



(12) 发明专利

(10) 授权公告号 CN 110574372 B

(45) 授权公告日 2023. 07. 28

(21) 申请号 201880026911.8

(22) 申请日 2018.02.26

(65) 同一申请的已公布的文献号
申请公布号 CN 110574372 A

(43) 申请公布日 2019.12.13

(30) 优先权数据
10-2017-0024643 2017.02.24 KR
10-2017-0024644 2017.02.24 KR

(85) PCT国际申请进入国家阶段日
2019.10.23

(86) PCT国际申请的申请数据
PCT/KR2018/002343 2018.02.26

(87) PCT国际申请的公布数据
W02018/155986 KO 2018.08.30

(73) 专利权人 株式会社KT
地址 韩国京畿道

(72) 发明人 李培根

(74) 专利代理机构 北京集佳知识产权代理有限公司 11227
专利代理师 王萍 李彦丽

(51) Int.Cl.
H04N 19/103 (2006.01)
H04N 19/122 (2006.01)
H04N 19/172 (2006.01)
H04N 19/174 (2006.01)
H04N 19/137 (2006.01)
H04N 19/119 (2006.01)
H04N 19/96 (2006.01)

(56) 对比文件
WO 2015101209 A1, 2015.07.09
沈燕飞; 李锦涛; 朱珍民; 张勇东; . 高效视频编码. 计算机学报. 2013, (第11期), 第164-179页.

审查员 牛耘佳

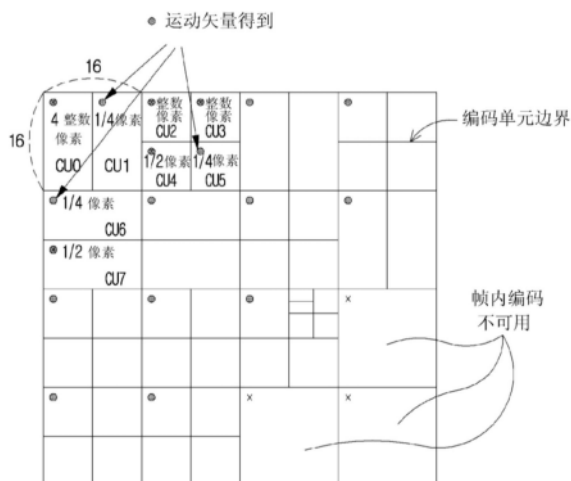
权利要求书2页 说明书38页 附图19页

(54) 发明名称

视频信号处理方法和计算机可读介质

(57) 摘要

本发明涉及视频信号处理方法和设备。根据本发明的图像解码方法包括以下步骤：从编码的语法元素中检查通过合并多个邻近编码单元生成的合并编码单元；并且解码每个所检查的合并编码单元，其中，合并编码单元共享相同的运动矢量。通过使得能够有效的合并以及编码或解码编码块，本发明可以增加图像信号编码/解码的效率。



1. 一种用于解码视频的方法,所述方法包括:

确定合并模式是否应用到当前编码块,

其中,响应于所述合并模式应用到所述当前编码块,所述当前编码块的运动矢量被设置为与所述当前编码块的合并候选相同,

其中,响应于所述合并模式没有应用到所述当前编码块,基于运动矢量预测器和运动矢量差值来确定所述当前编码块的运动矢量,

其中,基于索引信息,在多个运动矢量分辨率候选当中确定所述当前编码块的运动矢量分辨率,

其中,通过修改与所述当前编码块邻近的邻近块的运动矢量,得到所述运动矢量预测器,以及

其中,基于使用与所述当前编码块的运动矢量分辨率对应的偏移参数的偏移操作,执行所述修改。

2. 根据权利要求1所述的方法,其中,所述运动矢量分辨率候选的数量或类型在从当前图片得到所述当前编码块的预测块的情况下和在从与所述当前图片不同的参考图片得到所述当前编码块的预测块的情况下不同。

3. 根据权利要求2所述的方法,其中,以预先确定的运动矢量分辨率而不管所述当前编码块的运动矢量分辨率存储所述当前编码块的运动矢量。

4. 一种用于编码视频的方法,所述方法包括:

确定合并模式是否应用到当前编码块,

其中,响应于所述合并模式应用到所述当前编码块,所述当前编码块的运动矢量被设置为与所述当前编码块的合并候选相同,

其中,响应于所述合并模式没有应用到所述当前编码块,基于运动矢量预测器和运动矢量差值来确定所述当前编码块的运动矢量,

其中,基于指定被编码到比特流中的运动矢量分辨率的索引信息,在多个运动矢量分辨率候选当中确定所述当前编码块的运动矢量分辨率,

其中,通过修改与所述当前编码块邻近的邻近块的运动矢量,得到所述运动矢量预测器,以及

其中,基于使用与所述当前编码块的运动矢量分辨率对应的偏移参数的偏移操作,执行所述修改。

5. 根据权利要求4所述的方法,其中,所述运动矢量分辨率候选的数量或类型在从当前图片得到所述当前编码块的预测块的情况下和在从与所述当前图片不同的参考图片得到所述当前编码块的预测块的情况下不同。

6. 一种记录有程序的计算机可读介质,所述程序使计算机执行包括以下操作的方法对视频信号进行编码:

确定合并模式是否应用到当前编码块,

其中,响应于所述合并模式应用到所述当前编码块,所述当前编码块的运动矢量被设置为与所述当前编码块的合并候选相同,

其中,响应于所述合并模式没有应用到所述当前编码块,基于运动矢量预测器和运动矢量差值来确定所述当前编码块的运动矢量,

其中,基于指定被编码到比特流中的运动矢量分辨率的索引信息,在多个运动矢量分辨率候选当中确定所述当前编码块的运动矢量分辨率,

其中,通过修改与所述当前编码块邻近的邻近块的运动矢量,得到所述运动矢量预测器,以及

其中,基于使用与所述当前编码块的运动矢量分辨率对应的偏移参数的偏移操作,执行所述修改。

视频信号处理方法和计算机可读介质

技术领域

[0001] 本发明涉及用于处理视频信号的方法和设备。

背景技术

[0002] 近来,在各种应用领域中对高分辨率和高质量图像例如高清晰度(HD)图像和超高清清晰度(UHD)图像的需求已经增加了。然而,与常规图像数据相比,更高分辨率和质量的图像数据的数据量增加。因此,在通过使用介质例如常规的有线和无线宽带网络传输图像数据时,或者在通过使用常规的存储介质存储图像数据时,传输和存储的成本增加。为了解决随着图像数据的分辨率和质量的提高而出现的这些问题,可以利用高效的图像编码/解码技术。

[0003] 图像压缩技术包括各种技术,包括:根据当前图片的先前图片或后续图片来对包括在当前图片中的像素值进行预测的帧间预测技术;通过使用当前图片中的像素信息对包括在当前图片中的像素值进行预测的帧内预测技术;将短代码分配给出现频率高的值并且将长代码分配给出现频率低的值的熵编码技术等。可以通过使用这样的图像压缩技术来有效地压缩图像数据,并且可以传输或存储图像数据。

[0004] 同时,随着对高分辨率图像的需求的增加,对作为新的图像服务的立体图像内容的需求也在增加。正在讨论用于有效地提供具有高分辨率和超高分辨率的立体图像内容的视频压缩技术。

发明内容

[0005] 技术问题

[0006] 本发明的目的是提供一种用于多叉树划分的方法和设备,其可以用于在编码/解码视频信号时有效地划分编码/解码目标块。

[0007] 本发明的目的是提供一种用于多叉树划分的方法和设备,用于在编码/解码视频信号时将编码/解码目标块划分成对称块或者非对称块。

[0008] 本发明的目的是提供一种用于生成与由多叉树划分来划分的编码块相对应的合并编码块的方法和设备。

[0009] 本发明的目的是提供一种通过利用运动矢量精度确定代表性运动矢量的方法和设备。

[0010] 本发明的目的是提供一种包括通过各种编码方法编码的视频信号比特流的记录介质。

[0011] 本发明要实现的技术目的不限于上述技术问题。并且,本领域技术人员根据以下描述将明显地理解未提及的其他技术问题。

[0012] 技术方案

[0013] 根据本发明的用于解码视频信号的方法包括:基于编码的语法元素检查通过合并彼此邻近的多个编码单元而生成的合并编码单元,并且对所检查的合并编码单元进行解

码,其中在合并编码单元中共享相同运动矢量。

[0014] 另外,通过利用编码的语法元素来确定应用于合并编码单元的运动矢量。

[0015] 另外,将多个编码单元当中的按编码顺序的第一编码单元确定为合并候选编码单元,并且将所确定的合并候选编码单元的运动矢量应用为合并编码单元的运动矢量。

[0016] 另外,编码的语法元素包括:第一语法元素(CU_merge_flag)和第二语法元素(CU_merge_idx),第一语法元素指示编码单元之间是否存在合并,第二语法元素定义在根据第一语法元素发生了合并时合并编码单元的形状。

[0017] 另外,第二语法元素(CU_merge_idx)指示在应用了三叉树划分的三个编码单元当中,按编码顺序的前两个编码单元是否被合并,或者按编码顺序的后两个编码单元是否被合并。

[0018] 另外,第二语法元素(CU_merge_idx)指示:在应用了四叉树划分的四个编码单元当中,按编码顺序的第一编码单元和第二编码单元是否被合并,按编码顺序的第三编码单元和四个编码单元是否被合并,按编码顺序的第一编码单元和第三编码单元是否被合并,或者按编码顺序的第二编码单元和第四编码单元是否被合并。

[0019] 另外,通过使用第一语法元素(CU_merge_flag)和第二语法元素(CU_merge_idx)来区分编码单元的划分类型。

[0020] 另外,同一码字被应用于由第一语法元素(CU_merge_flag)和第二语法元素(CU_merge_idx)区分的编码单元的划分类型。

[0021] 根据本发明的用于解码视频信号的方法包括:确定用于包括多个下编码块的上编码块的代表性运动矢量,通过利用所确定的作为参考上编码块的当前编码块的时间运动矢量候选的代表性运动矢量来得到当前编码块的运动矢量,并且使用当前编码块的运动矢量执行当前编码块的运动补偿。

[0022] 另外,通过利用下编码块的运动矢量精度来确定表示上编码块的代表性运动矢量。

[0023] 另外,将表示上编码块的代表性运动矢量确定为在下编码块的运动矢量中具有最精确的运动矢量精度的运动矢量。

[0024] 另外,将表示上编码块的代表性运动矢量确定为下编码块的运动矢量当中的具有最不精确的运动矢量精度的运动矢量。

[0025] 另外,表示上编码块的代表性运动矢量是基于下编码块的位置确定的。

[0026] 另外,将表示上编码块的代表性运动矢量确定为在下编码块当中的包括左上样本的编码块的运动矢量。

[0027] 根据本发明的用于编码视频信号的方法包括:通过合并彼此邻近的多个编码单元来生成合并编码单元,编码所生成的合并编码单元,以及用信号发送与合并编码单元有关的语法元素,其中,在合并编码单元中共享相同运动矢量。

[0028] 另外,用信号发送的语法元素包括第一语法元素(CU_merge_flag)和第二语法元素(CU_merge_idx),第一语法元素指示在编码单元之间是否存在合并,以及第二语法元素定义在根据第一语法元素发生了合并时合并编码单元的形状。

[0029] 根据本发明的用于编码视频信号的方法包括:确定用于包括多个下编码块的上编码块的代表性运动矢量,通过利用所确定的代表性运动矢量作为参考上编码块的当前编码

块的时间运动矢量候选来得到当前编码块的运动矢量,以及使用当前编码块的运动矢量执行当前编码块的运动补偿。

[0030] 根据本发明的用于解码视频信号的设备包括解码单元,该解码单元用于:基于编码的语法元素对通过合并彼此邻近的多个编码单元而生成的合并编码单元进行检查,并且解码所检查的合并编码单元,其中在合并编码单元中共享相同运动矢量。

[0031] 根据本发明的用于解码视频信号的设备包括解码单元,该解码单元用于:确定用于包括多个下编码块的上编码块的代表性运动矢量,通过利用所确定的代表性运动矢量作为参考上编码块的当前编码块的时间运动矢量候选来得到当前编码块的运动矢量,并使用当前编码块的运动矢量来执行当前编码块的运动补偿。

[0032] 一种包括视频信号比特流的记录介质,包括在记录介质中的视频信号比特流通过编码方法编码,该编码方法包括:通过合并彼此邻近的多个编码单元来生成合并编码单元,编码所生成的合并编码单元,以及用信号发送与合并编码单元有关的语法元素,其中,在合并编码单元中共享相同运动矢量。

[0033] 一种包括视频信号比特流的记录介质,包括在记录介质中的视频信号比特流通过编码方法编码,该编码方法包括:确定用于包括多个下编码块的上编码块的代表性运动矢量,通过利用所确定的代表性运动矢量作为参考上编码块的当前编码块的时间运动矢量候选来得到当前编码块的运动矢量,并使用当前编码块的运动矢量执行当前编码块的运动补偿。

[0034] 以上对本发明简要概述的特征仅是随后对本发明的详细描述的说明确性方面,而不限制本发明的范围。

[0035] 有益效果

[0036] 根据本发明,通过有效地划分编码/解码目标块来提高视频信号的编码/解码效率。

[0037] 根据本发明,通过将编码/解码目标块划分成对称块或非对称块来增强视频信号的编码/解码效率。

[0038] 根据本发明,通过生成合并编码单元来增强由多叉树划分划分的视频信号的编码/解码效率。

[0039] 根据本发明,由于可以通过各种精度来表示运动矢量并且可以通过利用各种精度来确定代表性运动矢量,因此增强了视频信号的编码/解码效率。

[0040] 能够通过本发明获得的效果不限于上述效果,并且本领域技术人员可以根据以下描述清楚地理解未提及的其他效果。

附图说明

[0041] 图1是示出根据本发明的实施方式的用于对视频进行编码的设备的框图。

[0042] 图2是示出根据本发明的实施方式的用于对视频进行解码的设备的框图。

[0043] 图3是示出可以应用于编码块的划分模式的图。

[0044] 图4是示出根据本发明实施方式的允许四叉树划分和二叉树划分的划分类型的图。

[0045] 图5示出了根据应用本发明的实施方式的基于四叉树划分和二叉树划分来分层划

分编码块的示例。

[0046] 图6示出了根据应用本发明的实施方式的基于四叉树划分和对称二叉树划分来分层划分编码块的示例。

[0047] 图7是示出作为应用本发明的实施方式的允许非对称二叉树划分的划分类型的图。

[0048] 图8示出了作为应用本发明的实施方式的基于四叉树划分和对称/非对称二叉树划分的编码块的划分类型。

[0049] 图9是示出根据应用了本发明的实施方式的基于四叉树划分和二叉树划分的编码块划分方法的流程图。

[0050] 图10示出了作为应用本发明的实施方式,包括在应用了四叉树划分和二叉树划分的网络抽象层(NAL)中的语法元素。

[0051] 图11是示出作为应用本发明的另一实施方式的允许非对称四叉树划分的划分类型的图。

[0052] 图12是示出作为应用本发明的另一实施方式的基于非对称四叉树划分的编码块划分方法的流程图。

[0053] 图13示出了作为应用本发明的另一实施方式的包括在应用非对称四叉树划分的网络抽象层(NAL)中的语法元素。

[0054] 图14是示出作为应用本发明的另一实施方式的允许四叉树划分和三叉树划分的划分类型被允许的图。

[0055] 图15是示出作为应用本发明的另一实施方式的基于四叉树划分和三叉树划分的编码块划分方法的流程图。

[0056] 图16示出了作为应用本发明的另一实施方式,包括在应用了四叉树划分和三叉树划分的网络抽象层(NAL)中的语法元素。

[0057] 图17是示出根据本发明另一实施方式的允许多叉树划分的划分类型的图。

[0058] 图18是示出作为应用本发明的另一实施方式的允许多叉树划分的扩展划分类型的图。

[0059] 图19是示出作为应用本发明的另一实施方式的基于多叉树划分的编码块划分方法的流程图。

[0060] 图20是示出作为应用本发明的实施方式的生成和编码/解码合并编码单元的方法的流程图。

[0061] 图21和22示出了用于解释合并编码单元的编码单元的类型示例。

[0062] 图23是示出使用语法元素CuMerge_idx的值的视频编码方法或视频解码方法的流程图。

[0063] 图24是示出作为应用本发明的实施方式的帧间预测方法的流程图。

[0064] 图25是示出在将合并模式应用于当前块时得到当前块的运动信息的过程的图。

[0065] 图26是示出在将高级运动矢量预测器(AMVP)模式应用于当前块时得到当前块的运动信息的过程的图。

[0066] 图27和28是示出根据当前块的运动矢量精度的运动矢量推导方法的图。

[0067] 图29和30是用于说明在多个运动矢量单元中得到时间运动矢量(时间MV)的方法

的图。

[0068] 图31示出了作为应用本发明的另一实施方式,包括在应用于帧内预测样本差值的网络抽象层(NAL)中的语法元素。

具体实施方式

[0069] 可以对本发明进行各种修改,并且存在本发明的各种实施方式,现在将参照附图提供各种实施方式的示例,并且详细描述各种实施方式的示例。然而,本发明不限于此,并且示例性实施方式可以被解释为包括本发明的技术构思和技术范围内的所有修改、等同物或备选方案。在所描述的附图中,相似的附图标记指代相似的元件。

[0070] 说明书中使用的术语“第一”、“第二”等可以用于描述各种部件,但是这些部件不被解释为限制于这些术语。这些术语仅用于区分一个部件与其他部件。例如,在不偏离本发明的范围的情况下,“第一”部件可以被称作“第二”部件,并且“第二”部件也可以类似地被称作“第一”部件。术语“和/或”包括多个项的组合或者多个项中的任何一个项。

[0071] 本说明书中使用的术语仅用于描述特定实施方式,而并不意在限制本发明。以单数形式使用的表述包含复数形式的表述,除非其在上下文中具有明显不同的含义。在本说明书中,应当理解的是,诸如“包括”、“具有”等的术语意在指示本说明书中公开的特征、数字、步骤、动作、元件、部分或其组合的存在,并且不意在排除可以存在或可以添加一个或更多个其他特征、数字、步骤、动作、元件、部分或其组合的可能性。

[0072] 另外,本申请中使用的术语“单元”可以由“块”代替,并且因此,在本说明书中,成对的“编码树单元”和“编码树块”、“编码单元”和“编码块”、“预测单元”和“预测块”以及“变换单元”和“变换块”中的每个术语可以被解释为具有相同的含义。

[0073] 在下文中,将参照附图详细描述本发明的优选实施方式。在下文中,附图中的相同构成元件由相同的附图标记表示,并且将省略对相同元件的重复描述。

[0074] 图1是示出根据本发明的实施方式的用于对视频进行编码的设备的框图。

[0075] 参照图1,用于对视频进行编码的设备100可以包括:图片划分模块110、预测模块120和125、变换模块130、量化模块135、重排模块160、熵编码模块165、逆量化模块140、逆变换模块145、滤波器模块150以及存储器155。

[0076] 图1中所示的构成部分被独立地示出,以表示用于对视频进行编码的设备中的彼此不同的特征功能。因此,这并不意味着每个构成部分都是由单独的硬件或软件的构成单元构成。换言之,为了方便起见,每个构成部分包括所列举的构成部分中的每一个。因此,可以将每个构成部分的至少两个构成部分进行组合以形成一个构成部分,或者可以将一个构成部分分割成多个构成部分以执行每个功能。在不偏离本发明的实质的情况下,组合每个构成部分的实施方式和分割一个构成部分的实施方式也被包括在本发明的范围内。

[0077] 此外,构成部分中的一些可能不是执行本发明的基本功能的必不可少的构成部分,而是仅用于改善本发明的性能的可选构成部分。可以通过排除用于改善性能的构成部分而仅包括用于实现本发明的实质的必不可少的构成部分来实现本发明。排除仅用于改善性能的可选构成部分而仅包括必不可少的构成部分的结构也被包括在本发明的范围内。

[0078] 图片划分模块110可以将输入图片划分成一个或更多个处理单元。此处,处理单元可以是预测单元(PU)、变换单元(TU)或编码单元(CU)。图片划分模块110可以将一个图片划

分成多个编码单元、预测单元和变换单元的组合,并且可以通过使用预定准则(例如,成本函数)选择编码单元、预测单元和变换单元的一个组合来对图片进行编码。

[0079] 例如,一个图片可以被划分成多个编码单元。可以使用递归树结构例如四叉树结构来将图片划分成编码单元。在一个图片或最大编码单元作为根的情况下被划分成其他编码单元的编码单元可以以子节点对应于所划分的编码单元的数目的方式进行划分。通过预定限制不能再划分的编码单元用作叶节点。即,当假设对于一个编码单元仅正方形划分可行时,一个编码单元可以最多被划分成四个其他编码单元。

[0080] 在下文中,在本发明的实施方式中,编码单元可以意指执行编码的单元或者执行解码的单元。

[0081] 预测单元可以是被划分成在单个编码单元中具有相同大小的正方形形状或矩形形状的划分中的之一,或者预测单元可以是被划分成使得在单个编码单元中具有不同的形状/大小的划分中的之一。

[0082] 当基于编码单元生成要进行帧内预测的预测单元并且编码单元不是最小编码单元时,可以在不将编码单元划分成多个预测单元 $N \times N$ 的情况下执行帧内预测。

[0083] 预测模块120和125可以包括执行帧间预测的帧间预测模块120和执行帧内预测的帧内预测模块125。可以确定对于预测单元是执行帧间预测还是帧内预测,并且可以确定根据每个预测方法的详细信息(例如,帧内预测模式、运动矢量、参考图片等)。此处,要进行预测的处理单元可以不同于针对其确定预测方法和详细内容的处理单元。例如,可以由预测单元确定预测方法、预测模式等,并且可以由变换单元执行预测。所生成的预测块与原始块之间的残差值(残差块)可以被输入至变换模块130。此外,用于预测的预测模式信息、运动矢量信息等可以与残差值一起由熵编码模块165进行编码,并且可以被传输至用于对视频进行解码的设备。在使用特定编码模式时,可以通过对原始块按其原样进行编码而不通过预测模块120和125生成预测块来向用于对视频进行解码的设备进行传输。

[0084] 帧间预测模块120可以基于当前图片的先前图片或后续图片中的至少之一的信息来预测预测单元,或者在一些情况下,可以基于当前图片中的一些编码区域的信息来预测预测单元。帧间预测模块120可以包括参考图片插值模块、运动预测模块以及运动补偿模块。

[0085] 参考图片插值模块可以从存储器155接收参考图片信息,并且可以根据参考图片来生成整数像素或小于整数像素的像素信息。在亮度像素的情况下,可以使用具有不同滤波器系数的基于DCT的8抽头插值滤波器来以 $1/4$ 像素为单位生成整数像素或小于整数像素的像素信息。在色度信号的情况下,可以使用具有不同滤波器系数的基于DCT的4抽头插值滤波器来以 $1/8$ 像素为单位生成整数像素或小于整数像素的像素信息。

[0086] 运动预测模块可以基于由参考图片插值模块进行插值的参考图片来执行运动预测。作为用于计算运动矢量的方法,可以使用各种方法,例如,基于全搜索的块匹配算法(FBMA)、三步搜索(TSS)和新三步搜索算法(NTS)等。基于插值像素,运动矢量可以具有以 $1/2$ 像素或 $1/4$ 像素为单位的运动矢量值。运动预测模块可以通过改变运动预测方法来预测当前预测单元。作为运动预测方法,可以使用各种方法,例如,跳过方法、合并方法、AMVP(高级运动矢量预测)方法、帧内块复制方法等。

[0087] 帧内预测模块125可以基于作为当前图片中的像素信息的、与当前块邻近的参考

像素信息来生成预测单元。在当前预测单元的邻近块是要进行帧间预测的块并且因此参考像素是要进行帧间预测的像素时,可以使用要进行帧内预测的邻近块的参考像素信息来替换包括在要进行帧间预测的块中的参考像素。即,在参考像素不可用时,可以使用可用参考像素中的至少之一参考像素来替换不可用的参考像素信息。

[0088] 帧内预测中的预测模式可以包括依赖于预测方向使用参考像素信息的方向性预测模式和在执行预测时不使用方向信息的非方向性预测模式。用于预测亮度信息的模式可以与用于预测色度信息的模式不同,并且为了预测色度信息,可以利用用于预测亮度信息的帧内预测模式信息或者预测的亮度信号信息。

[0089] 在执行帧内预测时,在预测单元的大小与变换单元的大小相同时,可以基于位于该预测单元的左侧、左上侧和上部的像素对预测单元执行帧内预测。然而,在执行帧内预测时,在预测单元的大小与变换单元的大小不同时,可以使用基于变换单元的参考像素来执行帧内预测。此外,使用 $N \times N$ 划分的帧内预测可以仅用于最小编码单元。

[0090] 在帧内预测方法中,可以在依赖于预测模式将AIS(自适应帧内平滑)滤波器应用于参考像素之后生成预测块。应用于参考像素的AIS滤波器的类型可以变化。为了执行帧内预测方法,可以根据与当前预测单元邻近的预测单元的帧内预测模式来预测当前预测单元的帧内预测模式。在通过使用根据邻近预测单元预测到的模式信息来预测当前预测单元的预测模式中,在当前预测单元的帧内预测模式与邻近预测单元的帧内预测模式相同时,可以使用预定标志信息来传输指示当前预测单元的预测模式与邻近预测单元的预测模式彼此相同的信息。在当前预测单元的预测模式与邻近预测单元的预测模式不同时,可以执行熵编码以对当前块的预测模式信息进行编码。

[0091] 此外,可以基于由预测模块120和125生成的预测单元来生成包括关于残差值的信息的残差块,残差值是要进行预测的预测单元与预测单元的原始块之间的差。可以将所生成的残差块输入至变换模块130。

[0092] 变换模块130可以通过使用诸如离散余弦变换(DCT)、离散正弦变换(DST)和KLT的变换方法来对残差块进行变换,该残差块包括关于原始块与由预测模块120和125生成的预测单元之间的残差值的信息。可以基于用于生成残差块的预测单元的帧内预测模式信息来确定是应用DCT、DST还是KLT以对残差块进行变换。

[0093] 量化模块135可以对由变换模块130变换至频域的值进行量化。量化系数可以依赖于图片的块或重要性而变化。可以将由量化模块135计算的值提供至逆量化模块140和重排模块160。

[0094] 重排模块160可以对量化后的残差值的系数进行重排。

[0095] 重排模块160可以通过系数扫描方法将二维块形式的系数改成一维矢量形式的系数。例如,重排模块160可以使用锯齿形(zigzag)扫描方法从DC系数扫描至高频域中的系数,以将系数改成一维矢量形式。依赖于变换单元的大小和帧内预测模式,可以使用沿列方向扫描二维块形式的系数的垂直方向扫描或沿行方向扫描二维块形式的系数的水平方向扫描来代替锯齿形扫描。即,可以依赖于变换单元的大小和帧内预测模式来确定使用锯齿形扫描、垂直方向扫描和水平方向扫描中的哪种扫描方法。

[0096] 熵编码模块165可以基于由重排模块160计算的值来执行熵编码。熵编码可以使用各种编码方法,例如指数哥伦布(Golomb)编码、上下文自适应变长编码(CAVLC)和上下文自

适应二进制算术编码(CABAC)。

[0097] 熵编码模块165可以对来自重排模块160以及预测模块120和125的各种信息进行编码,各种信息例如编码单元的残差值系数信息和块类型信息、预测模式信息、划分单元信息、预测单元信息、变换单元信息、运动矢量信息、参考帧信息、块插值信息、滤波信息等。

[0098] 熵编码模块165可以对从重排模块160输入的编码单元的系数进行熵编码。

[0099] 逆量化模块140可以对由量化模块135量化的值进行逆量化,并且逆变换模块145可以对由变换模块130变换的值进行逆变换。可以将由逆量化模块140和逆变换模块145生成的残差值与由预测模块120和125的运动估计模块、运动补偿模块和帧内预测模块预测的预测单元进行组合,使得可以生成重构块。

[0100] 滤波器模块150可以包括去块滤波器、偏移校正单元以及自适应环路滤波器(ALF)中的至少之一。

[0101] 去块滤波器可以移除由于重构图片中的块之间的边界而出现的块失真。为了确定是否执行去块,包括在块的若干行或若干列中的像素可以是确定是否对当前块应用去块滤波器的基础。当去块滤波器被应用于块时,可以依赖于所需的去块滤波强度来应用强滤波器或弱滤波器。此外,在应用去块滤波器时,可以并行地处理水平方向滤波和垂直方向滤波。

[0102] 偏移校正模块可以在要进行去块的图片中以像素为单位来校正与原始图片的偏移。为了对特定图片执行偏移校正,可以使用考虑每个像素的边缘信息来施加偏移的方法,或者可以使用下述方法:将图片的像素划分成预定数量的区域,确定要执行偏移的区域,并且对所确定的区域施加偏移。

[0103] 可以基于通过将滤波后的重构图片与原始图片进行比较而获得的值来执行自适应环路滤波(ALF)。可以将包括在图片中的像素分成预定组,可以确定要应用于每个组的滤波器,并且可以针对每个组单独执行滤波。可以通过编码单元(CU)传输关于是否应用ALF和亮度信号的信息。ALF滤波器的形状和滤波器系数可以依赖于每个块而变化。此外,无论应用目标块的特征如何,都可以应用相同形状(固定形状)的ALF滤波器。

[0104] 存储器155可以存储通过滤波器模块150计算的重构块或重构图片。可以在执行帧间预测时将所存储的重构块或重构图片提供至预测模块120和125。

[0105] 图2是示出根据本发明的实施方式的用于对视频进行解码的设备的框图。

[0106] 参照图2,用于对视频进行解码的设备200可以包括:熵解码模块210、重排模块215、逆量化模块220、逆变换模块225、预测模块230和235、滤波器模块240以及存储器245。

[0107] 当从用于对视频进行编码的设备输入视频比特流时,可以根据用于对视频进行编码的设备的逆处理来对输入比特流进行解码。

[0108] 熵解码模块210可以根据由用于对视频进行编码的设备的熵编码模块进行的熵编码的逆处理来执行熵解码。例如,对应于由用于对视频进行编码的设备执行的方法,可以应用各种方法,例如指数哥伦布编码、上下文自适应变长编码(CAVLC)和上下文自适应二进制算术编码(CABAC)。

[0109] 熵解码模块210可以对关于由用于对视频进行编码的设备执行的帧内预测和帧间预测的信息进行解码。

[0110] 重排模块215可以基于在用于对视频进行编码的设备中使用的重排方法对由熵解

码模块210进行熵解码的比特流执行重排。重排模块可以将一维矢量形式的系数重构和重排成二维块形式的系数。重排模块215可以接收与在用于对视频进行编码的设备中执行的系数扫描有关的信息,并且可以经由基于在用于对视频进行编码的设备中执行的扫描顺序对系数进行逆扫描的方法来执行重排。

[0111] 逆量化模块220可以基于从用于对视频进行编码的设备接收的量化参数和重排后的块的系数来执行逆量化。

[0112] 逆变换模块225可以执行逆变换,即,逆DCT、逆DST和逆KLT,这是由变换模块对用于对视频进行编码的设备的量化结果执行的变换即DCT、DST和KLT的逆过程。可以基于用于对视频进行编码的设备所确定的变换单元来执行逆变换。用于对视频进行解码的设备的逆变换模块225可以依赖于多条信息例如预测方法、当前块的大小、预测方向等来选择性地执行变换方案(例如DCT、DST和KLT)。

[0113] 预测模块230和235可以基于从熵解码模块210接收到的关于预测块生成的信息和从存储器245接收到的先前解码的块或图片信息来生成预测块。

[0114] 如上所述,类似于用于对视频进行编码的设备的操作,在执行帧内预测时,在预测单元的大小与变换单元的大小相同时,可以基于位于预测单元的左侧、左上侧和上部的像素对预测单元执行帧内预测。在执行帧内预测时,在预测单元的大小与变换单元的大小不同时,可以使用基于变换单元的参考像素来执行帧内预测。此外,使用 $N \times N$ 划分的帧内预测可以仅用于最小编码单元。

[0115] 预测模块230和235可以包括预测单元确定模块、帧间预测模块以及帧内预测模块。预测单元确定模块可以从熵解码模块210接收各种信息,例如预测单元信息、帧内预测方法的预测模式信息、关于帧间预测方法的运动预测的信息等,可以将当前编码单元分割成预测单元,并且可以确定对预测单元执行帧间预测还是帧内预测。通过使用从用于对视频进行编码的设备接收的当前预测单元的帧间预测所需的信息,帧间预测模块230可以基于包括当前预测单元的当前图片的先前图片或后续图片中的至少之一的信息来对当前预测单元执行帧间预测。替选地,可以基于包括当前预测单元的当前图片中的一些预先重构区域的信息来执行帧间预测。

[0116] 为了执行帧间预测,可以针对编码单元来确定跳过模式、合并模式、AMVP模式和帧间块复制模式中的哪一个模式用作包括在编码单元中的预测单元的运动预测方法。

[0117] 帧内预测模块235可以基于当前图片中的像素信息来生成预测块。当预测单元是要进行帧内预测的预测单元时,可以基于从用于对视频进行编码的设备接收到的预测单元的帧内预测模式信息来执行帧内预测。帧内预测模块235可以包括自适应帧内平滑(AIS)滤波器、参考像素插值模块以及DC滤波器。AIS滤波器对当前块的参考像素执行滤波,并且可以依赖于当前预测单元的预测模式来确定是否应用滤波器。可以通过使用从用于对视频进行编码的设备接收到的预测单元的预测模式和AIS滤波器信息来对当前块的参考像素执行AIS滤波。在当前块的预测模式是不执行AIS滤波的模式时,可以不应用AIS滤波器。

[0118] 在预测单元的预测模式是基于通过对参考像素进行插值而获得的像素值来执行帧内预测的预测模式时,参考像素插值模块可以对参考像素进行插值以生成整数像素或小于整数像素的参考像素。在当前预测单元的预测模式是在不对参考像素进行插值的情况下生成预测块的预测模式时,可以不对参考像素进行插值。在当前块的预测模式是DC模式时,

DC滤波器可以通过滤波来生成预测块。

[0119] 可以将重构块或重构图片提供至滤波器模块240。滤波器模块240可以包括去块滤波器、偏移校正模块以及ALF。

[0120] 可以从用于对视频进行编码的设备接收关于是否将去块滤波器应用于相应的块或图片的信息以及关于在应用去块滤波器时应用强滤波器和弱滤波器中的哪个滤波器的信息。用于对视频进行解码的设备的去块滤波器可以从用于对视频进行编码的设备接收关于去块滤波器的信息,并且可以对相应的块执行去块滤波。

[0121] 偏移校正模块可以基于在执行编码时应用于图片的偏移值信息和偏移校正的类型来对重构图片执行偏移校正。

[0122] 可以基于从用于对视频进行编码的设备接收到的关于是否应用ALF的信息、ALF系数信息等来将ALF应用于编码单元。ALF信息可以被提供为被包括在特定参数集中。

[0123] 存储器245可以存储重构图片或重构块以用作参考图片或参考块,并且可以将重构图片提供至输出模块。

[0124] 如上面描述的,在本发明的实施方式中,为了便于说明,编码单元被用作表示用于编码的单元的术语,然而,编码单元可以用作执行解码以及编码的单元。

[0125] 另外,当前块可以表示要进行编码/解码的目标块。并且,依赖于编码/解码步骤,当前块可以表示编码树块(或编码树单元)、编码块(或编码单元)、变换块(或变换单元)、预测块(或预测单元)等。在本说明书中,术语“单元”可以表示用于执行特定编码/解码处理的基本单元,术语“块”可以表示预定大小的样本阵列。如果它们之间没有区别,则术语“块”和“单元”可用于具有相同的含义。例如,在下面描述的实施方式中,可以理解编码块和编码单元具有相互等同的含义。

[0126] 可以通过将图片分割成具有正方形或非正方形形状的基本块来对图片进行编码/解码。此时,基本块可以称作编码树单元。编码树单元可以被定义为序列或片内允许的最大大小的编码单元。可以通过序列参数集、图片参数集或片头(slice header)来用信号发送关于编码树单元是具有正方形形状还是具有非正方形形状的信息或者关于编码树单元的大小的信息。编码树单元可以被分割成更小大小的划分。此时,如果假设通过分割编码树单元而生成的划分深度是1,则通过分割具有深度1的划分而生成的划分的深度可以被定义为2。即,通过分割编码树单元中的深度k的划分而生成的划分可以被定义为具有深度k+1。

[0127] 图3是示出当通过帧内预测或帧间预测对编码块进行编码时可以应用于编码块的划分模式的图。通过分割编码树单元而生成的任意大小的划分可以被定义为编码单元。例如,在图3(a)中示出 $2N \times 2N$ 大小的编码单元。编码单元可以被递归地分割或分割成用于执行预测、量化、变换或环路滤波等的基本单元。例如,通过分割编码单元而生成的任意大小的划分可以被定义为编码单元,或者可以被定义为变换单元(TU)或预测单元(PU),其是用于执行预测、量化、变换或环路滤波等的基本单元。

[0128] 替选地,如果确定了编码块,则可以通过编码块的预测划分来确定与编码块具有相同的大小或大小小于编码块的预测块。可以通过指示编码块的划分类型的划分模式(Part_mode)来执行对编码块的预测划分。可以根据编码块的划分模式来确定预测块的大小或形状。可以通过指定划分候选中的任何一个的信息来确定编码块的划分类型。此时,取决于编码块的大小、形状、编码模式等,编码块可用的划分候选可以包括非对称划分类型

(例如 $nL \times 2N$ 、 $nR \times 2N$ 、 $2N \times nU$ 、 $2N \times nD$)。例如,可以根据当前块的编码模式来确定编码块可用的划分候选。例如,如图3(b)所示的示例中,当通过帧间预测对编码块进行编码时,可以将8个划分模式中的一个划分模式应用于编码块。另一方面,当通过帧间预测对编码块进行编码时,可以将图3(b)的8个划分模式中的PART_2N×2N或PART_N×N应用于编码块。

[0129] 在编码块具有最小大小的情况下,可以应用PART_N×N。此处,可以在编码器和解码器中预定义编码块的最小大小。替选地,可以经由比特流用信号发送关于编码块的最小大小的信息。例如,可以通过片头用信号发送编码块的最小大小,从而可以针对每个片定义编码块的最小大小。

[0130] 在另一个示例中,可以根据编码块的大小或形状中的至少之一来不同地确定编码块可用的划分候选。例如,可以根据编码块的大小或形状中的至少之一来不同地确定编码块可用的划分候选的数目或类型。

[0131] 替选地,可以根据编码块的大小或形状来限制编码块可用的划分候选中的非对称划分候选的类型或数目。例如,可以根据编码块的大小或形状中的至少之一来不同地确定编码块可用的非对称划分候选的数目或类型。

[0132] 通常,预测块可以具有从 64×64 到 4×4 的大小。然而,在通过帧间预测对编码块进行编码时,可以防止预测块具有 4×4 大小,以在执行运动补偿时减小存储器带宽。

[0133] 还可以使用划分模式来递归地分割编码块。也就是说,可以根据由划分索引指示的划分模式来分割编码块,并且可以将通过划分编码块生成的每个划分定义成编码块。

[0134] 在下文中,将更详细地描述递归地划分编码单元的方法。为了便于说明,假设编码树单元也包括在编码单元类别中。也就是说,在稍后描述的实施方式中,编码单元可以指编码树单元,或者可以指通过划分编码树单元而生成的编码单元。此外,在递归地分割编码块时,可以理解,通过划分编码块而生成的“划分”意指“编码块”。

[0135] 可以通过至少一条线来分割编码单元。此时,分割编码单元的线可以具有预定角度。此处,预定角度可以是0度至360度的范围内的值。例如,0度的线可以意指水平线,90度线可以意指垂直线,45度或135度线可以意指对角线。

[0136] 在通过多条线来分割编码单元时,多条线中的所有线都可以具有相同的角度。替选地,多条线中的至少一条线可以具有与其他线的角度不同的角度。替选地,分割编码树单元或编码单元的多条线可以被设置成具有预定义的角度差(例如,90度)。

[0137] 关于分割编码树单元或编码单元的线的信息可以被定义成划分模式并被编码。替选地,可以对关于线的数目、方向、角度、线在块中的位置等的信息进行编码。

[0138] 为了便于说明,在下面描述的实施方式中假设使用垂直线和水平线中的至少之一将编码树单元或编码单元分割成多个编码单元。

[0139] 当假设基于垂直线或水平线中的至少之一来执行对编码单元的划分,则划分编码单元的垂直线或水平线的数目可以是一个或更多个。例如,可以使用一条垂直线或一条水平线将编码树单元或编码单元分割成两个划分,或者可以使用两条垂直线或两条水平线将编码单元分割成三个划分。替选地,可以通过使用一条垂直线和一条水平线将编码单元划分成具有1/2的长度和宽度的四个划分。

[0140] 在使用至少一条垂直线或至少一条水平线将编码树单元或编码单元分割成多个划分时,划分可以具有统一的大小。替选地,任何一个划分可以与其余划分具有不同的大

小,或者每个划分可以具有不同的大小。

[0141] 在下面描述的实施方式中,假设将编码单元分割成四个划分是基于四叉树的划分,并且将编码单元分割成两个划分是基于二叉树的划分。另外,假设将编码单元分割成三个划分是基于三叉树的划分。另外,假设通过应用至少两个或更多个划分方案的划分方案是基于多叉树的划分。

[0142] 在以下附图中,将示出使用预定数量的垂直线或预定数量的水平线来分割编码单元,但是使用比示出的数目更多数目的垂直线或更多数目的水平线来将编码单元分割成比示出的划分更多的划分或更少的划分也在本发明的范围内。

[0143] 图4是示出根据本发明实施方式的允许四叉树划分和二叉树划分的划分类型的图。

[0144] 以预定的块单位对输入视频信号进行解码。用于对输入视频信号进行解码的这样的默认单位是编码块。编码块可以是执行帧内/帧间预测、变换以及量化的单元。另外,以编码块为单位确定预测模式(例如,帧内预测模式或帧间预测模式),并且包括在编码块中的预测块可以共享所确定的预测模式。编码块可以是具有在 8×8 至 64×64 范围内的任意大小的正方形块或非正方形块,或者可以是具有 128×128 、 256×256 或更大的大小的正方形块或非正方形块。

[0145] 具体地,可以基于四叉树划分和二叉树中的至少之一来对编码块进行分层划分。此处,基于四叉树的划分可以意指将 $2N \times 2N$ 的编码块划分成四个 $N \times N$ 的编码块(图4(a)),并且基于二叉树的划分可以意指将一个编码块划分成两个编码块。即使执行基于二叉树的划分,也可以在较低深度中存在正方形形状的编码块。

[0146] 可以对称地或者非对称地执行基于二叉树的划分。另外,基于二叉树划分的编码块可以是正方形块或非正方形块,例如矩形形状。例如,如图4(b)所示,允许基于二叉树的划分的划分类型可以是 $2N \times N$ (水平方向非正方形编码单元)或 $N \times 2N$ (垂直方向非正方形编码单元)的对称类型。另外,作为图4(c)中描绘的一个示例,允许基于二叉树的划分的划分类型可以是 $nL \times 2N$ 、 $nR \times 2N$ 、 $2N \times nU$ 或 $2N \times nD$ 的非对称类型。

[0147] 可以将基于二叉树的划分限制地允许为对称类型划分或非对称类型划分中的之一。在这种情况下,使用正方形块构造编码树单元可以与四叉树CU划分对应,并且使用对称非正方形块构造编码树单元可以与二叉树CU划分对应。使用正方形块和对称非正方形块构造编码树单元可以与四叉树CU划分和二叉树CU划分对应。

[0148] 在下文中,基于四叉树划分和二叉树的划分方案被称为四叉树和二叉树(QTBT)划分。

[0149] 作为基于四叉树和二叉树的划分的结果,不再被划分的编码块可以用作预测块或变换块。也就是说,在四叉树和二叉树(QTBT)划分方法中,编码块可以成为预测块,并且预测块可以成为变换块。例如,当使用QTBT划分方法时,可以以编码块为单位生成预测图像,并且以编码块为单位对作为原始图像于预测图像之间的差异的残差信号进行变换。此处,以编码块为单位生成预测图像可以意指基于编码块确定运动信息或者基于编码块确定帧内预测模式。因此,可以使用跳过模式、帧内预测或帧间预测中的至少之一来对编码块进行编码。

[0150] 作为另一示例,还可以划分编码块以便使用具有大小小于编码块的预测块或者变

换块。

[0151] 在QTBT划分方法中,可以将BT设置成仅针对对称划分被允许。然而,即使在块边界处分割对象和背景时仅允许对称二叉树,则编码效率可能降低。在本发明中,下面将描述作为另一实施方式的对编码块进行非对称划分以提高编码效率的方法。非对称二叉树划分表示将编码块划分成两个较小的编码块。作为非对称二叉树划分的结果,可以将编码块划分成非对称形状的两个编码块。

[0152] 可以对不再执行基于四叉树的划分的编码块执行基于二叉树的划分。可以不再对基于二叉树划分的编码块执行基于四叉树的划分。

[0153] 此外,可以依赖于较高深度的划分类型来确定较低深度的划分。例如,如果在两个或更多个深度中允许基于二叉树的划分,则可以在较低深度中仅允许与较高深度的二叉树划分相同的类型。例如,如果使用 $2N \times N$ 类型执行较高深度中的基于二叉树的划分,则也使用 $2N \times N$ 类型执行较低深度中的基于二叉树的划分。替选地,如果使用 $N \times 2N$ 类型执行较高深度中的基于二叉树的划分,则也使用 $N \times 2N$ 类型执行较低深度中的基于二叉树的划分。

[0154] 相比之下,也可以在较低深度中仅允许与较高深度的二叉树划分类型不同的类型。

[0155] 可以限制仅特定类型的基于二叉树的划分被用于序列、片、编码树单元或编码单元。作为示例,对于编码树单元,可以仅允许 $2N \times N$ 类型或 $N \times 2N$ 类型的基于二叉树的划分。可以在编码器或解码器中预定义可用划分类型。或者可以对关于可用划分类型的信息或关于不可用划分类型的信息进行编码,并且然后通过比特流用信号发送该信息。

[0156] 图5示出了根据应用本发明的实施方式的基于四叉树划分和二叉树划分来分层划分编码块的示例。

[0157] 如图5中所示,可以基于四叉树将划分深度(拆分深度)为 k 的第一编码块300划分成多个第二编码块。例如,第二编码块310至340可以是具有第一编码块的一半宽度和一半高度的正方形块,并且第二编码块的划分深度可以增加至 $k+1$ 。

[0158] 划分深度 $k+1$ 的第二编码块310可以被划分成划分深度 $k+2$ 的多个第三编码块。可以通过依赖于划分方法选择性地使用四叉树划分和二叉树中的之一来执行第二编码块310的划分。此处,可以基于指示基于四叉树划分的信息和指示基于二叉树划分的信息中的至少之一来确定划分方法。

[0159] 在基于四叉树对第二编码块310进行划分时,可以将第二编码块310划分成具有第二编码块的一半宽度和一半高度的四个第三编码块310a,并且第三编码块310a的划分深度可以增加至 $k+2$ 。相比之下,在基于二叉树对第二编码块310进行划分时,可以将第二编码块310划分成两个第三编码块。此处,两个第三编码块中的每一个可以是具有第二编码块的一半宽度和一半高度中的一者的非正方形块,并且划分深度可以增加至 $k+2$ 。可以依赖于划分方向将第二编码块确定为水平方向或垂直方向的非正方形块,并且可以基于关于沿垂直方向还是沿水平方向执行基于二叉树的划分的信息来确定划分方向。

[0160] 同时,第二编码块310可以被确定为不再基于四叉树或二叉树进行划分的叶编码块。在这种情况下,叶编码块可以被用作预测块或变换块。

[0161] 类似于第二编码块310的划分,第三编码块310a可以被确定为叶编码块,或者可以基于四叉树或二叉树被进一步划分。

[0162] 同时,可以基于二叉树将基于二叉树划分的第三编码块310b进一步划分成垂直方向的编码块310b-2或水平方向的编码块310b-3,并且相关编码块的划分深度可以增加至 $k+3$ 。替选地,第三编码块310b可以被确定为不再基于二叉树进行划分的叶编码块310b-1。在这种情况下,编码块310b-1可以被用作预测块或变换块。然而,可以基于以下信息中的至少之一来限制性地执行上述划分处理:关于允许基于四叉树的划分的编码块的大小/深度的信息、关于允许基于二叉树的划分的编码块的大小/深度的信息以及关于不允许基于二叉树的划分的编码块的大小/深度的信息。

[0163] 表示编码块的大小的候选的数量可以被限制为预定数量,或者预定单位中的编码块的大小可以具有固定值。作为示例,序列或图片中的编码块的大小可以被限制为具有 256×256 、 128×128 或 32×32 。可以通过序列头或图片头来用信号发送指示序列中或图片中的编码块的大小的信息。

[0164] 作为基于四叉树划分和二叉树的划分的结果,编码单元可以被表示为任意大小的正方形或矩形形状。

[0165] 图6示出了根据应用本发明的实施方式的基于四叉树划分和对称二叉树划分来分层地分割编码块的示例。

[0166] 图6示出了仅允许特定类型例如基于对称二叉树的划分的示例。图6(a)示出了仅限制地允许 $N \times 2N$ 类型中的基于二叉树的划分的示例。例如,深度1编码块601被分割成深度2的两个 $N \times 2N$ 块601a和601b,并且深度2编码块602被分割成深度3的两个 $N \times 2N$ 块602a和602b。

[0167] 图6(b)示出了仅限制地允许 $2N \times N$ 类型的基于二叉树的划分的示例。例如,深度1编码块603被分割成深度2的两个 $2N \times N$ 块603a和603b,并且深度2编码块604被分割成深度3的两个 $2N \times N$ 块604a和604b。

[0168] 图6(c)示出了对通过对称二叉树划分生成的块进行划分的示例。例如,深度1编码块605被分割成深度2的两个 $N \times 2N$ 块605a和605b,并且作为分割的结果而生成的深度2编码块605a被分割成两个 $N \times 2N$ 块605a1和605a2。上述分割方式也适用于通过对称二叉树划分生成的 $2N \times N$ 编码块。

[0169] 为了实现基于四叉树或二叉树的自适应划分,可以使用:指示基于四叉树的划分的信息、关于允许基于四叉树的划分的编码块的大小/深度的信息、指示基于二叉树的划分的信息、关于允许基于二叉树的划分的编码块的大小/深度的信息、关于不允许基于二叉树的划分的编码块的大小/深度的信息、是否沿垂直方向或水平方向执行基于二叉树的划分的信息等等。例如,quad_split_flag可以指示编码块是否被分割成四个编码块,并且binary_split_flag可以指示编码块是否被分割成两个编码块。在编码块被分割成两个编码块时,可以用信号发送指示编码块的划分方向是垂直方向还是水平方向的is_hor_split_flag。

[0170] 此外,对于编码树单元或预定编码单元,可以获得允许二叉树划分的次数、允许二叉树划分处的深度、或者允许二叉树划分的深度的数量。该信息可以以编码树单元或编码单元为单位进行编码,并且可以通过比特流传输至解码器。

[0171] 例如,可以通过比特流对指示在允许二叉树划分处的最大深度的语法“max_binary_depth_idx_minus1”进行编码/解码。在这种情况下,max_binary_depth_idx_

minus1+1可以指示在允许二叉树划分处的最大深度。

[0172] 此外,在上面描述的图6(c)的示例中,示出了与深度2编码单元(例如605a和605b)和深度3编码单元(例如605a1和605a2)有关的二叉树划分的结果。因此,可以通过比特流对以下信息中的至少之一进行编码/解码:指示已在编码树单元中执行的二叉树划分的次数(例如两次)的信息;指示在编码树单元中允许二叉树划分处的最大深度(例如深度3)的信息;或者指示允许二叉树划分的深度的数量(例如2,深度2和深度3)的信息。

[0173] 作为另一示例,针对每个序列或每个片可以获得允许二叉树划分的次数、允许二叉树划分处的深度或者允许二叉树划分的深度的数量中的至少之一。例如,该信息可以以序列、图片或者片为单位进行编码,并且通过比特流来传输。因此,第一片和第二片可以在以下方面中的至少之一不同:执行二叉树划分的次数、允许二叉树划分处的最大深度或者允许二叉树划分的深度的数量。例如,在第一片中,仅在一个深度处允许二叉树划分,而在第二片中,仅在深度二处允许二叉树划分。

[0174] 作为另一示例,可以根据片或图片的时间水平标识符(Temporal_ID)不同地设置允许二叉树划分的次数、允许二叉树划分处的深度或允许二叉树划分的深度的数量中的至少之一。此处,时间水平标识符(Temporal_ID)用于识别具有视图、空间、时间或图像质量中的至少之一的可扩展性的多个视频层中的每一个。

[0175] 还可以限制针对通过二叉树划分来划分的CU使用变换跳过。替选地,可以仅针对通过非正方形划分来划分的CU沿水平方向或垂直方向中的至少之一应用变换跳过。仅沿水平方向应用变换跳过可以意指仅沿水平方向执行缩放和量化而不沿水平方向执行变换,并且通过指定诸如DCT或DST的至少一种变换方案来沿垂直方向执行变换。

[0176] 类似地,仅沿垂直方向应用变换跳过可以意指通过指定诸如DCT或DST的至少一种变换方案来沿水平方向执行变换,并且仅沿垂直方向执行缩放和量化而不沿垂直方向执行变换。还可以用信号发送指示是否沿水平方向应用变换跳过的语法hor_transform_skip_flag和指示是否沿垂直方向应用变换跳过的语法ver_transform_skip_flag。

[0177] 当将变换跳过应用于水平方向或者垂直方向中的至少之一时,可以根据CU的形状用信号发送指示应用变换跳过的方向的信息。具体地,例如,对于 $2N \times N$ 形状的CU,沿水平方向执行变换并且可以沿垂直方向应用变换跳过,以及对于 $N \times 2N$ 形状的CU,可以沿水平方向应用变换跳过并且沿垂直方向执行变换。此处,变换可以是DCT或DST中的至少之一。

[0178] 作为另一示例,对于 $2N \times N$ 形状的CU,沿垂直方向执行变换并且可以沿水平方向应用变换跳过,以及对于 $N \times 2N$ 形状的CU,可以沿垂直方向应用变换跳过并且沿水平方向执行变换。此处,变换可以是DCT或DST中的至少之一。

[0179] 图7是示出作为应用本发明的实施方式的允许非对称二叉树划分的划分类型的图。 $2N \times 2N$ 的编码块可以被分割成宽度比为 $n:(1-n)$ 的两个编码块或者高度比为 $n:(1-n)$ 的两个编码块。其中 n 可以表示大于0且小于1的实数。

[0180] 例如,在图7中示出通过将非对称二叉树划分应用于编码块来生成宽度比为1:3的两个编码块701、702或者宽度比为3:1的两个编码块703、704,高度比为1:3的两个编码块705、706,高度比为3:1的两个编码块708、709。

[0181] 具体地,当沿垂直方向划分 $W \times H$ 大小的编码块时,可以生成宽度为 $1/4W$ 的左侧划分和宽度为 $3/4W$ 的右侧划分。如上所述,其中左侧划分的宽度小于右侧划分的宽度的划分

类型可以被称作 $nL \times 2N$ 二进制划分。

[0182] 当沿垂直方向划分 $W \times H$ 大小的编码块时,可以生成宽度为 $3/4W$ 的左侧划分和宽度为 $1/4W$ 的右侧划分。如上所述,其中右侧划分的宽度小于左侧划分的宽度的划分类型可以被称作 $nR \times 2N$ 二进制划分。

[0183] 当沿水平方向划分 $W \times H$ 大小的编码块时,可以生成高度为 $1/4H$ 的上部划分和高度为 $3/4H$ 的下部划分。如上所述,其中上部划分的高度小于下部划分的高度的划分类型可以被称作 $2N \times nU$ 二进制划分。

[0184] 当沿水平方向划分 $W \times H$ 大小的编码块时,可以生成高度为 $3/4H$ 的上部划分和高度为 $1/4H$ 的下部划分。如上所述,其中下部划分的高度小于上部划分的高度的划分类型可以被称作 $2N \times nD$ 二进制划分。

[0185] 在图7中,示出了两个编码块之间的宽度比或高度比是1:3或3:1。然而,通过非对称二叉树划分生成的两个编码块之间的宽度比或高度比不限于此。编码块可以被划分成具有与图7中示出的宽度比或高度比不同的宽度比或不同高度比的两个编码块。

[0186] 在使用非对称二叉树划分时,可以基于经由比特流用信号发送的信息来确定编码块的非对称二进制划分类型。例如,可以基于指示编码块的划分方向的信息和指示通过分割编码块生成的第一划分是否具有比第二划分的大小更小的大小的信息来确定编码块的划分类型。

[0187] 指示编码块的划分方向的信息可以是指示编码块是沿垂直方向被划分还是沿水平方向被划分的1比特的标志。例如,hor_binary_flag可以指示编码块是否沿水平方向被划分。如果hor_binary_flag的值是1,则其可以指示编码块沿水平方向被划分,并且如果hor_binary_flag的值是0,则其可以指示编码块沿垂直方向被划分。替选地,可以使用指示编码块是否沿垂直方向被划分的ver_binary_flag。

[0188] 指示第一划分是否具有比第二划分的大小更小的大小的信息可以是1比特的标志。例如,is_left_above_small_part_flag可以指示通过分割编码块而生成的左侧划分或上部划分的大小是否小于右侧划分或下部划分的大小。如果is_left_above_small_part_flag的值为1,则其意味着左侧划分或上部划分的大小小于右侧划分或下部划分的大小。如果is_left_above_small_part_flag的值为0,则其意味着左侧划分或上部划分的大小大于右侧划分或下部划分的大小。替选地,可以使用指示右侧划分或下部划分的大小是否小于左侧划分或上部划分的大小的is_right_bottom_small_part_flag。

[0189] 替选地,可以通过使用指示第一划分与第二划分之间的宽度比、高度比或面积比的信息来确定第一划分的大小和第二划分的大小。

[0190] 在hor_binary_flag的值为0且is_left_above_small_part_flag的值为1时,其可以表示 $nL \times 2N$ 二进制划分,并且在hor_binary_flag的值为0且is_left_above_small_part_flag的值为0时,其可以表示 $nR \times 2N$ 二进制划分。另外,在hor_binary_flag的值为1且is_left_above_small_part_flag的值为1时,其可以表示 $2N \times nU$ 二进制划分,并且在hor_binary_flag的值为1且is_left_above_small_part_flag的值为0时,其可以表示 $2N \times nD$ 二进制划分。

[0191] 作为另一示例,编码块的非对称二进制划分类型可以通过指示编码块的划分类型的索引信息来确定。此处,索引信息是要通过比特流用信号发送的信息,并且可以以固定长

度(即,固定的比特数)编码,或者可以以可变长度编码。例如,下表1示出了针对每个非对称二进制划分的划分索引。

[0192] [表1]

	非对称划分索引	二进制化
$nL \times 2N$	0	0
$nR \times 2N$	1	10
$2N \times nU$	2	100
$2N \times nD$	3	111

[0194] 可以根据QTBT划分方法使用非对称二叉树划分。例如,如果四叉树划分或二叉树划分不再适用于编码块,则可以确定是否将非对称二叉树划分应用于编码块。此处,可以通过经由比特流用信号发送的信息来确定是否将非对称二叉树划分应用于编码块。例如,该信息可以是1比特标志'asymmetric_binary_tree_flag',并且基于该标志,可以确定是否要将非对称二叉树划分应用于编码块。替选地,在确定编码块被划分成两个块时,可以确定划分类型是二叉树划分还是非对称二叉树划分。此处,可以通过经由比特流用信号发送的信息来确定编码块的划分类型是二叉树划分还是非对称二叉树划分。例如,该信息可以是1比特标志'is_asymmetric_split_flag',并且基于该标志,可以确定编码块要被划分成对称形式还是非对称形式。作为另一示例,分配给对称类型二进制划分和非对称类型二进制划分的索引可以是不同的,并且可以基于索引信息来确定编码块要以对称类型还是非对称类型划分。例如,表2示出了其中将不同的索引分配给对称二进制类型划分和非对称二进制类型划分的示例。

[0195] [表2]

	二进制划分索引	二进制化
$2N \times N$ (沿水平方向 的二进制划分)	0	0
$N \times 2N$ (沿垂直方向 的二进制划分)	1	10
$nL \times 2N$	2	110
$nR \times 2N$	3	1110
$2N \times nU$	4	11110
$2N \times nD$	5	11111

[0197] 可以通过四叉树划分、二叉树划分或非对称二叉树划分将编码块或编码块分割成多个编码块。例如,图8示出了其中使用QTBT和非对称二叉树划分将编码块分割成多个编码块的示例。参照图9,可以看出,分别在第一个图中的深度2划分、第二个图中的深度3划分和第三个图中的深度3划分中执行非对称二叉树划分。可以限制不再对通过非对称二叉树划分而分割的编码块进行分割。例如,对于通过非对称二叉树划分生成的编码块,可以不对

与二叉树、二叉树或非对称二叉树有关的信息进行编码/解码。也就是说,对于通过非对称二叉树划分生成的编码块,可以省略指示是否应用二叉树划分的标志、指示是否应用非对称二叉树划分的标志、指示是否应用非对称二叉树划分的标志、指示非对称二叉树划分的方向、或者指示非对称二进制划分的索引信息等。作为另一示例,可以根据是否允许QTBT来确定是否允许二叉树划分。例如,在不使用基于QTBT的划分方法的图片或片中,可以限制不使用非对称二叉树划分。

[0198] 可以以块、片或图片为单位对指示是否允许非对称二叉树划分的信息进行编码并用信号发送该信息。此处,指示是否允许非对称二叉树划分的信息可以是1比特的标志。例如,如果is_used_asymmetric_QTBT_enabled_flag的值是0,则其可以指示不使用非对称二叉树划分。在在图片或片中不使用二叉树划分的情况下,也可以将is_used_asymmetric_QTBT_enabled_Flag设置成0而不用信号进行发送。

[0199] 图8示出了作为应用本发明的实施方式的基于二叉树划分和对称/非对称二叉树划分的编码块的划分类型。

[0200] 图8(a)示出了允许基于 $nL \times 2N$ 类型的非对称二叉树划分的示例。例如,深度1编码块801被分割成深度2的两个非对称 $nL \times 2N$ 块801a和801b,并且深度2编码块801b也被分割成深度3的两个对称 $N \times 2N$ 块801b1和801b2。

[0201] 图8(b)示出了允许基于 $nR \times 2N$ 类型的非对称二叉树划分的示例。例如,深度2编码块802被分割成深度3的两个非对称 $nR \times 2N$ 块802a和802b。

[0202] 图8(c)示出了允许基于 $2N \times nU$ 类型的非对称二叉树划分的示例。例如,深度2编码块803被分割成深度3的两个非对称 $2N \times nU$ 块803a和803b。

[0203] 还可以基于编码块的大小、形状、划分深度或划分类型来确定编码块中允许的划分类型。例如,在通过二叉树划分生成的编码块和通过非对称二叉树划分生成的编码块中允许的划分类型、划分形状或划分数目中的至少之一可以彼此不同。

[0204] 例如,如果通过二叉树划分来生成编码块,则对于编码块可以允许所有二叉树划分、非对称二叉树划分和对称二叉树划分。也就是说,如果基于二叉树划分来生成编码块,则可以将图10中示出的所有划分类型应用于编码块。例如, $2N \times 2N$ 划分可以表示编码块不被进一步分割的情况, $N \times N$ 可以表示用二叉树划分编码块的情况,以及 $N \times 2N$ 和 $2N \times N$ 可以表示用非对称二叉树划分编码块的情况。另外, $nL \times 2N$ 、 $nR \times 2N$ 、 $2N \times nU$ 和 $2N \times nD$ 可以表示用非对称二叉树划分编码块的情况。

[0205] 另一方面,在通过非对称二叉树划分生成编码块时,对于编码块可能不允许使用非对称二叉树划分。也就是说,在基于非对称二叉树划分生成编码块时,可以限制不将在图7示出的划分类型中的非对称划分类型($nL \times 2N$ 、 $nR \times 2N$ 、 $2N \times nU$ 、 $2N \times nD$)应用于编码块。

[0206] 图9是示出作为应用本发明的实施方式的基于二叉树划分和非对称二叉树划分的编码块划分方法的流程图。

[0207] 假设深度 k 编码块被分割成深度 $k+1$ 编码块。首先,确定是否将二叉树划分应用于深度 k 的当前块S910。如果应用二叉树划分,则将当前块划分成四个方形块S920。另一方面,如果未应用二叉树划分,则确定是否将非对称二叉树划分应用于当前块S930。如果未应用非对称二叉树划分,则当前块变为深度 $k+1$ 编码块而不进行分割。作为S930的确定结果,如果对当前块应用非对称二叉树划分,则检查是否应用对称二进制划分或非对称二进制划分S940。根据S940的确定

定结果,确定应用于当前块的划分类型S950。例如,应用于步骤S950的划分类型可以是图4(b)中的对称类型中的任何之一或者图4(c)中的非对称类型之一。根据在S950、S960处确定的划分类型,将当前块分割成两个深度k+1的编码块。

[0208] 作为应用本发明的实施方式,图10示出了包括在应用了四叉树划分和二叉树划分的网络抽象层(NAL)中的语法元素。

[0209] 应用了本发明的压缩图像可以以网络抽象层(以下称为“NAL”)为单位进行打包,并通过传输介质进行传输。然而,本发明不限于NAL,而是可以应用于将来待开发的各种数据传输方案。例如,应用本发明的NAL单元可以包括视频参数集(VPS)、序列参数集(SPS)、图片参数集(PPS)和至少一个片集(片),如图10所示。

[0210] 例如,在图10中示出了包括在序列参数集(SPS)中的语法元素,但是也可以使图片参数集(PPS)或者片集(片)包括语法元素。另外,要共同应用于序列单元或图片单元的语法元素可以包括在序列参数集(SPS)中或者图片参数集(PPS)中。另一方面,仅应用于片的语法元素优选地包括在片集(片)中。因此,可以考虑编码性能和编码效率来选择。

[0211] 在这方面,应用四叉树划分和二叉树划分的语法元素如下。可以将图10中所示的所有语法元素设置成基本元素,但考虑到编码效率和编码性能,还可以选择性地将所有语法元素中的一些语法元素设置成基本元素。

[0212] 例如,‘quad_split_flag’指示编码块是否被分割成四个编码块。‘binary_split_flag’可以指示编码块是否被划分成两个编码块。当编码块被分割成两个编码块时,可以用信号发送指示编码块的划分方向是垂直还是水平的“is_hor_split_flag”。可以定义当“is_hor_split_flag=1”时表示水平方向,以及当“is_hor_split_flag=0”时表示垂直方向。

[0213] 在另一个替选中,可以通过‘isUseBinaryTreeFlag’表示是否将二叉树划分应用于当前块,并且作为表示编码块的划分方向的语法元素的“hor_binary_flag”可以表示当前块是否是沿水平方向划分。例如,当“hor_binary_flag=1”时,这可以指示编码块沿水平方向划分,并且当“hor_binary_flag=0”时,它可以指示编码块沿垂直方向划分。或者,可以以相同的方式使用指示是否沿垂直方向划分编码块的ver_binary_flag而不是“hor_binary_flag”。

[0214] 另外,‘max_binary_depth_idx_minus1’可以被定义为指示允许二叉树划分的最大深度的语法元素。例如,“max_binary_depth_idx_minus1+1”可以指示允许二叉树划分的最大深度。

[0215] 另外,‘ver_transform_skip_flag’可以被设置成指示是否沿水平方向应用变换跳过的语法元素,并且‘hor_transform_skip_flag’可以被设置成指示是否沿垂直方向应用变换跳过的语法元素。

[0216] 另外,‘is_used_asymmetric_QTBT_enabled_flag’可以被定义为指示是否允许非对称二叉树划分的语法元素。例如,当“is_used_asymmetric_QTBT_enabled_flag=1”时,它可以指示使用非对称二叉树划分,并且当“is_used_asymmetric_QTBT_enabled_flag=0”时,它可以指示不使用非对称二叉树划分。另一方面,当在图片单元中或者片单元中不使用二叉树划分时,可以将is_used_asymmetric_QTBT_enabled_flag的值设置成0而不用信号发送它。替选地,可以通过‘asymmetric_binary_tree_flag’来指示是否将非对称二叉树

划分应用于当前块。

[0217] 另外,作为指示非对称二叉树划分的语法元素,'is_left_above_small_part_flag'可以指示通过划分编码块而生成的左侧划分或者上部划分的大小是否小于右侧划分或者下部划分的大小。例如,如果“is_left_above_small_part_flag=1”,则可能意指左侧划分或者上部划分的大小小于右侧划分或者下部划分的大小,并且如果“is_left_above_small_part_flag=0”,则可能意指左侧划分或者上部划分的大小大于右侧划分或者下部划分的大小。替选地,可以使用指示右侧划分或者下部划分的大小是否小于左侧划分或者上部划分的'is_right_bottom_small_part_flag'而不是“is_left_above_small_part_flag”。

[0218] 在这方面,可以组合语法元素以定义编码块的非对称二进制划分类型。例如,当“hor_binary_flag=0”且“is_left_above_small_part_flag=1”时,它可以表示 $nL \times 2N$ 二进制划分,并且当“hor_binary_flag=0”且“is_left_above_small_part_flag=0”时,它可以表示 $nR \times 2N$ 二进制划分。另外,当“hor_binary_flag=1”且“is_left_above_small_part_flag=1”时,它可以表示 $2N \times nU$ 二进制划分,并且当“hor_binary_flag=1”和“is_left_above_small_part_flag=0”时,它可以表示 $2N \times nD$ 二进制划分。类似地,非对称二进制划分类型可以通过'ver_binary_flag'与'is_right_bottom_small_part_flag'的组合来表示。

[0219] 在另一个替选方案中,编码块的非对称二进制划分类型可以通过用'Asymmetric_partition_index'指示上述表1中的索引来定义,或者可以通过用'Binary_partition_index'指示上述表2中的索引来定义。

[0220] 如以上示例中所描述的,可以通过至少一条垂直线或水平线来递归地分割编码单元(或编码树单元)。例如,可以概括为,四叉树划分是使用水平线和垂直线来分割编码块的方法,并且二叉树划分是使用水平线或垂直线来分割编码块的方法。基于四叉树划分和二叉树划分的编码块的划分类型不限于图4至图8中示出的示例,并且可以使用除示出的类型之外的扩展划分类型。也就是说,可以用与图4至图8中示出的类型不同的类型来递归地分割编码块。

[0221] 图11是示出作为应用本发明的另一实施方式的允许非对称四叉树划分的划分类型的图。

[0222] 在当前块被四叉树划分的情况下,水平线或垂直线中的至少之一可以非对称地分割编码块。此处,非对称可以意指通过水平线分割的块的高度不相同,或者通过垂直线分割的块的宽度不相同。例如,水平线可以将编码块分割成非对称形状而垂直线将编码块分割成对称形状,或者水平线可以将编码块分割成对称形状而垂直线将编码块分割成非对称形状。替选地,水平线和垂直线二者都可以非对称地分割编码块。

[0223] 图11(a)示出了编码块的对称划分类型,并且图11(b)至图11(k)示出了编码块的非对称四叉树划分类型。图11(a)示出了水平线和垂直线二者均用于对称划分的示例。图11(b)和图11(c)示出了水平线用于对称划分而垂直线用于非对称划分的示例。图11(d)和图11(e)示出了垂直线用于对称划分而水平线用于非对称划分的示例。

[0224] 为了指定编码块的划分类型,可以对与编码块的划分类型有关的信息进行编码。此处,该信息可以包括指示编码块的划分类型是对称还是非对称的第一指示符。该第一指

示符可以以块为单位进行编码,或者可以针对每条垂直线或每条水平线对该第一指示符进行编码。例如,第一指示符可以包括指示是否将垂直线用于对称划分的信息和指示是否将水平线用于对称划分的信息。

[0225] 备选地,可以仅针对垂直线或水平线来对第一指示符进行编码,并且可以通过第一指示符依赖地得到针对其第一指示符未被编码的另一条线的划分类型。例如,针对其第一指示符未被编码的另一条线的划分类型可以具有与第一指示符的值相反的值。也就是说,如果第一指示符指示将垂直线用于非对称划分,则与第一指示符相反地可以设置使用水平线进行对称划分。

[0226] 在第一指示符指示非对称划分的情况下,还可以针对垂直线或水平线进一步对第二指示符进行编码。此处,第二指示符可以指示以下中的至少之一:用于非对称划分的垂直线或水平线的位置或者通过垂直线或水平线分割的块之间的比率。

[0227] 可以使用多条垂直线或多条水平线来执行二叉树划分。例如,还可以通过对一条或更多条垂直线或者一条或更多条水平线中的至少之一进行组合来将编码块分割成四个块。

[0228] 图11(f)至图11(k)是示出通过组合多条垂直线/水平线和一条水平线/垂直线来非对称地划分编码块的示例的图。

[0229] 参照图11(f)至图(k),通过以下方式来执行二叉树划分:通过两条垂直线或两条水平线将编码块分割成三个块,然后将三个经分割的块中的一个分割成两个块。此时,如图11(f)至图(k)示出的示例,可以通过水平线或垂直线来对通过两条垂直线或两条水平线分割的块中的位于中央的块进行分割。还可以通过使用水平线或垂直线来分割位于编码块一侧的块。备选地,可以通过比特流用信号发送用于指定三个划分中要被分割的划分的信息(例如,划分索引)。

[0230] 可以使用水平线或垂直线中的至少之一来非对称地分割编码块,并且可以使用另一条线来对称地分割编码块。例如,可以使用多条垂直线或多条水平线将编码块分割成对称形状,或者可以使用一条水平线或一条垂直线将编码块分割成对称形状。备选地,水平线和垂直线都可以用于将编码块分割成对称形状,或者可以用于将编码块分割成非对称形状。

[0231] 例如,图11(f)示出了由使用两条垂直线的非对称划分生成的中间处的编码块通过水平线被分割成两个对称类型编码块的划分类型。另外,图11(g)示出了由使用两条水平线的非对称划分生成的中间处的编码块通过垂直线被分割成两个对称类型编码块的划分类型。

[0232] 另一方面,图11(h)和图11(i)示出了由使用两条垂直线的非对称划分生成的中间处的编码块通过水平线再次被分割成两个非对称编码块的划分类型。另外,图11(j)和图11(k)示出了由使用两条水平线的非对称划分生成的中间处的编码块通过垂直线再次被分割成两个非对称编码块的划分类型。

[0233] 在组合多条垂直线/水平线和一条水平线/一条垂直线时,编码块可以被分割成由至少两个不同大小组成的四个划分(即,四个编码块)。将编码块分割成具有至少两个不同大小的四个划分的方法可以被称作三种类型的非对称二叉树划分(三型非对称二叉树CU划分)。

[0234] 可以基于上述第一指示符或第二指示符中的至少之一来对关于三型非对称二叉树划分的信息进行编码。例如,第一指示符可以指示编码块的划分类型是对称的还是非对称的。第一指示符可以以块为单位被编码,或者可以分别针对垂直线或水平线对第一指示符进行编码。例如,第一指示符可以包括指示是否将一条或更多条垂直线用于对称划分的信息以及指示是否将一条或更多条水平线用于对称划分的信息。

[0235] 替代地,可以仅针对垂直线或水平线来对第一指示符进行编码,并且可以通过第一指示符得到针对其第一指示符未被编码的另一条线的划分类型。

[0236] 在第一指示符指示非对称划分的情况下,还可以针对垂直线或水平线进一步对第二指示符进行编码。此处,第二指示符可以指示以下中的至少之一:用于非对称划分的垂直线或水平线的位置或者通过垂直线或水平线分割的块之间的比率。

[0237] 图12是示出作为应用本发明的另一实施方式的基于非对称二叉树划分的编码块划分方法的流程图。

[0238] 假设深度k编码块被分割成深度k+1编码块。首先,确定是否将二叉树划分应用于深度k的当前块S1210。作为步骤S1210的确定的结果,如果确定未应用二叉树划分,则当前块变为深度k+1编码块而不进行分割。如果在步骤S1210中确定应用了二叉树划分,则确定是否将非对称二叉树划分应用于当前块S1220。如果不应用非对称二叉树划分而应用对称二叉树划分,则将当前块划分成四个方形块S1230。

[0239] 另一方面,如果应用非对称二叉树划分,则确定是否将三种类型的非对称二叉树划分应用于当前块S1240。如果不应用三种类型的非对称二叉树划分,则将当前块分割成两种类型的四个非对称块S1250。在这种情况下,根据划分信息,当前块可以通过图11(b)至图11(e)中的任何一个划分类型来划分。

[0240] 另一方面,如果应用三种类型的非对称二叉树划分,则将当前块分割成三种类型的四个非对称块S1260。在这种情况下,根据划分信息,当前块可以通过图11(f)至图11(k)中的划分类型的任何一者来划分。

[0241] 图13示出了作为应用本发明的另一实施方式的在应用非对称二叉树划分的网络抽象层(NAL)中包括的语法元素。应用本发明的NAL单元可以包括例如视频参数集(VPS)、序列参数集(SPS)、图片参数集(PPS)和至少一个片集(片)。

[0242] 例如,在图13中示出了包括在序列参数集(SPS)中的语法元素,但是也可以使图片参数集(PPS)或者片集(片)包括语法元素。另外,要共同应用于序列单元或图片单元的语法元素可以包括在序列参数集(SPS)中或者图片参数集(PPS)中。另一方面,仅应用于片的语法元素优选地包括在片集(片)中。因此,可以考虑编码性能和编码效率来选择。

[0243] 语法元素'Is_used_asymmetric_quad_tree_flag'指示是否非对称地执行二叉树划分。另外,'Is_used_triple_asymmetric_quad_tree_flag'指示是否以三种类型非对称地执行二叉树划分。因此,当“Is_used_asymmetric_quad_tree_flag=0”时,意指对称地执行二叉树划分,因此,不用信号发送'Is_used_triple_asymmetric_quad_tree_flag'。另一方面,当“Is_used_asymmetric_quad_tree_flag=1”且“Is_used_triple_asymmetric_quad_tree_flag=1”时,意指执行三种类型的非对称二叉树划分。另外,当“Is_used_asymmetric_quad_tree_flag=1”且“Is_used_triple_asymmetric_quad_tree_flag=0”时,意指执行两种类型的非对称二叉树划分。

[0244] 语法元素'hor_asymmetric_flag'指示非对称二叉树划分的方向。也就是说,当“Is_used_asymmetric_quad_tree_flag=1”时,可以指示是沿水平方向还是沿垂直方向执行非对称划分。例如,当“hor_asymmetric_flag=1”时,可以指示沿水平方向的非对称划分,并且当“hor_asymmetric_flag=0”时,可以指示沿垂直方向的非对称划分。替代地,也可以使用'ver_asymmetric_flag'。

[0245] 语法元素'width_left_asymmetric_flag'指示非对称二叉树划分的另一方向。也就是说,当“Is_used_asymmetric_quad_tree_flag=1”时,可以指示基于宽度是沿左侧方向还是沿右侧方向执行非对称划分。例如,当“width_left_asymmetric_flag'=1”时,可以指示基于宽度对左侧方向执行非对称划分,并且当“width_left_asymmetric_flag=0”时,可以指示基于宽度对右侧方向执行非对称划分。

[0246] 另外,语法元素“height_top_asymmetric_flag”指示非对称二叉树划分的另一方向。也就是说,当“Is_used_asymmetric_quad_tree_flag=1”时,可以指示基于高度是沿上部方向还是沿下部方向执行非对称划分。例如,当“height_top_asymmetric_flag=1”时,可以指示基于高度对上部方向执行非对称划分,并且当“height_top_asymmetric_flag=0”时,它可以指示基于高度对下部方向执行非对称划分。

[0247] 另外,语法元素'is_used_symmetric_line_flag'指示当应用三种类型的二叉树划分时,中间处的块是否是对称的。也就是说,当“Is_used_asymmetric_quad_tree_flag=1”且“Is_used_triple_asymmetric_quad_tree_flag=1”时,指示中间处的块被对称地划分。

[0248] 因此,通过语法元素的组合,可以表示图11(a)至图11(k)中所示的划分类型。例如,当“Is_used_asymmetric_quad_tree_flag=0”时,意味着将块划分成如图11(a)的划分类型所示的4个对称块。

[0249] 另外,当“Is_used_asymmetric_quad_tree_flag=1”且“Is_used_triple_asymmetric_quad_tree_flag=0”时,可以意指图11(b)至图11(e)的划分类型中的任何一者。在这种情况下,当“hor_asymmetric_flag=1”且“width_left_asymmetric_flag'=1”时,意指图11(b)的划分类型。另外,当“hor_asymmetric_flag=1”且“width_left_asymmetric_flag'=0”时,意指图11(c)的划分类型。另外,当“hor_asymmetric_flag=0”且“height_top_asymmetric_flag'=1”时,意指图11(d)的划分类型。另外,当“hor_asymmetric_flag=0”且“height_top_asymmetric_flag'=0”时,意指图11(e)的划分类型。

[0250] 另外,当“Is_used_asymmetric_quad_tree_flag=1”且“Is_used_triple_asymmetric_quad_tree_flag=1”时,意指图11(f)至图11(k)的划分类型中的任何一个。在这种情况下,当“is_used_symmetric_line_flag=1”时,意指图11(f)和图11(g)的任何一种划分类型。并且当“is_used_symmetric_line_flag=0”时,意指图11(h)至图11(k)的划分类型中的任何一个。另外,当“is_used_symmetric_line_flag=1”且“hor_asymmetric_flag=1”时,可以将其定义为图11(f)的划分类型,并且当“hor_asymmetric_flag=0”时,可以将其定义为图11(g)的划分类型。

[0251] 另外,当“Is_used_asymmetric_quad_tree_flag=1”、“Is_used_triple_asymmetric_quad_tree_flag=1”且“is_used_symmetric_line_flag=0”时,可以通过

“hor_asymmetric_flag”、“width_left_asymmetric_flag”和“height_top_asymmetric_flag”来定义划分类型。例如,当“hor_asymmetric_flag=1”且“height_top_asymmetric_flag=0”时,意指图11(h)的划分类型。另外,当“hor_asymmetric_flag=1”且“height_top_asymmetric_flag=1”时,意指图11(i)的划分类型。另外,当“hor_asymmetric_flag=0”且“width_left_asymmetric_flag=0”时,意指图11(j)的划分类型。另外,当“hor_asymmetric_flag=0”且“width_left_asymmetric_flag=1”时,意指图11(k)的划分类型。

[0252] 另外,作为另一替选方案,图11(a)至图11(k)的每个划分类型可以由‘asymmetric_quadtrees_partition_index’表示为索引。

[0253] 图14是示出作为应用本发明的另一实施方式的允许四叉树划分和三叉树划分的划分类型被允许的图。

[0254] 可以基于四叉树和三叉树中的至少之一来分层地分割编码块。此处,基于四叉树的划分意指将 $2N \times 2N$ 编码块分割成四个 $N \times N$ 编码块的方式(图14(a)),并且基于三叉树的划分意指将编码块分割成三个编码块的方式。即使执行基于三叉树的划分,也可能存在较低深度的正方形编码块。

[0255] 可以对称地执行(图14(b))或者可以非对称地执行(图14(c))基于三叉树的划分。另外,基于三叉树划分的编码块可以是正方形块或者非正方形块例如矩形。例如,如图14(b)所示的示例,允许基于三叉树划分的划分类型是对称且具有相同宽度的 $2N \times (2N/3)$ (水平非正方形编码单元)或者对称且具有相同高度的 $(2N/3) \times 2N$ (垂直非正方形编码单元)。另外,作为示例,允许基于三叉树的划分的划分类型可以是包括具有不同宽度或高度的编码块的非对称划分类型,如图14(c)示出的示例中所示。例如,在根据图14(c)的非对称三叉树划分类型中,至少两个编码块1401、1403可以位于两侧并且具有相同的宽度(或高度) k ,并且块的其余部分1402可以位于具有相同大小的块1401、1404之间,并且可能具有宽度 $2k$ 。

[0256] 在这方面,将CTU或CU分割成如图14所示的具有非正方形形状的三个子划分的方法被称为三叉树划分方法(三叉树CU分割)。可以限制由三重树划分分割的CU另外被划分。

[0257] 图15是示出作为应用本发明的另一实施方式的基于四叉树划分和三叉树划分的编码块划分方法的流程图。

[0258] 假设深度 k 编码块被分割成深度 $k+1$ 编码块。首先,确定在深度 k 处是否将四叉树划分应用于当前块S1510。如果应用四叉树划分,则将当前块划分成四个正方形块S1520。另一方面,如果未应用四叉树划分,则确定是否将三叉树划分应用于当前块S1530。如果未应用三叉树划分,则当前块变为深度 $k+1$ 编码块而不进行划分。

[0259] 作为S1530的确定的结果,如果将三叉树划分应用于当前块,则检查是否应用对称三划分或非对称三划分中的任一个S1540。根据S1540的确定结果确定应用于当前块的划分类型S1550。例如,在步骤S1550应用的划分类型可以是在对称的情况下图14的任何一种类型,或者可以是在非对称的情况下图14(c)的任何一种类型。根据在步骤S1550处确定的划分类型,当前块被分割成深度 $k+1$ 的三个编码块S1560。

[0260] 图16示出了作为应用本发明的另一实施方式,在应用四叉树划分和三叉树划分的网络抽象层(NAL)中包括的语法元素。应用本发明的NAL单元可以包括例如视频参数集(VPS)、序列参数集(SPS)、图片参数集(PPS)和至少一个片集(片)。

[0261] 例如,在图16中示出了包括在序列参数集(SPS)中的语法元素,但是也可以使图片

参数集(PPS)或者片集(片)包括语法元素。另外,要共同应用于序列单元或图片单元的语法元素可以包括在序列参数集(SPS)中或者图片参数集(PPS)中。另一方面,仅应用于片的语法元素优选地包括在片集(片)中。因此,可以考虑编码性能和编码效率来选择。

[0262] 语法元素“quad_split_flag”指示编码块是否被分割成四个编码块。’triple_split_flag’可以指示编码块是否被分割成三个编码块。当编码块被分割成三个编码块时,可以用信号发送指示编码块的划分方向是垂直还是水平的“is_hor_split_flag”。当“is_hor_split_flag=1”时,可以意指水平方向,而当“is_hor_split_flag=0”时,可以意指垂直方向。

[0263] 另外,作为另一替换方案,还可以通过’isUseTripleTreeFlag’指示是否将三叉树划分应用于当前块,并且通过语法元素’triple_split_flag’指示是否沿水平方向划分编码块。例如,当“hor_triple_flag=1”时,可以指示沿水平方向划分编码块,并且当“hor_triple_flag=0”时,可以指示沿垂直方向划分编码块。替换地,代替“hor_triple_flag”,可以以相同的方式使用指示是否沿垂直方向划分编码块的ver_triple_flag。

[0264] 另外,可以将’asymmetric_triple_tree_flag’定义为指示是否允许非对称三叉树划分的语法元素。例如,当“asymmetric_triple_tree_flag=1”时,指示使用非对称三叉树划分,而当“asymmetric_triple_tree_flag=0”时,指示不使用非对称三叉树划分。另一方面,当在图片单元或者片单元中不使用三叉树划分时,可以将值设置成0而不用信号发送’asymmetric_triple_tree_flag’。

[0265] 因此,通过语法元素的组合,可以表示图14(a)至图14(c)中所示的划分类型。例如,如果“isUseTripleTreeFlag=0”,则意指将块划分成如图14(a)的划分类型中示出的四个对称块。

[0266] 另外,当“isUseTripleTreeFlag=1”且“asymmetric_triple_tree_flag=0”时,可以意指图14(b)的划分类型中的任何一者。在这种情况下,当“hor_triple_flag=1”时,可以意指图14(b)的 $(2N/3) \times 2N$ 划分类型并且当“hor_triple_flag=0”时,可以意指图14(b)的 $2N \times (2N/3)$ 划分类型。

[0267] 另外,当“isUseTripleTreeFlag=1”且“asymmetric_triple_tree_flag=1”时,可以意指图14(c)的划分类型中的任何一者。此时,当“hor_triple_flag=1”时,可以意指图14(c)的左侧所示出的划分类型并且当“hor_triple_flag=0”时,可以意指图14(c)的右侧所示出的划分类型。

[0268] 另外,作为另一替换方案,通过’asymmetric_tripletree_partition_index’可以将图14(a)至图14(c)的每个划分类型表示为索引。

[0269] 图17是示出根据本发明另一实施方式的允许多叉树划分的划分类型的图。

[0270] 使用上述四叉树划分、二叉树划分或者三叉树划分中的至少之一来划分CTU或者CU的方法可以称为多叉树划分(或者多叉树CU划分)。可以使用上述示例中的任何N个划分来划分CTU或者CU。具体地,例如,如图17所示,可以使用9种划分类型来划分CTU或者CU。

[0271] 对于序列或图片的单元,可以通过使用所有四叉树划分、二叉树划分以及三叉树划分来执行划分,或者可以通过使用四叉树划分、二叉树划分或者三叉树划分中的一个或两个来执行划分。

[0272] 还可以作为默认使用四叉树划分,并且选择性地使用二叉树划分和三叉树划分。

此时,可以通过序列参数集或者图片参数集来用信号发送是否使用二叉树划分和/或三叉树划分。

[0273] 替代地,还可以作为默认使用四叉树划分和三叉树划分,并选择性地使用二叉树划分。例如,可以在序列头中用信号发送指示是否使用二叉树划分的语法 `isUseBinaryTreeFlag`。如果 `isUseBinaryTreeFlag` 的值为1,则可以使用二叉树划分对当前序列中的CTU或CU进行划分。还可以通过序列头用信号发送指示是否使用了三重树划分的语法 `isUseTripleTreeFlag`。如果 `isUseTripleTreeFlag` 的值为1,则可以使用三叉树划分来对当前序列头中的CTU或CU进行划分。

[0274] 通过多叉树划分划分的划分形状可以限制于例如在17(a)至图17(i)中所示的9个基本划分。图17(a)示出四叉树划分类型,图17(b)至图17(c)示出对称二叉树划分类型,图17(d)至图17(e)示出三叉树划分类型,并且图17(f)至图17(i)示出非对称二叉树划分类型。省略了关于图17中所示出的每个划分类型的详细描述,因为它们与上述划分类型完全相同。

[0275] 另外,作为另一替代方案,可以扩展通过多叉树划分分割的划分类型以进一步包括图18(j)至图18(u)中所示出的12个划分。图18(j)至图18(m)示出了非对称四叉树划分类型,图(n)至图(s)示出了三种类型的非对称四叉树划分的划分类型,并且(t)至(u)示出了对称三叉树的划分类型划分。图18中所示的每个划分类型与上述划分类型相同,因此,将省略其的详细描述。

[0276] 图19是示出作为应用本发明的另一实施方式的基于多叉树划分的编码块划分方法的流程图。

[0277] 假设深度k编码块被分割成深度k+1编码块。首先,确定是否将四叉树划分应用于深度k的当前块S1910。如果未应用四叉树划分,则确定是否将二叉树划分应用于当前块S1950。另外,如果未应用二叉树划分,则确定是否将三叉树划分应用于当前块S1990。作为S1950处的确定的结果,如果未应用三叉树划分,则当前块变为深度k+1编码块而不进行划分。

[0278] 此处,作为步骤S1910的确定的结果,如果应用四叉树划分,则检查是否执行对称四叉树划分或非对称四叉树划分S1920。此后,检查划分信息以确定当前块的块划分类型S1930,并且根据确定的划分类型将当前块分割成四个块S1940。例如,当应用对称四叉树时,块被分割成图17(a)的划分类型。另外,当应用非对称四叉树时,块被分割成图18(j)至图18(m)的划分类型的任何之一。替代地,当应用三种类型的非对称四叉树划分时,块被分割成图18(n)至图18(s)的划分类型的任何之一。然而,如上所述,如果仅图17的基本划分类型可用作多叉树划分类型,则在不确定四叉树是否是非对称的情况下,只可以应用图17(a)的对称正方形块。

[0279] 另外,作为步骤S1950的确定的结果,如果应用二叉树划分,则检查是否执行对称二叉树划分或者非对称二叉树划分S1960。此后,通过检查划分信息来确定当前块的块划分类型S1970,并且根据所确定的划分类型将当前块分割成两个块S1980。例如,当应用对称二叉树时,块被分成图17(b)和图17(c)的划分类型的任何之一。另外,当应用非对称二叉树时,块被分割成图17(f)至图17(i)的划分类型的任何之一。

[0280] 另外,作为步骤S1990的确定的结果,如果应用三叉树划分,则检查是否执行对称

三叉树划分或者非对称三叉树划分S1960。此后,通过检查划分信息来确定当前块的块划分类型S1970,并且根据所确定的划分类型将当前块分割成三个块S1980。例如,当应用非对称三叉树时,块被分成图17(d)和图17(e)的划分类型的任何之一。另外,当应用对称二叉树时,块被分割成图18(t)至图18(u)的划分类型的任何之一。然而,如上所述,如果仅图17的基本划分类型可用于多叉树划分类型,则仅可以应用17(d)和17(e)中的预定义的非对称三元块,而无需确定三叉树是否是非对称。

[0281] 作为表示多叉树划分的语法元素,可以定义指示是否执行多叉树划分的'is_used_Multitree_flag'。另外,可以使用参考图10、图13和图16示出和描述的语法元素作为用于确定多叉树划分类型的信息。

[0282] 例如,可以通过合并多个邻近编码单元来生成合并编码单元。此时,多个编码单元中的任何一个将被称为合并候选编码单元。合并候选编码单元可以意指在多个编码单元中的扫描顺序中具有第一顺序的编码单元。或者,可以意指多个编码单元中位于特定方向的编码单元。特定方向可以意指最左侧、最上部或中心位置。

[0283] 参照图21,示出了应用上述多叉树划分的9种类型的CU,并且每个CU可以分别被称为 $CU_{2N \times 2N}$ 、 $CU_{N \times 2N}$ 、 $CU_{2N \times N}$ 、 $CU_{nL \times 2N \times nR}$ 、 $CU_{nU \times 2N \times nD}$ 、 $CU_{nL \times 2N}$ 、 $CU_{nR \times 2N}$ 、 $CU_{2N \times nU}$ 和 $CU_{2N \times nD}$ 。就此而言,可以选择包括9种类型的CU中的至少三种划分的编码单元(例如 $CU_{2N \times 2N}$ 、 $CU_{nL \times 2N \times nR}$ 和 $CU_{nU \times 2N \times nD}$)的划分类型作为合并目标编码单元。另一方面,作为另一替换方案,可以选择包括CU的类型中的至少 n ($n \geq 2$)个编码单元的划分类型作为合并目标编码单元。

[0284] 接下来,参照图20,使用合并候选编码单元生成合并编码单元S2020。

[0285] 例如,对于通过上述的三叉树划分生成的 $CU_{nL \times 2N \times nR}$,与其垂直邻近的三个编码单元 CU_0 、 CU_1 和 CU_2 中的任何之一可以是合并候选编码单元。在这方面,图22(a)示出了从 $CU_{nL \times 2N \times nR}$ 编码单元生成两种类型的合并编码单元的处理。可以通过对构成 $CU_{nL \times 2N \times nR}$ 的三个 CU_0 、 CU_1 和 CU_2 中的 CU_0 和 CU_1 进行合并或者通过合并 CU_1 和 CU_2 来生成合并编码单元(合并CU)。

[0286] 同时,可以用信号发送指定要合并的编码单元的信息。可以以与要合并的编码单元的位置有关的标志或索引的形式对信息进行编码。例如,语法元素CuMerge_idx可以如下定义以区分合并的编码单元(合并的CU)。也就是说,当语法元素CuMerge_idx值为'0'时,定义对划分编码单元中的按编码顺序的前两个编码单元进行合并。并且当语法元素CuMerge_idx值为'1'时,定义对按编码顺序的最后两个编码单元进行合并。

[0287] 另外,例如,对于通过上述的三叉树划分生成的 $CU_{nU \times 2N \times nD}$,与其水平邻近的三个编码单元(CU_3 、 CU_4 、 CU_5)中的任何之一可以是合并候选编码单元。在这方面,图22(b)示出了从 $CU_{nU \times 2N \times nD}$ 编码单元生成两种类型的合并编码单元的处理。通过对构成 $CU_{nU \times 2N \times nD}$ 的三个 CU_3 、 CU_4 和 CU_5 中的 CU_3 和 CU_4 进行合并,或者通过合并 CU_4 和 CU_5 来生成合并编码单元(合并CU)。另外,如上所述,为了区分要生成的合并CU,当对划分编码单元中的按编码顺序的前两个编码单元进行合并时,语法元素CuMerge_idx的值可以被设置成0,并且当对按编码顺序的最后两个编码单元进行合并时,语法元素CuMerge_idx的值可以被设置成1。

[0288] 最后,参照图20,对生成的合并编码单元执行编码或解码S2030。例如,合并候选编

码单元编码参数,运动数据以及/或者纹理数据可以用在合并编码单元的编码/解码处理(即预测、变换、量化等)中。例如,合并候选编码单元的运动矢量、参考图片索引、帧内模式、帧内参考样本等可以用作合并编码单元的运动矢量、参考图片索引、帧内模式、帧内参考样本等。

[0289] 图23是示出使用语法元素CuMerge_idx的值的视频编码方法或视频解码方法的流程图。

[0290] 首先,检查当前编码块是否通过三叉树划分被分割成三个编码单元S2200。如果被分割成三个编码单元,则检查CuMerge_idx的值S2210。当“CuMerge_idx=0”时,通过合并编码块的按编码顺序的前两个编码单元来生成合并的编码单元S2220。另一方面,当“CuMerge_idx=1”时,通过合并编码块的按编码顺序的最后两个编码单元来生成合并的编码单元S2230。

[0291] 然而,本发明不限于图23中所示的示例。就是说,例如,在通过四叉树划分而分割的4个CU的情况下,可以通过另外定义CuMerge_idx的值来区分合并编码单元(合并的CU)。例如,当“CuMerge_idx=0”时,可以定义对按编码顺序的前两个编码单元进行合并,当“CuMerge_idx=1”时,可以定义对按编码顺序的第一编码单元和第三编码单元进行合并,并且当“CuMerge_idx=2”时,可以定义对按编码顺序的第二编码单元和第四编码单元进行合并。

[0292] 表3和表4示出了确定应用如图21所示的多叉树划分(四叉树划分、二叉树划分和三叉树划分)的编码单元的类型语法元素。

[0293] 除了语法元素CuMerge_idx之外,可以用信号发送另一语法元素CU_merge_flag。如果Cu_merge_flag的值是'1',则指示邻近编码块与当前编码块被合并。此处,合并可以意指通过合并多个编码单元来生成新类型的编码单元的处理。替选地,可以将其视为通过合并由多叉树划分划分的多个编码单元中的一些编码单元来重新确定划分类型的处理。

[0294] 因此,通过使用语法元素CuMerge_idx和CU_merge_flag,可以区分和定义通过多叉树划分的编码块的划分类型。例如,将用下面表3或表4中所示的码字详细给出下面的描述。

[0295] 首先,通过应用如图22所示的合并编码单元方案,可以将三叉树划分编码单元CU_{nL}×2N×nR和CU_{nU}×2N×nD重新生成成为CU_{nL}×2N、CU_{nR}×2N、CU_{2N}×nU或CU_{2N}×nD的编码单元。在更普遍的意义,可以通过使用来自垂直方向三叉树划分的编码单元合并方案来进行非正方形垂直方向二叉树划分(例如图22(a))。另外,可以通过使用来自水平方向三叉树划分的编码单元合并方案来进行非正方形水平方向二叉树划分(例如图22(b))。

[0296] 也就是说,编码单元CU_{nL}×2N, CU_{nR}×2N, CU_{2N}×nU和CU_{2N}×nD具有与通过将合并方案应用至编码单元CU_{nL}×2N×nR或CU_{nU}×2N×nD而生成的编码单元相同的形状。因此,可以通过使用指示是否被合并的CU_merge_flag和指示合并类型或方向的CuMerge_idx来区分编码单元。

[0297] 例如,参照表3和表4, CU_{nL}×2N和CU_{nR}×2N编码单元的码字(例如表3-'0001'和表4-'001')被设置成与CU_{nL}×2N×nR的码字(例如表3-'0001'和表4-'001')相同,并且通过上述CuMerge_idx和Cu_merge_flag的值来区分。类似地, CU_{2N}×nU和CU_{2N}×nD编码单元的码字(例如表3-'0000'、表4-'000')被设置成与CU_{nU}×2N×nD的码字(例如表3-'

0000'、表4-'000')相同,但通过上述CuMerge_idx和Cu_merge_flag的值来区分。因此,通过有效地利用码字,可以提高编码效率和解码效率。

[0298] [表3]

划分索引	CU 划分	码字	Cu_merge_flag	CuMerge_idx
0	CU_2N×2N	1	-	-
1	CU_N×2N	01	-	-
2	CU_2N×N	001	-	-
3	CU_nL×2N×nR	0001	0	-
4	CU_nU×2N×nD	0000	0	-
5	CU_nL×2N	0001	1	1
6	CU_nR×2N	0001	1	0
7	CU_2N×nU	0000	1	1
8	CU_2N×nD	0000	1	0

[0301] [表4]

[0302]

划分索引	CU划分	码字	Cu_merge_flag	CuMerge_idx
0	CU_2N×2N	1	-	-
1	CU_N×2N	011	-	-
2	CU_2N×N	010	-	-
3	CU_nL×2N×nR	001	0	-
4	CU_nU×2N×nD	000	0	-
5	CU_nL×2N	001	1	1
6	CU_nR×2N	001	1	0
7	CU_2N×nU	000	1	1
8	CU_2N×nD	000	1	0

[0303] 图24是示出作为应用本发明的实施方式的帧间预测方法的流程图。图24中包括了确定当前块的运动信息S2410,并且通过使用所确定的运动信息执行当前块的运动补偿S2420的步骤。在下文中,将详细描述这些步骤。

[0304] 首先,参照图24,可以确定当前块的运动信息S2410。当前块的运动信息可以包括当前块的运动矢量、当前块的参考图片索引或者当前块的帧间预测方向中的至少之一。

[0305] 可以基于通过比特流用信号发送的信息或者与当前块邻近的邻近块的运动信息中的至少之一来获得当前块的运动信息。

[0306] 就此而言,图25是示出当将合并模式应用于当前块时得到当前块的运动信息的过程的图。另一方面,图26是示出在当将高级运动矢量预测器 (AMVP) 模式应用于当前块时得到当前块的运动信息的过程的图。

[0307] 首先,将参照图25描述将合并模式应用于当前块的情况。可以从当前块的空间邻近块得到空间合并候选S2510。空间邻近块可以包括与当前块的上部、左侧或者角(例如左上角、右上角或者左下角中的至少之一)邻近的块中的至少之一。

[0308] 可以将空间合并候选的运动信息设置成与空间邻近块的运动信息相同。

[0309] 可以从当前块的时间邻近块得到时间合并候选S2520。时间邻近块可以意指包括在同位图片中的共同定位块。同位图片具有与包括当前块的当前图片不同的时间顺序(图片顺序计数,POC)。可以将参考图片列表中具有预定义索引的图片确定为同位图片,或者由通过比特流用信号发送的索引来确定同位图片。可以将包括在具有与同位图片中的当前块相同的位置和大小的块中的块或者与具有与同位图片中的当前块相同的位置和大小的块邻近的块确定为时间邻近块。例如,可以将包括具有与同位图片中的当前块相同的位置和大小的块的中心坐标的块或者与块的右下边界邻近的块中的至少之一确定为时间邻近块。

[0310] 可以基于时间邻近块的运动信息来确定时间合并候选的运动信息。例如,可以基于时间邻近块的运动矢量来确定时间合并候选的运动矢量。另外,可以将时间合并候选的帧间预测方向设置成与时间邻近块的帧间预测方向相同。然而,时间合并候选的参考图片索引可具有固定值。例如,时间合并候选的参考图片索引可以被设置成'0'。

[0311] 此后,可以生成包括空间合并候选和时间合并候选的合并候选列表S2530。如果合并候选列表中包括的合并候选的数量小于合并候选的最大数量,则通过组合两个或更多个合并候选生成的组合合并候选或者具有零运动矢量(0,0)的合并候选可以包括在合并候选列表中。

[0312] 当生成合并候选列表时,可以基于合并候选索引来指定合并候选列表中包括的合并候选中的至少之一S2540。

[0313] 可以将当前块的运动信息设置成与由合并候选索引指定的合并候选的运动信息相同S2550。例如,当通过合并候选索引选择空间合并候选时,可以将当前块的运动信息设置成与空间邻近块的运动信息相同。替选地,当通过合并候选索引选择时间合并候选时,可以将当前块的运动信息设置成与时间邻近块的运动信息相同。

[0314] 另一方面,将参照图26描述将AMVP模式应用于当前块的情况。根据比特流,可以对当前块的帧间预测方向或者参考图片索引中的至少之一进行解码S2610。也就是说,当应用AMVP模式时,可以基于在比特流中编码的信息来确定当前块的帧间预测方向或参考图片索引中的至少之一。

[0315] 可以基于当前块的空间邻近块的运动矢量来确定空间运动矢量候选S2620。空间运动矢量候选可以包括从当前块的上部邻近块得到的第一空间运动矢量候选和从当前块

的左侧邻近块得到的第二空间运动矢量候选中的至少之一。此处,上部邻近块可以包括与当前块的上角或者右上角邻近的块中的至少之一,并且当前块的左侧邻近块可以包括与当前块的左角或者左下角邻近的块中的至少之一。与当前块的左上角邻近的块可以被视为上部邻近块,或者可以被视为左邻近块。

[0316] 如果当前块的参考图片与空间邻近块的参考图片不同,则可通过缩放空间邻近块的运动矢量来获得空间运动矢量。

[0317] 可基于当前块的时间邻近块的运动矢量来确定时间运动矢量候选S2630。如果当前块的参考图片与时间邻近块的参考图片不同,则可以通过缩放时间邻近块的运动矢量来获得时间运动矢量。

[0318] 可以生成包括空间运动矢量候选和时间运动矢量候选的运动矢量候选列表S2640。

[0319] 当生成运动矢量候选列表时,可以基于指定运动矢量候选列表中的至少之一的信息来指定运动矢量候选列表中包括的运动矢量候选中的至少之一S2650。

[0320] 可以将由信息指定的运动矢量候选设置成当前块的运动矢量预测值,然后将运动矢量差值添加至运动矢量预测值,以获得当前块的运动矢量S2660。在这种情况下,可以从比特流解析运动矢量差值。

[0321] 返回参照图24,当获得当前块的运动信息时,可以基于获得的运动信息执行对当前块的运动补偿S2420。具体地,可以基于当前块的帧间预测方向、参考图片索引和运动矢量来执行对当前块的运动补偿。

[0322] 如在以上示例中,基于当前块的运动信息,可以执行对当前块的运动补偿。在这种情况下,运动矢量可以具有整数像素单位或十进制(decimal)像素单位的精度(或分辨率)。

[0323] 整数像素单位可以包括N个整数像素,例如一个整数像素、两个整数像素、四个整数像素等。此处,N是1或者更大的自然数,特别地,它可以由2的指数幂表示。一个整数像素可以表示一个像素的精度(即一个像素单位),两个整数像素可以表示两个像素的精度(即两个像素单位),以及四个整数像素可以表示四个像素的精度(即四个像素单位)。根据所选择的整数像素,可以以N个像素为单位表示运动矢量,并且可以以N个像素为单位执行运动补偿。

[0324] 十进制像素单位可包括1/N像素,例如二分之一像素、四分之一像素、八分之一像素等。此处,N是1或者更大的自然数,特别地,它可以由2的指数幂表示。二分之一像素可以表示1/2像素的精度(即1/2像素单位),四分之一像素可以表示1/4像素的精度(即1/4像素单位),以及八分之一像素可以表示1/8像素的精度(即1/8像素单位)。根据所选择的十进制像素,可以以1/N像素为单位表示运动矢量,并且可以以1/N像素为单位执行运动补偿。

[0325] 图27和图28是示出根据当前块的运动矢量精度的运动矢量推导方法的图。图27示出了AMVP模式下的运动矢量推导方法。图28示出了合并模式下的运动矢量推导方法。

[0326] 首先,可以确定当前块的运动矢量精度S2710、S2810。

[0327] 可以以序列、图片、片或者块为单位确定运动矢量精度。此处,块可以指示CTU、CU、PU或者具有预定大小/形状的块。CTU可以意指具有编码器/解码器允许的最大大小的CU。当在比诸如序列、图片或片的块更高级别处确定运动矢量精度时,可以根据在更高级别处确定的运动矢量精度来执行对块的运动补偿。例如,使用以整数像素为单位的运动矢量来执

行对包括在第一片中的块的运动补偿,而使用以四分之一像素为单位的运动矢量来执行对包括在第二片中的块的运动补偿。

[0328] 为了确定运动矢量的精度,可以经由比特流用信号发送用于确定运动矢量的精度的信息。该信息可以是用于识别多个运动矢量精度中的至少之一的索引信息'mv_resolution_idx'。例如,表5示出了根据mv_resolution_idx的运动矢量精度。

[0329] [表5]

[0330]	mv_resolution_idx	运动矢量像素单位
	0	四分之一像素像素单位
	1	二分之一像素像素单位
	2	整数像素像素单位
[0331]	3	八分之一像素像素单位
	4	两个整数像素像素单位
	5	四个整数像素像素单位

[0332] 表5仅是可以应用本发明的示例。可以具有预定单位的运动矢量精度候选的类型和/或数量可以与表5中所示的那些不同。mv_resolution_idx的值和/或范围也可以根据运动矢量精度候选的类型和/或数量而不同。作为另一示例,可以从与预定单元空间上邻近或者时间上邻近的单元得到运动矢量精度。此处,预定单元可以指示图片、片、块等,并且邻近单元可以指示与预定单元空间上邻近或者时间上邻近的图片、片、块等。例如,可以将当前块的运动矢量精度设置成与当前块的空间邻近块和/或时间邻近块中的索引信息所指示的块的运动矢量精度相等。

[0333] 作为另一示例,可以根据当前块的运动信息自适应地确定当前块的运动矢量精度。例如,可以根据以下几项自适应地确定当前块的运动矢量精度:当前块的参考图片的时间顺序或者输出顺序(POC)是否在当前图片的时间顺序或输出顺序(POC)之前、当前块的参考图片的时间顺序或者输出顺序是否晚于当前图片的时间顺序或输出顺序(POC)、或者当前块的参考图片是否是当前图片。

[0334] 可以选择性地使用多个运动矢量精度候选的一部分。例如,在定义包括至少一个运动矢量精度候选的运动矢量精度集之后,可以将包括在运动矢量精度集中的运动矢量精度候选中的至少之一确定为运动矢量精度。

[0335] 可以以序列、片或块为单位确定运动矢量精度集。可以在编码器和解码器中对包括在运动矢量精度集中的运动矢量精度候选进行预定义。替选地,可以基于通过比特流用信号发送的编码信息来确定运动矢量精度集。此处,编码信息可以与运动矢量精度集中包

括的运动矢量精度候选的类型和/或数量中的至少之一相关。作为另一示例,可以从与预定单元空间上或者时间上邻近的单元得到运动矢量精度集。此处,预定单元可以指示图片、片、块等,并且邻近单元可以指示与预定单元在空间上或者时间上邻近的图片,片,块等。例如,可以将预定片的运动矢量精度集设置成等于与片空间上邻近的片的运动矢量精度集。替选地,取决于片之间的依赖性,可以将独立片的运动矢量精度集设置成从属片的运动矢量精度集。

[0336] 一旦确定了运动矢量精度集,就可以将包括在运动矢量精度集中的至少一个运动矢量精度候选确定为运动矢量精度。为此,可以通过比特流用信号发送识别包括在运动矢量精度集中的运动矢量精度候选中的至少之一的索引信息。例如,可以将包括在运动矢量精度集中的运动矢量精度候选中的由索引信息识别的候选设置成当前块的运动矢量精度。

[0337] 可以根据片类型、当前块的大小/形状、当前块的运动信息(例如当前块的参考图片或者前块的预测方向)等自适应地确定是否使用运动矢量精度集。替选地,可以经由比特流用信号发送指示是否使用运动矢量精度集的信息(例如标志)。

[0338] 当在高于诸如序列、图片或片的块的更高级别处确定运动矢量精度集时,可以根据在更高级别处确定的运动矢量精度集得到预定块的运动矢量精度。例如,如果在图片级别处定义包括四分之一像素和两个整数像素的运动矢量集,则可以限制该图片中包括的块使用四分之一像素或者两个整数像素中的至少之一。

[0339] 当将多方向预测应用于当前块时,根据多方向预测的多个运动矢量可具有不同的运动矢量精度。也就是说,当前块的多个运动矢量中的任何一个的精度可以与另一个运动矢量的精度不同。例如,当双向预测应用于当前块时,前向运动矢量mvL0的精度可以与后向运动矢量mvL1的精度不同。即使当具有多于三个方向的预测被应用于当前块时,多个运动矢量中的至少之一可具有与另一运动矢量不同的精度。因此,可以针对当前块的每个预测方向对用于确定运动矢量精度的信息进行编码/解码。

[0340] 当将AMVP模式应用于当前块并且针对每个块可变地确定运动矢量精度时,可能发生从邻近块得到的运动矢量预测值(运动矢量预测器,MVP)的运动矢量精度与当前块的运动矢量精度不同。为了使运动矢量预测值的运动矢量精度与当前块的运动矢量精度匹配,可以根据当前块的运动矢量精度来缩放运动矢量预测值S2720。可以缩放运动矢量预测值以使其与当前块的运动矢量精度匹配。可以将运动矢量差值(MVD)添加至缩放的运动矢量预测值,以得到当前块的运动矢量S2730。

[0341] 例如,当邻近块的运动矢量像素单位是四分之一像素并且当前块的运动矢量像素单位是整数像素时,从邻近块得到的运动矢量预测值可以按整数像素单位缩放,并且然后,可以通过将缩放的运动矢量预测值与运动矢量差值相加来获得整数像素单位的运动矢量。例如,下面的式1示出了通过按整数像素单位缩放运动矢量预测值来获得运动矢量的示例。

[0342] [式1]

$$[0343] \quad mvLX[0] = ((mvpLX[0] \gg 2) + mvdLX[0]) \ll 2$$

$$[0344] \quad mvLX[1] = ((mvpLX[1] \gg 2) + mvdLX[1]) \ll 2$$

[0345] 在式1中,mvpLX表示运动矢量预测值,并且mvdLX表示运动矢量差值。另外,mvLX[0]、mvpLX[0]和mvdLX[0]表示沿垂直方向的运动矢量分量,mvLX[1]、mvpLX[1]和mvdLX[1]表示沿水平方向的运动矢量分量。

[0346] 作为另一示例,当邻近块的运动矢量像素单位是两个整数像素并且当前块的运动矢量像素单位是四分之一像素时,可以按四分之一像素单位缩放从邻近块得到的运动矢量预测值,并且可以通过将缩放的运动矢量预测值与运动矢量差值相加来获得四分之一像素单位的运动矢量。例如,下面的式2示出了当将当前图片用作参考图片时获得运动矢量的示例。

[0347] [式2]

[0348] $mvLX[0] = (mvpLX[0] \gg 3 + mvdLX[0]) \ll 3$

[0349] $mvLX[1] = (mvpLX[1] \gg 3 + mvdLX[1]) \ll 3$

[0350] 在式1和式2中,可以根据当前块的运动矢量精度与邻近块的运动矢量精度之间的放大率(magnification)自适应地确定用于缩放运动矢量预测值的比特移位值。

[0351] 不同于图27中所示的示例,还可以对通过运动矢量预测值与运动矢量差值相加而生成的运动矢量进行缩放,以使其与当前块的运动矢量精度匹配。

[0352] 可以根据当前块的运动矢量精度对运动矢量差值进行编码/解码。例如,在当前块的运动矢量精度是四分之一像素时,可以以四分之一像素单位对当前块的运动矢量差值进行编码/解码。

[0353] 无论当前块的运动矢量精度如何,都还可以以预定单位对运动矢量差值进行编码/解码。此处,预定单位可以是由编码器和解码器预先约定的固定像素单位(例如整数像素或者四分之一像素),或者是在较高级别例如图片或片处确定的像素单位。如果当前块的运动矢量精度与运动矢量差值的运动矢量精度不同,则可以通过缩放运动矢量差值,或者通过对由缩放的运动矢量预测值与运动矢量差值相加得到的运动矢量进行缩放来得到当前块的运动矢量。例如,如果当前块的运动矢量精度是整数像素,而运动矢量差值是四分之一像素精度,则可以通过对由缩放的运动矢量预测值与运动矢量差值相加得到的运动矢量进行缩放来得到当前块的运动矢量,如式1所示。

[0354] 根据运动矢量精度,可以不同地确定编码/解码运动矢量差值的方法。例如,在十进制像素像素单位的情况下,运动矢量前缀部分可以表示运动矢量的整数部分,并且后缀部分可以表示运动矢量的分数部分。例如,下面的式3示出了得到前缀部分'prefix_mvd'和后缀部分'suffix_mvd'的示例。

[0355] [式3]

[0356] $prefix_mvd = MVD / N$

[0357] $suffix_mvd = MVD \% N$

[0358] 在式3中,N可以是固定值,或者可以是根据当前块的运动矢量精度可变地确定的值。例如,N可以是与当前块的运动矢量精度成比例的值。

[0359] 在当前块的运动矢量精度是两个或更多个整数像素单位时,可以对通过将运动矢量差值移位N获得的值进行编码。例如,如果当前块的运动矢量精度是两个整数像素,则可以对运动矢量差值的1/2进行编码/解码。如果当前块的运动矢量精度是4个整数像素,则可以对运动矢量差值的1/4进行编码/解码。在这种情况下,解码器可以通过根据当前块的运动矢量精度缩放解码的运动矢量差值来得到当前块的运动矢量。

[0360] 当将合并模式或跳过模式应用于当前块并且针对每个块可变地确定运动矢量精度时,可能发生空间/时间合并候选块的运动矢量精度与当前块的运动矢量精度不同。因

此,可以根据当前块的运动矢量精度来缩放空间/时间邻近块的运动矢量S2820,并且可以将缩放的运动矢量设置成空间/时间合并候选的运动信息S2830。例如,通过对空间/时间邻近块的运动矢量mvLX[0]和/或mvLX[1]缩放得到缩放的运动矢量mxLXscale[0]和/或mvLXscale[1],并且然后可以将缩放的运动矢量设置成空间/时间合并候选的运动矢量。

[0361] 例如,当与当前块邻近的邻近块的运动矢量精度是四分之一像素并且当前块的运动矢量精度是整数像素时,通过缩放邻近块的运动矢量来设置空间邻近候选的运动矢量,如下面的式4所示。

[0362] [式4]

[0363] $mvLXscale[0] = (mvLX[0] \gg 2) \ll 2$

[0364] $mvLXscale[1] = (mvLX[1] \gg 2) \ll 2$

[0365] 在式4中,可以根据当前块的运动矢量精度与邻近块的运动矢量精度之间的放大率自适应地确定用于缩放邻近块的运动矢量的比特移位值。

[0366] 作为另一示例,可以选择要与当前块合并的合并候选(即,由合并索引选择的合并候选),并且可以检查所选择的合并候选的运动矢量精度是否对应于当前块的运动矢量精度。当所选择的合并候选的运动矢量精度与当前块的运动矢量精度不匹配时,可以根据当前块的运动矢量精度来缩放所选择的合并候选的运动矢量。

[0367] 可以将当前块的运动矢量设置成与合并候选中的由索引信息选择的合并候选的运动矢量(即,缩放的运动矢量)相同S2840。

[0368] 不同于图28中所示的示例,可以考虑空间/时间邻近块的运动矢量精度来确定当前块的合并候选。例如,基于对空间邻近块的运动矢量精度与当前块的运动矢量精度之间的差或放大率是否大于或等于预定阈值的确定,可以确定空间/时间邻近块是否可以用作合并候选。例如,当空间合并候选的运动矢量精度是两个整数像素并且当前块的运动矢量精度是四分之一像素时,这可能意味着两个块之间的相关性不是很大。因此,可以将与当前块的运动矢量精度的差或放大率大于阈值的空间/时间邻近块设置成不可用作合并候选。也就是说,仅在空间/时间邻近块的运动矢量精度与当前块的运动矢量精度之间的差小于或等于阈值的情况下,空间/时间邻近块可以用作合并候选。不能称为合并候选的空间/时间邻近块可以不被添加至合并候选列表。

[0369] 在当前块的运动矢量精度与邻近块的运动矢量精度之间的差或放大率等于或小于阈值但两者精度不同时,可以将缩放的运动矢量设置成合并候选的运动矢量或者由合并索引指定的合并候选的运动矢量可以如在上面参照图28描述的实施方式中那样被缩放。

[0370] 可以从添加至合并候选列表的合并候选的运动矢量得到当前块的运动矢量。如果当前块的运动矢量精度和添加至合并候选列表的合并候选的运动矢量精度不同,则运动矢量精度差值可以指示运动矢量精度之间的差,或者可以指示对应于运动矢量精度的对应值之间的差值。此处,对应值可以指示与表5中示出的运动矢量精度对应的索引值,或者可以指示分配给如表6中示出的每个运动矢量精度的值。例如,在表6中,由于分配给四分之一的像素的对应值是2,并且分配给整数像素的对应值是3,因此可以将两个精度的差值确定为2。

[0371] [表6]

	运动矢量精度单位	对应值
[0372]	八分之一像素像素单位	0
	四分之一像素像素单位	1
	二分之一像素像素单位	2
	整数像素像素单位	3
	两个整数像素像素单位	4
[0373]	四个整数像素像素单位	5

[0374] 可以通过使用运动矢量精确放大率而不是运动矢量精度差值来确定时间/空间邻近块的可用性。此处,运动矢量精度放大率可以表示两个运动矢量精度之间的比率。例如,可以将四分之一像素与整数像素之间的放大率定义为4。图29和图30是用于说明得到多个运动矢量单位的时间运动矢量(时间MV)的方法的图。

[0375] 可以将先前图片中使用的运动矢量用作合并模式或者高级运动矢量预测器(AMVP)模式中的运动矢量候选。为了这个目的,可以将运动矢量存储在 $N \times M$ 大小的单元中。从 $N \times M$ 大小的预定义区域得到的运动矢量被称为时间代表性运动矢量。时间代表性运动矢量可以用作另一图片的时间MV。

[0376] 此处, N 和 M 可以是4、8、16、32或者更大的常数,并且 N 和 M 可以彼此相同或者可以彼此不同。 $N \times M$ 大小可以是应用于整个视频序列的固定值,或者可以针对图片、片、图块等的每个单位不同地定义。为此,编码器可以确定最佳 $N \times M$ 大小并对其进行编码。当在预定义区域单元内存在多个运动矢量时,可以存储所有运动矢量,或者可以仅选择性地存储多个运动矢量的一部分。可以将所有运动矢量中的之一存储为代表性运动矢量,并且代表性运动矢量可以是位于预定义区域单元内的左上角处的块的运动矢量。然而,本发明不限于此,并且代表性运动矢量可以是右上角块、左下角块、右下角块或包括中心位置的块的运动矢量。此处,预定义区域单元被称为运动矢量存储基本块。

[0377] 如图29所示,可以将包括左上样本的预测单元的运动矢量或者编码单元的运动矢量存储为代表性运动矢量。可以基于序列、图片或片选择性地使用运动矢量存储基本单元的大小。例如,可以在图片0(图片顺序计数(POC)值为0的图片)中使用 16×16 大小的运动矢量存储基本块,并且可以在图片1(图片顺序计数(POC)值为1的图片)中使用 32×32 大小的运动矢量存储基本块。

[0378] 当从运动矢量存储基本单元得到代表性运动矢量时,可以基于运动矢量的精度得到代表性运动矢量。例如,如果如图30所示在运动矢量存储基本块内存在各种运动矢量精度,则可以得到具有最精确的精度(1/4像素比1/2像素更精确)的运动矢量作为代表性运动

矢量。

[0379] 相反,也可以得到具有最不精确的精度(1/2像素不如1/4像素精确)的运动矢量作为代表性运动矢量。替选地,可以得到具有与当前CU相同的运动矢量精度的块的运动矢量作为代表性运动矢量。

[0380] 替选地,不管当前CU的运动矢量精度如何,都可以仅使用在编码器/解码器中预先约定的某些运动矢量精度。例如,如果约定编码器/解码器存储具有1/4像素精度的运动矢量作为代表性运动矢量,则可以将具有在运动矢量存储基本块中包括的多个运动矢量精度中的1/4像素精度的运动矢量存储为代表性运动矢量。如果存在具有1/4像素精度的多个运动矢量,则可以基于预定的优先级或者扫描顺序来执行存储代表性运动矢量。

[0381] 也就是说,如上所述,可以考虑运动矢量存储基本块中的块的位置或者运动矢量精度中的至少之一来存储代表性运动矢量。

[0382] 具体地,例如,如图30中指示的,在包括在 16×16 大小的左上运动矢量存储基本块中的两个编码单元CU0和CU1具有不同的运动矢量精度的情况下,可以考虑各个编码单元的运动矢量精度来使用代表性运动矢量。例如,如果CU0具有4个整数像素单位并且CU1具有四分之一像素单位,则可以得到从具有更精确的运动矢量精度的CU1的左上处的样本得到的运动矢量作为代表性运动矢量。也就是说,这意味着通过比较运动矢量存储基本单元中的每个编码单元的运动矢量精度来确定代表性运动矢量,而不是得到通过运动矢量存储基本单元的左上样本得到的运动矢量作为代表性的运动矢量来确定代表性运动矢量。

[0383] 另外,例如,在包括CU2、CU3、CU4和CU5的 16×16 运动矢量存储基本块中,可以将具有最精确精度的运动矢量的CU5的左上样本得到的运动矢量设置成代表性运动矢量。例如,当CU2和CU3具有整数像素像素单位、CU4具有二分之一像素像素单位并且CU5具有四分之一像素单位时,可以得到通过具有最精确运动矢量精度的CU5的左上样本得到的运动矢量作为代表性运动矢量。

[0384] 类似地,例如,在包括CU6和CU7的 16×16 运动矢量存储基本块中,可以将具有最精确精度的运动矢量的CU6的左上样本得到的运动矢量设置成代表性运动矢量。

[0385] 相反,还可以将在运动矢量存储基本块中具有最不精确的精度运动矢量设置成代表性运动矢量。在这种情况下,在以上示例中,可以将CU0、CU2(或CU3)和CU7分别被设置成代表性运动矢量。

[0386] 图31示出了作为应用本发明的另一实施方式,包括在应用于帧内预测样本插值的网络抽象层(NAL)中的语法元素。应用本发明的NAL单元可以包括例如视频参数集(VPS)、序列参数集(SPS)、图片参数集(PPS)和至少一个片集(片)。

[0387] 例如,在图31中示出包括在片集(SPS)中的语法元素,但是序列参数集(SPS)或者图片参数集(PPS)也可以包括语法元素。另外,要共同应用于序列单元或者图片单元的语法元素可以包括在序列参数集(SPS)或者图片参数集(PPS)中。另一方面,仅应用于片的语法元素优选地包括在片集(片)中。因此,可以考虑编码性能和效率来选择。

[0388] 语法元素'CU_partitioning'是指示划分编码单元的类型的信息。例如,'CU_partitioning'可以具有预定义的码字值,如表3或表4所示。

[0389] 另外,语法元素'CU_merge_flag'是指示邻近编码单元之间是否已经发生合并的信息。例如,当'CU_merge_flag=1'时,可以将其定义为发生合并的编码单元,并且当'CU_

merge_flag=0'时,可以将其定义为不发生合并的正常编码单元。

[0390] 另外,语法元素'CU_merge_idx'是指示在合并编码单元中发生合并的的形状的信息。也就是说,例如,如果'CU_merge_idx=0',则可以定义按编码顺序的前两个编码单元被合并,并且如果'CU_merge_idx=1',则可以定义按编码顺序的后两个编码单元被合并。另外,根据编码器-解码器之间的定义,可以定义当'CU_merge_idx=2'时按编码顺序的后两个编码单元被合并。

[0391] 另外,语法元素'mv_resolution_idx'是指示运动矢量的精度的索引信息。例如,可以如表5中所示来定义索引值'mv_resolution_idx'。

[0392] 尽管已经基于一系列步骤或流程图描述了上述实施方式,但是它们不限制本发明的时序顺序,并且可以根据需要同时或以不同顺序执行。此外,构成上述实施方式中的框图的部件(例如,单元、模块等)中的每一个可以由硬件设备或软件以及多个部件来实现。或者,可以通过单个硬件设备或软件来组合并实现多个部件。上述实施方式可以以程序指令的形式实现,所述程序指令可以通过各种计算机部件执行并被记录在计算机可读记录介质中。计算机可读记录介质可以包括程序命令、数据文件、数据结构等之一或其组合。计算机可读介质的示例包括诸如硬盘、软盘和磁带的磁介质、诸如CD-ROM和DVD的光学记录介质、诸如光磁软盘的磁光介质、专门被配置成存储和执行程序指令的介质和硬件设备例如ROM、RAM、闪存等。硬件设备可以被配置成作为一个或多个软件模块来进行操作以用于执行根据本发明的处理,反之亦然。

[0393] 工业应用性

[0394] 本发明可以应用于能够对视频进行编码/解码的电子设备。

100

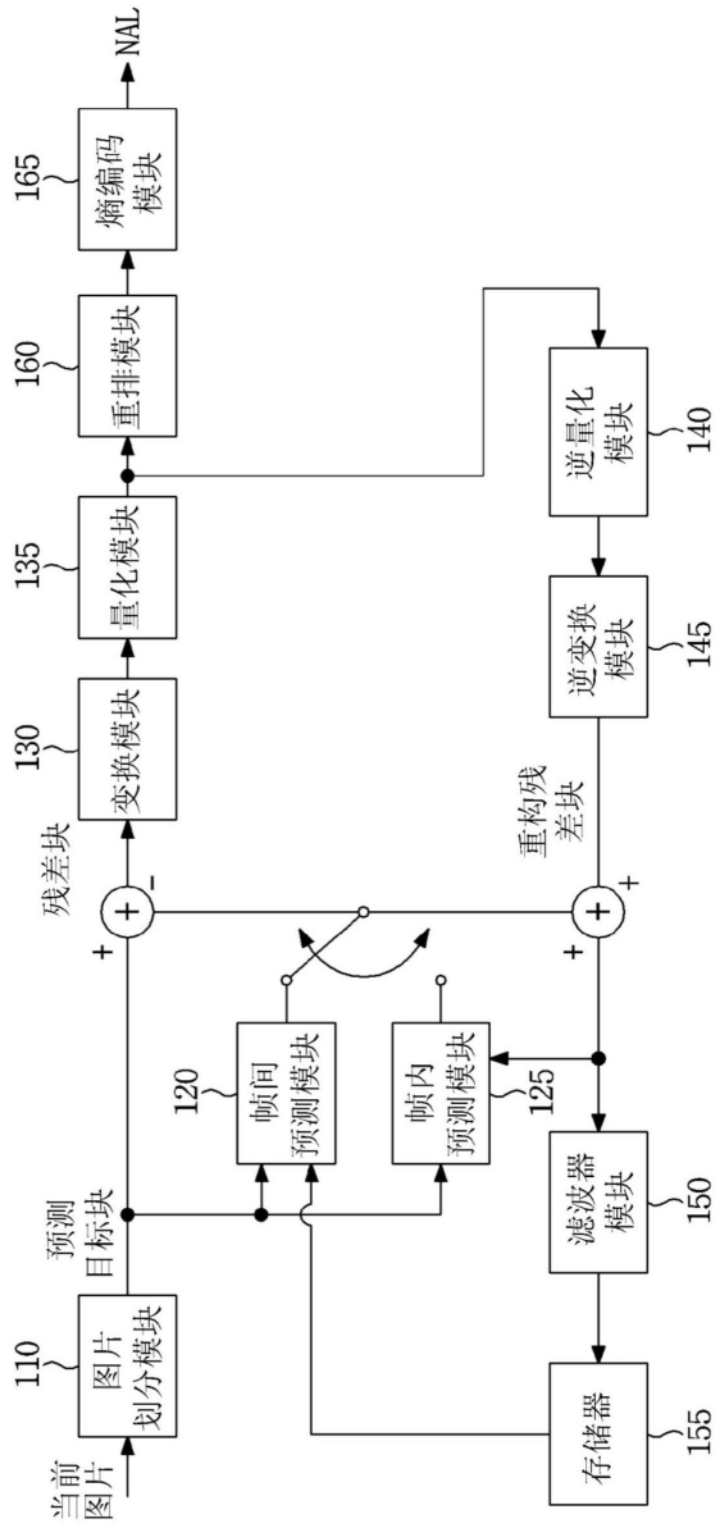
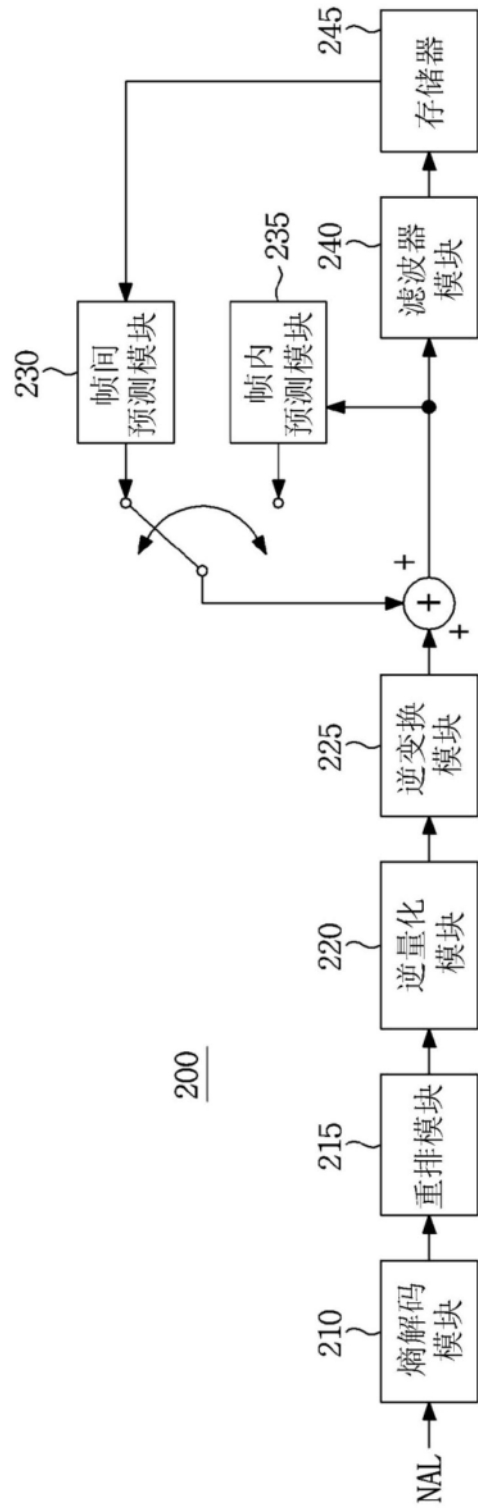


图1



200

图2

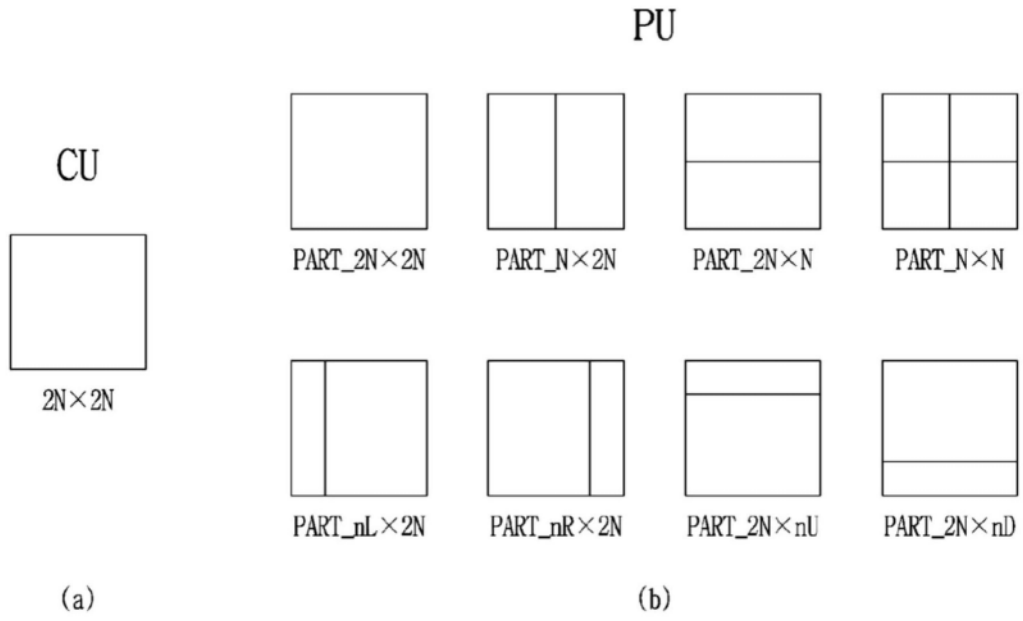


图3

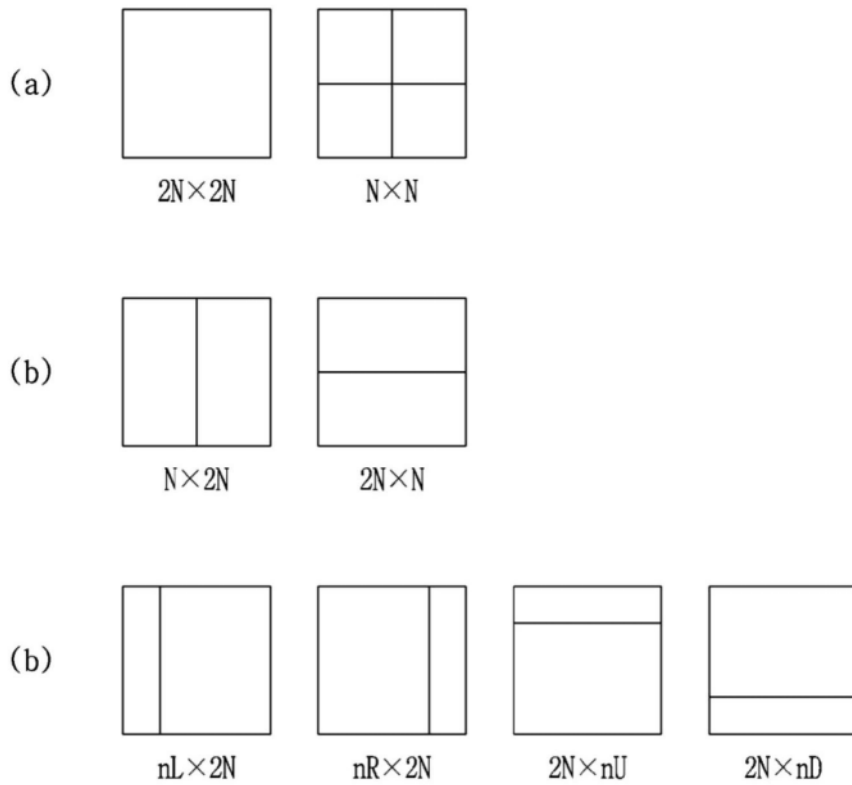


图4

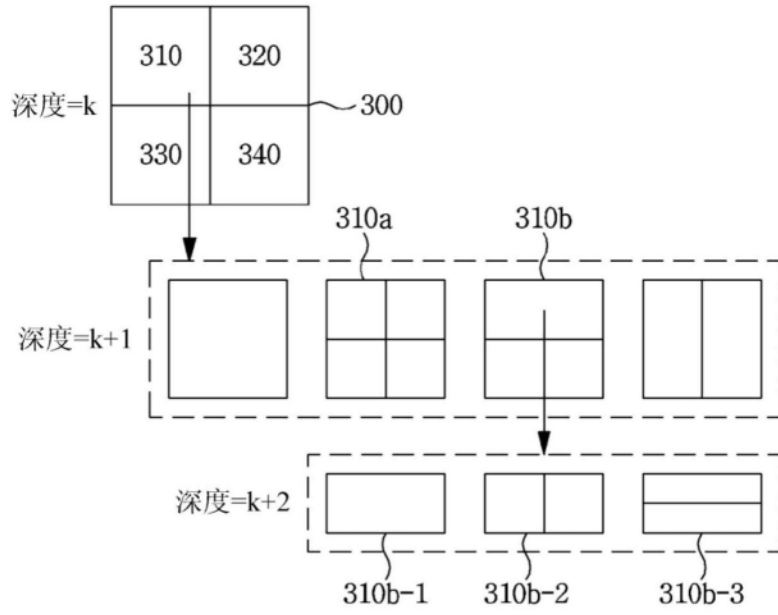


图5



图6

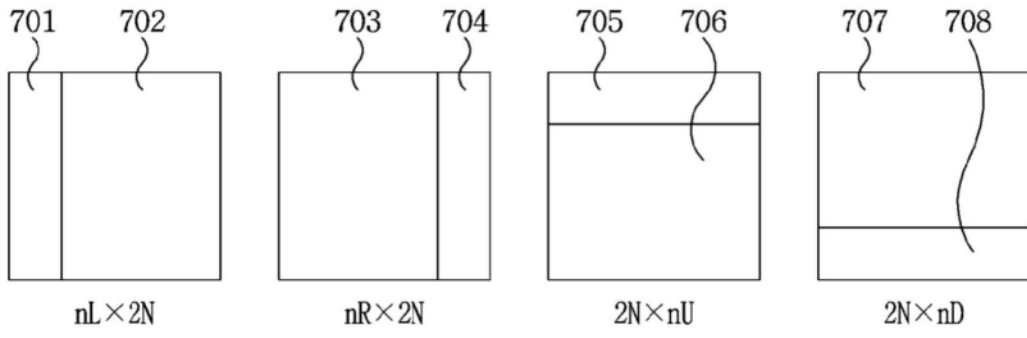


图7

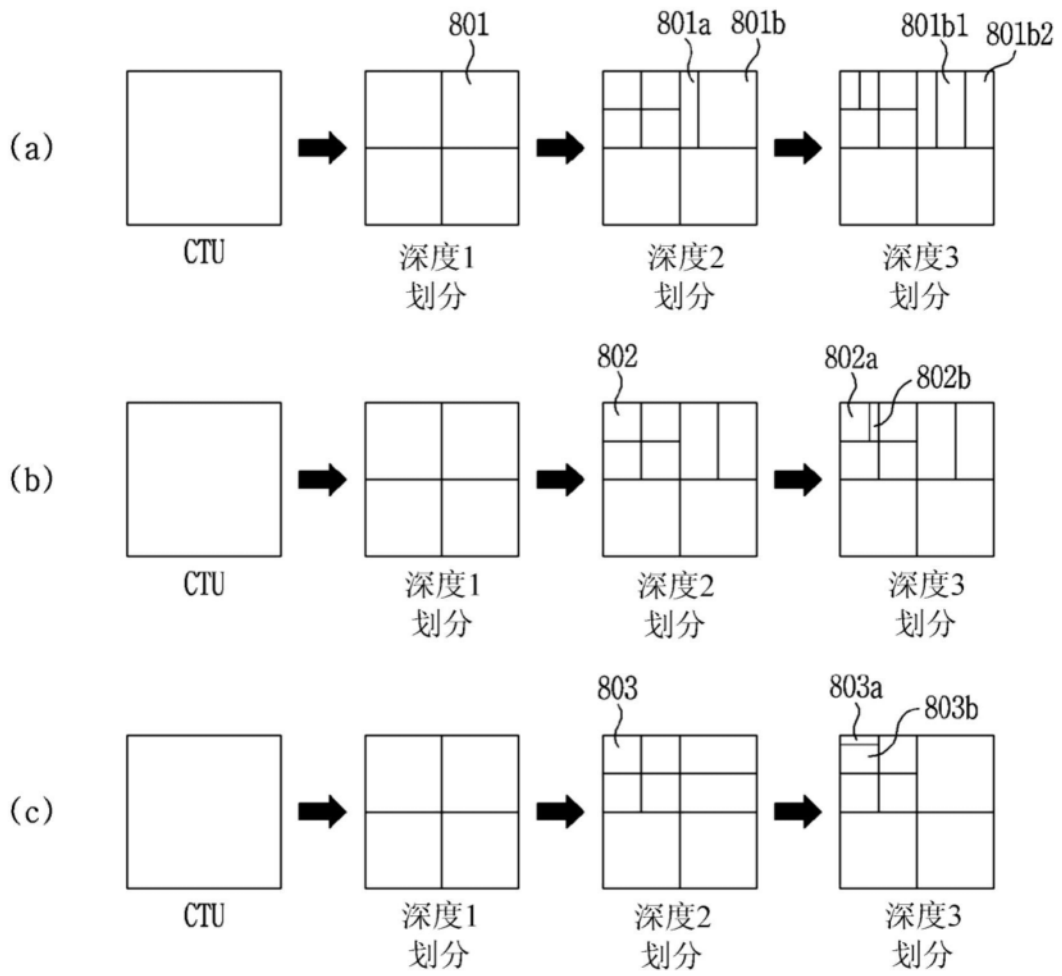


图8

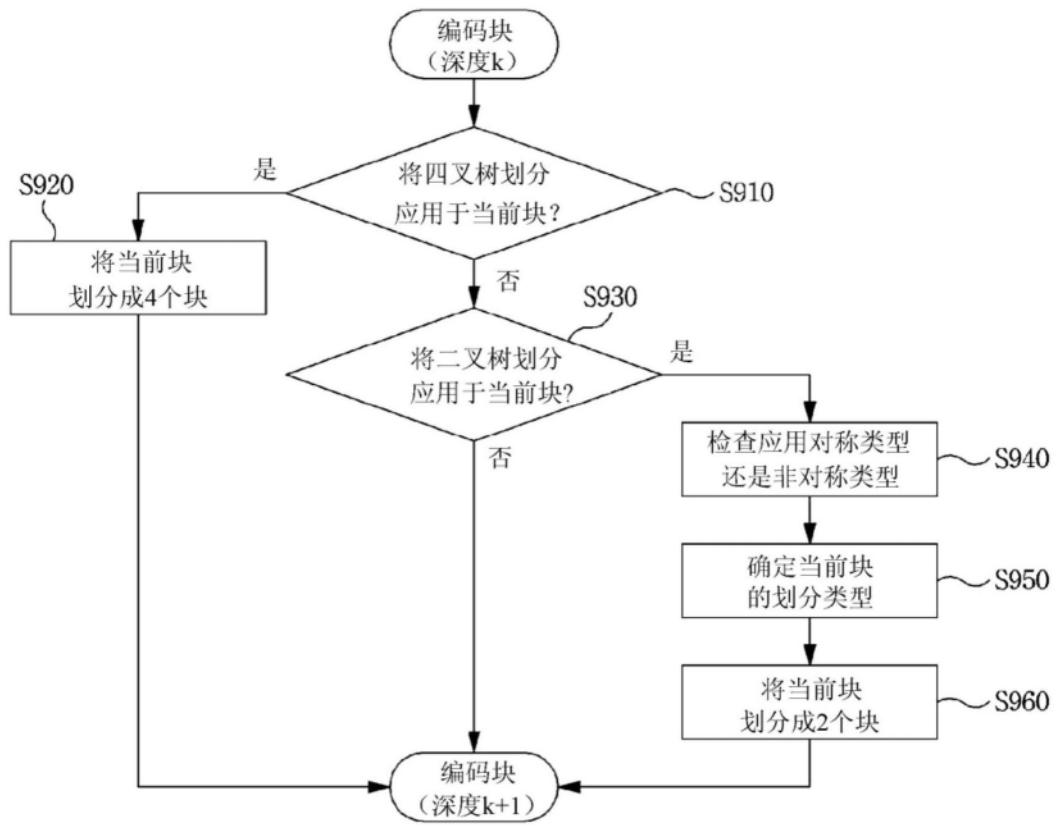


图9

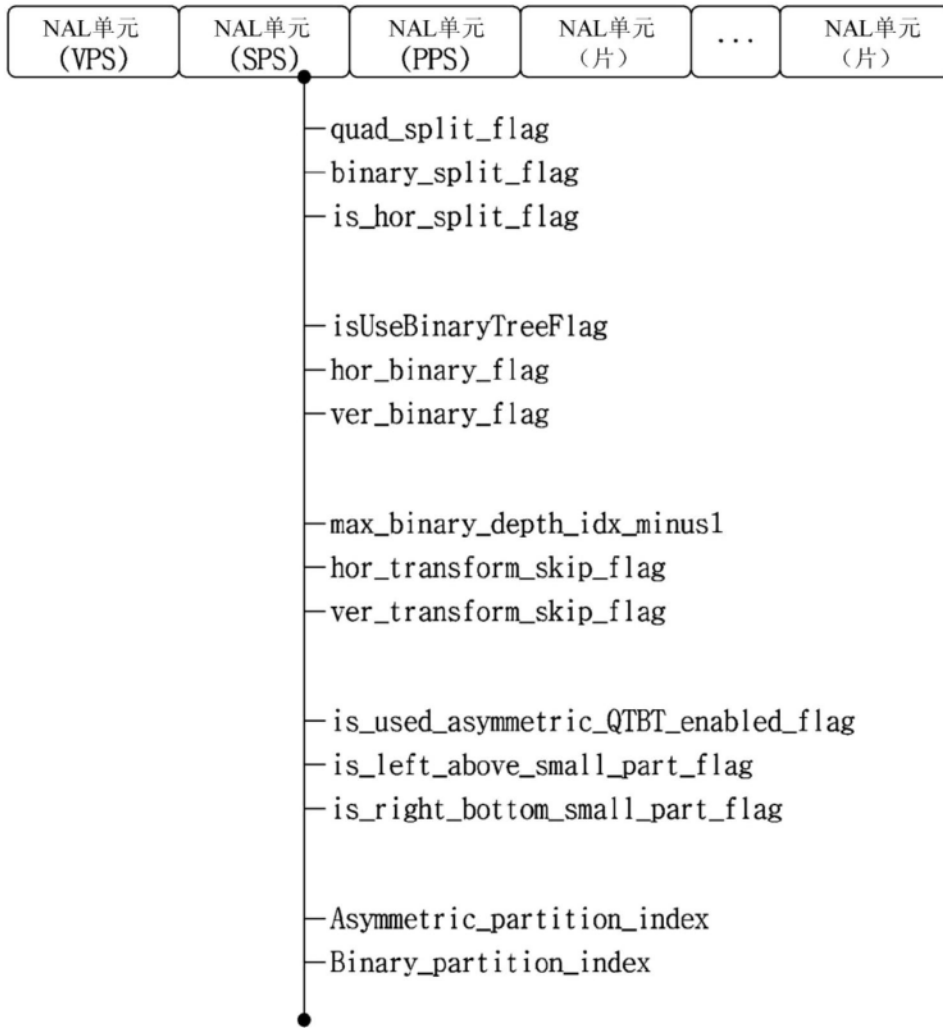


图10

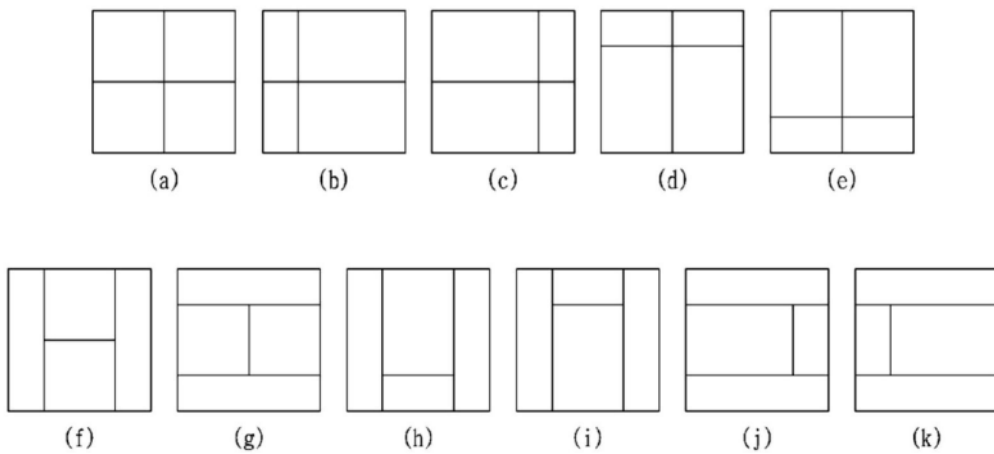


图11

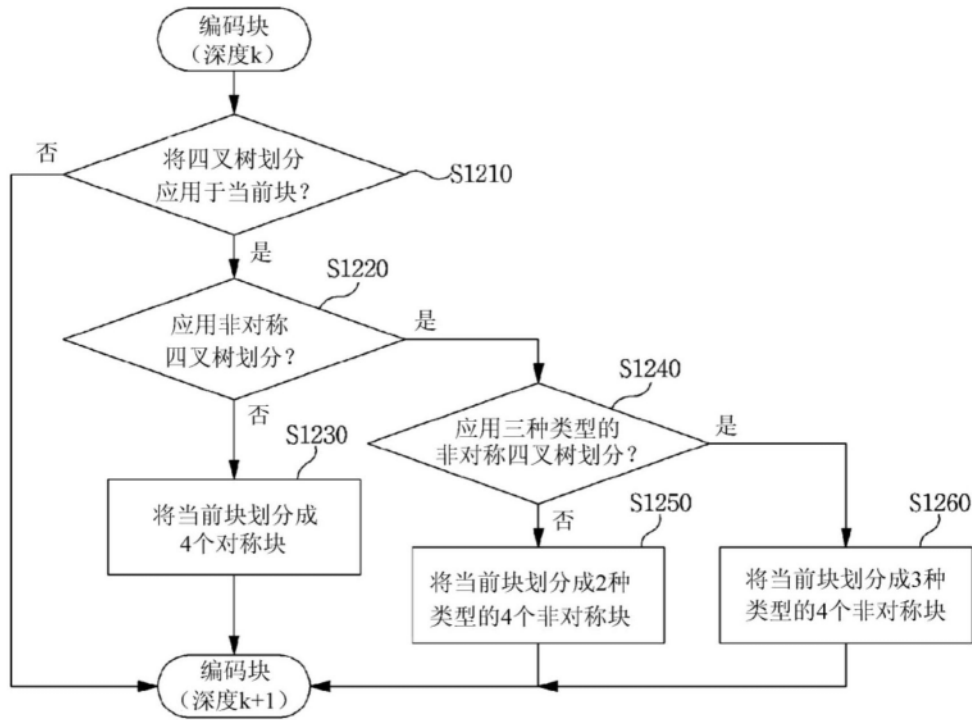


图12

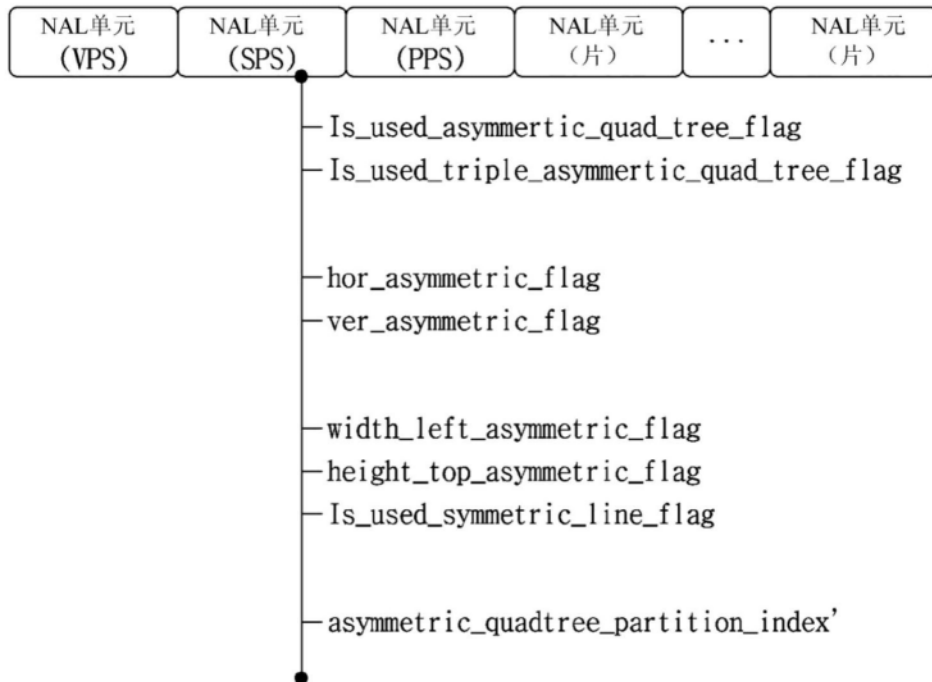


图13

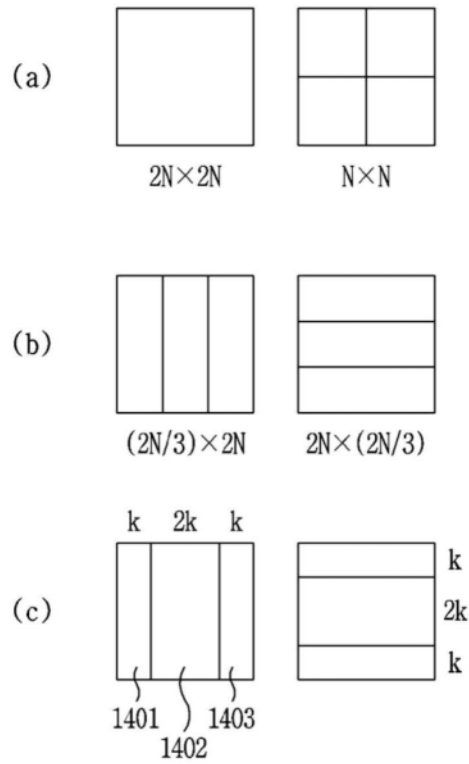


图14

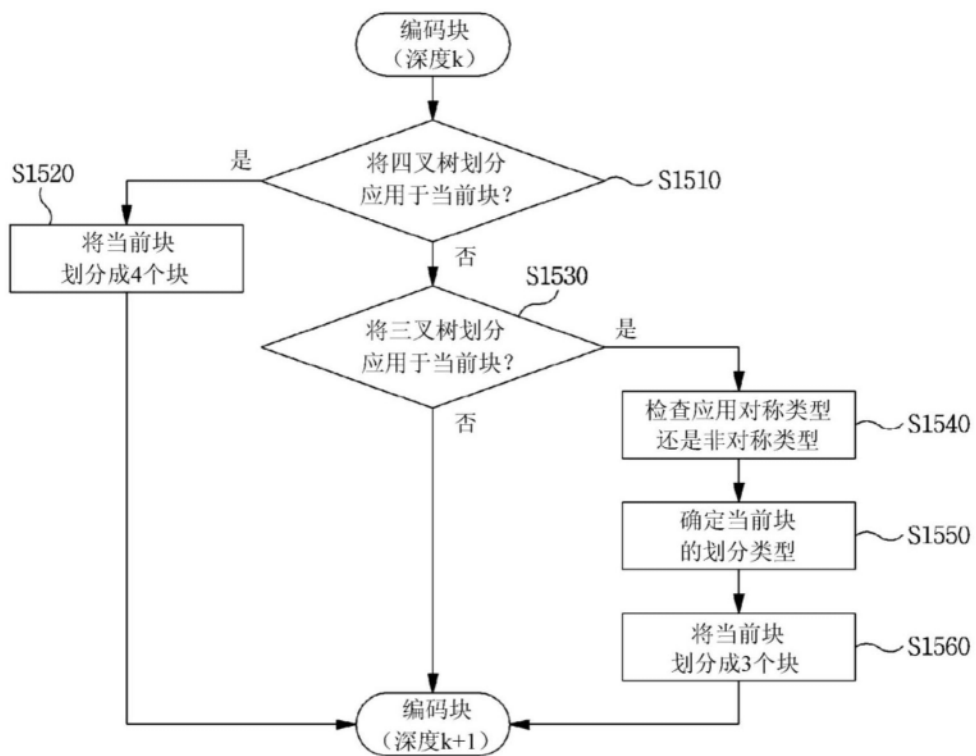


图15

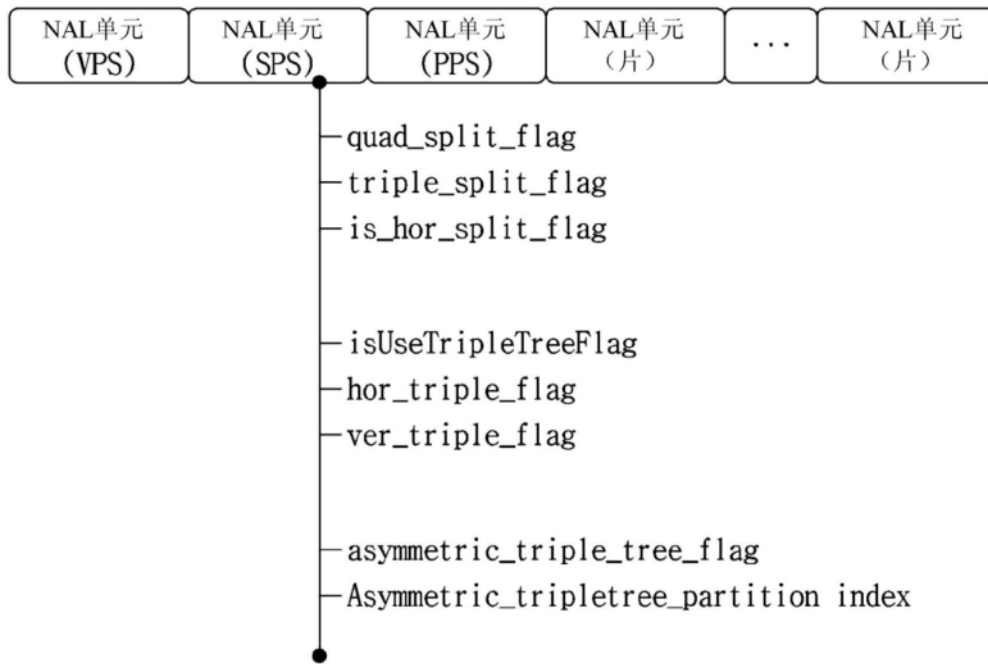


图16

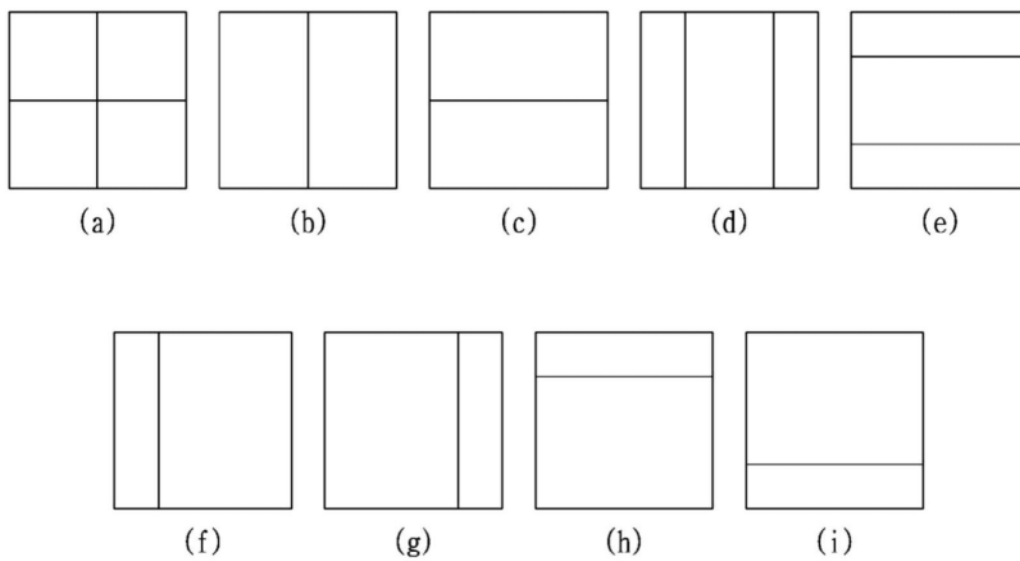


图17

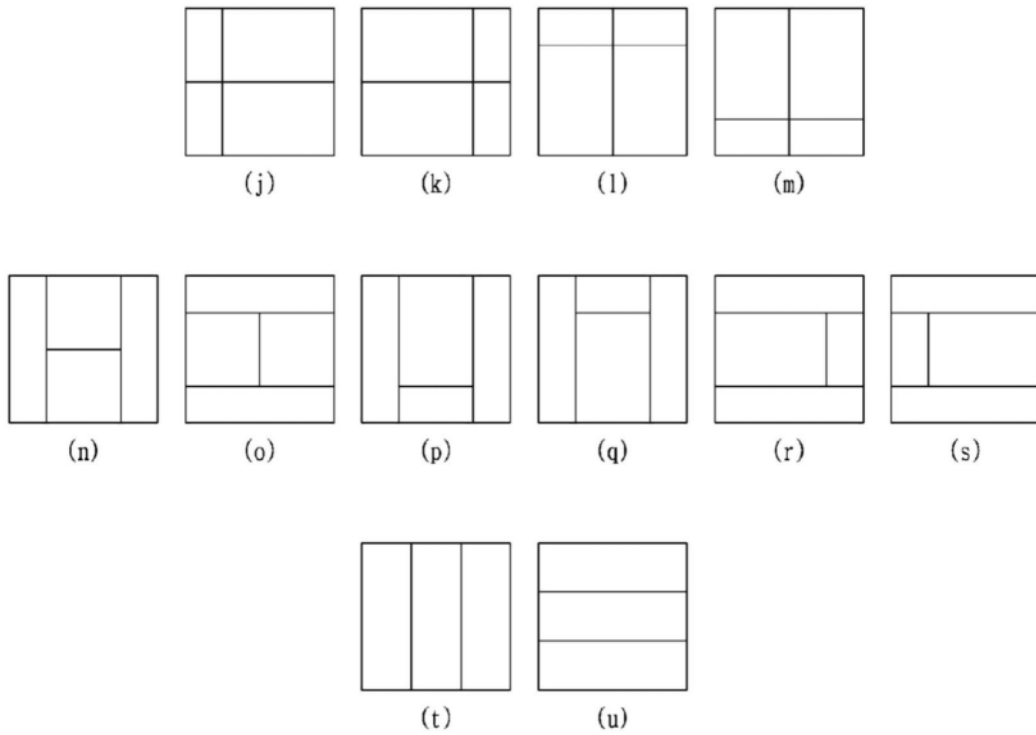


图18

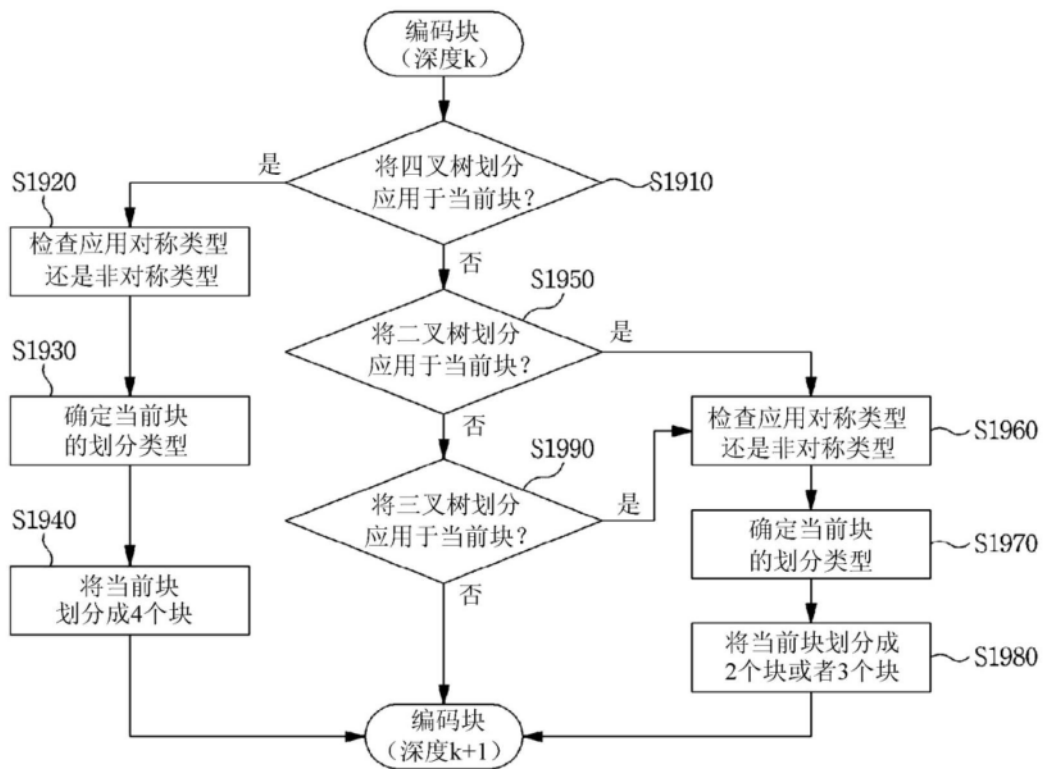


图19

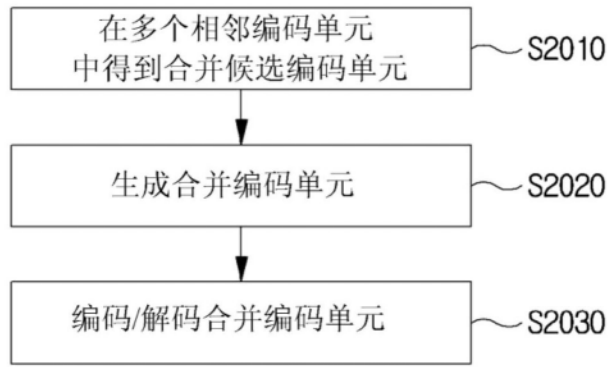


图20

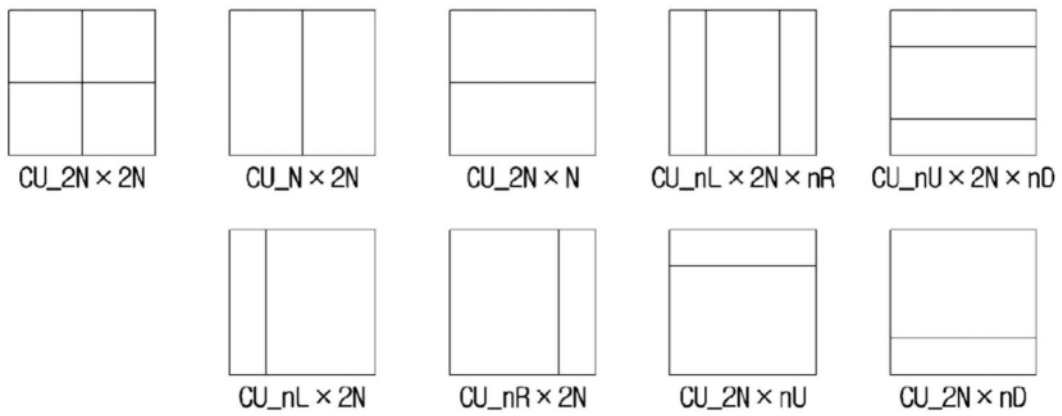


图21

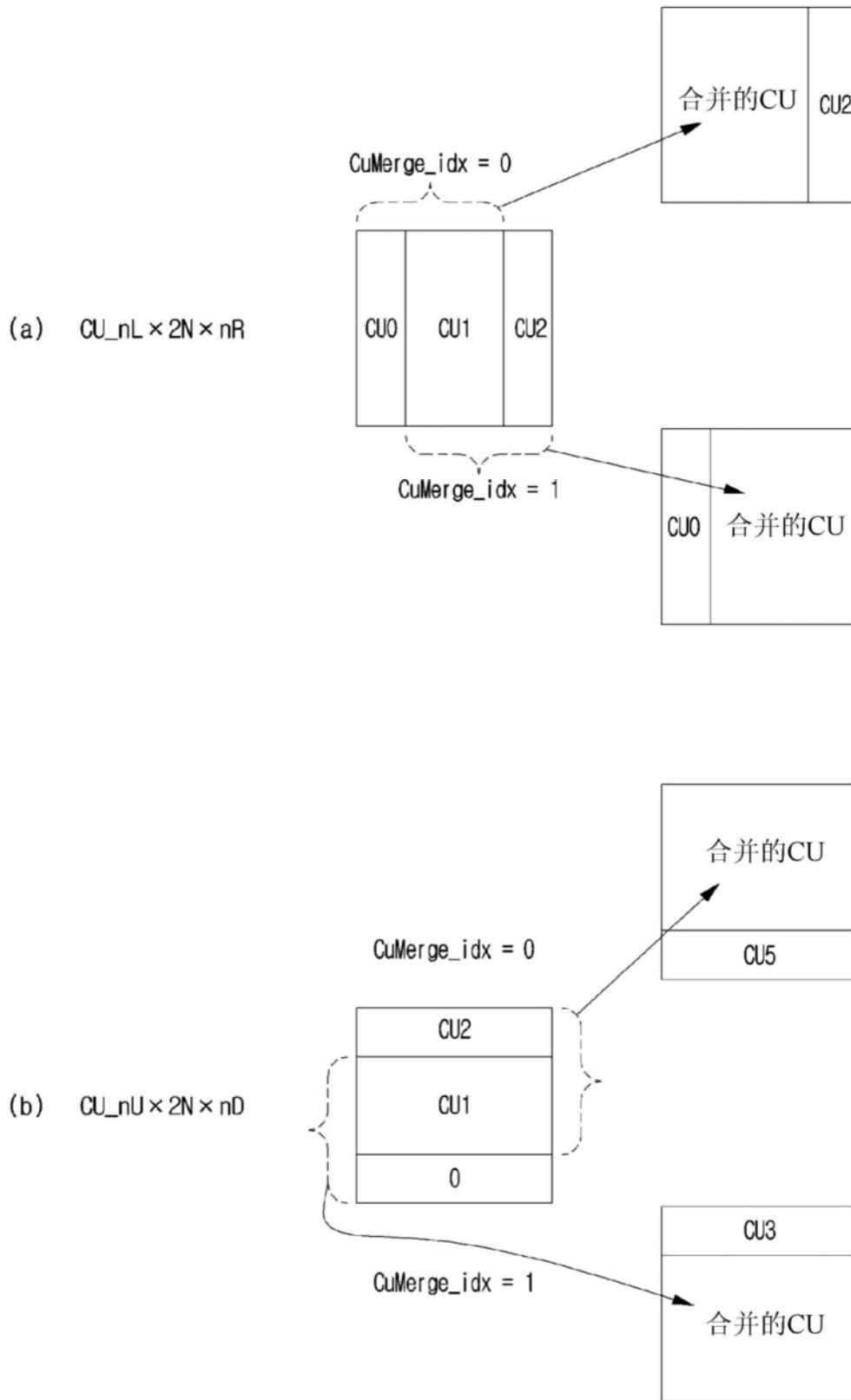


图22

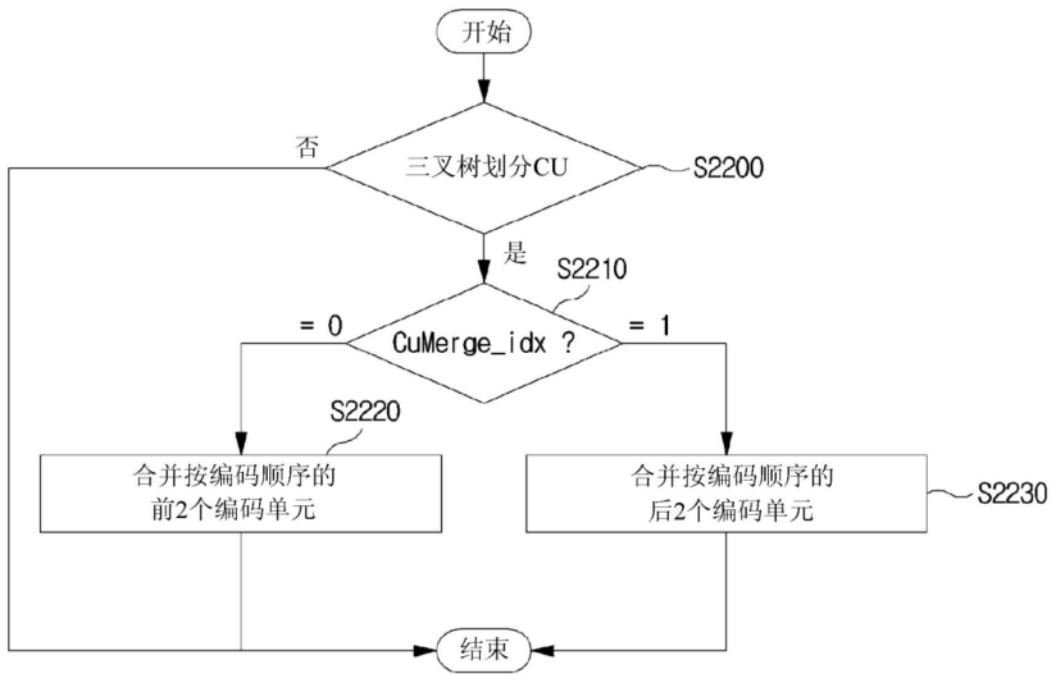


图23



图24

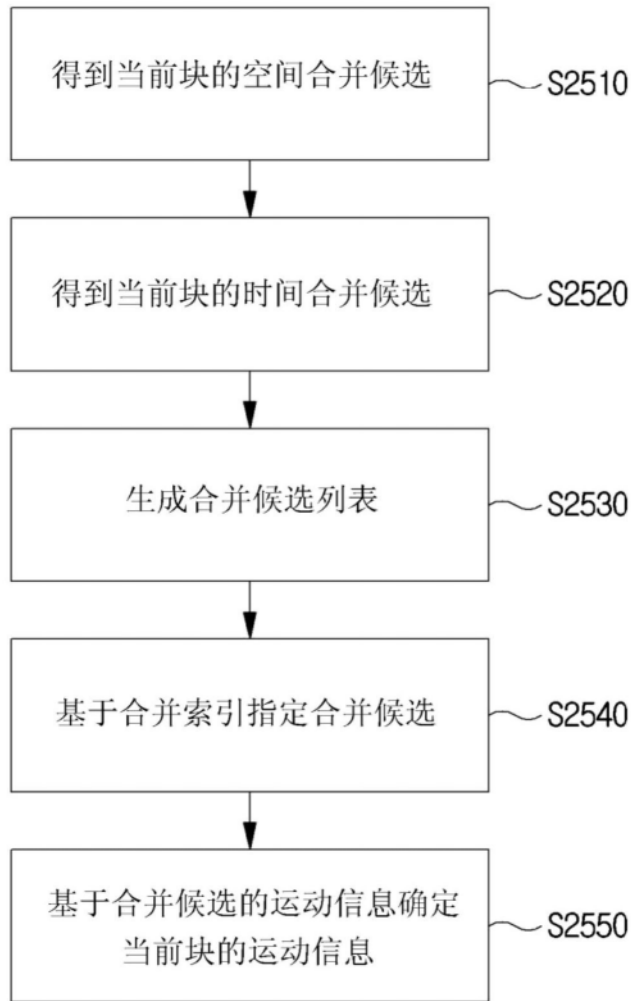


图25

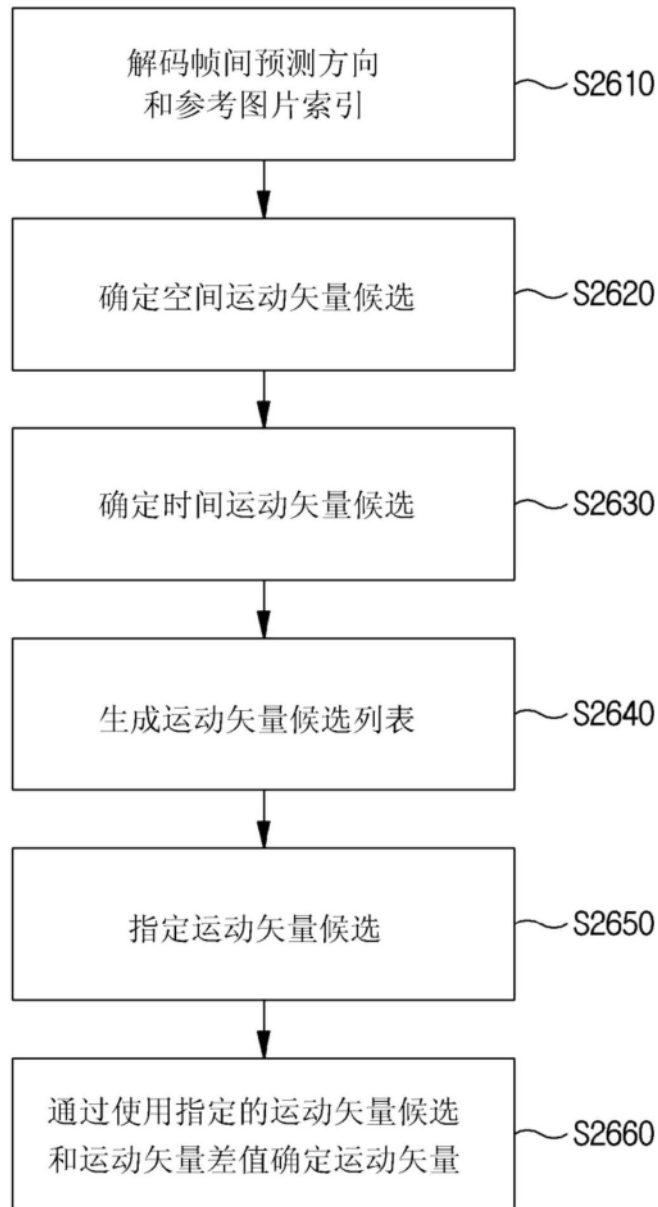


图26

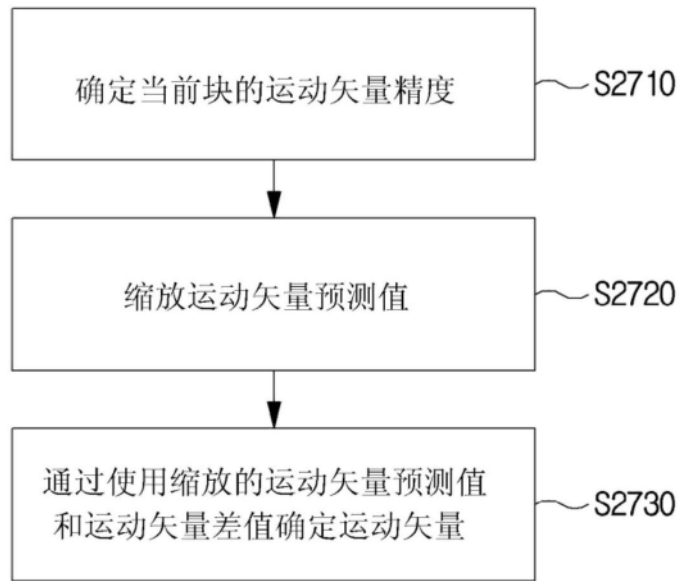


图27

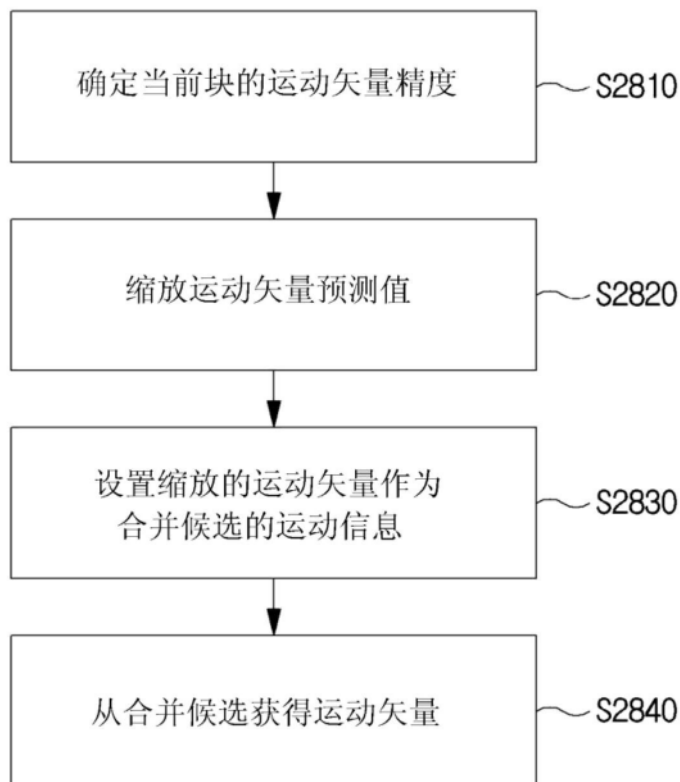


图28

● 运动矢量得到

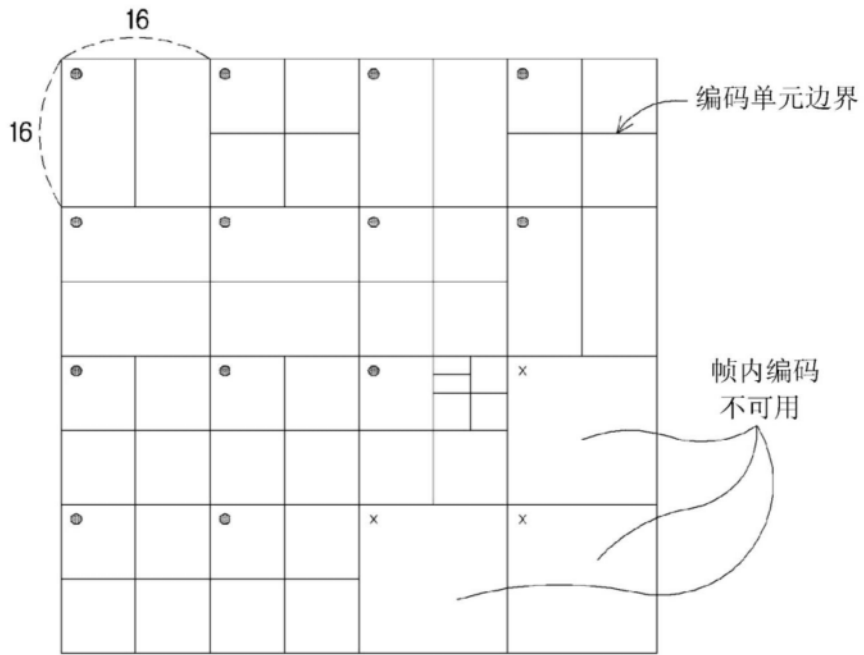


图29

● 运动矢量得到

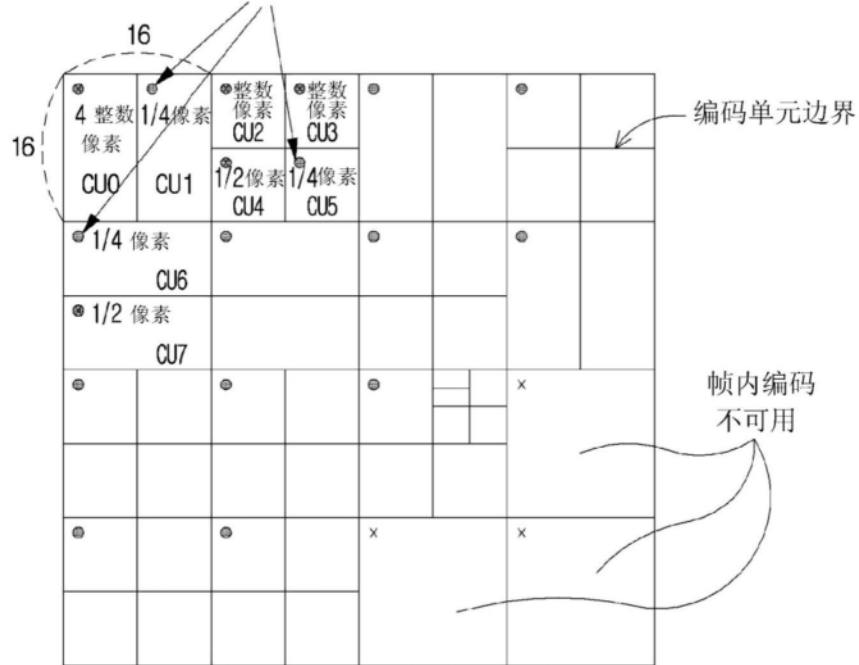


图30

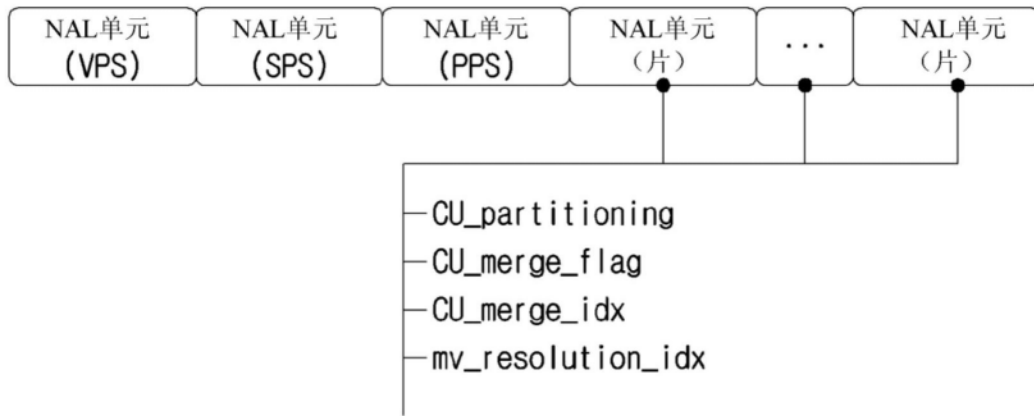


图31