



(12) 发明专利

(10) 授权公告号 CN 114762339 B

(45) 授权公告日 2023.10.27

(21) 申请号 202080083186.5

(22) 申请日 2020.10.05

(65) 同一申请的已公布的文献号  
申请公布号 CN 114762339 A

(43) 申请公布日 2022.07.15

(30) 优先权数据  
62/911,223 2019.10.05 US

(85) PCT国际申请进入国家阶段日  
2022.05.31

(86) PCT国际申请的申请数据  
PCT/KR2020/013481 2020.10.05

(87) PCT国际申请的公布数据  
W02021/066609 KO 2021.04.08

(73) 专利权人 LG电子株式会社  
地址 韩国首尔

(72) 发明人 柳先美 崔情娥 崔璋元

(74) 专利代理机构 北京三友知识产权代理有限公司 11127  
专利代理师 皇甫悦 刘久亮

(51) Int.Cl.  
H04N 19/60 (2006.01)  
H04N 19/70 (2006.01)  
H04N 19/186 (2006.01)  
H04N 19/137 (2006.01)  
H04N 19/132 (2006.01)  
H04N 19/124 (2006.01)  
H04N 19/122 (2006.01)

(56) 对比文件  
CN 105684442 A, 2016.06.15  
CN 105141957 A, 2015.12.09  
CN 107005717 A, 2017.08.01  
US 2017180737 A1, 2017.06.22

审查员 鲁小丽

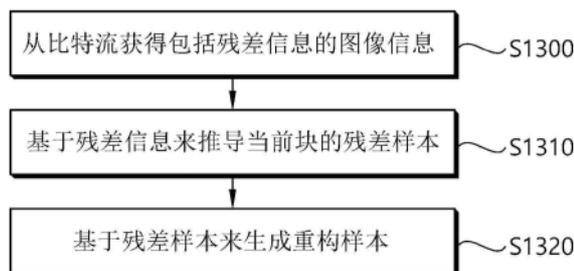
权利要求书2页 说明书60页 附图11页

(54) 发明名称

基于变换跳过和调色板编码相关高级语法元素的图像或视频编码

(57) 摘要

根据本文献的公开内容,可以定义高效解析/发信号通知对变换跳过和/或调色板编码相关信息有依赖性的语法元素的依赖性条件,并且基于该依赖性条件,可以确定是否解析变换跳过和/或调色板编码相关信息。因此,可以减少为了视频/图像编码而需要发信号通知的比特,并且编码效率可以提高。



1. 一种由解码设备执行的图像解码方法,该图像解码方法包括以下步骤:

从比特流获得包括残差信息的图像信息;  
基于所述残差信息来推导当前块的残差样本;以及  
基于所述残差样本来生成重构样本,  
其中,所述图像信息包括序列参数集SPS,

其中,所述SPS基于变换跳过启用信息来包括关于是否启用基于块的增量脉冲编码调制BDPCM的BDPCM启用信息,

其中,基于所述变换跳过启用信息或调色板编码启用信息中的至少一个,所述SPS包括与变换跳过模式的最小允许量化参数相关的最小量化参数信息,其中,基于所述变换跳过启用信息的值为1或者所述调色板编码启用信息的值为1的条件,在所述SPS中获得所述最小量化参数信息,并且其中,基于所述变换跳过启用信息的值和所述调色板编码启用信息的值两者都为0的条件,不在所述SPS中获得所述最小量化参数信息,

其中,基于所述变换跳过启用信息来确定是否对所述当前块应用变换,并且

其中,对于所述变换跳过模式或调色板模式,基于所述最小量化参数信息来推导所述残差样本或转义值。

2. 一种由编码设备执行的图像编码方法,该图像编码方法包括以下步骤:

推导当前块的残差样本;  
基于所述残差样本来生成残差信息;以及  
对包括所述残差信息的图像信息进行编码,  
其中,所述图像信息包括序列参数集SPS,

其中,所述SPS基于变换跳过启用信息来包括关于是否启用基于块的增量脉冲编码调制BDPCM的BDPCM启用信息,

其中,基于所述变换跳过启用信息或调色板编码启用信息中的至少一个,所述SPS包括与变换跳过模式的最小允许量化参数相关的最小量化参数信息,其中,基于所述变换跳过启用信息的值为1或者所述调色板编码启用信息的值为1的条件,在所述SPS中发信号通知所述最小量化参数信息,并且其中,基于所述变换跳过启用信息的值和所述调色板编码启用信息的值两者都为0的条件,不在所述SPS中发信号通知所述最小量化参数信息,

其中,基于所述变换跳过启用信息来确定是否对所述当前块应用变换,并且基于是否应用所述变换来对所述残差样本应用所述变换跳过模式。

3. 一种存储程序的非暂态计算机可读存储介质,当所述程序被执行时,执行图像编码方法,所述图像编码方法包括以下步骤:推导当前块的残差样本,基于所述残差样本来生成残差信息,并通过对包括所述残差信息的图像信息进行编码来生成比特流,

其中,所述图像信息包括序列参数集SPS,

其中,所述SPS基于变换跳过启用信息来包括关于是否启用基于块的增量脉冲编码调制BDPCM的BDPCM启用信息,

其中,基于所述变换跳过启用信息或调色板编码启用信息中的至少一个,所述SPS包括与变换跳过模式的最小允许量化参数相关的最小量化参数信息,其中,基于所述变换跳过启用信息的值为1或者所述调色板编码启用信息的值为1的条件,在所述SPS中发信号通知所述最小量化参数信息,并且其中,基于所述变换跳过启用信息的值和所述调色板编码启

用信息的值两者都为0的条件,不在所述SPS中发信号通知所述最小量化参数信息,

其中,基于所述变换跳过启用信息来确定是否对所述当前块应用变换,并且基于是否应用所述变换对所述残差样本应用所述变换跳过模式。

4.一种用于图像的数据的发送方法,该发送方法包括以下步骤:

获得比特流,其中,所述比特流是通过执行以下步骤来生成的:推导当前块的残差样本,基于所述残差样本来生成残差信息,并通过包括所述残差信息的图像信息进行编码来生成所述比特流;以及

发送包括所述比特流的所述数据,

其中,所述图像信息包括序列参数集SPS,

其中,所述SPS基于变换跳过启用信息来包括关于是否启用基于块的增量脉冲编码调制BDPCM的BDPCM启用信息,

其中,基于所述变换跳过启用信息或调色板编码启用信息中的至少一个,所述SPS包括与变换跳过模式的最小允许量化参数相关的最小量化参数信息,其中,基于所述变换跳过启用信息的值为1或者所述调色板编码启用信息的值为1的条件,在所述SPS中发信号通知所述最小量化参数信息,并且其中,基于所述变换跳过启用信息的值和所述调色板编码启用信息的值两者都为0的条件,不在所述SPS中发信号通知所述最小量化参数信息,

其中,基于所述变换跳过启用信息来确定是否对所述当前块应用变换,并且基于是否应用所述变换来对所述残差样本应用所述变换跳过模式。

## 基于变换跳过和调色板编码相关高级语法元素的图像或视频 编码

### 技术领域

[0001] 本技术涉及视频或图像编码,并且例如涉及基于与变换跳过和调色板编码相关的高级语法元素的图像或视频编码技术。

### 背景技术

[0002] 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。

[0003] 此外,对虚拟现实(VR)和人工现实(AR)内容以及诸如全息图这样的沉浸式媒体的兴趣和需求日益增长;并且表现出与实际图像/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。

[0004] 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。

[0005] 另外,需要用于通过根据在执行变换跳过和调色板编码时使用的绝对必要或辅助的信息的依赖性和非依赖性对是否对相关信息进行编码进行高效分类来提高整体图像/视频编码效率的方案。

### 发明内容

[0006] 技术问题

[0007] 本文献的技术主题是提供用于增强视频/图像编码效率的方法和设备。

[0008] 本文献的另一技术主题是提供用于高效地解析/发信号通知变换跳过和/或调色板编码相关信息的方法和设备。

[0009] 本文献的又一技术主题是提供用于根据在变换跳过和/或调色板编码期间使用的信息的依赖性和/或非依赖性来高效地确定是否执行编码的方法和设备。

[0010] 本文献的再一技术主题是提供用于定义用于有效地解析对变换跳过和/或调色板编码相关高级语法元素有依赖性的语法元素的依赖性条件并基于该依赖性条件来确定是否执行解析的方法和设备。

[0011] 技术方案

[0012] 根据本文献的实施方式,可以定义用于高效解析/发信号通知对变换跳过和/或调色板编码相关信息有依赖性的语法元素的依赖性条件,并且可以基于该依赖性条件来确定是否解析变换跳过和/或调色板编码相关信息。图像信息可以包括序列参数集(SPS),并且SPS可以包括变换跳过启用信息和调色板编码启用信息。例如,基于变换跳过启用信息的值为1或调色板编码启用信息的值为1的条件,SPS还可以包括与变换跳过模式的最小允许量化参数相关的最小量化参数信息。另外,例如,SPS还可以基于变换跳过启用信息的值为1的

条件来包括关于是否启用基于块的增量脉冲编码调制 (BDPCM) 的 BDPCM 启用信息。另外,例如,SPS 还可以基于变换跳过启用信息的值为 1 的条件来包括关于在变换跳过模式下使用的最大块尺寸的信息。

[0013] 根据本文献的实施方式,提供了由解码设备执行的视频/图像解码方法。视频/图像解码方法可以包括本文献的实施方式中公开的方法。

[0014] 根据本文献的实施方式,提供了执行视频/图像解码的解码设备。解码设备可以执行本文献的实施方式中公开的方法。

[0015] 根据本文献的实施方式,提供了由编码设备执行的视频/图像编码方法。视频/图像编码方法可以包括本文献的实施方式中公开的方法。

[0016] 根据本文献的实施方式,提供了执行视频/图像编码的编码设备。编码设备可以执行本文献的实施方式中公开的方法。

[0017] 根据本文献的实施方式,提供了存储根据本文献的实施方式中的至少一个中公开的视频/图像编码方法生成的编码后的视频/图像信息的计算机可读数字存储介质。

[0018] 根据本文献的实施方式,提供了存储使解码设备执行在本文献的实施方式中的至少一个中公开的视频/图像解码方法的编码后的信息或编码后的视频/图像信息的计算机可读数字存储介质。

[0019] 技术效果

[0020] 本文献可以具有各种效果。例如,根据本文献的实施方式,整体图像/视频压缩效率可以增强。另外,根据本文献的实施方式,可以高效地解析/发信号通知变换跳过和/或调色板编码相关信息。另外,根据本文献的实施方式,可以按照在变换跳过和/或调色板编码期间使用的信息的依赖性和/或非依赖性有效地确定是否执行编码。另外,根据本文献的实施方式,通过定义用于有效解析对变换跳过和/或调色板编码相关高级语法元素有依赖性的语法元素的依赖性条件并且按照依赖性条件确定是否执行解析,可以进行高效编码。另外,根据本文献的实施方式,通过按照与变换跳过和/或调色板编码相关高级语法元素有关的依赖性条件确定是否执行解析,可以节省发送的比特。

[0021] 可以通过本文献的详细示例获得的效果不限于以上列举的效果。例如,可以存在相关领域的普通技术人员能够从本文献中理解或导出的各种技术效果。因此,本文献的详细效果不限于在本文献中明确阐述的效果,而是可以包括可以从本文献的技术特征中理解或导出的各种效果。

## 附图说明

[0022] 图1示意性例示了适用本文献的实施方式的视频/图像编码系统的示例。

[0023] 图2是示意性例示了适用本文献的实施方式的视频/图像编码设备的配置的示图。

[0024] 图3是示意性说明了适用本文献的实施方式的视频/图像解码设备的配置的示图。

[0025] 图4例示了适用本文献的实施方式的示意性视频/图像编码过程的示例。

[0026] 图5例示了适用本文献的实施方式的示意性视频/图像解码过程的示例。

[0027] 图6示意性例示了适用本文献的实施方式的熵编码方法的示例,并且图7示意性例示了编码设备中的熵编码器。

[0028] 图8示意性例示了适用本文献的实施方式的熵解码方法的示例,并且图9示意性例

示了解码设备中的熵解码器。

[0029] 图10示例性例示了编码图像/视频的层次结构。

[0030] 图11和图12示意性例示了根据本文献的实施方式的视频/图像编码方法和相关部件的示例。

[0031] 图13和图14示意性例示了根据本文献的实施方式的视频/图像解码方法和相关部件的示例。

[0032] 图15例示了适用本文献中公开的实施方式的内容流传输系统的示例。

### 具体实施方式

[0033] 本公开可按各种形式修改,将描述并且在附图中图示其特定实施方式。然而,这些实施方式并非旨在限制本公开。以下描述中使用的术语仅用于描述特定实施方式,而非旨在限制本公开。单数表达包括复数表达,只要清楚地不同解读即可。诸如“包括”和“具有”这样的术语旨在指示存在以下描述中使用的特征、数量、步骤、操作、元件、部件或其组合,因此应该理解,不排除存在或添加一个或更多个不同的特征、数量、步骤、操作、元件、部件或其组合的可能性。

[0034] 另外,本文献描述的附图的各个配置是为了说明作为彼此不同的特征的功能而独立的图示的,并不意指各个配置通过彼此不同的硬件或不同的软件被实现。例如,可以组合配置中的两个或更多个的配置以形成一个配置,并且也可以将一个配置划分为多个配置。在不脱离本文献的主旨的情况下,配置被组合和/或分离的实施方式被包括在权利要求的范围内。

[0035] 在本文献中,在一个附图中被单独说明的技术特征可以单独实现,或者可以同时实现。

[0036] 下文中,参考附图更具体地描述本文献的优选实施方式。下文中,在附图中,相同的附图标记被用于相同的元件,并且可以省略对相同元件的冗余描述。

[0037] 图1例示了可以应用本文献的实施方式的视频/图像编码系统的示例。

[0038] 参照图1,视频/图像编码系统可以包括源装置和接收装置。源装置可以通过数字存储介质或网络以文件或流传输的形式将编码后的视频/图像信息或数据发送到接收装置。

[0039] 源装置可包括视频源、编码设备和发送器。接收装置可包括接收器、解码设备和渲染器。编码设备可被称为视频/图像编码设备,解码设备可被称为视频/图像解码设备。发送器可被包括在编码设备中。接收器可被包括在解码设备中。渲染器可包括显示器,并且显示器可被配置为单独的装置或外部部件。

[0040] 视频源可通过捕获、合成或生成视频/图像的处理来获取视频/图像。视频源可包括视频/图像捕获装置和/或视频/图像生成装置。例如,视频/图像捕获装置可包括一个或更多个相机、包括先前捕获的视频/图像的视频/图像档案等。例如,视频/图像生成装置可包括计算机、平板计算机和智能电话,并且可(以电子方式)生成视频/图像。例如,可通过计算机等生成虚拟视频/图像。在这种情况下,视频/图像捕获处理可由生成相关数据的处理代替。

[0041] 编码设备可对输入视频/图像进行编码。为了压缩和编码效率,编码设备可执行诸

如预测、变换和量化这样的一系列过程。编码的数据(编码的视频/图像信息)可按比特流的形式输出。

[0042] 发送器可通过数字存储介质或网络将以比特流的形式输出的编码的图像/图像信息或数据以文件或流传输的形式发送至接收装置的接收器。数字存储介质可包括诸如USB、SD、CD、DVD、蓝光、HDD、SSD等的各种存储介质。发送器可包括用于通过预定文件格式生成媒体文件的元件,并且可包括用于通过广播/通信网络传输的元件。接收器可接收/提取比特流并且将所接收的比特流发送至解码设备。

[0043] 解码设备可通过执行与编码设备的操作对应的诸如反量化、逆变换和预测这样的一系列过程对视频/图像进行解码。

[0044] 渲染器可渲染解码的视频/图像。渲染的视频/图像可通过显示器显示。

[0045] 在本文献中,术语“A或B”可以意指“仅A”、“仅B”或“A和B二者”。换句话说,在本文献中,术语“A或B”可以被解释为指示“A和/或B”。例如,在本文献中,术语“A、B或C”可以意指“仅A”、“仅B”、“仅C”或“A、B、C的任何组合”。

[0046] 在本文献中使用的斜杠(/)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二者”。例如,“A、B、C”可以意指“A、B或C”。

[0047] 在本文献中,“A和B中的至少一个”可以意指“仅A”、“仅B”或“A和B二者”。另外,在本文献中,表述“A或B中的至少一者”或“A和/或B中的至少一者”可以被解释为与“A和B中的至少一者”相同。

[0048] 另外,在本文献中,“A、B和C中的至少一者”可以意指“仅A”、“仅B”、“仅C”或“A、B和C的任何组合”。另外,“A、B或C中的至少一者”或“A、B和/或C中的至少一者”可以意指“A、B和C中的至少一者”。

[0049] 另外,在本文献中使用的括号可以意指“例如”。具体地,在表达“预测(帧内预测)”的情况下,可以指示“帧内预测”被提议为“预测”的示例。换句话说,本文献中的术语“预测”不限于“帧内预测”,并且可以指示“帧内预测”被提议为“预测”的示例。另外,即使在表达“预测(即,帧内预测)”的情况下,也可以指示“帧内预测”被提议为“预测”的示例。

[0050] 本文献涉及视频/图像编码。例如,本文献中公开的方法/实施方式可以应用于通用视频编码(VVC)标准中公开的方法。另外,本文献中公开的方法/实施方式可以应用于基本视频编码(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。

[0051] 本文献提议了视频/图像编码的各种实施方式,并且除非另外提到,否则以上实施方式也可以彼此组合地执行。

[0052] 在本文献中,视频可以意指根据时间推移的一系列图像的集合。图片通常意指表示特定时间段中的一副图像的单元,并且切片/图块是编码时构成图片的一部分的单元。切片/图块可以包括一个或更多个编码树单元(CTU)。一个图片可以由一个或更多个切片/图块组成。图块是图片中的特定图块列和特定图块行内的CTU的矩形区域。图块列是高度等于图片的高度并且宽度由图片参数集中的语法元素指定的CTU的矩形区域。图块行是宽度由图片参数集中的语法元素指定并且高度等于图片高度的CTU的矩形区域。图块扫描是以下的分割图片的CTU的特定顺序排序:CTU在图块中的CTU光栅扫描中被连续排序,而图片中的图块在图片的图块的光栅扫描中被连续排序。切片包括可以被排他性包含在单个NAL单元

中的图片的整数个完整图块或图块内的整数个连续的完整CTU行。

[0053] 此外,一个图片可以被划分为两个或更多个子图片。子图片可以是图片内的一个或更多个切片的矩形区域。

[0054] 像素或画素(pixel)可意指构成一个图片(或图像)的最小单元。另外,“样本”可用作与像素对应的术语。样本通常可表示像素或像素值,并且可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。或者,样本可以意指空间域中的像素值,或者当像素值被变换到频域中时可以意指频域中的变换系数。

[0055] 单元可表示图像处理的基本单位。单元可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,单元可与诸如块或区域这样的术语互换使用。在一般情况下, $M \times N$ 块可包括M列和N行的样本(或样本数组)或变换系数的集合(或数组)。

[0056] 另外,在本文献中,量化/反量化和/或变换/逆变换中的至少一者可以被省略。当量化/反量化被省略时,量化变换系数可以被称为变换系数。当变换/逆变换被省略时,变换系数可以被称为系数或残差系数,或者为了表述一致性的缘故,可以仍被称为变换系数。

[0057] 在本文献中,量化变换系数和变换系数可以分别被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,并可以通过残差编码语法发信号通知关于变换系数的信息。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的逆变换(缩放)来推导缩放变换系数。可以基于对缩放变换系数的逆变换(变换)来推导残差样本。这也可以在本文献的其它部分中应用/表达。

[0058] 图2是示意性例示了可以应用本文献的实施方式的视频/图像编码设备的配置的示意图。下文中,所谓的编码设备可以包括图像编码设备和/或视频编码设备。

[0059] 参照图2,编码设备200可以包括并配置有图像分割器210、预测器220、残差处理器230、熵编码器240、加法器250、滤波器260和存储器270。预测器220可包括帧间预测器221和帧内预测器222。残差处理器230可包括变换器232、量化器233、反量化器234和逆变换器235。残差处理器230还可包括减法器231。加法器250可被称为重构器或重构块生成器。根据实施方式,上面已经描述的图像分割器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可由一个或更多个硬件部件(例如,编码器芯片组或处理器)配置。另外,存储器270可包括解码图片缓冲器(DPB),并且可由数字存储介质配置。硬件部件还可包括存储器270作为内部/外部部件。

[0060] 图像分割器210可将输入到编码设备200的输入图像(或者图片或帧)分割成一个或更多个处理单元。例如,处理单元可被称为编码单元(CU)。在这种情况下,编码单元可根据四叉树二叉树三叉树(QTBT)结构从编码树单元(CTU)或最大编码单元(LCU)递归地分割。例如,一个编码单元可基于四叉树结构、二叉树结构和/或三叉树结构被分割成深度更深的多个编码单元。在这种情况下,例如,可首先应用四叉树结构,稍后可应用二叉树结构和/或三叉树结构。另选地,可首先应用二叉树结构。可基于不再分割的最终编码单元来执行根据本公开的编码过程。在这种情况下,根据图像特性基于编码效率等,最大编码单元可用作最终编码单元,或者如果需要,编码单元可被递归地分割成深度更深的编码单元,使得具有最优尺寸的编码单元可用作最终编码单元。这里,编码过程可包括诸如预测、变换和重构这样的过程(将稍后描述)。作为另一示例,处理单元还可包括预测单元(PU)或变换单元

(TU)。在这种情况下,预测单元和变换单元中的每一者可从上述最终编码单元拆分或分割。预测单元可以是样本预测的单元,并且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。

[0061] 在一些情况下,单元可与诸如块或区域这样的术语互换使用。通常, $M \times N$ 块可表示由M列和N行组成的样本或变换系数的集合。样本通常可表示像素或像素值,并且可仅表示亮度分量的像素/像素值,也可以仅表示色度分量的像素/像素值。样本可用作与构成一个图片(或图像)的像素或像元对应的术语。

[0062] 编码设备200可以通过从输入图像信号(原始块、原始样本数组)中减去从帧间预测器221或帧内预测器222输出的预测信号(预测块、预测样本数组)来生成残差信号(残差块、残差样本数组),并且所生成的残差信号被发送到变换器232。在这种情况下,如所例示的,用于在编码器200内从输入图像信号(原始块、原始样本数组)中减去预测信号(预测块、预测样本数组)的单元可以被称为减法器231。预测器可以对待处理块(下文中,被称为当前块)执行预测并生成包括当前块的预测样本的预测块。预测器可以以当前块或CU为单位确定是应用帧内预测还是应用帧间预测。预测器可以生成诸如预测模式信息这样的各种关于预测的信息,以将所生成的信息传送到熵编码器240,如随后在描述每个预测模式时描述的。关于预测的信息可以被熵编码器240编码,从而以比特流的形式输出。

[0063] 帧内预测器222可参考当前图片中的样本来预测当前块。根据预测模式,所参考的样本可位于当前块附近,也可以远离当前块。在帧内预测中,预测模式可包括多个非定向模式和多个定向模式。例如,非定向模式可包括DC模式和平面模式。例如,根据预测方向的精细程度,定向模式可包括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。

[0064] 帧间预测器221可基于参考图片上运动矢量所指定的参考块(参考样本数组)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。运动信息可包括运动矢量和参考图片索引。运动信息还可包括帧间预测方向(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,还可以彼此不同。时间邻近块可被称为并置参考块、并置CU(co1CU)等,并且包括时间邻近块的参考图片还可被称为并置图片(co1Pic)。例如,帧间预测器221可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,可不发送残差信号。运动矢量预测(MVP)模式可通过使用邻近块的运动矢量作为运动矢量预测器并且发信号通知运动矢量差来指示当前块的运动矢量。

[0065] 预测器200可以基于随后将描述的各种预测方法来生成预测信号。例如,预测器不仅可以应用帧内预测或帧间预测来预测一个块,而且还可以同时应用帧内预测和帧间预测。这可以被称为组合的帧间和帧内预测(CIIP)。另外,预测器可以基于块内复制(IBC)预测模式或调色板模式,以便对块执行预测。IBC预测模式或调色板模式可以用于诸如屏幕内

容编码(SCC)这样的游戏等的内容图像/视频编码。IBC基本上在当前图片中执行预测,但在推导当前图片中的参考块方面,其与帧间预测类似地执行。即,IBC可以使用本文献中描述的帧间预测技术中的至少一种。调色板模式可以被视为帧内编码或帧内预测的示例。当应用调色板模式时,可以基于关于调色板索引和调色板表的信息来发信号通知图片中的样本值。

[0066] 通过预测器(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者用于生成残差信号。变换器232可以通过向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)、基于图的变换(GBT)或有条件非线性变换(CNT)中的至少一种。这里,当像素之间的关系信息被例示为曲线图时,GBT意指从该曲线图获得的变换。CNT意指基于通过使用所有先前重构的像素生成的预测信号而获取的变换。另外,变换处理也可以被应用于正方形的尺寸相同的像素块,并也可以应用于不是正方形的尺寸可变的块。

[0067] 量化器233可以对变换系数进行量化以将量化变换系数发送到熵编码器240,并且熵编码器240可以将量化信号(关于量化变换系数的信息)编码为编码量化信号并作为比特流输出。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具有块形式的量化变换系数重新布置成一维矢量形式,并还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。熵编码器240可以执行例如诸如指数哥伦布(Golomb)编码、上下文自适应可变长度编码(CAVLC)和上下文自适应二进制算术编码(CABAC)这样的各种编码方法。熵编码器240还可以一起或分别地对量化变换系数之外的视频/图像重构所必需的信息(例如,语法元素的值等)进行编码。编码后的信息(例如,编码后的视频/图像信息)可以以比特流的形式发送或以网络抽象层(NAL)单元为单位存储。视频/图像信息还可以包括关于诸如适应参数集(APS)、图片参数集(PPS)、序列参数集(SPS)或视频参数集(VPS)这样的各种参数集的信息。另外,视频/图像信息还可以包括常规约束信息。随后将在本文献中描述的被发信号通知/发送的信息和/或语法元素可以通过以上提到的编码过程被编码,从而被包括在比特流中。比特流可以通过网络发送或者被存储在数字存储介质中。这里,网络可以包括广播网络和/或通信网络等,并且数字存储介质可以包括诸如USB、SD、CD、DVD、蓝光、HDD和SSD这样的各种存储介质。用于发送从熵编码器240输出的信号的发送器(未例示)和/或用于存储信号的存储器(未例示)可以被配置为编码设备200的内部/外部元件,或者发送器也可以被包括在熵编码器240中。

[0068] 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,反量化器234和逆变换器235向量化变换系数应用反量化和逆变换,使得可以重构残差信号(残差块或残差样本)。加法器250将重构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,使得可以生成重构信号(重构图片、重构块、重构样本数组)。如在应用跳过模式的情况下,如果待处理块不存在残差,则可以将预测块用作重构块。加法器250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将在当前图片内处理的下一个块的帧内预测,并且如随后描述的,还用于通过滤波对下一个图片进行帧间预测。

[0069] 此外,可以在图片编码和/或重构处理中应用具有色度缩放的亮度映射(LMCS)。

[0070] 滤波器260可以向重构信号应用滤波从而改善主观/客观图像质量。例如,滤波器260可以向重构图片应用各种滤波方法以生成修改后的重构图片,并将修改后的重构图片

存储在存储器270中,具体地,存储在存储器270的DPB中。各种滤波方法可以包括例如解块滤波、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可以生成与滤波相关的各种信息,以将所生成的信息传送到熵编码器240,如随后在每种滤波方法的描述中描述的。与滤波相关的信息可以由熵编码器240编码并以比特流的形式被输出。

[0071] 发送到存储器270的修改后的重构图片可以被用作帧间预测器221中的参考图片。如果通过帧间预测器应用帧间预测,则可以避免编码设备200与解码设备之间的预测失配,并且可以提高编码效率。

[0072] 存储器270的DPB可以存储修改后的重构图片,以用作帧间预测器221中的参考图片。存储器270可以存储从其推导出(或编码)当前图片内的运动信息的块的运动信息和/或先前重构的图片内的块的运动信息。所存储的运动信息可以被传送到帧间预测器221,以被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,并可以将重构样本传送到帧内预测器222。

[0073] 图3是示意性说明了适用本文献的视频/图像解码设备的配置的示图。下文中,所谓的解码设备可以包括图像解码设备和/或视频解码设备。

[0074] 参照图3,解码设备300可包括熵解码器310、残差处理器320、预测器330、加法器340、滤波器350和存储器360。预测器330可包括帧间预测器332和帧内预测器331。残差处理器320可包括反量化器321和逆变换器322。根据实施方式,以上已描述的熵解码器310、残差处理器320、预测器330、加法器340和滤波器350可由硬件部件(例如,解码器芯片组或处理器)配置。另外,存储器360可包括解码图片缓冲器(DPB)或者可由数字存储介质配置。硬件部件还可包括存储器360作为内部/外部部件。

[0075] 当输入包括视频/图像信息的比特流时,解码设备300可响应于在图2中例示的编码设备中处理视频/图像信息的处理来重构图像。例如,解码设备300可基于从比特流获得的块分割相关信息来推导单元/块。解码设备300可使用应用于编码设备的处理单元来执行解码。因此,例如,解码的处理单元可以是编码单元,并且编码单元可根据二叉树结构、二叉树结构和/或三叉树结构从编码树单元或最大编码单元分割。可从编码单元推导一个或多个变换单元。另外,通过解码设备300解码和输出的重构图像信号可通过再现设备再现。

[0076] 解码设备300可以以比特流的形式接收从图2中例示的编码设备输出的信号,并且可以通过熵解码器310对接收到的信号进行解码。例如,熵解码器310可以通过对比特流进行解析来推导图像重构(或图片重构)所必需的信息(例如,视频/图像信息)。视频/图像信息还可以包括关于诸如自适应参数集(APS)、图片参数集(PPS)、序列参数集(SPS)和视频参数集(VPS)这样的各种参数集的信息。另外,视频/图像信息还可以包括常规约束信息。解码设备可以进一步基于关于参数集的信息和/或常规约束信息对图片进行解码。随后将在本文献中描述的被发信号通知/接收的信息和/或语法元素可以通过解码过程被解码并从比特流获取。例如,熵解码器310可以基于诸如指数哥伦布编码、CAVLC或CABAC这样的编码方法对比特流内的信息进行解码,并输出图像重构所必需的语法元素的值和残差相关变换系数的量化值。更具体地,CABAC熵解码方法可以从比特流接收与每个语法元素对应的bin,使用待解码的语法元素信息以及邻近块和待解码块的解码信息或在前一级中解码的符号/bin的信息来确定上下文模型,并通过根据所确定的上下文模型预测bin生成概率以执行bin的算术解码来生成与每个语法元素的值对应的符号。此时,CABAC熵解码方法可以确定

上下文模型,然后使用针对下一个符号/bin的上下文模型的解码符号/bin的信息来更新上下文模型。由熵解码器310解码的信息当中的关于预测的信息可以被提供到预测器(帧间预测器332和帧内预测器331),并且由熵解码器310执行熵解码的残差值(即,量化变换系数和相关的参数信息)可以被输入到残差处理器320。残差处理器320可以推导出残差信号(残差块、残差样本、残差样本数组)。另外,由熵解码器310解码的信息当中的关于滤波的信息可以被提供到滤波器350。此外,用于接收从编码设备输出的信号的接收器(未例示)可以进一步被配置为解码设备300的内部/外部元件,或者接收器也可以是熵解码器310的部件。此外,根据本文献的解码设备可以被称为视频/图像/图片解码设备,并且解码设备也可以被分为信息解码器(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反量化器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。

[0077] 反量化器321可将量化后的变换系数反量化以输出变换系数。反量化器321可按二维块形式重排量化后的变换系数。在这种情况下,可基于由编码设备执行的系数扫描顺序来执行重排。反量化器321可使用量化参数(例如,量化步长信息)对量化后的变换系数执行反量化并且获得变换系数。

[0078] 逆变换器322对变换系数逆变换以获得残差信号(残差块、残差样本数组)。

[0079] 预测器330可执行当前块的预测并生成包括当前块的预测样本的预测块。预测器可基于从熵解码器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确定特定帧内/帧间预测模式。

[0080] 预测器可以基于随后将描述的各种预测方法来生成预测信号。例如,预测器不仅可以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。这可以被称为组合的帧间和帧内预测(CIIP)。另外,预测器可以基于块内复制(IBC)预测模式或调色板模式,以便对块执行预测。IBC预测模式或调色板模式可以用于诸如屏幕内容编码(SCC)这样的游戏等的图像/视频编码。IBC基本上在当前图片中执行预测,但在推导当前图片中的参考块方面,其与帧间预测类似地执行。即,IBC可以使用本文献中描述的帧间预测技术中的至少一种。调色板模式可以被视为帧内编码或帧内预测的示例。当应用调色板模式时,关于调色板表和调色板索引的信息可以被包括在视频/图像信息中并被发信号通知。

[0081] 帧内预测器331可以通过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的邻居中,或者其位置可以与当前块分开。在帧内预测中,预测模式可以包括多种非定向模式和多种定向模式。帧内预测器331还可以通过使用应用于邻近块的预测模式来确定将应用于当前块的预测模式。

[0082] 帧间预测器332可以基于参考图片上的运动矢量所指定的参考块(参考样本数组)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的量,可以基于邻近块与当前块之间的运动信息相关性以块、子块或样本为单位来预测运动信息。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关于帧间预测方向(L0预测、L1预测、Bi预测等)的信息。在帧间预测的情况下,邻近块可以包括当前图片中存在的空间邻近块和参考图片中存在的时间邻近块。例如,帧间预测器332可以基于邻近块来构造运动信息候选列表,并基于接收到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针对当前块的帧间预测的模式的信息。

[0083] 加法器340可以将所获得的残差信号与从预测器330(包括帧内预测器331和帧间预测器332)输出的预测信号(预测块或预测样本数组)相加以生成重构信号(重构图片、重构块或重构样本数组)。如在应用跳变模式的情况下,如果不存在待处理块的残差,则预测块可以用作重构块。

[0084] 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在当前图片中将处理的下一个块的帧内预测,并且如随后描述的,还可以通过滤波来输出或者还可以用于下一个图片的帧间预测。

[0085] 此外,具有色度缩放的亮度映射(LMCS)还可以被应用于图片解码处理。

[0086] 滤波器350可以通过向重构信号应用滤波来改善主观/客观图像质量。例如,滤波器350可以通过向重构图片应用各种滤波方法来生成修改后的重构图片,并将修改后的重构图片存储在存储器360中,具体地,存储在存储器360的DPB中。各种滤波方法可以包括例如解块滤波、样本自适应偏移、自适应环路滤波器、双边滤波器等。

[0087] 存储在存储器360的DPB中的(修改后的)重构图片可以被用作帧间预测器332中的参考图片。存储器360可以存储从其推导出(或解码出)当前图片中的运动信息的块的运动信息和/或已经重构的图片中的块的运动信息。所存储的运动信息可以被传送到帧间预测器332,以便被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,并将重构样本传送到帧内预测器331。

[0088] 在本文献中,在编码设备200的滤波器260、帧间预测器221和帧内预测器222中描述的示例性实施方式可以分别同等地或对应地应用于解码设备300的滤波器350、帧间预测器332和帧内预测器331。

[0089] 此外,如上所述,在执行视频编码时,执行预测以增强压缩效率。据此,可生成包括作为待编码的当前块(即,目标编码块)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。预测块在编码设备和解码设备中相同地推导,并且编码设备可向解码设备发信号通知关于原始块与预测块之间的残差(而非原始块的原始样本值本身)的信息(残差信息)从而增强图像编码效率。解码设备可基于残差信息来推导包括残差样本的残差块,可将残差块和预测块相加以生成包括重构样本的重构块,并且可生成包括重构块的重构图片。

[0090] 残差信息可通过变换过程和量化过程来生成。例如,编码设备可推导原始块与预测块之间的残差块,可对包括在残差块中的残差样本(残差样本数组)执行变换过程以推导变换系数,可对变换系数执行量化过程以推导量化后的变换系数,并且可将相关残差信息(通过比特流)用信号通知给解码设备。在这种情况下,残差信息可包括量化后的变换系数、位置信息、变换方案、变换核和量化参数等的值信息。解码设备可基于残差信息来执行反量化/逆变换过程并且可推导残差样本(或残差块)。解码设备可基于预测块和残差块来生成重构图片。此外,为了后续图片的帧间预测参考,编码设备还可对量化后的变换系数进行反量化/逆变换以推导残差块,并且可基于此来生成重构图片。

[0091] 准备以下附图来说明本文献的详细示例。由于示例性提出了附图中描述的详细装置的名称或详细术语或名称(例如,语法的名称),因此本文献的技术特征不限于附图中使

用的详细名称。

[0092] 图4例示了适用本文献的实施方式的示意性视频/图像编码过程的示例。在图4中，S400可以由以上参考图2描述的编码设备的预测器220执行，S410可以由残差处理器230执行，并且S420可以由熵编码器240执行。S400可以包括如本文献中说明的帧间/帧内预测过程，S410可以包括如本文献中说明的残差处理过程，并且S420可以包括如本文献中说明的信息编码过程。

[0093] 参照图4，示意性地，视频/图像编码过程可以不仅包括对用于图片重构的信息（例如，预测信息、残差信息、分割信息等）进行编码并且将编码后的信息以比特流的形式输出的过程，而且包括针对当前图片生成重构图片的过程以及对重构图片应用环路滤波的过程（可选的）。编码设备可以通过反量化器234和逆变换器235从量化变换系数推导（修改后的）残差样本，并可以基于作为S400的输出的预测样本和（修改后的）残差样本来生成重构图片。如上生成的重构图片可以等于由上述解码设备生成的重构图片。修改后的重构图片可以通过针对重构图片的环路滤波过程生成，并可以被存储在解码图片缓冲器或存储器270中。以与解码设备相同的方式，在后续图片编码期间，修改后的重构图片可以在帧间预测过程中被用作参考图片。如上所述，根据情况，可以省略环路滤波过程的部分或全部。在执行环路滤波过程的情况下，（环路）滤波相关信息（参数）可以由熵编码器240编码并以比特流的形式输出，并且解码设备可以基于滤波相关信息以与编码设备中的方法相同的方法执行环路滤波过程。

[0094] 通过这种环路滤波过程，可以减少诸如块伪影和振铃伪影这样的在图像/运动图像编码期间出现的噪声，并可以增强主观/客观视觉质量。另外，由于编码设备和解码设备二者执行环路滤波过程，因此它们可以推导出相同的预测结果，增强图片编码的可靠性，并减少为了图片编码而要发送的数据量。

[0095] 如上所述，不仅解码设备而且编码设备可以执行图片重构过程。可以基于对每个块的帧内预测/帧间预测来生成重构块，并且可以生成包括重构块的重构图片。在当前图片/切片/图块组是I图片/切片/图块组的情况下，可以仅基于帧内预测来重构当前图片/切片/图块组中所包括的块。此外，在当前图片/切片/图块组是P或B图片/切片/图块组的情况下，可以基于帧内预测或帧间预测来重构当前图片/切片/图块组中所包括的块。在这种情况下，可以对当前图片/切片/图块组中的一些块应用帧间预测，并且可以对其余块应用帧内预测。图片的颜色分量可以包括亮度分量和色度分量，并且除非有明确限制，否则本文献中提出的方法和实施方式可以应用于亮度分量和色度分量。

[0096] 图5例示了适用本文献的实施方式的示意性视频/图像解码过程的示例。在图5中，S500可以由以上参考图3描述的解码设备的熵解码器310执行，S510可以由预测器330执行，S520可以由残差处理器320执行，S530可以由加法器340执行，并且S540可以由滤波器350执行。S500可以包括如本文献中说明的信息解码过程，S510可以包括如本文献中说明的帧间/帧内预测过程，S520可以包括如本文献中说明的残差处理过程，S530可以包括如本文献中说明的块/图片重构过程，并且S540可以包括如本文献中说明的环路滤波过程。

[0097] 参照图5，图片解码过程可以示意性包括从比特流（通过解码）获得图像/视频信息的过程（S500）、图片重构过程（S510至S530）和用于重构图片的环路滤波过程（S540），如以上参考图3说明的。可以基于通过帧间/帧内预测的处理（S510）和残差处理（量化变换系数

的反量化和逆变换) (S520) 获得的预测样本和残差样本来执行图片重构过程。对于通过图片重构过程生成的重构图片, 可以通过环路滤波过程生成修改后的重构图片, 修改后的重构图片可以被作为解码图片输出, 并可以被存储在解码设备的解码图片缓冲器或存储器 360 中, 以在后面的图片解码期间用作帧间预测过程中的参考图片。

[0098] 根据情况, 可以省略环路滤波过程, 并且在这种情况下, 重构图片可以被作为解码图片输出, 并还可以被存储在解码设备的解码图片缓冲器或存储器 360 中, 以在后面的图片解码期间用作帧间预测过程中的参考图片。如上所述, 环路滤波过程 (S540) 可以包括去块滤波过程、采样自适应偏移 (SAO) 过程、自适应环路滤波 (ALF) 过程和/或双向滤波器过程, 并且可以省略其中的一些或全部。另外, 可以顺序地应用去块滤波过程、采样自适应偏移 (SAO) 过程、自适应环路滤波器 (ALF) 过程和/或双向滤波器过程中的一个或一些, 或者可以顺序地应用以上过程的全部。例如, 可以在对重构图片应用去块滤波过程之后执行 SAO 过程。另外, 例如, 可以在对重构图片应用去块滤波过程之后执行 ALF 过程。这些过程可以甚至以与编码设备相同的方式执行。

[0099] 此外, 如上所述, 编码设备可以执行基于例如诸如指数哥伦布、上下文自适应可变长度编码 (CAVLC) 和上下文自适应二进制算术编码 (CABAC) 这样的各种编码方法来执行熵编码。另外, 解码设备可以基于诸如指数哥伦布编码、CAVLC 或 CABAC 这样的编码方法来执行熵解码。下文中, 将描述熵编码/解码处理。

[0100] 图 6 示意性例示了适用本文献的实施方式的熵编码方法的示例, 并且图 7 示意性例示了编码设备中的熵编码器。图 7 的编码设备中的熵编码器可以同等地或甚至对应地应用于如上所述图 2 的编码设备 200 的熵编码器 240。

[0101] 参照图 6 和图 7, 编码设备 (熵编码器) 可以对图像/视频信息执行熵编码处理。图像/视频信息可以包括分区相关信息、预测相关信息 (例如, 帧间/帧内预测分类信息、帧内预测模式信息、帧间预测模式信息等)、残差信息和环路滤波相关信息, 并还可以包括其各种语法元素。熵编码可以以语法元素为单元执行。S600 至 S610 可以由如上所述图 2 的编码设备 200 的熵编码器 240 执行。

[0102] 编码设备可以对目标语法元素执行二进制化 (S600)。这里, 二进制化可以基于诸如截断莱斯二进制化处理或固定长度二进制化处理这样的各种二进制化方法, 并且用于目标语法元素的二进制化方法可以是预定义的。二进制化处理可以由熵编码器 240 中的二进制化器 242 执行。

[0103] 编码设备可以对目标语法元素执行熵编码 (S610)。编码设备可以基于诸如上下文自适应算术编码 (CABAC) 或上下文自适应可变长度编码 (CAVLC) 这样的熵编码技术来执行目标语法元素的 bin 串的基于正常编码 (基于上下文) 或基于旁路编码的编码, 并且其输出可以被包括在比特流中。熵编码处理可以由熵编码器 240 中的熵编码处理器 243 执行。比特流可以通过如上所述的 (数字) 存储介质或网络被传送到解码设备。

[0104] 图 8 示意性例示了适用本文献的实施方式的熵解码方法的示例, 并且图 9 示意性例示了解码设备中的熵解码器。图 9 的解码设备中的熵解码器可以同等地或甚至对应地应用于如上所述图 3 的解码设备 300 的熵解码器 310。

[0105] 参照图 8 和图 9, 解码设备 (熵解码器) 可以对编码后的图像/视频信息进行解码。图像/视频信息可以包括分区相关信息、预测相关信息 (例如, 帧间/帧内预测分类信息、帧内

预测模式信息、帧间预测模式信息等)、残差信息和环路滤波相关信息,并还可以包括其各种语法元素。熵编码可以以语法元素为单位执行。S800至S810可以由如上所述图3的解码设备300的熵解码器310执行。

[0106] 解码设备可以对目标语法元素执行二进制化(S800)。这里,二进制化可以是基于诸如截断莱斯二进制化处理或固定长度二进制化处理这样的各种二进制化方法的,并且用于目标语法元素的二进制化方法可以是预定义的。解码设备可以通过二进制化处理推导目标语法元素的启用值的启用bin串(bin串候选)。二进制化处理可以由熵解码器310中的二进制化器312执行。

[0107] 解码设备可以对目标语法元素执行熵解码(S810)。解码设备在从比特流中的输入比特依次解码和解析目标语法元素的各个bin的同时,将推导出的bin串与相应语法元素的启用bin串进行比较。如果推导出的bin串等于启用bin串中的一个,则与相应bin串对应的值可以被推导为相应语法元素的值。如若不然,则可以在进一步解析比特流中的下一个比特之后再次执行上述处理。通过这样的处理,可以使用可变长度比特来发信号通知相应信息,甚至无需使用比特流中特定信息(特定语法元素)的起始比特或结束比特。通过这样,可以针对低值分配数目相对较少的比特,因此整体编码效率可以增强。

[0108] 解码设备可以基于诸如CABAC或CAVLC这样的熵编码技术来执行来自比特流的bin串中的相应bin的基于上下文或基于旁路的解码。这里,比特流可以包括如上所述的用于图像/视频解码的各种信息。比特流可以通过如上所述的(数字)存储介质或网络被传送到解码设备。

[0109] 图10示例性例示了编码图像/视频的层次结构。

[0110] 参照图10,编码图像/视频被划分为处理图像/视频的解码过程及其本身的视频编码层(VCL)、发送和存储编码信息的子系统以及存在于VCL和子系统之间并负责网络自适应功能的网络抽象层(NAL)。

[0111] 在VCL中,可以生成包括压缩图像数据(切片数据)的VCL数据,或者可以生成包括图片参数集(PPS)、序列参数集(SPS)和视频参数集(VPS)的参数集或图像解码处理另外需要的补充增强信息(SEI)消息。

[0112] 在NAL中,可以通过向在VCL中生成的原始字节序列有效载荷(RBSP)添加头信息(NAL单元头)来生成NAL单元。在这种情况下,RBSP是指在VCL中生成的切片数据、参数集、SEI消息等。NAL单元头可以包括根据相应NAL单元中所包括的RBSP数据指定的NAL单元类型信息。

[0113] 另外,根据在VCL中生成的RBSP,NAL单元可以分为VCL NAL单元和非VCL NAL单元。VCL NAL单元可以意指包括关于图像的信息(切片数据)的NAL单元,并且非VCL NAL单元可以意指包括解码图像所需的信息(参数集或SEI消息)的NAL单元。

[0114] 根据子系统的数据标准,VCL NAL单元和非VCL NAL单元可以通过其上附接头信息通过网络来发送。例如,NAL单元可以被变换为诸如H.266/VVC文件格式、实时传输协议(RTP)、传输流(TS)等这样的预定标准的数据格式,并可以通过各种网络来发送。

[0115] 如上所述,可以根据相应NAL单元中所包括的RBSP数据结构以NAL单元类型指定NAL单元,并且关于NAL单元类型的信息可以被存储在NAL单元头中以便被发信号通知。

[0116] 例如,根据NAL单元是否包括关于图像的信息(切片数据),NAL单元可以简单地分

为VCL NAL单元类型和非VCL NAL单元类型。VCL NAL单元类型可以根据VCL NAL单元中所包括的图的性质和类型来分类,并且非VCL NAL单元类型可以根据参数集的类型来分类。

[0117] 以下是根据非VCL NAL单元类型中所包括的参数集的类型指定的NAL单元类型的示例。

[0118] -APS(自适应参数集)NAL单元:包括APS的NAL单元的类型

[0119] -DPS(解码参数集)NAL单元:包括DPS的NAL单元的类型

[0120] -VPS(视频参数集)NAL单元:包括VPS的NAL单元的类型

[0121] -SPS(序列参数集)NAL单元:包括SPS的NAL单元的类型

[0122] -PPS(图片参数集)NAL单元:包括PPS的NAL单元的类型

[0123] -PH(图片头)NAL单元:包括PH的NAL单元的类型

[0124] 上述NAL单元类型可以具有针对NAL单元类型的语法信息,并且语法信息可以被存储在NAL单元头中以便被发信号通知。例如,语法信息可以是nal\_unit\_type,并且可以通过nal\_unit\_type值指定NAL单元类型。

[0125] 此外,如上所述,一个图片可以包括多个切片,并且一个切片可以包括切片头和切片数据。在这种情况下,可以向一个图片中的多个切片(切片头和切片数据集)进一步添加一个图片头。图片头(图片头语法)可以包括常图片通用的信息/参数。在本文献中,图块组可以与切片或图片相混合或者被切片或图片替换。另外,在本文献中,图块组头可以与切片头或图片头相混合或者被切片头或图片头替换。

[0126] 切片头(切片头语法)可以包括可以切片通用的信息/参数。APS(APS语法)或PPS(PPS语法)可以包括可以常应用于一个或多个切片或图片的信息/参数。SPS(SPS语法)可以包括可以一个或多个序列通用的信息/参数。VPS(VPS语法)可以包括可以多个层通用的信息/参数。DPS(DPS语法)可以包括可以整个视频通用的信息/参数。DPS可以包括与编码视频序列(CVS)的级联相关的信息/参数。本文献中的高级语法(HLS)可以包括APS语法、PPS语法、SPS语法、VPS语法、DPS语法、图片头语法和切片头语法中的至少一个。

[0127] 在本文献中,从编码设备编码并以比特流的形式发信号通知给解码设备的图像/视频信息可以不仅包括图片中的分割相关信息、帧内/帧间预测信息、残差信息、环路滤波信息等,而且包括切片头中所包括的信息、图片头中所包括的信息、APS中所包括的信息、PPS中所包括的信息、SPS中所包括的信息、VPS中所包括的信息和/或DPS中所包括的信息。另外,图像/视频信息还可以包括NAL单元头的信息。

[0128] 在本文献中,为了表示从编码设备到解码设备的信息的信令,可以使用包括语法元素的表(语法表)。本文献中使用的语法表中的语法元素的顺序可以表示来自比特流的语法元素的解析顺序。编码设备可以配置和编码语法表,使得解码设备可以按解析顺序解析语法元素,并且解码设备可以通过按解析顺序从比特流解析和解码相应语法表中的语法元素来获得语法元素的值。

[0129] 此外,如上所述,可以通过变换和量化处理将残差样本推导为量化变换系数。量化变换系数可以被称为变换系数。在这种情况下,块中的变换系数可以以残差信息的形式被发信号通知。残差信息可以包括残差编码语法。即,编码设备可以用残差信息配置残差编码语法,并可以对所配置的残差编码语法进行编码,从而以比特流的形式输出编码后的残差编码语法。解码设备可以通过从比特流解码残差编码语法来推导残差(量化)变换系数。如

下所述,残差编码语法可以包括表示是否已向相应块应用变换、块中最后的有效变换系数的位置在哪里、子块中是否存在有效变换系数、以及有效变换系数的尺寸/符号是什么的语法元素。

[0130] 例如,可以基于残差信息中所包括的诸如last\_sig\_coeff\_x\_prefix、last\_sig\_coeff\_y\_prefix、last\_sig\_coeff\_x\_suffix、last\_sig\_coeff\_y\_suffix、coded\_sub\_block\_flag、sig\_coeff\_flag、par\_level\_flag、abs\_level\_gtX\_flag、abs\_remainder、coeff\_sign\_flag、dec\_abs\_level这样的语法元素对(量化)变换系数进行编码和/或解码。这可以被称为残差(数据)编码或(变换)系数编码。在这种情况下,可以省略变换/量化处理。在这种情况下,可以按照所确定的方法对残差样本的值进行编码和发信号通知。与残差数据编码/解码相关的语法元素可以如下表1中那样表示。

[0131] [表1]

[0132]

| residual_coding( x0, y0, log2TbWidth, log2TbHeight, cIdx ) {   | 描述符   |
|--|-------|
| if( ( tu_mts_idx[ x0 ][ y0 ] > 0   <br>( cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6 )<br>&& cIdx == 0 && log2TbWidth > 4 ) |       |
| log2ZoTbWidth = 4  |       |
| else   |       |
| log2ZoTbWidth = Min( log2TbWidth, 5 )  |       |
| MaxCbs = 2 * ( 1 << log2TbWidth ) * ( 1 << log2TbHeight )  |       |
| if( tu_mts_idx[ x0 ][ y0 ] > 0   <br>( cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6 )<br>&& cIdx == 0 && log2TbHeight > 4 )  |       |
| log2ZoTbHeight = 4   |       |
| else   |       |
| log2ZoTbHeight = Min( log2TbHeight, 5 )  |       |
| if( log2TbWidth > 0 )  |       |
| <b>last_sig_coeff_x_prefix</b>   | ac(v) |
| if( log2TbHeight > 0 )   |       |
| <b>last_sig_coeff_y_prefix</b>   | ac(v) |
| if( last_sig_coeff_x_prefix > 3 )  |       |
| <b>last_sig_coeff_x_suffix</b>   | ac(v) |
| if( last_sig_coeff_y_prefix > 3 )  |       |
| <b>last_sig_coeff_y_suffix</b>   | ac(v) |
| log2TbWidth = log2ZoTbWidth  |       |
| log2TbHeight = log2ZoTbHeight  |       |
| remBinsPass1 = ( ( 1 << ( log2TbWidth + log2TbHeight ) ) * 7 ) >> 2  |       |
| log2SbW = ( Min( log2TbWidth, log2TbHeight ) < 2 ? 1 : 2 )   |       |
| log2SbH = log2SbW  |       |
| if( log2TbWidth + log2TbHeight > 3 ) {   |       |
| if( log2TbWidth < 2 ) {  |       |
| log2SbW = log2TbWidth  |       |
| log2SbH = 4 - log2SbW  |       |
| } else if( log2TbHeight < 2 ) {  |       |
| log2SbH = log2TbHeight   |       |
| log2SbW = 4 - log2SbH  |       |
| }  |       |
| }  |       |
| numSbCoeff = 1 << ( log2SbW + log2SbH )  |       |
| lastScanPos = numSbCoeff   |       |
| lastSubBlock =<br>( 1 << ( log2TbWidth + log2TbHeight - ( log2SbW + log2SbH ) ) ) - 1  |       |
| do {   |       |
| if( lastScanPos == 0 ) {   |       |
| lastScanPos = numSbCoeff   |       |
| lastSubBlock--   |       |
| }  |       |
| lastScanPos--  |       |

[0133]

|   |       |
|---|-------|
| xS = DiagScanOrder[ log2TbWidth - log2SbW ][ log2TbHeight - log2SbH ]<br>[ lastSubBlock ][ 0 ]  |       |
| yS = DiagScanOrder[ log2TbWidth - log2SbW ][ log2TbHeight - log2SbH ]<br>[ lastSubBlock ][ 1 ]  |       |
| xC = ( xS << log2SbW ) + DiagScanOrder[ log2SbW ][ log2SbH ]<br>[ lastScanPos ][ 0 ]  |       |
| yC = ( yS << log2SbH ) + DiagScanOrder[ log2SbW ][ log2SbH ]<br>[ lastScanPos ][ 1 ]  |       |
| } while( ( xC != LastSignificantCoeffX )    ( yC != LastSignificantCoeffY ) )   |       |
| if( lastSubBlock == 0 && log2TbWidth >= 2 && log2TbHeight >= 2 && !transform_skip_flag[ x0 ][ y0 ] && lastScanPos > 0 )   |       |
| LfstDcOnly = 0  |       |
| if( ( lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2 )    ( lastScanPos > 7 && ( log2TbWidth == 2    log2TbWidth == 3 ) && log2TbWidth == log2TbHeight ) ) |       |
| LfstZeroOutSigCoeffFlag = 0   |       |
| QState = 0  |       |
| for( i = lastSubBlock; i >= 0; i-- ) {  |       |
| startQStateSb = QState  |       |
| xS = DiagScanOrder[ log2TbWidth - log2SbW ][ log2TbHeight - log2SbH ]<br>[ i ][ 0 ]   |       |
| yS = DiagScanOrder[ log2TbWidth - log2SbW ][ log2TbHeight - log2SbH ]<br>[ i ][ 1 ]   |       |
| inferSbDcSigCoeffFlag = 0   |       |
| if( ( i < lastSubBlock ) && ( i > 0 ) ) {   |       |
| <b>coded_sub_block_flag</b> [ xS ][ yS ]  | ae(v) |
| inferSbDcSigCoeffFlag = 1   |       |
| }   |       |
| firstSigScanPosSb = numSbCoeff  |       |
| lastSigScanPosSb = -1   |       |
| firstPosMode0 = ( i == lastSubBlock ? lastScanPos : numSbCoeff - 1 )  |       |
| firstPosMode1 = -1  |       |
| for( n = firstPosMode0; n >= 0 && remBinsPass1 >= 4; n-- ) {  |       |
| xC = ( xS << log2SbW ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 0 ]  |       |
| yC = ( yS << log2SbH ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 1 ]  |       |
| if( coded_sub_block_flag[ xS ][ yS ] && ( n > 0    !inferSbDcSigCoeffFlag ) && ( xC != LastSignificantCoeffX    yC != LastSignificantCoeffY ) ) {                     |       |
| <b>sig_coeff_flag</b> [ xC ][ yC ]  | ae(v) |
| remBinsPass1--  |       |
| if( sig_coeff_flag[ xC ][ yC ] )  |       |
| inferSbDcSigCoeffFlag = 0   |       |
| }   |       |
| if( sig_coeff_flag[ xC ][ yC ] ) {  |       |
| <b>abs_level_gtx_flag</b> [ n ][ 0 ]  | ae(v) |
| remBinsPass1--  |       |
| if( abs_level_gtx_flag[ n ][ 0 ] ) {  |       |

[0134]

|   |       |
|---|-------|
| <b>par_level_flag[ n ]</b>  | ae(v) |
| remBinsPass1--  |       |
| <b>abs_level_gtx_flag[ n ][ 1 ]</b>   | ae(v) |
| remBinsPass1--  |       |
| }   |       |
| if( lastSigScanPosSb == -1 )  |       |
| lastSigScanPosSb = n  |       |
| firstSigScanPosSb = n   |       |
| }   |       |
| AbsLevelPass1[ xC ][ yC ] = sig_coeff_flag[ xC ][ yC ] + par_level_flag[ n ] +<br>abs_level_gtx_flag[ n ][ 0 ] + 2 * abs_level_gtx_flag[ n ][ 1 ] |       |
| if( dep_quant_enabled_flag )  |       |
| QState = QStateTransTable[ QState ][ AbsLevelPass1[ xC ][ yC ] & 1 ]  |       |
| if( remBinsPass1 < 4 )  |       |
| firstPosMode1 = n - 1   |       |
| }   |       |
| for( n = numSbCoeff - 1; n >= firstPosMode1; n-- ) {  |       |
| xC = ( xS << log2SbW ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 0 ]  |       |
| yC = ( yS << log2SbH ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 1 ]  |       |
| if( abs_level_gtx_flag[ n ][ 1 ] )  |       |
| <b>abs_remainder[ n ]</b>   | ae(v) |
| AbsLevel[ xC ][ yC ] = AbsLevelPass1[ xC ][ yC ] + 2 * abs_remainder[ n ]   |       |
| }   |       |
| for( n = firstPosMode1; n >= 0; n-- ) {   |       |
| xC = ( xS << log2SbW ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 0 ]  |       |
| yC = ( yS << log2SbH ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 1 ]  |       |
| <b>dec_abs_level[ n ]</b>   | ae(v) |
| if( AbsLevel[ xC ][ yC ] > 0 )  |       |
| firstSigScanPosSb = n   |       |
| if( dep_quant_enabled_flag )  |       |
| QState = QStateTransTable[ QState ][ AbsLevel[ xC ][ yC ] & 1 ]   |       |
| }   |       |
| if( dep_quant_enabled_flag    !sign_data_hiding_enabled_flag )  |       |
| signHidden = 0  |       |
| else  |       |
| signHidden = ( lastSigScanPosSb - firstSigScanPosSb > 3 ? 1 : 0 )   |       |
| for( n = numSbCoeff - 1; n >= 0; n-- ) {  |       |
| xC = ( xS << log2SbW ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 0 ]  |       |
| yC = ( yS << log2SbH ) + DiagScanOrder[ log2SbW ][ log2SbH ][ n ][ 1 ]  |       |
| if( ( AbsLevel[ xC ][ yC ] > 0 ) &&<br>( !signHidden    ( n != firstSigScanPosSb ) ) )  |       |
| <b>coeff_sign_flag[ n ]</b>   | ae(v) |
| }   |       |
| if( dep_quant_enabled_flag ) {  |       |
| QState = startQStateSb  |       |
| for( n = numSbCoeff - 1; n >= 0; n-- ) {  |       |

|        |  |  |
|--------|--|--|
|        | $xC = (xS \ll \log2SbW) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][0]$  |  |
|        | $yC = (yS \ll \log2SbH) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][1]$  |  |
|        | $\text{if}(\text{AbsLevel}[xC][yC] > 0)$   |  |
|        | $\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$<br>$(2 * \text{AbsLevel}[xC][yC] - (\text{QState} > 1 ? 1 : 0)) *$<br>$(1 - 2 * \text{coeff\_sign\_flag}[n])$ |  |
|        | $\text{QState} = \text{QStateTransTable}[\text{QState}][\text{par\_level\_flag}[n]]$   |  |
|        | $\}$ else {  |  |
|        | $\text{sumAbsLevel} = 0$   |  |
|        | $\text{for}(n = \text{numSbCoeff} - 1; n \geq 0; n--) \{$  |  |
|        | $xC = (xS \ll \log2SbW) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][0]$  |  |
|        | $yC = (yS \ll \log2SbH) + \text{DiagScanOrder}[\log2SbW][\log2SbH][n][1]$  |  |
| [0135] | $\text{if}(\text{AbsLevel}[xC][yC] > 0) \{$  |  |
|        | $\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$<br>$\text{AbsLevel}[xC][yC] * (1 - 2 * \text{coeff\_sign\_flag}[n])$  |  |
|        | $\text{if}(\text{signHidden}) \{$  |  |
|        | $\text{sumAbsLevel} += \text{AbsLevel}[xC][yC]$  |  |
|        | $\text{if}((n == \text{firstSigScanPosSb}) \&\& (\text{sumAbsLevel} \% 2) == 1)$   |  |
|        | $\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC] =$<br>$-\text{TransCoeffLevel}[x0][y0][cIdx][xC][yC]$  |  |
|        | $\}$   |  |
|        | $\}$   |  |
|        | $\}$   |  |
|        | $\}$   |  |
|        | $\}$   |  |
|        | $\}$   |  |

[0136] 参照表1, last\_sig\_coeff\_x\_prefix、last\_sig\_coeff\_y\_prefix、last\_sig\_coeff\_x\_suffix和last\_sig\_coeff\_y\_suffix是用于对关联块中的非0的最后一个系数的(x,y)位置信息进行编码的语法元素。关联块可以是编码块(CB)或变换块(TB)。对于变换(和量化)和残差编码处理,CB和TB可以可互换地使用。例如,可以针对CB推导残差样本,并且可以通过针对如上所述的残差样本的变换和量化来推导(量化)变换系数,并且可以通过残差编码处理生成和发信号通知高效地表示(量化)变换系数(的位置、尺寸和符号)的信息(或语法元素)。量化变换系数可以被简称为变换系数。通常,如果CB不大于最大TB,则CB的尺寸可以等于TB的尺寸,并且在这种情况下,被变换(和量化)和残差编码的目标块可以被称为CB或TB。另外,如果CB大于最大TB,则被变换(和量化)和残差编码的目标块可以被称为TB。下文中,尽管说明以变换块(TB)为单位发信号通知与残差编码相关的语法元素,但这是示例性的,并且TB可以与如上所述的编码块(CB)可互换地使用。

[0137] 此外,根据是否对残差编码应用变换跳过,可以应用不同的残差编码方法。作为实施方式,可以使用变换跳过标志语法元素来表示是否应用变换跳过,并且可以按照变换跳过标志的语法元素transform\_skip\_flag的值对残差编码进行分支。即,基于变换跳过标志的值(基于是否应用变换跳过),可以针对残差编码使用不同的语法元素。在未应用变换跳过的情况下(即,在应用变换的情况下)使用的残差编码可以被称为常规残差编码(RRC),并且在应用变换跳过的情况下(即,在未应用变换的情况下)的残差编码可以被称为变换跳过残差编码(TSRC)。

[0138] 下表2表示基于变换跳过标志的语法元素来分支的残差编码的处理。

[0139] [表2]

|   |       |
|---|-------|
| transform unit( x0, y0, tbWidth, tbHeight, treeType, subTulIndex, chType ) {  | 描述符   |
| <b>transform skip flag</b> [ x0    y0    0 ]  | ae(v) |
| if( !transform skip flag[ x0    y0    0 ]    slice ts residual coding disabled flag )   |       |
| residual coding( x0, y0, Log2( tbWidth ), Log2( tbHeight ), 0 )   |       |
| else  |       |
| residual ts coding( x0, y0, Log2( tbWidth ), Log2( tbHeight ), 0 )  |       |
| }   |       |
| if( tu_cbf_cb[ xC    yC ] && treeType != DUAL_TREE_LUMA ) {   |       |
| if( sps_transform_skip_enabled_flag && !BdpcmFlag[ x0    y0    1 ] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag )          |       |
| <b>transform skip flag</b> [ xC    yC    1 ]  | ae(v) |
| if( !transform skip flag[ xC    yC    1 ]    slice ts residual coding disabled flag )   |       |
| residual coding( xC, yC, Log2( wC ), Log2( hC ), 1 )  |       |
| else  |       |
| residual ts coding( xC, yC, Log2( wC ), Log2( hC ), 1 )   |       |
| }   |       |
| if( tu_cbf_cr[ xC    yC ] && treeType != DUAL_TREE_LUMA && !( tu_cbf_cb[ xC    yC ] && tu_joint_cbr_residual_flag[ xC    yC ] ) ) { |       |
| if( sps_transform_skip_enabled_flag && !BdpcmFlag[ x0    y0    2 ] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag )          |       |
| <b>transform skip flag</b> [ xC    yC    2 ]  | ae(v) |
| if( !transform skip flag[ xC    yC    2 ]    slice ts residual coding disabled flag )   |       |
| residual coding( xC, yC, Log2( wC ), Log2( hC ), 2 )  |       |
| else  |       |
| residual ts coding( xC, yC, Log2( wC ), Log2( hC ), 2 )   |       |
| }   |       |
| }   |       |

[0140]

[0141] 参照上表2,在未应用变换跳过的情况下(例如,在transform\_skip\_flag的值为0的情况下),执行常规残差编码,并且这可以基于上述表1中公开的语法元素来执行。另外,在应用变换跳过的情况下(例如,在transform\_skip\_flag的值为1的情况下),执行变换跳过残差编码,并且这可以基于下表3中公开的语法元素来执行。

[0142] 下表3表示变换跳过残差编码的语法元素。

[0143] [表3]

[0144]

|  | 描述符   |
|--|-------|
| <code>residual_ts_coding( x0, y0, log2TbWidth, log2TbHeight, cldx ) {</code>                             |       |
| <code>log2SbSize = ( Min( log2TbWidth, log2TbHeight ) &lt; 2 ? 1 : 2 )</code>                            |       |
| <code>numSbCoeff = 1 &lt;&lt; ( log2SbSize &lt;&lt; 1 )</code>   |       |
| <code>lastSubBlock = ( 1 &lt;&lt; ( log2TbWidth + log2TbHeight - 2 * log2SbSize ) ) - 1</code>           |       |
| <code>inferSbCbf = 1</code>  |       |
| <code>MaxCcbs = 2 * ( 1 &lt;&lt; log2TbWidth ) * ( 1 &lt;&lt; log2TbHeight )</code>                      |       |
| <code>for( i = 0; i &lt;= lastSubBlock; i++ ) {</code>   |       |
| <code>  xS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][0]</code>                 |       |
| <code>  yS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][1]</code>                 |       |
| <code>  if( ( i != lastSubBlock    !inferSbCbf ) {</code>  |       |
| <code>    <b>coded_sub_block_flag</b>[ xS ][ yS ]</code>   | ac(v) |
| <code>  }</code>   |       |
| <code>  if( coded_sub_block_flag[ xS ][ yS ] &amp;&amp; i &lt; lastSubBlock )</code>                     |       |
| <code>    inferSbCbf = 0</code>  |       |
| <code>  /* First scan pass */</code>   |       |
| <code>    inferSbSigCoeffFlag = 1</code>   |       |
| <code>    for( n = 0; n &lt;= numSbCoeff - 1; n++ ) {</code>   |       |
| <code>      xC = ( xS &lt;&lt; log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 0 ]</code> |       |
| <code>      yC = ( yS &lt;&lt; log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 1 ]</code> |       |
| <code>      if( coded_sub_block_flag[ xS ][ yS ] &amp;&amp;</code>                                       |       |
| <code>        ( n != numSbCoeff - 1    !inferSbSigCoeffFlag ) ) {</code>                                 |       |
| <code>        <b>sig_coeff_flag</b>[ xC ][ yC ]</code>   | ac(v) |
| <code>        MaxCcbs--</code>   |       |
| <code>        if( sig_coeff_flag[ xC ][ yC ] )</code>  |       |
| <code>          inferSbSigCoeffFlag = 0</code>   |       |
| <code>      }</code>   |       |
| <code>      CoeffSignLevel[ xC ][ yC ] = 0</code>  |       |
| <code>      if( sig_coeff_flag[ xC ][ yC ] {</code>  |       |
| <code>        <b>coeff_sign_flag</b>[ n ]</code>   | ac(v) |
| <code>        MaxCcbs--</code>   |       |
| <code>        CoeffSignLevel[ xC ][ yC ] = ( coeff_sign_flag[ n ] &gt; 0 ? -1 : 1 )</code>               |       |
| <code>        <b>abs_level_gtx_flag</b>[ n ][ 0 ]</code>   | ac(v) |
| <code>        MaxCcbs--</code>   |       |
| <code>        if( abs_level_gtx_flag[ n ][ 0 ] ) {</code>  |       |
| <code>          <b>par_level_flag</b>[ n ]</code>  | ac(v) |
| <code>          MaxCcbs--</code>   |       |
| <code>        }</code>   |       |
| <code>      }</code>   |       |
| <code>      AbsLevelPassX[ xC ][ yC ] = sig_coeff_flag[ xC ][ yC ] + par_level_flag[ n ] +</code>        |       |
| <code>        abs_level_gtx_flag[ n ][ 0 ]</code>  |       |
| <code>    }</code>   |       |
| <code>  /* Greater than X scan pass (numGtXFlags=5) */</code>  |       |
| <code>  for( n = 0; n &lt;= numSbCoeff - 1; n++ ) {</code>   |       |
| <code>    xC = ( xS &lt;&lt; log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 0 ]</code>   |       |
| <code>    yC = ( yS &lt;&lt; log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 1 ]</code>   |       |
| <code>    for( j = 1; j &lt; 5; j++ ) {</code>   |       |

|        |  |       |
|--------|--|-------|
|        | if( abs_level_gtx_flag[ n ][ j - 1 ] )   |       |
|        | <b>abs_level_gtx_flag[ n ][ j ]</b>  | ae(v) |
|        | MaxCcbs--  |       |
|        | AbsLevelPassX[ xC ][ yC ] += 2 * abs_level_gtx_flag[ n ][ j ]                  |       |
|        | }  |       |
|        | }  |       |
|        | /* remainder scan pass */  |       |
|        | for( n = 0; n <= numSbCoeff - 1; n++ ) {                                       |       |
|        | xC = (xS << log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 0 ] |       |
|        | yC = (yS << log2SbSize ) + DiagScanOrder[ log2SbSize ][ log2SbSize ][ n ][ 1 ] |       |
|        | if( abs_level_gtx_flag[ n ][ 4 ] )   |       |
|        | <b>abs_remainder[ n ]</b>  | ae(v) |
|        | if( intra_bdpcm_flag == 0 ) {  |       |
|        | absRightCoeff = abs( TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC - 1 ][ yC ] )     |       |
|        | absBelowCoeff = abs( TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC ][ yC - 1 ] )     |       |
|        | predCoeff = Max( absRightCoeff, absBelowCoeff )                                |       |
| [0145] | if( AbsLevelPassX[ xC ][ yC ] + abs_remainder[ n ] == 1 && predCoeff >         |       |
|        | 0 )  |       |
|        | TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC ][ yC ] =                              |       |
|        | ( 1 - 2 * coeff_sign_flag[ n ] ) * predCoeff                                   |       |
|        | else if( AbsLevelPassX[ xC ][ yC ] + abs_remainder[ n ] <= predCoeff )         |       |
|        | TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC ][ yC ] = ( 1 - 2 *                    |       |
|        | coeff_sign_flag[ n ] ) *   |       |
|        | ( AbsLevelPassX[ xC ][ yC ] + abs_remainder[ n ] - 1 )                         |       |
|        | else   |       |
|        | TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC ][ yC ] = ( 1 - 2 *                    |       |
|        | coeff_sign_flag[ n ] ) *   |       |
|        | ( AbsLevelPassX[ xC ][ yC ] + abs_remainder[ n ] )                             |       |
|        | } else   |       |
|        | TransCoeffLevel[ x0 ][ y0 ][ cIdx ][ xC ][ yC ] = ( 1 - 2 *                    |       |
|        | coeff_sign_flag[ n ] ) *   |       |
|        | ( AbsLevelPassX[ xC ][ yC ] + abs_remainder[ n ] )                             |       |
|        | }  |       |
|        | }  |       |
|        | }  |       |

[0146] 例如,可以解析指示是否执行变换块的变换跳过的变换跳过标志,并且可以确定变换跳过标志是否为1。在变换跳过标志的值为1的情况下,如表3中所示,可以解析针对变换块的残差系数的语法元素sig\_coeff\_flag、coeff\_sign\_flag、abs\_level\_gtx\_flag、par\_level\_flag和/或abs\_remainder,并且可以基于语法元素来推导残差系数。在这种情况下,语法元素可以被依次解析,并且解析顺序可以改变。这里,abs\_level\_gtx\_flag可以表示abs\_level\_gt1\_flag、abs\_level\_gt3\_flag、abs\_level\_gt5\_flag、abs\_level\_gt7\_flag和/或abs\_level\_gt9\_flag。例如,abs\_level\_gtx\_flag[n][j]可以是表示在扫描位置n处变换系数级别的绝对值(或通过将变换系数级别向右移1位而获得的值)是否大于(j<<1)+1的标志。在某些情况下,(j<<1)+1可以被诸如第一阈值或第二阈值这样的特定阈值替换。

[0147] 另外,在变换跳过标志的值为0的情况下,如表1中所示,可以解析针对变换块的残差系数的语法元素sig\_coeff\_flag、abs\_level\_gtx\_flag、par\_level\_flag、abs\_remainder、dec\_abs\_level和coeff\_sign\_flag,并且可以基于语法元素来推导残差系数。

在这种情况下,语法元素可以被依次解析,并且解析顺序可以改变。这里,abs\_level\_gtx\_flag可以表示abs\_level\_gt1\_flag和/或abs\_level\_gt3\_flag。

[0148] 此外,如上所述,编码设备可以通过帧内/帧间/IBC/调色板预测基于预测块(预测样本)来推导残差块(残差样本),并可以通过针对推导出的残差样本应用变换和量化来推导量化变换系数。关于量化变换系数的信息(残差信息)可以被包括在残差编码语法中,并可以在编码之后以比特流的形式输出。解码设备可以从比特流获得关于量化变换系数的信息(残差信息),并可以通过对所获得的信息进行解码来推导量化变换系数。解码设备可以通过基于量化变换系数的反量化/逆变换来推导残差样本。如上所述,可以省略量化/反量化和/或变换/逆变换中的至少一者。在省略变换/逆变换的情况下,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,可以仍被称为变换系数。可以基于transform\_skip\_flag来发信号通知是否省略变换/逆变换。例如,如果transform\_skip\_flag的值为1,则它可以表示变换/逆变换被省略,并且这可以被称为变换跳过模式。

[0149] 通常,在视频/图像编码中,可以改变量化率,并且可以使用改变后的量化率来调整压缩率。从实现方式的角度来看,考虑到复杂度,可以使用量化参数(QP)而不是直接使用量化率。例如,可以使用范围为0至63的整数值的量化参数,并且量化参数的每个值可以对应于实际量化率。另外,例如,可以不同地配置亮度分量(亮度样本)的量化参数 $QP_Y$ 和色度分量(色度样本)的量化参数 $QP_C$ 。

[0150] 在量化处理中,可以通过将输入变换系数 $C$ 除以量化率 $Q_{step}$ 来获得量化变换系数 $C'$ 。在这种情况下,考虑到计算复杂度,可以通过将量化率乘以标度来得到整数形式的量化率,并且可以与对应于标度值的值一样多地执行移位操作。可以基于量化率与标度值的乘积来推导量化标度。即,可以根据QP来推导量化标度。例如,可以向变换系数 $C$ 应用量化标度,并且基于此,可以推导出量化变换系数 $C'$ 。

[0151] 反量化处理与量化处理相反,并且量化变换系数 $C'$ 可以与量化率 $Q_{step}$ 相乘,并且基于此,可以获得重构变换系数 $C''$ 。在这种情况下,可以根据量化参数来推导水平标度,并且可以向量化变换系数 $C'$ 应用水平标度,并且基于此,可以推导出重构变换系数 $C''$ 。由于变换和/或量化处理中的损失,重构变换系数 $C''$ 与初始变换系数 $C$ 略有不同。因此,编码设备甚至以与解码设备中相同的方式执行反量化。

[0152] 另外,在执行预测时,它可以是基于调色板编码的。调色板编码是用于表示包括少量唯一颜色值的块的有用技术。作为向块应用预测和变换的替代,在调色板模式中,发信号通知索引来表示每个样本的值。调色板模式对于节省视频存储器缓冲区空间是有用的。可以使用调色板模式(例如,MODE\_PLT)来对块进行编码。为了如上所述对编码块进行解码,解码器应该对调色板条目和索引进行解码。调色板条目可以由调色板表来表示,并可以由调色板表编码工具来编码。

[0153] 调色板编码可以被称为(帧内)调色板模式或(帧内)调色板编码模式。可以根据调色板编码或调色板模式来重构当前块。调色板编码可以被视为帧内编码的示例,或者可以被视为帧内预测方法之一。然而,以与上述跳过模式类似的方式,可以不发信号通知相应块的单独残差值。

[0154] 例如,在选择了调色板模式的情况下,可以发信号通知关于调色板表的信息。调色板表可以包括对应于每个像素的索引。调色板表可以从先前块中使用的像素值配置调色板

预测表。例如,先前使用的像素值可以被存储在特定缓冲区(调色板预测器)中,并且可以从缓冲区接收用于配置当前调色板的调色板预测器信息palette\_predictor\_run。即,调色板预测器可以包括表示用于当前块的调色板索引图的至少一部分的索引的数据。在用于表示当前块的调色板条目对于从调色板预测器配置的调色板预测条目来说不够的情况下,可以单独发送关于当前调色板条目的像素信息。

[0155] 调色板模式可以被在CU级发信号通知,并通常可以在CU中的大多数像素可以被表示为一组代表性像素值的情况下使用。即,在调色板模式中,CU中的样本可以被表示为一组代表性像素值。这种集合可以被称为调色板。在样本具有接近调色板中的像素值的值的情况下,可以发信号通知调色板中对应于像素值的调色板索引palette\_idx\_idc或能够指示该索引的信息(run\_copy\_flag,copy\_above\_palette\_indices\_flag)。在样本具有不包括调色板条目的像素值的情况下,样本可以被指示为转义符号,并且可以直接发信号通知量化样本值。在本文献中,像素或像素值可以被称为样本或样本值。

[0156] 为了对以调色板模式编码的块进行解码,解码器需要调色板条目信息和调色板索引信息。在调色板索引对应于转义符号的情况下,(量化)转义值可以被作为附加分量发信号通知。另外,编码器应该针对相应CU推导适当的调色板,并将调色板传送到解码器。

[0157] 为了调色板条目的高效编码,可以保持调色板预测器。可以通过SPS发信号通知调色板预测器和调色板的最大尺寸。另外,调色板预测器和调色板的最大尺寸可以是预定义的。例如,调色板预测器和调色板的最大尺寸可以根据当前块是单树还是双树而被分别定义为31和15。在VVC标准中,可以发送表示调色板模式是否被启用的sps\_palette\_enabled\_flag。然后,可以发送表示当前编码单元是否被以调色板模式编码的pred\_mode\_plt\_coding标志。调色板预测器可以在每个砖块或每个切片的开始处被初始化。

[0158] 针对调色板预测器中的每个条目,可以发信号通知重用标志,并可以表示该条目是否是当前调色板的一部分。可以使用0的游程编码来发送重用标志。此后,可以使用零阶指数哥伦布编码来发信号通知新调色板条目的数目。最后,可以发信号通知新调色板条目的分量值。在对当前CU进行编码之后,可以使用当前调色板来更新调色板预测器,并且未在当前调色板中重用的先前调色板预测器的条目可以被添加到新调色板预测器的末尾(调色板填充),直到它达到可允许的最大尺寸。

[0159] 为了对调色板索引图进行编码,可以使用水平和垂直遍历扫描对索引进行编码。可以使用标志信息(例如,palette\_transpose\_flag)从比特流显式地发信号通知扫描顺序。

[0160] 此外,可以使用两种调色板样本模式对调色板索引进行编码,并且例如,可以使用“INDEX”模式和“COPY\_ABOVE”模式。可以使用表示该调色板模式是“INDEX”模式还是“COPY\_ABOVE”模式的标志来发信号通知这种调色板模式。在这种情况下,可以在“INDEX”模式中发信号通知转义符号,并且可以分配尺寸与当前调色板尺寸相同的索引。例如,如果假定当前调色板的尺寸为10,则0至9号索引可以意指调色板中的条目索引,并且10号索引可以意指转义符号的索引。在使用水平扫描的情况下,除了顶行之外可以发信号通知该标志,并且在使用垂直扫描或者在先前模式是“COPY\_ABOVE”模式的情况下,除了第一列之外可以发信号通知该标志。在“COPY\_ABOVE”模式下,可以复制顶行中样本的调色板索引。在“INDEX”模式下,可以显式地发信号通知调色板索引。在“INDEX”模式和“COPY\_ABOVE”模式二者下,可以

发信号通知表示使用相同模式编码的接下来的样本的数目的运行值。在转义符号是“INDEX”模式或“COPY\_ABOVE”模式下运行的一部分的情况下,可以针对每个转义符号来发信号通知转义分量值。

[0161] 调色板索引的编码可以如下。首先,可以发信号通知CU的索引的数目。接下来,可以使用固定长度编码来发信号通知整个CU的实际索引。索引的数目和索引可以以旁路模式编码。据此,索引相关的旁路bin可以被分组在一起。接下来,调色板样本模式copy\_above\_palette\_indices\_flag和运行可以以交错方式发信号通知。最后,与整个CU的转义样本对应的分量转义值可以被分组在一起,并可以以旁路模式被编码。

[0162] 下表4表示包括与用于编码单元的基于调色板模式的编码相关的语法元素的语法结构的示例,并且下表5表示表4的语法中所包括的语法元素的语义。

[0163] [表4]

[0164]

|   | 描述符   |
|---|-------|
| palette_coding( x0, y0, cbWidth, cbHeight, treeType ) {   |       |
| startComp = ( treeType == DUAL_TREE_CHROMA ) ? 1 : 0  |       |
| numComps = ( treeType == SINGLE_TREE ) ? ( sps_chroma_format_idc == 0 ? 1 : 3 ) : ( treeType == DUAL_TREE_CHROMA ) ? 2 : 1  |       |
| maxNumPaletteEntries = ( treeType == SINGLE_TREE ) ? 31 : 15  |       |
| palettePredictionFinished = 0   |       |
| NumPredictedPaletteEntries = 0  |       |
| for( predictorEntryIdx = 0; predictorEntryIdx < PredictorPaletteSize[ startComp ] && !palettePredictionFinished && NumPredictedPaletteEntries < maxNumPaletteEntries; predictorEntryIdx++ ) { |       |
| <b>palette_predictor_run</b>  | ac(v) |
| if( palette_predictor_run != 1 ) {  |       |
| if( palette_predictor_run > 1 )   |       |
| predictorEntryIdx += palette_predictor_run - 1  |       |
| PalettePredictorEntryReuseFlags[ predictorEntryIdx ] = 1  |       |
| NumPredictedPaletteEntries++  |       |
| } else  |       |
| palettePredictionFinished = 1   |       |
| }   |       |
| if( NumPredictedPaletteEntries < maxNumPaletteEntries )   |       |
| <b>num_signalled_palette_entries</b>  | ac(v) |
| for( cIdx = startComp; cIdx < ( startComp + numComps ); cIdx++ )  |       |
| for( i = 0; i < num_signalled_palette_entries; i++ )  |       |
| <b>new_palette_entries[ cIdx ][ i ]</b>   | ac(v) |
| if( CurrentPaletteSize[ startComp ] > 0 )   |       |
| <b>palette_escape_val_present_flag</b>  | ac(v) |
| if( MaxPaletteIndex > 0 ) {   |       |
| adjust = 0  |       |
| <b>palette_transpose_flag</b>   | ac(v) |
| }   |       |
| if( treeType != DUAL_TREE_CHROMA && palette_escape_val_present_flag )   |       |
| if( pps_cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded ) {   |       |
| <b>cu_qp_delta_abs</b>  | ac(v) |
| if( cu_qp_delta_abs )   |       |
| <b>cu_qp_delta_sign_flag</b>  | ac(v) |
| }   |       |
| if( treeType != DUAL_TREE_LUMA && palette_escape_val_present_flag )   |       |
| if( sh_cu_chroma_qp_offset_enabled_flag && !IsCuChromaQpOffsetCoded ) {   |       |
| <b>cu_chroma_qp_offset_flag</b>   | ac(v) |
| if( cu_chroma_qp_offset_flag && pps_chroma_qp_offset_list_len_minus1 > 0 )  |       |
| <b>cu_chroma_qp_offset_idx</b>  | ac(v) |
| }   |       |
| PreviousRunPosition = 0   |       |
| PreviousRunType = 0   |       |
| for( subSetId = 0; subSetId <= ( cbWidth * cbHeight - 1 ) / 16; subSetId++ ) {  |       |
| minSubPos = subSetId * 16   |       |

|        |   |       |
|--------|---|-------|
|        | if( minSubPos + 16 > cbWidth * cbHeight)                                |       |
|        | maxSubPos = cbWidth * cbHeight  |       |
|        | else  |       |
|        | maxSubPos = minSubPos + 16  |       |
|        | RunCopyMap[ x0 ][ y0 ] = 0  |       |
|        | PaletteScanPos = minSubPos  |       |
|        | log2CbWidth = Log2( cbWidth )   |       |
|        | log2CbHeight = Log2( cbHeight )   |       |
|        | while( PaletteScanPos < maxSubPos ) {                                   |       |
|        | xC = x0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][0] |       |
|        | yC = y0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][1] |       |
|        | if( PaletteScanPos > 0 ) {  |       |
|        | xcPrev = x0 +   |       |
|        | TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]       |       |
|        | ycPrev = y0 +   |       |
|        | TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][1]       |       |
|        | }   |       |
|        | if( MaxPaletteIndex > 0 && PaletteScanPos > 0 ) {                       |       |
|        | <b>run_copy_flag</b>  | ae(v) |
|        | RunCopyMap[ xC ][ yC ] = run_copy_flag                                  |       |
|        | }   |       |
|        | CopyAboveIndicesFlag[ xC ][ yC ] = 0                                    |       |
| [0165] | if( MaxPaletteIndex > 0 && !RunCopyMap[ xC ][ yC ] ) {                  |       |
|        | if( ( !palette_transpose_flag && yC > y0 )                              |       |
|        | ( palette_transpose_flag && xC > x0 ) ) &&                              |       |
|        | CopyAboveIndicesFlag[ xcPrev ][ ycPrev ] == 0 &&                        |       |
|        | PaletteScanPos > 0 ) {  |       |
|        | <b>copy_above_palette_indices_flag</b>                                  | ae(v) |
|        | CopyAboveIndicesFlag[ xC ][ yC ] = copy_above_palette_indices_flag      |       |
|        | }   |       |
|        | PreviousRunType = CopyAboveIndicesFlag[ xC ][ yC ]                      |       |
|        | PreviousRunPosition = PaletteScanPos                                    |       |
|        | } else if( PaletteScanPos > 0 )   |       |
|        | CopyAboveIndicesFlag[xC][yC] = CopyAboveIndicesFlag[xcPrev][ycPrev]     |       |
|        | PaletteScanPos ++   |       |
|        | }   |       |
|        | PaletteScanPos = minSubPos  |       |
|        | while( PaletteScanPos < maxSubPos ) {                                   |       |
|        | xC = x0 +   |       |
|        | TraverseScanOrder[ log2CbWidth ][ log2CbHeight ][ PaletteScanPos ][ 0 ] |       |
|        | yC = y0 +   |       |
|        | TraverseScanOrder[ log2CbWidth ][ log2CbHeight ][ PaletteScanPos ][ 1 ] |       |
|        | if( PaletteScanPos > 0 ) {  |       |
|        | xcPrev =x0 +  |       |
|        | TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]       |       |

|        |   |       |
|--------|---|-------|
|        | ycPrev = y0 +<br>TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][1]                      |       |
|        | }   |       |
|        | if( MaxPaletteIndex > 0 && !RunCopyMap[ xC ][ yC ] &&<br>CopyAboveIndicesFlag[ xC ][ yC ] == 0 ) {      |       |
|        | if( MaxPaletteIndex - adjust > 0 )  |       |
|        | <b>palette_idx_idc</b>  | ae(v) |
|        | adjust = 1  |       |
|        | }   |       |
|        | if( !RunCopyMap[ xC ][ yC ] && CopyAboveIndicesFlag[ xC ][ yC ] == 0 )                                  |       |
|        | CurrPaletteIndex = palette_idx_idc  |       |
|        | if( CopyAboveIndicesFlag[ xC ][ yC ] == 0 )   |       |
|        | PaletteIndexMap[ xC ][ yC ] = CurrPaletteIndex  |       |
|        | else if( !palette_transpose_flag )  |       |
|        | PaletteIndexMap[ xC ][ yC ] = PaletteIndexMap[ xC ][ yC - 1 ]   |       |
|        | else  |       |
|        | PaletteIndexMap[ xC ][ yC ] = PaletteIndexMap[ xC - 1 ][ yC ]   |       |
|        | PaletteScanPos ++   |       |
| [0166] | }   |       |
|        | if( palette_escape_val_present_flag ) {   |       |
|        | for( cIdx = startComp; cIdx < ( startComp + numComps ); cIdx++ ) {                                      |       |
|        | for( sPos = minSubPos; sPos < maxSubPos; sPos++ ) {   |       |
|        | xC = x0 + TraverseScanOrder[ log2CbWidth ][ log2CbHeight ][ sPos ][ 0 ]                                 |       |
|        | yC = y0 + TraverseScanOrder[ log2CbWidth ][ log2CbHeight ][ sPos ][ 1 ]                                 |       |
|        | if( !( treeType == SINGLE_TREE && cIdx != 0 &&<br>( xC % SubWidthC != 0    yC % SubHeightC != 0 ) ) ) { |       |
|        | if( PaletteIndexMap[ cIdx ][ xC ][ yC ] == MaxPaletteIndex ) {  |       |
|        | <b>palette_escape_val</b>   | ae(v) |
|        | PaletteEscapeVal[ cIdx ][ xC ][ yC ] = palette_escape_val   |       |
|        | }   |       |
|        | }   |       |
|        | }   |       |
|        | }   |       |
|        | }   |       |
|        | }   |       |

[0167] [表5]

[0168] 在下面的语义中，数组索引 x0, y0 指明相对于图片的左上亮度样本的所考虑的编码块的左上亮度样本的位置 (x0, y0)。当 treeType 等于 SINGLE\_TREE 或 DUAL\_TREE\_LUMA 时，数组索引 xC, yC 指明相对于图片的左上亮度样本的样本的位置 (xC, yC)；并且当 treeType 等于 DUAL\_TREE\_CHROMA 时，数组索引 xC, yC 指明相对于图片的左上色度样本的样本的位置 (xC, yC)。数组索引 startComp 指明当前调色板表的第一颜色分量。等于 0 的 startComp 指示 Y 分量；等于 1 的 startComp 指示 Cb 分量；等于 2 的 startComp 指示 Cr 分量。numComps 指明当前调

色板表中的颜色分量数目。

预测器调色板由来自用于预测当前调色板中的条目的先前编码单元的调色板条目组成。

`PredictorPaletteSize[startComp]`指明当前调色板表的第一颜色分量 `startComp` 的预测器调色板的尺寸。如子条款 8.4.5.3 中指定地推导 `PredictorPaletteSize[startComp]`。

等于 1 的 `PalettePredictorEntryReuseFlag[i]`指明在当前调色板中重用预测器调色板中的第 `i` 条目。等于 0 的 `PalettePredictorEntryReuseFlag[i]`指明预测器调色板中的第 `i` 条目不是当前调色板中的条目。数组 `PalettePredictorEntryReuseFlags[i]`中的所有元素被初始化为 0。

**palette\_predictor\_run** 用于确定数组 `PalettePredictorEntryReuseFlags` 中的非零条目之前的零的数目。

`palette_predictor_run` 的值应当在 0 至  $(\text{PredictorPaletteSize}[\text{startComp}] - \text{predictorEntryIdx})$  的范围中（包括端值）是比特流一致性的要求，其中，`predictorEntryIdx` 对应于数组 `PalettePredictorEntryReuseFlags` 中的当前位置。变量 `NumPredictedPaletteEntries` 指明从预测器调色板重用的当前调色板中的条目的数目。`NumPredictedPaletteEntries` 的值应当在 0 至 `maxNumPaletteEntries` 的范围内，包括端值。

**num\_signalled\_palette\_entries** 指明针对当前调色板表的第一颜色分量 `startComp` 显式发信号通知的当前调色板中的条目的数目。

当 `num_signalled_palette_entries` 不存在时，推断其等于 0。

变量 `CurrentPaletteSize[startComp]`指明针对当前调色板表的第一颜色分量 `startComp` 的当前调色板的尺寸，并被如下地推导：

$$\text{CurrentPaletteSize}[\text{startComp}] = \text{NumPredictedPaletteEntries} + \text{num\_signalled\_palette\_entries} \quad (176)$$

`CurrentPaletteSize[startComp]`的值应当在 0 至 `maxNumPaletteEntries` 的范围内，包括端值。

**new\_palette\_entries[cIdx][i]**指明针对颜色分量 `cIdx` 的第 `i` 个发信号通知的调色板条目的值。

[0169]

变量 LocalDualTreeFlag 被如下地推导：

```
LocalDualTreeFlag = ( treeType != SINGLE_TREE &&
  ( sh_slice_type != I || ( sh_slice_type == I &&
    sps_qtbt_dual_tree_intra_flag == 0 ) ) ) ? 1 : 0      (177)
```

变量 PredictorPaletteEntries[cIdx][i] 指明针对颜色分量 cIdx 的预测器调色板中的第 i 元素。

变量 CurrentPaletteEntries[cIdx][i] 指明针对颜色分量 cIdx 的当前调色板中的第 i 元素，并被如下地推导：

```
numPredictedPaletteEntries = 0
for( i = 0; i < PredictorPaletteSize[ startComp ]; i++ )
  if( PalettePredictorEntryReuseFlags[ i ] ) {
    for( cIdx = LocalDualTreeFlag ? 0 : startComp;
      cIdx < LocalDualTreeFlag ? 3 : ( startComp + numComps ); cIdx++ )
      CurrentPaletteEntries[ cIdx ][ numPredictedPaletteEntries ] =
        PredictorPaletteEntries[ cIdx ][ i ]
    numPredictedPaletteEntries++
  }
for( cIdx = startComp; cIdx < ( startComp + numComps ); cIdx++ )      (178)
  for( i = 0; i < num_signalled_palette_entries; i++ )
    CurrentPaletteEntries[ cIdx ][ numPredictedPaletteEntries + i ] =
      new_palette_entries[ cIdx ][ i ]
```

[0170]

等于 1 的 **palette\_escape\_val\_present\_flag** 指明当前编码单元包含至少一个转义编码样本。等于 0 的 **palette\_escape\_val\_present\_flag** 指明当前编码单元中不存在转义编码样本。当不存在时，推断 **palette\_escape\_val\_present\_flag** 的值等于 1。

变量 MaxPaletteIndex 指明当前编码单元的调色板索引的最大可能值。MaxPaletteIndex 的值被设置为等于 CurrentPaletteSize[startComp]-1+palette\_escape\_val\_present\_flag。

**palette\_idx\_idc** 是针对调色板表 CurrentPaletteEntries 的索引的指示。palette\_idx\_idc 的值对于块中的第一索引而言应当在 0 至 MaxPaletteIndex 的范围内，包括端值，而对于块中的其余索引而言应当在 0 至 (MaxPaletteIndex-1) 的范围内，包括端值。

当不存在 `palette_idx_idc` 时，推断其等于 0。

等于 1 的 `palette_transpose_flag` 指明应用垂直遍历扫描来扫描当前编码单元中的样本的索引。等于 0 的 `palette_transpose_flag` 指明应用水平遍历扫描来扫描当前编码单元中的样本的索引。当不存在时，推断 `palette_transpose_flag` 的值等于 0。

数组 `TraverseScanOrder` 指明用于调色板编码的扫描顺序数组。如果 `palette_transpose_flag` 等于 0，则 `TraverseScanOrder` 被指派水平扫描顺序 `HorTravScanOrder`。否则（`palette_transpose_flag` 等于 1），`TraverseScanOrder` 被指派垂直扫描顺序 `VerTravScanOrder`。

等于 1 的 `run_copy_flag` 指明如果 `CopyAboveIndicesFlag[xC][yC]` 等于 0，则调色板索引与先前扫描位置处的索引相同，并且调色板运行类型与先前扫描位置处的运行类型相同。

[0171]

否则，等于 0 的 `run_copy_flag` 指明调色板运行类型与先前扫描位置处的运行类型不同。

等于 1 的 `copy_above_palette_indices_flag` 指明如果使用水平遍历扫描则调色板索引等于上行中的相同位置处的调色板索引，或者如果使用垂直遍历扫描则调色板索引等于左列中的相同位置处的调色板索引。

等于 0 的 `copy_above_palette_indices_flag` 指明样本的调色板索引的指示被在比特流中编码或被推断出。

等于 1 的变量 `CopyAboveIndicesFlag[xC][yC]` 指明从上行（水平扫描）或左列（垂直扫描）中的调色板索引复制调色板索引。等于 0 的 `CopyAboveIndicesFlag[xC][yC]` 指明调色板索引被在比特流中显式地编码或推断出。

变量 `PaletteIndexMap[xC][yC]` 指明作为针对由 `CurrentPaletteEntries` 表示的数组

的索引的调色板索引。PaletteIndexMap[xC][yC]的值应当在 0 至 MaxPaletteIndex 的范围内，包括端值。

变量 adjustedRefPaletteIndex 被如下地推导：

```
adjustedRefPaletteIndex = MaxPaletteIndex + 1
if( PaletteScanPos > 0 ) {
    xcPrev = x0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][0]
    ycPrev = y0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][1]
    if( CopyAboveIndicesFlag[ xcPrev ][ ycPrev ] == 0 )
        adjustedRefPaletteIndex = PaletteIndexMap[xcPrev][ycPrev]    (179)
    else {
        if( !palette_transpose_flag )
            adjustedRefPaletteIndex = PaletteIndexMap[ xC ][ yC - 1 ]
        else
            adjustedRefPaletteIndex = PaletteIndexMap[ xC - 1 ][ yC ]
    }
}
else
    adjustedRefPaletteIndex = PaletteIndexMap[ xC - 1 ][ yC ]
}
```

[0172]

当 CopyAboveIndicesFlag[xC][yC]等于 0 时，变量 CurrPaletteIndex 被如下地推导：

```
if( CurrPaletteIndex >= adjustedRefPaletteIndex )
    CurrPaletteIndex++    (180)
```

**palette\_escape\_val** 指明针对分量的量化转义编码样本值。

变量 PaletteEscapeVal[cIdx][xC][yC] 指明 PaletteIndexMap[xC][yC] 等于 MaxPaletteIndex 并且 palette\_escape\_val\_present\_flag 等于 1 的样本的转义值。数组索引 cIdx 指明颜色分量。

PaletteEscapeVal[cIdx][xC][yC]应当在 0 至  $(1 \ll \text{BitDepth}_C) - 1$  的范围内（包括端值）是比特流一致性的要求。

[0173] 参照以上表4和表5,在对当前块(即,当前编码单元)应用调色板模式的情况下,可以解析/发信号通知如上表4中的调色板编码语法(例如,palette\_coding())。

[0174] 例如,可以基于调色板条目信息来配置调色板表。调色板条目信息可以包括诸如 palette\_predictor\_run、num\_signalled\_palette\_entries和new\_palette\_entries这样的语法元素。

[0175] 另外,可以基于调色板索引信息来配置针对当前块的调色板索引图。调色板索引

信息可以包括诸如num\_palette\_indices\_minus1、palette\_idx\_idc和palette\_transpose\_flag这样的语法元素。基于如上所述的调色板索引信息,可以通过在按照遍历扫描方向(垂直方向或水平方向)循环的同时推导当前块中的样本的调色板索引(例如,PaletteIndexIdc)来配置调色板索引图(例如,PaletteIndexMap)。

[0176] 另外,基于调色板索引图,可以推导调色板表中的调色板条目的样本值,并且可以基于映射到调色板条目上的样本值来生成当前块的重构样本。

[0177] 另外,在当前块中存在具有转义值的样本的情况下(即,在palette\_escape\_val\_present\_flag的值为1的情况下),可以基于转义信息来推导当前块的转义值。转义信息可以包括诸如palette\_escape\_val\_present\_flag和palette\_escape\_val这样的语法元素。例如,基于量化后的转义值信息(例如,palette\_escape\_val),可以推导当前块中的转义编码样本的转义值。可以基于转义值来生成当前块的重构样本。

[0178] 此外,在编码/解码处理中,可以使用块差分脉冲编码调制或基于块的增量脉冲编码调制(BDPCM)技术。BDPCM可以被命名为基于量化残差块的增量脉冲编码调制(RDPCM)。

[0179] 在通过应用BDPCM来预测块的情况下,可以利用重构样本以便逐行地预测块的行或列。在这种情况下,所使用的参考样本可以是未经滤波的样本。BDPCM方向可以表示是使用了垂直方向预测还是水平方向预测。即,在应用BDPCM的情况下,垂直方向或水平方向可以被选择为BDPCM方向,并且可以在BDPCM方向上执行预测。可以在空间域中量化预测误差,并且可以通过向预测(即,预测样本)添加反量化后的预测误差来重构样本。预测误差可能意指残差。量化残差域BDPCM可以被提议为BDPCM的替代物,并且预测方向或信令可以等于应用于空间域的BDPCM。即,在量化系数本身通过量化残差域BDPCM如同增量脉冲编码调制(DPCM)一样被建立之后,可以通过反量化来重构残差。因此,量化残差域BDPCM可以用作残差编码端应用DPCM的含义。下文中使用的量化残差域是通过在不进行变换的情况下对基于预测推导出的残差进行量化来获得的,并意指用于量化残差样本的域。例如,量化残差域可以包括被应用变换跳过即针对残差样本跳过变换但对其应用量化的量化残差(或量化残差系数)。另外,例如,量化残差域可以包括量化变换系数。

[0180] 如上所述,BDPCM可以应用于量化残差域,量化残差域可以包括量化残差(或量化残差系数),并且在这种情况下,可以针对残差应用变换跳过。即,在应用BDPCM的情况下,可以跳过变换,并且可以针对残差样本应用量化。另外,量化残差域可以包括量化变换系数。可以在序列级(SPS)发信号通知表示BDPCM是否适用的标志,并且可以仅在被发信号通知在SPS处变换跳过模式可能的情况下才发信号通知该标志。该标志可以被称为BDPCM启用标志或SPS BDPCM启用标志。

[0181] 在应用BDPCM的情况下,可以根据与帧内预测方向类似的预测方向(例如,垂直预测或水平预测)通过样本复制来针对整个块执行帧内预测。可以通过跳过变换来量化作为原始块与预测块之间的差值的残差,并且可以对增量值即量化残差与预测器在水平或垂直方向上的差值(即,在水平或垂直方向上的量化残差)进行编码。

[0182] 如果BDPCM适用,则CU尺寸可以等于或小于亮度样本的MaxTsSize(最大变换跳过块尺寸),并且在通过帧内预测对CU进行编码的情况下,可以在CU级发送标志信息。标志信息可以被称为BDPCM标志。这里,MaxTsSize可以意味着允许的变换跳过模式的最大块尺寸。标志信息可以指示是应用典型帧内编码还是应用BDPCM。如果应用了BDPCM,则可以发送指

示预测方向是水平方向还是垂直方向的BDPCM预测方向标志。BDPCM预测方向标志可以被称为BDPCM方向标志。此后,可以通过使用未滤波参考样本的典型水平或垂直帧内预测处理来预测块。另外,可以量化残差,并且可以对量化残差和预测器之间(例如,根据BDPCM预测方向在水平或垂直方向上的周围位置的已经量化的残差之间)的差值进行编码。

[0183] 此外,如上所述,本文献中公开的语法表中的信息(语法元素)可以被包括在图像/视频信息中,并可以由编码设备配置/编码,以便以比特流的形式传送到解码设备。解码设备可以解析/解码相应语法表中的信息(语法元素)。解码设备可以基于解码信息对当前块执行解码处理(预测、(基于变换跳过的)残差处理、BDPCM和调色板编码)。

[0184] 下文中,在本文献中,提出了用于解析/发信号通知对变换跳过相关高级语法元素和/或调色板编码相关高级语法元素有依赖性的语法元素的有效方案。即,根据本文献的实施方式,在视频/图像编码期间,可以按照在执行变换跳过和/或调色板编码时使用的绝对必要或辅助的信息的依赖性和非依赖性对是否执行编码进行分类,因此可以执行高效编码。

[0185] 在视频编码中,编码工具的切换可以在特定的高级语法(HLS)中定义。在相关技术中的VVC的情况下,可以在SPS中定义关于相应编码工具的标志信息。另外,在VVC中,标准化一直在朝着各个高级语法集(例如,视频参数集(VPS)、序列参数集(SPS)、图片参数集(PPS)、自适应参数集(APS)、解码参数集(DPS)和切片头)之间的独立性的方向发展。因此,在存在编码工具的标志的高级语法集中,存在具有依赖性的多个语法元素。在本文献的实施方式中,提出了用于按照变换跳过和/或调色板编码来解析/发信号通知具有依赖性的高级语法元素的方法。

[0186] 作为实施方式,本文献提出了用于通过使得对变换跳过相关高级语法元素有依赖性的语法元素能够根据依赖性条件确定是否执行解析/发信号通知来节省发送的比特的方法。作为示例,提出了用于解析具有依赖于变换跳过(启用)标志是否使用变换跳过的依赖性的高级语法元素的方法。

[0187] 例如,作为依赖于基于变换跳过的编码的语法元素,存在变换跳过(启用)标志(例如,sps\_transform\_skip\_enabled\_flag)、用于变换跳过的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)以及关于是否应用BDPCM的信息(例如,sps\_bdpcm\_enabled\_flag)。作为示例,如果变换跳过(启用)标志的值被定义为1,则应该有必要发送相关标志或信息语法元素,而如果变换跳过(启用)标志的值被定义为0,则可以不发送除了变换跳过(启用)标志语法元素之外的语法元素。

[0188] 即,提出了用于根据高级语法HLS(例如,VPS、SPS、PPS、APS、DPS和切片头)中的变换跳过(启用)标志的值来发送诸如用于变换跳过块的最小量化参数信息以及在变换跳过期间是否应用BDPCM这样的依赖于是否执行变换跳过的高级语法元素的方法。另外,所提出的方法不限于本实施方式中提到的语法元素,而是可以包括在具有依赖于是否执行变换跳过的依赖性并包括变换跳过(启用)标志的高级语法集中定义的所有高级语法元素。

[0189] 如上所述,与基于变换跳过的编码相关的语法元素可以在高级语法集中定义,并可以在序列参数集(SPS)中定义,如下表6中的实施方式中一样。

[0190] [表6]

|        |   |       |
|--------|---|-------|
|        | seq_parameter_set_rbsp() {              | 描述符   |
|        | (...)                                   |       |
|        | <b>sps_transform_skip_enabled_flag</b>  | u(1)  |
|        | if( sps_transform_skip_enabled_flag ) { |       |
| [0191] | <b>sps_bdpcm_enabled_flag</b>           | u(1)  |
|        | <b>min_qp_prime_ts_minus4</b>           | ue(v) |
|        | }                                       |       |
|        | (...)                                   |       |
|        | rbsp_trailing_bits()                    |       |
|        | }                                       |       |

[0192] 另外,例如,SPS语法的语法元素当中的上述实施方式的语法元素的语义可以如下表7中表示的。

[0193] [表7]

|        |  |
|--------|--|
| [0194] | <p>等于 1 的 <b>sps_transform_skip_enabled_flag</b> 指明在变换单元语法中可以存在 transform_skip_flag。等于 0 的 sps_transform_skip_enabled_flag 指明在变换单元语法中不存在 transform_skip_flag。</p> <p>等于 1 的 <b>sps_bdpcm_enabled_flag</b> 指明在针对帧内编码单元的编码单元语法中可以存在 intra_bdpcm_flag。等于 0 的 sps_bdpcm_enabled_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra_bdpcm_flag。当不存在时,推断 sps_bdpcm_enabled_flag 的值等于 0。</p> <p><b>min_qp_prime_ts_minus4</b> 指明变换跳过模式的最小允许量化参数如下:</p> $QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_minus4}$ |
|--------|--|

[0195] 参照以上表6和表7,与变换跳过相关的语法元素可以在SPS中定义,并可以包括语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag和min\_qp\_prime\_ts\_minus4。

[0196] sps\_transform\_skip\_enabled\_flag语法元素可以基于其值是0还是1来表示是否启用变换跳过。例如,如果sps\_transform\_skip\_enabled\_flag的值为1,则它可以表示启用了变换跳过,并且在这种情况下,可以通过变换单元语法来解析/发信号通知transform\_skip\_flag。这里,transform\_skip\_flag语法元素可以表示是否可以对相应的关联变换块应用变换。如果sps\_transform\_skip\_enabled\_flag的值为0,则它可以表示未启用变换跳过,并且在这种情况下,可以不在变换单元语法中解析/发信号通知transform\_skip\_flag。换句话说,基于变换跳过启用标志sps\_transform\_skip\_enabled\_flag,可以表示变换单元语法中是否存在transform\_skip\_flag。

[0197] sps\_bdpcm\_enabled\_flag语法元素可以基于其值是0还是1来表示是否启用BDPCM。例如,如果sps\_bdpcm\_enabled\_flag的值为1,则它可以表示启用BDPCM,并且在这种情况下,可以通过帧内编码单元的编码单元语法来解析/发信号通知intra\_bdpcm\_flag(或

intra\_bdpcm\_luma\_flag和intra\_bdpcm\_chroma\_flag)。这里,intra\_bdpcm\_flag语法元素可以表示BDPCM是否应用于当前编码块。如果sps\_bdpcm\_enabled\_flag的值为0,则它可以表示未启用BDPCM,并且在这种情况下,可以不在帧内编码单元的编码单元语法中解析/发信号通知intra\_bdpcm\_flag(或intra\_bdpcm\_luma\_flag和intra\_bdpcm\_chroma\_flag)。换句话说,可以基于BDPCM启用标志sps\_bdpcm\_enabled\_flag来表示编码单元语法中是否存在intra\_bdpcm\_flag(或intra\_bdpcm\_luma\_flag和intra\_bdpcm\_chroma\_flag)。

[0198] min\_qp\_prime\_ts\_minus4语法元素可以表示针对变换跳过模式允许的最小允许量化参数。例如,基于min\_qp\_prime\_ts\_minus4语法元素,可以推导出变换跳过模式下的最小量化参数值(例如,QpPrimeTsMin)。基于变换跳过模式下的最小量化参数,可以推导出在缩放处理(反量化处理)中使用的量化参数。另外,可以通过基于量化参数对当前块执行缩放处理(反量化处理)来推导缩放变换系数(反量化变换系数),并且基于此,可以推导出当前块的残差样本。

[0199] 另外,在SPS中,可以定义对与变换跳过相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)有依赖性的语法元素。例如,如以上表6和表7中公开的,基于SPS中的变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值,表示变换跳过模式下变换跳过块的最小量化参数信息的min\_qp\_prime\_ts\_minus4语法元素和表示是否启用BDPCM的sps\_bdpcm\_enabled\_flag语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则可以解析/发信号通知min\_qp\_prime\_ts\_minus4语法元素和sps\_bdpcm\_enabled\_flag语法元素。另外,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不解析/发信号通知min\_qp\_prime\_ts\_minus4语法元素和sps\_bdpcm\_enabled\_flag语法元素。

[0200] 另外,作为实施方式,本文献提出了用于通过使得对变换跳过相关高级语法元素有依赖性的语法元素能够根据依赖性条件确定是否执行解析/发信号通知来节省发送的比特的方法。作为示例,提出了用于解析具有依赖于变换跳过(启用)标志是否使用变换跳过的依赖性的高级语法元素的方法。

[0201] 例如,作为依赖于基于变换跳过的编码的语法元素,存在变换跳过(启用)标志(例如,sps\_transform\_skip\_enabled\_flag)、关于变换跳过应用尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)、用于变换跳过的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)以及关于是否应用BDPCM的信息(例如,sps\_bdpcm\_enabled\_flag)。作为示例,如果变换跳过(启用)标志的值被定义为1,则应该有必要发送相关标志或信息语法元素,而如果变换跳过(启用)标志的值被定义为0,则可以不发送除了变换跳过(启用)标志语法元素之外的语法元素。

[0202] 即,提出了用于根据高级语法HLS(例如,VPS、SPS、PPS、APS、DPS和切片头)中的变换跳过(启用)标志的值来发送诸如关于变换跳过应用的最大尺寸的信息、最小量化参数信息以及在变换跳过期间是否应用BDPCM这样的依赖于是否执行变换跳过的高级语法元素的方法。另外,所提出的方法不限于本实施方式中提到的语法元素,而是可以包括在具有依赖于是否执行变换跳过的依赖性并包括变换跳过(启用)标志的高级语法集中定义的所有高级语法元素。

[0203] 如上所述,与基于变换跳过的编码相关的语法元素可以在高级语法集中定义,并可以在序列参数集 (SPS) 中定义,如下表8的实施方式中一样。然而,在相关技术中在图片参数集 (PPS) 中定义的变换跳过的最大块尺寸信息可以在SPS中新定义,以避免HLS之间的依赖性,并且这可以如在下表8中表示的。

[0204] [表8]

|        |  |       |
|--------|--|-------|
|        | seq_parameter_set_rbsp() {                 | 描述符   |
|        | (...)                                      |       |
|        | <b>sps_transform_skip_enabled_flag</b>     | u(1)  |
|        | if( sps_transform_skip_enabled_flag ) {    |       |
|        | <b>sps_bdpcm_enabled_flag</b>              | u(1)  |
| [0205] | <b>min_qp_prime_ts_minus4</b>              | ue(v) |
|        | <b>log2_transform_skip_max_size_minus2</b> | ue(v) |
|        | }  |       |
|        | (...)                                      |       |
|        | rbsp_trailing_bits( )                      |       |
|        | }  |       |

[0206] 另外,例如,SPS语法的语法元素当中的上述实施方式的语法元素的语义可以如下表9中表示的。

[0207] [表9]

等于 1 的 **sps\_transform\_skip\_enabled\_flag** 指明在变换单元语法中可以存在 transform\_skip\_flag。等于 0 的 sps\_transform\_skip\_enabled\_flag 指明在变换单元语法中不存在 transform\_skip\_flag。

等于 1 的 **sps\_bdpcm\_enabled\_flag** 指明在针对帧内编码单元的编码单元语法中可以存在 intra\_bdpcm\_flag。等于 0 的 sps\_bdpcm\_enabled\_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra\_bdpcm\_flag。当不存在时，推断 sps\_bdpcm\_enabled\_flag 的值等于 0。

[0208]

**min\_qp\_prime\_ts\_minus4** 指明变换跳过模式的最小允许量化参数如下：

$$QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_minus4}$$

**log2\_transform\_skip\_max\_size\_minus2** 指明用于变换跳过的最大块尺寸，并应当在 0 至 3 的范围内。

当不存在时，推断 log2\_transform\_skip\_max\_size\_minus2 的值等于 0。

变量 MaxTsSize 被设置为等于  $1 \ll (\text{log2\_transform\_skip\_max\_size\_minus2} + 2)$ 。

[0209] 参照以上表8和表9，与变换跳过相关的语法元素可以在SPS中定义，并可以包括语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag、min\_qp\_prime\_ts\_minus4和log2\_transform\_skip\_max\_size\_minus2。

[0210] 这里，由于语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag和min\_qp\_prime\_ts\_minus4已经在上表6和表7中详细说明，因此在本实施方式中，为了便于说明，将省略对其的详细说明。

[0211] log2\_transform\_skip\_max\_size\_minus2语法元素可以表示在变换跳过模式中使用的最大块尺寸。在这种情况下，log2\_transform\_skip\_max\_size\_minus2语法元素可以在0至3的范围内。例如，如以上表9中公开的，在变换跳过模式中使用的最大块尺寸（例如，MaxTsSize）可以基于诸如 $1 \ll (\text{log2\_transform\_skip\_max\_size\_minus2} + 2)$ 这样的计算来推导。

[0212] 另外，在SPS中，可以定义对与变换跳过相关的语法元素当中的变换跳过启用标志语法元素（例如，sps\_transform\_skip\_enabled\_flag）有依赖性的语法元素。例如，如以上表8和表9中公开的，在SPS中，基于变换跳过启用标志（例如，sps\_transform\_skip\_enabled\_flag）的值表示是否启用BDPCM的sps\_bdpcm\_enabled\_flag语法元素、表示变换跳过模式下变换跳过块的最小量化参数信息的min\_qp\_prime\_ts\_minus4语法元素和表示在变换跳过模式下使用的最大块尺寸的log2\_transform\_skip\_max\_size\_minus2语法元素可以具有依赖性。作为示例，如果变换跳过启用标志（例如，sps\_transform\_skip\_enabled\_

flag)的值为1,则可以解析/发信号通知语法元素sps\_bdpcm\_enabled\_flag、min\_qp\_prime\_ts\_minus4和log2\_transform\_skip\_max\_size\_minus2。另外,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不解析/发信号通知语法元素sps\_bdpcm\_enabled\_flag、min\_qp\_prime\_ts\_minus4和log2\_transform\_skip\_max\_size\_minus2。

[0213] 另外,作为实施方式,本文献提出了用于通过使得对变换跳过相关高级语法元素和调色板编码相关高级语法元素有依赖性的语法元素能够根据依赖性条件确定是否执行解析/发信号通知来节省发送的比特的方法。作为示例,提出了用于通过变换跳过(启用)标志和/或调色板编码(启用)标志来解析具有依赖性的高级语法元素的方法。

[0214] 例如,作为依赖于基于变换跳过的编码的语法元素,存在变换跳过(启用)标志(例如,sps\_transform\_skip\_enabled\_flag)、关于变换跳过应用尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)、用于变换跳过的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)以及关于是否应用BDPCM的信息(例如,sps\_bdpcm\_enabled\_flag)。另外,如上所述,由于在调色板编码期间转义值也没有改变,因此在执行量化时可以使用用于变换跳过的最小量化参数信息。因此,用于基于调色板模式的编码的调色板编码(启用)标志(例如,sps\_palette\_enabled\_flag)和变换跳过期间的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus 4)可以具有依赖性。作为示例,如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为1,则应该有必要发送相关标志或信息语法元素,而如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为0,则可以不发送除了相应标志语法元素之外的语法元素。

[0215] 即,提出了用于根据高级语法(例如,VPS、SPS、PPS、APS、DPS和切片头)中的变换跳过(启用)标志的值和/或调色板编码(启用)标志的值来发送诸如关于变换跳过应用的最大尺寸的信息、变换跳过期间的最小量化参数信息以及是否应用BDPCM这样的依赖于是否执行变换跳过或者是否执行调色板编码的高级语法元素的方法。

[0216] 例如,(i)在变换跳过(启用)标志和调色板编码(启用)标志二者被定义为1的情况下,可以解析与依赖于变换跳过(启用)标志和调色板编码(启用)标志的语法元素的并集相对应的语法元素。(ii)在变换跳过(启用)标志被定义为1并且调色板编码(启用)标志为0的情况下,可以解析依赖于变换跳过(启用)标志的语法元素。(iii)在变换跳过(启用)标志被定义为0并且调色板编码(启用)标志为1的情况下,可以解析依赖于调色板编码(启用)标志的语法元素。(iv)在变换跳过(启用)标志和调色板编码(启用)标志二者为0的情况下,可以不解析依赖于两种编码工具的其它高级语法元素。

[0217] 本实施方式中提到的语法元素的解析顺序不受具体限制,并且在根据语法元素之间的依赖性确定是否执行解析的情况下,它们被认为彼此一致。另外,所提出的方法不限于本实施方式中提到的语法元素,而是可以具有依赖于是否执行变换跳过或调色板编码的依赖性,并且可以包括在包括变换跳过(启用)标志和调色板编码(启用)标志的高级语法集中定义的所有高级语法元素,

[0218] 如上所述,与基于变换跳过的编码和/或基于调色板模式的编码相关的语法元素可以在高级语法集中定义,并可以在序列参数集(SPS)中定义,如下表10的实施方式中一样。

[0219] [表10]

|   | 描述符   |
|---|-------|
| seq_parameter_set_rbsp() {  |       |
| (...)   |       |
| <b>sps_transform_skip_enabled_flag</b>                            | u(1)  |
| if( sps_transform_skip_enabled_flag )                             |       |
| <b>sps_bdpcm_enabled_flag</b>                                     | u(1)  |
| (...)   |       |
| [0220] if( chroma_format_idc == 3 )                               |       |
| <b>sps_palette_enabled_flag</b>                                   | u(1)  |
| (...)   |       |
| if( sps_transform_skip_enabled_flag    sps_palette_enabled_flag ) |       |
| <b>min_qp_prime_ts_minus4</b>                                     | ue(v) |
| (...)   |       |
| }   |       |

[0221] 另外,例如,SPS语法的语法元素当中的上述实施方式的语法元素的语义可以如下表11中表示的。

[0222] [表11]

|        |  |
|--------|--|
| [0223] | <p>等于 1 的 <b>sps_transform_skip_enabled_flag</b> 指明在变换单元语法中可以存在 transform_skip_flag。等于 0 的 sps_transform_skip_enabled_flag 指明在变换单元语法中不存在 transform_skip_flag。</p> <p>等于 1 的 <b>sps_bdpcm_enabled_flag</b> 指明在针对帧内编码单元的编码单元语法中可以存在 intra_bdpcm_flag。等于 0 的 sps_bdpcm_enabled_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra_bdpcm_flag。当不存在时,推断 sps_bdpcm_enabled_flag 的值等于 0。</p> <p><b>min_qp_prime_ts_minus4</b> 指明变换跳过模式下的最小允许量化参数如下:<br/> <math display="block">QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_minus4}</math></p> <p>等于 1 的 <b>sps_palette_enabled_flag</b> 指明在编码单元语法中可以存在</p> |
| [0224] | <p>pred_mode_plt_flag。等于 0 的 sps_palette_enabled_flag 指明在编码单元语法中不存在 pred_mode_plt_flag。当不存在 sps_palette_enabled_flag 时,推断其等于 0。</p>  |

[0225] 参照以上表10和表11,在SPS中,与变换跳过和/或调色板编码相关的语法元素可以被定义,并且可以包括语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag、sps\_palette\_enabled\_flag和min\_qp\_prime\_ts\_minus4。

[0226] 这里,由于语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag和min\_qp\_prime\_ts\_minus4已经在以上表6至表9中详细说明,因此在本实施方式中,为了便于说明,将省略对其的详细说明。

[0227] sps\_palette\_enabled\_flag语法元素可以基于其值为0还是1来表示是否启用调色板编码(即,调色板预测模式)。例如,如果sps\_palette\_enabled\_flag的值为1,则它可以表示启用了调色板编码,并且在这种情况下,可以通过编码单元语法解析/发信号通知pred\_mode\_plt\_flag。这里,pred\_mode\_plt\_flag语法元素可以表示是否可以针对当前编码单元使用调色板模式。如果sps\_palette\_enabled\_flag的值为0,则它可以表示未启用调色板编码,并且在这种情况下,可以不在编码单元语法中解析/发信号通知pred\_mode\_plt\_flag。换句话说,基于调色板编码启用标志sps\_palette\_enabled\_flag,可以表示编码单元语法中是否存在pred\_mode\_plt\_flag。

[0228] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)有依赖性的语法元素。例如,如以上表10和表11中公开的,在SPS中,基于变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值表示是否启用BDPCM的sps\_bdpcm\_enabled\_flag语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则可以解析/发信号通知sps\_bdpcm\_enabled\_flag语法元素。另外,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不解析/发信号通知sps\_bdpcm\_enabled\_flag语法元素。

[0229] 另外,在SPS中,可以定义涉及与变换跳过和/或调色板编码相关的语法元素当中的调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)的依赖性条件。例如,如以上表10和表11中公开的,在SPS中,可以基于chroma\_format\_idc语法元素来解析/发信号通知调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)。作为示例,如果chroma\_format\_idc语法元素的值为3,则可以解析/发信号通知sps\_palette\_enabled\_flag语法元素。

[0230] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)有依赖性的语法元素。例如,如在以上表10和表11中公开的,在SPS中,基于变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)的值,表示变换跳过模式下的最小量化参数信息的min\_qp\_prime\_ts\_minus4语法元素可以具有依赖性。作为示例,在变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用标志(例如,sps\_palette\_enabled\_flag)的值为1的情况下,可以解析/发信号通知min\_qp\_prime\_ts\_minus4语法元素。

[0231] 另外,如上所述,与基于变换跳过的编码和/或基于调色板模式的编码相关的语法元素可以在高级语法集中定义,并且如下表12的实施方式中一样,可以在序列参数集(SPS)中定义。然而,在相关技术的图片参数集(PPS)中定义的变换跳过的最大块尺寸信息可以在SPS中新定义,以避免HLS之间的依赖性,并且这可以如在下表12中表示的。

[0232] [表12]

|        |   |       |
|--------|---|-------|
|        | seq_parameter_set_rbsp() {  | 描述符   |
|        | (...)   |       |
|        | <b>sps_transform_skip_enabled_flag</b>                            | u(1)  |
|        | if( sps_transform_skip_enabled_flag ) {                           |       |
|        | <b>sps_bdpcm_enabled_flag</b>                                     | u(1)  |
|        | <b>log2_transform_skip_max_size_minus2</b>                        | ue(v) |
|        | }   |       |
| [0233] | (...)   |       |
|        | if( chroma_format_idc == 3 )                                      |       |
|        | <b>sps_palette_enabled_flag</b>                                   | u(1)  |
|        | (...)   |       |
|        | if( sps_transform_skip_enabled_flag    sps_palette_enabled_flag ) |       |
|        | <b>min_qp_prime_ts_minus4</b>                                     | ue(v) |
|        | (...)   |       |
|        | }   |       |

[0234] 另外,例如,SPS语法的语法元素当中的用于上述实施方式的语法元素的语义可以如在下表13中表示的。

[0235] [表13]

等于 1 的 **sps\_transform\_skip\_enabled\_flag** 指明在变换单元语法中可以存在 transform\_skip\_flag。等于 0 的 sps\_transform\_skip\_enabled\_flag 指明在变换单元语法中不存在 transform\_skip\_flag。

等于 1 的 **sps\_bdpcm\_enabled\_flag** 指明在针对帧内编码单元的编码单元语法中可以存在 intra\_bdpcm\_flag。等于 0 的 sps\_bdpcm\_enabled\_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra\_bdpcm\_flag。当不存在时，推断 sps\_bdpcm\_enabled\_flag 的值等于 0。

[0236]

**min\_qp\_prime\_ts\_minus4** 指明变换跳过模式下的最小允许量化参数如下：

$QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_minus4}$

**log2\_transform\_skip\_max\_size\_minus2** 指明用于变换跳过的最大块尺寸，并应当在 0 至 3 的范围内。

当不存在时，推断 log2\_transform\_skip\_max\_size\_minus2 的值等于 0。

变量 MaxTsSize 被设置为等于  $1 \ll (\text{log2\_transform\_skip\_max\_size\_minus2} + 2)$ 。

等于 1 的 **sps\_palette\_enabled\_flag** 指明在编码单元语法中可以存在 pred\_mode\_plt\_flag。等于 0 的 sps\_palette\_enabled\_flag 指明在编码单元语法中不存在 pred\_mode\_plt\_flag。当不存在 sps\_palette\_enabled\_flag 时，推断其等于 0。

[0237] 参照以上表12和表13,在SPS中,与变换跳过和/或调色板编码相关的语法元素可以被定义,并且可以包括语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag、log2\_transform\_skip\_max\_size\_minus2、sps\_palette\_enabled\_flag和min\_qp\_prime\_ts\_minus4。

[0238] 这里,由于语法元素sps\_transform\_skip\_enabled\_flag、sps\_bdpcm\_enabled\_flag、log2\_transform\_skip\_max\_size\_minus2、sps\_palette\_enabled\_flag和min\_qp\_prime\_ts\_minus4已经在以上表6至表11中详细描述,因此在本实施方式中,为了便于说明,将省略对其的详细说明。

[0239] 如在以上表12和表13的实施方式中公开的,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)有依赖性的语法元素。例如,如以上表12和表13中公开的,在SPS中,基于变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值表示是否启用BDPCM的sps\_bdpcm\_enabled\_flag语法元素和表示在变换跳过模式中使用的最大块尺寸的log2\_transform\_skip\_max\_size\_minus2语法元素可以具有依赖性。作为示例,如果变换跳过启

用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则可以解析/发信号通知语法元素sps\_bdpcm\_enabled\_flag和log2\_transform\_skip\_max\_size\_minus2。另外,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不解析/发信号通知语法元素sps\_bdpcm\_enabled\_flag和log2\_transform\_skip\_max\_size\_minus2。

[0240] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)有依赖性的语法元素。例如,如在以上表12和表13中公开的,基于变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)的值表示变换跳过模式的最小量化参数信息的min\_qp\_prime\_ts\_minus4语法元素在SPS中可以具有依赖性。作为示例,在变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用标志(例如,sps\_palette\_enabled\_flag)的值为1的情况下,可以解析/发信号通知min\_qp\_prime\_ts\_minus4语法元素。

[0241] 此外,源或编码图片/图像可以包括亮度分量数组,并且在一些情况下,还可以包括两个色度分量(cb,cr)数组,即,图片/图像的一个像素可以包括亮度样本和色度样本(cb,cr)。

[0242] 颜色格式可以表示亮度分量和色度分量(cb,cr)的配置格式,并可以被称为色度格式。颜色格式(或色度格式)可以是预定的,或者可以自适应地发信号通知。例如,可以基于chroma\_format\_idc和separate\_colour\_plane\_flag中的至少一者发信号通知色度格式,如在下表14中一样。

[0243] [表14]

| chroma_format_idc | separate_colour_plane_flag | 色度格式  | SubWidthC | SubHeightC |
|-------------------|----------------------------|-------|-----------|------------|
| 0                 | 0                          | 单色    | 1         | 1          |
| 1                 | 0                          | 4:2:0 | 2         | 2          |
| 2                 | 0                          | 4:2:2 | 2         | 1          |
| 3                 | 0                          | 4:4:4 | 1         | 1          |
| 3                 | 1                          | 4:4:4 | 1         | 1          |

[0245] 参照以上表14,在单色采样中,只有一个样本数组在名义上被视为亮度数组。

[0246] 在4:2:0采样中,两个色度数组中的每一个具有亮度数组的一半高度和一半宽度。

[0247] 在4:2:2采样中,两个色度数组中的每一个具有与亮度数组相同的高度和一半宽度。

[0248] 在4:4:4采样中,可以按照separate\_colour\_plane\_flag的值应用以下。

[0249] -如果separate\_colour\_plane\_flag的值为0,则两个色度数组中的每一个具有与亮度数组相同的高度和相同的宽度。

[0250] -否则,如果separate\_colour\_plane\_flag的值为1,则三种颜色平面可以被分别作为单色采样图片处理。

[0251] SubWidthC和SubHeightC可以表示亮度样本与色度样本之间的比率。例如,如果chroma\_format\_idc为3,则色度格式是4:4:4,并且在这种情况下,如果亮度样本块的宽度为16,则相应色度样本块的宽度可以为16/SubWidthC。通常,仅在色度数组类型(例如,

chromaArrayType)不为0的情况下才可以解析色度样本相关语法和比特流。

[0252] 另外,作为实施方式,本文献提出了用于通过使得对变换跳过相关高级语法元素和调色板编码相关高级语法元素有依赖性的高级元素能够根据依赖性条件确定是否执行解析/发信号通知来节省发送的比特的方法。作为示例,提出了用于通过变换跳过(启用)标志和调色板编码(启用)标志来解析具有依赖性的高级语法元素的方法。

[0253] 例如,作为依赖于基于变换跳过的编码的语法元素,存在变换跳过(启用)标志(例如,sps\_transform\_skip\_enabled\_flag)、关于变换跳过应用的尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)、变换跳过期间的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)以及关于是否应用BDPCM的信息(例如,sps\_bdpcm\_enabled\_flag)。另外,如上所述,由于在调色板编码期间转义值也没有改变,因此在执行量化时可以使用用于变换跳过的最小量化参数信息。

[0254] 如在上述实施方式中表示的,如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为1,则应该有必要发送相关标志或信息语法元素,而如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为0,则可以不发送除了相应标志语法元素之外的语法元素。即,提出了用于根据高级语法(例如,VPS、SPS、PPS、APS、DPS和切片头)中的变换跳过(启用)标志和调色板编码(启用)标志的值来发送诸如变换跳过或调色板编码期间的最小量化参数信息以及是否应用BDPCM这样的依赖于是否执行变换跳过或者是否执行调色板编码的高级语法元素的方法。

[0255] 例如,(i)在变换跳过(启用)标志和调色板编码(启用)标志二者被定义为1的情况下,可以解析与依赖于变换跳过(启用)标志和调色板编码(启用)标志的语法元素的并集相对应的语法元素。(ii)在变换跳过(启用)标志被定义为1并且调色板编码(启用)标志为0的情况下,可以解析依赖于变换跳过(启用)标志的语法元素。(iii)在变换跳过(启用)标志被定义为0并且调色板编码(启用)标志为1的情况下,可以解析依赖于调色板编码(启用)标志的语法元素。(iv)在变换跳过(启用)标志和调色板编码(启用)标志二者为0的情况下,可以不解析依赖于两种编码工具的其他高级语法元素。

[0256] 本实施方式中提到的语法元素的解析顺序不受具体限制,并且在根据语法元素之间的依赖性确定是否执行解析的情况下,它们被认为彼此一致。另外,所提出的方法不限于本实施方式中提到的语法元素,而是可以具有依赖于是否执行变换跳过或调色板编码的依赖性,并可以包括在包括变换跳过(启用)标志和调色板编码(启用)标志的高级语法集中定义的所有高级语法元素。

[0257] 如上所述,与基于变换跳过的编码和/或基于调色板模式的编码相关的语法元素可以在高级语法集中定义,并可以在序列参数集(SPS)中定义,如下表15的实施方式中一样。

[0258] [表15]

[0259]

| seq_parameter_set_rbsp() {  | 描述符   |
|---|-------|
| (...)   |       |
| <b>sps_transform_skip_enabled_flag</b>                            | u(1)  |
| if( sps_transform_skip_enabled_flag )                             |       |
| <b>sps_bdpcm_enabled_flag</b>                                     | u(1)  |
| (...)   |       |
| if( chroma_format_idc == 3 )                                      |       |
| <b>sps_palette_enabled_flag</b>                                   | u(1)  |
| (...)   |       |
| if( sps_palette_enabled_flag )                                    |       |
| <b>min_qp_prime_ts_chroma_minus4</b>                              | ue(v) |
| (...)   |       |
| if( sps_transform_skip_enabled_flag    sps_palette_enabled_flag ) |       |
| <b>min_qp_prime_ts_luma_minus4</b>                                | ue(v) |
| (...)   |       |
| }   |       |

[0260] 另外,例如,SPS语法的语法元素当中的上述实施方式的语法元素的语义可以如下表16中表示的。

[0261] [表16]

[0262]

等于 1 的 **sps\_transform\_skip\_enabled\_flag** 指明在变换单元语法中可以存在 transform\_skip\_flag。等于 0 的 sps\_transform\_skip\_enabled\_flag 指明在变换单元语法中不存在 transform\_skip\_flag。

等于 1 的 **sps\_bdpcm\_enabled\_flag** 指明在针对帧内编码单元的编码单元语法中可以存在 intra\_bdpcm\_flag。等于 0 的 sps\_bdpcm\_enabled\_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra\_bdpcm\_flag。当不存在时,推断 sps\_bdpcm\_enabled\_flag 的值等于 0。

**min\_qp\_prime\_ts\_minus4** 指明亮度分量中变换跳过模式的最小允许量化参数如下:

$$QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_luma\_minus4}$$

**min\_qp\_prime\_ts\_chroma\_minus4** 指明色度分量中变换跳过模式的最小允许量化参数如下:

$$QpPrimeTsMin=4+ \text{min\_qp\_prime\_ts\_chroma\_minus4}$$

[0263]

等于 1 的 `sps_palette_enabled_flag` 指明在编码单元语法中可以存在 `pred_mode_plt_flag`。等于 0 的 `sps_palette_enabled_flag` 指明在编码单元语法中不存在 `pred_mode_plt_flag`。当不存在 `sps_palette_enabled_flag` 时，推断其等于 0。

[0264] 参照以上表15和表16,在SPS中,与变换跳过和/或调色板编码相关的语法元素可以被定义,并且该语法元素可以包括语法元素`sps_transform_skip_enabled_flag`、`sps_bdpcm_enabled_flag`、`sps_palette_enabled_flag`、`min_qp_prime_ts_luma_minus4`和`min_qp_prime_ts_chroma_minus4`。

[0265] 这里,由于语法元素`sps_transform_skip_enabled_flag`、`sps_bdpcm_enabled_flag`和`sps_palette_enabled_flag`已经在以上表6至表11中详细说明,因此在本实施方式中,为了便于说明,将省略对其的详细说明。

[0266] 如在以上表15和表16的实施方式中公开的,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,`sps_transform_skip_enabled_flag`)有依赖性的语法元素。例如,如以上表15和表16中公开的,在SPS中,基于变换跳过启用标志(例如,`sps_transform_skip_enabled_flag`)的值表示是否启用BDPCM的`sps_bdpcm_enabled_flag`语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如,`sps_transform_skip_enabled_flag`)的值为1,则可以解析/发信号通知`sps_bdpcm_enabled_flag`语法元素。另外,如果变换跳过启用标志(例如,`sps_transform_skip_enabled_flag`)的值为0,则可以不解析/发信号通知`sps_bdpcm_enabled_flag`语法元素。

[0267] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的调色板编码启用标志语法元素(例如,`sps_palette_enabled_flag`)有依赖性的语法元素。例如,如在以上表15和表16中公开的,在SPS中,基于调色板编码启用标志(例如,`sps_palette_enabled_flag`)的值表示针对色度分量的变换跳过模式下的最小量化参数信息的`min_qp_prime_ts_chroma_minus4`语法元素可以具有依赖性。作为示例,如果调色板编码启用标志(例如,`sps_palette_enabled_flag`)的值为1,则可以解析/发信号通知`min_qp_prime_ts_chroma_minus4`语法元素。另外,如果调色板编码启用标志(例如,`sps_palette_enabled_flag`)的值为0,则可以不解析/发信号通知`min_qp_prime_ts_chroma_minus4`语法元素。

[0268] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,`sps_transform_skip_enabled_flag`)和/或调色板编码启用标志语法元素(例如,`sps_palette_enabled_flag`)有依赖性的语法元素。例如,如在以上表15和表16中公开的,在SPS中,基于变换跳过启用标志(例如,`sps_transform_skip_enabled_flag`)和/或调色板编码启用标志语法元素(例如,`sps_palette_enabled_flag`)的值表示针对亮度分量的变换跳过模式下的最小量化参数信息的`min_qp_prime_ts_luma_minus4`语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如,`sps_transform_skip_enabled_flag`)的值为1或者调色板编码启用标志(例如,`sps_palette_enabled_flag`)的值为1,可以解析/发信号通知`min_qp_prime_ts_luma_minus4`语法元素。

[0269] 另外,作为实施方式,本文献提出了通过使得对变换跳过相关高级语法元素和调

色板编码相关高级语法元素有依赖性的高级元素能够根据依赖性条件确定是否执行解析/发信号通知来节省发送的比特的方法。作为示例,提出了用于通过变换跳过(启用)标志和调色板编码(启用)标志来解析具有依赖性的高级语法元素的方法。

[0270] 例如,作为依赖于基于变换跳过的编码的语法元素,存在变换跳过(启用)标志(例如,sps\_transform\_skip\_enabled\_flag)、关于变换跳过应用的尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)、变换跳过期间的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)以及关于是否应用BDPCM的信息(例如,sps\_bdpcm\_enabled\_flag)。另外,如上所述,由于在调色板编码期间转义值也没有改变,因此在执行量化时可以使用用于变换跳过的最小量化参数信息。

[0271] 如在上述实施方式中表示的,如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为1,则应该有必要发送相关标志或信息语法元素,而如果变换跳过(启用)标志或调色板编码(启用)标志的值被定义为0,则可以不发送除了相应标志语法元素之外的语法元素。即,提出了用于根据高级语法(例如,VPS、SPS、PPS、APS、DPS和切片头)中的变换跳过(启用)标志和调色板编码(启用)标志的值来发送诸如关于变换跳过应用的最大尺寸的信息、变换跳过或调色板编码期间的最小量化参数信息以及是否应用BDPCM这样的依赖于是否执行变换跳过或者是否执行调色板编码的高级语法元素的方法。

[0272] 例如,(i)在变换跳过(启用)标志和调色板编码(启用)标志二者被定义为1的情况下,可以解析与依赖于变换跳过(启用)标志和调色板编码(启用)标志的语法元素的并集相对应的语法元素。(ii)在变换跳过(启用)标志被定义为1并且调色板编码(启用)标志为0的情况下,可以解析依赖于变换跳过(启用)标志的语法元素。(iii)在变换跳过(启用)标志被定义为0并且调色板编码(启用)标志为1的情况下,可以解析依赖于调色板编码(启用)标志的语法元素。(iv)在变换跳过(启用)标志和调色板编码(启用)标志二者为0的情况下,可以不解析依赖于两种编码工具的其他高级语法元素。

[0273] 本实施方式中提到的语法元素的解析顺序不受具体限制,并且在根据语法元素之间的依赖性确定是否执行解析的情况下,它们被认为彼此一致。另外,所提出的方法不限于本实施方式中提到的语法元素,而是可以具有依赖于是否执行变换跳过或调色板编码的依赖性,并可以包括在包括变换跳过(启用)标志和调色板编码(启用)标志的高级语法集中定义的所有高级语法元素,

[0274] 如上所述,与基于变换跳过的编码和/或基于调色板模式的编码相关的语法元素可以在高级语法集中定义,并可以在序列参数集(SPS)中定义,如下表17的实施方式中一样。在本实施方式中,可以在SPS中新定义关于在相关技术中的图片参数集(PPS)中定义的变换跳过的最大尺寸的信息,以避免HLS之间的依赖性,并且提出了基于先前使用的变换跳过和调色板编码相关语法元素的依赖性来执行解析/发信号通知的方法。

[0275] [表17]

[0276]

|   |       |
|---|-------|
| seq_parameter_sct_rbsp( ) {                                       | 描述符   |
| (...)   |       |
| <b>sps_transform_skip_enabled_flag</b>                            | u(1)  |
| if( sps_transform_skip_enabled_flag ) {                           |       |
| <b>sps_bdpcm_enabled_flag</b>                                     | u(1)  |
| <b>log2_transform_skip_max_size_minus2</b>                        | ue(v) |
| }   |       |
| (...)   |       |
| if( chroma_format_idc == 3 )                                      |       |
| <b>sps_palette_enabled_flag</b>                                   | u(1)  |
| (...)   |       |
| if( sps_palette_enabled_flag )                                    |       |
| <b>min_qp_prime_ts_chroma_minus4</b>                              | uc(v) |
| (...)   |       |
| if( sps_transform_skip_enabled_flag    sps_palette_enabled_flag ) |       |
| <b>min_qp_prime_ts_luma_minus4</b>                                | uc(v) |
| (...)   |       |
| }   |       |

[0277] 另外,例如,SPS语法的语法元素当中的上述实施方式的语法元素的语义可以如下表18中表示的。

[0278] [表18]

[0279]

|   |
|---|
| <p>等于 1 的 <b>sps_transform_skip_enabled_flag</b> 指明在变换单元语法中可以存在 transform_skip_flag。等于 0 的 sps_transform_skip_enabled_flag 指明在变换单元语法中不存在 transform_skip_flag。</p> <p>等于 1 的 <b>sps_bdpcm_enabled_flag</b> 指明在针对帧内编码单元的编码单元语法中可以存在 intra_bdpcm_flag。等于 0 的 sps_bdpcm_enabled_flag 指明在针对帧内编码单元的编码单元语法中不存在 intra_bdpcm_flag。当不存在时,推断 sps_bdpcm_enabled_flag 的值等于 0。</p> <p><b>log2_transform_skip_max_size_minus2</b> 指明用于变换跳过的最大块尺寸,并应当在 0 至 3 的范围内。</p> <p>当不存在时,推断 log2_transform_skip_max_size_minus2 的值等于 0。</p> <p>变量 MaxTsSize 被设置为等于 <math>1 \ll (\text{log2\_transform\_skip\_max\_size\_minus2} + 2)</math>。</p> |
|---|

[0280]

**min\_qp\_prime\_ts\_luma\_minus4** 指明亮度分量中的变换跳过模式下的最小允许量化参数如下:

$$\text{QpPrimeTsMin}=4+\text{min\_qp\_prime\_ts\_luma\_minus4}$$

**min\_qp\_prime\_ts\_chroma\_minus4** 指明色度分量中的变换跳过模式下的最小允许量化参数如下:

$$\text{QpPrimeTsMin}=4+\text{min\_qp\_prime\_ts\_chroma\_minus4}$$

等于 1 的 **sps\_palette\_enabled\_flag** 指明在编码单元语法中可以存在 **pred\_mode\_plt\_flag**。等于 0 的 **sps\_palette\_enabled\_flag** 指明在编码单元语法中不存在 **pred\_mode\_plt\_flag**。当不存在 **sps\_palette\_enabled\_flag** 时, 推断其等于 0。

[0281] 参照以上表17和表18,在SPS中,与变换跳过和/或调色板编码相关的语法元素可以被定义,并且可以包括语法元素 **sps\_transform\_skip\_enabled\_flag**、**sps\_bdpcm\_enabled\_flag**、**log2\_transform\_skip\_max\_size\_minus2**、**sps\_palette\_enabled\_flag**、**min\_qp\_prime\_ts\_luma\_minus4**和**min\_qp\_prime\_ts\_chroma\_minus4**。

[0282] 这里,由于语法元素 **sps\_transform\_skip\_enabled\_flag**、**sps\_bdpcm\_enabled\_flag**、**log2\_transform\_skip\_max\_size\_minus2**、**sps\_palette\_enabled\_flag**、**min\_qp\_prime\_ts\_luma\_minus4**和**min\_qp\_prime\_ts\_chroma\_minus4**已经在以上表6至表11中详细说明,因此在本实施方式中,为了便于说明,将省略对其的详细说明。

[0283] 如在以上表17和表18的实施方式中公开的,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如, **sps\_transform\_skip\_enabled\_flag**)有依赖性的语法元素。例如,如以上表17和表18中公开的,在SPS中,基于变换跳过启用标志(例如, **sps\_transform\_skip\_enabled\_flag**)的值表示是否启用BDPCM的 **sps\_bdpcm\_enabled\_flag**语法元素和表示在变换跳过模式下使用的最大块尺寸的 **log2\_transform\_skip\_max\_size\_minus2**语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如, **sps\_transform\_skip\_enabled\_flag**)的值为1,则可以解析/发信号通知语法元素 **sps\_bdpcm\_enabled\_flag**和**log2\_transform\_skip\_max\_size\_minus2**。另外,如果变换跳过启用标志(例如, **sps\_transform\_skip\_enabled\_flag**)的值为0,则可以不解析/发信号通知语法元素 **sps\_bdpcm\_enabled\_flag**和**log2\_transform\_skip\_max\_size\_minus2**。另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的调色板编码启用标志语法元素(例如, **sps\_palette\_enabled\_flag**)有依赖性的语法元素。例如,如在以上表17和表18中公开的,在SPS中,基于调色板编码启用标志(例如, **sps\_palette\_enabled\_flag**)的值表示针对色度分量的变换跳过模式下的最小量化参数信息的 **min\_qp\_prime\_ts\_chroma\_minus4**语法元素可以具有依赖性。作为示例,如果调色板编码启用标志(例如, **sps\_palette\_enabled\_flag**)的值为1,则可以解析/发信号通知 **min\_qp\_prime\_ts\_chroma**

minus4语法元素。另外,如果调色板编码启用标志(例如,sps\_palette\_enabled\_flag)的值为0,则可以不解析/发信号通知min\_qp\_prime\_ts\_chroma\_minus4语法元素。

[0284] 另外,在SPS中,可以定义对与变换跳过和/或调色板编码相关的语法元素当中的变换跳过启用标志语法元素(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)有依赖性的语法元素。例如,如在以上表17和表18中公开的,在SPS中,基于变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用标志语法元素(例如,sps\_palette\_enabled\_flag)的值表示针对亮度分量的变换跳过模式下的最小量化参数信息的min\_qp\_prime\_ts\_luma\_minus4语法元素可以具有依赖性。作为示例,如果变换跳过启用标志(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用标志(例如,sps\_palette\_enabled\_flag)的值为1,可以解析/发信号通知min\_qp\_prime\_ts\_luma\_minus4语法元素。

[0285] 已准备下图来说明本文献的详细示例。由于示例性地呈现了附图中描述的详细装置的名称或详细术语或名称(例如,语法元素的语法/名称),因此本文献的技术特征不限于附图中使用的详细名称。图10和图11示意性例示了根据本文献的实施方式的视频/图像编码方法和相关部件的示例。

[0286] 图11和图12示意性例示了根据本文献的实施方式的视频/图像编码方法和相关部件的示例。

[0287] 图11中公开的方法可以由图2或图12中公开的编码设备200执行。这里,图12中公开的编码设备200简要地表示图2中公开的编码设备200。具体地,图11中的步骤S1100至S1110可以由图2中公开的残差处理器230执行,并且图11中的步骤S1120可以由图2中公开的熵编码器240执行。另外,尽管未例示,但推导预测样本的处理可以由编码设备200的预测器220执行,并且基于当前块的残差样本和预测样本生成当前块的重构样本和重构图片的处理可以由编码设备200的加法器250执行,并且对当前块的预测信息进行编码的处理可以由编码设备200的熵编码器240执行。另外,图11中公开的方法可以包括将执行的上述实施方式。因此,参照图11,将省略或简化对上述实施方式的重复内容的详细说明。

[0288] 参照图11,编码设备可以推导当前块的残差样本(S1100)。

[0289] 作为实施方式,编码设备可以首先确定当前块的预测模式,并可以推导预测样本。例如,编码设备可以确定对当前块是执行帧间预测还是帧内预测,并还可以基于RD成本来确定具体帧间预测模式或具体帧内预测模式。另外,编码设备可以基于CIIP模式、IBC模式、BDPCM模式或调色板模式来确定是否对当前块执行预测。编码设备可以通过按照所确定的预测模式执行预测来推导当前块的预测样本。在这种情况下,可以应用诸如帧间预测或帧内预测这样的本文献中公开的各种预测方法。另外,编码设备可以生成并编码与应用于当前块的预测相关的信息(例如,预测模式信息)。

[0290] 此外,编码设备可以通过将预测样本与当前块的原始样本彼此比较来推导残差样本。编码设备可以通过对残差样本的变换处理来推导变换系数。在这种情况下,编码设备可以考虑编码效率来确定是否对当前块应用变换。即,编码设备可以确定是否对当前块的残差样本应用变换。

[0291] 例如,编码设备可以基于变换跳过启用信息来确定是对当前块(残差样本)应用变换还是变换跳过模式。

[0292] 如上所述,变换跳过启用信息可以是关于是否启用变换跳过的信息,并且如表6至表18中公开的,它可以被表示为sps\_transform\_skip\_enabled\_flag语法元素。

[0293] 例如,如果sps\_transform\_skip\_enabled\_flag的值为1,则它可以表示启用了变换跳过,并且在这种情况下,可以通过变换单元语法来解析/发信号通知transform\_skip\_flag。这里,transform\_skip\_flag语法元素可以表示是否可以对关联变换块应用变换。如果sps\_transform\_skip\_enabled\_flag的值为0,则它可以表示不启用变换跳过,并且在这种情况下,可以不在变换单元语法中解析/发信号通知transform\_skip\_flag。变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)可以被包括在SPS中,并可以被发信号通知给解码设备。即,基于SPS中所包括的变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,变换单元语法可以包括变换跳过标志(例如,transform\_skip\_flag)。在这种情况下,如果变换单元语法中所包括的变换跳过标志(例如,transform\_skip\_flag)的值为1,则可以对当前块执行不应用变换的模式(变换跳过模式)。另外,如果变换单元语法中所包括的变换跳过标志(例如,transform\_skip\_flag)的值为0,则可以对当前块应用变换。

[0294] 例如,如变换跳过启用信息的值为1(即,关于变换跳过启用信息表示变换跳过被启用),则编码设备可以确定是否对当前块应用变换。即,编码设备可以基于变换跳过启用信息的值为1来生成关于是否对当前块应用变换的信息(变换跳过标志),并可以通过变换单元语法来发信号通知变换跳过标志。在这种情况下,如果未对当前块应用变换(即,在变换跳过模式的情况下),编码设备可以生成值为1的变换跳过标志,并可以将其包括在变换单元语法中。另外,在对当前块应用变换的情况下,编码设备可以生成值为0的变换跳过标志,并可以将其包括在变换单元语法中。

[0295] 编码设备可以基于残差样本来生成残差信息(S1110)。

[0296] 作为实施方式,编码设备可以推导出当前块的残差样本,并可以通过基于是否应用变换对当前块的残差样本应用变换或变换跳过来生成残差信息。例如,对于变换跳过标志值为1的当前块的残差样本,编码设备可以应用变换跳过模式。在这种情况下,编码设备可以推导出当前块的残差样本作为变换系数。另外,对于变换跳过标志值为0的当前块的残差样本,编码设备可以通过执行变换来推导变换系数。编码设备可以通过基于通过变换跳过或变换推导出的变换系数执行量化处理来推导量化变换系数。编码设备可以基于量化变换系数来生成残差信息。

[0297] 这里,残差信息可以通过变换和/或量化处理生成的信息,并且可以是关于量化变换系数的信息,并且例如,可以包括关于量化变换系数的值信息、位置信息、变换技术、变换核和量化参数的信息。

[0298] 编码设备可以对图像信息(或视频信息)进行编码(S1120)。

[0299] 这里,图像信息可以包括残差信息。另外,图像信息可以包括与用于推导预测样本的预测相关的信息(例如,预测模式信息)。另外,图像信息可以包括与变换跳过相关的信息,例如,变换跳过启用信息、变换跳过标志信息以及关于在变换跳过模式下使用的最大块尺寸的信息。另外,图像信息可以包括与调色板编码相关的信息(例如,调色板启用信息)、BDPCM相关信息(例如,BDPCM启用信息以及BDPCM标志)等。另外,图像信息可以包括与变换跳过和/或调色板编码相关的信息,例如,关于变换跳过模式的最小允许量化参数的信息。

即,图像信息可以包括在编码处理中推导出的各种信息,并可以用这各种信息进行编码。

[0300] 另外,根据本文献中的上述实施方式图像信息可以包括各种信息,并可以包括在如上所述的表1至表18中的至少一个中公开的信息。

[0301] 例如,图像信息可以包括序列参数集(SPS)。SPS可以包括变换跳过相关信息和调色板编码相关信息。作为示例,变换跳过相关信息可以包括变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)、BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)、关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)以及与变换跳过模式的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。另外,作为示例,调色板编码相关信息可以包括调色板编码启用信息(例如,sps\_palette\_enabled\_flag)和与变换跳过模式的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。

[0302] 另外,例如,如上所述,在SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)有依赖性的信息。

[0303] 作为示例,SPS可以被配置为基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值来解析/发信号通知关于是否启用BDPCM的BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)可以被包括在SPS中,并且可以从SPS解析/发信号通知BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从SPS解析/发信号通知BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)。

[0304] 另外,基于SPS中的BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值,可以通过编码单元语法解析/发信号通知关于是否对当前块应用BDPCM的BDPCM标志信息(例如,intra\_bdpcm\_flag)。在这种情况下,如果BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值为1,则BDPCM标志信息(例如,intra\_bdpcm\_flag)可以被包括在编码单元语法中,并且可以从编码单元语法解析/发信号通知BDPCM标志信息(例如,intra\_bdpcm\_flag)。另外,如果BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值为0,则可以不从编码单元语法解析/发信号通知BDPCM标志信息(例如,intra\_bdpcm\_flag)。

[0305] 另外,作为示例,SPS可以被配置为基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值来解析/发信号通知关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)可以被包括在SPS中,并且可以从SPS解析/发信号通知该信息(例如,log2\_transform\_skip\_max\_size\_minus2)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从SPS解析/发信号通知关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)。

[0306] 另外,作为示例,基于在SPS中定义的变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值,可以通过变换单元语法解析/发信号通知关于是否对当前块应

用变换跳过的变换跳过标志信息(例如,transform\_skip\_flag)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则变换跳过标志信息(例如,transform\_skip\_flag)可以被包括在变换单元语法中,并且可以从变换单元语法解析/发信号通知变换跳过标志信息(例如,transform\_skip\_flag)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从变换单元语法解析/发信号通知变换跳过标志信息(例如,transform\_skip\_flag)。

[0307] 另外,例如,在如上所述的SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对关于是否启用调色板编码的调色板编码启用信息(例如,sps\_palette\_enabled\_flag)有依赖性的信息。作为示例,基于在SPS中定义的调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值,可以通过编码单元语法解析/发信号通知关于是否对当前块应用调色板编码(调色板预测模式)的调色板预测模式标志信息(例如,pred\_mode\_plt\_flag)。在这种情况下,如果调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为1,则调色板预测模式标志信息(例如,pred\_mode\_plt\_flag)可以被包括在编码单元语法中,并且可以从编码单元语法解析/发信号通知该信息(例如,pred\_mode\_plt\_flag)。另外,如果调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为0,则可以不从编码单元语法解析/发信号通知调色板预测模式标志信息(例如,pred\_mode\_plt\_flag)。

[0308] 另外,例如,如上所述,在SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用信息(例如,sps\_palette\_enabled\_flag)有依赖性的信息。

[0309] 作为示例,在SPS中,基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用信息(例如,sps\_palette\_enabled\_flag)中的至少一个,可以解析/发信号通知与变换跳过模式下的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。换句话说,基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为1的条件,最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)可以被包括在SPS中,并且只有在满足以上条件的情况下,才可以解析/发信号通知最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。

[0310] 这里,如上所述,最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)可以是与变换跳过模式下的最小允许量化参数相关的信息,并且基于此,可以推导出当前块的量化参数。

[0311] 例如,在对当前块应用变换跳过模式的情况下,可以基于最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)来推导当前块的量化参数,并且可以通过基于量化参数执行量化处理来推导量化变换系数。

[0312] 另外,例如,在对当前块应用调色板编码模式的情况下,可以基于最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)来推导当前块的转义值的量化参数。在这种情况下,可以通过基于量化参数将量化参数应用于当前块的转义值来推导量化转义值(例如,palette\_escape\_val)。应用调色板编码模式的处理可以如以上表4和表5中公开地执行。

[0313] 如上所述,包括各种信息的图像信息可以以比特流的形式编码和输出。可以通过

网络或(数字)存储介质将比特流发送到解码设备。这里,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和SSD这样的各种存储介质。

[0314] 图13和图14示意性例示了根据本文献的实施方式的视频/图像解码方法和相关部件的示例。

[0315] 图13中公开的方法可以由图3或图14中公开的解码设备300执行。这里,图14中公开的解码设备300简要地表示图3中公开的解码设备300。具体地,图13中的步骤S1300可以由图3中公开的熵解码器310执行,图13中的步骤S1310可以由图3中公开的残差处理器320执行,并且图13中的步骤S1320可以由图3中公开的加法器340执行。另外,尽管未例示,但接收当前块的预测信息的处理可以由解码设备300的熵解码器310执行,并且推导当前块的预测样本的处理可以由解码设备300的预测器330执行。另外,图13中公开的方法可以包括将执行的本文文献的上述实施方式。因此,参照图13,将省略或简化对上述实施方式的重复内容的详细说明。

[0316] 参照图13,解码设备可以从比特流接收图像信息(或视频信息)(S1300)。

[0317] 作为实施方式,解码设备可以通过解析比特流来推导图像重构(或图片重构)所必需的信息(例如,视频/图像信息)。在这种情况下,图像信息可以包括残差信息,并且残差信息可以包括量化变换系数的值信息、位置信息、变换技术、变换核和量化参数信息。另外,图像信息可以包括预测相关信息(例如,预测模式信息)。另外,图像信息可以包括与变换跳过相关的信息,例如,变换跳过启用信息、变换跳过标志信息以及关于在变换跳过模式下使用的最大块尺寸的信息。另外,图像信息可以包括与调色板编码相关的信息(例如,调色板启用信息)、BDPCM相关信息(例如,BDPCM启用信息以及BDPCM标志)等。另外,图像信息可以包括与变换跳过和/或调色板编码相关的信息,例如,关于变换跳过模式的最小允许量化参数的信息。即,图像信息可以包括解码处理中必需的各种信息,并可以基于诸如指数哥伦布编码、CAVLC或CABAC这样的编码方法来解码。

[0318] 另外,图像信息可以包括根据本文献的上述实施方式的各种信息,并可以包括在如上所述的表1至表18中的至少一个中公开的信息。

[0319] 例如,图像信息可以包括序列参数集(SPS)。SPS可以包括变换跳过相关信息和调色板编码相关信息。作为示例,变换跳过相关信息可以包括变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)、BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)、关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)以及与变换跳过模式的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。另外,作为示例,调色板编码相关信息可以包括调色板编码启用信息(例如,sps\_palette\_enabled\_flag)和与变换跳过模式的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。

[0320] 另外,例如,如上所述,在SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)有依赖性的信息。

[0321] 作为示例,SPS可以被配置为基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值来解析/发信号通知关于是否启用BDPCM的BDPCM启用信息(例如,sps\_

bdpcm\_enabled\_flag)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)可以被包括在SPS中,并且可以从SPS解析/发信号通知BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从SPS解析/发信号通知BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)。

[0322] 另外,基于SPS中的BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值,可以通过编码单元语法解析/发信号通知关于是否对当前块应用BDPCM的BDPCM标志信息(例如,intra\_bdpcm\_flag)。在这种情况下,如果BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值为1,则BDPCM标志信息(例如,intra\_bdpcm\_flag)可以被包括在编码单元语法中,并且可以从编码单元语法解析/发信号通知BDPCM标志信息(例如,intra\_bdpcm\_flag)。另外,如果BDPCM启用信息(例如,sps\_bdpcm\_enabled\_flag)的值为0,则可以不从编码单元语法解析/发信号通知BDPCM标志信息(例如,intra\_bdpcm\_flag)。

[0323] 另外,作为示例,SPS可以被配置为基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值来解析/发信号通知关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)可以被包括在SPS中,并且可以从SPS解析/发信号通知该信息(例如,log2\_transform\_skip\_max\_size\_minus2)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从SPS解析/发信号通知关于在变换跳过模式下使用的最大块尺寸的信息(例如,log2\_transform\_skip\_max\_size\_minus2)。

[0324] 另外,作为示例,基于在SPS中定义的变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值,可以通过变换单元语法解析/发信号通知关于是否对当前块应用变换跳过的变换跳过标志信息(例如,transform\_skip\_flag)。在这种情况下,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1,则变换跳过标志信息(例如,transform\_skip\_flag)可以被包括在变换单元语法中,并且可以从变换单元语法解析/发信号通知变换跳过标志信息(例如,transform\_skip\_flag)。另外,如果变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为0,则可以不从变换单元语法解析/发信号通知变换跳过标志信息(例如,transform\_skip\_flag)。

[0325] 另外,例如,在如上所述的SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对关于是否启用调色板编码的调色板编码启用信息(例如,sps\_palette\_enabled\_flag)有依赖性的信息。作为示例,基于在SPS中定义的调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值,可以通过编码单元语法来解析/发信号通知关于是否对当前块应用调色板编码(调色板预测模式)的调色板预测模式标志信息(例如,pred\_mode\_plt\_flag)。在这种情况下,如果调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为1,则调色板预测模式标志信息(例如,pred\_mode\_plt\_flag)可以被包括在编码单元语法中,并且可以从编码单元语法解析/发信号通知该信息(例如,pred\_mode\_plt\_flag)。另外,如果调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为0,则可以不从编码单元语法解析/发信号通知调色板预测模式标志信息(例如,pred\_mode\_plt\_

flag)。

[0326] 另外,例如,如上所述,在SPS中所包括的与变换跳过和/或调色板编码相关的信息当中,可以定义对变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用信息(例如,sps\_palette\_enabled\_flag)有依赖性的信息。

[0327] 作为示例,在SPS中,基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用信息(例如,sps\_palette\_enabled\_flag)中的至少一个,可以解析/发信号通知与变换跳过模式下的最小允许量化参数相关的最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。换句话说,基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为1的条件,最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)可以被包括在SPS中,并且只有在满足以上条件的情况下,才可以解析/发信号通知最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。

[0328] 这里,如上所述,最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)可以是与变换跳过模式下的最小允许量化参数相关的信息,并且基于此,可以推导出当前块的量化参数。

[0329] 例如,在对当前块应用变换跳过模式的情况下,可以基于最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)来推导当前块的量化参数,并且可以通过基于量化参数执行反量化处理(缩放处理)来推导反量化变换系数(缩放变换系数)。基于反量化变换系数,可以推导出当前块的残差样本。

[0330] 另外,例如,在对当前块应用调色板编码模式的情况下,可以基于最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)来推导当前块的转义值的量化参数。在这种情况下,可以通过基于量化参数执行反量化(缩放处理)来推导当前块的转义值。基于转义值,可以生成当前块的重构样本。应用调色板编码模式的处理可以如以上表4和表5中公开地执行。

[0331] 解码设备可以基于残差信息来推导当前块的残差样本(S1310)。

[0332] 作为实施方式,解码设备可以获得图像信息中所包括的残差信息。如上所述,残差信息可以包括量化变换系数的值信息、位置信息、变换技术、变换核和量化参数信息。解码设备可以基于残差信息中所包括的量化变换系数信息来推导当前块的量化变换系数,并可以基于量化变换系数来推导变换系数。另外,解码设备可以基于变换系数来推导残差样本。

[0333] 另外,在推导残差样本时,解码设备可以基于变换跳过启用信息来确定对当前块是应用变换还是变换跳过模式。

[0334] 如上所述,变换跳过启用信息可以是关于是否启用变换跳过的信息,并可以被表示为如以上表6至表18中公开的sps\_transform\_skip\_enabled\_flag语法元素。例如,如果sps\_transform\_skip\_enabled\_flag的值为1,则它可以表示启用了变换跳过,并且在这种情况下,可以通过变换单元语法来解析/发信号通知transform\_skip\_flag。这里,transform\_skip\_flag语法元素可以表示是否可以对关联变换块应用变换。如果sps\_transform\_skip\_enabled\_flag的值为0,则它可以表示不启用变换跳过,并且在这种情况下,可以不在变换单元语法中解析/发信号通知transform\_skip\_flag。变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)可以被包括在SPS中,并可以被从编码设备发信号通知给解码设备。即,基于SPS中所包括的变换跳过启用信息(例如,sps\_transform\_

skip\_enabled\_flag)的值为1,变换单元语法可以包括变换跳过标志(例如,transform\_skip\_flag)。在这种情况下,如果变换单元语法中所包括的变换跳过标志(例如,transform\_skip\_flag)的值为1,则可以对当前块执行不应用变换的模式(变换跳过模式)。另外,如果变换单元语法中所包括的变换跳过标志(例如,transform\_skip\_flag)的值为0,则可以对当前块应用变换。

[0335] 例如,如果变换跳过启用信息的值为1(即,针对变换跳过启用信息表示变换跳过被启用),则解码设备可以从变换单元语法获得是否对当前块应用变换的信息(变换跳过标志),并可以基于变换跳过标志的值来确定是否对当前块应用变换。例如,可以对变换跳过标志的值为1的当前块应用变换跳过模式,并且在这种情况下,解码设备可以推导变换系数作为当前块的残差样本。另外,可以对变换跳过标志的值为0的当前块应用变换,并且在这种情况下,解码设备可以通过对变换系数的逆变换来推导当前块的残差样本。

[0336] 另外,针对变换跳过标志的值为1的当前块(即,变换跳过模式),解码设备可以基于最小量化参数信息来推导在反量化处理中使用的量化参数。另外,解码设备可以通过基于量化参数执行反量化处理来推导反量化变换系数,并可以基于反量化变换系数来推导残差样本。

[0337] 这里,如上所述,最小量化参数信息可以是与变换跳过模式下的最小允许量化参数相关的信息,并可以基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)和/或调色板编码启用信息(例如,sps\_palette\_enabled\_flag)中的至少一者被包括在图像信息(例如,SPS)中。例如,基于变换跳过启用信息(例如,sps\_transform\_skip\_enabled\_flag)的值为1或者调色板编码启用信息(例如,sps\_palette\_enabled\_flag)的值为1的条件,最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)可以被包括在SPS中。即,只有在满足以上条件的情况下,才可以解析/发信号通知最小量化参数信息(例如,min\_qp\_prime\_ts\_minus4)。

[0338] 解码设备可以基于残差样本来生成重构样本(S1320)。

[0339] 作为实施方式,解码设备可以获得图像信息中所包括的预测信息(例如,预测模式信息)。解码设备可以基于预测信息(例如,预测模式信息)来确定是对当前块执行帧间预测还是帧内预测,并可以通过按照如上的确定执行预测来推导当前块的预测样本。

[0340] 另外,解码设备可以基于预测样本和残差样本来生成重构样本。在这种情况下,按照预测模式,解码设备可以直接使用预测样本作为重构样本,或者可以通过将残差样本与预测样本相加来生成重构样本。另外,解码设备可以基于重构样本来推导重构块或重构图片。此后,在需要时,解码设备可以对重构图片应用诸如去块滤波和/或SAO过程这样的环路滤波过程,以便如上所述改进主观/客观图片质量。

[0341] 尽管在上述实施方式中已经基于顺序列出步骤或块的流程图描述了方法,但本文的步骤不限于特定的顺序,并且特定步骤可以相对于上述步骤在不同的步骤中或以不同的顺序或者同时地执行。另外,本领域的普通技术人员将理解,流程图中的步骤不是排他性的,并且在没有对本公开的范围施加影响的情况下,可以在其中包括另一步骤,或者可以删除流程图中的一个或更多个步骤。

[0342] 根据本公开的以上提到的方法可以是软件的形式,并且根据本公开的编码设备和/或解码设备可以被包括在用于执行图像处理的装置(例如,TV、计算机、智能电话、机顶

盒、显示装置等)中。

[0343] 当用软件实现本公开的实施方式时,可以用执行以上提到的功能的模块(处理或功能)实现以上提到的方法。模块可以被存储在存储器中并且由处理器执行。存储器可以安装在处理器的内部或外部,并可以经由各种公知装置连接到处理器。处理器可以包括专用集成电路(ASIC)、其他芯片集、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)、闪速存储器、存储卡、存储介质和/或其他存储装置。换句话说,根据本公开的实施方式可以在处理器、微处理器、控制器或芯片上实现和执行。例如,相应图中例示的功能单元可以在计算机、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,关于实现方式的信息(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。

[0344] 另外,应用本文献的实施方式的解码设备和编码设备可以被包括在多媒体广播收发器、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、诸如视频通信这样的实时通信装置、移动流传输装置、存储介质、便携式摄像机、视频点播(VoD)服务提供器、顶置(OTT)视频装置、互联网流传输服务提供器、3D视频装置、虚拟现实(VR)装置、增强现实(AR)装置、图像电话视频装置、车载终端(例如,车(包括自主车辆)载终端、飞机终端或轮船终端)和医疗视频装置中;并且可以被用于处理图像信号或数据。例如,OTT视频装置可以包括游戏控制台、Blueray(蓝光)播放器、联网TV、家庭影院系统、智能手机、平板PC和数字视频记录仪(DVR)。

[0345] 另外,应用本文献的实施方式的处理方法可以由计算机执行的程序的形式产生,并可以被存储在计算机可读记录介质中。根据本文献的实施方式的具有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有计算机可读数据的所有种类的存储装置和分布式存储装置。计算机可读记录介质可以包括例如蓝光光盘(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。计算机可读记录介质还包括以载波(例如,互联网上的传输)的形式实施的媒体。另外,通过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,或者可以通过有线或无线通信网络传输。

[0346] 另外,本文献的实施方式可以基于程序代码被实施为计算机程序产品,并且程序代码可以根据本文献的实施方式在计算机上执行。程序代码可以被存储在计算机可读载体上。

[0347] 图15例示了适用本文献中公开的实施方式的内容流传输系统的示例。

[0348] 参照图15,应用于本文献的实施方式的内容流传输系统可以简单地包括编码服务器、流传输服务器、网络服务器、媒体存储器、用户装置和多媒体输入装置。

[0349] 编码服务器用来将从诸如智能电话、照相机、便携式摄像机等这样的多媒体输入装置输入的内容压缩为数字数据,生成比特流,并且将其传送至流传输服务器。作为另一示例,在诸如智能电话、照相机、便携式摄像机等这样的多媒体输入装置直接生成码流的情况下,可省略编码服务器。

[0350] 可通过本文献的实施方式应用于的编码方法或比特流生成方法来生成比特流。并且流传输服务器可在发送或接收比特流的处理中暂时存储比特流。

[0351] 流传输服务器基于用户的请求通过网络服务器向用户设备传送多媒体数据,该网

络服务器充当向用户通知存在什么服务的工具。当用户请求用户想要的服务时,网络服务器将请求转移至流传输服务器,并且流传输服务器将多媒体数据传送至用户。在这方面,内容流系统可包括单独的控制服务器,并且在这种情况下,控制服务器用来控制内容流系统中的各个设备之间的命令/响应。

[0352] 流传输服务器可从媒体存储装置和/或编码服务器接收内容。例如,在从编码服务器接收到内容的情况下,可实时地接收内容。在这种情况下,流传输服务器可将比特流存储预定时间段以流畅地提供流传输服务。

[0353] 例如,用户设备可包括移动电话、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、板式PC、平板PC、超级本、可穿戴装置(例如,手表型终端(智能手表)、眼镜型终端(智能眼镜)、头戴式显示器(HMD))、数字TV、台式计算机、数字标牌等。

[0354] 可将内容流系统中的每个服务器作为分布式服务器操作,并且在这种情况下,可以分布式方式处理由每个服务器接收的数据。

[0355] 本说明书中的权利要求可以以各种方式组合。例如,本说明书的方法权利要求中的技术特征可以组合以在设备中实施或执行,并且设备权利要求中的技术特征可以组合以在方法中实施或执行。此外,方法权利要求和设备权利要求中的技术特征可以组合以在设备中实施或执行。此外,方法权利要求和设备权利要求中的技术特征可以组合以在方法中实施或执行。

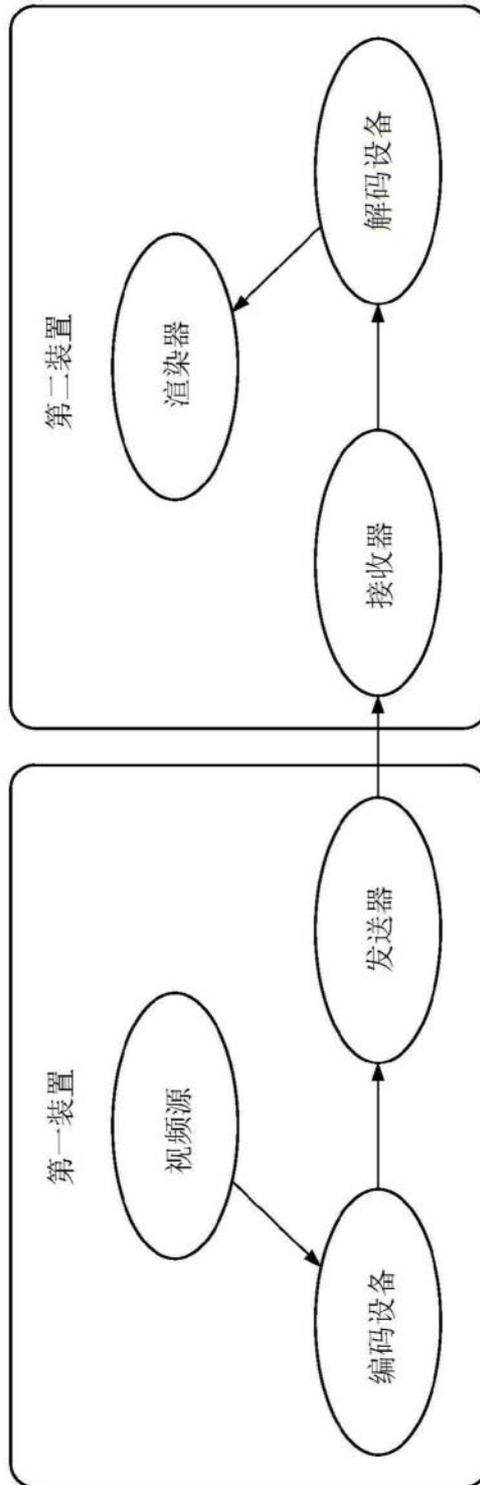


图1

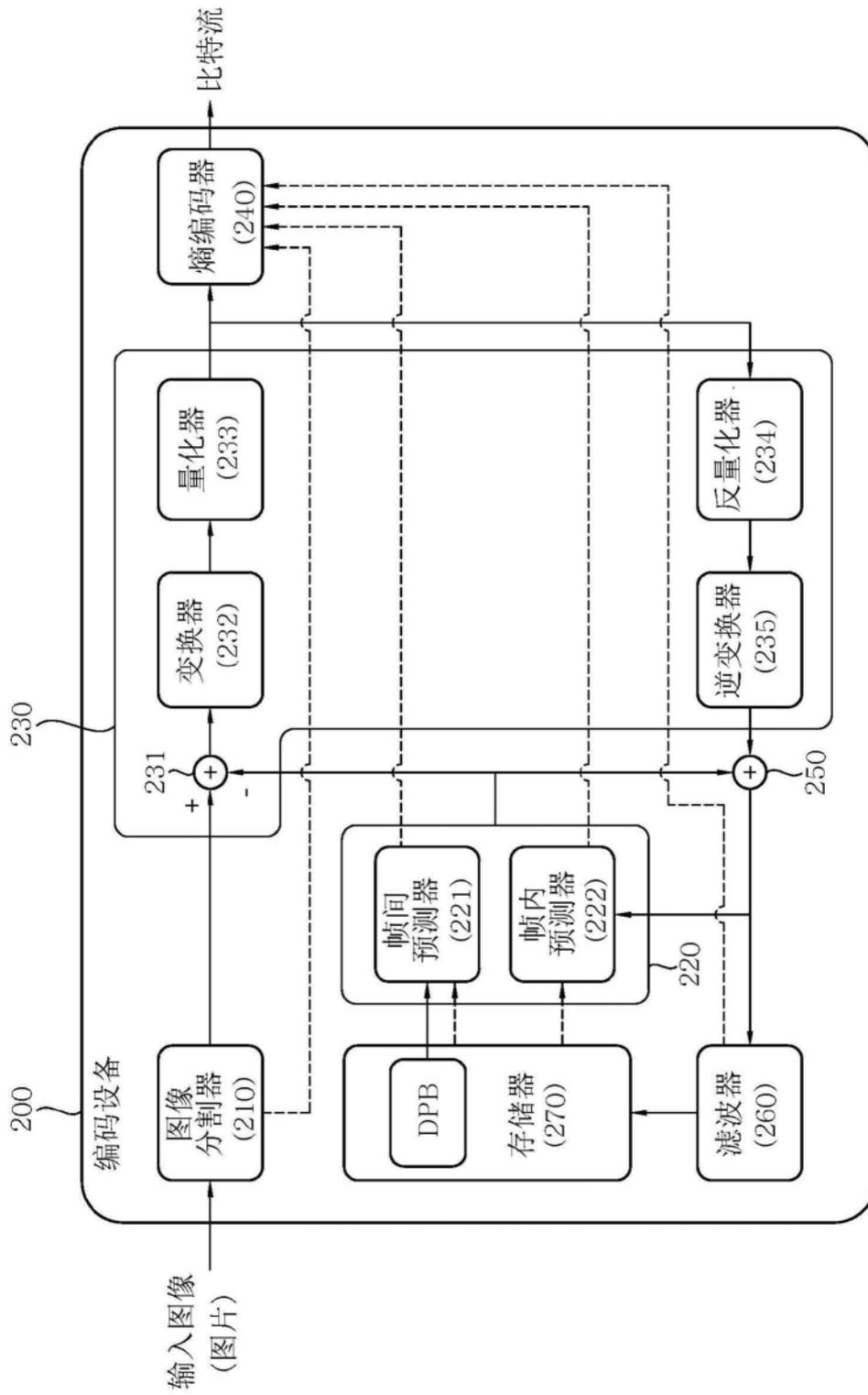


图2

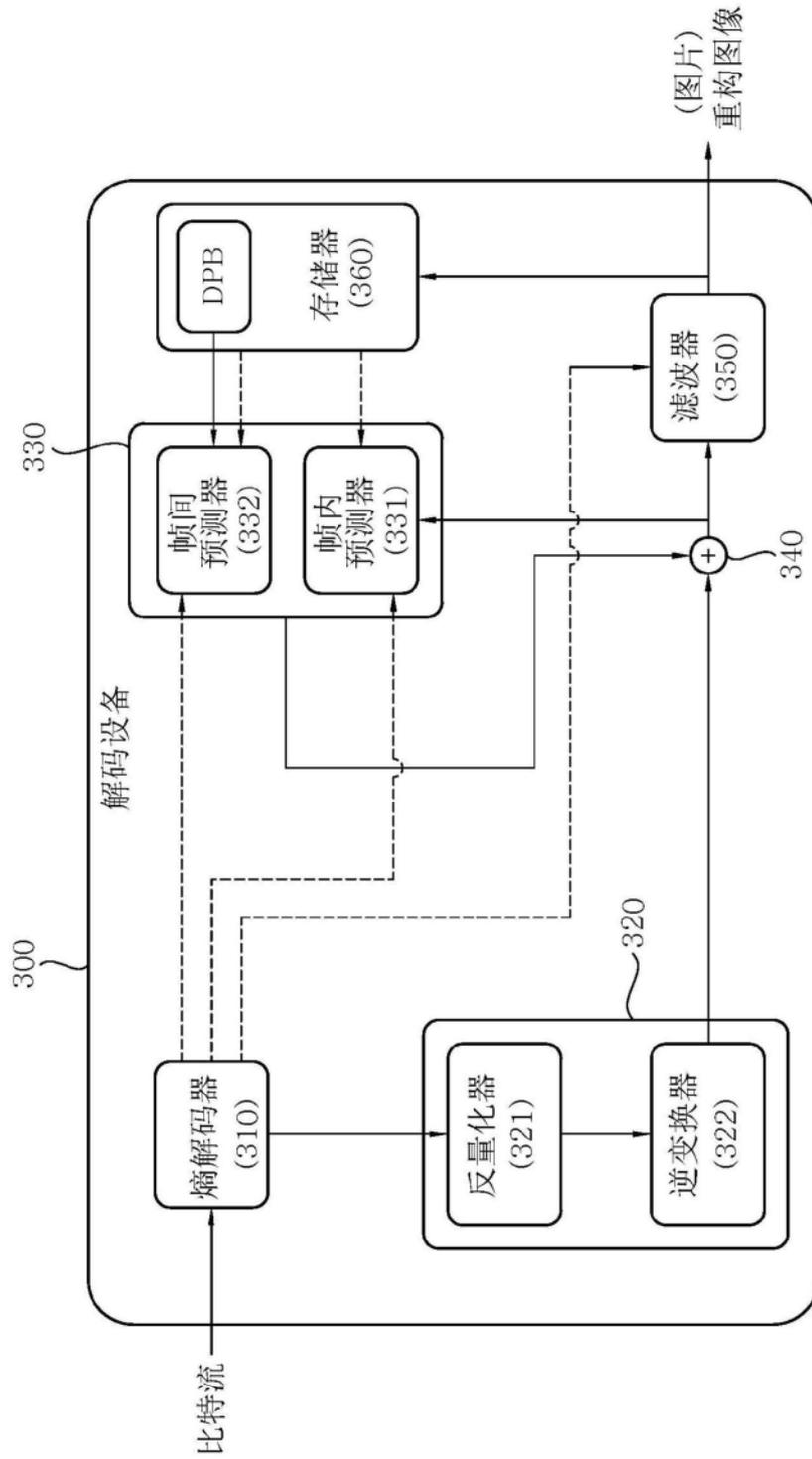


图3

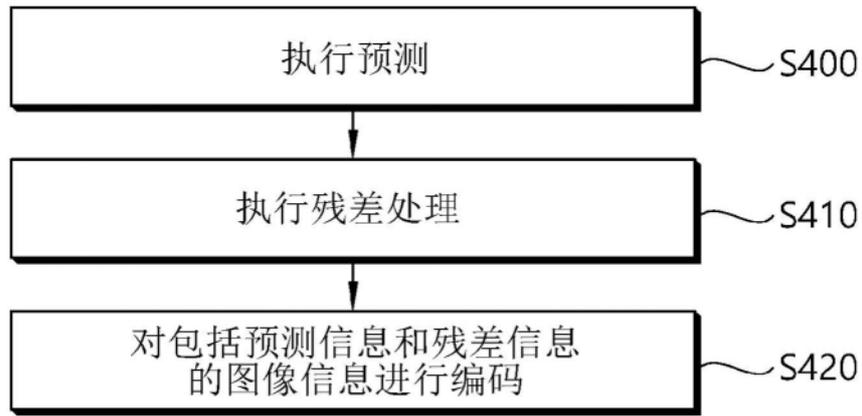


图4

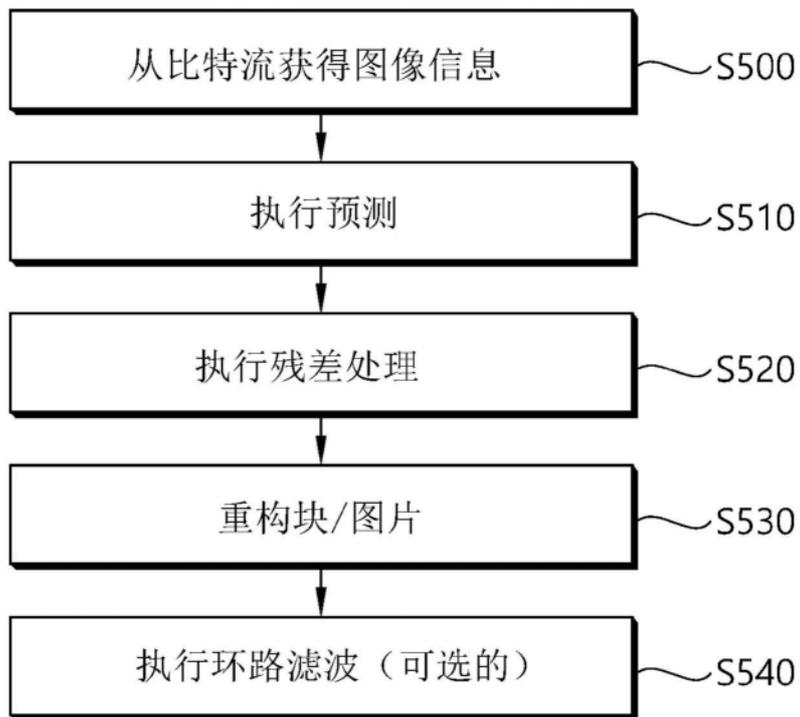


图5

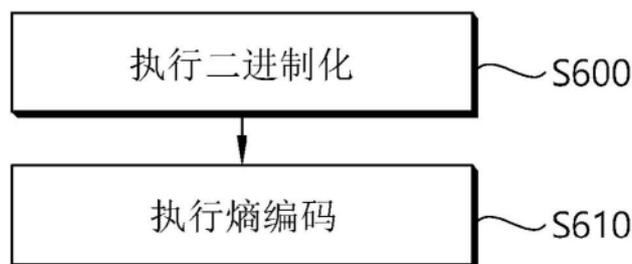


图6

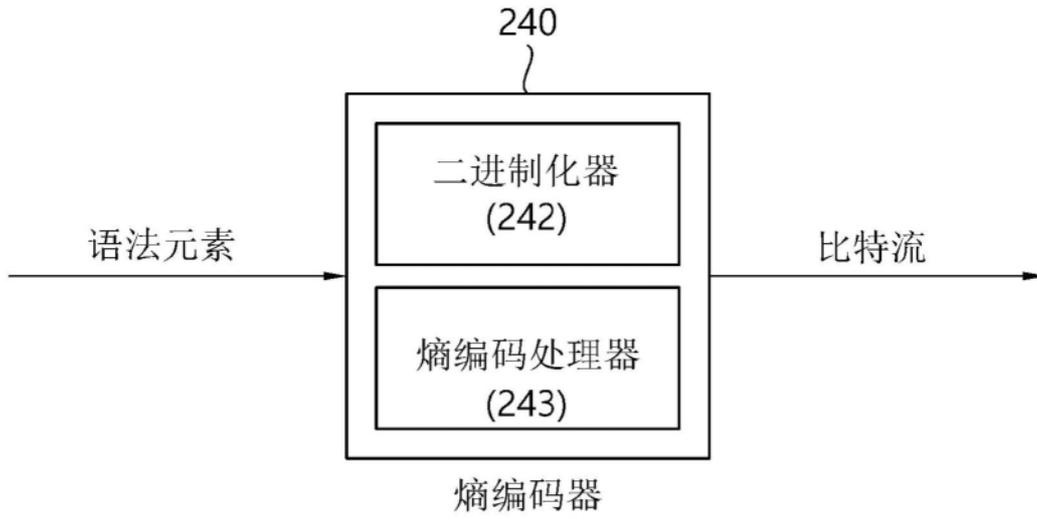


图7

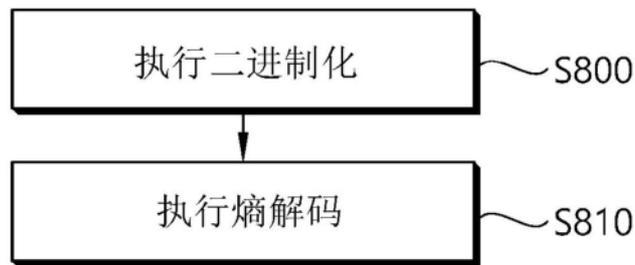


图8

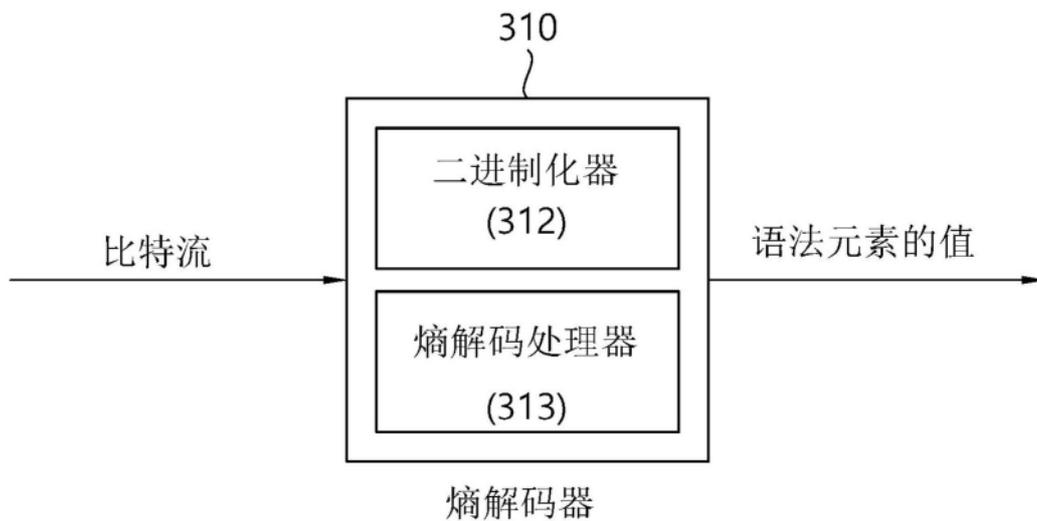


图9

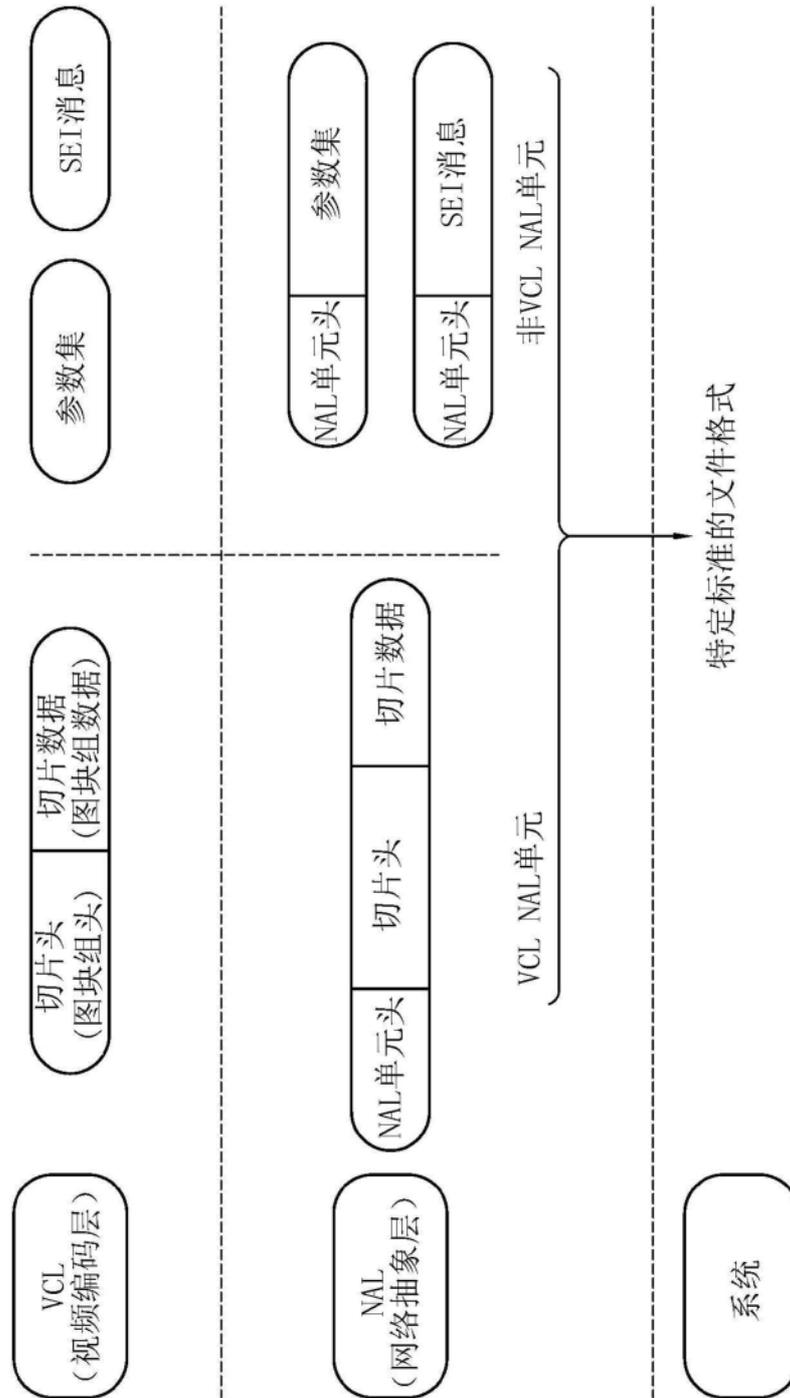


图10

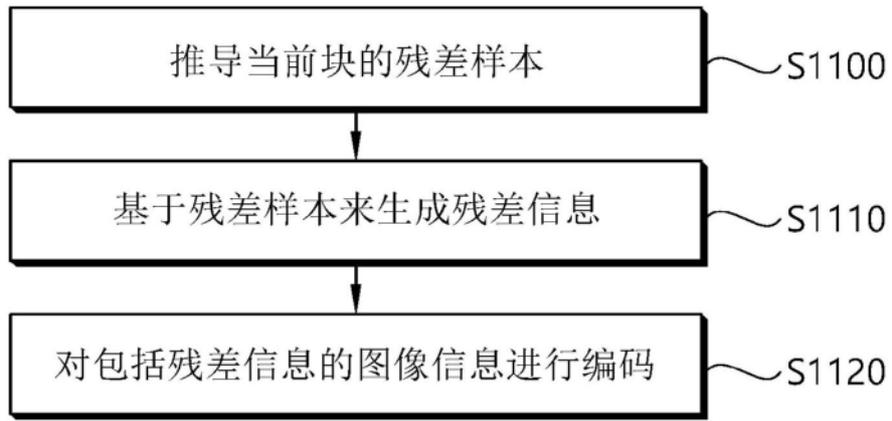


图11

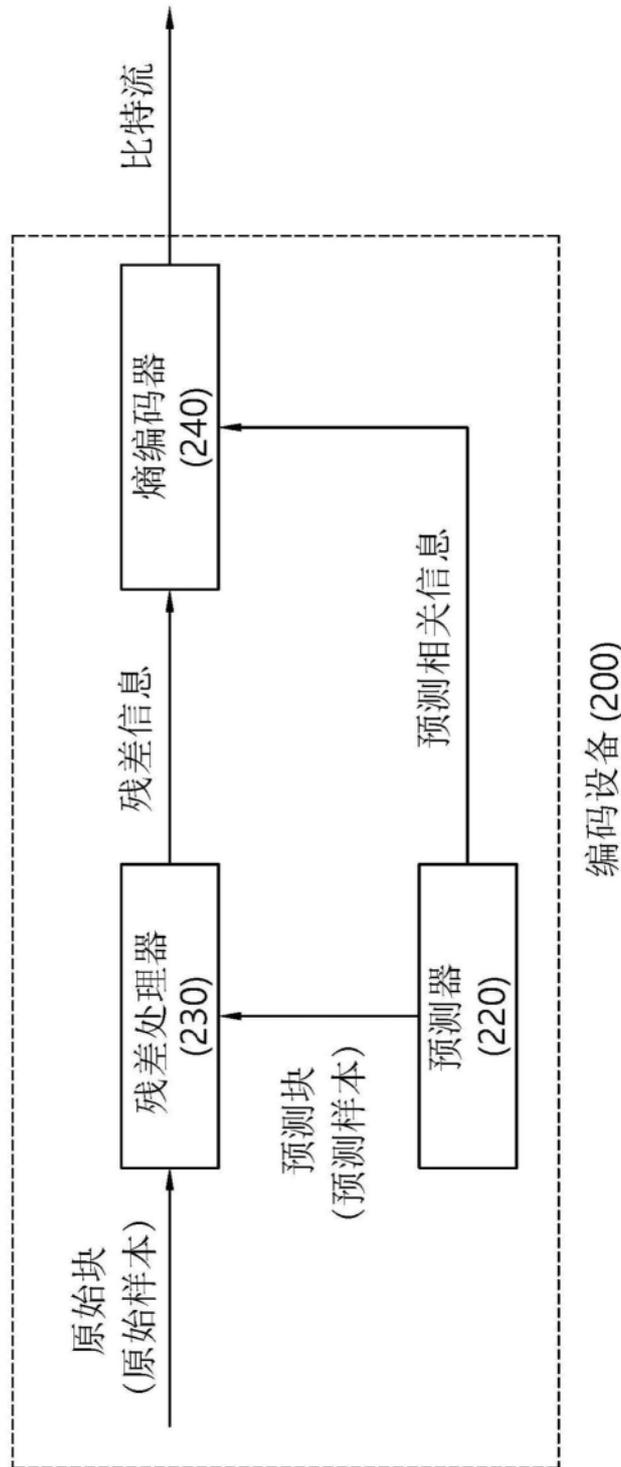


图12

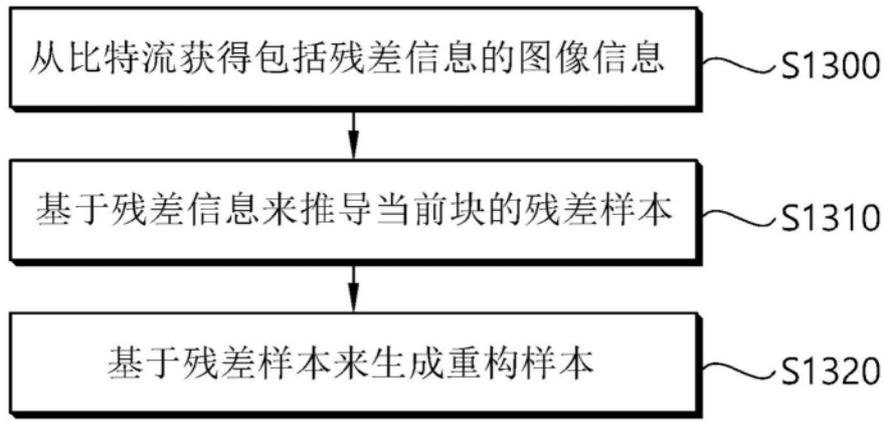


图13

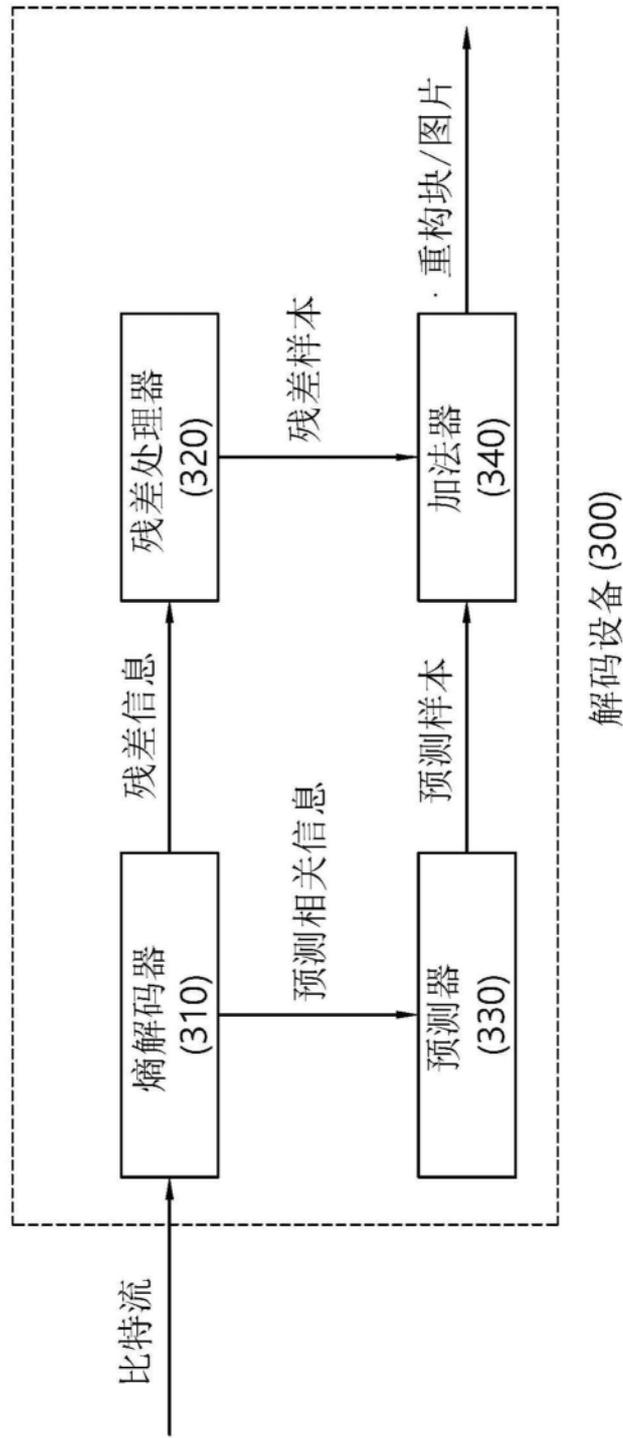


图14

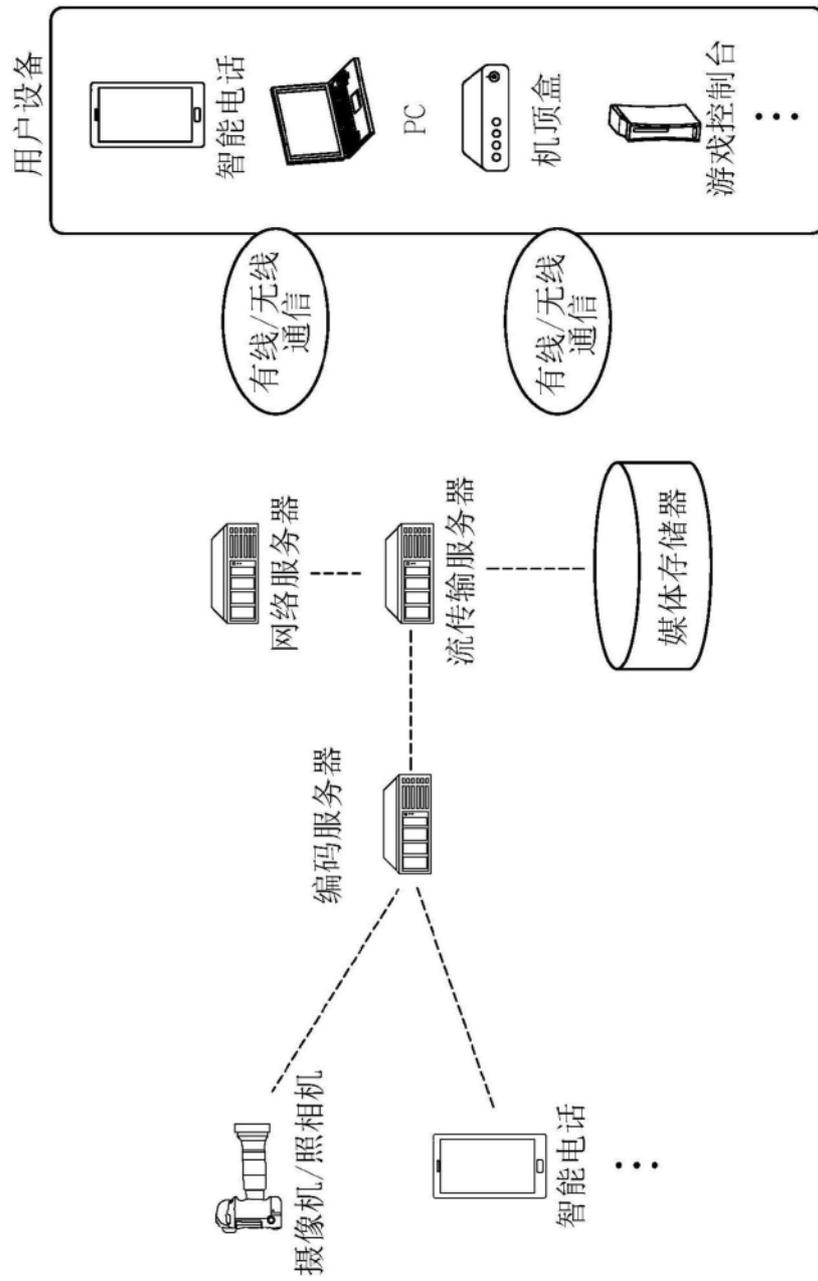


图15