(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 104041052 B
(45)授权公告日 2017.11.14

(21)申请号 201280065586.9

(22)申请日 2012.10.31

(65)同一申请的已公布的文献号
申请公布号 CN 104041052 A

(43)申请公布日 2014.09.10

(30)优先权数据
61/553,668 2011.10.31 US
61/671,955 2012.07.16 US

(85)PCT国际申请进入国家阶段日
2014.06.30

(86)PCT国际申请的申请数据
PCT/KR2012/009074 2012.10.31

(87)PCT国际申请的公布数据
WO2013/066051 KO 2013.05.10

(73)专利权人 三星电子株式会社
地址 韩国京畿道水原市

(72)发明人 朴银姬 闵正惠

(74)专利代理机构 北京铭硕知识产权代理有限公司 11286
代理人 苏银虹 王艳娇

(51)Int.Cl.
H04N 19/91(2014.01)
H04N 19/13(2014.01)
H04N 19/70(2014.01)
H04N 19/103(2014.01)
H04N 19/122(2014.01)
H04N 19/18(2014.01)

(56)对比文件
US 2004/0184549 A1,2004.09.23,
CN 1812756 A,2006.08.02,
CN 101304526 A,2008.11.12,
US 2011/0116549 A1,2011.05.19,
Joel Sole,et al.CE11:Unified scans for the significance map and coefficient level coding in high efficiency.《Joint Collaborative Team on Video Coding(JCT-VC)of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11》.2011,第1-9页.

审查员 韩盼

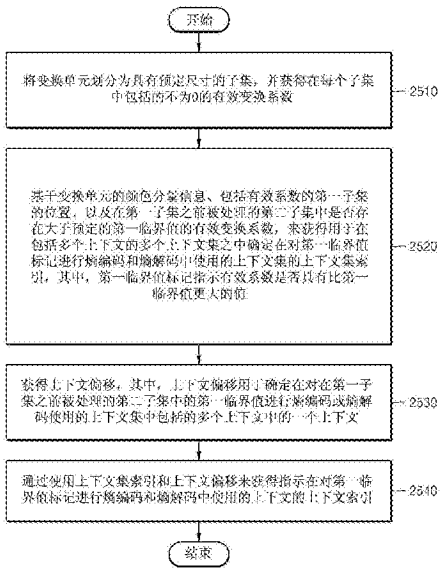
权利要求书1页 说明书28页 附图20页

(54)发明名称

确定用于变换系数级别熵编码和熵解码的上下文模型的方法和设备

(57)摘要

提供一种用于确定对变换系数进行熵编码和熵解码的上下文模型的方法和设备。根据所述方法和设备，基于变换单元的颜色分量信息、当前子集的位置、以及在先前子集中是否存在具有比第一临界值更大的值的有效变换系数来获得上下文集索引ctxset，并基于具有连续的1的先前变换系数的长度来获得上下文偏移c1。此外，基于获得的上下文集索引和上下文偏移来确定用于对第一临界值标记进行熵编码和熵解码的上下文索引ctxidx。



CN 104041052 B

1.一种确定用于对变换系数级别进行熵解码的上下文模型的方法,所述方法包括:

将变换单元划分为具有预定尺寸的子集;

获得在对第一临界值标记进行熵解码中使用的上下文集索引,其中,第一临界值标记指示在当前子集中包含的有效变换系数是否具有比第一临界值更大的值;

基于具有连续的1的先前变换系数的数量来获得在对第一临界值标记进行熵解码中使用的上下文偏移;

通过使用上下文集索引和上下文偏移来获得指示在对第一临界值标记进行熵解码中使用的上下文的上下文索引;

基于在对第一临界值标记进行熵解码中使用的上下文集索引来获得在对第二临界值标记进行熵解码中使用的上下文集索引,其中,第二临界值标记指示有效变换系数是否具有比第二临界值更大的值,

其中,当变换单元的颜色分量是亮度分量时,上下文集索引基于当前子集的位置信息被获得,其中,所述位置信息指示当前子集位于变换单元中的哪个位置,

其中,当变换单元的颜色分量是色度分量时,上下文集索引独立于当前子集的位置信息被获得。

2.如权利要求1所述的方法,其中,第一临界值是1。

确定用于变换系数级别熵编码和熵解码的上下文模型的方法和设备

技术领域

[0001] 本发明涉及对视频进行编码和解码，更具体地讲，涉及一种用于选择在对变换系数的大小信息进行熵编码和熵解码中使用的上下文模型的方法和设备。

背景技术

[0002] 根据图像压缩方法(诸如MPEG-1、MPEG-2或MPEG-4H.264/MPEG-4先进视频编码(AVC))，图像被划分为具有预定尺寸的块，然后，通过帧间预测或帧内预测获得块的残差数据。通过变换、量化、扫描、游程编码和熵编码对残差数据进行压缩。在熵编码中，对句法元素(诸如变换系数或运动矢量)进行熵编码以输出比特流。在解码器端，从比特流提取句法元素，并基于提取的句法元素执行解码。

发明内容

[0003] 技术问题

[0004] 将由本发明解决的技术问题是去除在对变换系数级别进行熵编码和熵解码中使用的不必要的上下文模型，并进行简化以减少用于存储上下文模型的存储器耗费。

[0005] 本发明还提供增加选择上下文模型的操作的速度，并在不明显降低熵编码和熵解码性能的情况下简化所述操作。

[0006] 解决方案

[0007] 本发明提供一种用于选择在对变换系数级别进行熵编码和熵解码中使用的上下文模型的方法和设备，其中，基于颜色信息、包括变换系数的子集的位置、以及1的连续长度的信息等来选择在对变换系数级别进行熵编码和熵解码中使用的上下文模型。

[0008] 有益效果

[0009] 根据本发明的实施例，通过使用数量减少的上下文模型，可减少用于存储上下文模型的存储器的使用量，并且可在性能无明显降低的情况下执行对变换系数级别的熵编码和熵解码。

附图说明

[0010] 图1是根据本发明的实施例的用于对视频进行编码的设备的框图；

[0011] 图2是根据本发明的实施例的用于对视频进行解码的设备的框图；

[0012] 图3是用于描述根据本发明的实施例的编码单元的概念的示图；

[0013] 图4是根据本发明的实施例的基于具有分层结构的编码单元的视频编码器的框图；

[0014] 图5是根据本发明的实施例的基于具有分层结构的编码单元的视频解码器的框图；

[0015] 图6是示出根据本发明的实施例的根据深度的较深层编码单元和分区的示图；

[0016] 图7是用于描述根据本发明的实施例的编码单元与变换单元之间的关系的示图；

[0017] 图8是用于描述根据本发明的实施例的与编码深度相应的编码单元的编码信息的示图；

[0018] 图9是根据本发明的实施例的根据深度的较深层编码单元的示图；

[0019] 图10至图12是用于描述根据本发明的实施例的编码单元、预测单元和频率变换单元之间的关系的示图；

[0020] 图13是用于描述根据表1的编码模式信息的编码单元、预测单元和变换单元之间的关系的示图；

[0021] 图14是示出根据本发明的实施例的对在变换单元中包括的变换系数信息进行熵编码和熵解码的操作的流程图；

[0022] 图15示出根据本发明的实施例的通过划分变换单元而获得的子集；

[0023] 图16示出根据本发明的实施例的在图15的变换单元中包括的子集；

[0024] 图17示出与图16的子集相应的有效图(significant map)；

[0025] 图18示出与图16的子集相应的第一临界值标记；

[0026] 图19示出与图16的子集相应的第二临界值标记；

[0027] 图20是示出在图16至图19中示出的子集中包括的变换系数以及被熵编码和熵解码的变换系数信息的表；

[0028] 图21a是示出根据本发明的实施例的熵编码设备的结构框图；

[0029] 图21b是示出根据本发明的实施例的熵解码设备的结构框图；

[0030] 图22是示出根据本发明的实施例的上下文建模器的结构框图；

[0031] 图23示出根据本发明的实施例的被应用于亮度分量的变换单元的多个上下文集以及在每个上下文集中包括的多个上下文；

[0032] 图24示出根据本发明的实施例的被应用于色度分量的变换单元的多个上下文集以及在每个上下文集中包括的多个上下文；

[0033] 图25是示出根据本发明的实施例的确定用于对变换系数级别进行熵编码和熵解码的上下文模型的方法的流程图；

[0034] 图26是示出根据本发明的实施例的确定用于对变换系数级别进行熵编码和熵解码的上下文模型的方法的详细流程图；

[0035] 图27a示出根据本发明的实施例的用于确定上下文集的上下文集索引ctxset，其中，在对亮度分量的有效变换系数和色度分量的有效变换系数的第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码中使用所述上下文集；

[0036] 图27b示出根据本发明的实施例的在对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文偏移；

[0037] 图28示出根据本发明的实施例的显示上下文偏移索引c1以及图20的被熵编码或熵解码的变换系数信息的表，其中，在对子集中包括的变换系数进行熵编码或熵解码中使用所述上下文偏移索引c1；

[0038] 图29示出根据本发明的另一实施例的显示上下文偏移索引c1以及被熵编码或熵解码的变换系数信息的表，其中，在对子集中包括的变换系数进行熵编码和熵解码中使用所述上下文偏移索引c1。

[0039] 最佳实施方式

[0040] 根据本发明的一方面，提供一种确定用于对变换系数级别进行熵编码和熵解码的上下文模型的方法，所述方法包括：将变换单元划分为具有预定尺寸的子集，并获得在每个子集中包括的不为0的有效变换系数；基于变换单元的颜色分量信息、包括有效变换系数的第一子集的位置信息、以及在第一子集之前被处理的第二子集中是否存在具有比第一临界值更大的值的有效变换系数，来获得用于在包括多个上下文的多个上下文集之中确定在对第一临界值标记进行熵编码和熵解码中使用的上下文集的上下文集索引，其中，第一临界值标记指示有效变换系数是否具有比第一临界值更大的值；基于具有连续的1的先前变换系数的长度来获得用于确定多个上下文中的一个上下文的上下文偏移，其中，所述多个上下文包括在对第一临界值标记进行熵编码和熵解码中使用的上下文集中；通过使用上下文集索引和上下文偏移来获得指示在对第一临界值标记进行熵编码和熵解码中使用的上下文的上下文索引。

[0041] 根据本发明的另一方面，提供一种用于确定对变换系数级别进行熵编码和熵解码的上下文模型的设备，包括：映射单元，将变换单元划分为具有预定尺寸的子集，并获得在每个子集中包括的不为0的有效变换系数；上下文集获得单元，基于变换单元的颜色分量信息、包括有效变换系数的第一子集的位置信息、以及在第一子集之前被处理的第二子集中是否存在具有比第一临界值更大的值的有效变换系数，来获得用于在包括多个上下文的多个上下文集之中确定在对第一临界值标记进行熵编码和熵解码中使用的上下文集的上下文集索引，其中，第一临界值标记指示有效变换系数是否具有比第一临界值更大的值；上下文偏移获得单元，基于具有连续1的先前变换系数的长度来获得用于确定多个上下文中的一个上下文的上下文偏移，其中，所述多个上下文包括在对第一临界值标记进行熵编码和熵解码中使用的上下文集中；上下文确定单元，通过使用上下文集索引和上下文偏移来获得指示在对第一临界值标记进行熵编码和熵解码中使用的上下文的上下文索引。

具体实施方式

[0042] 以下，在本申请的各种实施例中描述的“图像”可以是不仅指静止图像还指视频图像的包容性的概念。

[0043] 当对与图像相关的数据执行各种操作时，与图像相关的数据被划分为数据组，并且可对在同一数据组中包括的数据执行相同操作。在此说明书中，根据预定标准形成的数据组被称为“数据单元”。以下，对每个“数据单元”执行的操作被理解为使用在数据单元中包括的数据来执行。

[0044] 以下，将参照图1至图13描述根据本发明的实施例的用于对视频进行编码和解码的方法和设备，其中，基于具有分层树结构的编码单元对具有树结构的句法元素进行编码或解码。另外，将参照图14至图29详细地描述在参照图1至图13描述的对视频进行编码和解码中使用的熵编码和熵解码的方法。

[0045] 图1是根据本发明的实施例的视频编码设备100的框图。

[0046] 视频编码设备100包括分层编码器110和熵编码器120。

[0047] 分层编码器110可以以预定数据单元为单位划分将编码的当前画面，以对每个数据单元执行编码。详细地讲，分层编码器110可基于最大编码单元划分当前画面，其中，最大

编码单元是最大尺寸的编码单元。根据本发明的实施例的最大编码单元可以是尺寸为32×32、64×64、128×128、256×256等的数据单元，其中，数据单元的形状是具有2的若干次方且大于8的宽度和长度的正方形。

[0048] 根据本发明的实施例的编码单元可由最大尺寸和深度表征。深度表示编码单元从最大编码单元被空间划分的次数，并且随着深度加深，根据深度的较深层编码单元可从最大编码单元被划分到最小编码单元。最大编码单元的深度为最高深度，最小编码单元的深度为最低深度。由于随着最大编码单元的深度加深，与每个深度相应的编码单元的尺寸减小，因此与更高深度相应的编码单元可包括多个与更低深度相应的编码单元。

[0049] 如上所述，当前画面的图像数据根据编码单元的最大尺寸被划分为最大编码单元，并且每个最大编码单元可包括根据深度被划分的较深层编码单元。由于根据深度对根据本发明的实施例的最大编码单元进行划分，因此可根据深度对在最大编码单元中包括的空间域的图像数据进行分层地分类。

[0050] 可预先确定编码单元的最大深度和最大尺寸，所述最大深度和最大尺寸限制对最大编码单元的高度和宽度进行分层划分的次数。

[0051] 分层编码器110对通过根据深度对最大编码单元的区域进行划分而获得的至少一个划分区域进行编码，并且根据所述至少一个划分区域来确定用于输出最终编码的图像数据的深度。换句话说，分层编码器110通过根据当前画面的最大编码单元以根据深度的较深层编码单元对图像数据进行编码，选择具有最小编码误差的深度，来确定编码深度。确定的编码深度和根据最大编码单元的被编码的图像数据被输出到熵编码器120。

[0052] 基于与等于或小于最大深度的至少一个深度相应的较深层编码单元，对最大编码单元中的图像数据进行编码，并且基于每个较深层编码单元来比较对图像数据进行编码的结果。在对较深层编码单元的编码误差进行比较之后，可选择具有最小编码误差的深度。可针对每个最大编码单元选择至少一个编码深度。

[0053] 随着编码单元根据深度被分层地划分并随着编码单元的数量增加，最大编码单元的尺寸被划分。另外，即使在一个最大编码单元中编码单元与同一深度相应，也通过分别测量每个编码单元的图像数据的编码误差来确定是否将与同一深度相应的每个编码单元划分为更低深度。因此，即使图像数据被包括在一个最大编码单元中，图像数据仍被划分为根据深度的区域，并且在一个最大编码单元中编码误差根据区域而不同，因此在图像数据中编码深度可根据区域而不同。因此，可在一个最大编码单元中确定一个或更多个编码深度，并且可根据至少一个编码深度的编码单元来对最大编码单元的图像数据进行划分。

[0054] 因此，分层编码器110可确定在最大编码单元中包括的具有树结构的编码单元。根据本发明的实施例的“具有树结构的编码单元”包括在最大编码单元中包括的所有较深层编码单元中的与确定为编码深度的深度相应的编码单元。可在最大编码单元的同一区域中根据深度来分层地确定具有编码深度的编码单元，并可在不同区域中独立地确定具有编码深度的编码单元。类似地，当前区域中的编码深度可独立于另一区域的编码深度而被确定。

[0055] 根据本发明的实施例的最大深度是与从最大编码单元到最小编码单元执行划分的次数有关的索引。根据本发明的实施例的第一最大深度可表示从最大编码单元到最小编码单元执行划分的总次数。根据本发明实施例的第二最大深度可表示从最大编码单元到最小编码单元的深度等级的总数。例如，当最大编码单元的深度是0时，对最大编码单元划分

一次的编码单元的深度可被设置为1，对最大编码单元划分两次的编码单元的深度可被设置为2。这里，如果最小编码单元是对最大编码单元划分四次的编码单元，则存在深度0、1、2、3和4的5个深度等级，并因此第一最大深度可被设置为4，第二最大深度可被设置为5。

[0056] 可根据最大编码单元执行预测编码和变换。还根据最大编码单元，基于根据等于或小于最大深度的深度的较深层编码单元来执行预测编码和变换。

[0057] 由于每当根据深度对最大编码单元进行划分时，较深层编码单元的数量增加，因此对随着深度加深而产生的所有较深层编码单元执行包括预测编码和变换的编码。为了便于描述，在最大编码单元中，现在将基于当前深度的编码单元来描述预测编码和变换。

[0058] 视频编码设备100可不同地选择用于对图像数据进行编码的数据单元的尺寸或形状。为了对图像数据进行编码，执行诸如预测编码、变换和熵编码的操作，此时，可针对所有操作使用相同的数据单元，或者可针对每个操作使用不同的数据单元。

[0059] 例如，视频编码设备100不仅可选择用于对图像数据进行编码的编码单元，还可选择不同于编码单元的数据单元，以便对编码单元中的图像数据执行预测编码。

[0060] 为了对最大编码单元执行预测编码，可基于与编码深度相应的编码单元(即，不再被划分成与更低深度相应的编码单元的编码单元)来执行预测编码。以下，不再被划分且成为用于预测编码的基本单元的编码单元现在将被称为“预测单元”。通过划分预测单元获得的分区可包括预测单元或通过对预测单元的高度和宽度中的至少一个进行划分而获得的数据单元。

[0061] 例如，当2N×2N(其中，N是正整数)的编码单元不再被划分，并且成为2N×2N的预测单元时，分区的尺寸可以是2N×2N、2N×N、N×2N或N×N。分区类型的示例包括通过对预测单元的高度或宽度进行对称地划分而获得的对称分区、通过对预测单元的高度或宽度进行非对称地划分(诸如，1:n或n:1)而获得的分区、通过对预测单元进行几何地划分而获得的分区、以及具有任意形状的分区。

[0062] 预测单元的预测模式可以是帧内模式、帧间模式和跳过模式中的至少一个。例如，可对2N×2N、2N×N、N×2N或N×N的分区执行帧内模式或帧间模式。另外，可仅对2N×2N的分区执行跳过模式。可对编码单元中的一个预测单元独立地执行编码，从而选择具有最小编码误差的预测模式。

[0063] 视频编码设备100不仅可基于用于对图像数据进行编码的编码单元还可基于与编码单元不同的数据单元，来对编码单元中的图像数据执行变换。

[0064] 为了对编码单元执行变换，可基于具有小于或等于编码单元的尺寸的数据单元，来执行变换。例如，用于变换的数据单元可包括用于帧内模式的数据单元和用于帧间模式的数据单元。

[0065] 现在将用作变换的基础的数据单元称为“变换单元”。与编码单元类似，编码单元中的变换单元可被递归地划分为更小尺寸的区域，使得变换单元可以以区域为单位被独立地确定。因此，可基于根据变换深度的具有树结构的变换单元，对编码单元中的残差数据进行划分。

[0066] 还可在变换单元中设置变换深度，其中，变换深度表示对编码单元的高度和宽度进行划分以获得变换单元所执行的划分的次数。例如，在2N×2N的当前编码单元中，当变换单元的尺寸是2N×2N时，变换深度可以是0，当变换单元的尺寸是N×N时，变换深度可以是

1，当变换单元的尺寸是N/2×N/2时，变换深度可以是2。也就是说，还可根据变换深度设置具有树结构的变换单元。

[0067] 根据与编码深度相应的编码单元的编码信息不仅需要关于编码深度的信息，还需要关于与预测编码和变换相关的信息的信息。因此，分层编码器110不仅确定具有最小编码误差的编码深度，还确定预测单元中的分区类型、根据预测单元的预测模式和用于变换的变换单元的尺寸。

[0068] 稍后将参照图3至图12详细描述根据本发明的实施例的最大编码单元中的根据树结构的编码单元，以及确定分区的方法。

[0069] 分层编码器110可通过使用基于拉格朗日乘数的率失真优化，来测量根据深度的较深层编码单元的编码误差。

[0070] 熵编码器120在比特流中输出最大编码单元的图像数据和关于根据编码深度的编码模式的信息，其中，所述最大编码单元的图像数据基于由分层编码器110确定的至少一个编码深度被编码。编码图像数据可以是图像的残差数据的编码结果。关于根据编码深度的编码模式的信息可包括关于编码深度的信息、关于在预测单元中的分区类型的信息、预测模式信息和变换单元的尺寸信息。具体地讲，如稍后将描述的，熵编码器120可基于在变换单元的颜色分量信息中是否存在具有比第一临界值更大的值的有效变换系数、当前子集的位置以及先前子集，来获得指示多个上下文集中的一个上下文集的上下文集索引，以便基于具有连续的1的先前变换系数的长度来获得上下文偏移。另外，熵编码器120基于获得的上下文集索引和获得的上下文偏移来确定指示上下文模型的上下文索引ctdIdx，其中，上下文模型将被应用于第一临界值标记Greaterthan1标记和第二临界值标记Greaterthan2标记，其中，第一临界值标记Greaterthan1标记指示有效变换系数是否大于第一临界值(即，1)，第二临界值标记Greaterthan2标记指示有效变换系数是否大于第二临界值(即，2)。稍后将描述将由熵编码器120执行的确定用于对变换系数进行熵编码的上下文模型的操作。

[0071] 可通过使用根据深度的划分信息来定义关于编码深度的信息，其中，根据深度的划分信息指示是否对更低深度而不是当前深度的编码单元执行编码。如果当前编码单元的当前深度是编码深度，则对当前编码单元中的图像数据进行编码并输出，因此可定义划分信息以不将当前编码单元划分到更低深度。可选地，如果当前编码单元的当前深度不是编码深度，则对更低深度的编码单元执行编码，并因此可定义划分信息以对当前编码单元进行划分来获得更低深度的编码单元。

[0072] 如果当前深度不是编码深度，则对被划分到更低深度的编码单元的编码单元执行编码。由于更低深度的至少一个编码单元存在于当前深度的一个编码单元中，因此对更低深度的每个编码单元重复执行编码，并因此可对具有相同深度的编码单元递归地执行编码。

[0073] 由于针对一个最大编码单元确定具有树结构的编码单元，并且针对编码深度的编码单元确定关于至少一个编码模式的信息，所以可针对一个最大编码单元确定关于至少一个编码模式的信息。另外，由于根据深度对图像数据进行分层划分，因此最大编码单元的图像数据的编码深度可根据位置而不同，因此可针对图像数据设置关于编码深度和编码模式的信息。

[0074] 因此，熵编码器120可将关于相应的编码深度和编码模式的编码信息分配给包括在最大编码单元中的编码单元、预测单元和最小单元中的至少一个。

[0075] 根据本发明的实施例的最小单元是通过将构成最低深度的最小编码单元划分为4份而获得的方形数据单元。可选择地，最小单元可以是包括在最大编码单元中所包括的所有编码单元、预测单元、分区单元和变换单元中的最大方形数据单元。

[0076] 例如，通过熵编码器120输出的编码信息可被分类为根据深度的较深层编码单元的编码信息和根据预测单元的编码信息。根据深度的较深层编码单元的编码信息可包括关于预测模式的信息和关于分区尺寸的信息。根据预测单元的编码信息可包括关于帧间模式的估计方向的信息、关于帧间模式的参考图像索引的信息、关于运动矢量的信息、关于帧内模式的色度分量的信息、以及关于帧内模式的插值方法的信息。此外，根据画面、条带或GOP定义的关于编码单元的最大尺寸的信息和关于最大深度的信息可被插入到比特流的头。

[0077] 在视频编码设备100中，较深层编码单元可以是通过将更高深度的编码单元(更高一层)的高度或宽度划分成两份而获得的编码单元。换言之，当当前深度的编码单元的尺寸是2N×2N时，更低深度的编码单元的尺寸是N×N。另外，尺寸为2N×2N的当前深度的编码单元可包括最多4个更低深度的编码单元。

[0078] 因此，视频编码设备100可基于考虑当前画面的特征而确定的最大编码单元的尺寸和最大深度，通过针对每个最大编码单元确定具有最优形状和最优尺寸的编码单元来形成具有树结构的编码单元。另外，由于可通过使用各种预测模式和变换中的任意一个对每个最大编码单元执行编码，因此可考虑各种图像尺寸的编码单元的特征来确定最优编码模式。

[0079] 因此，如果以传统宏块对具有高分辨率或大数据量的图像进行编码，则每个画面的宏块的数量极度增加。因此，针对每个宏块产生的压缩信息的条数增加，因此难以发送压缩的信息，并且数据压缩效率降低。因此，通过使用视频编码设备100，由于考虑图像的尺寸，在增加编码单元的最大尺寸的同时，基于图像的特征来调整编码单元，因此可增加图像压缩效率。

[0080] 图2是根据本发明的实施例的视频解码设备200的框图。

[0081] 视频解码设备200包括句法元素提取单元210、熵解码器220和分层解码器230。用于视频解码设备200的解码操作的各种术语(诸如编码单元、深度、预测单元、变换单元和关于各种编码模式的信息)的定义与参照图1和视频编码设备100描述的定义相同。

[0082] 句法元素提取单元210接收和解析编码视频的比特流。熵解码器220从解析的比特流，针对每个编码单元提取编码图像数据，并将提取的图像数据输出到分层解码器230，其中，编码单元具有根据每个最大编码单元的树结构。

[0083] 另外，熵解码器220从解析的比特流，根据每个最大编码单元，提取关于具有树结构的编码单元的编码深度、编码模式、颜色分量信息、预测模式信息等的信息。提取的关于编码深度和编码模式的信息被输出到分层解码器230。比特流中的图像数据被划分为最大编码单元，使得分层解码器230可针对每个最大编码单元对图像数据进行解码。

[0084] 可针对关于与编码深度相应的至少一个编码单元的信息设置关于根据最大编码单元的编码深度和编码模式的信息，关于编码模式的信息可包括关于与编码深度相应的相应编码单元的分区类型的信息、关于预测模式的信息和关于变换单元的尺寸的信息。另外，

根据深度的划分信息可被提取为关于编码深度的信息。

[0085] 由熵解码器220提取的关于根据每个最大编码单元的编码深度和编码模式的信息是关于在以下情况下被确定为产生最小编码误差的编码深度和编码模式的信息，即，在编码端(诸如，视频编码设备100)根据最大编码单元对每个根据深度的较深层编码单元重复地执行编码的时候。因此，视频解码设备200可通过根据产生最小编码误差的编码深度和编码模式对图像数据进行解码来恢复图像。

[0086] 由于关于编码深度和编码模式的编码信息可被分配给相应的编码单元、预测单元和最小单元中的预定数据单元，因此熵解码器220可根据预定数据单元，提取关于编码深度和编码模式的信息。可将被分配了相同的关于编码深度和编码模式的信息的预定数据单元推断为是包括在同一最大编码单元中的数据单元。

[0087] 此外，如稍后将描述的，熵解码器220可基于在变换单元的颜色分量信息中是否存在具有比第一临界值更大的值的有效变换系数、当前子集的位置以及先前子集，来获得指示多个上下文集中的一个上下文集的上下文集索引，以便基于具有连续的1的先前变换系数的长度来获得上下文偏移。另外，熵解码器220基于获得的上下文集索引和获得的上下文偏移来确定指示上下文模型的上下文索引ctdIdx，其中，上下文模型将被应用于第一临界值标记Greaterthan1标记和第二临界值标记Greaterthan2标记，其中，第一临界值标记Greaterthan1标记指示有效变换系数是否大于第一临界值(即，1)，第二临界值Greaterthan2标记指示有效变换系数是否大于第二临界值(即，2)。

[0088] 分层解码器230基于关于根据最大编码单元的编码深度和编码模式的信息，通过对每个最大编码单元中的图像数据进行解码，来恢复当前画面。换言之，分层解码器230可基于提取出的关于包括在每个最大编码单元中的具有树结构的编码单元之中的每个编码单元的分区类型、预测模式和变换单元的信息，对编码的图像数据进行解码。解码处理可包括预测(包含帧内预测和运动补偿)和逆变换。

[0089] 分层解码器230可基于关于根据编码深度的编码单元的预测单元的分区类型和预测模式的信息，根据每个编码单元的分区和预测模式，执行帧内预测或运动补偿。

[0090] 此外，分层解码器230可基于关于根据编码深度的编码单元的变换单元的尺寸的信息，根据编码单元中的每个变换单元执行逆变换，以便根据最大编码单元执行逆变换。

[0091] 分层解码器230可通过使用根据深度的划分信息来确定当前最大编码单元的至少一个编码深度。如果划分信息指示图像数据在当前深度中不再被划分，则当前深度是编码深度。因此，分层解码器230可通过使用关于预测单元的分区类型的信息、关于预测模式的信息和关于变换单元的尺寸的信息，针对当前最大编码单元的图像数据，对当前深度的编码单元进行解码。

[0092] 换言之，可通过观察分配给编码单元、预测单元和最小单元中的预定数据单元的编码信息集来收集包含包括相同划分信息的编码信息的数据单元，并且收集的数据单元可被认为是将由分层解码器230以相同编码模式进行解码的一个数据单元。

[0093] 视频解码设备200可获得关于当针对每个最大编码单元递归地执行编码时产生最小编码误差的至少一个编码单元的信息，并且可使用所述信息来对当前画面进行解码。换言之，被确定为最大编码单元中的最优编码单元的具有树结构的编码单元的编码的图像数据可被解码。

[0094] 因此，即使图像数据具有高分辨率和大数据量，也可通过使用编码单元的尺寸和编码模式，对图像数据进行有效地解码和恢复，其中，通过使用从编码器接收到的关于最优编码模式的信息，根据图像数据的特征自适应地确定所述编码单元的尺寸和编码模式。

[0095] 现在将参照图3至图13描述根据本发明的实施例的确定具有树结构的编码单元、预测单元和变换单元的方法。

[0096] 图3是用于描述根据本发明的实施例的编码单元的概念的示图。

[0097] 编码单元的尺寸可被表示为宽度×高度，并可以是64×64、32×32、16×16和8×8。64×64的编码单元可被划分为64×64、64×32、32×64或32×32的分区，32×32的编码单元可被划分为32×32、32×16、16×32或16×16的分区，16×16的编码单元可被划分为16×16、16×8、8×16或8×8的分区，8×8的编码单元可被划分为8×8、8×4、4×8或4×4的分区。

[0098] 在视频数据310中，分辨率是1920×1080，编码单元的最大尺寸是64，最大深度是2。在视频数据320中，分辨率是1920×1080，编码单元的最大尺寸是64，最大深度是3。在视频数据330中，分辨率是352×288，编码单元的最大尺寸是16，最大深度是1。图3中示出的最大深度表示从最大编码单元到最小编码单元的划分总次数。

[0099] 如果分辨率高或数据量大，则编码单元的最大尺寸可能较大，从而不仅提高编码效率，而且准确地反映图像的特征。因此，具有比视频数据330更高分辨率的视频数据310和320的编码单元的最大尺寸可以是64。

[0100] 由于视频数据310的最大深度是2，因此由于通过对最大编码单元划分两次，深度加深至两层，因此视频数据310的编码单元315可包括长轴尺寸为64的最大编码单元和长轴尺寸为32和16的编码单元。同时，由于视频数据330的最大深度是1，因此由于通过对最大编码单元划分一次，深度加深至一层，因此视频数据330的编码单元335可包括长轴尺寸为16的最大编码单元和长轴尺寸为8的编码单元。

[0101] 由于视频数据320的最大深度是3，因此由于通过对最大编码单元划分三次，深度加深至3层，因此视频数据320的编码单元325可包括长轴尺寸为64的最大编码单元和长轴尺寸为32、16和8的编码单元。随着深度加深，详细信息可被精确地表示。

[0102] 图4是根据本发明的实施例的基于具有分层结构的编码单元的视频编码器400的框图。

[0103] 帧内预测器410针对当前帧405在帧内模式下对编码单元执行帧内预测，运动估计器420和运动补偿器425通过使用当前帧405和参考帧495在帧间模式下分别对编码单元执行帧间估计和运动补偿。

[0104] 从帧内预测器410、运动估计器420和运动补偿器425输出的数据通过变换器430和量化器440被输出为量化后的变换系数。量化后的变换系数通过反量化器460和逆变换器470被恢复为空间域中的数据，恢复的空间域中的数据在通过去块单元480和环路滤波单元490后处理之后被输出为参考帧495。量化后的变换系数可通过熵编码器450被输出为比特流455。

[0105] 当对变换单元的句法元素(诸如第一临界值标记Gtr1标记或第二临界值标记Gtr2标记)进行编码时，熵编码器450基于在变换单元的颜色分量信息中是否存在具有比第一临界值更大的值的有效变换系数、当前子集的位置和先前子集，来获得上下文集索引，基于具

有连续的1的先前变换系数的长度来获得上下文偏移，并基于获得的上下文集索引和获得的上下文偏移来确定指示上下文模型的上下文索引。

[0106] 为了将图像编码器400应用到视频编码设备100中，图像编码器400的所有元件(即，帧内预测器410、运动估计器420、运动补偿器425、变换器430、量化器440、熵编码器450、反量化器460、逆变换器470、去块单元480和环路滤波单元490)在考虑每个最大编码单元的最大深度的同时，基于具有树结构的编码单元中的每个编码单元执行操作。

[0107] 具体地，帧内预测器410、运动估计器420和运动补偿器425在考虑当前最大编码单元的最大尺寸和最大深度的同时，确定具有树结构的编码单元中的每个编码单元的分区和预测模式，变换器430确定具有树结构的编码单元中的每个编码单元中的变换单元的尺寸。

[0108] 图5是根据本发明的实施例的基于编码单元的视频解码器500的框图。

[0109] 解析器510从比特流505解析将被解码的编码图像数据和解码所需的编码信息。编码图像数据通过熵解码器520和反量化器530被输出为反量化的数据，反量化的数据通过逆变换器540被恢复为空间域中的图像数据。

[0110] 针对空间域中的图像数据，帧内预测器550在帧内模式下对编码单元执行帧内预测，运动补偿器560通过使用参考帧585在帧间模式下对编码单元执行运动补偿。

[0111] 通过帧内预测器550和运动补偿器560的空间域中的图像数据可在通过去块单元570和环路滤波单元580后处理之后被输出为恢复帧595。另外，通过去块单元570和环路滤波单元580后处理的图像数据可被输出为参考帧585。

[0112] 为了将图像解码器500应用到视频解码设备200中，图像解码器500的所有元件(即，解析器510、熵解码器520、反量化器530、逆变换器540、帧内预测器550、运动补偿器560、去块单元570和环路滤波单元580)针对每个最大编码单元，基于具有树结构的编码单元执行操作。

[0113] 帧内预测器550和运动补偿器560确定具有树结构的每个编码单元的分区和预测模式，逆变换器540必须确定每个编码单元的变换单元的尺寸。此外，当对变换单元的句法元素(诸如第一临界值标记Gtr1标记或第二临界值标记Gtr2标记)进行解码时，熵解码器520基于在变换单元的颜色分量信息中是否存在具有比第一临界值更大的值的有效变换系数、当前子集的位置和先前子集，来获得上下文集索引，基于具有连续的1的先前变换系数的长度来获得上下文偏移，并基于获得的上下文索引和获得的上下文偏移来确定指示上下文模型的上下文索引。

[0114] 图6是示出根据本发明的实施例的根据深度的较深层编码单元以及分区的示图。

[0115] 视频编码设备100和视频解码设备200使用分层编码单元以考虑图像的特征。可根据图像的特征自适应地确定编码单元的最大高度、最大宽度和最大深度，或可由用户不同地设置编码单元的最大高度、最大宽度和最大深度。可根据编码单元的预定最大尺寸来确定根据深度的较深层编码单元的尺寸。

[0116] 在根据本发明的实施例的编码单元的分层结构600中，编码单元的最大高度和最大宽度均是64，最大深度是4。由于沿着分层结构600的垂直轴深度加深，因此较深层编码单元的高度和宽度均被划分。另外，预测单元和分区沿着分层结构600的水平轴被示出，其中，所述预测单元和分区是对每个较深层编码单元进行预测编码的基础。

[0117] 换言之，在分层结构600中，编码单元610是最大编码单元，其中，深度为0，尺寸

(即，高度乘宽度)为64×64。随着深度沿着垂直轴加深，存在尺寸为32×32和深度为1的编码单元620、尺寸为16×16和深度为2的编码单元630、尺寸为8×8和深度为3的编码单元640、尺寸为4×4和深度为4的编码单元650。尺寸为4×4和深度为4的编码单元650是最小编码单元。

[0118] 编码单元的预测单元和分区根据每个深度沿着水平轴被排列。换言之，如果尺寸为64×64和深度为0的编码单元610是预测单元，则可将预测单元划分成包括在编码单元610中的分区，即，尺寸为64×64的分区610、尺寸为64×32的分区612、尺寸为32×64的分区614或尺寸为32×32的分区616。

[0119] 类似地，可将尺寸为32×32和深度为1的编码单元620的预测单元划分成包括在编码单元620中的分区，即，尺寸为32×32的分区620、尺寸为32×16的分区622、尺寸为16×32的分区624和尺寸为16×16的分区626。

[0120] 类似地，可将尺寸为16×16和深度为2的编码单元630的预测单元划分成包括在编码单元630中的分区，即，包括在编码度单元630中的尺寸为16×16的分区630、尺寸为16×8的分区632、尺寸为8×16的分区634和尺寸为8×8的分区636。

[0121] 类似地，可将尺寸为8×8和深度为3的编码单元640的预测单元划分成包括在编码单元640中的分区，即，包括在编码单元640中的尺寸为8×8的分区、尺寸为8×4的分区642、尺寸为4×8的分区644和尺寸为4×4的分区646。

[0122] 尺寸为4×4和深度为4的编码单元650是最小编码单元以及最低深度的编码单元。编码单元650的预测单元仅被分配给尺寸为4×4的分区。

[0123] 为了确定构成最大编码单元610的编码单元的至少一个编码深度，视频编码设备100的编码单元确定器120对包括在最大编码单元610中的与每个深度相应的编码单元执行编码。

[0124] 随着深度加深，包括具有相同范围和相同尺寸的数据的根据深度的较深层编码单元的数量增加。例如，需要四个与深度2相应的编码单元来覆盖包括在与深度1相应的一个编码单元中的数据。因此，为了根据深度比较对相同数据进行编码的结果，与深度1相应的编码单元和四个与深度2相应的编码单元均被编码。

[0125] 为了针对深度之中的当前深度执行编码，可沿着分层结构600的水平轴，通过对与当前深度相应的编码单元中的每个预测单元执行编码，来将针对当前深度选择最小编码误差。可选地，随着深度沿着分层结构600的垂直轴加深，可通过针对每个深度执行编码，比较根据深度的最小编码误差，来搜索最小编码误差。在编码单元610中的具有最小编码误差的深度和分区可被选为编码单元610的编码深度和分区类型。

[0126] 图7是用于描述根据本发明的实施例的在编码单元710和变换单元720之间的关系的示图。

[0127] 视频编码设备100或视频解码设备200针对每个最大编码单元，根据具有小于或等于最大编码单元的尺寸的编码单元，对图像进行编码或解码。可基于不大于相应的编码单元的数据单元，来选择用于在编码期间进行变换的变换单元的尺寸。

[0128] 例如，在视频编码设备100或视频解码设备200中，如果编码单元710的尺寸是64×64，则可通过使用尺寸为32×32的变换单元720来执行变换。

[0129] 此外，可通过对小于64×64的尺寸为32×32、16×16、8×8和4×4的每个变换单元

执行变换，来对尺寸为64×64的编码单元710的数据进行编码，然后可选择具有最小编码误差的变换单元。

[0130] 图8是用于描述根据本发明的实施例的与编码深度相应的编码单元的编码信息的示图。

[0131] 视频编码设备100的输出单元130可对与编码深度相应的每个编码单元的关于分区类型的信息800、关于预测模式的信息810以及关于变换单元尺寸的信息820进行编码，并将信息800、信息810和信息820作为关于编码模式的信息来发送。

[0132] 信息800指示关于通过划分当前编码单元的预测单元而获得的分区的形状的信息，其中，所述分区是用于对当前编码单元进行预测编码的数据单元。例如，可将尺寸为2N×2N的当前编码单元CU_0划分成以下分区中的任意一个：尺寸为2N×2N的分区802、尺寸为2N×N的分区804、尺寸为N×2N的分区806以及尺寸为N×N的分区808。这里，关于分区类型的信息800被设置来指示尺寸为2N×N的分区804、尺寸为N×2N的分区806以及尺寸为N×N的分区808中的一个。

[0133] 信息810指示每个分区的预测模式。例如，信息810可指示对由信息800指示的分区执行的预测编码的模式，即，帧内模式812、帧间模式814或跳过模式816。

[0134] 信息820指示当对当前编码单元执行变换时所基于的变换单元。例如，变换单元可以是第一帧内变换单元822、第二帧内变换单元824、第一帧间变换单元826或第二帧内变换单元828。

[0135] 视频解码设备200的熵解码器220可根据每个较深层编码单元，提取并使用用于解码的信息800、810和820。

[0136] 图9是根据本发明的实施例的根据深度的较深层编码单元的示图。

[0137] 划分信息可用来指示深度的改变。划分信息指示当前深度的编码单元是否被划分成更低深度的编码单元。

[0138] 用于对深度为0和尺寸为2N_0×2N_0的编码单元900进行预测编码的预测单元910可包括以下分区类型的分区：尺寸为2N_0×2N_0的分区类型912、尺寸为2N_0×N_0的分区类型914、尺寸为N_0×2N_0的分区类型916和尺寸为N_0×N_0的分区类型918。图9仅示出了通过对称地划分预测单元910而获得的分区类型912至918，但是分区类型不限于此，并且预测单元910的分区可包括非对称分区、具有预定形状的分区和具有几何形状的分区。

[0139] 根据每种分区类型，对尺寸为2N_0×2N_0的一个分区、尺寸为2N_0×N_0的两个分区、尺寸为N_0×2N_0的两个分区和尺寸为N_0×N_0的四个分区重复地执行预测编码。可对尺寸为2N_0×2N_0、N_0×2N_0、2N_0×N_0和N_0×N_0的分区执行帧内模式和帧间模式下的预测编码。可仅对尺寸为2N_0×2N_0的分区执行跳过模式下的预测编码。

[0140] 如果在尺寸为2N_0×2N_0、2N_0×N_0和N_0×2N_0的分区类型912至916中的一个分区类型中编码误差最小，则可不将预测单元910划分到更低深度。

[0141] 如果在尺寸为N_0×N_0的分区类型918中编码误差最小，则深度从0改变到1以在操作920中划分分区类型918，并对深度为2和尺寸为N_0×N_0的分区类型编码单元重复地执行编码来搜索最小编码误差。

[0142] 用于对深度为1和尺寸为2N_1×2N_1(=N_0×N_0)的(分区类型)编码单元930进行预测编码的预测单元940可包括以下分区类型的分区：尺寸为2N_1×2N_1的分区类型

942、尺寸为2N_1×N_1的分区类型944、尺寸为N_1×2N_1的分区类型946以及尺寸为N_1×N_1的分区类型948。

[0143] 如果在尺寸为N_1×N_1的分区类型948中编码误差最小，则深度从1改变到2以在操作950中划分分区类型948，并对深度为2和尺寸为N_2×N_2的编码单元960重复执行编码来搜索最小编码误差。

[0144] 当最大深度是d时，根据每个深度的划分操作可被执行直到深度变成d-1，并且划分信息可被编码直到深度是0到d-2中的一个。换句话说，当编码被执行直到在与d-2的深度相应的编码单元在操作970中被划分之后深度是d-1时，用于对深度为d-1和尺寸为2N_(d-1)×2N_(d-1)的编码单元980进行预测编码的预测单元990可包括以下分区类型的分区：尺寸为2N_(d-1)×2N(d-1)的分区类型992、尺寸为2N_(d-1)×N(d-1)的分区类型994、尺寸为N_(d-1)×2N(d-1)的分区类型996和尺寸为N_(d-1)×N(d-1)的分区类型998。

[0145] 可对分区类型992至998中的尺寸为2N_(d-1)×2N_(d-1)的一个分区、尺寸为2N_(d-1)×N_(d-1)的两个分区、尺寸为N_(d-1)×2N_(d-1)的两个分区、尺寸为N_(d-1)×N_(d-1)的四个分区重复地执行预测编码，以搜索具有最小编码误差的分区类型。

[0146] 即使当尺寸为N_(d-1)×N_(d-1)的分区类型998具有最小编码误差时，由于最大深度是d，因此深度为d-1的编码单元CU_(d-1)也不再被划分到更低深度，构成当前最大编码单元900的编码单元的编码深度被确定为d-1，并且当前最大编码单元900的分区类型可被确定为N_(d-1)×N(d-1)。此外，由于最大深度是d，因此不设置深度为d-1的编码单元952的划分信息。

[0147] 数据单元999可以是用于当前最大编码单元的“最小单元”。根据本发明的实施例的最小单元可以是通过将最小编码单元980划分成4份而获得的矩形数据单元。通过重复地执行编码，视频编码设备100可通过比较根据编码单元900的深度的编码误差来选择具有最小编码误差的深度以确定编码深度，并将相应分区类型和预测模式设置为编码深度的编码模式。

[0148] 这样，在所有深度1至d中对根据深度的最小编码误差进行比较，并且具有最小编码误差的深度可被确定为编码深度。编码深度、预测单元的分区类型和预测模式可作为关于编码模式的信息被编码并发送。另外，由于编码单元从0的深度被划分到编码深度，因此仅编码深度的划分信息被设置为0，并且除了编码深度以外的深度的划分信息被设置为1。

[0149] 视频解码设备200的熵解码器220可提取并使用关于编码单元900的编码深度和预测单元的信息，来对编码单元912进行解码。视频解码设备200可通过使用根据深度的划分信息，将划分信息为0的深度确定为编码深度，并且使用关于相应深度的编码模式的信息来进行解码。

[0150] 图10至图12是用于描述根据本发明的实施例的编码单元1010、预测单元1060和变换单元1070之间的关系的示图。

[0151] 编码单元1010是最大编码单元中的与由视频编码设备100确定的编码深度相应的具有树结构的编码单元。预测单元1060是每个编码单元1010中的预测单元的分区，变换单元1070是每个编码单元1010的变换单元。

[0152] 当在编码单元1010中最大编码单元的深度是0时，编码单元1012和编码单元1054的深度是1，编码单元1014、1016、1018、1028、1050和1052的深度是2，编码单元1020、1022、

1024、1026、1030、1032和1048的深度是3，编码单元1040、1042、1044和1046的深度是4。

[0153] 在预测单元1060中，通过划分编码单元来获得一些编码单元1014、1016、1022、1032、1048、1050、1052和1054。换句话说，编码单元1014、1022、1050和1054中的分区类型的尺寸是2N×N，编码单元1016、1048和1052中的分区类型的尺寸是N×2N，编码单元1032的分区类型的尺寸就N×N。编码单元1010的预测单元和分区小于或等于每个编码单元。

[0154] 在小于编码单元1052的数据单元中的变换单元1070中，对编码单元1052的图像数据执行变换或逆变换。另外，在尺寸和形状方面，变换单元1070中的编码单元1014、1016、1022、1032、1048、1050、1052和1054不同于预测单元1060中的编码单元1014、1016、1022、1032、1048、1050、1052和1054。换句话说，视频编码设备100和视频解码设备200可对同一编码单元中的数据单元独立地执行帧内预测、运动估计、运动补偿、变换和逆变换。

[0155] 因此，对最大编码单元的每个区域中的具有分层结构的每个编码单元递归地执行编码来确定最优编码单元，从而可获得具有递归树结构的编码单元。编码信息可包括关于编码单元的划分信息、关于分区类型的信息、关于预测模式的信息和关于变换单元的尺寸的信息。表1示出可由视频编码设备100和视频解码设备200设置的编码信息。

[0156] [表1]

[0157]

<table>
<tr><td colspan="5">划分信息0（对尺寸为2N×2N和当前深度为d的编码单元进行编码）</td><td>划分信息1</td></tr>
<tr><td>预测模式</td><td colspan="2">分区类型</td><td colspan="2">变换单元的尺寸</td><td rowspan="3">对具有更低深度d+1的编码单元重复编码</td></tr>
<tr><td rowspan="2">帧内
帧间
跳过（仅2N×2N）</td><td>对称分区类型</td><td>非对称分区类型</td><td>变换单元的划分信息0</td><td>变换单元的划分信息1</td></tr>
<tr><td>2N×2N
2N×N
N×2N
N×N</td><td>2N×nU
2N×nD
nL×2N
nR×2N</td><td>2N×2N</td><td>N×N（对称类型）
N/2×N/2（非对称类型）</td></tr>
</table>

[0158] 视频编码设备100的熵编码器120可输出关于具有树结构的编码单元的编码信息，视频解码设备200的熵解码器220可从接收到的比特流提取关于具有树结构的编码单元的编码信息。

[0159] 划分信息指示是否将当前编码单元划分成更低深度的编码单元。如果当前深度d的划分信息是0，则当前编码单元不再被划分成更低深度的深度是编码深度，从而可针对所述编码深度来定义关于分区类型、预测模式和变换单元的尺寸的信息。如果当前编码单元根据划分信息被进一步划分，则对更低深度的四个划分编码单元独立地执行编码。

[0160] 预测模式可以是帧内模式、帧间模式和跳过模式中的一种。可针对所有分区类型

定义帧内模式和帧间模式，仅针对尺寸为2N×2N的分区类型定义跳过模式。

[0161] 关于分区类型的信息可指示通过对称地划分预测单元的高度或宽度而获得的尺寸为2N×2N、2N×N、N×2N和N×N的对称分区类型，以及通过非对称地划分预测单元的高度或宽度而获得的尺寸为2N×nU、2N×nD、nL×2N和nR×2N的非对称分区类型。可通过按1:n和n:1(其中，n是大于1的整数)来划分预测单元的高度来分别获得尺寸为2N×nU和2N×nD的非对称分区类型，可通过按1:n和n:1来划分预测单元的宽度来分别获得尺寸为nL×2N和nR×2N的非对称分区类型。

[0162] 可将变换单元的尺寸设置成帧内模式下的两种类型和帧间模式下的两种类型。换句话说，如果变换单元的划分信息是0，则变换单元的尺寸可以是2N×2N，即当前编码单元的尺寸。如果变换单元的划分信息是1，则可通过对当前编码单元进行划分来获得变换单元。另外，如果尺寸为2N×2N的当前编码单元的分区类型是对称分区类型时，则变换单元的尺寸可以是N×N，如果当前编码单元的分区类型是非对称分区类型，则变换单元的尺寸可以是N/2×N/2。

[0163] 关于具有树结构的编码单元的编码信息可包括与编码深度相应的编码单元、预测单元和最小单元中的至少一个。与编码深度相应的编码单元可包括包含相同编码信息的预测单元和最小单元中的至少一个。

[0164] 因此，通过比较邻近数据单元的编码信息来确定邻近数据单元是否被包括在与编码深度相应的同一编码单元中。另外，通过使用数据单元的编码信息来确定与编码深度相应的相应编码单元，并因此可确定最大编码单元中的编码深度的分布。

[0165] 因此，如果基于邻近数据单元的编码信息来对当前编码单元进行预测，则可直接参考并使用与当前编码单元邻近的较深层编码单元中的数据单元的编码信息。

[0166] 可选地，如果基于邻近数据单元的编码信息来对当前编码单元进行预测，则使用数据单元的编码信息来搜索与当前编码单元邻近的数据单元，并可参考搜索到的邻近编码单元以对当前编码单元进行预测。

[0167] 图13是用于描述根据表1的编码模式信息的编码单元、预测单元和变换单元之间的关系的示图。

[0168] 最大编码单元1300包括编码深度的编码单元1302、1304、1306、1312、1314、1316和1318。这里，由于编码单元1318是编码深度的编码单元，因此划分信息可以被设置成0。可将关于尺寸为2N×2N的编码单元1318的分区类型的信息设置成以下分区类型中的一种：尺寸为2N×2N的分区类型1322、尺寸为2N×N的分区类型1324、尺寸为N×2N的分区类型1326、尺寸为N×N的分区类型1328、尺寸为2N×nU的分区类型1332、尺寸为2N×nD的分区类型1334、尺寸为nL×2N的分区类型1336以及尺寸为nR×2N的分区类型1338。

[0169] 当分区类型被设置成对称(即，分区类型1322、1324、1326或1328)时，如果变换单元的划分信息(TU尺寸标记)是0，则设置尺寸为2N×2N的变换单元1342，如果TU尺寸标记是1，则设置尺寸为N×N的变换单元1344。

[0170] 当分区类型被设置成非对称(例如，分区类型1332、1334、1336或1338)时，如果TU尺寸标记是0，则设置尺寸为2N×2N的变换单元1352，如果TU尺寸标记是1，则设置尺寸为N/2×N/2的变换单元1354。

[0171] TU尺寸标记是变换索引的类型；可根据编码单元的预测单元类型或分区类型来识

别与变换索引相应的变换单元的尺寸。

[0172] 当分区类型被设置成对称(即,分区类型1322、1324、1326或1328)时,如果变换单元的TU尺寸标记是0,则设置尺寸为2N×2N的变换单元1342,如果TU尺寸标记是1,则设置尺寸为N×N的变换单元1344。

[0173] 当分区类型被设置成非对称(例如,分区类型1332(2N×nU)、1334(2N×nD)、1336(nL×2N)或1338(nR×2N))时,如果TU尺寸标记是0,则设置尺寸为2N×2N的变换单元1352,如果TU尺寸标记是1,则设置尺寸为N/2×N/2的变换单元1354。

[0174] 参照图13,以上描述的TU尺寸标记是具有值0或1的标记,但是TU尺寸标记不限于1比特,在TU尺寸标记从0增加的同时,变换单元可被分层划分。变换单元划分信息(TU尺寸标记)可用作变换索引的示例。

[0175] 在此情况下,当使用根据实施例的TU尺寸标记以及变换单元的最大尺寸和最小尺寸时,可表示实际上使用的变换单元的尺寸。视频编码设备100可对最大变换单元尺寸信息、最小变换单元尺寸信息、和最大变换单元划分信息进行编码。可将编码的最大变换单元尺寸信息、最小变换单元尺寸信息、和最大变换单元划分信息插入到序列参数集(SPS)中。视频解码设备200可使用最大变换单元尺寸信息、最小变换单元尺寸信息、和最大变换单元划分信息来进行视频解码。

[0176] 例如,(a)如果当前编码单元的尺寸是64×64并且最大变换单元尺寸是32×32,则(a-1)如果TU尺寸标记为0,则变换单元的尺寸是32×32,(a-2)如果TU尺寸标记为1,则变换单元的尺寸是16×16,(a-3)如果TU尺寸标记为2,则变换单元的尺寸是8×8。

[0177] 可选地,(b)如果当前编码单元的尺寸是32×32并且最小变换单元的尺寸是32×32,则(b-1)如果TU尺寸标记为0,则变换单元的尺寸是32×32,并且由于变换单元的尺寸不能够小于32×32,因此不可设置更多的TU尺寸标记。

[0178] 可选地,(c)如果当前编码单元的尺寸是64×64并且最大TU尺寸标记为1,则TU尺寸标记可以是0或1,并且不可以设置其他的TU尺寸标记。

[0179] 因此,当定义最大TU尺寸标记为“MaxTransformSizeIndex”,最小TU尺寸标记为“MinTransformSize”,并且在TU尺寸标记为0的情况下的变换单元(即,基本变换单元RootTu)为“RootTuSize”时,可通过以下的等式(1)来定义在当前编码单元中可用的最小变换单元的尺寸“CurrMinTuSize”。CurrMinTuSize=max(MinTransformSize,RootTuSize/(2^MaxTransformSizeIndex))…(1)

[0180] 与在当前编码单元中可用的最小变换单元的尺寸“CurrMinTuSize”相比,作为当TU尺寸标记为0时的变换单元的尺寸的基本变换单元尺寸“RootTuSize”可指示对于系统可选择的最大变换单元。也就是说,根据等式(1),“RootTuSize/(2^MaxTransformSizeIndex)”是通过将作为当变换单元划分信息为0时的变换单元的尺寸的“RootTuSize”划分与最大变换单元划分信息相应的划分次数而获得的变换单元的尺寸,“MinTransformSize”是最小变换单元的尺寸,因此,“RootTuSize/(2^MaxTransformSizeIndex)”和“MinTransformSize”中较小的值可以是在当前编码单元中可用的最小变换单元的尺寸“CurrMinTuSize”。

[0181] 根据本发明的实施例的基本变换单元“RootTuSize”的尺寸可根据预测模式而不同。

[0182] 例如，如果当前预测模式是帧间模式，则可根据以下的等式(2)来确定“RootTuSize”。在等式(2)中，“MaxTransformSize”是指最大变换单元尺寸，“PUSize”是指当前预测单元尺寸。

[0183] RootTuSize＝min(MaxTransformSize,PUSize)……(2)

[0184] 换句话说，如果当前预测模式是帧间模式，则作为当TU尺寸标记为0时的变换单元的基本变换单元尺寸“RootTuSize”的大小可被设置为最大变换单元尺寸和当前预测单元尺寸中较小的值。

[0185] 如果当前分区单元的预测模式是帧内模式，则可根据以下的等式(3)来确定“RootTuSize”。“PartitionSize”是指当前分区单元的尺寸。

[0186] RootTuSize＝min(MaxTransformSize,PartitionSize)……(3)

[0187] 换句话说，如果当前预测模式是帧内模式，则基本变换单元尺寸“RootTuSize”可被设置为最大变换单元尺寸和当前分区单元尺寸之中较小的值。

[0188] 然而，应注意，基本变换单元尺寸“RootTuSize”的大小是示例，用于确定当前最大变换单元尺寸的因素不限于此，其中，基本变换单元尺寸“RootTuSize”的大小是根据本发明的实施例的当前最大变换单元尺寸，并根据分区单元的预测模式而不同。

[0189] 以下，将详细描述在图1的视频编码设备100的熵编码器120中执行的句法元素的熵编码操作，以及在图2的视频解码设备200的熵解码器220中执行的句法元素的熵解码操作。

[0190] 如以上所描述的，视频编码设备100和视频解码设备200通过将最大编码单元划分为小于或等于最大编码单元的编码单元来执行编码和解码。可独立于其他数据单元基于损耗来确定在预测和变换中使用的预测单元和变换单元。由于可通过对包括在最大编码单元中的具有分层结构的每个编码单元进行迭代地编码来确定最佳编码单元，因此可构造具有树结构的数据单元。换句话说，对于每个最大编码单元，可构造具有树结构的编码单元、以及均具有树结构的预测单元和变换单元。对于解码，分层信息和除了分层信息之外的用于解码的非分层信息需要被发送，其中，分层信息是指示具有分层结构的数据单元的结构信息。

[0191] 如以上参照图10至图12所描述，与分层结构有关的信息是确定具有树结构的编码单元、具有树结构的预测单元和具有树结构的变换单元所需的信息，并且可包括指示最大编码单元的尺寸的变换单元划分标记(TU尺寸标记)、编码深度、预测单元的分区信息、指示编码单元是否被划分的划分标记、变换单元的尺寸信息、以及指示变换单元是否被划分的变换单元划分标记(TU尺寸标记)。除了分层结构信息之外的编码信息的示例包括被应用于每个预测单元的帧内/帧间预测的预测模式信息、运动矢量信息、预测方向信息、在使用多个颜色分量的情况下被应用于每个数据单元的颜色分量信息、以及变换系数信息。以下，分层信息和额外的分层信息可被称为将被熵编码或熵解码的句法元素。

[0192] 具体地，根据本发明的实施例，提供确定用于对变换系数的级别(即，句法元素的大小信息)进行高效地熵编码和熵解码的上下文模型的方法。以下，将详细描述确定用于对变换系数的级别进行熵编码和熵解码的上下文模型的方法。

[0193] 图14是示出根据本发明的实施例的对包括在变换单元中的变换系数信息进行熵编码和熵解码的操作的流程图。

[0194] 参照图14，在操作1410，首先对coded_block_flag进行熵编码或熵解码，其中，coded_block_flag指示在当前变换单元中所包括的变换系数之中是否存在不为0的变换系数(以下，被称为“有效系数”)。

[0195] 如果coded_block_flag为0，则在当前变换单元中仅存在0的变换系数，并因此仅将值0熵编码或熵解码为coded_block_flag，并且不对变换系数级别信息进行熵编码或熵解码。

[0196] 在操作1420，如果在当前变换单元中存在有效系数，则对指示有效系数的位置的有效图SigMap进行熵编码或熵解码。

[0197] 有效图SigMap可由有效位和指示最后一个有效系数的预定信息构成。有效位指示根据每个扫描索引的变换系数是有效系数还是0，并可由significant_coeff_flag[i]来表示。如稍后将描述的，以通过划分变换单元而获得的具有预定尺寸的子集为单位来设置有效图。因此，significant_coeff_flag[i]指示在子集中包括的变换系数之中的第i个扫描索引的变换系数是否为0，其中，所述子集包括在变换单元中。

[0198] 根据传统的H.264，指示每个有效系数是否为最后一个有效系数的标记(End-of-Block(块的结束))被另外地熵编码或熵解码。然而，根据本发明的实施例，对最后一个有效系数自身的位置信息进行熵编码或熵解码。如以上参照图1至图13所描述的，根据本发明的实施例的变换单元的尺寸不限于4×4，而还可以是更大的尺寸，诸如8×8、16×16或32×32。由于指示每个有效系数是否是最后一个有效系数的标记(End-Of-Block)的大小增加，因此另外地对标记(End-Of-Block)进行熵编码或熵解码的效率低。因此，根据本发明的实施例，可对最后一个有效系数自身的位置信息进行熵编码或熵解码。例如，如果最后一个有效系数的位置是(x,y)，则可对坐标值(x,y)进行熵编码或熵解码，其中，x和y是整数。

[0199] 图15示出根据本发明的实施例的通过划分变换单元而获得的子集。虽然图15中示出了变换单元1500的16×16尺寸，但是变换单元1500的尺寸不限于16×16，还可以从4×4到32×32。

[0200] 参照图15，为了对在变换单元1500中包括的变换系数进行熵编码和熵解码，将变换单元1500划分为具有预定尺寸的子集。例如，如图15中所示，可将变换单元1500划分为尺寸为4×4的子集。子集的尺寸不限于4×4，可以是各种尺寸。

[0201] 如图15中所示，将变换单元1500划分为16个子集，针对每个子集单元对变换系数信息进行熵编码或熵解码。被熵编码或熵解码的变换系数信息可以是例如第一临界值标记(Greaterthan1标记，以下被称为“Gtr1标记”)、第二临界值标记(Greaterthan2标记，以下被称为“Gtr2标记”)、或level-3，其中，Gtr1标记指示在子集中包括的有效系数是否具有比预定的第一临界值(诸如1)更大的值，Gtr2标记指示在子集中包括的有效系数是否具有比预定的第二临界值(诸如2)更大的值，level-3指示比预定的第二临界值大的有效系数的级别。这里，假设第一临界值被设置为1，第二临界值被设置为2，但是第一临界值和第二临界值不限于此，并可被修改。仅针对包括1的有效图的变换系数(即，仅针对有效系数)设置第一临界值标记(Gtr1标记)，而不针对0的变换系数设置第一临界值标记(Gtr1标记)。另外，仅针对第一临界值标记为1的变换系数设置第二临界值标记(Gtr2标记)。

[0202] 在图15中，假设包括最后一个有效系数的子集为子集111510。从包括有最后一个有效系数的子集111510开始，根据扫描顺序逆向地执行针对每个子集执行的对变换系数信

息进行的熵编码和熵解码。

[0203] 图16示出根据本发明的实施例的在图15的变换单元1500中包括的子集1600。

[0204] 参照图15和图16，假设在图15示出的变换单元1500中包括的任意子集包括如图16所示的尺寸为4×4的变换系数。如以上所描述的，根据本发明的实施例，以子集为单位对有效图SigMap、第一临界值标记Gtr1标记和第二临界值标记Gtr2进行熵编码或熵解码。

[0205] 图17示出与图16的子集1600相应的有效图SigMap1700。

[0206] 参照图16和图17，设置针对在图16的子集1600中包括的值不为0的变换系数之中的有效系数的具有值1的有效图SigMap1700。使用预定上下文模型对有效图SigMap1700进行熵编码或熵解码。

[0207] 图18示出与图16的子集1600相应的第一临界值标记Gtr1标记1800。

[0208] 参照图16至图18，设置针对值为1的变换系数的指示相应变换系数是否具有比第一临界值(即，1)更大的值的第一临界值标记Gtr1标记1800。如果临界值标记Gtr1标记1800是1，则这表示相应变换系数具有比1更大的值，如果临界值标记Gtr1标记1800是0，则这表示相应系数的值为1。

[0209] 图19示出与图16的子集1600相应的第二临界值标记Gtr2标记1900。

[0210] 参照图16至图19，仅针对在变换系数之中的第一临界值标记Gtr1标记1800被设置为1的变换系数设置第二临界值标记Gtr2标记1900，第二临界值标记Gtr2标记1900指示相应变换系数是否具有比第二临界值(即，2)更大的值。

[0211] 参照图19，对第二临界值标记Gtr2标记1900为1的变换系数(即，图16中的值为23和3的变换系数)的级别信息自身进行熵编码或熵解码。这里，包括1的第二临界值标记Gtr2标记1900的变换系数是具有比2更大的值的变换系数，因此，将通过从相应变换系数的级别减去3而获得的值(level-3)编码为相应变换系数的级别信息。在以上描述的示例中，当执行熵编码时，对20进行编码，而不对23进行编码，并且0被编码为级别信息，而3不被编码为级别信息。当执行熵解码时，通过对包括1的第二临界值标记Gtr2标记1900的变换系数的level-3进行熵解码并然后将3添加到level-3，来恢复变换系数的级别信息。

[0212] 图20是示出在图16至图19中示出的子集中包括的变换系数以及被熵编码或熵解码的变换系数信息的表。如以上所描述的，根据本发明的实施例，以子集为单位从最后一个有效系数开始，根据预定扫描顺序对指示有效系数的位置和级别信息的有效图SigMap、第一临界值标记Gtr1标记、第二临界值标记Gtr2标记和级别信息(level-3)进行熵编码或熵解码。

[0213] 图21a是示出根据本发明的实施例的熵编码设备2100的结构的结构框图。图21a的熵编码设备2100与图1的视频编码设备100的熵编码器120相应。

[0214] 参照图21a，熵编码设备2100包括二值化器2110、上下文建模器2120和二进制算术编码器2130。此外，二进制算数编码器2130包括常规编码引擎2132和旁路编码引擎2134。

[0215] 输入到熵编码设备2100的句法元素可以不是二进制值，因此，如果句法元素不是二进制值，则二值化器2110对句法元素进行二值化以输出由二进制值0或1构成的二进制(bin)串。bin表示流的每个比特由0或1构成，并通过自适应二进制算术编码(CABAC)来编码。如果句法元素是以相同频率包括0和1的数据，则句法元素被输出到不使用概率值的旁路编码引擎2134。

[0216] 上下文建模器2120将关于当前编码符号的概率模型提供给常规编码引擎2132。详细地讲，上下文建模器2120将用于对当前编码符号的二进制值进行编码的二进制值的概率输出到二进制算术编码器2130。当编码的当前句法元素通过二值化形成时(即，当编码的当前句法元素由二进制值形成时)，当前编码符号是指每个二进制值。

[0217] 上下文模型是关于bin的概率模型，可包括指示0和1中的哪个与最大概率符号(MPS)和最小概率符号(LPS)相应的信息、以及MPS或LPS的概率。以下，上下文模型可被简单地称为上下文。此外，上下文集是指包括多个上下文的集。

[0218] 常规编码引擎1432基于从上下文建模器1420提供的关于MPS或LPS的信息以及MPS或LPS的概率信息，对当前编码符号执行二进制算术编码。

[0219] 如稍后将描述的，为了确定用于对子集的第一临界值标记Gtr1标记进行熵编码的上下文模型，根据本发明的实施例的上下文建模器2120基于变换单元的颜色分量信息、指示当前子集位于变换单元中的哪个位置的子集的位置信息、以及在根据以上针对15描述的处理顺序在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来获得上下文集索引ctxset，其中，上下文集索引ctxset用于在包括多个上下文的多个上下文集之中确定在对第一临界值标记进行熵编码中使用的上下文集。另外，上下文建模器2120基于具有连续的1的先前变换系数的长度来获得上下文偏移c1，其中，上下文偏移c1用于确定在对第一临界值标记Gtr1标记进行熵编码中使用的上下文集中包括的多个上下文中的一个上下文。此外，上下文建模器2120通过使用上下文集索引ctxset和上下文偏移c1来获得指示在对第一临界值标记Gtr1标记进行熵编码中使用的上下文的上下文索引ctxIdx1。当经常对变换单元的第一临界值标记Gtr1标记进行熵编码或熵解码时，上下文偏移c1的值被保持或修改，因此，每当对变换单元的第一临界值标记Gtr1标记进行熵编码或熵解码时，总要保持或更新上下文。

[0220] 此外，上下文建模器2120基于变换单元的颜色分量信息、指示当前子集位于变换单元中的哪个位置的子集的位置信息、以及在根据以上针对图15描述的处理顺序在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来获得上下文集索引ctxset，其中，上下文集索引ctxset指示在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的多个上下文集中的一个上下文集。用于获得在对第二临界值标志Gtr2标志进行熵编码和熵解码中使用的上下文集索引ctsxet的参数与用于获得在对第一临界值标记Gtr1标记进行熵编码和熵解码中使用的上下文集索引ctxset的参数相同。因此，当确定用于对第二临界值标记Gtr2标记进行熵编码的上下文集时，上下文建模器2120也可使用以上所描述的用于确定在对第一临界值标记进行熵编码中使用的上下文集的上下文集索引ctxset。用于确定在对第二临界值标记Gtr2标记进行熵编码中使用的上下文集中所包括的多个上下文之一的上下文偏移c2具有值0。因此，将指示在对第二临界值标记Gtr2标记进行熵编码中使用的上下文的上下文索引ctxIdx2设置为与第一临界值标记Gtr1标记的上下文集索引ctxset相同。

[0221] 图21b是示出根据本发明的实施例的熵解码设备2150的结构的结构框图。图21b的熵解码设备2150与图2的视频解码设备200的熵解码器220相应。熵解码设备2150执行在以上描述的熵编码设备2100中执行的熵编码的逆操作。

[0222] 参照图21b，熵解码设备2150包括上下文建模器2160、常规解码引擎2170、旁路解

码器2180和去二值化器2190。

[0223] 通过旁路编码进行编码的符号被输出到旁路解码器2180以进行解码，通过常规编码的符号由常规解码引擎2170进行解码。常规解码引擎2170基于由上下文建模器2160提供的上下文模型对当前编码符号的二进制值进行算术解码。

[0224] 上下文建模器2160以与以上描述的图21a的上下文建模器2120相同的方式来确定用于对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵解码的上下文模型。图21b的上下文建模器2160以与以上针对编码所描述的图21a的上下文建模器2120相同的方式来确定用于对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵解码的上下文模型，不同之处在于上下文建模器2160针对解码进行操作。

[0225] 去二值化器2340将在常规解码引擎2170或旁路解码器2180中恢复的二进制串恢复到句法元素。

[0226] 以下，将详细描述通过使用图21a和图21b的上下文建模器2120和2160而实施的确定用于对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码的上下文模型的操作。

[0227] 图22是示出根据本发明的实施例的上下文建模器2200的结构框图。

[0228] 参照图22，上下文建模器2200包括映射单元2210、上下文集获得单元2220、上下文偏移获得单元2230、和上下文确定单元2240。

[0229] 映射单元2210获得在被熵编码或熵解码的当前子集中包括的有效系数的位置信息。当执行熵编码时，映射单元2210可从在当前子集中包括的变换系数的信息获得有效系数的位置。当执行熵解码时，映射单元2210可从有效图SigMap获得在子集中包括的有效系数的位置。

[0230] 上下文集获得单元2220获得指示包括多个上下文的多个上下文集中的一个上下文集的上下文集索引ctxset，其中，所述多个上下文被用于对关于有效系数的第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码。

[0231] 详细地讲，上下文集获得单元2220基于变换单元的颜色分量信息、正被处理的当前子集的位置信息、在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来获得上下文集索引，其中，上下文索引用于确定在对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文模型。

[0232] 上下文偏移获得单元2230确定指示在上下文集中包括的多个上下文中的一个上下文的上下文偏移。当对第一临界值标记Gtr1标记执行熵编码或熵解码时，可基于在当前子集中包括的有效系数根据预定扫描顺序被处理时在被处理的当前有效系数之前的具有连续的1的先前变换系数的长度来确定上下文偏移c1。第二临界值标记Gtr2标记的上下文偏移c2的值为0。

[0233] 上下文确定单元2240通过使用上下文集索引ctxset和上下文偏移c1来获得上下文索引ctxIdx1，其中，上下文索引ctxIdx1指示在对第一临界值标记Gtr1标记进行熵编码或熵解码中使用的上下文。为了对第一临界值标记Gtr1标记进行熵编码和熵解码，当假设设置n个上下文集(n是整数)，并且n个上下文集具有m个上下文(m是整数)时，在对第一临界值标记Gtr1标记进行熵编码或熵解码中可使用总共n×m个上下文。当假设指示n个上下文集中的一个上下文集的上下文集索引ctxSet是从0到(n-1)的整数，并且指示m个上下文偏

移中的一个上下文偏移的上下文偏移c1是从0到(m-1)的整数时，可基于以下等式确定指示n×m个上下文中的一个上下文的上下文索引ctxIdx1：ctxIdx1＝ctxSet×m+c1。

[0234] 可基于以下等式确定指示在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的一个上下文的上下文索引ctxIdx2：ctxIdx2＝ctxSet×1+c2。如以上所描述的，由于c2是0，因此仅基于集索引ctxSet的值来确定指示在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文的上下文索引ctxIdx2。

[0235] 图23示出根据本发明的实施例的被应用于亮度分量的变换单元的多个上下文集以及在每个上下文集中包括的多个上下文。图24示出根据本发明的实施例的被应用于色度分量的变换单元的多个上下文集以及在每个上下文集中包括的多个上下文。

[0236] 参照图23，为了针对在亮度分量子集中包括的变换系数对第一临界值标记Gtr1标记进行熵编码和熵解码，使用在第一上下文集2310、第二上下文集2320、第三上下文集2330和第四上下文集2340中的任意一个中包括的上下文。如图23中所示，通过从0到3的上下文偏移c1来区分在四个上下文集中的任意一个中包括的不同上下文。因此，可通过使用上下文集索引ctxset和上下文偏移c1来确定在将被应用于亮度分量的子集的变换单元的第一临界值标记Gtr1标记的总共16个上下文之中的一个上下文。也就是说，可通过以下等式确定上下文索引ctxIdx1：ctxIdx1＝ctxSet×4+c1，其中，所述上下文索引ctxIdx1指示将被应用于亮度分量的子集的变换单元的第一临界值标记Gtr1标记的总共16个上下文中的一个上下文。

[0237] 类似地，当参照图24时，为了针对在色度分量子集中包括的变换单元对第一临界值标记Gtr1标记进行熵编码和熵解码，使用在来自总共两个上下文集(第一上下文集2410和第二上下文集2420)的上下文集中的任意一个上下文集中包括的上下文。如图24中所示，通过从0到3的上下文偏移c1_chroma来区分在上下文集中包括的不同上下文。如将在稍后描述的，可以以与针对亮度分量的上下文偏移c1相同的方式来设置关于色度分量的上下文偏移c1_chroma。通过以下等式来确定上下文索引ctxIdx1_chroma：ctxIdx1_chroma＝ctxSet×4+c1_chroma，其中，上下文索引ctxIdx1_chroma指示将被应用于在色度分量的子集中包括的变换系数的第一临界值Gtr1标记的总共8个上下文中的一个上下文。

[0238] 图27a示出根据本发明的实施例的用于确定上下文集的上下文集索引ctxset，其中，在对亮度分量的有效系数和色度分量的有效系数的第一临界值标记Gtr1标记和第二临界值标记Ctr2标记进行熵编码和熵解码中使用所述上下文集。

[0239] 参照图27a，上下文集获得单元2220基于变换单元的颜色分量信息、正被处理的当前子集的位置信息、以及在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来获得上下文集索引ctxset，其中，上下文集索引ctxset用于确定在对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文集。

[0240] 例如，关于在亮度分量的变换单元中包括的变换系数，如果在先前处理的子集中不存在具有比1更大的值的有效系数(没有GreatT1)，并且当对在位于最左上方位置的子集中包括的有效系数的第一临界值标记进行熵编码或熵解码时，则获得第一上下文集(ctset＝0)。如果在先前处理的子集中存在具有比1更大的值的有效系数(至少一个GreatT1)，并且当对在位于最左上方位置的子集(子集0)中包括的有效系数的第一临界值标记进行熵编

码或熵解码时，则获得第二上下文集(ctxset＝1)。如果在先前处理的子集中不存在具有比1更大的值的有效系数(没有GtreatT1)，并且当对在不位于最左上方位置的子集(其他子集)中包括的有效系数的第一临界值标记进行熵编码或熵解码时，则获得第三上下文集(ctxset＝2)。此外，如果在先前处理的子集中存在具有比1更大的值的有效系数(至少一个GreatT1)，并且当对在不位于最左上方位置的子集(其他子集)中包括的有效系数的第一临界值标记进行熵编码或熵解码时，则获得第四上下文集(ctxset＝3)。

[0241] 关于在色度分量的变换单元中包括的变换系数，仅基于在先前处理的子集中是否存在具有比1更大的值的有效系数，来获得上下文集。换句话说，如果在先前处理的子集中不存在具有比1更大的值的有效系数(没有GreatT1)，则获得第一上下文集(ctxset＝0)，如果在先前处理的子集中存在具有比1更大的值的有效系数(至少一个GreatT1)，则获得第二上下文集(ctxset＝1)。

[0242] 图27b示出根据本发明的实施例的在对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文偏移。

[0243] 参照图27b，为了确定在对第一临界值标记Gtr1标记进行熵编码和熵解码中使用的上下文集中包括的上下文，针对在子集中包括的有效系数之中具有比1更大的值的有效系数，获得第一上下文偏移(c1＝0)。关于在子集中包括的有效系数之中被第一次处理的有效系数，获得第二上下文偏移(c1＝1)。如果具有连续的1的先前变换系数的长度为1，则获得第三上下文偏移(c1＝2)。如果具有连续的1的先前变换系数的长度为2或大于2，则获得第四上下文偏移(c1＝3)。可将上下文偏移c1应用于亮度分量和色度分量。

[0244] 在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文偏移c2的值为0。

[0245] 当基于图27a和图27b的表来确定用于对在子集中包括的变换系数进行熵编码或熵解码的上下文集索引ctxset和上下文偏移c1或c2时，上下文确定单元2240根据以下等式确定上下文索引ctxIdx1:ctxIdx1＝ctxSet×4+c1，其中，上下文索引ctxIdx1指示将被应用于在亮度分量的子集中包括的变换单元的第一临界值标记Gtr1标记的总共16个上下文中的一个上下文。此外，上下文确定单元2240根据以下等式确定上下文索引ctxIdx1_chroma:ctxIdx1_chroma＝ctxSet×4+c1，其中，上下文索引ctxIdx1_chroma指示将被应用于在色度分量的子集中包括的变换单元的第一临界值标记Gtr1标记的总共8个上下文中的一个上下文。参照图27a和图27b的表，被应用于第一临界值标记Gtr1标记的上下文的总数是24(即，用于亮度分量的4×4＝16个加上用于色度分量的2×4＝8个)。

[0246] 此外，上下文确定单元2240根据以下等式确定上下文索引ctxIdx2:ctxIdx2＝ctxset×1+c2，其中，上下文索引ctxIdx2指示将被应用于第二临界值标记Gtr2标记的上下文。也就是说，将指示将被应用于第二临界值标记Gtr2标记的上下文的上下文索引ctxIdx2设置为与上下文集索引ctxset的值相同。因此，参照图27a和图27b的表，被应用于第二临界值标记Gtr2标记的上下文的总数为8(即，用于亮度分量的4个加上用于色度分量的4个)。

[0247] 图25是示出根据本发明的实施例的确定用于对变换系数级别进行熵编码和熵解码的上下文模型的方法的流程图。

[0248] 参照图25，在操作2510，映射单元2210将变换单元划分为具有预定单元的子集，并获得在每个子集中包括的不为0的有效系数。如以上所描述的，当执行熵编码时，映射单元

2210可从在当前子集中包括的变换系数的信息获得有效系数的位置。当执行熵解码时，映射单元2210可从有效图SigMap获得在子集中包括的有效系数的位置。

[0249] 在操作2520，上下文集获得单元2220基于变换单元的颜色分量信息、包括有效系数且当前被处理的第一子集的位置、以及在在第一子集之前被处理的第二子集中是否存在比预定的第一临界值更大的有效系数，来获得上下文集索引ctxset，其中，上下文集索引ctxset用于在包括多个上下文的多个上下文集之中确定在对第一临界值标记进行熵编码和熵解码中使用的上下文集，其中，第一临界值标记指示有效系数是否具有比第一临界值更大的值。如图27a和图27b中所示，关于在亮度分量的变换单元中包括的变换系数，上下文集获得单元2220可依据第一子集的位置是否是位于最左上方位置的子集0、以及在先前处理的子集中是否存在具有比1更大的值的有效系数，来获得指示在四个上下文集中的一个上下文集的上下文集ctxset。此外，针对在色度分量的变换单元中包括的变换系数，上下文集获得单元2220可仅基于在先前处理的子集中是否存在比1更大的值的有效系数，来获得指示两个上下文集中的一个上下文集的上下文集ctxset。

[0250] 在操作2530，上下文偏移获得单元2230基于具有连续的1的先前变换系数的长度获得上下文偏移，其中，所述上下文偏移用于确定在对第一临界值标记Gtr1标记进行熵编码或熵解码中使用的上下文集中包括的多个上下文中的一个上下文。如以上所描述的，上下文偏移获得单元2230可基于在包括在当前子集中的有效系数根据预定扫描顺序正被处理时在被处理的当前有效系数之前的先前变换系数的连续的1的长度，来确定上下文偏移c1。上下文偏移获得单元2230可将在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文偏移c2设置为总是具有值0，而无需考虑其它参数。

[0251] 在操作2540，上下文确定单元2240通过使用上下文集索引ctxset和上下文偏移c1来获得指示在对第一临界值标记Gtr1标记进行熵编码和熵解码中使用的上下文的上下文索引ctxIdx1。如以上所描述的，当假设指示n个上下文集中的一个上下文集的上下文集索引ctxSet是从0到(n-1)的整数，并且指示m个上下文偏移中的一个上下文偏移的上下文偏移c1是从0到(m-1)的整数时，上下文确定单元2240可根据以下等式确定指示n×m个上下文中的一个上下文的上下文索引ctxIdx1：ctxIdx1＝ctxSet×m+c1。此外，上下文确定单元2240可根据以下等式确定指示在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的一个上下文的上下文索引ctxIdx2：ctxIdx2＝ctxSet×1+c2。由于c2是0，因此仅基于集索引ctxSet的值来确定指示在对第二临界值标记Gtr2标记进行熵编码和熵解码中使用的上下文的上下文索引ctxIdx2。

[0252] 图26是示出根据本发明的实施例的确定用于对变换系数级别进行熵编码和熵解码的上下文模型的方法的详细流程图。

[0253] 参照图26，在操作2611，映射单元2210将变换单元划分为具有预定尺寸的子集，并获得在每个子集中包括的不为0的有效系数。

[0254] 在操作2612，上下文集获得单元2220基于在确定上下文集中使用的参数之中的当前子集的位置和颜色分量信息来确定上下文集索引ctxset。例如，当当前子集是位于变换单元最左上方的子集0或色度分量时，ctxset被设置为0，如果当前子集不是位于变换单元的最左上方位置的子集并且是亮度分量，则ctxset被设置为2。

[0255] 在操作2613，上下文集获得单元2220确定在刚好在当前子集之前被处理的先前子

集中是否存在具有比第一临界值更大的值的有效系数。作为在操作2613确定的结果，如果在先前子集中存在具有比第一临界值更大的值的有效系数，则上下文集获得单元2220使在操作2612设置的上下文集索引ctxset的值增加1；否则，如果在先前子集中不存在具有比第一临界值更大的值的有效系数，则上下文集获得单元2220保持在操作2612设置的上下文集索引ctxset的值。

[0256] 在操作2615，上下文偏移获得单元2230设置上下文偏移c1的值，其中，上下文偏移c1将被应用于首次被处理的当前子集的有效系数的第一临界值标记Gtr1标记。

[0257] 在操作2616，上下文确定单元2240根据以下等式获得指示将被应用于第一临界值标记Gtr1标记的上下文的上下文索引ctxIdx1：ctxIdx1＝ctxset×4+min(c1,3)。所述等式基于图27a和图27b，并在操作2620确定具有连续的1的变换系数的长度，如果至少三个第一临界值标记Gtr1标记连续，则上下文偏移索引c1可具有等于3或大于3的值。然而，参照图27b，由于上下文偏移索引c1针对至少两个连续的1被设置为具有值2，因此min(c1,3)被用于将上下文偏移索引c1限制为不大于3。

[0258] 在操作2617，常规编码引擎2132和常规解码引擎2170基于由获得的上下文索引ctxIdx1指示的上下文模型对第一临界值标记Gtr1标记进行熵编码或熵解码。

[0259] 在操作2618，上下文偏移获得单元2230确定当前被熵编码或熵解码的第一临界值标记Gtr1标记是否具有值0，以及上下文偏移索引c1是否为0。执行操作2618的确定操作，以确定有效系数之中连续的1的数量。作为在操作2618的确定的结果，如果当前编码或解码的第一临界值标记Gtr1标记具有值0，且上下文偏移索引c1不为0，则在操作2620将上下文偏移索引c1增加1。否则，作为在操作2618的确定的结果，如果当前编码或解码的第一临界值标记Gtr1标记不具有值0，或上下文偏移索引c1为0，则在操作2619将上下文偏移索引c1重置为0。

[0260] 操作2615至操作2620是对在子集中包括的变换系数的第一临界值标记Gtr1标记进行熵编码或熵解码的操作。为了加快操作，不对关于每个有效系数的第一临界值标记Gtr1标记进行熵编码或熵解码，而仅针对从最后一个有效系数(最大循环#)开始的预定数量(#)的有效数据，对第一临界值标记Gtr1标记进行熵编码或熵解码。对相应的第一临界值标记Gtr1标记未被熵编码或熵解码的有效系数的级别信息自身进行熵编码或熵解码。

[0261] 在操作2621至操作2624，对第二临界值标记Gtr2标记进行熵编码或熵解码。

[0262] 在操作2621，确定上下文偏移索引c1的值是否为0。在操作2622，将指示将被应用于对第二临界值标记Gtr2标记进行熵编码或熵解码的上下文的上下文索引ctxIdx2设置为与当对第一临界值标记Gtr1标记执行熵编码和熵解码时被确定的上下文集索引ctxset相同。

[0263] 在操作2623，常规编码引擎2132和常规解码引擎2170基于由获得的上下文索引ctxIdx2指示的上下文模型对第二临界值标记Gtr2标记进行熵编码或熵解码。

[0264] 在操作2624，常规编码引擎2132和常规解码引擎2170对具有比第二临界值更大的值的变换系数的级别进行熵编码或熵解码。如以上描述的，可对通过从相应的变换系数减去预定值而获得的级别值(level-3)进行熵编码或熵解码。

[0265] 图28示出根据本发明的实施例的包括在图20的子集中所包括的变换系数以及在对被熵编码或熵解码的变换系数信息进行熵编码或熵解码中使用的上下文偏移索引c1的

表。如以上所描述的，基于具有连续值1的变换系数的长度来确定上下文偏移索引c1，并假设根据图27b获得上下文偏移索引c1。此外，在图28中，假设处理每个变换系数的顺序是从右到左。

[0266] 此外，这里，假设不基于关于包括在当前子集中的变换系数的信息，而基于变换单元的颜色分量信息、当前子集的位置、以及在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来确定上下文集索引ctxset，其中，上下文集索引ctxset用于对第一临界值标记Gtr1标记和第二临界值标记Gtr2标记进行熵编码和熵解码。

[0267] 参照图28，被初步处理的子集的最后一个有效系数(2810)1不大于1，并因此第一临界值标记Gtr1标记具有值0。参照图27b，针对被初步处理的有效系数的第一临界值标记Gtr1标记的上下文偏移c1被设置为1。

[0268] 然后，关于最后一个有效系数(2810)的下一个有效系数的第一临界值标记Gtr1标记的上下文偏移c1具有值2。这是因为在先前存在具有值1的有效系数(2810)。类似地，关于下一个有效系数2830的第一临界值标记Gtr1标记的上下文偏移c1具有值3。这是因为在先前存在具有两个连续的1的两个有效系数(2810和2820)。

[0269] 由于有效变换系数(2830)的第一临界值标记Gtr1标记具有值1，因此关于在有效系数(2830)之后的所有有效系数的第一临界值标记Gtr1标记的上下文偏移c1具有值0。这是因为，如果作为图26的操作2618的确定的结果，第一临界值标记Gtr1标记具有值1，则上下文偏移c1根据操作2619被设置为0，并因此关于所有有效系数的上下文偏移c1此后都被设置为0。

[0270] 图29示出根据本发明的实施例的包括在子集中所包括的变换系数以及在对被熵编码或熵解码的变换系数信息进行熵编码或熵解码中使用的上下文偏移索引c1的表。比较图28和图29，图29与图28的不同之处在于最后一个有效系数(2910)具有值2，并因此第一临界值标记Gtr1标记具有值1。如以上描述的，关于被初步处理的有效系数2910的第一临界值Gtr1标记的上下文偏移c1被设置为1，并且关于在有效系数(2910)之后的所有有效系数的第一临界值标记Gtr1的上下文偏移c1具有值0。

[0271] 根据以上描述的用于确定上下文模型的方法和设备，基于变换单元的颜色分量信息、当前子集的位置、以及在当前子集之前被处理的子集中是否存在具有比第一临界值更大的值的有效系数，来获得上下文集索引ctxset，并基于具有连续的1的先前变换系数的长度来获得上下文偏移c1。

[0272] 用于确定上下文索引或上下文偏移的标准不限于本发明的实施例，并且还可被修改。

[0273] 例如，如表2中所示，可通过对概率相似的情况进行分组来设置上下文偏移c1。

[0274] [表2]

[0275]

上下文偏移（c1）	
0	1 个或更多个大于 1
1	初始-没有拖尾 1
2	1 个或 2 个拖尾 1
3	3 个或更多个拖尾 1

[0276] 当比较表2和图27b时，表2示出被分到将被映射在一个上下文中的同一个组中的一个连续变换系数和两个连续变换系数。此外，可基于在先前子集中包括的等于或大于1的有效系数的数量来对上下文集进行分组，以便将上下文集索引ctxset设置为如表3中所示。

[0277] [表3]

[0278]

上下文集索引（ctxset）（针对先前子集）	
0	初始-不大于 1
1	1 个或 2 个大于 1（1 个或 2 个有效系数）
2	3 个或更多个大于 1（3 个或更多个有效系数）

[0279] 另外，可基于包括当前变换单元的条带的类型、当前子集的位置、以及在先前子集中具有等于或大于预定临界值的值的变换系数的数量，将上下文索引ctxset设置为如以下的表4至表6中所示。

[0280] [表4]

[0281]

上下文集（ctxset）/在条带 I 的情况下		
0	子集 0	在先前子集中有 0 个大于 T1
1		在先前子集中有 1-4 个大于 T1
2		在先前子集中有>4 个大于 T1
3	其他子集	在先前子集中有 0-1 个大于 T1
4		在先前子集中有 1-4 个大于 T1
5		在先前子集中有>4 个大于 T1

[0282] 参照表4，例如，当对包括在条带I中所包括的变换单元的子集中的变换系数进行熵编码或熵解码时，可基于当前子集的位置、在先前子集中具有等于或大于预定临界值T1的值的变换系数的数量是0、1至4还是大于4，将0至5中的一个设置为上下文集索引ctxset的值。

[0283] [表5]

[0284]

上下文集（ctxset）/在条带 P 的情况下		
0	子集 0	在先前子集中有 0 个大于 T1
1		在先前子集中有 1-3 个大于 T1
2		在先前子集中有>3 个大于 T1
3	其他子集	在先前子集中有 0-1 个大于 T1
4		在先前子集中有 1-3 个大于 T1
5		在先前子集中有>3 个大于 T1

[0285] 参照表5，例如，当对包括在条带P中所包括的变换单元的子集中的变换系数进行熵编码或熵解码时，可基于当前子集的位置、在先前子集中具有等于或大于预定临界值T1的值的变换系数的数量是0、1至3还是大于3，将0至5中的一个设置为上下文集索引ctxset的值。

[0286] [表6]

[0287]

上下文集（ctxset）/在条带 B 的情况下		
0	子集 0	在先前子集中有 0 个大于 T1
1		在先前子集中有 1-2 个大于 T1
2		在先前子集中有>2 个大于 T1
3	其他子集	在先前子集中有 0 个大于 T1
4		在先前子集中有 1-2 个大于 T1
5		在先前子集中有>2 个大于 T1

[0288] 参照表6，例如，当对包括在条带B中所包括的变换单元的子集中的变换系数进行熵编码或熵解码时，可基于当前子集的位置、在先前子集中具有等于或大于预定临界值T1的值的变换系数的数量是0、1或2还是大于2，将0至5中的一个设置为上下文集索引ctxset的值。

[0289] 本发明的实施例可被编写为计算机程序，并可被实现在使用计算机可读记录介质执行程序的通用数字计算机中。计算机可读记录介质的示例包括磁存储介质(例如，ROM、软盘、硬盘等)和光记录介质(例如，CD-ROM或DVD)。

[0290] 虽然已经参照本发明的示例性实施例具体示出并描述了本发明，但是本领域普通技术人员将理解，在不脱离由权利要求限定的本发明的精神和范围的情况下，可做出形式和细节上的各种改变。

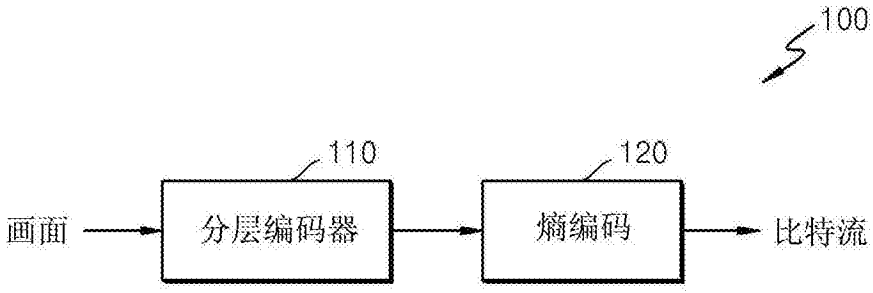

100
110
120
画面
分层编码器
熵编码
比特流

图1

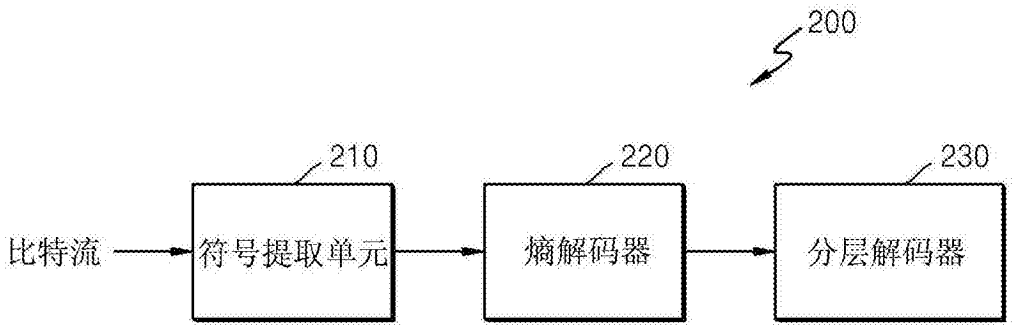

200
210
220
230
比特流
符号提取单元
熵解码器
分层解码器

图2

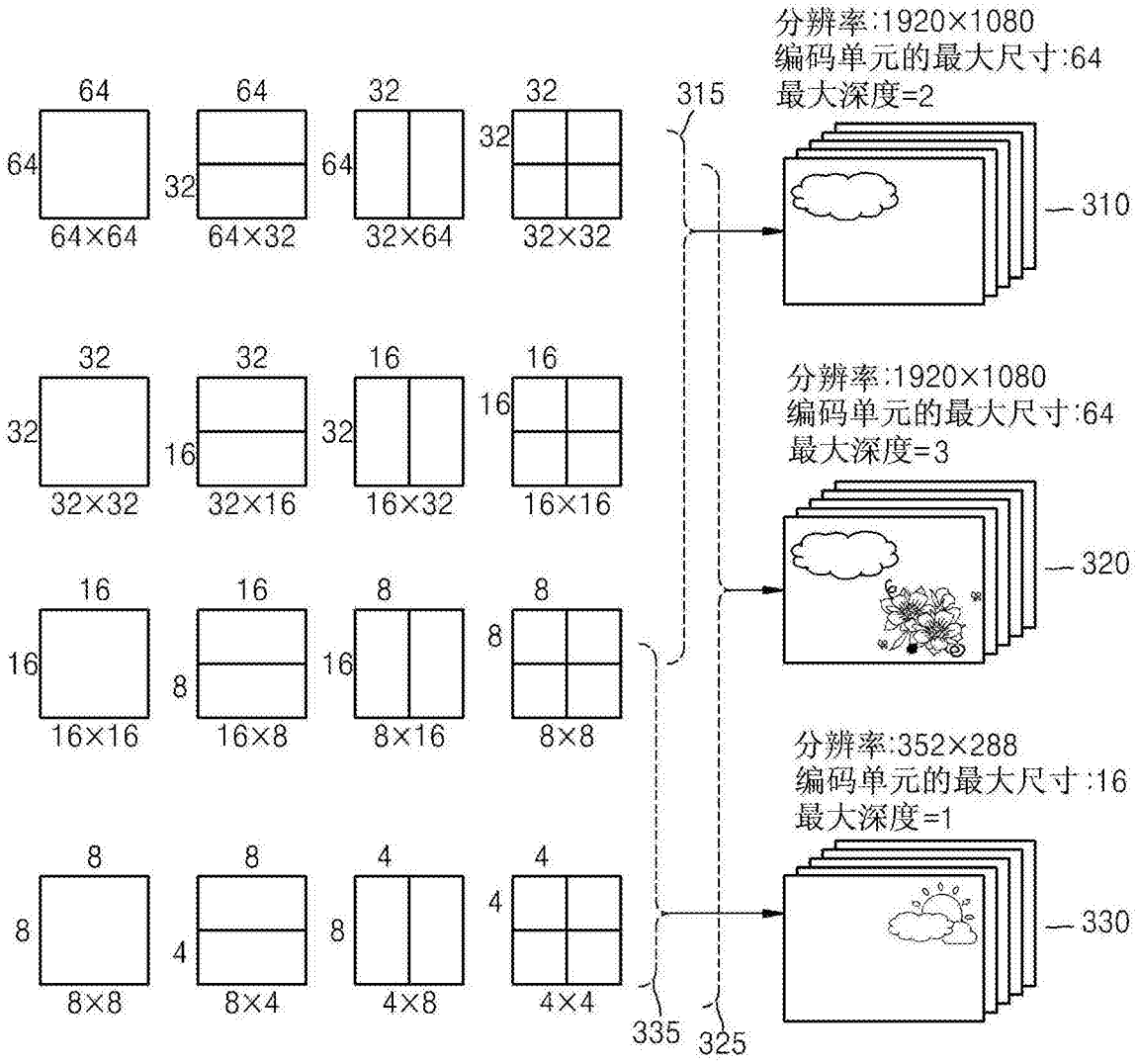

分辨率:1920×1080
编码单元的最大尺寸:64
最大深度=2
315
64
64
64×64
64
32
64×32
32
64
32×64
32
32
32×32
310
32
32
32×32
32
16
32×16
16
32
16×32
16
16
16×16
分辨率:1920×1080
编码单元的最大尺寸:64
最大深度=3
320
16
16
16×16
16
8
16×8
8
16
8×16
8
8
8×8
分辨率:352×288
编码单元的最大尺寸:16
最大深度=1
8
8
8×8
8
4
8×4
4
8
4×8
4
4
4×4
330
335
325

图3

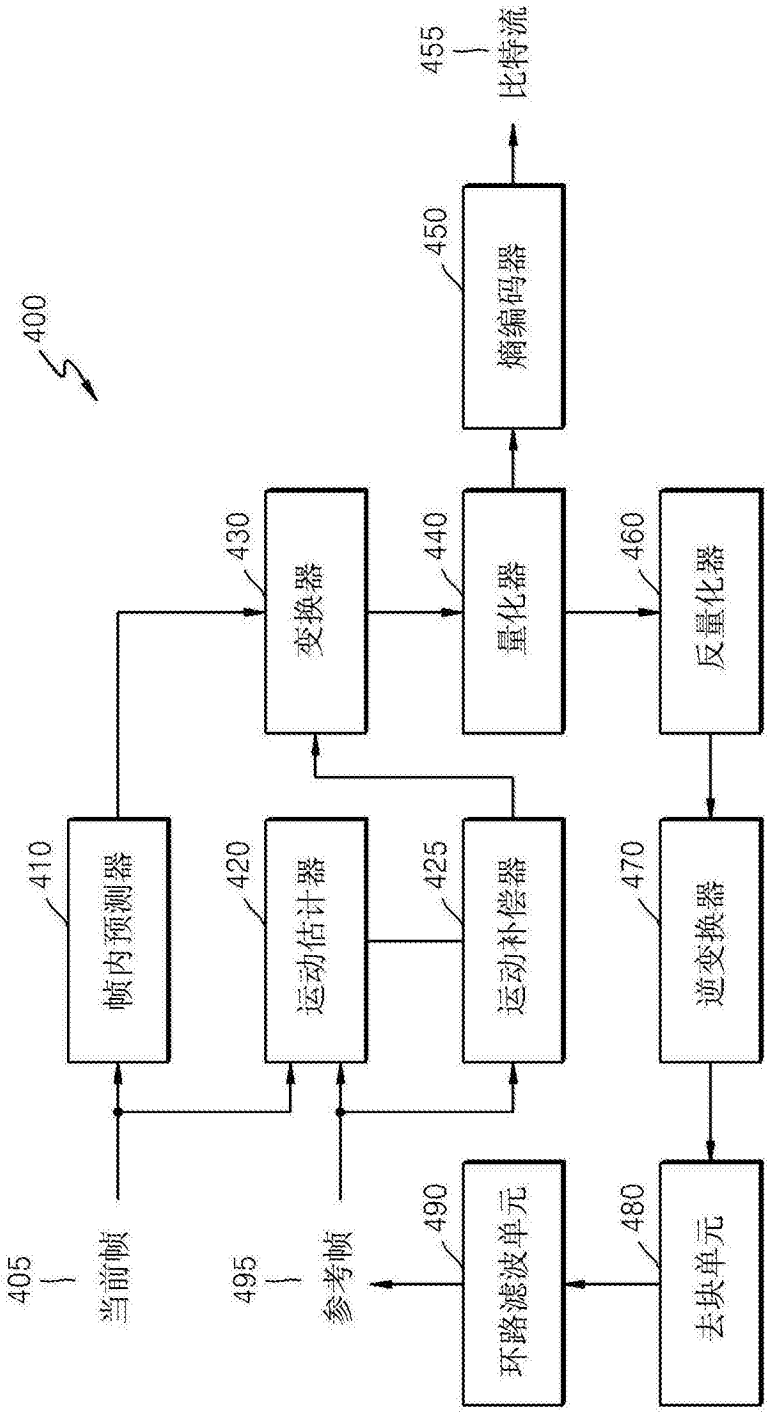

400
405
当前帧
410
帧内预测器
420
运动估计器
425
运动补偿器
430
变换器
440
量化器
450
熵编码器
455
比特流
460
反量化器
470
逆变换器
480
去块单元
490
环路滤波单元
495
参考帧

图4

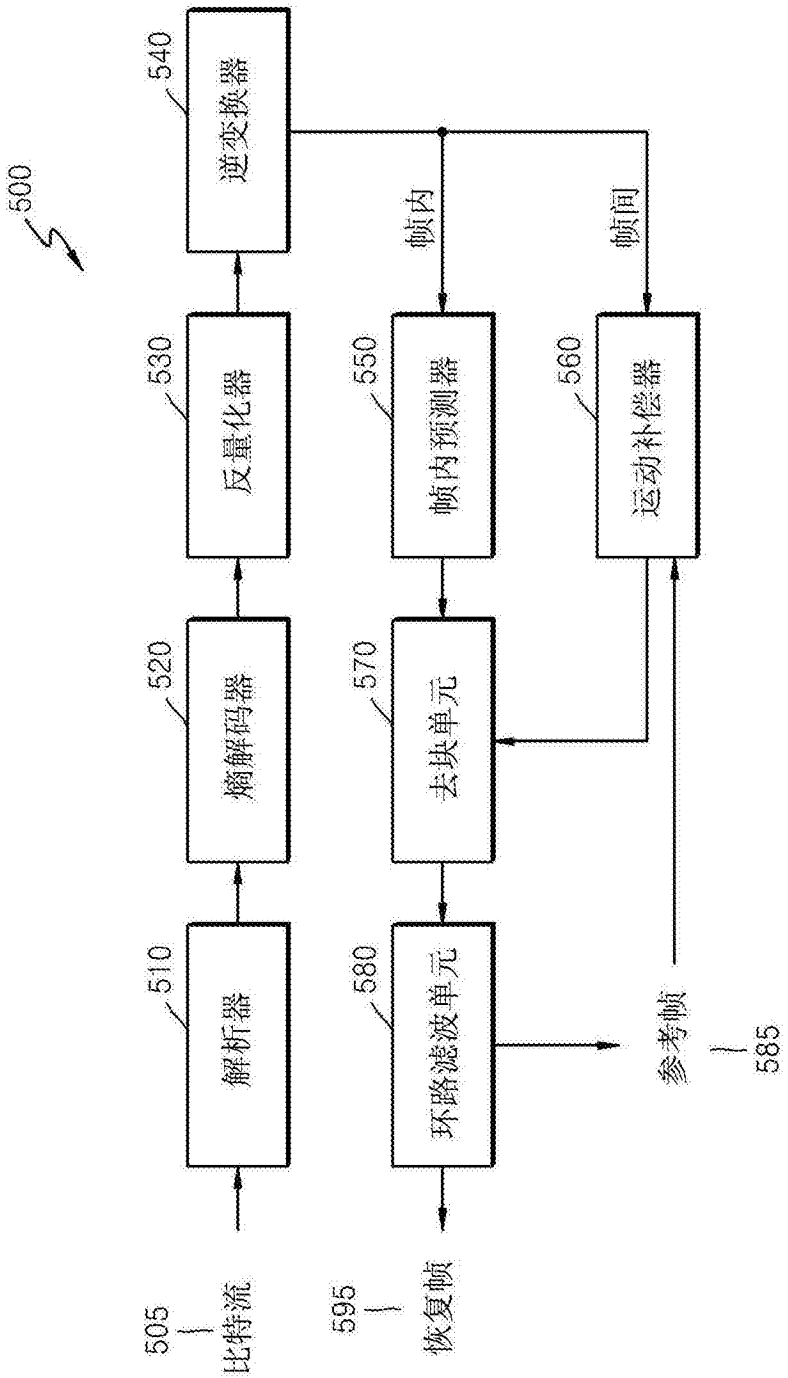

500
540
逆变换器
530
反量化器
520
熵解码器
510
解析器
505
比特流
帧内
帧间
550
帧内预测器
560
运动补偿器
570
去块单元
580
环路滤波单元
595
恢复帧
参考帧
585

图5

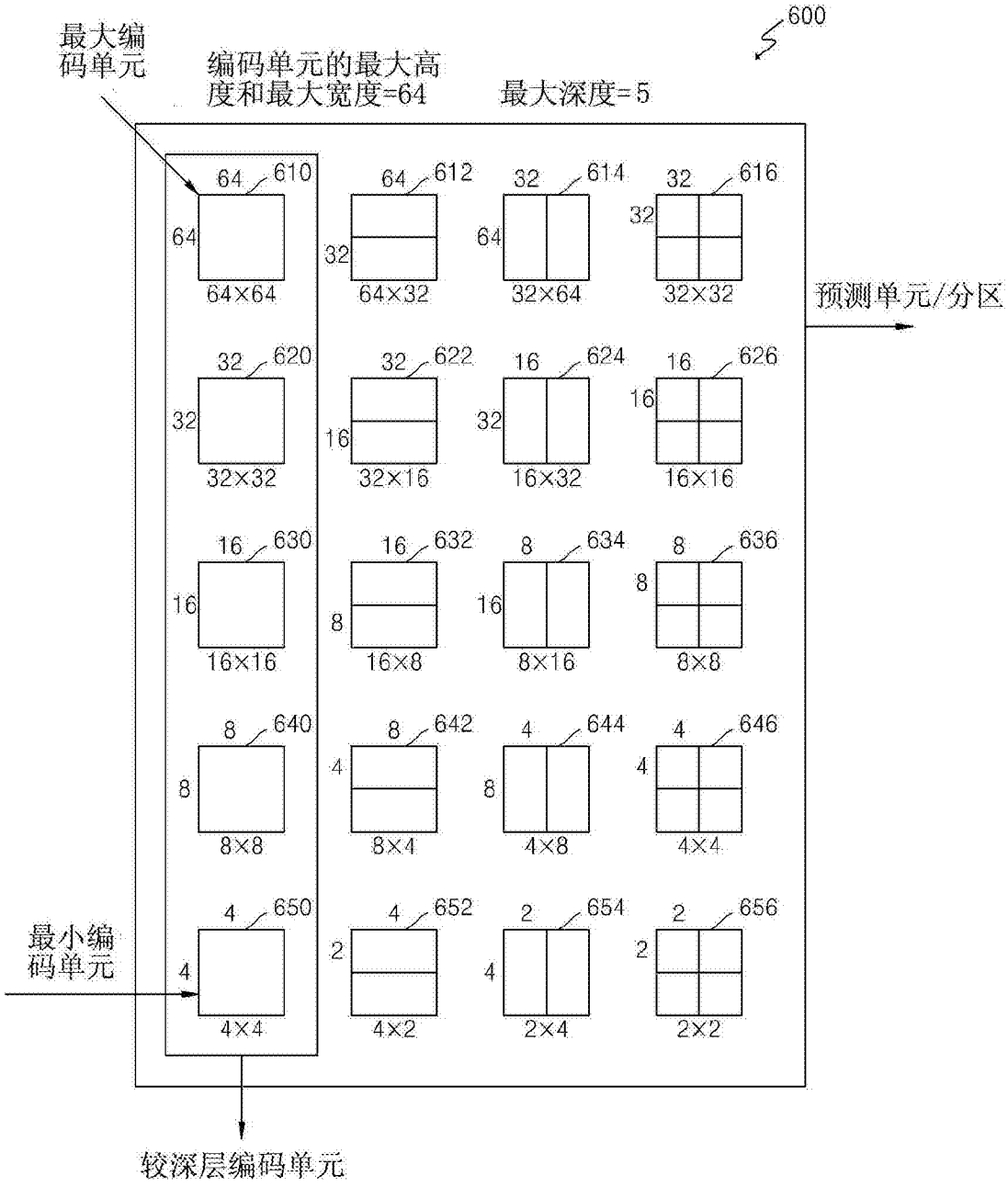

600
最大编码单元
编码单元的最大高度和最大宽度=64
最大深度=5
64 610
64×64
64 612
64×32
32 614
32×64
32 616
32×32
预测单元/分区
32 620
32×32
32 622
32×16
16 624
16×32
16 626
16×16
16 630
16×16
16 632
16×8
8 634
8×16
8 636
8×8
8 640
8×8
8 642
8×4
4 644
4×8
4 646
4×4
最小编码单元
4 650
4×4
4 652
4×2
2 654
2×4
2 656
2×2
较深层编码单元

图6

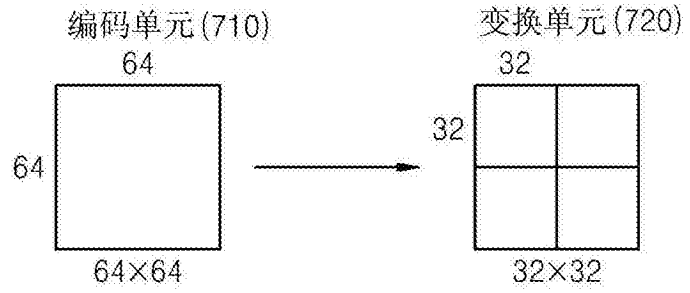

编码单元(710)
64
64
64×64
变换单元(720)
32
32
32×32

图7

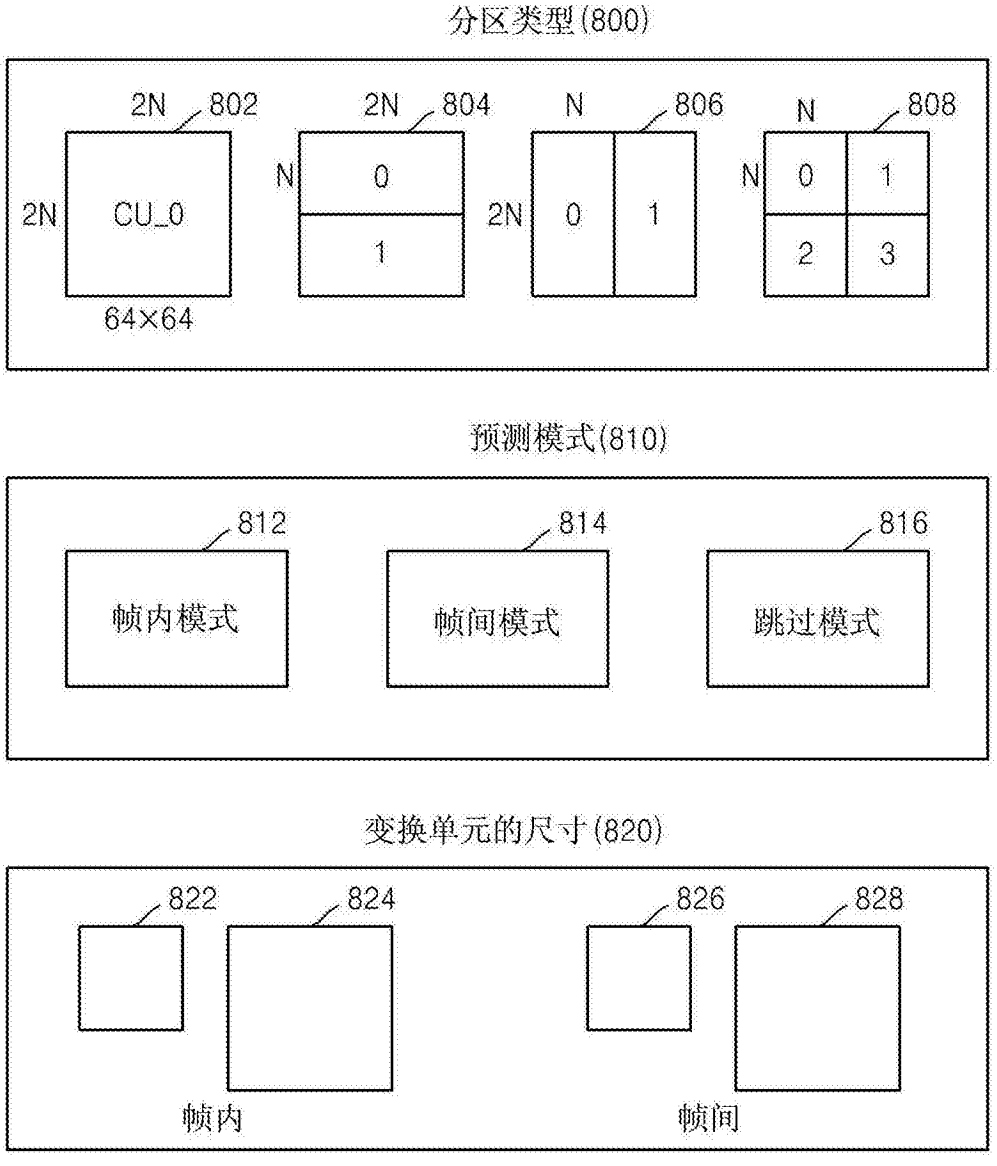

分区类型(800)
2N
802
2N
CU_0
64×64
2N
804
N
0
1
N
806
2N
0
1
N
808
N
0
1
2
3
预测模式(810)
812
帧内模式
814
帧间模式
816
跳过模式
变换单元的尺寸(820)
822
824
826
828
帧内
帧间

图8

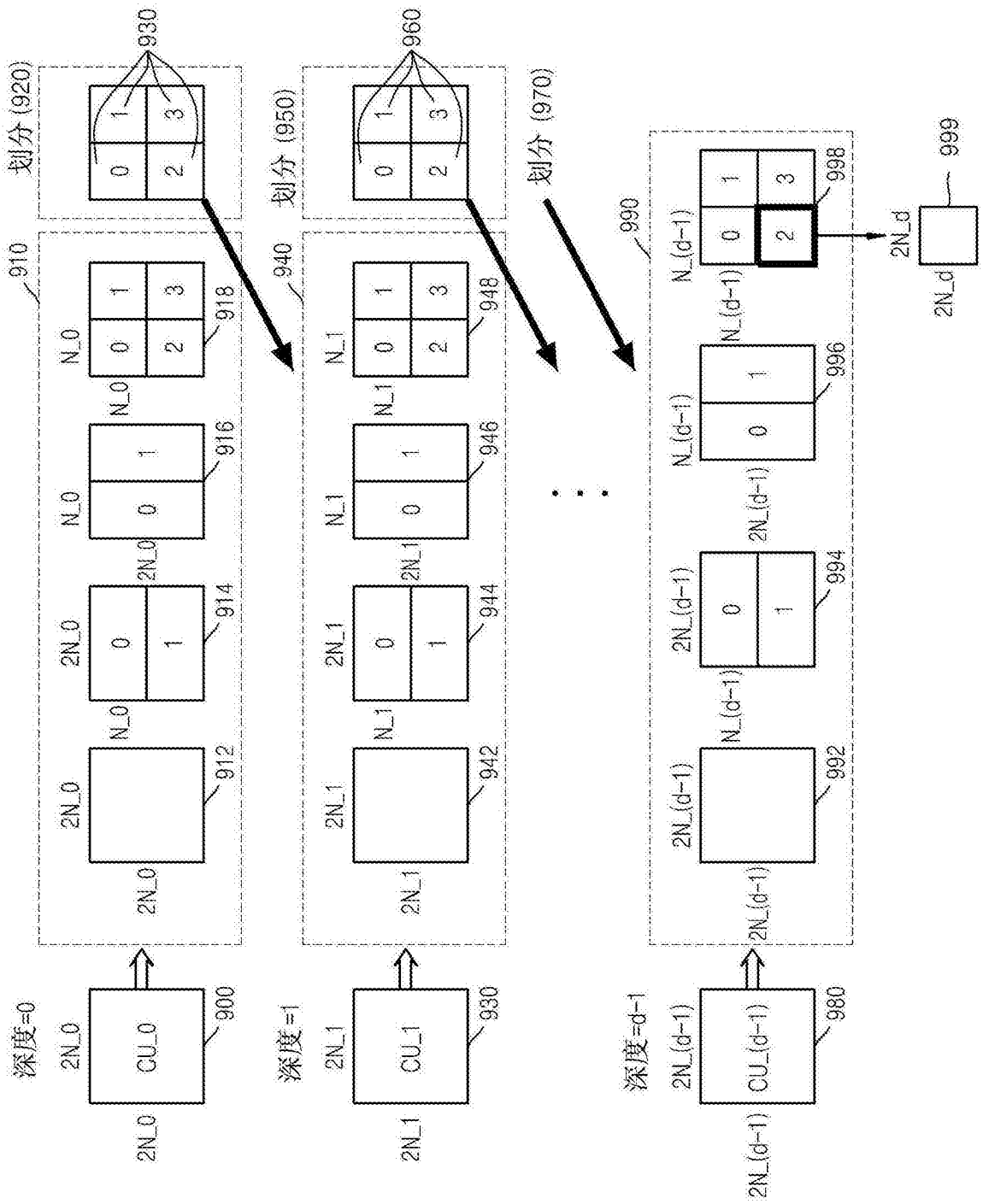

深度=0
深度=1
深度=d-1
划分 (920)
划分 (950)
划分 (970)
CU_0
CU_1
CU_(d-1)
900
910
912
914
916
918
930
940
942
944
946
948
960
980
990
992
994
996
998
999
2N_0
N_0
2N_1
N_1
2N_(d-1)
N_(d-1)
2N_d

图9

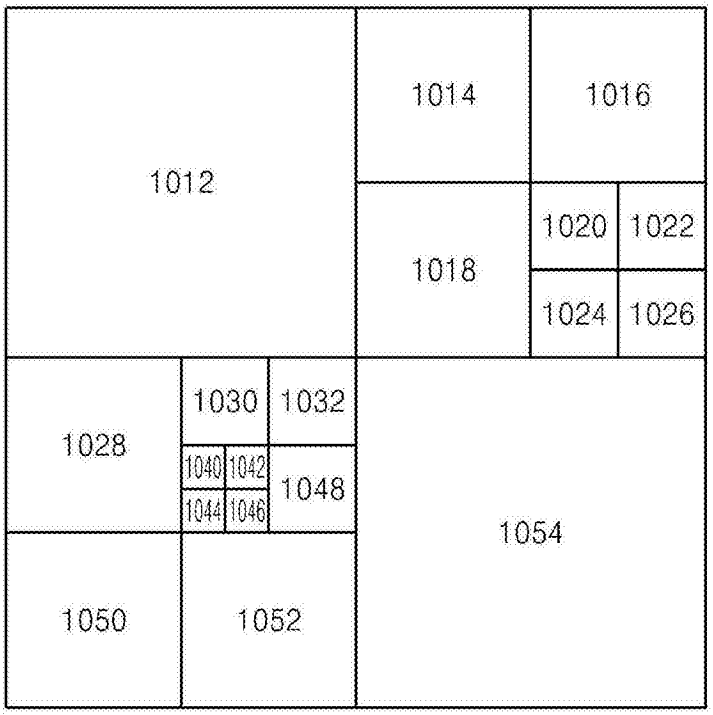

1012
1014
1016
1018
1020
1022
1024
1026
1028
1030
1032
1040
1042
1044
1046
1048
1050
1052
1054
编码单元 (1010)

图10

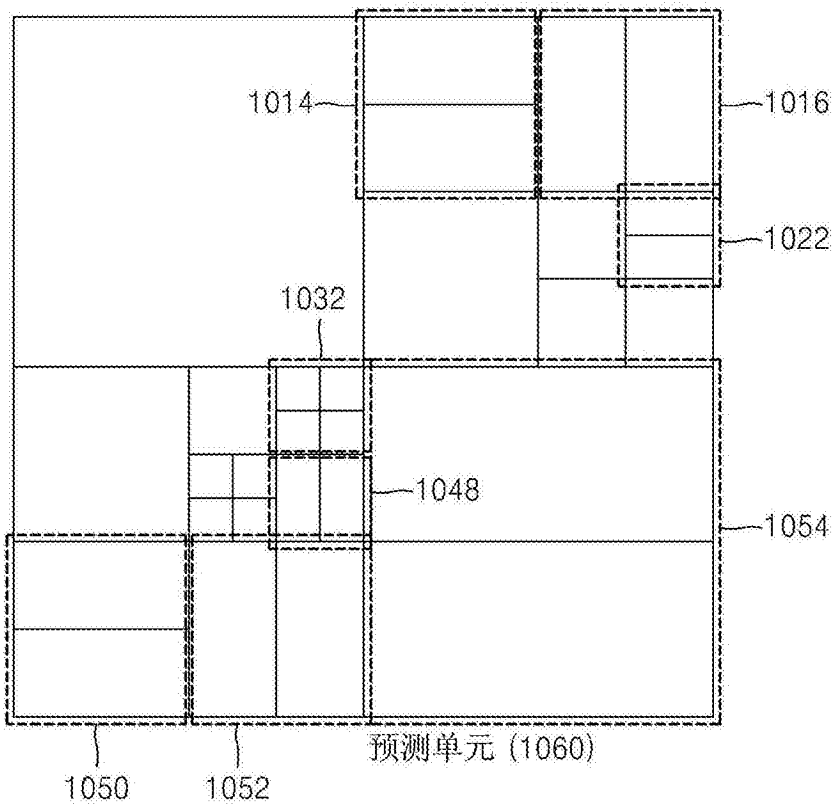

1014
1016
1022
1032
1048
1054
1050
1052
预测单元 (1060)

图11

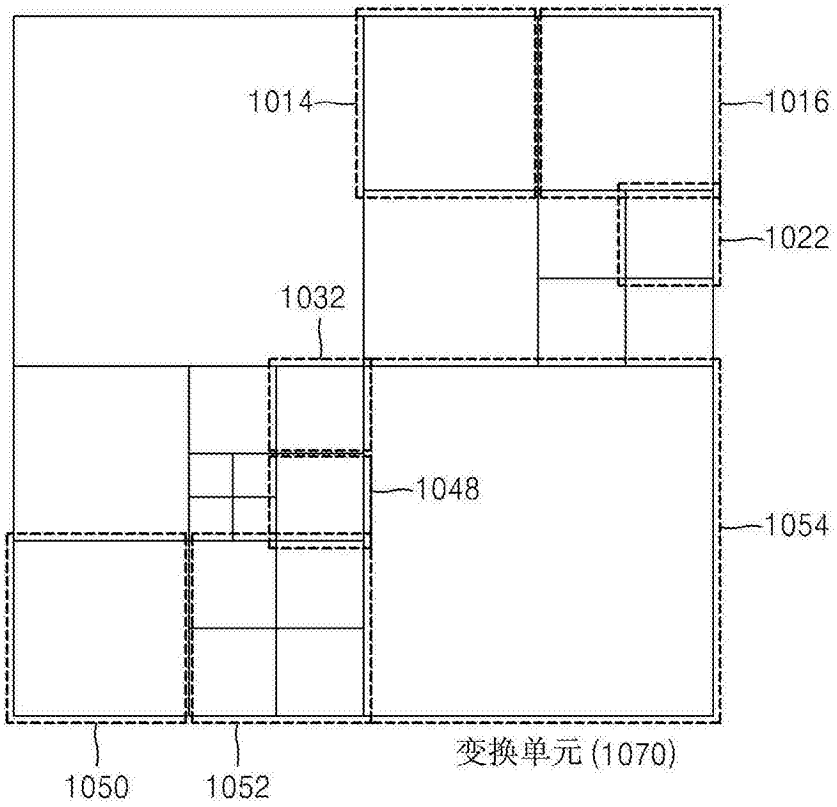


图12

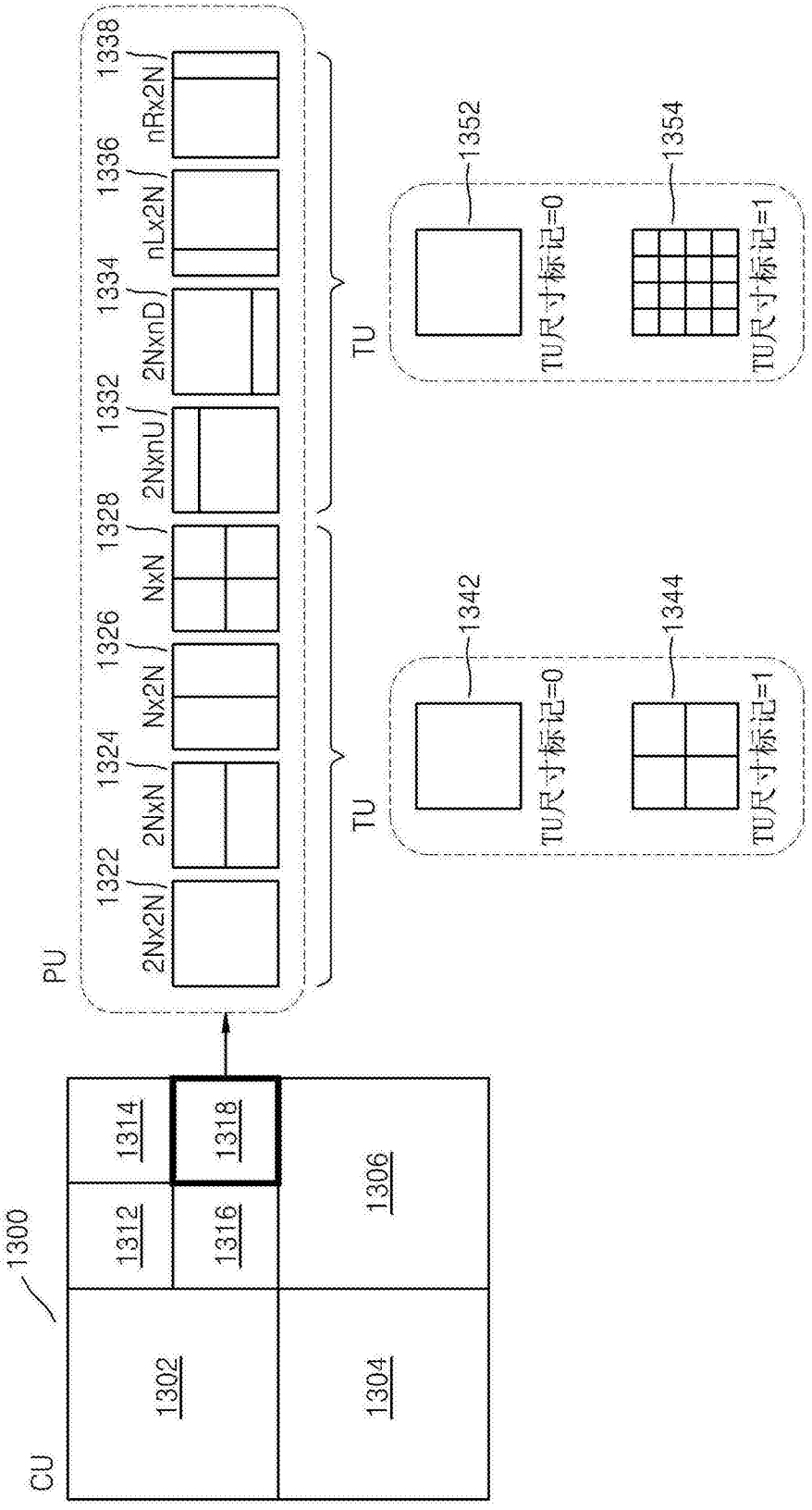

CU
1300
1302
1304
1306
1312
1314
1316
1318
PU
1322
2Nx2N
1324
2NxN
1326
Nx2N
1328
NxN
1332
2NxnU
1334
2NxnD
1336
nLx2N
1338
nRx2N
TU
TU
1342
TU尺寸标记=0
1344
TU尺寸标记=1
1352
TU尺寸标记=0
1354
TU尺寸标记=1

图13

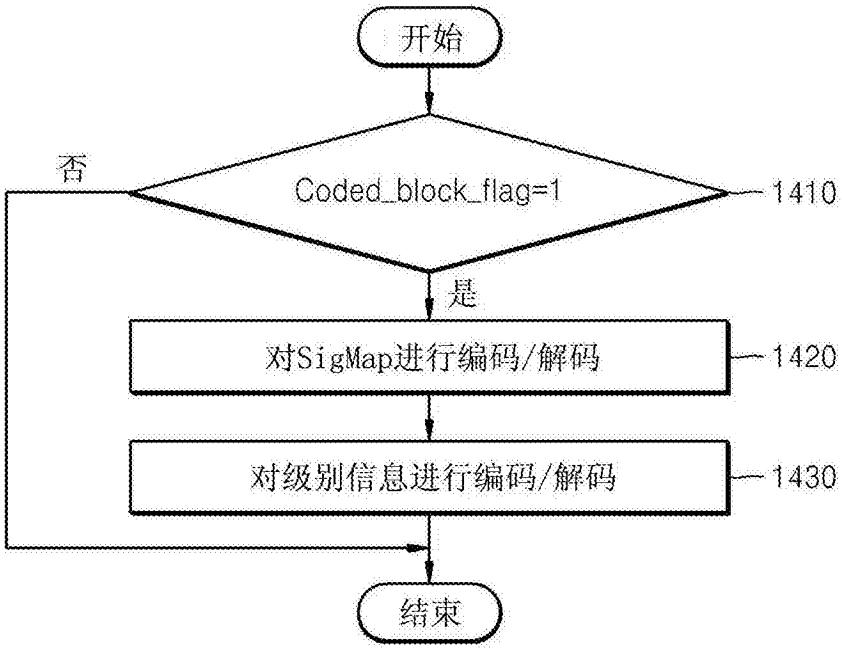

开始
Coded_block_flag=1
1410
否
是
对SigMap进行编码/解码
1420
对级别信息进行编码/解码
1430
结束

图14

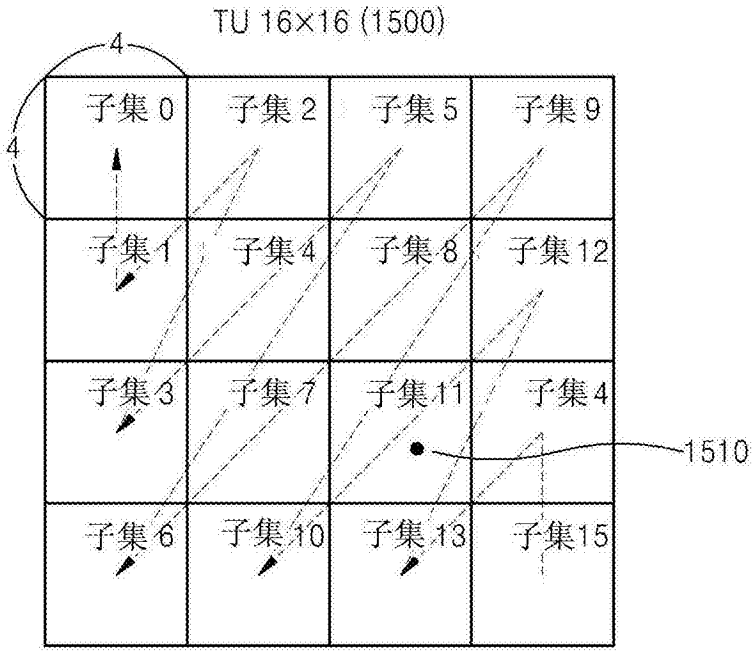

TU 16×16 (1500)
4
4
子集 0
子集 2
子集 5
子集 9
子集 1
子集 4
子集 8
子集 12
子集 3
子集 7
子集 11
子集 4
1510
子集 6
子集 10
子集 13
子集15

图15

子集(1600)

23	0	3	1
1	2	0	0
1	0	1	0
2	0	0	0

图16

SigMap (1700)

1	0	1	1
1	1	0	0
1	0	1	0
1	0	0	0

图17

GTR1标记 (1800)

1		1	0
0	1		
0		0	
1			

图18

GTR2 标记(1900)

1		1	
	0		
0			

图19

处理顺序（扫描顺序）

系数	23	1	0	1	2	3	2	0	0	1	0	1
SIGMAP	1	1	0	1	1	1	1	0	0	1	0	1
GTR1	1	0		0	1	1	1			0		0
GTR2	1				0	1	0					
Level-3	20					0						

图20

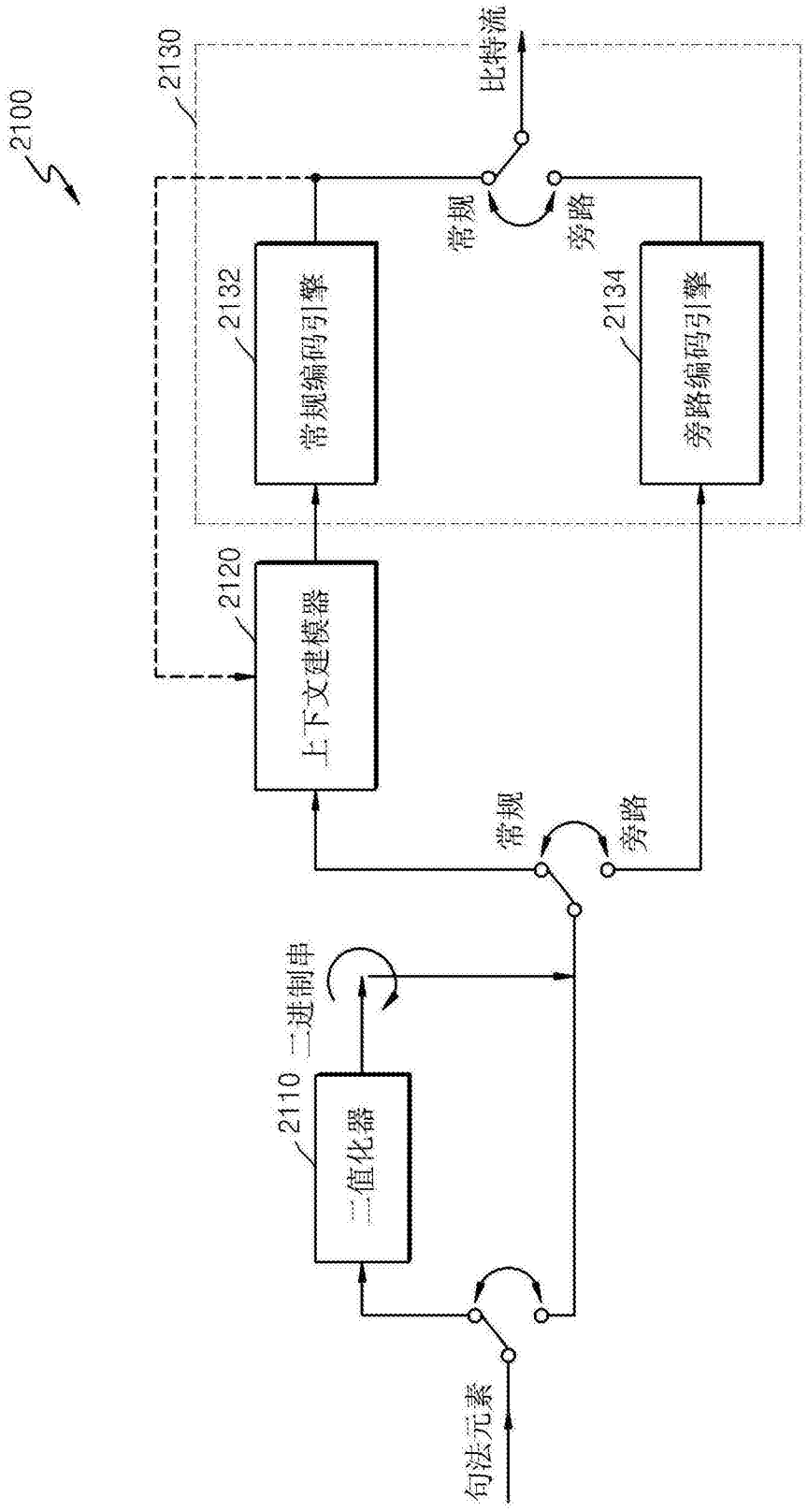

2100
2130
比特流
常规
旁路
2132
常规编码引擎
2134
旁路编码引擎
2120
上下文建模器
常规
旁路
二进制串
2110
二值化器
句法元素

图21a

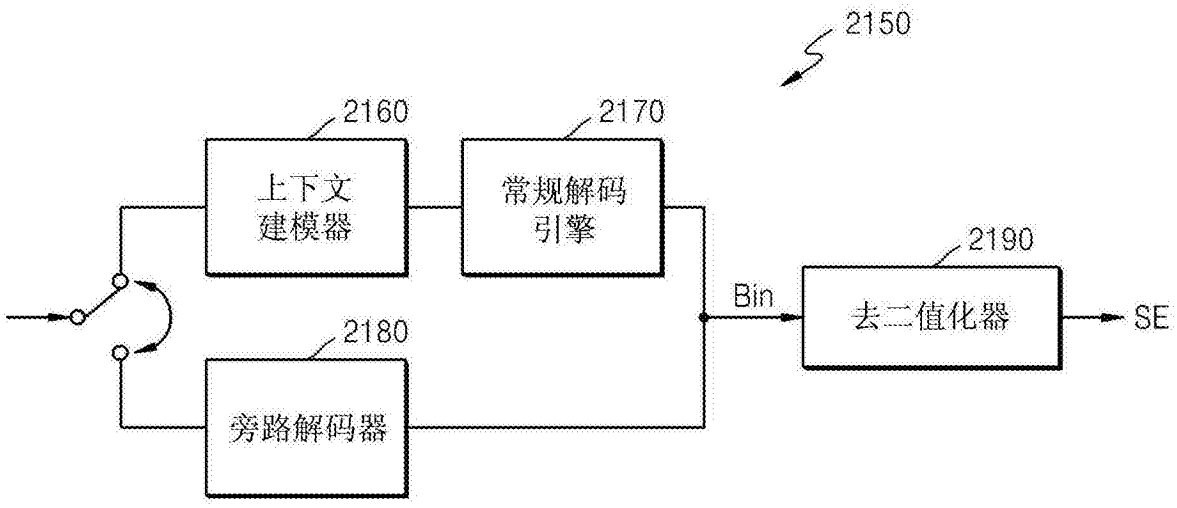

2150
2160
上下文
建模器
2170
常规解码
引擎
2190
Bin
去二值化器
SE
2180
旁路解码器

图21b

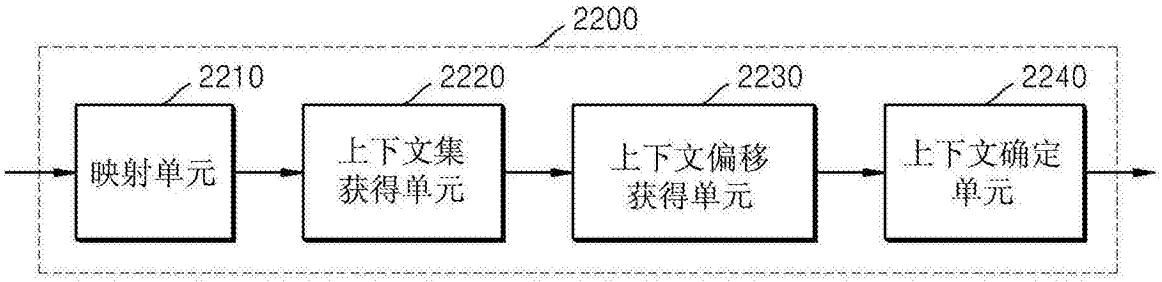

2200
2210
映射单元
2220
上下文集
获得单元
2230
上下文偏移
获得单元
2240
上下文确定
单元

图22

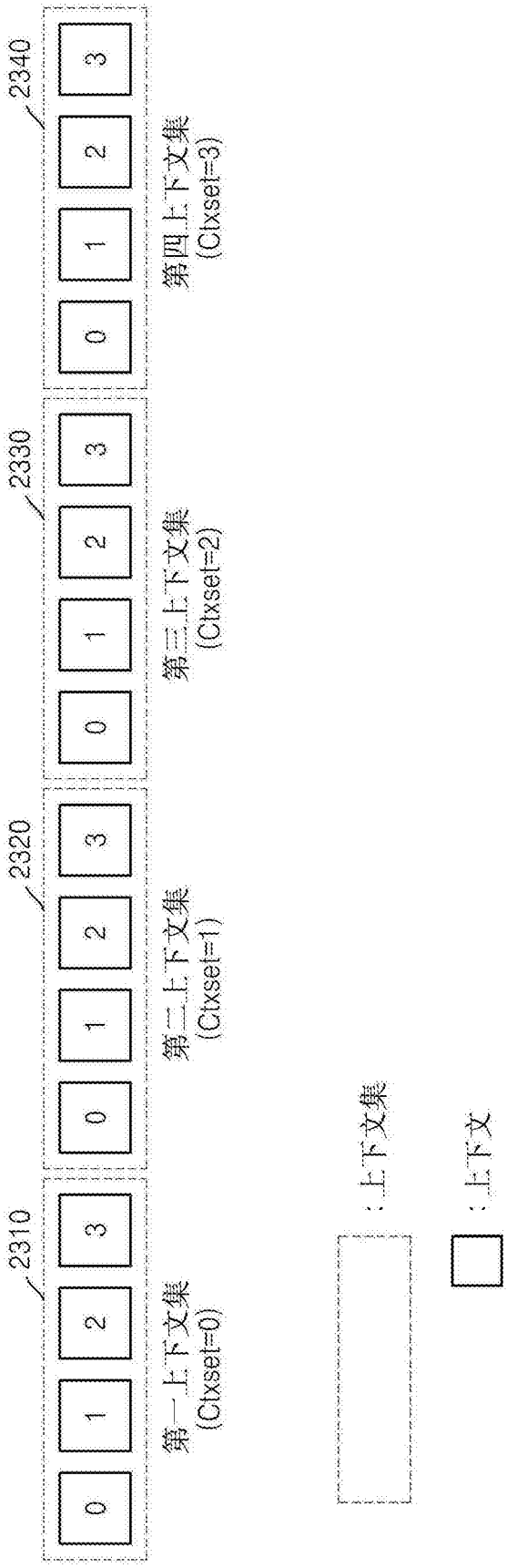

2310
0
1
2
3
第一上下文集
(Ctxset=0)
2320
0
1
2
3
第二上下文集
(Ctxset=1)
2330
0
1
2
3
第三上下文集
(Ctxset=2)
2340
0
1
2
3
第四上下文集
(Ctxset=3)
:上下文集
:上下文

图23

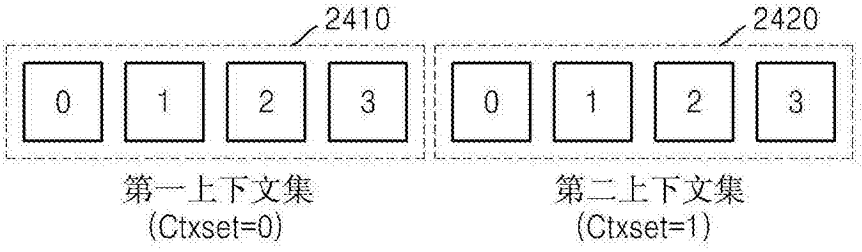

2410
2420
0
1
2
3
0
1
2
3
第一上下文集
(Ctxset=0)
第二上下文集
(Ctxset=1)

图24

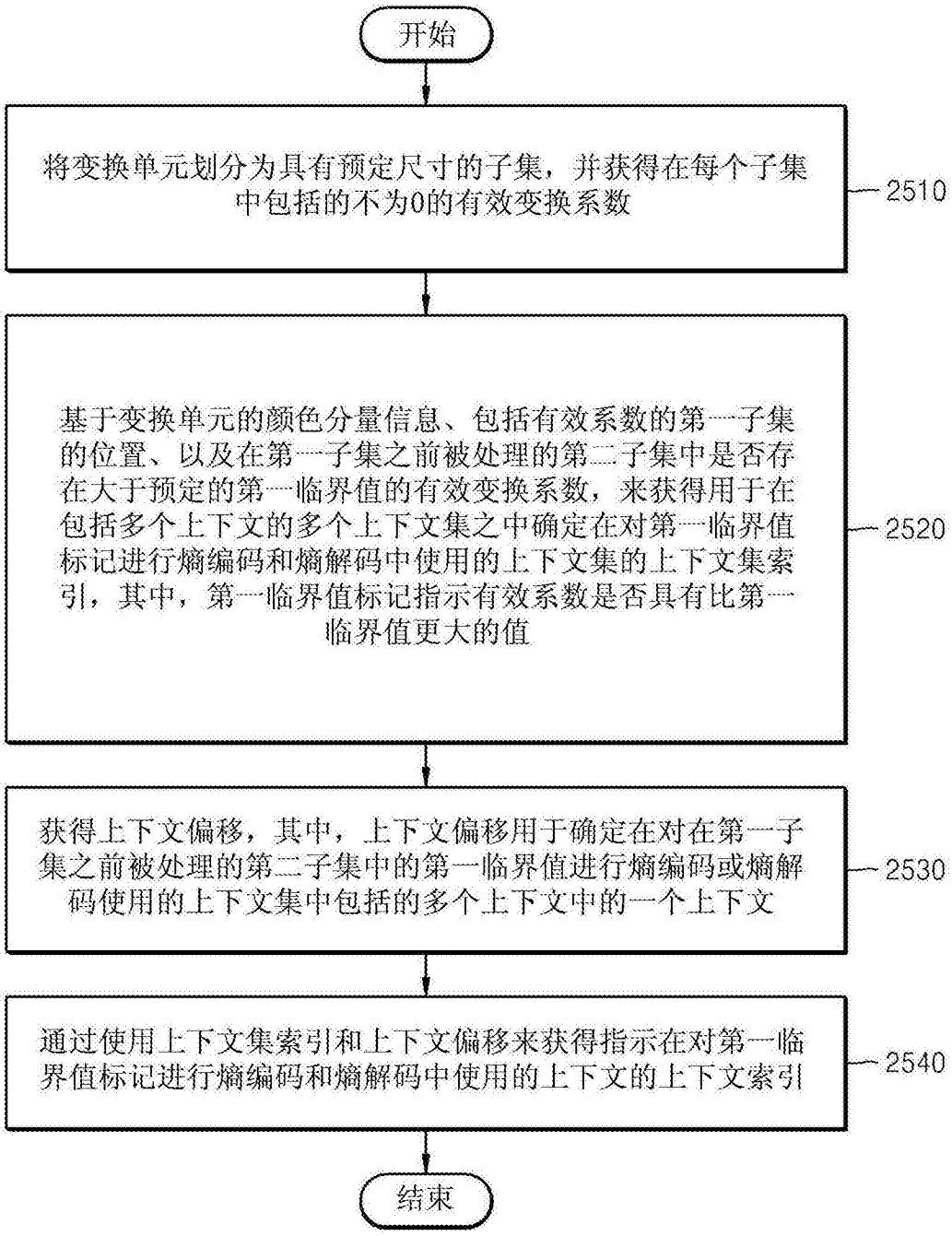

开始
将变换单元划分为具有预定尺寸的子集，并获得在每个子集中包括的不为0的有效变换系数
2510
基于变换单元的颜色分量信息、包括有效系数的第一子集的位置、以及在第一子集之前被处理的第二子集中是否存在大于预定的第一临界值的有效变换系数，来获得用于在包括多个上下文的多个上下文集之中确定在对第一临界值标记进行熵编码和熵解码中使用的上下文集的上下文集索引，其中，第一临界值标记指示有效系数是否具有比第一临界值更大的值
2520
获得上下文偏移，其中，上下文偏移用于确定在对在第一子集之前被处理的第二子集中的第一临界值进行熵编码或熵解码使用的上下文集中包括的多个上下文中的一个上下文
2530
通过使用上下文集索引和上下文偏移来获得指示在对第一临界值标记进行熵编码和熵解码中使用的上下文的上下文索引
2540
结束

图25

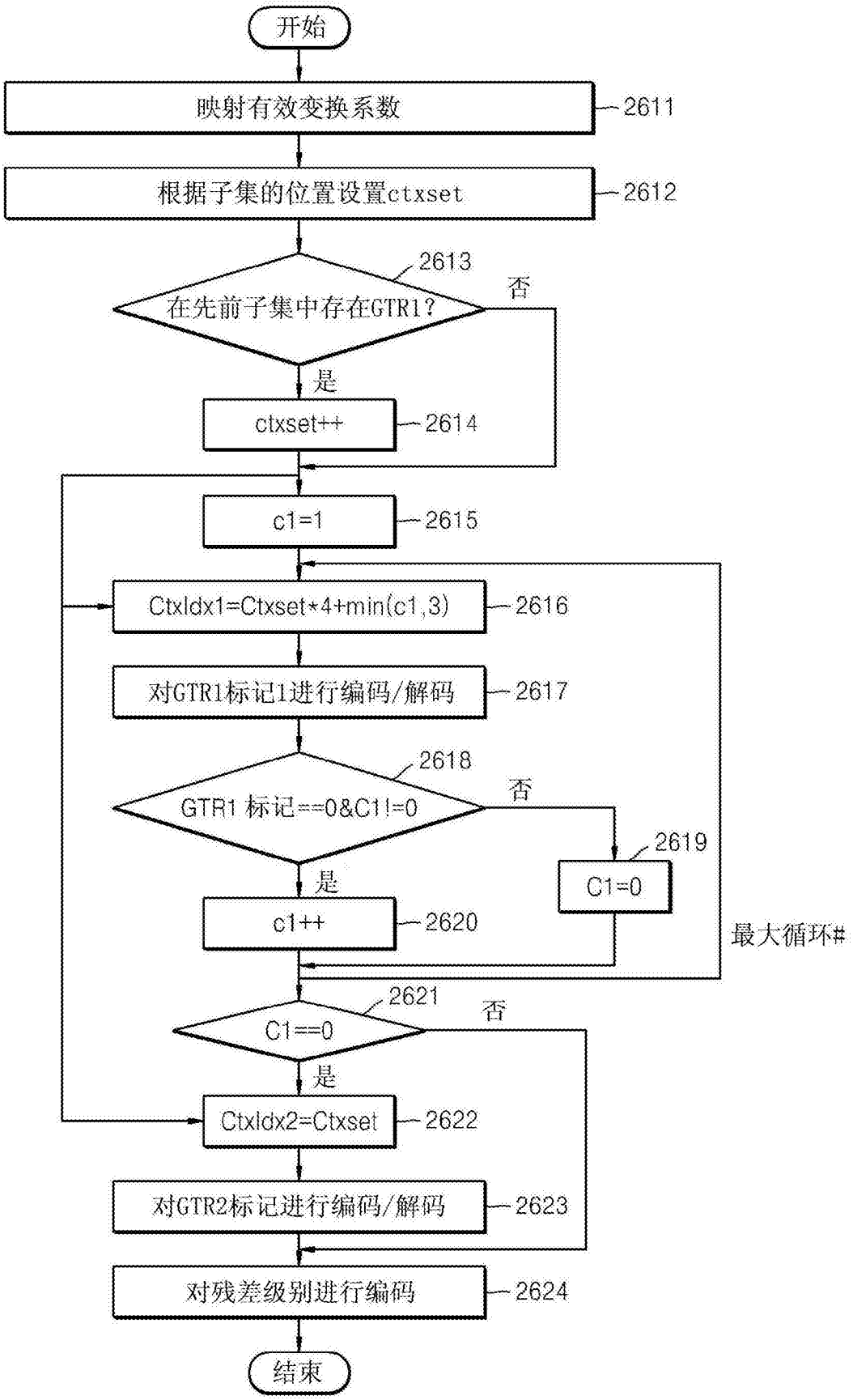

开始
映射有效变换系数
2611
根据子集的位置设置ctxset
2612
2613
在先前子集中存在GTR1?
否
是
ctxset++
2614
c1=1
2615
CtxIdx1=Ctxset*4+min(c1,3)
2616
对GTR1标记1进行编码/解码
2617
2618
GTR1 标记==0&C1!=0
否
2619
C1=0
是
c1++
2620
最大循环#
2621
C1==0
否
是
CtxIdx2=Ctxset
2622
对GTR2标记进行编码/解码
2623
对残差级别进行编码
2624
结束

图26

上下文集

亮度

Ctxset亮度（先前子集）		
0 1	子集0	没有GreatT1 至少一个GreatT1
2 3	其他子集	没有GreatT1 至少一个GreatT1

色度

CtxSet_Chroma（先前子集）	
0 1	没有GreatT1 至少一个GreatT1

图27a

上下文集中的偏移

GT1 标记

c1（当前子集）	
0	1个或更多个GreatT1
1	初始-没有拖尾1
2	1个拖尾1
3	2个或更多拖尾1

GT2 标记

c2（当前子集）	
0	第一greatT2

图27b

处理顺序 ←

2830 2820 2810

系数	23	1	0	1	2	3	②	0	0	①	0	①
SIGMAP	1	1	0	1	1	1	1	0	0	1	0	1
GTR1	1	0		0	1	1	1			0		0
GTR2	1				0	1	0					
Level-3	20					0						
C1	0	0		0	0	0	3			2		1

图28

处理顺序 ←

2910

系数	23	1	0	1	2	3	2	0	0	1	0	②
SIGMAP	1	1	0	1	1	1	1	0	0	1	0	1
GTR1	1	0		0	1	1	1			0		1
GTR2	1				0	1	0					0
Level-3	20					0						
C1	0	0		0	0	0	0			0		1

图29