 0 MvdCpL0[x0][y0][1][1] != 0 MvdCpL1[x0][y0][1][0] != 0 MvdCpL1[x0][y0][1][1] != 0 MvdCpL0[x0][y0][2][0] != 0 MvdCpL0[x0][y0][2][1] != 0 MvdCpL1[x0][y0][2][0] != 0 MvdCpL1[x0][y0][2][1] != 0)) { </pre>	
<pre> amvr_flag[x0][y0] </pre>	ac(v)
<pre> if(amvr_flag[x0][y0]) </pre>	
<pre> amvr_precision_flag[x0][y0] </pre>	ac(v)
<pre> } </pre>	
<pre> if(sps_bcw_enabled_flag && inter_pred_idc[x0][y0] == PRED_B1 && luma_weight_10_flag[ref_idx_10 [x0][y0]] == 0 && luma_weight_11_flag[ref_idx_11 [x0][y0]] == 0 && chroma_weight_10_flag[ref_idx_10 [x0][y0]] == 0 && chroma_weight_11_flag[ref_idx_11 [x0][y0]] == 0 && cbWidth * cbHeight >= 256) </pre>	
<pre> bcw_idx[x0][y0] </pre>	ae(v)
<pre> } </pre>	
<pre> } </pre>	
<pre> if(!pcm_flag[x0][y0]) { </pre>	
<pre> if(CuPredMode[x0][y0] != MODE_INTRA && general_merge_flag[x0][y0] == 0) </pre>	
<pre> cu_cbf </pre>	ae(v)

	if(cu_cbf) {	
	if(CuPredMode[x0][y0] == MODE_INTER && sps_sbt_enabled_flag	
	&& !ciip_flag[x0][y0] && !MergeTriangleFlag[x0][y0]) {	
	if(cbWidth <= MaxSbtSize && cbHeight <= MaxSbtSize) {	
	allowSbtVerH = cbWidth >= 8	
	allowSbtVerQ = cbWidth >= 16	
	allowSbtHorH = cbHeight >= 8	
	allowSbtHorQ = cbHeight >= 16	
	if(allowSbtVerH allowSbtHorH allowSbtVerQ	
	allowSbtHorQ)	
	cu_sbt_flag	ae(v)
	}	
	if(cu_sbt_flag) {	
	if((allowSbtVerH allowSbtHorH) && (allowSbtVerQ	
	allowSbtHorQ)	
	cu_sbt_quad_flag	ae(v)
	if((cu_sbt_quad_flag && allowSbtVerQ && allowSbtHorQ)	
	(!cu_sbt_quad_flag && allowSbtVerH && allowSbtHorH))	
	cu_sbt_horizontal_flag	ac(v)
	cu_sbt_pos_flag	ae(v)
	}	
	}	
	numSigCoeff = 0	
	numZeroOutSigCoeff = 0	
	transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
	lfstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC	
	: cbWidth	
	lfstHeight = (treeType ==	
	DUAL_TREE_CHROMA) ? cbHeight / SubHeightC	
	: cbHeight	
	if(Min(lfstWidth, lfstHeight) >= 4 && sps_lfst_enabled_flag == 1	
	&&	
	CuPredMode[x0][y0] == MODE_INTRA &&	
	IntraSubPartitionsSplitType == ISP_NO_SPLIT &&	
	!intra_mip_flag[x0][y0]) {	
	if((numSigCoeff > ((treeType == SINGLE_TREE) ? 2 : 1)) &&	
	numZeroOutSigCoeff == 0)	
	lfst_idx[x0][y0]	ac(v)
	}	
	}	
	}	
	}	

[0153]

[0154] 参考表2, general_merge_flag可以指示一般合并可用,并且当general_merge_flag的值为1时,常规合并模式、mmvd模式和合并子块模式(子块合并模式)可能可用。例如,当general_merge_flag的值为1时,可以从编码后的视频/图像信息(或比特流)解析合并数据语法,并且可以将合并数据语法配置/编译为包括如下表所示的信息。

[0155] [表3]

	merge_data(x0, y0, cbWidth, cbHeight) {	描述符
	if (CuPredMode[x0][y0] == MODE_IBC) {	
	if(MaxNumMergeCand > 1)	
	merge_idx [x0][y0]	ae(v)
	} else {	
	if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
	regular_merge_flag [x0][y0]	ae(v)
	if (regular_merge_flag[x0][y0] == 1) {	
	if(MaxNumMergeCand > 1)	
	merge_idx [x0][y0]	ae(v)
	} else {	
	if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
	mmvd_merge_flag [x0][y0]	ae(v)
	if(mmvd_merge_flag[x0][y0] == 1) {	
	if(MaxNumMergeCand > 1)	
	mmvd_cand_flag [x0][y0]	ae(v)
	mmvd_distance_idx [x0][y0]	ae(v)
	mmvd_direction_idx [x0][y0]	ae(v)
	} else {	
	if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >=	
[0156]	8)	
	merge_subblock_flag [x0][y0]	ae(v)
	if(merge_subblock_flag[x0][y0] == 1) {	
	if(MaxNumSubblockMergeCand > 1)	
	merge_subblock_idx [x0][y0]	ae(v)
	} else {	
	if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 &&	
	(cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128)	
	{	
	ciip_flag [x0][y0]	ae(v)
	if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
	merge_idx [x0][y0]	ae(v)
	}	
	if(MergeTriangleFlag[x0][y0]) {	
	merge_triangle_split_dir [x0][y0]	ae(v)
	merge_triangle_idx0 [x0][y0]	ae(v)
	merge_triangle_idx1 [x0][y0]	ae(v)
	}	
	}	
	}	
	}	
	}	
	}	

[0157] 编译设备推导用于当前块的运动信息(S610)。可以基于帧间预测模式来推导运动信息。

[0158] 编码设备可使用当前块的运动信息来执行帧间预测。编码设备可以通过运动估计过程来推导当前块的最优运动信息。例如,编码设备可针对当前块使用原始图片中的原始块在参考图片中的预定搜索范围内以分数像素为单位搜索具有高相关性的相似参考块,从而推导运动信息。可以根据基于相位的样本值之间的差来推导块的相似度。例如,可以基于

当前块(或当前块的模板)与参考块(或参考块的模板)之间的SAD来计算块的相似度。在这种情况下,可以基于在搜索区域中具有最小SAD的参考块来推导运动信息。可以基于帧间预测模式根据各种方法将所推导的运动信息用信号发送给解码设备。

[0159] 编译设备基于当前块的运动信息来执行帧间预测(S620)。编译设备可以基于运动信息来推导当前块的预测样本。包括预测样本的当前块可以被称为预测块。

[0160] 当应用合并模式时,不直接发送当前预测块的运动信息,并且使用邻近预测块的运动信息来推导当前预测块的运动信息。因此,可以通过发送指示使用合并模式的标记信息和指示使用哪个邻近预测块的合并索引来指示当前预测块的运动信息。合并模式可以称为常规合并模式。

[0161] 编码器必须搜索用于推导当前预测块的运动信息以执行合并模式的合并候选块。例如,可以使用最多五个合并候选块,但是本文档的实施例不限于此。可以在切片报头或拼块组报头中发送合并候选块的最大数量,并且本文档的实施例不限于此。在找到合并候选块之后,编码器可以生成合并候选列表,并且选择其中成本最小的合并候选块作为最终合并候选块。

[0162] 合并候选列表可以使用例如五个合并候选块。例如,可以使用四个空间合并候选和一个时间合并候选。在下文中,稍后描述的空间合并候选或空间MVP候选可以被称为SMVP,并且稍后描述的时间合并候选或时间MVP候选可以被称为TMVP。

[0163] 下面,描述根据本文档构建合并候选列表的方法。

[0164] 编译设备(编码器/解码器)将通过搜索当前块的空间邻近块而推导的空间合并候选插入合并候选列表中。例如,空间邻近块可以包括当前块的左下角邻近块、左邻近块、右上角邻近块、上邻近块和左上角邻近块。然而,这是示例,并且除了上述空间邻近块之外,诸如右邻近块、下邻近块和右下邻近块的附加邻近块可以进一步用作空间邻近块。编译设备可以通过基于优先级搜索空间邻近块来检测可用块,并且可以推导检测到的块的运动信息作为空间合并候选。

[0165] 编译设备将通过搜索当前块的时间邻近块而推导的时间合并候选插入合并候选列表中。时间邻近块可以位于作为不同于当前块所在的当前图片的参考图片上。时间邻近块所在的参考图片可以称为并置图片或col图片。可以按col图片上的当前块的并置块的右下角邻近块和右下中心块的顺序搜索时间邻近块。同时,当应用运动数据压缩时,可以将特定运动信息存储为col图片中的每个预定存储单元的代表性运动信息。在这种情况下,不需要将所有块的运动信息存储在预定存储单元中,从而获得运动数据压缩效果。在这种情况下,可以例如以16x16个样本单元、8x8个样本单元等先前地确定预定存储单元,或者可以从编码器向解码器用信号发送关于预定存储单元的大小信息。当应用运动数据压缩时,时间邻近块的运动信息可以用时间邻近块所在的预定存储单元的代表性运动信息替换。也就是说,在这种情况下,从实现的角度来看,基于时间邻近块的坐标(左上样本位置)将预定值在算术上向右移位,并且此后,可以基于覆盖在算术上左移位位置的预测块的运动信息来推导时间合并候选。例如,在具有预定存储单元的样本单元是 $2n \times 2n$ 的情况下,如果时间邻近块的坐标是 (x_{Tnb}, y_{Tnb}) ,则位于修改位置 $((x_{Tnb} \gg n) \ll n)$, $(y_{Tnb} \gg n) \ll n$ 处的预测块的运动信息可以用于时间合并候选。具体地,例如,在预定存储单元是16x16样本单元的情况下,如果时间邻近块的坐标是 (x_{Tnb}, y_{Tnb}) ,则位于修改位置 $((x_{Tnb} \gg 4) \ll$

4), $(y_{Tnb} \gg 4) \ll 4$) 处的预测块的运动信息可以用于时间合并候选。或者,例如,在预定存储单元是 8×8 样本单元的情况下,如果时间邻近块的坐标是 (x_{Tnb}, y_{Tnb}) , 则位于修改位置 $((x_{Tnb} \gg 3) \ll 3)$, $(y_{Tnb} \gg 3) \ll 3$) 处的预测块的运动信息可以用于时间合并候选。

[0166] 编译设备可以确定当前合并候选的数量是否小于最大合并候选的数量。最大合并候选的数目可以被预定义或从编码器用信号发送到解码器。例如,编码器可以生成关于最大合并候选的数目的信息,对该信息进行编码,并且以比特流的形式将编码信息发送到解码器。如果填满最大合并候选的数目,则可以不执行后续候选添加过程。

[0167] 作为检查的结果,如果当前合并候选的数量小于最大合并候选的数量,则编译设备将附加合并候选插入合并候选列表中。

[0168] 作为检查的结果,如果当前合并候选的数目不小于最大合并候选的数目,则编译设备可以终止合并候选列表的构造。在这种情况下,编码器可以基于速率失真(RD)成本在配置合并候选列表的合并候选当中选择最佳合并候选并且向解码器用信号发送指示选定合并候选的选择信息(例如合并索引)。解码器可以基于合并候选列表和选择信息来选择最佳合并候选。

[0169] 选定合并候选的运动信息可以用作当前块的运动信息,并且可以基于当前块的运动信息来推导当前块的预测样本。编码器可以基于预测样本来推导当前块的残差样本,并且可以将关于残差样本的残差信息用信号发送到解码器。解码器可以基于残差样本和基于残差信息推导的预测样本来生成重构样本,并且如上所述基于该重构样本来生成重构图片。

[0170] 当应用跳过模式时,可以以与应用合并模式的情况相同的方式推导当前块的运动信息。然而,当应用跳过模式时,省略对应块的残差信号,因此预测样本可以用作重构样本。

[0171] 当应用MVP模式时,可以使用重构空间邻近块的运动矢量和/或时间邻近块(或Co1块)的运动矢量来生成运动矢量预测子(mvp)候选列表。即,可以将对应于重构空间邻近块的运动矢量和/或对应于时间邻近块的运动矢量用作运动矢量预测子候选。当应用双预测时,可以分别生成并且使用用于推导L0运动信息的mvp候选列表和用于推导L1运动信息的mvp候选列表。上述预测信息(或关于预测的信息)可以包括指示从包括在列表中的运动矢量预测子候选中选择的最佳运动矢量预测子候选的选择信息(例如,MVP标志或MVP索引)。在这种情况下,预测器可以使用选择信息从包括在运动矢量候选列表中的运动矢量预测子候选中选择当前块的运动矢量预测子。编码设备的预测子可以获得当前块的运动矢量和运动矢量预测子之间的运动矢量差(MVD),对它进行编码,并且以比特流形式输出它。即,MVD可以被获得作为通过从当前块的运动矢量减去运动矢量预测子而获得的值。在这种情况下,解码设备的预测器可以获得包括在关于预测的信息中的运动矢量差,并且通过将运动矢量差与运动矢量预测子相加来推导当前块的运动矢量。解码设备的预测器可以从关于预测的信息获得或推导指示参考图片的参考图片索引。

[0172] 下面,描述根据本文档的构造运动矢量预测子候选列表的方法。

[0173] 实施例可以首先搜索用于运动矢量预测的空间候选块,并且将其插入预测候选列表中。此后,实施例可以确定空间候选块的数量是否小于2。例如,在一个实施例中,当空间候选块的数量小于2时,可以搜索时间候选块并且将其附加地插入到预测候选列表中,并且当时间候选块不可用时,可以使用零运动矢量。即,零运动矢量可以附加地插入到预测候选

列表中。此后,实施例可以结束初步候选列表的构造。可替换地,根据实施例,当空间候选块的数量不小于两个时,可以终止初步候选列表的构造。这里,初步候选列表可以指示MVP候选列表。

[0174] 同时,当应用MVP模式时,可以显式地用信号发送参考图片索引。在这种情况下,用于L0预测的参考图片索引refIdxL0和用于L1预测的参考图片索引refIdxL1可以分别用信号发送。例如,当应用MVP模式并且应用BI预测时,可以用信号发送关于refIdxL0的信息和关于refIdxL1的信息两者。

[0175] 当应用MVP模式时,如上所述,可以向解码设备用信号发送从编码设备推导的关于MVD的信息。关于MVD的信息可以包括例如表示MVD绝对值和符号的x和y分量的信息。在这种情况下,指示MVD绝对值是否大于0且大于1的信息和MVD剩余可以被逐步地用信号发送。例如,仅当指示MVD绝对值是否大于0的标志信息的值为1时,才可以用信号发送指示MVD绝对值是否大于1的信息。

[0176] 例如,可以将关于MVD的信息配置为以下语法,在编码设备中编码,并且用信号发送给解码设备。

[0177] [表4]

	描述符
mvd_coding(x0, y0, refList, cpIdx) {	
abs_mvd_greater0_flag[0]	ae(v)
abs_mvd_greater0_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0])	
abs_mvd_greater1_flag[0]	ae(v)
if(abs_mvd_greater0_flag[1])	
abs_mvd_greater1_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0]) {	
if(abs_mvd_greater1_flag[0])	
abs_mvd_minus2[0]	ae(v)
mvd_sign_flag[0]	ae(v)
}	
if(abs_mvd_greater0_flag[1]) {	
if(abs_mvd_greater1_flag[1])	
abs_mvd_minus2[1]	ae(v)
mvd_sign_flag[1]	ae(v)
}	
}	
}	

[0178] 例如,在表4中,abs_mvd_greater0_flag语法元素可以指示关于差MVD是否大于0的信息,并且abs_mvd_greater1_flag语法元素可以指示关于差MVD是否大于1的信息。此外,abs_mvd_minus2语法元素可以指示关于通过对差MVD-2获得的值的信息,并且mvd_sign_flag语法元素可以指示关于差MVD的符号的信息。此外,在表4中,每个语法元素的[0]可以指示关于L0的信息,并且[1]可以指示关于L1的信息。

[0180] 例如,MVD[compIdx]可以基于abs_mvd_greater0_flag[compIdx]*(abs_mvd_minus2[compIdx]+2)*(1-2 * mvd_sign_flag[compIdx])来推导。这里,compIdx(或cpIdx)表示每个分量的索引,并且可以具有值0或1。compIdx 0可以指示X分量,compIdx 1可以指示Y分量。然而,这仅仅是示例,并且可以通过使用除x和y坐标系之外的坐标系来表达每个分量的值。

[0181] 同时,用于L0预测的MVD (MVDL0) 和用于L1预测的MVD (MVDL1) 可以单独地用信号发送,并且关于MVD的信息可以包括关于MVDL0的信息和/或关于MVDL1的信息。例如,当MVP模式应用于当前块并且应用BI预测时,可以用信号发送关于MVDL0的信息和关于MVDL1的信息两者。

[0182] 图7是用于描述对称运动矢量差 (SMVD) 的图。

[0183] 当应用BI预测时,考虑到编译效率,可以使用对称MVD。在这种情况下,可以省略一些运动信息的信令。例如,当对称MVD应用于当前块时,可以不将关于refidxL0的信息、关于refidxL1的信息和关于MVDL1的信息从编码设备用信号发送到解码设备,并且可以在内部推导。例如,当MVP模式和BI预测应用于当前块时,指示是否应用对称MVD的标志信息(例如对称MVD标志信息或sym_mvd_flag语法元素)可以被用信号发送,并且当标志信息的值为1时,解码设备可以确定对称MVD应用于当前块。

[0184] 当应用对称MVD模式时(即,对称MVD标志信息的值为1),可以显式地用信号发送关于mvp_l0_flag、mvp_l1_flag和MVDL0的信息,并且如上所述,可以省略并在内部推导关于refidxL0的信息、关于refidxL1的信息和关于MVDL1的信息的信令。例如,可以推导refidxL0作为索引,该索引指示在参考图片列表0(其可以称为列表0或L0)中按POC顺序最接近当前图片的先前参考图片。可以推导refidxL1作为索引,该索引指示在参考图片列表1(其可以称为列表1或L1)中按POC顺序中最接近当前图片的下一个参考图片。或者,例如,可以将refidxL0和refidxL1两者都推导为0。或者,例如,可以将refidxL0和refidxL1推导为与当前图片的关系具有相同POC差的最小索引。具体地,例如,当[当前图片的POC] - [由refidxL0指示的第一参考图片的POC]为第一POC差并且[当前图片的POC] - [由refidxL1指示的第二参考图片的POC]为第二POC差时,只有当第一POC差和第二POC差相同时,才可以推导指示第一参考图片的refidxL0的值作为当前块的refidxL0的值,并且推导指示第二参考图片的refidxL1的值作为当前块的refidxL1的值。另外,例如,当存在第一POC差和第二POC差相同的多个集合时,可以将具有最小差的集合的refidxL0和refidxL1推导为当前块的refidxL0和refidxL1。

[0185] 参考图7,示出了参考图片列表0、参考图片列表1以及MVDL0和MVDL1。这里,MVDL1与MVDL0对称。

[0186] MVDL1可以推导为负(-)MVDL0。例如,用于当前块的最终(改进的或修改的)运动信息(运动矢量;MV)可以基于以下等式来推导。

[0187] [等式1]

$$[0188] \quad \begin{cases} (mvx_0, mvy_0) = (mvp_x_0 + mvd_x_0, mvp_y_0 + mvd_y_0) \\ (mvx_1, mvy_1) = (mvp_x_1 - mvd_x_0, mvp_y_1 - mvd_y_0) \end{cases}$$

[0189] 在等式1中,mvx₀和mvy₀可以表示用于L0预测的L0运动信息或运动矢量的x分量和y分量,并且mvx₁和mvy₁可以表示用于L1预测的L1运动信息或运动矢量的x分量和y分量。此外,mvp_{x0}和mvp_{y0}可以表示用于L0预测的运动矢量预测子的x分量和y分量,并且mvp_{x1}和mvp_{y1}可以表示用于L1预测的运动矢量预测子的x分量和y分量。此外,mvd_{x0}和mvd_{y0}可以表示用于L0预测的运动矢量差的x分量和y分量。

[0190] 同时,MMVD模式是将运动矢量差(MVD)应用于合并模式的方法,并且可以隐含地推导直接用于生成当前块(即,当前CU)的预测样本的运动信息。例如,可以用信号发送指示是否将MMVD用于当前块(即,当前CU)的MMVD标志(即,mmvd_flag),并且可以基于该MMVD标志来执行MMVD。当MMVD应用于当前块时(即,当mmvd_flag为1时),可以用信号发送关于MMVD的附加信息。

[0191] 这里,关于MMVD的附加信息包括指示合并候选列表中的第一候选或第二候选是否与MVD一起使用的合并候选标志(即,mmvd_cand_flag),以及用于指示运动幅度的距离索引(即,mmvd_distance_idx)和用于指示运动方向的方向索引(即,mmvd_direction_idx)。

[0192] 在MMVD模式中,可以使用位于合并候选列表中的候选当中的第一和第二条目中的两个候选(即,第一候选或第二候选),并且可以使用两个候选(即,第一候选或第二候选)。其中一个可以用作基本MV。例如,可以用信号发送合并候选标志(即,mmvd_cand_flag)以指示合并候选列表中的两个候选(即,第一候选或第二候选)中的任一个。

[0193] 此外,距离索引(即,mmvd_distance_idx)指定运动幅度信息并且指示距起始点的预定义偏移。偏移可以添加到起始MV的水平分量或垂直分量。下表指定了距离索引和预定义偏移的关系。

[0194] [表5]

mmvd_distance_idx[x0][y0]	MmvdDistance[x0][y0]	
	tile_group_fpel_mmvd_enabled_flag = 0	tile_group_fpel_mmvd_enabled_flag = 1
0	1	4
1	2	8
2	4	16
3	8	32
4	16	64
5	32	128
6	64	256
7	128	512

[0195] 参考上面的表5,根据距离索引(即mmvd_distance_idx)的值来确定MVD的距离(即MmvdDistance),并且可以通过使用基于tile_group_fpel_mmvd_enabled_flag的值的整数样本精度或分数样本精度来推导MVD的距离(即MmvdDistance)。例如,当tile_group_fpel_mmvd_enabled_flag等于1时,它指示通过使用当前拼块组(或图片报头)中的整数样本精度来推导MVD的距离,并且当tile_group_fpel_mmvd_enabled_flag等于0时,它指示通过使用拼块组(或图片报头)中的分数样本精度来推导MVD的距离。在表1中,拼块组的信息(标志)可以用图片报头的信息替换,例如,tile_group_fpel_mmvd_enabled_flag可以用ph_fpel_mmvd_enabled_flag(或ph_mmvd_fullpel_only_flag)替换。

[0197] 另外,方向索引(即,mmvd_direction_idx)指示MVD相对于起始点的方向,并且可以指示如下表6所示的四个方向。在这种情况下,MVD的方向可以指示MVD的符号。方向索引和MVD符号之间的关系可以表达为下表。

[0198] [表6]

	mmvd_direction_idx[x0][y0]	MmvdSign[x0][y0][0]	MmvdSign[x0][y0][1]
[0199]	0	+1	0
	1	-1	0
	2	0	+1
	3	0	-1

[0200] 参考表6, MVD的符号(即MmvdSign)根据方向索引(即mmvd_direction_idx)的值来确定,并且可以针对L0参考图片和L1参考图片推导MVD的符号(即MmvdSign)。

[0201] 基于上述距离索引(即mmvd_distance_idx)和方向索引(即,mmvd_direction_idx),可以使用以下等式来计算MVD的偏移。

[0202] [等式2]

[0203] $MmvdOffset[x0][y0][0] = (MmvdDistance[x0][y0] \ll 2) * MmvdSign[x0][y0][0]$

[0204] [等式3]

[0205] $MmvdOffset[x0][y0][1] = (MmvdDistance[x0][y0] \ll 2) * MmvdSign[x0][y0][1]$

[0206] 在等式2和3中,可以基于表5和/或表6推导MMVD距离(MmvdDistance[x0][y0])和MMVD符号(MmvdSign[x0][y0][0], MmvdSign[x0][y0][1])。总之,在MMVD模式中,从基于邻近块推导的合并候选列表中的合并候选中选择由合并候选标志(例如,mmvd_cand_flag)指示的合并候选,并且将选定合并候选用作基本候选(即MVP)。另外,可以通过基于基本候选将使用距离索引(例如,mmvd_distance_idx)和方向索引(即,mmvd_direction_idx)推导的MVD相加来推导当前块的运动信息(即,运动矢量)。

[0207] 可以基于根据预测模式推导的运动信息来推导当前块的预测块。预测块可以包括当前块的预测样本(预测样本阵列)。在当前块的运动矢量指示分数样本单元时,可以执行插值过程,通过该插值过程可以基于参考图片内的以分数样本为单位的参考样本来推导当前块的预测样本。当应用双预测时,通过基于L0预测(即,使用参考图片列表L0中的参考图片和MVL0的预测)推导的预测样本和基于L1预测(即,使用参考图片列表L1中的参考图片和MVL1的预测)推导的预测样本的加权或加权平均(根据相位)推导的预测样本可以用作当前块的预测样本。当应用双预测时,如果用于L0预测的参考图片和用于L1预测的参考图片相对于当前图片位于不同时间方向(即,双预测和双向预测),则可以将其称为真双预测。

[0208] 如上所述,可以基于推导的预测样本生成重构样本和重构图片,然后可以执行诸如环内滤波的过程。

[0209] 如上所述,根据本文档,当双预测应用于当前块时,可以基于加权平均来推导预测样本。通常,可以通过L0预测信号(L0预测样本)和L1预测信号(L1预测样本)的简单平均来推导双预测信号(即,双预测样本)。即,双预测样本被推导作为基于L0参考图片和MVL0的L0预测样本和基于L1参考图片和MVL1的L1预测样本的平均值。然而,根据该文档,当应用双预测时,可以如下通过L0预测信号和L1预测信号的加权平均来推导双预测信号(双预测样本)。

[0210] 在上述MMVD相关实施例中,可以提出在MMVD的MVD推导过程中考虑长期参考图片的方法,从而在各种应用中维持和增加压缩效率。此外,除了在MERGE中使用的MMVD技术之外,本文档的实施例中所提出的方法可以同样地应用于SMVD, SMVD是在帧间模式(MVP模式)

中使用的对称MVD技术。

[0211] 图8是用于描述在帧间预测中推导运动矢量的方法的图。

[0212] 在本文档的实施例中,在时间运动候选(时间运动候选、时间合并候选或时间mvp候选)的运动矢量缩放过程中使用考虑长期参考图片的MV推导方法。时间运动候选可以对应于mvCo1 (mvLXCo1)。时间运动候选可以称为TMVP。

[0213] 下表描述了长期参考图片的定义。

[0214] [表7]

[0215]

函数 LongTermRefPic(aPic, aPb, refIdx, LX) (其中, X 为 0 或 1) 如下被定义:

-如果来自图片 aPic 中的包含预测块 aPb 的切片的参考图片列表 LX 的具有索引 refIdx 的图片在 aPic 是当前图片时被标记为“用于长期参考”, 则 LongTermRefPic(aPic, aPb, refIdx, LX) 等于 1。

-否则, LongTermRefPic(aPic, aPb, refIdx, LX) 等于 0。

[0216] 参考上述图7,如果LongTermRefPic(aPic, aPb, refIdx, LX) 等于1(真),则可以将相应参考图片标记为用于长期参考。例如,未标记为用于长期参考的参考图片可以是标记为用于短期参考的参考图片。在另一示例中,未标记为用于长期参考且未标记为不使用的参考图片可以是标记为用于短期参考的参考图片。在下文中,标记为用于长期参考的参考图片可以称为长期参考图片,标记为用于短期参考的参考图片可以称为短期参考图片。

[0217] 下表描述了TMVP (mvLXCo1)的推导。

[0218] [表8]

[0219]

当 availableFlagLXCol 等于真时, mvLXCol 和 availableFlagLXCol 如下被推导:

- 如果 LongTermRefPic(currPic, currCb, refIdxLX, LX) 不等于 LongTermRefPic(ColPic, colCb, refIdxCol, listCol), mvLXCol 的两个分量都被设置为 0 并且 availableFlagLXCol 被设置为等于 0。

-否则, 变量 availableFlagLXCol 被设置为等于 1, refPicList[listCol][refIdxCol]被设置为包含由 ColPic 指定的并置图片中的编译块 colCb 的切片的参考图片列表 listCol 中的具有参考索引 refIdxCol 的图片, 并且以下应用:

$$\text{colPocDiff} = \text{DiffPicOrderCnt}(\text{ColPic}, \text{refPicList}[\text{listCol}][\text{refIdxCol}]) \quad (8-402)$$

$$\text{currPocDiff} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[X][\text{refIdxLX}]) \quad (8-403)$$

-如在条款 8.5.2.15 中指定的并置运动矢量的时间运动缓冲压缩过程用 mvCol 作为输入被调用并且修改后的 myCol 作为输出。

-如果 RefPicList[X][refIdxLX]是长期参考图片或者 colPocDiff 等于 currPocDiff, 则 mvLXCol 如下被推导:

$$\text{mvLXCol} = \text{mvCol} \quad (8-404)$$

-否则, mvLXCol 如下被推导为运动矢量 mvCol 的缩放版本:

$$\text{tx} = (16384 + (\text{Abs}(\text{td}) \gg 1)) / \text{td} \quad (8-405)$$

$$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (\text{tb} * \text{tx} + 32) \gg 6) \quad (8-406)$$

$$\text{mvLXCol} = \text{Clip3}(-131072, 131071, (\text{distScaleFactor} * \text{mvCol} + 128 - (\text{distScaleFactor} * \text{mvCol} \geq 0)) \gg 8) \quad (8-407)$$

其中, td 和 tb 如下被推导:

$$\text{td} = \text{Clip3}(-128, 127, \text{colPocDiff}) \quad (8-408)$$

$$\text{tb} = \text{Clip3}(-128, 127, \text{currPocDiff}) \quad (8-409)$$

[0220] 参考图8和表8,在当前图片所指向的参考图像的类型(例如,指示是长期参考图片(LTRP)还是短期参考图片(STRP))不等于并置图片所指向的并置参考图片的类型时,不使用时间运动矢量mvLXCol。也就是说,当所有这些都是长期参考图片或短期参考图片时,推导colMV,否则,不推导colMV。另外,在所有这些都是长期参考图片的情况下,并且在当前图片与当前图片的参考图片之间的POC差与并置图片与并置图片的参考图片之间的POC差相同的情况下,可以按原样使用并置运动矢量而不缩放。如果是短期参考图片并且POC差不同,则在缩放后使用并置块的运动矢量。

[0221] 在本文档的实施例中,在MERGE/SKIP(合并/跳过)模式中使用的MMVD用信号发送用于一个编译块的基本运动矢量索引(基本MV索引)、距离索引和方向索引作为用于推导MVD信息的信息。在单向预测的情况下,从运动信息推导MVD,并且在双向预测的情况下,使用镜像和缩放方法生成对称MVD信息。

[0222] 在双向预测的情况下,对于L0或L1的MVD信息进行缩放以生成L1或L0的MVD。然

而,当参考长期参考图片时,它需要在MVD推导过程中进行修改。

[0223] 图9至图13示出了根据本文档的实施例的MMVD的MVD推导方法。图9至图13所示的方法可以用于应用双向预测的块。

[0224] 在根据图9的一个实施例中,当到L0参考图片的距离和到L1参考图片的距离相同时,推导的MmvdOffset可以原样用作MVD,并且当POC差(L0参考图片和当前图片之间的POC差以及L1参考图片和当前图片之间的POC差)不同时,MVD可以通过缩放或简单镜像(即, $-1 * \text{MmvdOffset}$)根据POC差以及它是长期还是短期参考图片来推导。

[0225] 在一个示例中,一种针对应用了双预测的块,使用MMVD来推导对称MVD的方法不适用于使用长期参考图片的块。很难预期性能改善。因此,在下面的附图和实施例中,引入了当L0和L1的参考图片类型不同时不应用MMVD的示例。

[0226] 在根据图10的一个实施例中,根据由当前图片(或当前切片、当前块)参考的参考图片是长期参考图片(LTRP)还是短期参考图片(STRP),用于推导MVD的方法可以是不同的。在一个示例中,当应用根据图10的实施例的方法时,可以如下表中所示描述根据本实施例的标准文档的一部分。

[0227] [表9]

8.5.2.7 用于合并运动矢量差的推导过程

该过程的输入是：

-当前亮度编译块的左上样本相对于当前图片的左上亮度样本的亮度位置(xCb,yCb),

-参考索引 refIdxL0 和 refIdxL1,

-预测列表利用标志 preFlagL0 和 predFlagL1。

该过程的输出是 1/16 分数样本准确度的亮度合并运动矢量差 mMvdL0 和 mMvdL1。

变量 currPic 指定当前图片。

亮度合并运动矢量差 mMvdL0 和 mMvdL1 如下被推导：

-如果 preFlagL0 和 preFlagL1 二者都等于 1，以下应用：

currPocDiffL0 = DiffPicOrderCnt(currPic, RefPicList[0][refIdxL0])

currPocDiffL1 = DiffPicOrderCnt(currPic, RefPicList[1][refIdxL1])

-如果 currPocDiffL0 等于 currPocDiffL1,并且以下应用：

mMvdL0[0] = MmvdOffset[xCb][yCb][0]

mMvdL0[1] = MmvdOffset[xCb][yCb][1]

mMvdL1[0] = MmvdOffset[xCb][yCb][0]

mMvdL1[1] = MmvdOffset[xCb][yCb][1]

-否则，如果 Abs(currPocDiffL0)大于或等于 Abs(currPocDiffL1)，则以下应用：

td = Clip3(-128, 127, currPocDiffL0)

tb = Clip3(-128, 127, currPocDiffL1)

tx = (16384 + (Abs(td) >> 1)) / td

distScaleFactor = Clip3(-4096, 4095, (tb * tx + 32) >> 6)

mMvdL0[0] = MmvdOffset[xCb][yCb][0]

mMvdL0[1] = MmvdOffset[xCb][yCb][1]

[0228]

[0229]

- 如果 RefPicList[0][refIdxL0] 不是长期参考图片并且 RefPicList[1][refIdxL1]不是长期参考图片，以下应用：

$$mMvdL1[0] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL0[0] + 128 - (\text{distScaleFactor} * mMvdL0[0] \geq 0)) \gg 8)$$

$$mMvdL1[1] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL0[1] + 128 - (\text{distScaleFactor} * mMvdL0[1] \geq 0)) \gg 8)$$

- 否则，如果 RefPicList[0][refIdxL0] 是长期参考图片并且 RefPicList[1][refIdxL1]是长期参考图片，以下应用：

$$mMvdL1[0] = \text{Sign}(\text{currPocDiffL0}) == \text{Sign}(\text{currPocDiffL1}) ? mMvdL0[0] : -mMvdL0[0]$$

[0230]

$$mMvdL1[0] = \text{Sign}(\text{currPocDiffL0}) == \text{Sign}(\text{currPocDiffL1}) ? mMvdL0[0] : -mMvdL0[0]$$

-否则，以下应用

$$mMvdL0[0] = 0$$

$$mMvdL0[1] = 0$$

$$mMvdL1[0] = 0$$

$$mMvdL1[1] = 0$$

-否则 (Abs(currPocDiffL0)小于 Abs(currPocDiffL1))，以下应用：

$$td = \text{Clip3}(-128, 127, \text{currPocDiffL1})$$

$$tb = \text{Clip3}(-128, 127, \text{currPocDiffL0})$$

$$tx = (16384 + (\text{Abs}(td) \gg 1)) / td$$

$$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (tb * tx + 32) \gg 6)$$

$$mMvdL1[0] = \text{MmvdOffset}[xCb][yCb][0]$$

$$mMvdL1[1] = \text{MmvdOffset}[xCb][yCb][1]$$

- 如果 RefPicList[0][refIdxL0] 不是长期参考图片并且 RefPicList[1][refIdxL1]不是长期参考图片，以下应用：

[0231]

```

mMvdL0[ 0 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 0 ] +
128 - (distScaleFactor * mMvdL1[ 0 ] >= 0) ) >> 8 )

mMvdL0[ 1 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 1 ] +
128 - (distScaleFactor * mMvdL1[ 1 ] >= 0) ) >> 8 )

- 否则，如果 RefPicList[0][refIdxL0] 是长期参考图片并且
RefPicList[1][refIdxL1]是长期参考图片，以下应用：
mMvdL0[ 0 ] = Sign( currPocDiffL0 ) == Sign( currPocDiffL1 ) ?
                mMvdL1[ 0 ] : -mMvdL1[ 0 ]

mMvdL0[ 1 ] = Sign( currPocDiffL0 ) == Sign( currPocDiffL1 ) ?
                mMvdL1[ 1 ] : -mMvdL1[ 1 ]

-否则，以下应用：
mMvdL0[ 0 ] = 0

mMvdL0[ 1 ] = 0

mMvdL1[ 0 ] = 0

mMvdL1[ 1 ] = 0

-否则（preFlagL0 或 predFlagL1 等于 1），以下针对 X 等于 0 和 1 而应用：
mMvdLX[ 0 ] = ( predFlagLX == 1 ) ? MmvdOffset[ xCb ][ yCb ][ 0 ] : 0

mMvdLX[ 1 ] = ( predFlagLX == 1 ) ? MmvdOffset[ xCb ][ yCb ][ 1 ] : 0

```

[0232] 在根据图11的一个实施例中，根据由当前图片（或当前切片、当前块）参考的参考图片是长期参考图片（LTRP）还是短期参考图片（STRP），用于推导MVD的方法可以是不同的。在一个示例中，当应用根据图11的实施例的方法时，可以如下表中所示描述根据本实施例的标准文档的一部分。

[0233] [表10]

[0234]

8.5.2.7 用于合并运动矢量差的推导过程

该过程的输入是：

- 当前亮度编译块的左上样本相对于当前图片的左上亮度样本的亮度位置(xCb,yCb)，
- 参考索引 refIdxL0 和 refIdxL1，

-预测列表利用标志 preFlagL0 和 predFlagL1。

该过程的输出是 1/16 分数样本准确度的亮度合并运动矢量差 mMvdL0 和 mMvdL1。

变量 currPic 指定当前图片。

亮度合并运动矢量差 mMvdL0 和 mMvdL1 如下被推导：

-如果 preFlagL0 和 preFlagL1 二者都等于 1，以下应用：

$$\text{currPocDiffL0} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{refIdxL0}])$$

$$\text{currPocDiffL1} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{refIdxL1}])$$

- 如果 (RefPicList[0][refIdxL0] 是长期参考图片并且 RefPicList[1][refIdxL1]是短期参考图片) 或者 (RefPicList[0][refIdxL0]是短期参考图片并且 RefPicList[1][refIdxL1]是长期参考图片)，以下应用：

$$\text{mMvdL0}[0] = 0$$

$$\text{mMvdL0}[1] = 0$$

$$\text{mMvdL1}[0] = 0$$

$$\text{mMvdL1}[1] = 0$$

-否则，以下应用：

-如果 currPocDiffL0 等于 currPocDiffL1,以下应用：

$$\text{mMvdL0}[0] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][0]$$

$$\text{mMvdL0}[1] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][1]$$

$$\text{mMvdL1}[0] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][0]$$

$$\text{mMvdL1}[1] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][1]$$

-否则，如果 Abs(currPocDiffL0)大于或等于 Abs(currPocDiffL1)，则以下应用：

[0235]

[0236]

```

td = Clip3( -128, 127, currPocDiffL0 )
tb = Clip3( -128, 127, currPocDiffL1 )
tx = ( 16384 + ( Abs( td ) >> 1 ) ) / td
distScaleFactor = Clip3( -4096, 4095, ( tb * tx + 32 ) >> 6 )
mMvdL0[ 0 ] = MmvdOffset[ xCb ][ yCb ][ 0 ]
mMvdL0[ 1 ] = MmvdOffset[ xCb ][ yCb ][ 1 ]

```

如果 RefPicList[0][refIdxL0]不是长期参考图片，以下应用：

[0237]

```

mMvdL1[ 0 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL0[ 0 ] +
128 - ( distScaleFactor * mMvdL0[ 0 ] >= 0 ) ) >> 8 )

```

```

mMvdL1[ 1 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL0[ 1 ] +
128 - ( distScaleFactor * mMvdL0[ 1 ] >= 0 ) ) >> 8 )

```

否则，以下应用

```

mMvdL1[ 0 ] = Sign( currPocDiffL0 ) == Sign( currPocDiffL1 ) ?
                mMvdL0[ 0 ] : -mMvdL0[ 0 ]

```

```

mMvdL1[ 0 ] = Sign( currPocDiffL0 ) == Sign( currPocDiffL1 ) ?
                mMvdL0[ 0 ] : -mMvdL0[ 0 ]

```

-否则 (Abs(currPocDiffL0)小于 Abs(currPocDiffL1))，以下应用：

```

td = Clip3( -128, 127, currPocDiffL1 )

```

```

tb = Clip3( -128, 127, currPocDiffL0 )

```

```

tx = ( 16384 + ( Abs( td ) >> 1 ) ) / td

```

```

distScaleFactor = Clip3( -4096, 4095, ( tb * tx + 32 ) >> 6 )

```

```

mMvdL1[ 0 ] = MmvdOffset[ xCb ][ yCb ][ 0 ]

```

```

mMvdL1[ 1 ] = MmvdOffset[ xCb ][ yCb ][ 1 ]

```

如果 RefPicList[0][refIdxL0]不是长期参考图片，以下应用：

```

mMvdL0[ 0 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 0 ] +
128 - (distScaleFactor * mMvdL1[ 0 ] >= 0 ) ) >> 8 )

```

```

mMvdL0[ 1 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 1 ] +
128 - (distScaleFactor * mMvdL1[ 1 ] >= 0 ) ) >> 8 )

```

[0238]

否则，以下应用：

$$mMvdL10[0] = \text{Sign}(currPocDiffL0) == \text{Sign}(currPocDiffL1) ? mMvdL1[0] : -mMvdL1[0]$$

$$mMvdL0[1] = \text{Sign}(currPocDiffL0) == \text{Sign}(currPocDiffL1) ? mMvdL1[1] : -mMvdL1[1]$$

-否则（preFlagL0 或 predFlagL1 等于 1），针对 X 等于 0 和 1 以下应用：

$$mMvdLX[0] = (\text{predFlagLX} == 1) ? MmvdOffset[xCb][yCb][0] : 0$$

$$mMvdLX[1] = (\text{predFlagLX} == 1) ? MmvdOffset[xCb][yCb][1] : 0$$

[0239] 总之，已经描述了当每个方向上的参考图片类型不同时不推导MVD的MMVD的MVD推导过程。

[0240] 在根据图12的一个实施例中，在参考长期参考图片的所有情况下可能不推导MVD。也就是说，当至少一个L0和L1参考图片是长期参考图片时，MVD被设置为0，并且仅当包括短期参考图片时才能够推导MVD。

[0241] 在一个示例中，基于最高优先级条件 (RefPicL0!=LTRP && RefPicL1!=STRP)，在当前图片（或当前切片、当前块）仅参考短期参考图片时，可以推导用于MMVD的MVD。在一个示例中，当应用根据图12的实施例的方法时，可以如下表中所示描述根据本实施例的标准文档的一部分。

[0242] [表11]

[0243]

8.5.2.7 用于合并运动矢量差的推导过程

该过程的输入是：

-当前亮度编译块的左上样本相对于当前图片的左上亮度样本的亮度位

置(xCb,yCb)，

-参考索引 refIdxL0 和 refIdxL1，

-预测列表利用标志 preFlagL0 和 predFlagL1。

该过程的输出是 1/16 分数样本准确度的亮度合并运动矢量差 $mMvdL0$ 和 $mMvdL1$ 。

变量 $currPic$ 指定当前图片。

亮度合并运动矢量差 $mMvdL0$ 和 $mMvdL1$ 如下被推导：

-如果 $preFlagL0$ 和 $preFlagL1$ 二者都等于 1，以下应用：

$$currPocDiffL0 = DiffPicOrderCnt(currPic, RefPicList[0][refldxL0])$$

$$currPocDiffL1 = DiffPicOrderCnt(currPic, RefPicList[1][refldxL1])$$

- 如果 $RefPicList[0][refIdxL0]$ 不是短期参考图片或者 $RefPicList[1][refIdxL1]$ 不是短期参考图片，以下应用：

$$mMvdL0[0] = 0$$

$$mMvdL0[1] = 0$$

$$mMvdL1[0] = 0$$

$$mMvdL1[1] = 0$$

-否则，以下应用：

-如果 $currPocDiffL0$ 等于 $currPocDiffL1$ ，以下应用：

$$mMvdL0[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL1[1] = MmvdOffset[xCb][yCb][1]$$

-否则，如果 $Abs(currPocDiffL0)$ 大于或等于 $Abs(currPocDiffL1)$ ，则以下应用：

[0244]

[0245]

```

td = Clip3( -128, 127, currPocDiffL0 )
tb = Clip3( -128, 127, currPocDiffL1 )
tx = ( 16384 + ( Abs( td ) >> 1 ) ) / td
distScaleFactor = Clip3( -4096, 4095, ( tb * tx + 32 ) >> 6 )
mMvdL0[ 0 ] = MmvdOffset[ xCb ][ yCb ][ 0 ]
mMvdL0[ 1 ] = MmvdOffset[ xCb ][ yCb ][ 1 ]
mMvdL1[ 0 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL0[ 0 ] +
128 - ( distScaleFactor * mMvdL0[ 0 ] >= 0 ) ) >> 8 )

```

[0246]

```

mMvdL1[ 1 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL0[ 1 ] +
128 - ( distScaleFactor * mMvdL0[ 1 ] >= 0 ) ) >> 8 )
-否则 ( Abs(currPocDiffL0)小于 Abs(currPocDiffL1) ) , 以下应用:
td = Clip3( -128, 127, currPocDiffL1 )
tb = Clip3( -128, 127, currPocDiffL0 )
tx = ( 16384 + ( Abs( td ) >> 1 ) ) / td
distScaleFactor = Clip3( -4096, 4095, ( tb * tx + 32 ) >> 6 )
mMvdL1[ 0 ] = MmvdOffset[ xCb ][ yCb ][ 0 ]
mMvdL1[ 1 ] = MmvdOffset[ xCb ][ yCb ][ 1 ]
mMvdL0[ 0 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 0 ] +
128 - (distScaleFactor * mMvdL1[ 0 ] >= 0 ) ) >> 8 )
mMvdL0[ 1 ] = Clip3( -215, 215 - 1, (distScaleFactor * mMvdL1[ 1 ] +
128 - (distScaleFactor * mMvdL1[ 1 ] >= 0 ) ) >> 8 )
-否则 ( preFlagL0 或 predFlagL1 等于 1 ) , 针对 X 等于 0 和 1 以下应用:
如果 RefPicList[X][refIdxLX]不是短期参考图片, 以下应用:
mMvdLX[ 0 ] = 0
mMvdLX[ 1 ] = 0
-否则, 以下应用:
mMvdLX[ 0 ] = ( predFlagLX == 1 ) ? MmvdOffset[ xCb ][ yCb ][ 0 ] : 0
mMvdLX[ 1 ] = ( predFlagLX == 1 ) ? MmvdOffset[ xCb ][ yCb ][ 1 ] : 0

```

[0247] 在根据图13的实施例中,当每个方向上的参考图片类型不同时,当获得短期参考图片时推导MVD,并且当包括长期参考图片时推导MVD为0。

[0248] 在一个示例中,当每个方向上的参考图片类型不同时,当参考接近当前图片的参考图片(短期参考图片)时应用MmvdOffset,并且当参考远离当前图片的参考图片(长期参

考图片)时MVD具有值0。在这种情况下,可以将接近当前图片的图片视为具有短期参考图片,但是当接近图片是长期参考图片时,可以将mmvdOffset应用于指示短期参考图片的列表的运动矢量。

[0249] [表12]

[0250]

<code>mMvdL0 = 0</code>
<code>mMvdL1 = MmvdOffset</code>
<code>mMvdL0 = 0</code>
<code>mMvdL1 = (-1) * MmvdOffset</code>
<code>mMvdL0 = MmvdOffset</code>
<code>mMvdL1 = 0</code>
<code>mMvdL0 = (-1) * MmvdOffset</code>
<code>mMvdL1 = 0</code>

[0251] 例如,包括在表12中的四个段落可以顺序地替换包括在图13中的流程图的最下方的块(内容)。

[0252] 在一个示例中,当应用根据图13的实施例的方法时,可以如下表中所示描述根据本实施例的标准文档的一部分。

[0253] [表13]

[0254]

8.5.2.7 用于合并运动矢量差的推导过程
该过程的输入是:

-当前亮度编译块的左上样本相对于当前图片的左上亮度样本的亮度位置
(xCb,yCb),

-参考索引 refIdxL0 和 refIdxL1,

-预测列表利用标志 preFlagL0 和 predFlagL1。

该过程的输出是 1/16 分数样本准确度的亮度合并运动矢量差 mMvdL0 和
mMvdL1。

变量 currPic 指定当前图片。

亮度合并运动矢量差 mMvdL0 和 mMvdL1 如下被推导:

-如果 preFlagL0 和 preFlagL1 二者都等于 1, 以下应用:

$\text{currPocDiffL0} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{refIdxL0}])$

$\text{currPocDiffL1} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{refIdxL1}])$

-如果 currPocDiffL0 等于 currPocDiffL1, 并且以下应用:

$\text{mMvdL0}[0] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][0]$

$\text{mMvdL0}[1] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][1]$

$\text{mMvdL1}[0] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][0]$

$\text{mMvdL1}[1] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][1]$

-否则, 如果 $\text{Abs}(\text{currPocDiffL0})$ 大于或等于 $\text{Abs}(\text{currPocDiffL1})$, 则以下
应用:

$\text{td} = \text{Clip3}(-128, 127, \text{currPocDiffL0})$

$\text{tb} = \text{Clip3}(-128, 127, \text{currPocDiffL1})$

$\text{tx} = (\text{16384} + (\text{Abs}(\text{td}) \gg 1)) / \text{td}$

$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (\text{tb} * \text{tx} + 32) \gg 6)$

$\text{mMvdL0}[0] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][0]$

$\text{mMvdL0}[1] = \text{MmvdOffset}[\text{xCb}][\text{yCb}][1]$

- 如果 $\text{RefPicList}[0][\text{refIdxL0}]$ 不是长期参考图片并且
 $\text{RefPicList}[1][\text{refIdxL1}]$ 不是长期参考图片, 以下应用:

[0255]

[0256]

$$mMvdL1[0] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL0[0] + 128 - (\text{distScaleFactor} * mMvdL0[0] \geq 0)) \gg 8)$$

$$mMvdL1[1] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL0[1] + 128 - (\text{distScaleFactor} * mMvdL0[1] \geq 0)) \gg 8)$$

- 否则，如果 RefPicList[0][refIdxL0] 是长期参考图片并且 RefPicList[1][refIdxL1] 是长期参考图片，以下应用：

$$mMvdL1[0] = \text{Sign}(\text{currPocDiffL0}) == \text{Sign}(\text{currPocDiffL1}) ? mMvdL0[0] : -mMvdL0[0]$$

[0257]

$$mMvdL1[0] = \text{Sign}(\text{currPocDiffL0}) == \text{Sign}(\text{currPocDiffL1}) ? mMvdL0[0] : -mMvdL0[0]$$

-否则，以下应用：

$$mMvdL0[0] = 0$$

$$mMvdL0[1] = 0$$

$$mMvdL1[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$mMvdL1[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

-否则（Abs(currPocDiffL0)小于 Abs(currPocDiffL1)），以下应用：

$$td = \text{Clip3}(-128, 127, \text{currPocDiffL1})$$

$$tb = \text{Clip3}(-128, 127, \text{currPocDiffL0})$$

$$tx = (16384 + (\text{Abs}(td) \gg 1)) / td$$

$$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (tb * tx + 32) \gg 6)$$

$$mMvdL1[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$mMvdL1[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

-如果 RefPicList[0][refIdxL0]不是长期参考图片并且 RefPicList[1][refIdxL1]不是长期参考图片，以下应用：

$$mMvdL0[0] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL1[0] + 128 - (\text{distScaleFactor} * mMvdL1[0] \geq 0)) \gg 8)$$

$$mMvdL0[1] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * mMvdL1[1] + 128 - (\text{distScaleFactor} * mMvdL1[1] \geq 0)) \gg 8)$$

否则，如果 RefPicList[0][refIdxL0] 是长期参考图片并且

[0258]

RefPicList[1][refIdxL1]是长期参考图片，以下应用：

$$mMvdL0[0] = \text{Sign}(currPocDiffL0) == \text{Sign}(currPocDiffL1) ? mMvdL1[0] : -mMvdL1[0]$$

$$mMvdL0[1] = \text{Sign}(currPocDiffL0) == \text{Sign}(currPocDiffL1) ? mMvdL1[1] : -mMvdL1[1]$$

-否则，以下应用：

$$mMvdL0[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = 0$$

$$mMvdL1[1] = 0$$

-否则（preFlagL0 或 predFlagL1 等于 1），针对 X 等于 0 和 1 以下应用：

$$mMvdLX[0] = (\text{predFlagLX} == 1) ? MmvdOffset[xCb][yCb][0] : 0$$

$$mMvdLX[1] = (\text{predFlagLX} == 1) ? MmvdOffset[xCb][yCb][1] : 0$$

[0259] 下表示出了本文档中包括的示例之间的比较表。

[0260] [表14]

[0261]

			实施例 A		实施例 B		实施例 C		实施例 D	
L0	L1	POC	L0 偏移	L1 偏移	L0 偏移	L1 偏移	L0 偏移	L1 偏移	L0 偏移	L1 偏移
短	短	相同	偏移	偏移	偏移	偏移	偏移	偏移	偏移	偏移
		Diff (LO>=L1)	偏移	缩放	偏移	缩放	偏移	缩放	偏移	缩放
		Diff (LO<L1)	缩放	偏移	缩放	偏移	缩放	偏移	缩放	偏移
长	长	相同	偏移	偏移	偏移	偏移	0	0	偏移	偏移
		Diff (LO>=L1)	偏移	(-1)偏移	偏移	(-1)偏移	0	0	偏移	(-1)偏移
		Diff (LO<L1)	(-1)偏移	偏移	(-1)偏移	偏移	0	0	(-1)偏移	偏移
短	长	相同	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
		Diff (LO>=L1)	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
		Diff (LO<L1)	(-1)偏移	偏移	0	0	0	0	偏移	0
长	短	相同	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
		Diff (LO>=L1)	偏移	(-1)偏移	0	0	0	0	0	偏移
		Diff (LO<L1)	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

[0262] 参考表14,示出了在考虑根据图9至图13的实施例中描述的MMVD的MVD推导的参考图片类型应用偏移的方法之间的比较。在表14中,实施例A可以涉及现有MMVD,实施例B可以示出根据图9至图11的实施例,实施例C可以示出根据图12的实施例,以及实施例D可以示出根据图13的实施例。

[0263] 即,在根据图9、图10和图11的实施例中,已经描述了仅当两个方向的参考图片类型相同时才推导MVD的方法,并且在根据图12的实施例中,已经描述了仅当两个方向的参考图片类型相同时才推导MVD的方法。在根据图12的实施例的情况下,在用于单向预测的长期参考图片的情况下,可以将MVD设置为0。另外,在根据图13的实施例中,已经描述了当两个方向上的参考图片类型不同时仅在一个方向上推导MVD的方法。实施例之间的差异表示本文档中描述的技术的各种特征,并且本领域的普通技术人员可以理解,根据本文档的实施例将实现的效果可以基于这些特征来实现。

[0264] 在根据本文档的实施例中,当参考图片类型是长期参考图片时,执行单独处理。当包括长期参考图片时,基于POC差(POCDiff)的缩放或镜像不影响性能改进,因此在具有短期参考图片的方向上将MmvdOffset值分配给MVD,并且在具有长期参考图片的方向上将值0分配给MVD。在一个示例中,当应用该实施例时,可以如下表中所示描述符合本实施例的标准文档的一部分。

[0265] [表15]

[0266]

8.5.2.7 用于合并运动矢量差的推导过程

该过程的输入是:

- 当前亮度编译块的左上样本相对于当前图片的左上亮度样本的亮度位置(xCb,yCb),
- 参考索引 refIdxL0 和 refIdxL1,
- 预测列表利用标志 preFlagL0 和 predFlagL1。

该过程的输出是 1/16 分数样本准确度的亮度合并运动矢量差 mMvdL0 和 mMvdL1。

变量 currPic 指定当前图片。

亮度合并运动矢量差 mMvdL0 和 mMvdL1 如下被推导：

-如果 preFlagL0 和 preFlagL1 二者都等于 1，以下应用：

-如果 RefPicList[0][refIdxL0]是长期参考图片或 RefPicList[1][refIdxL1]是

长期参考图片，以下应用：

- 如果 RefPicList[0][refIdxL0]是短期参考图片，

$$\text{mMvdL0}[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$\text{mMvdL0}[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

$$\text{mMvdL1}[0] = 0$$

$$\text{mMvdL1}[1] = 0$$

-否则如果 RefPicList[1][refIdxL1]是短期参考图片，

$$\text{mMvdL0}[0] = 0$$

$$\text{mMvdL0}[1] = 0$$

$$\text{mMvdL1}[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$\text{mMvdL1}[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

-否则，

$$\text{mMvdL0}[0] = 0$$

$$\text{mMvdL0}[1] = 0$$

$$\text{mMvdL1}[0] = 0$$

$$\text{mMvdL1}[1] = 0$$

-否则，以下应用：

$$\text{currPocDiffL0} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{refIdxL0}])$$

$$\text{currPocDiffL1} = \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{refIdxL1}])$$

-如果 currPocDiffL0 等于 currPocDiffL1,以下应用：

$$\text{mMvdL0}[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$\text{mMvdL0}[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

$$\text{mMvdL1}[0] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][0]$$

$$\text{mMvdL1}[1] = \text{MmvdOffset}[x_{Cb}][y_{Cb}][1]$$

[0267]

[0268]

[0269]

-否则, 如果 $\text{Abs}(\text{currPocDiffL0})$ 大于或等于 $\text{Abs}(\text{currPocDiffL1})$, 以下应用:

$$\text{td} = \text{Clip3}(-128, 127, \text{currPocDiffL0})$$

$$\text{tb} = \text{Clip3}(-128, 127, \text{currPocDiffL1})$$

$$\text{tx} = (16384 + (\text{Abs}(\text{td}) \gg 1)) / \text{td}$$

$$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (\text{tb} * \text{tx} + 32) \gg 6)$$

$$\text{mMvdL0}[0] = \text{MmvdOffset}[\text{xCb}][[\text{yCb}][0]]$$

$$\text{mMvdL0}[1] = \text{MmvdOffset}[\text{xCb}][[\text{yCb}][1]]$$

$$\text{mMvdL1}[0] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * \text{mMvdL0}[0] + 128 - (\text{distScaleFactor} * \text{mMvdL0}[0] \geq 0)) \gg 8)$$

$$\text{mMvdL1}[1] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * \text{mMvdL0}[1] + 128 - (\text{distScaleFactor} * \text{mMvdL0}[1] \geq 0)) \gg 8)$$

-否则 ($\text{Abs}(\text{currPocDiffL0})$ 小于 $\text{Abs}(\text{currPocDiffL1})$), 以下应用:

$$\text{td} = \text{Clip3}(-128, 127, \text{currPocDiffL1})$$

$$\text{tb} = \text{Clip3}(-128, 127, \text{currPocDiffL0})$$

$$\text{tx} = (16384 + (\text{Abs}(\text{td}) \gg 1)) / \text{td}$$

$$\text{distScaleFactor} = \text{Clip3}(-4096, 4095, (\text{tb} * \text{tx} + 32) \gg 6)$$

$$\text{mMvdL1}[0] = \text{MmvdOffset}[\text{xCb}][[\text{yCb}][0]]$$

$$\text{mMvdL1}[1] = \text{MmvdOffset}[\text{xCb}][[\text{yCb}][1]]$$

$$\text{mMvdL0}[0] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * \text{mMvdL1}[0] + 128 - (\text{distScaleFactor} * \text{mMvdL1}[0] \geq 0)) \gg 8)$$

$$\text{mMvdL0}[1] = \text{Clip3}(-2^{15}, 2^{15} - 1, (\text{distScaleFactor} * \text{mMvdL1}[1] + 128 - (\text{distScaleFactor} * \text{mMvdL1}[1] \geq 0)) \gg 8)$$

-否则 (preFlagL0 或 predFlagL1 等于 1), 针对 X 等于 0 和 1 以下应用:

-如果 $\text{RefPicList}[X][\text{refIdxLX}]$ 是长期参考图片

$$\text{mMvdLX}[0] = 0$$

$$\text{mMvdLX}[1] = 0$$

-否则,

$$\text{mMvdLX}[0] = (\text{predFlagLX} == 1) ? \text{MmvdOffset}[\text{xCb}][[\text{yCb}][0]] : 0$$

$$\text{mMvdLX}[1] = (\text{predFlagLX} == 1) ? \text{MmvdOffset}[\text{xCb}][[\text{yCb}][1]] : 0$$

[0270] 在另一个示例中, 表15的一部分可以用下表替换。参考表16, 可以基于除了POCDiff之外的参考图片类型来应用偏移。

[0271] [表16]

[0272]

-如果 predFlagL0 和 predFlagL1 二者都等于 1, 以下应用:

- 如果 RefPicList[0][refIdxL0] 是长期参考图片或 RefPicList[1][refIdxL1]是长期参考图片, 以下应用:

$$mMvdL0[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = -MmvdOffset[xCb][yCb][0]$$

$$mMvdL1[1] = -MmvdOffset[xCb][yCb][1]$$
- 否则, 如果 RefPicList[1][refIdxL1]是短期参考图片,

$$mMvdL0[0] = -MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = -MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL1[1] = MmvdOffset[xCb][yCb][1]$$
- 否则,

$$mMvdL0[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = -MmvdOffset[xCb][yCb][0]$$

$$mMvdL1[1] = -MmvdOffset[xCb][yCb][1]$$

[0273] 在另一个示例中,表15的一部分可以用下表替换。参考表17,可以始终将MmvdOffset设置为L0,并且将MmvdOffset设置为L1,而不考虑参考图片类型。

[0274] [表17]

[0275]

-如果 predFlagL0 和 predFlagL1 二者都等于 1, 以下应用:

- 如果 RefPicList[0][refIdxL0] 是长期参考图片或 RefPicList[1][refIdxL1]是长期参考图片, 以下应用:

$$mMvdL0[0] = MmvdOffset[xCb][yCb][0]$$

$$mMvdL0[1] = MmvdOffset[xCb][yCb][1]$$

$$mMvdL1[0] = -MmvdOffset[xCb][yCb][0]$$

$$mMvdL1[1] = -MmvdOffset[xCb][yCb][1]$$

[0276] 根据本文档的实施例,可以与在上述MERGE模式中使用的MMVD类似地执行帧间模式中的SMVD。在双向预测的情况下,从编码设备向解码设备用信号发送是否推导对称MVD,并且当相关标志(例如sym_mvd_flag)为真(或值为1)时,通过镜像第一方向MVD(例如

MVDL0) 来推导第二方向MVD (即MVDL1)。在这种情况下,可以不执行第一方向MVD的缩放。

[0277] 下表示出了根据本文档的实施例的编译单元的语法。

[0278] [表18]

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	描述符
if(sps_smvd_enabled_flag && inter_pred_idc[x0][y0] == PRED_BI && !inter_affine_flag[x0][y0] && RefIdxSymL0 > -1 && RefIdxSymL1 > -1)	
sym_mvd_flag [x0][y0]	ae(v)
if(inter_pred_idc[x0][y0] != PRED_L1) {	
if(NumRefIdxActive[0] > 1 && !sym_mvd_flag[x0][y0])	
ref_idx_l0 [x0][y0]	ae(v)
mvd_coding(x0, y0, 0, 0)	
if(MotionModelIdx[x0][y0] > 0)	
mvd_coding(x0, y0, 0, 1)	
if(MotionModelIdx[x0][y0] > 1)	
mvd_coding(x0, y0, 0, 2)	
mvp_l0_flag [x0][y0]	ae(v)
} else {	
MvdL0[x0][y0][0] = 0	
MvdL0[x0][y0][1] = 0	
}	
if(inter_pred_idc[x0][y0] != PRED_L0) {	
if(NumRefIdxActive[1] > 1 && !sym_mvd_flag[x0][y0])	
ref_idx_l1 [x0][y0]	ae(v)
if(mvd_l1_zero_flag && inter_pred_idc[x0][y0] == PRED_BI) {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
MvdCpL1[x0][y0][0][0] = 0	
MvdCpL1[x0][y0][0][1] = 0	
MvdCpL1[x0][y0][1][0] = 0	
MvdCpL1[x0][y0][1][1] = 0	
MvdCpL1[x0][y0][2][0] = 0	
MvdCpL1[x0][y0][2][1] = 0	
} else {	
if(sym_mvd_flag[x0][y0]) {	
MvdL1[x0][y0][0] = -MvdL0[x0][y0][0]	
MvdL1[x0][y0][1] = -MvdL0[x0][y0][1]	
} else	
mvd_coding(x0, y0, 1, 0)	
if(MotionModelIdx[x0][y0] > 0)	
mvd_coding(x0, y0, 1, 1)	
if(MotionModelIdx[x0][y0] > 1)	
mvd_coding(x0, y0, 1, 2)	
mvp_l1_flag [x0][y0]	ae(v)
}	
} else {	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
}	
...	
}	

[0279]

[0280] [表19]

[0281]	$\text{if}(\text{sps_smvd_enabled_flag} \ \&\& \ \text{inter_pred_idc}[\text{x0}][\text{y0}] == \text{PRED_BI} \ \&\& \ \text{!inter_affine_flag}[\text{x0}][\text{y0}] \ \&\& \ \text{RefIdxSymL0} > -1 \ \&\& \ \text{RefIdxSymL1} > -1)$	
	$\text{sym_mvd_flag}[\text{x0}][\text{y0}]$	ac(v)

[0282] 参照表18和19,当inter_pred_idc == PRED_BI并且L0和L1的参考图片可用(例如,RefIdxSymL0>-1 && RefIdxSymL1>-1)时,用信号发送sym_mvd_flag。

[0283] 下表示出根据示例的MMVD参考索引的解码过程。

[0284] [表20]

8.3.5 用于对称运动矢量差参考索引的解码过程

该过程的输出是指定用于对称运动矢量差的列表 0 和列表 1 参考图片索引的 RefIdxSymL0 和 RefIdxSymL1, 即针对编译单元, 当 sym_mvd_flag 等于 1 时。

变量 RefIdxSymLX (其中 X 为 0 和 1) 如下被推导:

- 变量 currPic 指定当前图片。
- RefIdxSymL0 被设置为-1。
- 针对每个索引 i, 其中, i=0... NumRefIdxActive[0]-1,以下应用:
 - 当所有以下条件为真时, RefIdxSymL0 被设置为 i:
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) > 0,
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) < DiffPicOrderCnt(currPic, RefPicList[0][RefIdxSymL0])

或者

RefIdxSymL0 等于-1。

- RefIdxSymL1 被设置为等于-1。
- 针对每个索引 i, 其中, i=0... NumRefIdxActive[1]-1,以下应用:
 - 当所有以下条件为真时, RefIdxSymL1 被设置为 i:
 - DiffPicOrderCnt(currPic, RefPicList[1][i]) < 0,
 - DiffPicOrderCnt(currPic, RefPicList[1][i]) >

[0285]

[0286]

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{RefIdxSymL1}])$ 或者

RefIdxSymL1 等于-1。

-当 RefIdxSymL0 等于-1 或者 RefIdxSymL1 等于-1, 以下应用:

-针对每个索引 i , 其中, $i=0\dots\text{NumRefIdxActive}[0]-1$, 以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i :

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < 0$,
- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) >$

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{RefIdxSymL0}])$ 或者

RefIdxSymL0 等于-1。

针对每个索引 i , 其中, $i=0\dots\text{NumRefIdxActive}[1]-1$, 以下应用:

-当所有以下条件为真时, RefIdxSymL1 被设置为 i :

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][i]) > 0$,
- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][i]) <$

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{RefIdxSymL1}])$ 或者

RefIdxSymL1 等于-1。

[0287] 参考表20,描述了用于推导L0和L1的参考图片的可用性的过程。也就是说,如果在L0参考图片中存在沿前向的参考图片,则将最接近当前图片的参考图片的索引设置为 RefIdxSymL0 ,并且将对应的值设置为L0的参考索引。此外,当在L1参考图片中存在沿后向的参考图片时,将最接近当前图片的参考图片的索引设置为 RefIdxSymL1 ,并且将对应的值设置为L1的参考索引。

[0288] 下面的表21示出了根据另一示例的MMVD参考索引的解码过程。

[0289] [表21]

[0290]

8.3.5 用于对称运动矢量差参考索引的解码过程

该过程的输出是指定用于对称运动矢量差的列表 0 和列表 1 参考图片索引的 RefIdxSymL0 和 RefIdxSymL1 , 即针对编译单元, 当 sym_mvd_flag 等于

1 时。

变量 RefIdxSymLX (其中 X 为 0 和 1) 如下被推导:

-变量 currPic 指定当前图片。

-RefIdxSymL0 被设置为-1。

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[0]-1$, 以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) > 0$,

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) <$

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{RefIdxSymL0}])$

或 者

RefIdxSymL0 等于-1。

-RefIdxSymL1 被设置为等于-1。

[0291]

针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[1]-1$, 以下应用:

-当所有以下条件为真时, RefIdxSymL1 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < 0$,

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) >$

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{RefIdxSymL1}])$

或 者

RefIdxSymL1 等于-1。

-当 RefIdxSymL0 等于-1 或者 RefIdxSymL1 等于-1, 以下应用:

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[0]-1$, 以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < 0$,

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) >$

$\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{RefIdxSymL0}])$

或 者

RefIdxSymL0 等于-1。

针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[1]-1$, 以下应用:

[0292]

-当所有以下条件为真时，RefIdxSymL1 被设置为 i:

- DiffPicOrderCnt(currPic, RefPicList[1][i]) > 0,
- DiffPicOrderCnt(currPic, RefPicList[1][i]) < DiffPicOrderCnt(currPic, RefPicList[1][RefIdxSymL1]) 或者

RefIdxSymL1 等于-1。

-当 RefIdxSymL0 不等于-1 并且 RefIdxSymL1 不等于-1，以下应用:

- 如果 (RefPicList[0][refIdxL0] 是长期参考图片 并且 RefPicList[1][refIdxL1]是短期参考图片) 或者 (RefPicList[0][refIdxL0]是短期参考图片 并且 RefPicList[1][refIdxL1]是长期参考图片) ，则 RefIdxSymL0 和 RefIdxSymL1 被设置为-1。

[0293] 参考表21,如在图9、图10和图11所描述的实施例中,当L0或L1参考图片的类型不同时,即如果L0和L1的参考图片类型在用于SMVD的参考索引推导之后不同,则在使用长期参考图片和短期参考图片的情况下不使用SMVD以便阻止SMVD(参见表20的最下段)。

[0294] 在本文档的实施例中,可以以类似于合并模式中使用的MMVD的中间模式应用SMVD。当如在图12中描述的实施例中那样使用长期参考图片时,可以从用于SMVD的参考索引推导过程中排除长期参考图片,如下表所示,以便阻止SMVD。

[0295] [表22]

[0296]

8.3.5 用于对称运动矢量差参考索引的解码过程

该过程的输出是指定用于对称运动矢量差的列表 0 和列表 1 参考图片索引的 RefIdxSymL0 和 RefIdxSymL1, 即针对编译单元, 当 sym_mvd_flag 等于 1 时。

变量 RefIdxSymLX (其中 X 为 0 和 1) 如下被推导:

-变量 currPic 指定当前图片。

[0297]

-RefIdxSymL0 被设置为-1。

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[0]-1$,以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) > 0,$
- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{RefIdxSymL0}])$

或 者

RefIdxSymL0 等于-1,

-RefIdxSymL[0][i]是短期参考图片。

-RefIdxSymL1 被设置为等于-1。

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[1]-1$,以下应用:

-当所有以下条件为真时, RefIdxSymL1 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < 0,$
- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) > \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[1][\text{RefIdxSymL1}])$

或 者

RefIdxSymL1 等于-1,

-RefIdxSymL[1][i]是短期参考图片。

-当 RefIdxSymL0 等于-1 或者 RefIdxSymL1 等于-1, 以下应用:

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[0]-1$,以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i:

- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) < 0,$
- $\text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][i]) > \text{DiffPicOrderCnt}(\text{currPic}, \text{RefPicList}[0][\text{RefIdxSymL0}])$

或 者

RefIdxSymL0 等于-1,

-RefIdxSymL[0][i]是短期参考图片。

-针对每个索引 i, 其中, $i=0\dots\text{NumRefIdxActive}[1]-1$,以下应用:

-当所有以下条件为真时, RefIdxSymL1 被设置为 i:

[0298]

- DiffPicOrderCnt(currPic, RefPicList[1][i]) > 0,
 - DiffPicOrderCnt(currPic, RefPicList[1][i]) <
 DiffPicOrderCnt(currPic, RefPicList[1][RefIdxSymL1]) 或 者
 RefIdxSymL1 等于-1,
 -RefPicList[1][i]是短期参考图片。

[0299] 根据本实施例的另一示例的下表示出了当在用于SMVD的参考图片索引推导之后使用长期参考图片时不应用SMVD的处理示例。

[0300] [表23]

[0301]

8.3.5 用于对称运动矢量差参考索引的解码过程

该过程的输出是指定用于对称运动矢量差的列表 0 和列表 1 参考图片索引的 RefIdxSymL0 和 RefIdxSymL1, 即针对编译单元, 当 sym_mv_d_flag 等于 1 时。

变量 RefIdxSymLX (其中 X 为 0 和 1) 如下被推导:

-变量 currPic 指定当前图片。

-RefIdxSymL0 被设置为-1。

-针对每个索引 i, 其中, i=0... NumRefIdxActive[0]-1, 以下应用:

-当所有以下条件为真时, RefIdxSymL0 被设置为 i:

- DiffPicOrderCnt(currPic, RefPicList[0][i]) > 0,
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) <
 DiffPicOrderCnt(currPic, RefPicList[0][RefIdxSymL0]) 或 者

RefIdxSymL0 等于-1。

-RefIdxSymL1 被设置为等于-1。

-针对每个索引 i, 其中, i=0... NumRefIdxActive[1]-1, 以下应用:

-当所有以下条件为真时, RefIdxSymL1 被设置为 i:

[0302]

- DiffPicOrderCnt(currPic, RefPicList[0][i]) < 0,
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) >
 DiffPicOrderCnt(currPic, RefPicList[1][RefIdxSymL1]) 或 者

RefIdxSymL1 等于-1。

-当 RefIdxSymL0 等于-1 或者 RefIdxSymL1 等于-1, 以下应用:
 -针对每个索引 i, 其中, i=0... NumRefIdxActive[0]-1,以下应用:
 -当所有以下条件为真时, RefIdxSymL0 被设置为 i:
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) < 0,
 - DiffPicOrderCnt(currPic, RefPicList[0][i]) >
 DiffPicOrderCnt(currPic, RefPicList[0][RefIdxSymL0]) 或 者

RefIdxSymL0 等于-1。

-针对每个索引 i, 其中, i=0... NumRefIdxActive[1]-1,以下应用:
 -当所有以下条件为真时, RefIdxSymL1 被设置为 i:
 - DiffPicOrderCnt(currPic, RefPicList[1][i]) > 0,
 - DiffPicOrderCnt(currPic, RefPicList[1][i]) <
 DiffPicOrderCnt(currPic, RefPicList[1][RefIdxSymL1]) 或 者

RefIdxSymL1 等于-1。

-当 RefIdxSymL0 不等于-1 并且 RefIdxSymL1 不等于-1, 以下应用:
 - 如果 (RefPicList[0][refIdxL0] 是长期参考图片 或者
 RefPicList[1][refIdxL1]是长期参考图片), 则 RefIdxSymL0 和 RefIdxSymL1 被
 设置为-1。

[0303] 在本文档的一个实施例中,在TMVP的colMV推导过程中,在当前图片的参考图片类型和并置图片的参考图片类型不同时,将运动矢量MV设置为0,但MMVD和SMVD的情况下的推导方法与TMVP不同,并且它们需要被统一。

[0304] 即使在当前图片的参考图片类型是长期参考图片并且并置图片的参考图片类型是长期参考图片时,运动矢量也按原样使用并置运动矢量的值,但是在MMVD和SMVD中可以将MV设置为0。在这种情况下,TMVP也将MV设置为0,而没有附加诱导。

[0305] 此外,即使参考图片类型不同,也可以存在与当前图片具有近距离的长期参考图片。因此,代替将MV设置为0,colMV可以用作MV而不缩放。

[0306] 创建以下附图以解释本说明书的具体示例。由于在附图中描述的特定装置的名称或特定信号/消息/字段的名称通过示例的方式呈现,因此本说明书的技术特征不限于以下附图中使用的特定名称。

[0307] 图14和图15示出了根据本公开的实施例的视频/图像编码方法和相关组件的一个示例。图2的编码设备可以执行图14的方法。具体地,例如,编码设备的预测器220可以执行图14的S1400至图S1450的步骤,并且编码设备的残差处理器230可以执行S1460步骤。编码设备的熵编码器240可以执行S1470步骤。图14的方法可以包括上述本公开的实施例。

[0308] 参照图14,编码设备推导当前图片内的当前块的帧间预测模式S1400。这里,帧间预测模式可以包括合并模式、AMVP模式(使用运动矢量预测子候选的模式)、MMVD和SMVD。

[0309] 编码设备可以推导用于帧间预测模式的参考图片。编码设备可以配置用于推导参考图片的参考图片列表。在一个示例中,参考图片列表可以包括参考图片列表0(或L0,参考图片列表L0)或参考图片列表1(或L1,参考图片列表L1)。例如,编码设备可以为当前图片中包括的每个切片配置参考图片列表。

[0310] 编码设备基于当前块的邻近块来构造当前块的MVP候选列表S1410。在一个示例中,包括当前块的当前图片可以包括邻近块。在另一示例中,当前图片的先前(参考)图片或下一(参考)图片可以包括邻近块。这里,先前图片的POC可以小于当前图片的POC,后续图片的POC可以大于当前图片的POC。根据一个示例,当前图片与当前图片的先前(参考)图片之间的POC差可以大于0。在另一示例中,当前图片和当前图片的下一(参考)图片之间的POC差可以小于0。但是,上面给出的描述仅表示一个示例。

[0311] 编码设备可以基于MVP候选列表来推导用于当前块的MVP S1420。编码设备可以在包括在MVP候选列表中的运动矢量预测子候选当中推导最佳运动矢量预测子候选。编码设备可以生成指示最佳运动矢量预测子候选的选择信息(例如,MVP标志或MVP索引)。

[0312] 编码设备生成包括帧间预测模式的预测相关信息S1430。在一个示例中,预测相关信息可以包括关于当前块的运动矢量差(MVD)的信息。另外,预测相关信息可以包括关于MMVD的信息和关于SMVD的信息。

[0313] 编码设备基于帧间预测模式来推导用于当前块的预测的运动信息S1440。例如,运动信息可以包括用于SMVD的参考索引。SMVD的参考索引可以指示SMVD应用的参考图片。SMVD的参考索引可以包括参考索引L0(RefIdxSumL0)和参考索引L1(RefIdxSumL1)。

[0314] 编码设备基于运动信息来生成预测样本S1450。编码设备可以基于包括在运动信息中的运动矢量和参考图片索引来生成预测样本。例如,可以基于由参考图片索引指示的参考图片内的块(或样本)中的由运动矢量指示的块(或样本)来生成预测样本。

[0315] 编码设备基于预测样本来推导残差信息S1460。具体地,编码设备可以基于预测样本和原始样本来推导残差样本。编码设备可以基于残差样本来推导残差信息。可以执行上述变换和量化处理以推导残差信息。

[0316] 编码设备对包括预测相关信息和残差信息的图像/视频信息进行编码S1470。编码后的图像/视频信息可以以比特流的形式输出。比特流可以通过网络或(数字)存储介质发送到解码设备。

[0317] 根据本公开的实施例,图像/视频信息可以包括各种类型的信息。例如,图像/视频信息可以包括在上述表1至23中的至少一个中公开的信息。

[0318] 在一个实施例中,运动信息可以包括运动矢量和参考图片索引。可以基于关于MVD的信息和MVP来推导运动矢量。参考图片索引可以从包括在参考图片列表中的参考图片当中的短期参考图片来推导。

[0319] 在一个实施例中,可以基于MVD和MVP的和来推导当前块的运动矢量。

[0320] 在一个实施例中,预测相关信息可以包括关于SMVD的信息。例如,当基于短期参考图片和包括当前块的当前图片之间的POC差来推导参考图片索引时,关于SMVD的信息的值可以是1。

[0321] 在一个实施例中,可以推导用于L0预测的MVD L0。例如,当从MVD L0推导用于L1预测的MVD L1并且基于MVD L0和MVD L1推导运动信息时,关于SMVD的信息的值可以是1。

[0322] 在一个实施例中,MVD L1的大小可以与MVD L0的大小相同。MVD L1的符号可以与MVD L0的符号相反。

[0323] 在一个实施例中,参考图片索引可以指向短期参考图片。

[0324] 在一个实施例中,参考图片列表0可以包括短期参考图片。可以基于参考图片列表0中包括的每个参考图片与包括当前块的当前图片之间的POC差来推导参考图片索引。

[0325] 在一个实施例中,可以基于POC差之间的比较来推导参考图片索引。

[0326] 图16和17示出根据本公开的实施例的视频/图像解码方法和相关组件的一个示例。图3的解码设备可以执行图16的方法。具体地,例如,解码设备的熵解码器310可以执行图16的S1600步骤,并且解码设备的预测器330可以执行S1610至S1650步骤。图16的方法可以包括上述本公开的实施例。

[0327] 参照图16,解码设备接收/获得图像/视频信息S1600。解码设备可以通过比特流接收/获得图像/视频信息。图像/视频信息可以包括预测相关信息(包括预测模式信息)和残差信息。预测相关信息可以包括关于MMVD的信息和关于SMVD的信息。此外,根据本公开的实施例,图像/视频信息可以包括各种类型的信息。例如,图像/视频信息可以包括参照图1至图15描述的信息和/或上述表1至表23中的至少一个中公开的信息。

[0328] 解码设备基于预测相关信息来推导当前块的帧间预测模式S1610。这里,帧间预测模式可以包括合并模式、AMVP模式(使用运动矢量预测子候选的模式)、MMVD和SMVD。

[0329] 解码设备基于当前块的邻近块来构造当前块的MVP候选列表S1620。在一个示例中,包括当前块的当前图片可以包括邻近块。在另一示例中,当前图片的先前(参考)图片或下一(参考)图片可以包括邻近块。这里,先前图片的POC可以小于当前图片的POC,并且后续图片的POC可以大于当前图片的POC。根据一个示例,当前图片与当前图片的先前(参考)图片之间的POC差可以大于0。在另一示例中,当前图片和当前图片的下一(参考)图片之间的POC差可以小于0。但是,上面给出的描述仅表示一个示例。

[0330] 解码设备可以基于MVP候选列表来推导用于当前块的MVP S1630。解码设备可以在包括在MVP候选列表中的运动矢量预测子候选当中推导最佳运动矢量预测子候选。解码设备可以生成指示最佳运动矢量预测子候选的选择信息(例如,MVP标志或MVP索引)。

[0331] 解码设备基于关于MVD的信息和MVP来推导当前块的运动信息S1640。例如,运动信息可以包括用于SMVD的参考索引。SMVD的参考索引可以指示SMVD应用的参考图片。SMVD的参考索引可以包括参考索引L0(RefIdxSumL0)和参考索引L1(RefIdxSumL1)。

[0332] 解码设备基于运动信息来生成预测样本S1650。解码设备可以基于包括在运动信息中的运动矢量和参考图片索引来生成预测样本。例如,可以基于由参考图片索引指示的参考图片内的块(或样本)中的由运动矢量指示的块(或样本)来生成预测样本。

[0333] 解码设备可以基于残差信息来生成残差样本。具体地,解码设备可以基于残差信

息来推导量化变换系数。量化变换系数可以具有基于系数扫描顺序的一维矢量形式。解码设备可以基于用于量化变换系数的逆量化过程来推导变换系数。解码设备可以基于变换系数的逆变换过程来推导残差样本。

[0334] 解码设备可以基于预测样本和残差样本来生成当前图片的重构样本。解码设备还可以执行滤波过程以生成(修改的)重构样本。

[0335] 在一个实施例中,运动信息可以包括运动矢量和参考图片索引。可以基于关于MVD的信息和MVP来推导运动矢量。参考图片索引可以从包括在参考图片列表中的参考图片中的短期参考图片来推导。

[0336] 在一个实施例中,可以基于关于MVD的信息来推导MVD。可以基于MVD和MVP的和来推导当前块的运动矢量。

[0337] 在一个实施例中,预测相关信息可以包括关于SMVD的信息。例如,当关于SMVD的信息的值为1时,可以基于短期参考图片和包括当前块的当前图片之间的POC差来推导参考图片索引。

[0338] 在一个实施例中,可以基于关于MVD的信息来推导用于L0预测的MVD L0。例如,当关于SMVD的信息的值为1时,可以从MVD L0推导用于L1预测的MVD L1,并且可以基于MVD L0和MVD L1来推导运动信息。

[0339] 在一个实施例中,MVD L1的大小可以与MVD L0的大小相同。MVD L1的符号可以与MVD L0的符号相反。

[0340] 在一个实施例中,参考图片索引可以指向短期参考图片。

[0341] 在一个实施例中,参考图片列表0可以包括短期参考图片。可以基于参考图片列表0中包括的每个参考图片与包括当前块的当前图片之间的POC差来推导参考图片索引。

[0342] 在一个实施例中,可以基于POC差之间的比较来推导参考图片索引。

[0343] 在一个实施例中,参考图片索引(例如,ref_idx_l1[x0][y0])可以基于关于SMVD的信息来推导而不直接用信号发送。

[0344] 在上述实施例中,基于流程图作为一系列步骤或方框来描述方法,但是对应实施例不限于该步骤顺序,特定步骤可按照与上述顺序不同的顺序发生或者与不同于上述的步骤同时发生。另外,本领域技术人员将理解,流程图中所示的步骤不是排他性的,可以包括其它步骤或者可删除流程图中的一个或更多个步骤,而不影响本文档的实施例的范围。

[0345] 根据本文档的实施例的上述方法可按软件的形式实现,并且根据本文档的编码设备和/或解码设备可以包括在用于执行例如TV、计算机、智能电话、机顶盒、显示装置等的图像处理的设备中。

[0346] 当本文档中的实施例以软件实现时,上述方法可以被实现为用于执行上述功能的模块(进程、函数等)。该模块可以被存储在存储器中并由处理器执行。存储器可位于处理器内部或外部,并且可以通过各种熟知手段与处理器联接。处理器可以包括专用集成电路(ASIC)、其它芯片组、逻辑电路和/或数据处理装置。存储器可以包括只读存储器(ROM)、随机存取存储器(RAM)、闪存、存储卡、存储介质和/或其它存储装置。即,本文档中描述的实施例可在处理器、微处理器、控制器或芯片上实现和执行。例如,各个附图中所示的功能单元可在计算机、处理器、微处理器、控制器或芯片上实现和执行。在这种情况下,用于实现的信息(例如,关于指令的信息)或算法可以被存储在数字存储介质中。

[0347] 另外,应用了本文档的实施例的解码设备和编码设备可以被包括在多媒体广播发送和接收装置、移动通信终端、家庭影院视频装置、数字影院视频装置、监视相机、视频聊天装置、实时通信装置(例如,视频通信)、移动流装置、存储介质、摄像机、视频点播(VoD)服务提供器、过顶(OTT)视频装置、互联网流服务提供器、3D视频装置、虚拟现实装置、增强现实(AR)装置、视频电话视频装置、运输工具终端(例如,车辆(包括自主车辆)终端、飞机终端、船舶终端等)、医疗视频装置等中,并且可以用于处理视频信号或数据信号。例如,顶置视频(OTT)装置可以包括游戏机、蓝光播放器、互联网访问TV、家庭影院系统、智能电话、平板PC、数字视频记录仪(DVR)等。

[0348] 另外,应用了本文档的实施例的处理方法可按照由计算机执行的程序的形式生成,并且可以被存储在计算机可读记录介质中。具有根据本文档的实施例的数据结构的多媒体数据也可以被存储在计算机可读记录介质中。计算机可读记录介质包括存储计算机可读数据的所有类型的存储装置和分布式存储装置。例如,计算机可读记录介质可以包括蓝光光盘(BD)、通用串行总线(USB)、ROM、PROM、EPROM、EEPROM、RAM、CD-ROM、磁带、软盘和光学数据存储装置。另外,计算机可读记录介质还包括以载波的形式实现的介质(例如,经由互联网的传输)。另外,通过该编码方法生成的比特流可以被存储在计算机可读记录介质中或通过有线和无线通信网络发送。

[0349] 另外,本文档的实施例可以通过程序代码被实现为计算机程序产品,并且可以通过本文档的实施例在计算机上执行程序代码。程序代码可以被存储在可以由计算机读取的载体上。

[0350] 图18示出本文档中公开的实施例可以应用于的内容流系统的示例。

[0351] 参考图18,应用了本文档的实施例可以应用于的内容流系统可主要包括编码服务器、流服务器、web服务器、媒体存储装置、用户装置和多媒体输入装置。

[0352] 编码服务器起到通过将多媒体输入装置(例如,智能电话、相机或摄像机)输入的内容压缩为数字数据来生成比特流并将所生成的比特流发送到流服务器的作用。作为另一示例,如果多媒体输入装置(例如,智能电话、相机或摄像机)直接生成比特流,则可省略编码服务器。

[0353] 可以通过应用了本公开的编码方法或生成比特流的方法来生成比特流,并且在发送或接收比特流的同时,流服务器可暂时存储比特流。

[0354] 流服务器起到通过web服务器基于用户请求将多媒体数据发送到用户装置的作用,并且web服务器起到告知用户哪些服务可用的作用。如果用户向web服务器请求期望的服务,则web服务器将请求发送到流服务器,并且流服务器将多媒体数据发送到用户。此时,内容流系统可以包括单独的控制服务器,并且在这种情况下,控制服务器起到控制内容流系统内的装置之间的命令/响应的作用。

[0355] 流服务器可从媒体存储装置和/或编码服务器接收内容。例如,如果从编码服务器接收内容,则内容可实时地接收。在这种情况下,为了提供平滑的流服务,流服务器可以将比特流存储达预定时间段。

[0356] 用户装置的示例可以包括移动电话、智能电话、膝上型计算机、数字广播终端、个人数字助理(PDA)、便携式多媒体播放器(PMP)、导航终端、石板PC、平板PC、超级本、可穿戴装置(例如,智能手表或智能眼镜)、数字TV、台式计算机和数字标牌。

[0357] 内容流系统内的各个单独的服务器可作为分布式服务器操作,并且在这种情况下,可按分布式方式处理由各个服务器接收的数据。

[0358] 这里描述的权利要求可以以各种方式组合。例如,可以将本文档的方法权利要求的技术特征组合并实现为设备,并且可以将本文档的设备权利要求的技术特征组合并实现为方法。另外,本文档的方法权利要求的技术特征和设备权利要求的技术特征可以被组合以实现为设备,并且本文档的方法权利要求的技术特征和设备权利要求的技术特征可以被组合并实现为方法。

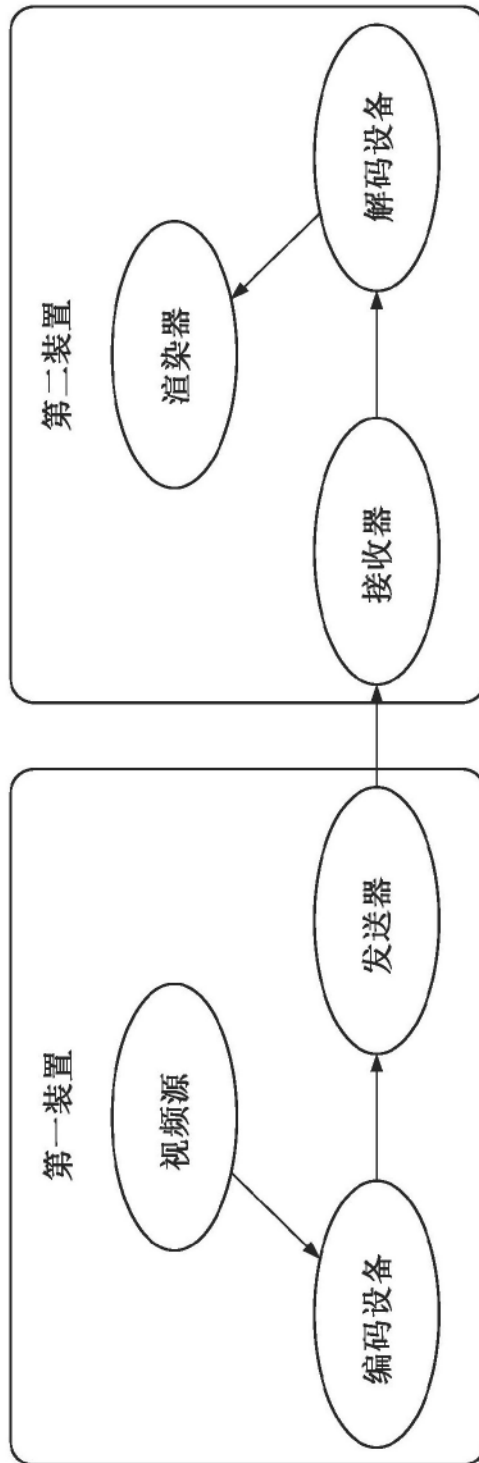


图1

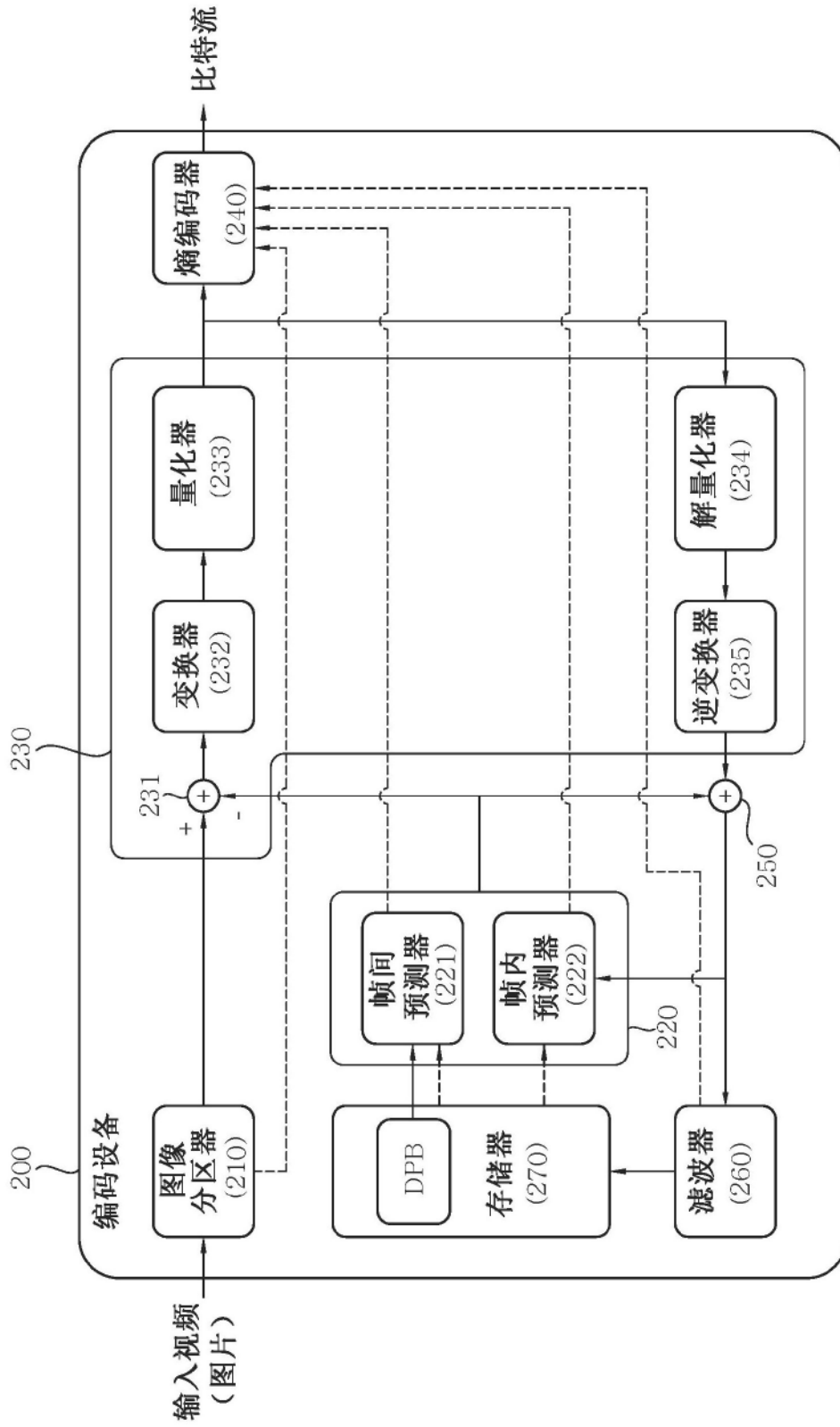


图2

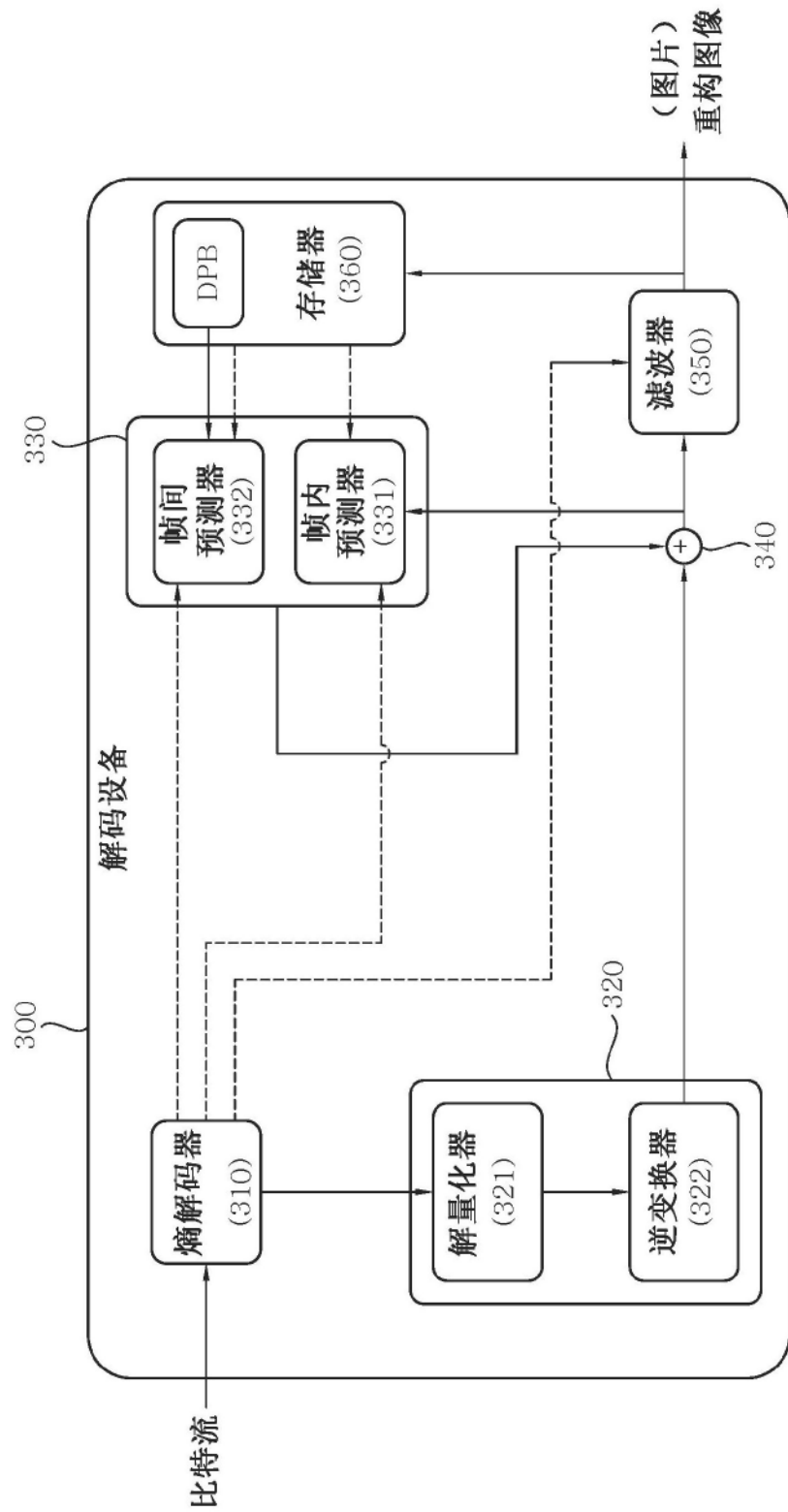


图3

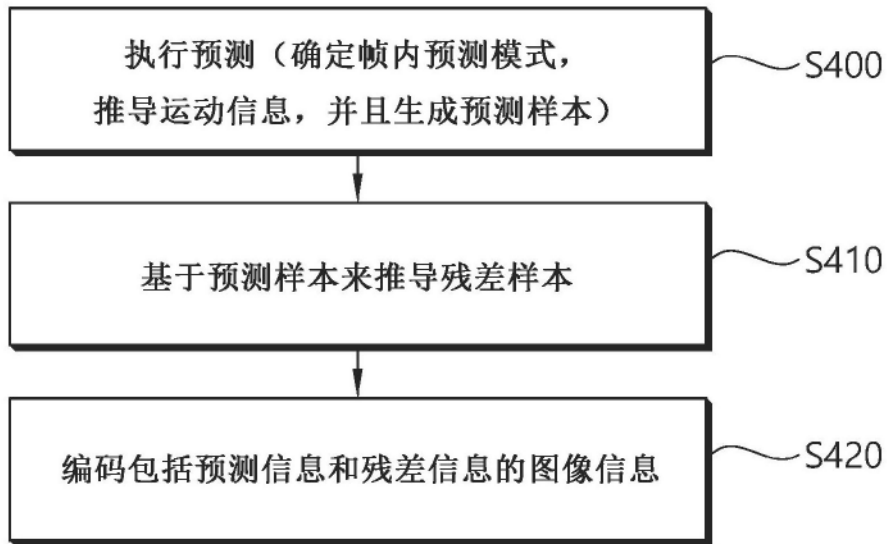


图4

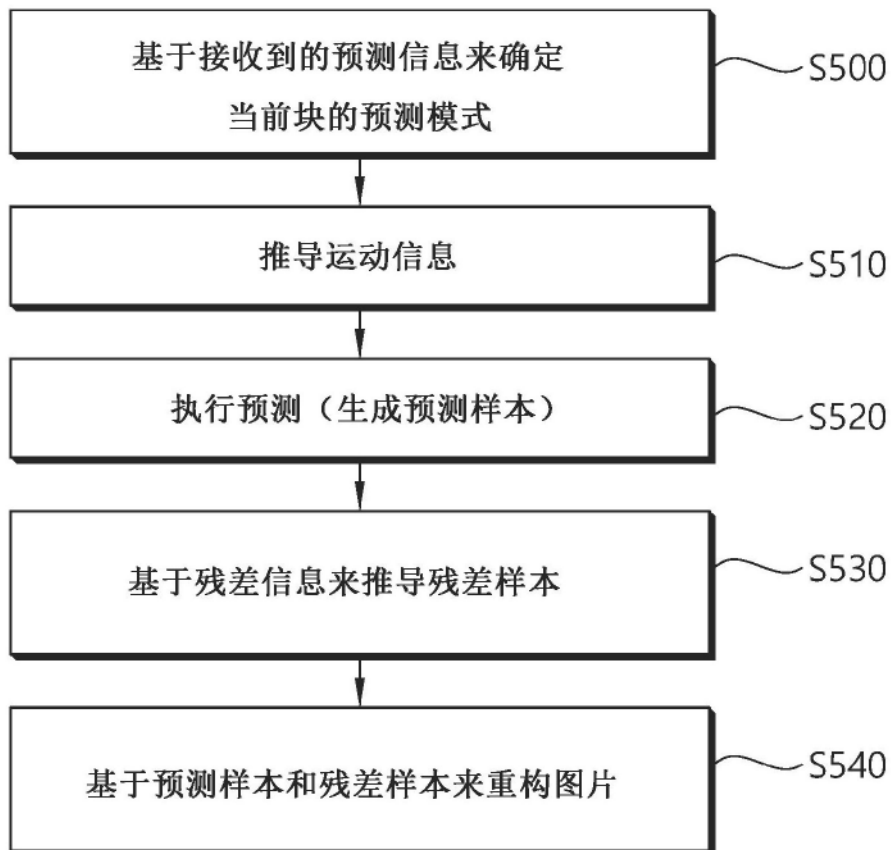


图5

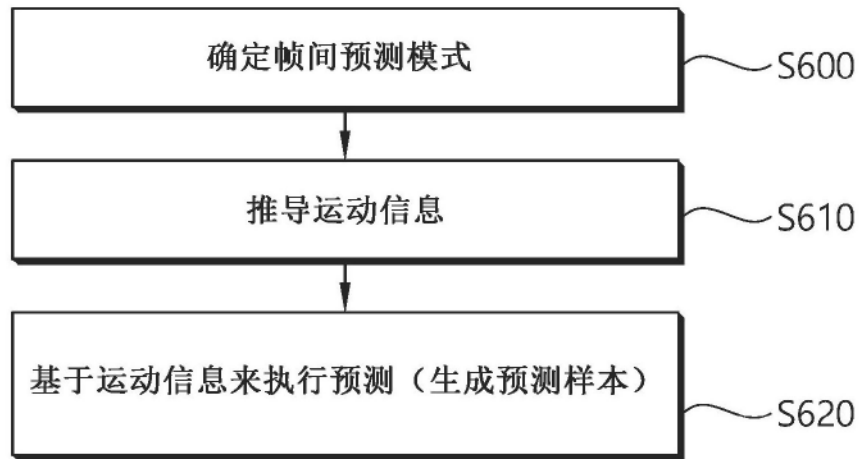
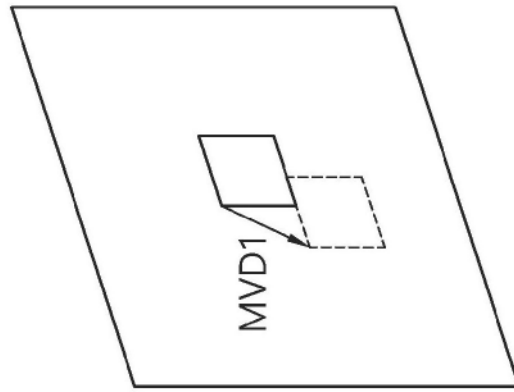
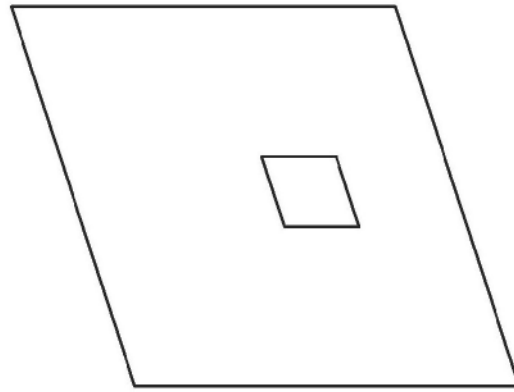


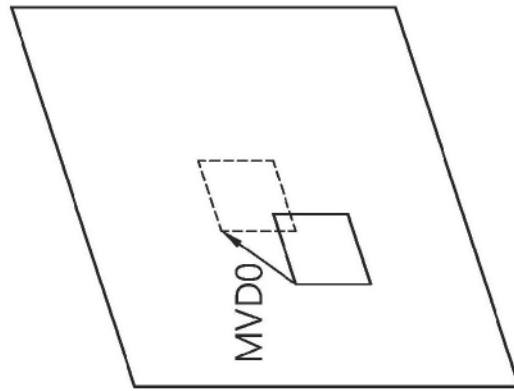
图6



列表1参考图片



当前图片



列表0参考图片

图7

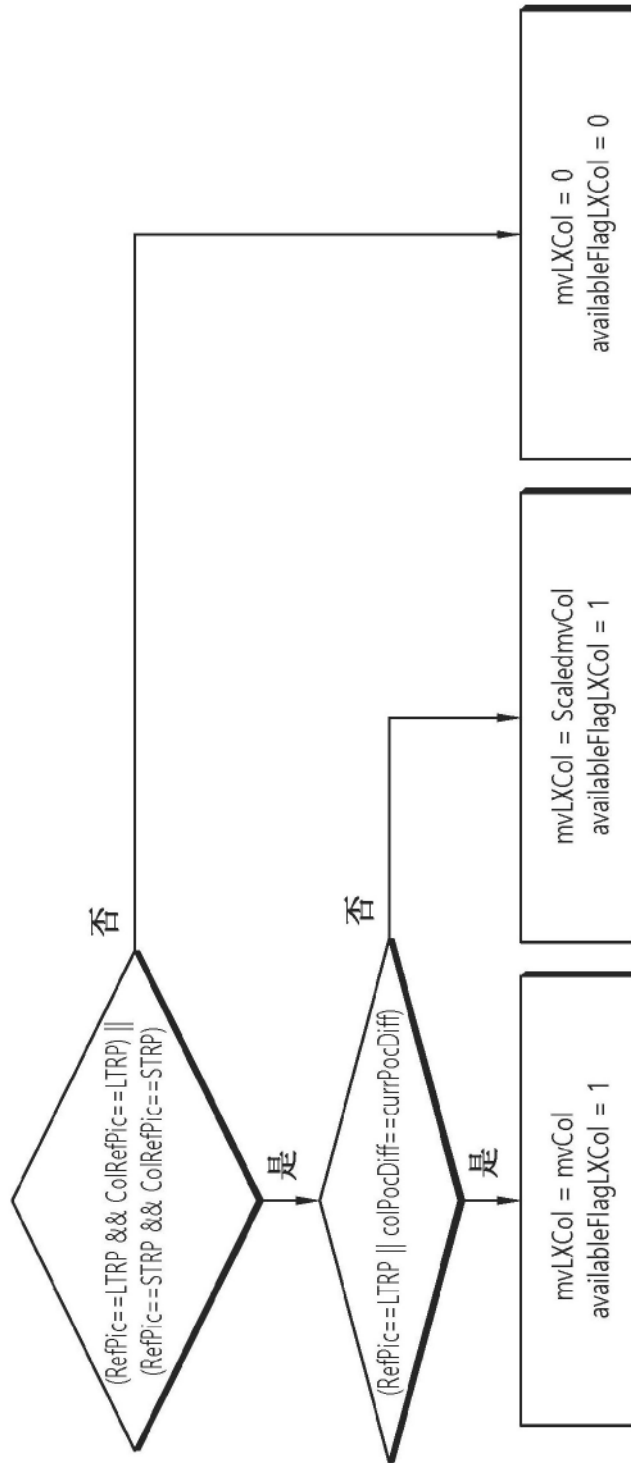


图8

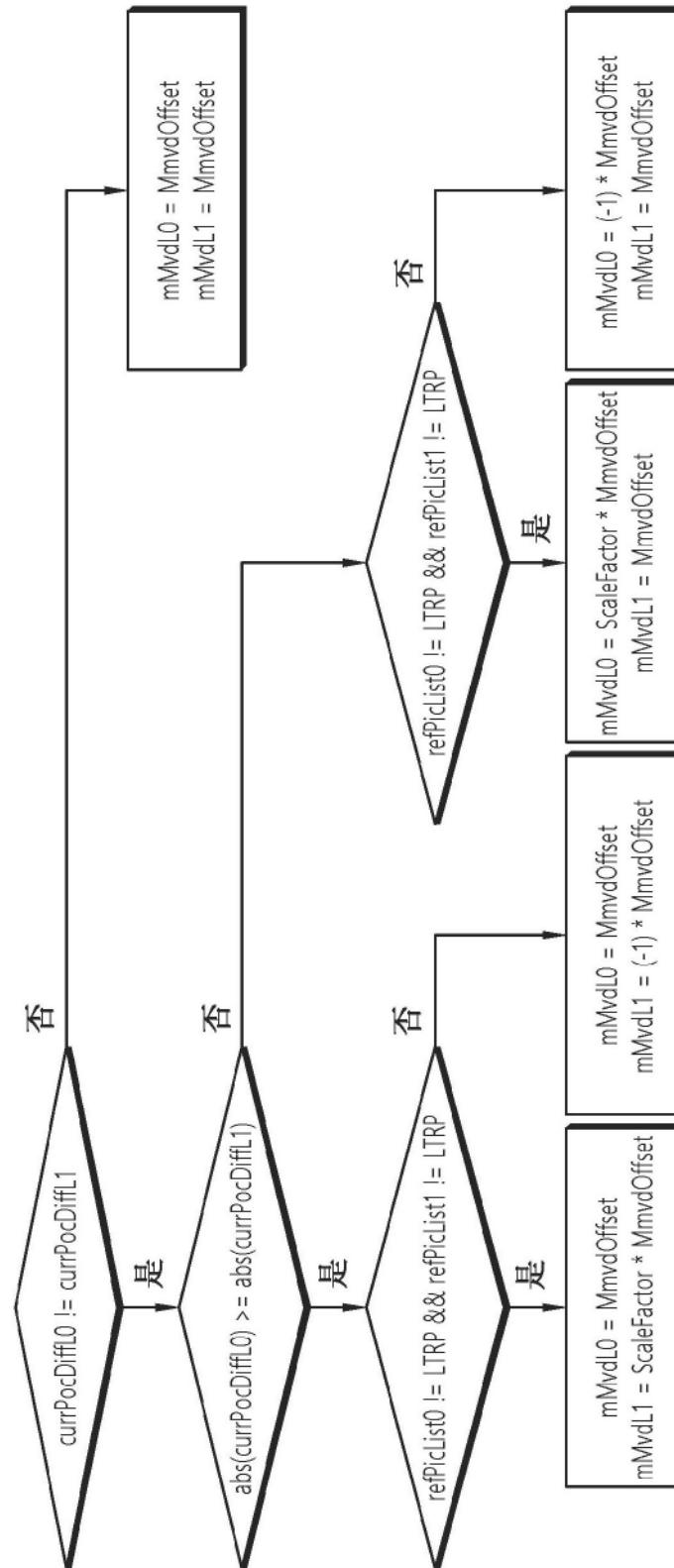


图9

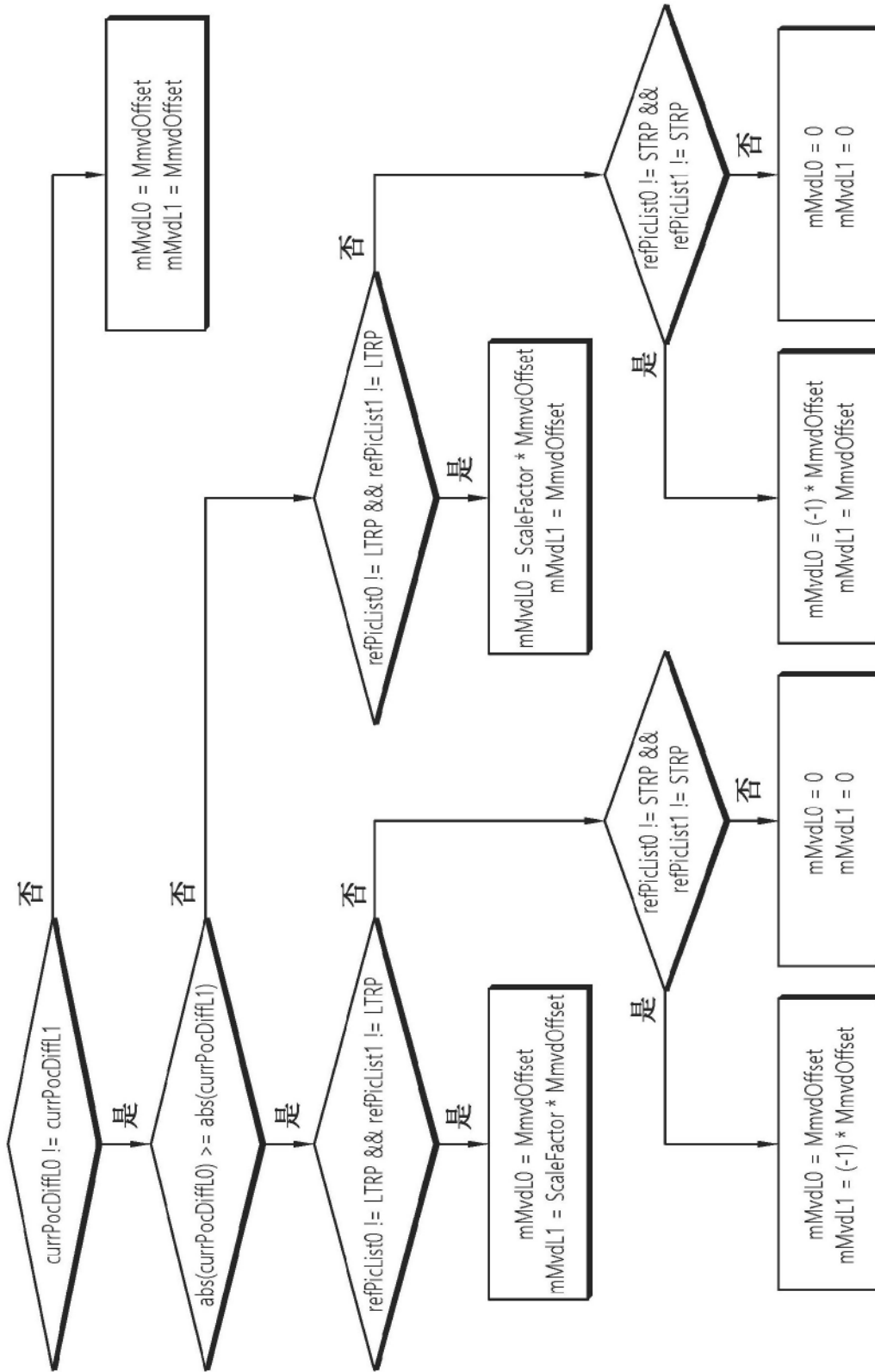


图10

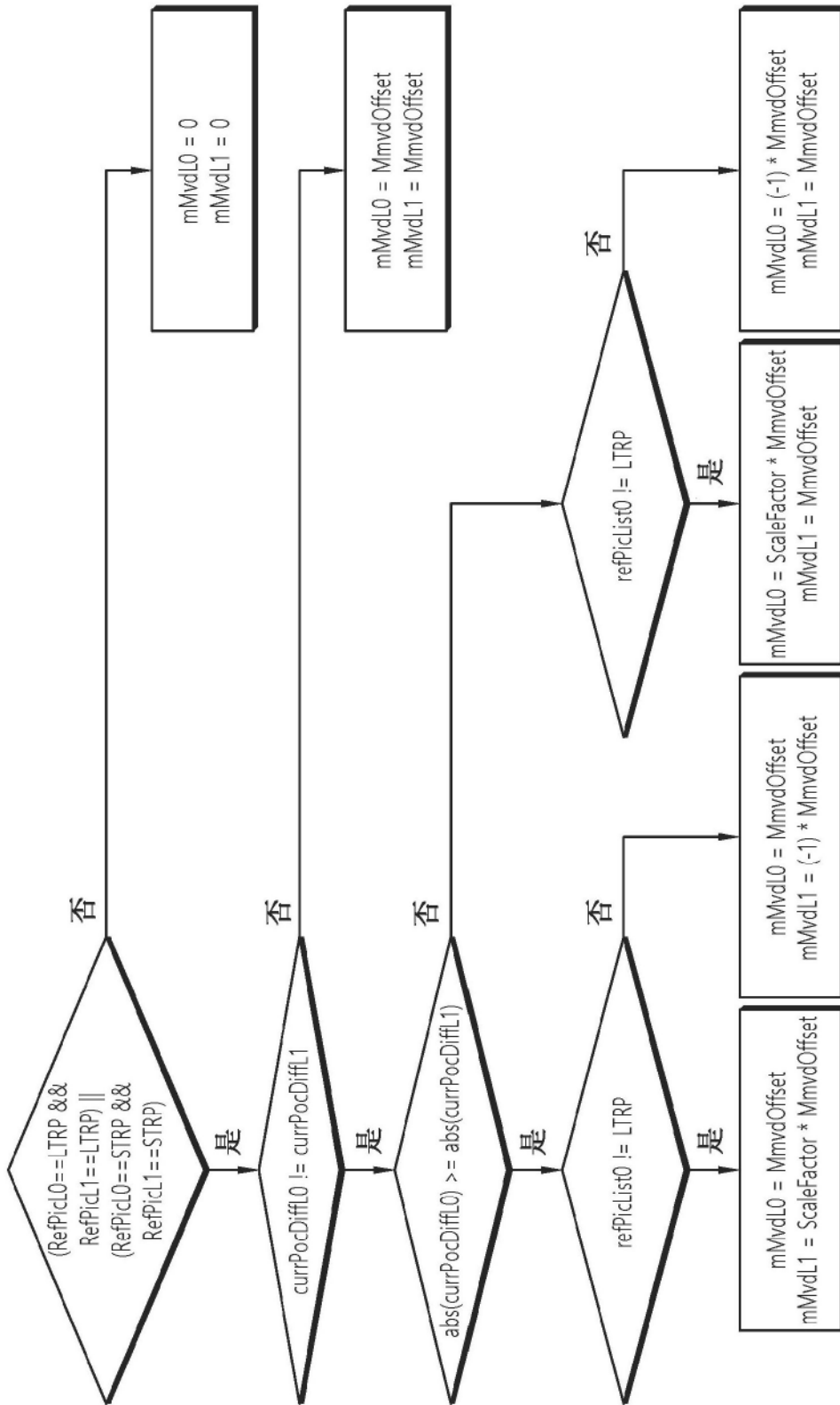


图11

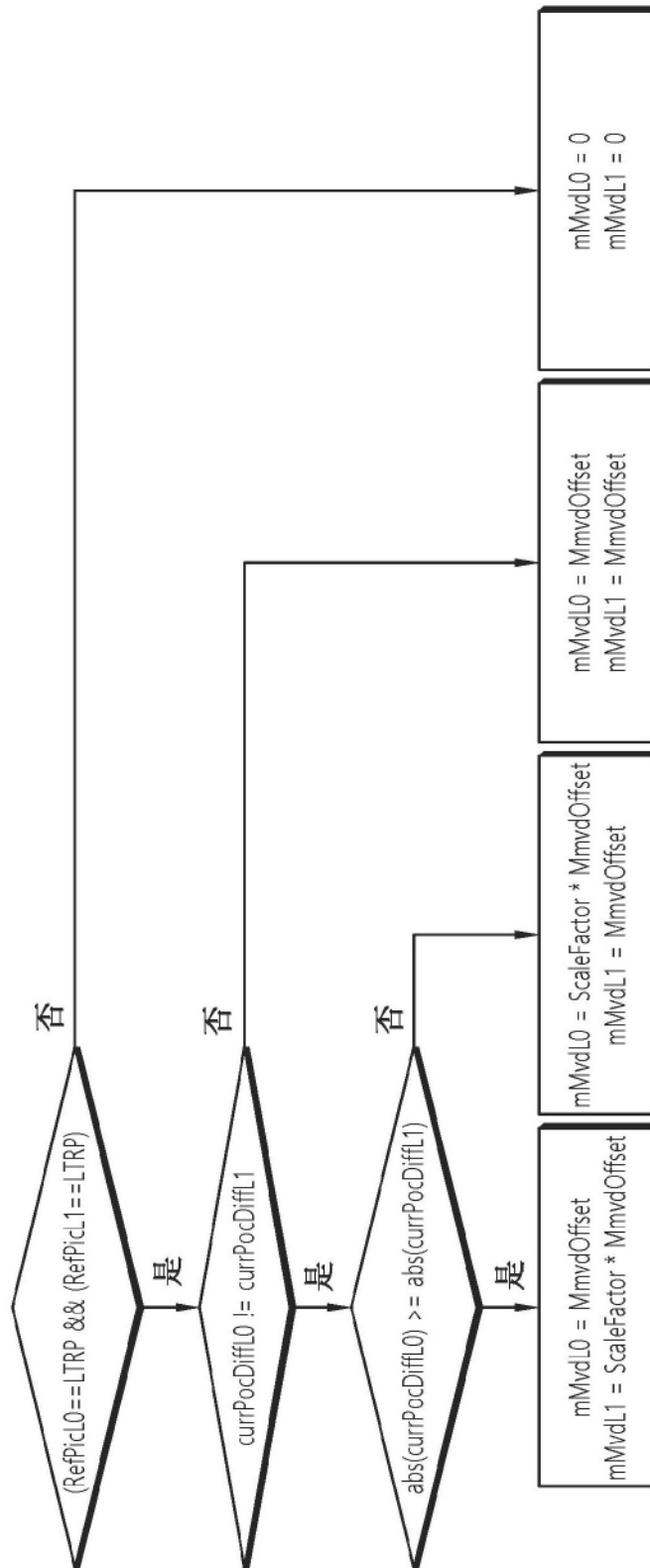


图12

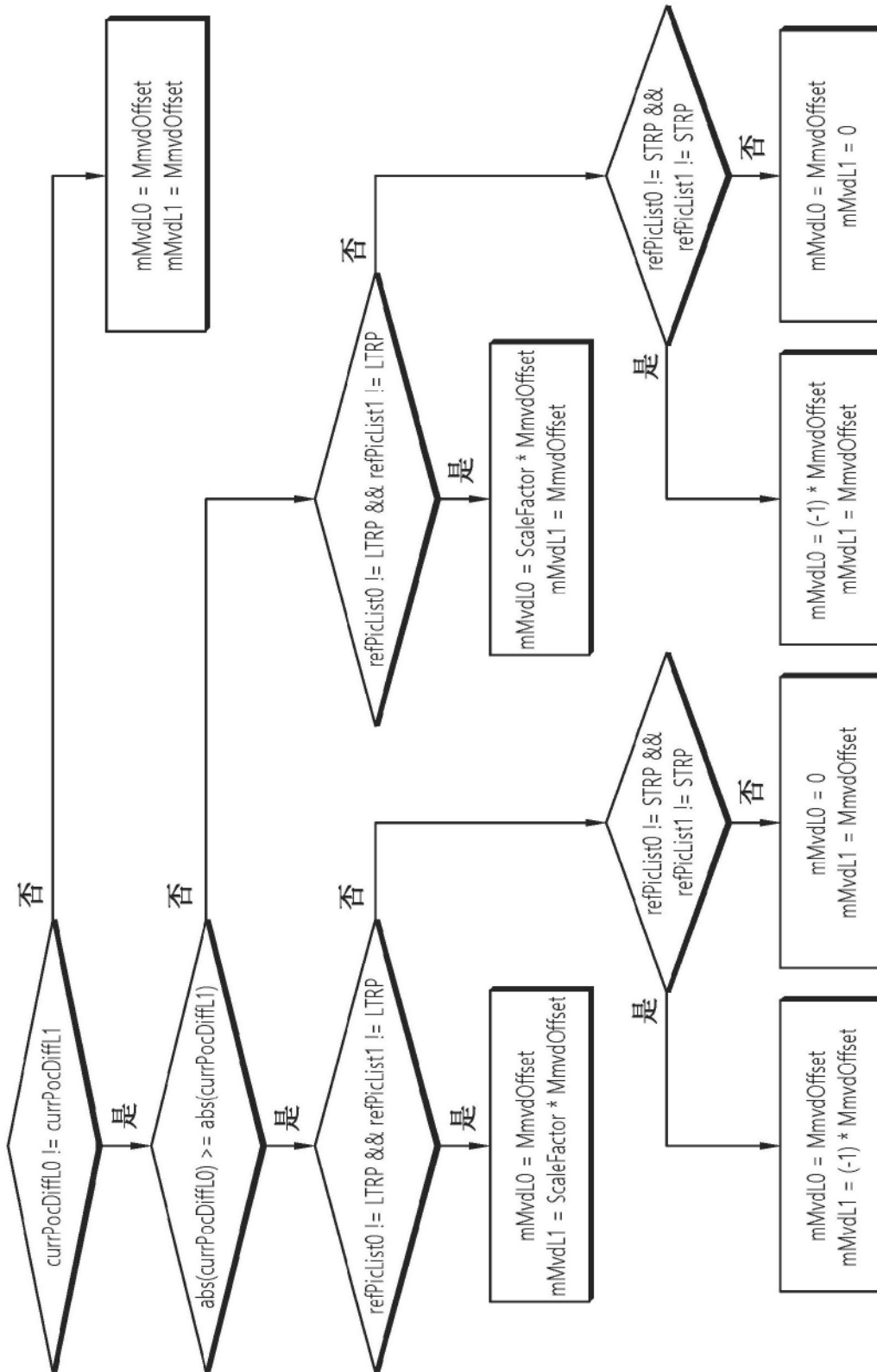


图13

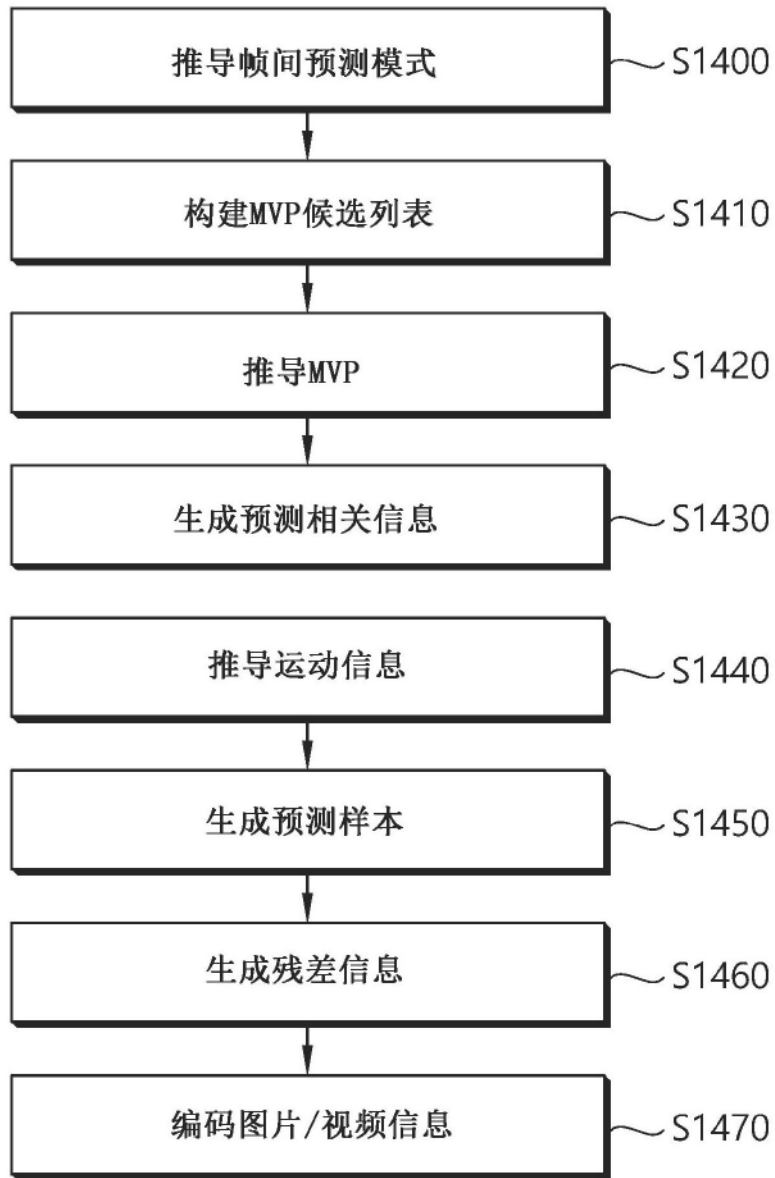


图14

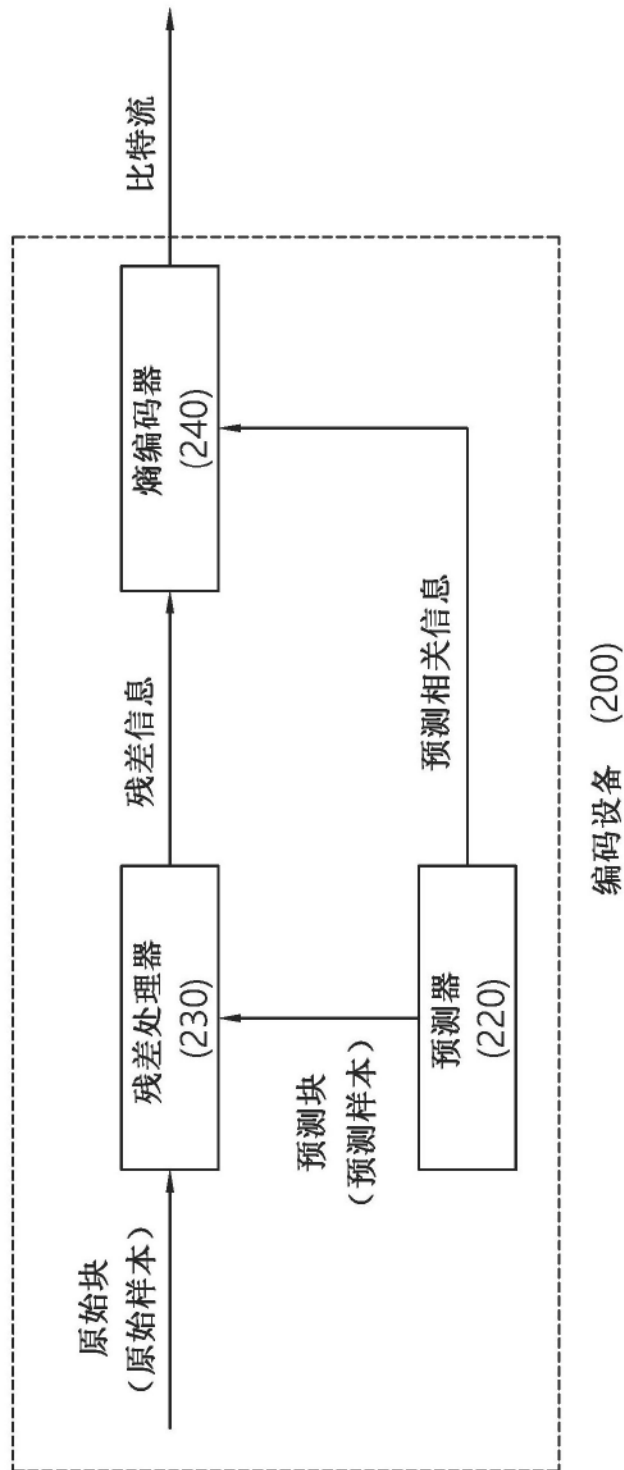


图15

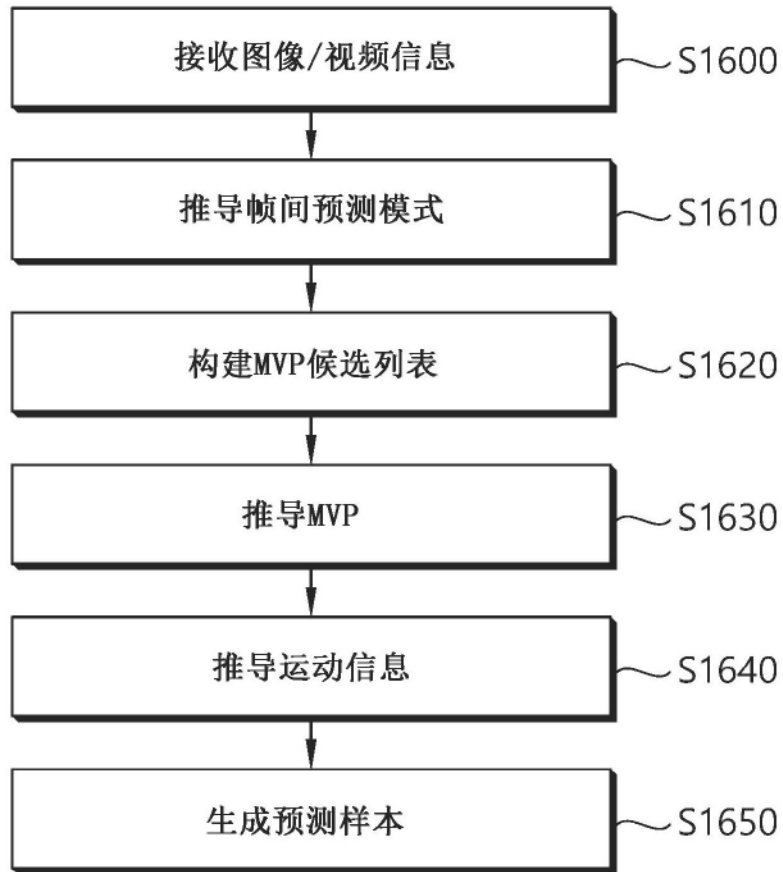


图16

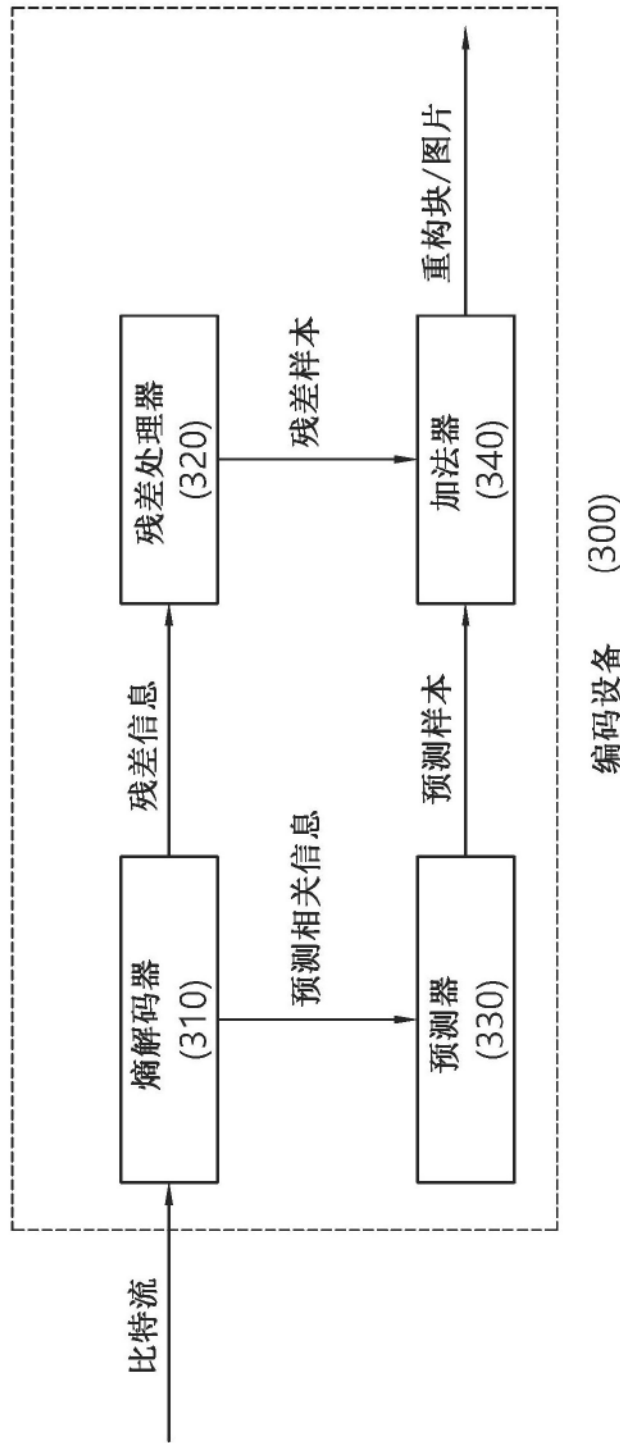


图17

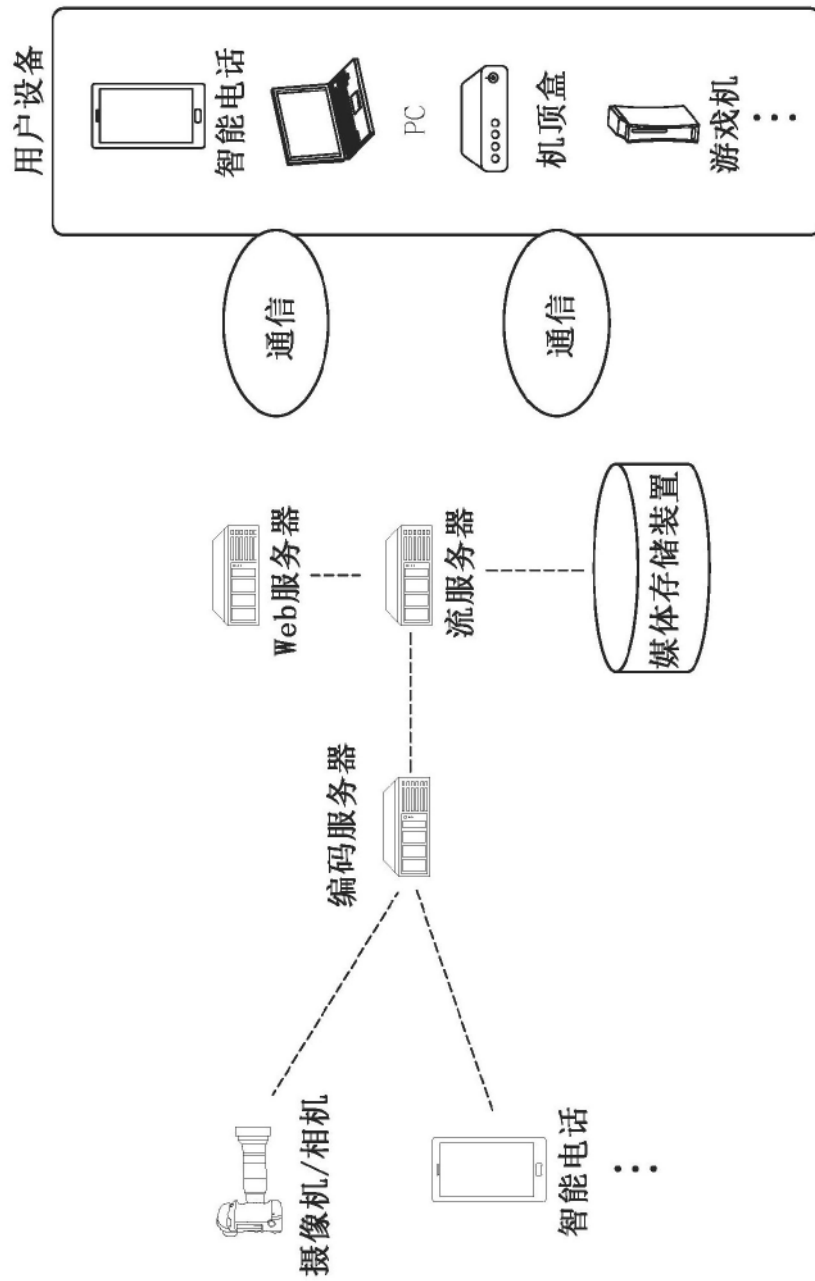


图18