(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 105847811 B
(45)授权公告日 2019.06.07

(21)申请号 201610305424.8

(22)申请日 2011.07.08

(65)同一申请的已公布的文献号
申请公布号 CN 105847811 A

(43)申请公布日 2016.08.10

(30)优先权数据
61/362,844 2010.07.09 US

(62)分案原申请数据
201180043638.8 2011.07.08

(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 李培根 孙有美

(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 张云珠 苏银虹

(51)Int.Cl.
H04N 19/176(2014.01)
H04N 19/13(2014.01)
H04N 19/129(2014.01)
H04N 19/463(2014.01)
H04N 19/91(2014.01)
H04N 19/18(2014.01)

(56)对比文件
WO 2010018138 A1,2010.02.18,
US 2004114683 A1,2004.06.17,
WO 2010054178 A1,2010.05.14,
无.Test Model under Consideration.《Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11》.2010,

审查员 史亦澍

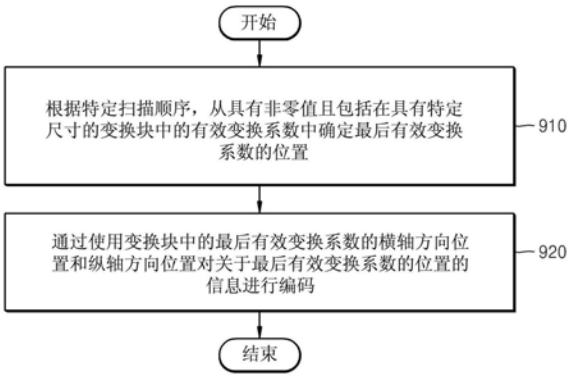
权利要求书1页 说明书13页 附图18页

(54)发明名称

用于对变换系数进行熵解码的设备

(57)摘要

一种用于对变换系数进行熵编码/熵解码的方法和设备。对变换块进行熵编码的方法包括：根据特定扫描顺序，确定包括在具有特定尺寸的变换块中的变换系数之中具有非零值的最后有效变换系数的位置；通过使用变换块中的最后有效变换系数的横轴方向位置和最后有效变换系数的纵轴方向位置来对关于最后有效变换系数的位置的信息进行编码。



CN 105847811 B

1.一种用于对变换系数进行熵解码的设备,所述设备包括熵解码器,所述熵解码器被配置用于:

从接收的比特流提取关于具有非零值并包括在变换块中的最后有效变换系数的水平轴方向位置和垂直轴方向位置的信息,其中,最后有效变换系数是变换块的有效变换系数中从变换块的左上位置开始位于最后的有效变换系数,

通过对关于水平轴方向位置和垂直轴方向位置的信息进行解码来确定最后有效变换系数的位置,

基于最后有效变换系数的位置对最后有效系数进行解码,

对包括在接收的比特流中表示第二有效变换系数的位置的有效系数标志进行解码,以及

通过使用有效系数标志对第二有效变换系数进行解码,

其中,所述信息包括与水平轴方向位置相应的x值和与垂直轴方向位置相应的y值。

用于对变换系数进行熵解码的设备

[0001] 本申请是申请日为2011年7月8日、申请号为201180043638.8、题为“用于对变换系数进行熵编码/熵解码的方法和设备”的专利申请的分案申请。

技术领域

[0002] 本发明涉及变换系数的熵编码和熵解码，更具体地讲，涉及一种对关于变换块中的最后有效变换系数的位置的信息进行有效的熵编码和熵解码的方法和设备。

背景技术

[0003] 根据诸如H.264和MPEG-4的国际视频编码标准，视频信号被分层划分为序列、帧、条带、宏块和块，并且块是最小处理单元。在编码处理中，通过执行帧内预测或帧间预测获得块的残差数据。此外，通过执行变换、量化、扫描、行程编码和熵编码来压缩残差数据。解码处理是编码处理的逆处理。首先，从比特流提取在熵编码处理中产生的变换块的系数。然后，通过执行反量化和逆变换来重构块的残差数据，预测信息用于重构块的视频数据。

发明内容

[0004] 技术问题

[0005] 本发明提供一种对关于具有大尺寸的变换块中的最后有效变换系数的位置的信息进行有效的熵编码和熵解码的方法和设备。

[0006] 技术方案

[0007] 根据本发明的实施例，通过使用变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置来对关于变换块中的最后有效变换系数的位置的信息进行编码。

[0008] 有益效果

[0009] 根据本发明，可有效地表示包括在具有大尺寸的变换块中的最后有效变换系数的位置，并且可独立于对变换系数进行解码的处理来对关于最后有效变换系数的位置的信息进行解码。

附图说明

[0010] 图1是根据本发明实施例的图像编码设备的框图。

[0011] 图2是根据本发明实施例的图像解码设备的框图。

[0012] 图3是示出根据本发明实施例的分层编码单元的示图。

[0013] 图4是根据本发明实施例的基于编码单元的图像编码器的框图。

[0014] 图5是根据本发明实施例的基于编码单元的图像解码器的框图。

[0015] 图6是示出根据本发明实施例的最大编码单元、子编码单元和预测单元的示图。

[0016] 图7是示出根据本发明实施例的编码单元和变换单元的示图。

[0017] 图8A和图8B是示出根据本发明实施例的编码单元、预测单元和变换单元的划分形状的示图。

[0018] 图9是根据本发明实施例的对变换系数进行熵编码的方法的流程图。

[0019] 图10是用于描述根据本发明实施例的对变换系数进行熵编码的处理的参考示图。

[0020] 图11是根据本发明实施例的熵编码设备的框图。

[0021] 图12是根据本发明实施例的上下文自适应二进制算术编码(CABAC)装置的框图。

[0022] 图13是用于描述根据本发明实施例的选择用于对关于最后有效变换系数的位置的信息进行编码的上下文的处理的参考示图。

[0023] 图14示出与图10相应的有效映射。

[0024] 图15是用于描述在图10中示出的对包括在变换块中的有效变换系数的级别值进行编码的处理的参考示图。

[0025] 图16是示出根据本发明实施例使用的多个可变长度编码(VLC)表的示例的示图。

[0026] 图17是用于描述根据本发明另一实施例的对变换系数进行熵编码的方法的参考示图。

[0027] 图18A和图18B是用于描述根据本发明另一实施例的对变换系数进行熵编码的方法的参考示图。

[0028] 图19是根据本发明实施例的对变换系数进行熵解码的方法的流程图。

[0029] 图20是根据本发明实施例的熵解码设备的框图。

[0030] 最佳模式

[0031] 根据本发明的一方面,提供一种对变换系数进行熵编码的方法,所述方法包括:根据特定扫描顺序,确定包括在具有特定尺寸的变换块中的变换系数之中具有非零值的最后有效变换系数的位置;通过使用变换块中的最后有效变换系数的横轴方向位置和最后有效变换系数的纵轴方向位置来对关于最后有效变换系数的位置的信息进行编码。

[0032] 根据本发明的另一方面,提供一种对变换系数进行熵解码的方法,所述方法包括:根据特定扫描顺序,从接收的比特流提取关于具有非零值且包括在变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置的信息;通过对关于横轴方向位置和纵轴方向位置的信息进行解码来确定最后有效变换系数的位置。

[0033] 根据本发明的另一方面,提供一种对变换系数进行熵编码的设备,所述设备包括:熵编码器,根据特定扫描顺序,确定包括在具有特定尺寸的变换块中的变换系数之中具有非零值的最后有效变换系数的位置,并通过使用变换块中的最后有效变换系数的横轴方向位置和最后有效变换系数的纵轴方向位置来对关于最后有效变换系数的位置的信息进行编码。

[0034] 根据本发明的另一方面,提供一种对变换系数进行熵解码的设备,所述设备包括:熵解码器,根据特定扫描顺序,从接收的比特流提取关于具有非零值且包括在变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置的信息,并通过对关于横轴方向位置和纵轴方向位置的信息进行解码来确定最后有效变换系数的位置。

具体实施方式

[0035] 在下文中,将通过参照附图解释本发明的实施例来详细描述本发明。

[0036] 图1是根据本发明实施例的图像编码设备100的框图。

[0037] 参照图1,图像编码设备100包括最大编码单元划分器110、编码深度确定器120、图

像数据编码器130和编码信息编码器140。

[0038] 最大编码单元划分器110可基于作为最大尺寸的编码单元的最大编码单元对当前帧或当前条带进行划分。可将当前帧或当前条带划分为至少一个最大编码单元。

[0039] 根据本发明实施例，可使用最大编码单元和深度来表示编码单元。如上所述，最大编码单元表示在当前帧的编码单元中具有最大尺寸的编码单元，深度表示编码单元在尺寸上如何分层地减小。随着深度增大，编码单元在尺寸上可从最大编码单元减小到最小编码单元，最大编码单元的深度可被定义为最小深度，而最小编码单元的深度可被定义为最大深度。由于编码单元在尺寸上随着深度增大而从最大编码单元减小，因此深度为k的子编码单元可包括多个具有大于k的深度的子编码单元。

[0040] 当将被编码的帧具有大尺寸时，如果以大的单元对图像进行编码，则可以以高的图像压缩率对图像进行编码。然而如果编码单元的尺寸增大并且被固定，则无法通过反映图像的连续改变特性来有效地对图像进行编码。

[0041] 例如，当对诸如海洋或天空的图像的平滑图像进行编码时，如果编码单元尺寸增大，则压缩率可提高。然而，当对诸如人或建筑物的图像的复杂图像进行编码时，如果编码单元尺寸减小，则压缩率提高。

[0042] 为此，根据本发明实施例，将不同尺寸的最大编码单元和不同的最大深度设置到不同的帧或条带。因为最大深度表示编码单元在尺寸上可减小的最大次数，所以包括在最大编码单元中的最小编码单元的尺寸可根据最大深度被可变地设置。

[0043] 编码深度确定器120确定最大深度。可基于率失真(R-D)成本来确定最大深度。最大深度可被不同地确定到每个帧或条带或者每个最大编码单元。关于确定的最大深度的信息被输出到编码信息编码器140，每个最大编码单元的图像数据被输出到图像数据编码器130。

[0044] 最大深度表示在最大编码单元中具有最小尺寸的编码单元，即，最小编码单元。换句话说，可根据不同深度将最大编码单元划分为不同尺寸的子编码单元。以下将参照图8A和图8B对此进行详细描述。此外，可基于不同尺寸的处理单元对包括在最大编码单元中的不同尺寸的子编码单元进行预测或正交变换。换句话说，图像编码设备100可基于各种尺寸和各种形状的处理单元执行多种用于图像编码的处理。当诸如预测、正交变换和熵编码的三种处理被执行以对图像数据进行编码时，可在所有处理中使用相同尺寸的处理单元，或者可在不同处理中使用不同尺寸的处理单元。

[0045] 例如，图像编码设备100可选择与特定编码单元不同的处理单元以预测编码单元。

[0046] 如果编码单元具有2N×2N(其中，N是正整数)的尺寸，则用于预测的处理单元可具有2N×2N、2N×N、N×2N、N×N等的尺寸。换句话说，可基于具有通过将编码单元的高度和宽度中的至少一个二等分而获得的尺寸的处理单元来执行运动预测。在下文中，用于预测的处理单元被称为“预测单元”。

[0047] 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，并且可仅对特定尺寸或特定形状的预测单元执行特定的预测模式。例如，可仅对具有2N×2N或N×N的尺寸的正方形的预测单元执行帧内模式。此外，可仅对具有2N×2N的尺寸的预测单元执行跳过模式。如果编码单元包括多个预测单元，则可对每个预测单元执行预测，并可选择具有最小编码误差的预测单元。

[0048] 此外，图像编码设备100可基于具有与编码单元的尺寸不同的尺寸的处理单元对图像数据进行正交变换。可基于尺寸小于或等于编码单元的尺寸的数据单元对编码单元进行正交变换。在下文中，用于正交变换的处理单元被称为“变换单元”。

[0049] 编码深度确定器120可通过使用基于拉格朗日乘数(Lagrangian multiplier)的率失真最优化来确定包括在最大编码单元中的子编码单元。换句话说，可确定最大编码单元被划分为多个子编码单元的划分形状。这里，多个子编码单元根据深度具有不同的尺寸。其后，图像数据编码器130通过基于由编码深度确定器120确定的划分形状对最大编码单元进行编码来输出比特流。

[0050] 编码信息编码器140对关于由编码深度确定器120确定的最大编码单元的编码模式的信息进行编码。通过对关于最大编码单元的划分形状的信息、关于最大深度的信息以及关于根据深度的子编码单元的编码模式的信息进行编码来输出比特流。关于子编码单元的编码模式的信息可包括例如关于子编码单元的预测单元的信息、关于预测单元的预测模式的信息和关于子编码单元的变换单元的信息。

[0051] 关于最大编码单元的划分形状的信息可以是表示每个编码单元是否被划分的信息。例如，当最大编码单元被划分和编码时，表示最大编码单元是否被划分的信息被编码。此外，当通过划分最大编码单元而产生的子编码单元被划分和编码时，表示每个子编码单元是否被划分的信息被编码。表示编码单元是否被划分的信息可以是表示编码单元是否被划分的标志信息。

[0052] 因为最大编码单元包括不同尺寸的子编码单元，并且关于每个子编码单元的编码模式的信息必须被确定，所以针对一个最大编码单元可确定关于至少一个编码模式的信息。

[0053] 随着深度加深，图像编码设备100可通过划分最大编码单元的高度和宽度来产生子编码单元。也就是说，如果深度为k的编码单元具有2N×2N的尺寸，则深度为k+1的编码单元具有N×N的尺寸。

[0054] 因此，图像编码设备100可基于考虑图像特性的最大编码单元的尺寸和最大深度来确定每个最大编码单元的最优划分形状。通过考虑图像特性可变地调整最大编码单元的尺寸并通过将最大编码单元划分成具有不同深度的子编码单元来对图像进行编码，可有效地对具有各种分辨率的图像进行编码。

[0055] 图2是根据本发明实施例的图像解码设备200的框图。

[0056] 参照图2，图像解码设备200包括图像数据获取器210、编码信息提取器 220和图像数据解码器230。

[0057] 图像数据获取器210解析图像解码设备200接收的比特流，获得每个最大编码单元的图像数据并将每个最大编码单元的图像数据输出到图像数据解码器230。图像数据获取器210可从当前帧或当前条带的头提取关于当前帧或当前条带的最大编码单元的信息。换句话说，比特流被划分为最大编码单元，以允许图像数据解码器230对每个最大编码单元的图像数据进行解码。

[0058] 编码信息提取器220解析图像解码设备200接收的比特流，并从当前帧的头提取关于最大编码单元、最大深度、最大编码单元的划分形状和子编码单元的编码模式的信息。关于划分形状和编码模式的信息被输出到图像数据解码器230。

[0059] 关于最大编码单元的划分形状的信息可包括关于包括在最大编码单元中并且根据深度具有不同尺寸的子编码单元的信息。如上关于图1所述，关于划分形状的信息可以是编码的表示每个编码单元是否被划分的信息(例如，标志信息)。关于编码模式的信息可包括例如关于预测单元的信息、关于预测模式的信息、关于子编码单元的变换单元的信息。

[0060] 图像数据解码器230通过基于编码信息提取器220提取的信息对每个最大编码单元的图像数据进行解码，来恢复当前帧。

[0061] 图像数据解码器230可基于关于最大编码单元的划分形状的信息对包括在最大编码单元中的子编码单元进行解码。解码处理可包括帧间预测处理(包括帧内预测和运动补偿)以及逆正交变换处理。

[0062] 图像数据解码器230可基于关于预测单元的信息和关于子编码单元的预测模式的信息执行帧内预测和帧间预测，以预测子编码单元。此外，图像数据解码器230可基于关于子编码单元的变换单元的信息对每个子编码单元执行逆正交变换。

[0063] 图3是示出根据本发明实施例的分层编码单元的示图。

[0064] 参照图3，分层编码单元可包括宽度×高度为64×64、32×32、16×16、8×8和4×4的编码单元。除了正方形的编码单元，还可存在宽度×高度为 64×32、32×64、32×16、16×32、16×8、8×16、8×4和4×8的编码单元。

[0065] 在图3中，对于分辨率为1920×1080的图像数据310，最大编码单元的尺寸被设置为64×64，最大深度被设置为2。

[0066] 对于分辨率为1920×1080的另一图像数据320，最大编码单元的尺寸被设置为64×64，最大深度被设置为4。对于分辨率为352×288的图像数据 330，最大编码单元的尺寸被设置为16×16，最大深度被设置为2。

[0067] 如果分辨率高或者数据量大，则为了提高压缩率并且为了准确反映图像特性，最大编码尺寸可以相对大。因此，对于分辨率高于图像数据330的分辨率的图像数据310和320，最大编码单元的尺寸可以被选择为64×64。

[0068] 最大深度表示分层编码单元的总层数。因为图像数据310的最大深度为 2，所以图像数据310的编码单元315可包括具有长轴大小64的最大编码单元以及随着深度加深而具有长轴大小32和16的子编码单元。

[0069] 同时，因为图像数据330的最大深度为2，所以图像数据330的编码单元335可包括具有长轴大小16的最大编码单元以及随着深度加深而具有长轴大小8和4的子编码单元。

[0070] 因为图像数据320的最大深度为4，所以图像数据320的编码单元325 可包括具有长轴大小64的最大编码单元以及随着深度加深而具有长轴大小 32、16、8和4的子编码单元。这样，因为随着深度加深基于小的子编码单元对图像进行编码，所以可适当地对包括详细场景的图像进行编码。

[0071] 图4是根据本发明实施例的基于编码单元的图像编码器400的框图。

[0072] 帧内预测器410对当前帧405中的帧内模式的预测单元执行帧内预测，运动估计器420和运动补偿器425通过使用当前帧405以及参考帧495分别对帧间模式的预测单元执行帧间预测和运动补偿。

[0073] 基于从帧内预测器410、运动估计器420和运动补偿器425输出的预测单元来产生残差值，产生的残差值经过正交变换器430和量化器440，从而被输出为量化的变换系数。

[0074] 量化的变换系数经过反量化器460和逆频率变换器470，从而被恢复为残差值，恢复的残差值通过去块器480和环路滤波单元490被后处理，从而被输出为参考帧495。量化的变换系数还可经过熵编码器450，从而被输出为比特流455。

[0075] 为了基于根据本发明实施例的图像编码方法对图像进行编码，图像编码器400的所有组件(即，帧内预测器410、运动估计器420、运动补偿器425、正交变换器430、量化器440、熵编码器450、反量化器460、逆频率变换器 470、去块器480和环路滤波单元490)基于最大编码单元、根据深度的子编码单元、预测单元和变换单元执行图像编码处理。

[0076] 图5是根据本发明实施例的基于编码单元的图像解码器500的框图。

[0077] 比特流505经过解析器510，从而被解析为将被解码的编码的图像数据以及对编码的图像数据进行解码所需要的编码信息。编码的图像数据经过熵解码器520和反量化器530，从而被输出为反量化的数据，并经过逆频率变换器540，从而被恢复为残差值。残差值与帧内预测器550执行的帧内预测的结果以及运动补偿器560执行的运动补偿的结果相加，从而被恢复为编码单元。恢复的编码单元经过去块器570和环路滤波单元580，从而被用于对下一编码单元进行解码或预测下一帧。

[0078] 为了基于根据本发明实施例的图像解码方法对图像进行解码，图像解码器500的所有组件(即，解析器510、熵解码器520、反量化器530、逆频率变换器540、帧内预测器550、运动补偿器560、去块器570和环路滤波单元 580)基于最大编码单元、根据深度的子编码单元、预测单元和变换单元执行图像解码处理。

[0079] 具体地讲，帧内预测器550和运动补偿器560考虑最大编码单元和深度来确定子编码单元中的预测单元和预测模式，逆频率变换器540考虑变换单元的尺寸来执行逆正交变换。

[0080] 图6是示出根据本发明实施例的最大编码单元、子编码单元和预测单元的示图。

[0081] 图1中示出的图像编码设备100和图2中示出的图像解码设备200使用分层编码单元，以考虑图像特性来执行编码和解码。可根据图像特性自适应地设置最大编码单元和最大深度，或者可根据用户的请求不同地设置最大编码单元和最大深度。

[0082] 图6示出编码单元的分层结构600，在该编码单元中，最大编码单元610 的高度和宽度均为64，最大深度为4。深度根据编码单元的分层结构600的纵轴加深，子编码单元620、630、640和650的宽度和高度随着深度加深而减小。此外，沿着编码单元的分层结构600的横轴，示出最大编码单元610 以及子编码单元620、630、640和650的预测单元。

[0083] 最大编码单元610具有深度0，并且具有尺寸(即，宽度×高度)64×64。深度沿着纵轴加深，并且存在尺寸为32×32、深度为1的子编码单元620、尺寸为16×16、深度为2的子编码单元630、尺寸为8×8、深度为3的子编码单元640以及尺寸为4×4、深度为4的子编码单元650。尺寸为4×4、深度为4的子编码单元650是最小编码单元。

[0084] 参照图6，沿着横轴示出根据深度的预测单元的示例。也就是说，深度为0的最大编码单元610可包括尺寸为64×64的预测单元610、尺寸为64×32 的预测单元612、尺寸为32×64的预测单元614、尺寸为32×32的预测单元 616，这些预测单元的尺寸等于或小于编码单元610的尺寸(即，64×64)。

[0085] 深度为1、尺寸为32×32的编码单元620可包括尺寸为32×32的预测单元620、尺寸为32×16的预测单元622、尺寸为16×32的预测单元624、尺寸为16×16的预测单元626，这

些预测单元的尺寸等于或小于编码单元620 的尺寸(即,32×32)。

[0086] 深度为2、尺寸为16×16的编码单元630可包括尺寸为16×16的预测单元630、尺寸为16×8的预测单元632、尺寸为8×16的预测单元634、尺寸为8×8的预测单元636,这些预测单元的尺寸等于或小于编码单元630的尺寸(即,16×16)。

[0087] 深度为3、尺寸为8×8的编码单元640可包括尺寸为8×8的预测单元 640、尺寸为8×4的预测单元642、尺寸为4×8的预测单元644、尺寸为4×4 的预测单元646,这些预测单元的尺寸等于或小于编码单元640的尺寸(即, 8×8)。

[0088] 最后,深度为4、尺寸为4×4的编码单元650具有最大深度,并包括尺寸为4×4的预测单元650。然而,具有最大深度的编码单元650不是必然需要包括尺寸等于编码单元的尺寸的预测单元,而是可以像其它编码单元610、 620、630和640那样,为了预测被划分为尺寸小于编码单元的尺寸的预测单元。

[0089] 图7是示出根据本发明实施例的编码单元和变换单元的示图。

[0090] 图1中示出的图像编码设备100和图2中示出的图像解码设备200对最大编码单元或从最大编码单元划分且尺寸小于最大编码单元的尺寸的子编码单元进行编码。可选择用于在编码处理中执行正交变换的变换单元的尺寸以实现最高的压缩率,而不管编码单元和预测单元。例如,如果当前编码单元 710具有64×64的尺寸,则可通过使用具有32×32的尺寸的变换单元720 来执行正交变换。此外,可设置尺寸大于编码单元的尺寸的变换单元。

[0091] 图8A和图8B是示出根据本发明实施例的编码单元、预测单元和变换单元的划分形状的示图。

[0092] 图8A示出根据本发明实施例的编码单元和预测单元。

[0093] 图8A的左侧示出图1中示出的图像编码设备100选择以对最大编码单元 810进行编码的划分形状。图像编码设备100将最大编码单元810划分为各种形状并对各种形状进行编码,基于R-D成本比较编码的划分形状,并选择最优的划分形状。如果最优的划分形状对应于最大编码单元810,则可直接对最大编码单元810进行编码,而无需如图8A所示划分最大编码单元810。

[0094] 参照图8A的左侧,深度为0的最大编码单元810被划分并编码为深度等于或大于1的子编码单元。最大编码单元810被划分为四个深度为1的子编码单元,然后所有或部分深度为1的子编码单元被划分为深度为2的子编码单元。

[0095] 在深度为1的子编码单元中,右上方子编码单元和左下方子编码单元被划分为深度等于或大于2的子编码单元。部分深度等于或大于2的子编码单元可被划分为深度等于或大于3的子编码单元。

[0096] 图8A的右侧示出关于最大编码单元810的预测单元860的划分形状。

[0097] 参照图8A的右侧,关于最大编码单元810的预测单元860可与最大编码单元810不同地被划分。换句话说,关于每个子编码单元的预测单元可小于子编码单元。

[0098] 例如,在深度为1的子编码单元中,关于右下方子编码单元854的预测单元可以小于子编码单元854。在深度为2的子编码单元814、816、818、828、 850和852中,关于部分子编码单元814、816、850和852的预测单元可小于子编码单元814、816、850和852。此外,关于深度为3的子编码单元822、 832和848的预测单元可小于子编码单元822、832和848。预测单元可具有通过将每个子编码单元沿高度或宽度方向划分为两个单元而获得的形状,或者具有

通过将每个子编码单元沿高度和宽度方向划分为四个单元而获得的形状。

[0099] 图8B示出根据本发明实施例的预测单元和变换单元。

[0100] 图8B的左侧示出关于最大编码单元810的预测单元860的划分形状，图 8B的右侧示出关于最大编码单元810的变换单元870的划分形状。

[0101] 参照图8B的右侧，变换单元870可以与预测单元860不同地被划分。

[0102] 例如，虽然关于深度为1的编码单元854的预测单元被选择为通过将编码单元854的高度二等分获得的形状，但是关于编码单元854的变换单元可以被选择为尺寸等于编码单元854的尺寸。同样地，虽然关于深度为2的编码单元814和850的预测单元被选择为通过将编码单元814和850的高度二等分获得的形状，但是关于编码单元814和850的变换单元可以被选择为尺寸等于编码单元814和850的尺寸。

[0103] 变换单元可以被选择为尺寸小于预测单元的尺寸。例如，如果关于深度为2的编码单元852的预测单元被选择为通过将编码单元852的宽度二等分获得的形状，则变换单元可以被选择为尺寸小于预测单元的尺寸且通过将编码单元852的高度和宽度二等分获得形状。还可设置尺寸为2×2的最小变换单元。还可不管编码单元的尺寸来设置变换单元(例如，尺寸大于编码单元的尺寸)。

[0104] 现在将详细描述图4中示出的图像编码设备400的熵编码器450执行的熵编码处理和图5中示出的图像解码设备500的熵解码器520执行的熵解码处理。

[0105] 如上关于图4和图5所述，图像编码设备400和图像解码设备500对最大编码单元或从最大编码单元划分的尺寸小于最大编码单元的尺寸的子编码单元进行编码。可选择用于在编码处理中执行正交变换的变换单元的尺寸以实现最高压缩率，而不管编码单元和预测单元。例如，如果当前编码单元具有64×64的尺寸，则可通过使用具有32×32的尺寸的变换单元来执行正交变换。此外，可设置尺寸大于编码单元的尺寸的变换单元。在传统的编码处理(例如，H.264)中，对基于具有相对小尺寸(例如4×4)的变换单元变换和量化的残差数据进行熵编码。然而，根据本发明实施例，因为将被熵编码的变换单元(以下称为“变换块”)可具有相对大的尺寸16×16、32×32、 64×64或128×128以及4×4或8×8，所以表示在具有非零值的有效变换系数之间具有值0的连续系数的数量的行程的长度可增加，需要对大行程值进行适当的编码。此外，根据传统技术，为了对包括在变换块中的系数的信息进行编码，作为表示每个有效变换系数是否是最后有效变换系数的语法元素的last_significant_coeff_flag连同表示具有非零值的有效变换系数的位置的有效映射一起被熵编码。然而，如果变换块连同有效映射和 last_significant_coeff_flag一起被熵编码，则应该在熵解码处理中确定每个有效变换系数是否是最后有效变换系数。因此，根据传统技术，无法从接收的比特流容易地、直接地识别表示全部有效变换系数的数据。因此，根据本发明实施例，提供了对关于具有大尺寸的变换块中的最后有效变换系数的位置的信息进行有效的熵编码和熵解码的方法。

[0106] 图9是根据本发明实施例的对变换系数进行熵编码的方法的流程图。

[0107] 参照图9，在操作910，熵编码器450根据特定扫描顺序，从具有非零值且包括在具有特定尺寸的变换块中的有效变换系数中确定最后有效变换系数的位置。更具体地讲，如果包括通过执行变换和量化处理而获得的变换系数的变换块被输入，则熵编码器450根据特定扫描顺序(例如，Z字形扫描顺) 确定包括在变换块中的有效变换系数，并确定最后扫

描到的最后有效变换系数的位置。

[0108] 在操作920,熵编码器450通过使用变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置对关于最后有效变换系数的位置的信息进行编码。如果最后有效变换系数位于从变换块的左上位置开始的横轴方向的第x 位置(x是等于或大于0的整数)和纵轴方向的第y位置(y是等于或大于0 的整数),则熵编码器450对表示最后有效变换系数的位置的值x和y进行编码。根据传统技术,对表示每个有效变换系数是否是最后有效变换系数的last_significant_coeff_flag进行编码。然而,根据本发明实施例,仅对表示变换块中的最后有效变换系数的位置的坐标信息进行编码。可通过使用上下文自适应二进制算术编码(CABAC)方法或可变长度编码(VLC)方法对关于最后有效变换系数的位置的信息进行编码。以下将描述通过使用CABAC或 VLC对关于最后有效变换系数的位置的信息进行编码的方法。

[0109] 在对关于最后有效变换系数的位置的信息进行编码之后,熵编码器450 对位于变换块中的每个有效变换系数的级别信息进行编码。熵编码器450可对每个有效变换系数的符号值和绝对值进行编码,作为级别信息。

[0110] 图10是用于描述根据本发明实施例的对变换系数进行熵编码的处理的参考示图。

[0111] 参照图10,熵编码器450根据Z字形扫描顺序扫描变换块1000中的变换系数。假设图10中所有空的空间表示具有值“0”的变换系数。在图10中,最后有效变换系数是具有值“-1”的变换系数1010。如图10所示,具有值“-1”的最后有效变换系数1010位于从左上变换系数开始的横轴方向的第5 位置和纵轴方向的第5位置。因此,熵编码器450将值x=5和y=5编码为关于最后有效变换系数1010的位置的信息。

[0112] 图11是根据本发明实施例的熵编码设备1100的框图。

[0113] 参照图11,熵编码设备1100包括开关1110、上下文自适应二进制算术编码器(CABAC)1120和可变长度编码器(VLC)1130。

[0114] 如上关于图9所述,如果最后有效变换系数的位置被确定,则通过使用 CABAC或VLC来对关于最后有效变换系数的位置的信息进行编码。开关1110 控制关于最后有效变换系数的位置的信息被输出到CABAC 1120或VLC 1130,其中,以条带、画面或画面组为单位对所述信息进行编码。还可通过比较通过使用CABAC或VLC获得的R-D成本来确定是使用CABAC还是使用VLC来对所述信息进行编码。

[0115] 现在将描述通过使用CABAC对关于最后有效变换系数的位置的信息进行编码的方法。

[0116] 图12是根据本发明实施例的CABAC装置1200的框图。

[0117] 参照图12,CABAC装置1200主要包括二进制化编码器(binarizer)1210、上下文建模器1220和二进制算术编码器1230。此外,二进制算术编码器1230 包括常规编码引擎1232和旁路编码引擎1234。

[0118] 二进制化编码器1210将表示最后有效变换系数的位置的变换块中的横轴方向位置x和纵轴方向位置y转换为二进制值并输出bin串。bin表示bin 串的每个比特。二进制化关于最后有效变换系数的位置的信息的方法包括各种二进制化方法,诸如一元二进制化、截断一元二进制化、级联一元/k阶指数Golomb二进制化和固定长度二进制化。例如,在图10中,如果最后有效变换系数的位置被表示为X=5,Y=5,则通过使用截断一元二进制化可将

值X 和Y二进制化为X＝0000001和Y＝000001。

[0119] 通过二进制化编码器1210映射到二进制值的关于最后有效变换系数的位置的信息被输入到上下文建模器1220。上下文建模器1220基于输入的bin 值或先前编码的语法元素确定对当前输入的bin进行编码所需的概率模型 (即,上下文)。具体地讲,根据本发明实施例,可根据最后有效变换系数的位置选择一个先前确定的上下文。

[0120] 常规编码引擎1232通过基于由上下文建模器1220确定的概率模型对输入的bin值进行算术编码来产生比特流。

[0121] 旁路编码引擎1234是用于输出输入值而不压缩输入值的引擎,并且对诸如脉冲编码调制(PCM)数据的数据进行编码。

[0122] 图13是用于描述根据本发明实施例的选择用于对关于最后有效变换系数的位置的信息进行编码的上下文的处理的参考示图。

[0123] 上下文建模器1220根据最后有效变换系数的位置选择多个先前编写的上下文之一。所述多个上下文根据最后有效变换系数的位置将二进制信号的“0”和“1”分类为最大可能符号(MPS)和最小可能符号(LPS),并设置 MPS和LPS的概率值。“0”和“1”中的哪个符号被设置为MPS或LPS,以及如何设置MPS和LPS的概率值,这可根据需要来设计。图13示出当根据Z字形扫描顺序扫描变换块中的变换系数时,根据除了与DC相应的左上位置之外的最后有效变换系数的位置选择的上下文。参照图13,上下文建模器1220 包括关于4×4变换块的具有索引0、1、2的三个上下文,并且根据除了左上位置之外的4×4变换块中的位置之中的最后有效变换系数的位置来选择具有索引0、1、2的三个上下文之一。也就是说,如果最后有效变换系数位于 4×4变换块中的(1,0),则选择具有索引0的上下文,如果最后有效变换系数位于(0,1),则选择具有索引1的上下文,如果最后有效变换系数位于 (1,1),则选择具有索引2的上下文。同样,上下文建模器1220包括关于 8×8变换块的具有索引0至6的七个上下文,并且根据除了左上位置之外的 8×8变换块中的位置之中的最后有效变换系数的位置来选择具有索引0至6 的七个上下文之一。相似地,上下文建模器1220包括关于16×16变换块的具有索引0至11的十二个上下文,并且根据除了左上位置之外的16×16变换块中的位置之中的最后有效变换系数的位置来选择具有索引0至11的十二个上下文之一。

[0124] 常规编码引擎1232通过基于由上下文建模器1220确定上下文对表示最后有效变换系数的位置的输入的bin值进行算术编码来产生比特流。例如,假设最后有效变换系数的横轴方向位置(即,X)具有值3,二进制化编码器 1210通过对值3二进制化产生bin串“010”。还假设,基于上下文建模器1220 根据最后有效变换系数的位置选择的上下文,MPS是具有概率值0.8的“0”, LPS是具有概率值0.2的“1”。常规编码引擎1232通过根据作为用于形成bin 串“010”的初始bin的“0”的概率值分割区间[0,1]来将区间[0,1]更新为区间[0,0.8],并且根据作为下一bin的“1”的概率值将区间[0,0.8]更新为新的区间[0.64,0.8]。此外,常规编码引擎1232根据作为最后bin的“0”的概率值将区间[0.64,0.8]更新为区间[0.64,0.768]。常规编码引擎1232将通过从“0.11”排除初始数字而获得的“11”输出为与表示最后有效变换系数的横轴方向位置的值3相应的码字,其中,“0.11”是包括在区间 [0.64,0.768]中的实数0.75的二进制数。同样,常规编码引擎1232对表示最后有效变换系数的纵轴方向位置的坐标值Y二进制化,并通过根据选择的上下文对二进制化的坐标值Y进行编码来产生比特流。

[0125] 除上述关于最后有效变换系数的位置的信息之外,图11中示出的CABAC 1120对表示具有非零值的有效变换系数是否存在于变换块中的 coded_block_flag以及表示变换块中的每个有效变换系数的位置的 significant_coeff_flag进行编码。coded_block_flag和 significant_coeff_flag的编码处理可以与传统的H.264的编码处理相同。例如,参照示出与图10相应的有效映射1400的图14,可通过如果具有第i 扫描顺序的变换系数是有效变换系数则将significant_coeff_flag[i]设置为1,如果具有第i扫描顺序的变换系数是0则将significant_coeff_flag[i] 设置为0,来表示有效映射1400。可通过使用如H.264中的十五个概率模型来对有效映射1400进行编码。

[0126] 在对关于最后有效变换系数的位置的信息进行编码之后,熵编码器450 对位于变换块中的每个有效变换系数的级别信息进行编码。

[0127] 图15是用于描述在图10中示出的对包括在变换块1000中的有效变换系数的级别值进行编码的处理的参考示图。

[0128] 参照图10和图15,熵编码器450根据Z字形扫描顺序,从左上变换系数到最后有效变换系数扫描图10中示出的变换系数,并获得如图15所示的一维对齐的变换系数。

[0129] 熵编码器450通过使用如H.264中的表示在有效变换系数之间具有值0 的连续系数的数量的行程以及表示每个有效变换系数的值的级别来对一维对齐的变换系数进行编码。更详细地讲,熵编码器450以与扫描顺序相反的顺序(即,图15的从右侧到左侧的方向)确定行程和级别,并通过使用特定 VLC表对行程和级别进行编码。

[0130] 图16是示出根据本发明实施例使用的多个VLC表VLC0至VLC8的示例的示图。熵编码器450可根据最后有效变换系数的位置选择VLC表VLC0至VLC8 之一,并可通过使用选择的VLC表对行程和级别进行编码。例如,如图15所示,通过使用VLC表,从具有值“-1”的最后有效变换系数1510开始,对表示在有效变换系数之间具有值0的连续变换系数的数量的行程1530、1540、1550和1560执行VLC。

[0131] 因为根据本发明的变换块可具有等于或大于16×16的大尺寸,所以行程值可能增大。例如,如果VLC表仅包括从0至63的行程值并且行程值大于 63,则不可通过使用VLC表对该值进行编码。因此,根据本发明实施例,考虑VLC表可用的最大行程值,如果变换块具有大于最大行程值的行程值,则熵编码器450对等于或小于最大行程值的行程值进行编码,然后对其他行程值进行编码。例如,如果最大行程值是63,并且将被编码的行程值是70,则行程值70被分割为63和7,并且行程值63和7被分别编码为行程信息。

[0132] 根据本发明另一实施例,除了上述CABAC以外,还可通过使用VLC对最后有效变换系数的位置(x,y)进行编码。也就是说,熵编码器450可参考根据值x和y先前编写的VLC表来对值x和y执行VLC。

[0133] 图17是用于描述根据本发明另一实施例的对变换系数进行熵编码的方法的参考示图。

[0134] 根据当前实施例,熵编码器450将变换块划分为具有特定尺寸的子块,并将关于包括最后有效变换系数的子块中的最后有效变换系数的位置的信息连同该子块的索引一起进行编码。在图17中,假设(Sa,b)表示子块a中的具有第b扫描索引的变换系数。参照图17,如果假设最后有效变换系数是子块1 1771中的(S1,12),则熵编码器450将表示子块1 1771中的最后有效变换系数(S1,12)的位置的(2,2)(作为关于子块中的最后有效变换系数的位

置的信息)连同表示子块1 1771的特定索引一起编码。

[0135] 返回参照图10,根据本发明另一实施例,通过使用变换块1000中的最后有效变换系数1010(以下称为“第一最后有效变换系数”)以及在第一最后有效变换系数1010之前的有效变换系数1020(以下称为“第二最后有效变换系数”)对变换系数进行编码。更详细地讲,如上所述,熵编码器450对表示第二最后有效变换系数1020的(3,4)进行编码。然后,熵编码器450 对第一最后有效变换系数1010和第二最后有效变换系数1020之间的行程值进行编码。如上所述,如果第二最后有效变换系数1020的位置已知,则可通过将第一最后有效变换系数1010和第二最后有效变换系数1020之间的行程值加到第二最后有效变换系数1020的位置来获得第一最后有效变换系数 1010的位置。

[0136] 图18A和图18B是用于描述根据本发明另一实施例的对变换系数进行熵编码的方法的参考示图。

[0137] 熵编码器450可根据最后有效变换系数以哪种顺序被首先扫描(即,根据最后有效变换系数是接近变换块的左上位置还是接近变换块的右下位置),来选择与Z字形扫描顺序和反Z字形扫描顺序之一相应的扫描方向,并可将关于最后有效变换系数的位置的信息连同表示选择的扫描方向的索引一起编码。例如,如图18A所示,如果标号1812表示中心位置,则最后有效变换系数1811更接近左上位置。在这种情况下,熵编码器450可将关于最后有效变换系数1811的位置的信息连同表示从左上位置开始的扫描方向的索引(正向扫描索引)一起编码。此外,如图18B所示,如果标号1822表示中心位置并且最后有效变换系数1821更接近右下位置,则熵编码器450可将关于最后有效变换系数1821的位置的信息连同表示从右下位置开始的扫描方向的索引 (反向扫描索引)一起编码。

[0138] 根据本发明另一实施例,为了对行程值进行编码,可设置特定全局行程,并且当行程值除以全局行程时,可通过使用商和余数来表示行程值(行程细化)。例如,假设行程值是78并且全局行程具有值16。在这种情况下,当78 除以16时,商是4,余数是14。因此,可通过使用作为全局行程的值的16、作为商的值的4和作为余数的值的14来表示行程值78。熵编码器450可通过使用VLC或CABAC对关于全局行程、商和余数的信息进行编码。如果编码器和解码器预先设置相同的全局行程,则可不额外地对关于全局行程的信息进行编码。

[0139] 此外,根据本发明另一实施例,熵编码器450可根据包括最后有效变换系数的变换块的尺寸,通过应用不同的VLC表来对关于最后有效变换系数的位置的信息进行编码。

[0140] 图19是根据本发明实施例的对变换系数进行熵解码的方法的流程图。

[0141] 参照图19,在操作1910,熵解码器520根据特定扫描顺序,从接收的比特流提取关于具有非零值且包括在变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置的信息。如上关于图9所述,如果最后有效变换系数位于从变换块的左上位置开始的横轴方向的第x位置(x是等于或大于0的整数)和纵轴方向的第y位置(y是等于或大于0的整数),则关于最后有效变换系数的位置的信息包括值x和y。

[0142] 在操作1920,熵解码器520通过对关于横轴方向位置和纵轴方向位置的信息进行解码来确定最后有效变换系数的位置。熵解码器520通过使用上下文自适应二进制算术解码(CABAD)或可变长度解码(VLD)对关于最后有效变换系数的位置的信息进行解码,并确定最后有效变换系数的位置,其中,CABAD 和VLD是熵编码器450执行的编码处理的逆处理。更详细地讲,熵解码器520 可参考特定VLC查找表对最后有效变换系数的横轴方向位置和纵

轴方向位置执行VLD。此外，熵解码器520可根据最后有效变换系数的位置选择多个上下文之一，并可根据选择的上下文对关于最后有效变换系数的位置的信息执行CABAD。

[0143] 在操作1930，熵解码器520通过使用最后有效变换系数的位置对包括在比特流中的行程和级别信息进行解码。如果行程的长度大于特定阈值，则熵解码器520首先对行程信息中的等于该阈值的长度的部分进行解码，然后对行程信息中的剩余部分进行解码。如上关于图16所述，如果对行程值进行编码的阈值被设置为63，则行程值70被分割为行程值63和7，然后行程值63 和7被分别编码。因此，熵解码器520可通过对行程值63和7分别进行解码，然后将它们组合，来对行程值70进行解码。

[0144] 图20是根据本发明实施例的熵解码设备2000的框图。

[0145] 参照图20，熵解码设备2000包括开关2010、CABAD 2020和VLD 2030。

[0146] 开关2010通过使用以条带、画面或画面组为单位设置的变换系数的编码模式信息，将关于编码的变换系数的信息输出到CABAD 2020和VLD 2030之一。

[0147] VLD 2030可参考特定VLC查找表对最后有效变换系数的横轴方向位置和纵轴方向位置执行VLD。此外，CABAD 2020可根据最后有效变换系数的位置选择多个上下文之一，并可根据选择的上下文对关于最后有效变换系数的位置的信息执行CABAD。

[0148] 本发明还可实施为计算机可读记录介质上的计算机可读编码。计算机可读记录介质是任何可存储其后可由计算机系统读取的数据的数据存储装置。计算机可读记录介质的示例包括只读存储器(ROM)、随机存取存储器(RAM)、CD-ROM、磁带、软盘、光学数据存储装置等。计算机可读记录介质还可分布于联网的计算机系统上，从而以分布式方式存储和执行计算机可读代码。

[0149] 虽然已经参照本发明的示例性实施例具体显示和描述了本发明，但是本领域普通技术人员将理解，在不脱离权利要求限定的本发明的精神和范围的情况下，可以在形式和细节上进行各种改变。因此，本发明的范围不是由本发明的详细描述限定，而是由权利要求限定，并且所述范围内的所有不同将被解释为包括在本发明中。

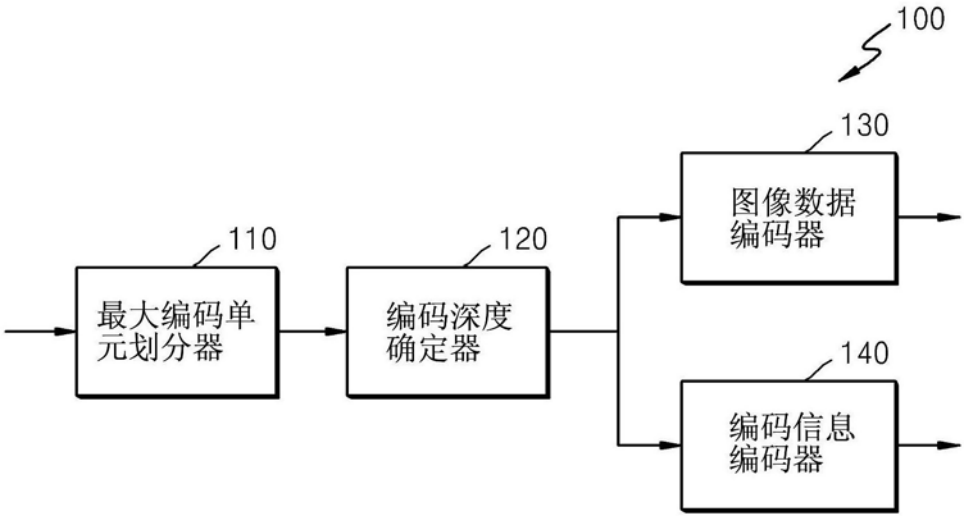

100
130
图像数据
编码器
110
120
最大编码单
元划分器
编码深度
确定器
140
编码信息
编码器

图1

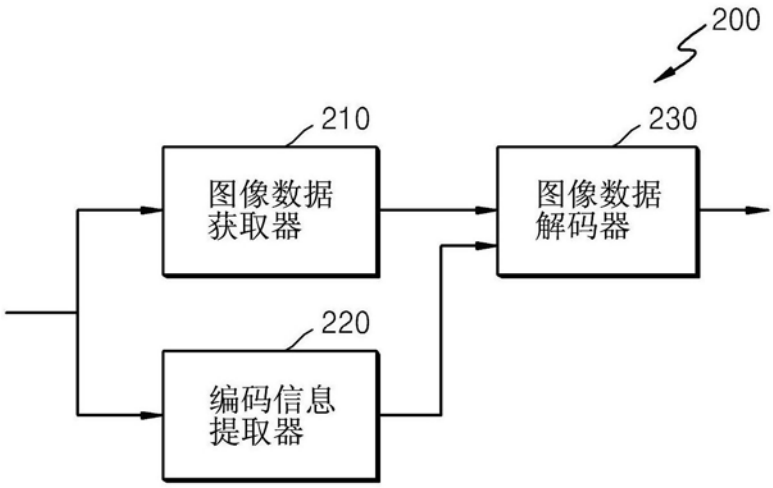

200
210
230
图像数据
获取器
图像数据
解码器
220
编码信息
提取器

图2

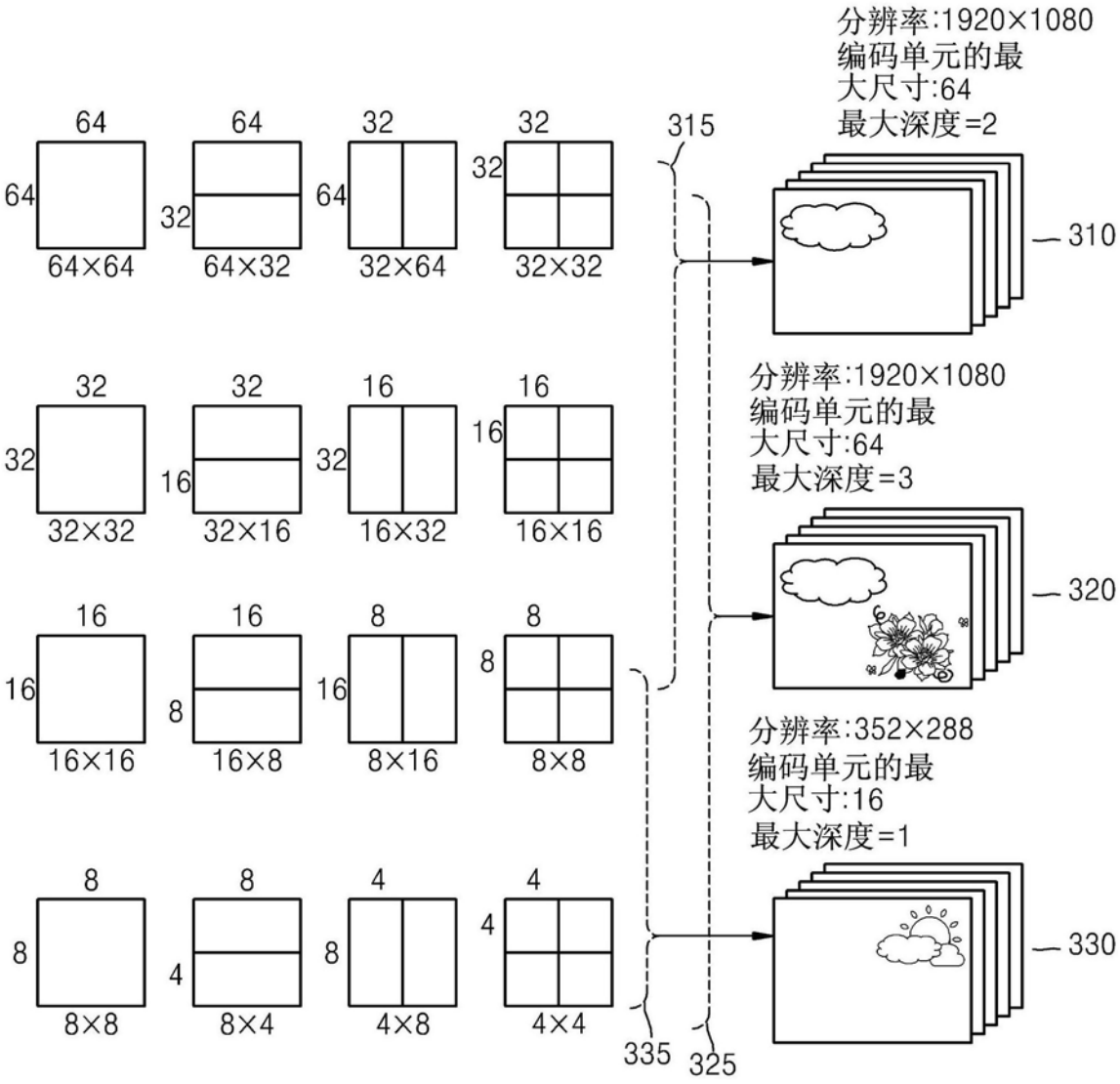

分辨率:1920×1080
编码单元的最
大尺寸:64
最大深度=2
315
310
64×64
64×32
32×64
32×32
分辨率:1920×1080
编码单元的最
大尺寸:64
最大深度=3
320
32×32
32×16
16×32
16×16
分辨率:352×288
编码单元的最
大尺寸:16
最大深度=1
330
16×16
16×8
8×16
8×8
8×8
8×4
4×8
4×4
335
325

图3

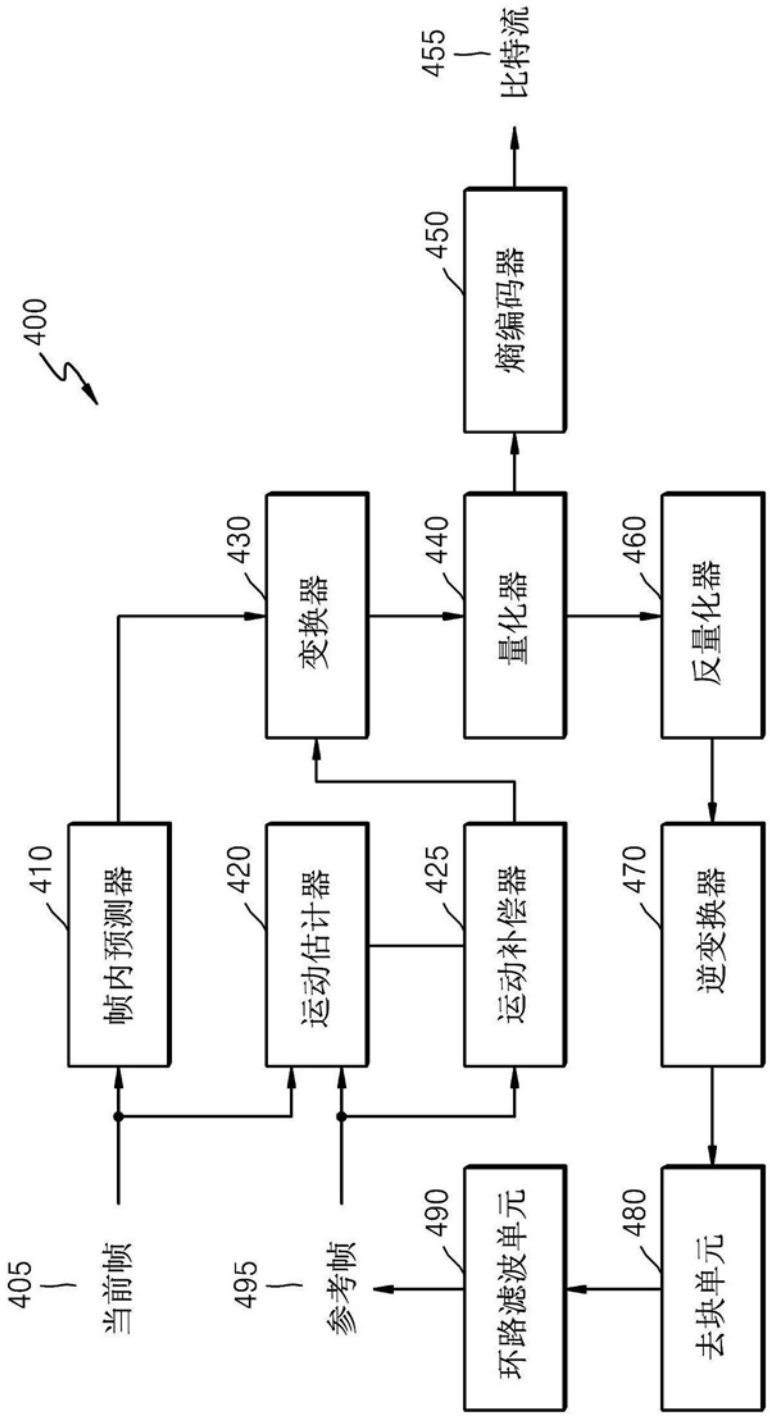

400
405
当前帧
410
帧内预测器
420
运动估计器
425
运动补偿器
430
变换器
440
量化器
450
熵编码器
455
比特流
460
反量化器
470
逆变换器
480
去块单元
490
环路滤波单元
495
参考帧

图4

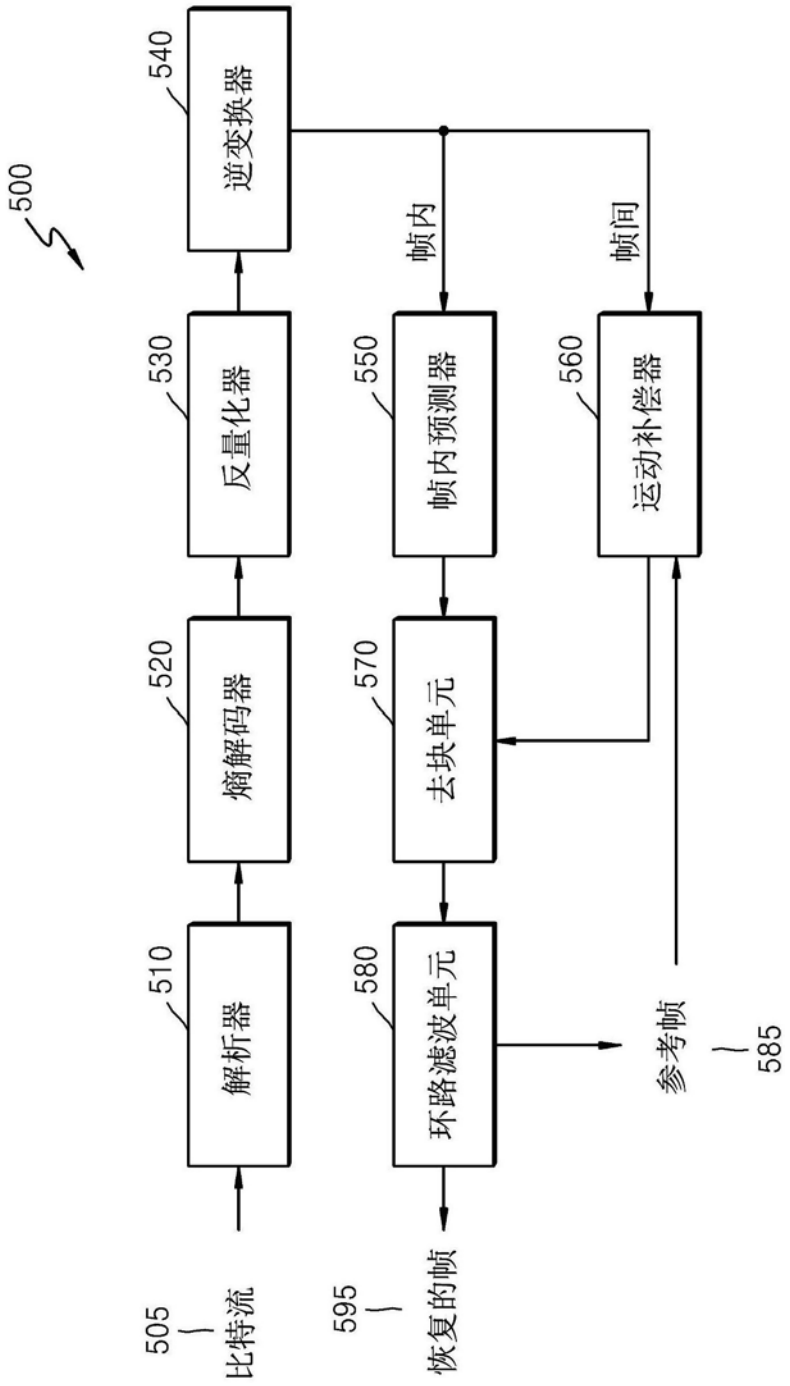

500
505
比特流
510
解析器
520
熵解码器
530
反量化器
540
逆变换器
帧内
帧间
550
帧内预测器
560
运动补偿器
570
去块单元
580
环路滤波单元
595
恢复的帧
参考帧
585

图5

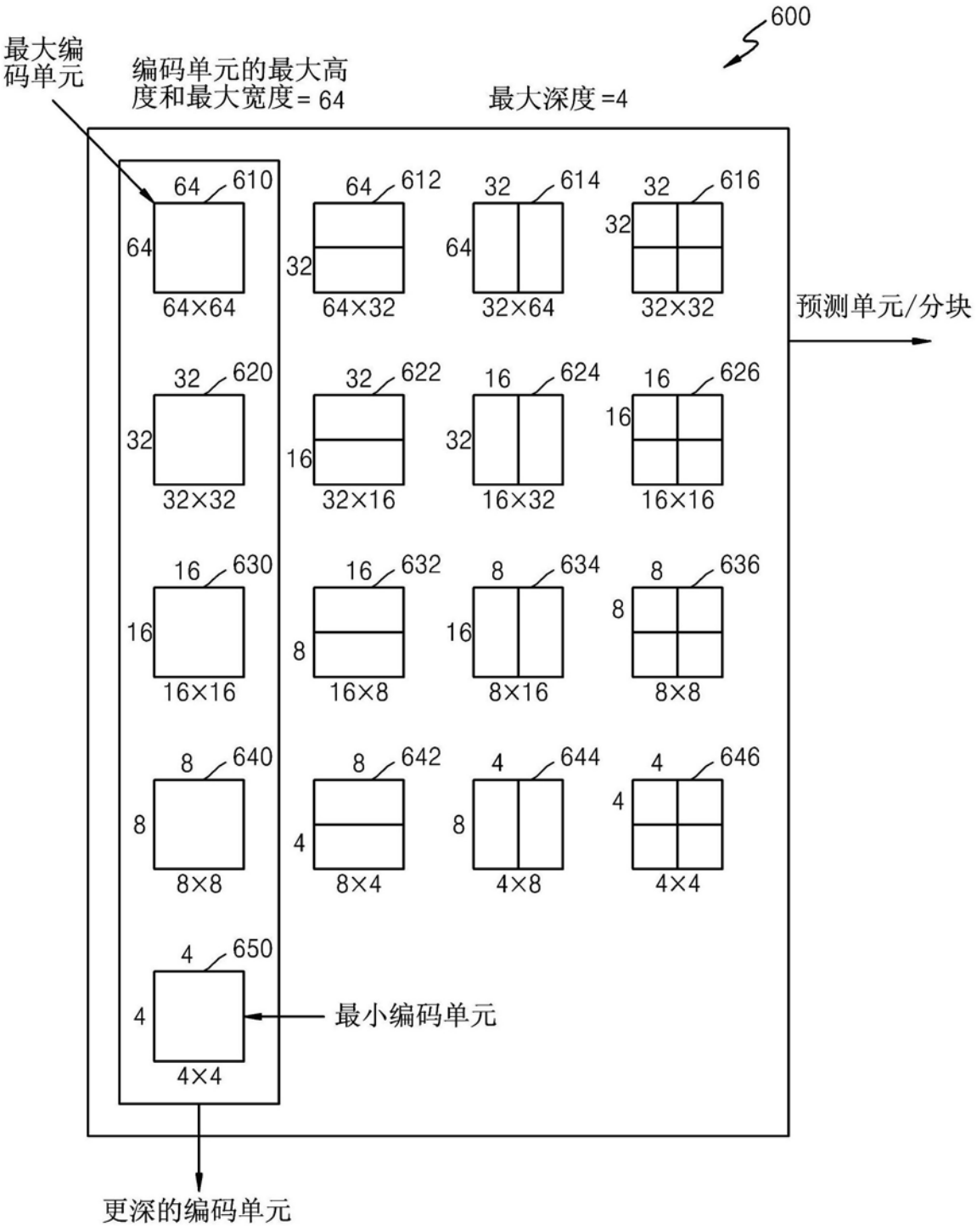

600
最大编码单元
编码单元的最大高度和最大宽度= 64
最大深度 =4
64
610
64
64×64
64
612
32
64×32
32
614
64
32×64
32
616
32
32×32
预测单元/分块
32
620
32
32×32
32
622
16
32×16
16
624
32
16×32
16
626
16
16×16
16
630
16
16×16
16
632
8
16×8
8
634
16
8×16
8
636
8
8×8
8
640
8
8×8
8
642
4
8×4
4
644
8
4×8
4
646
4
4×4
4
650
4
4×4
最小编码单元
更深的编码单元

图6

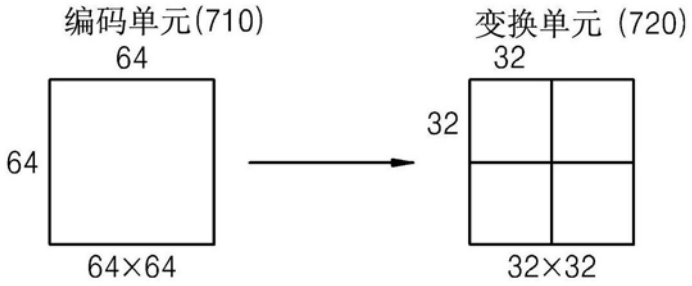


图7

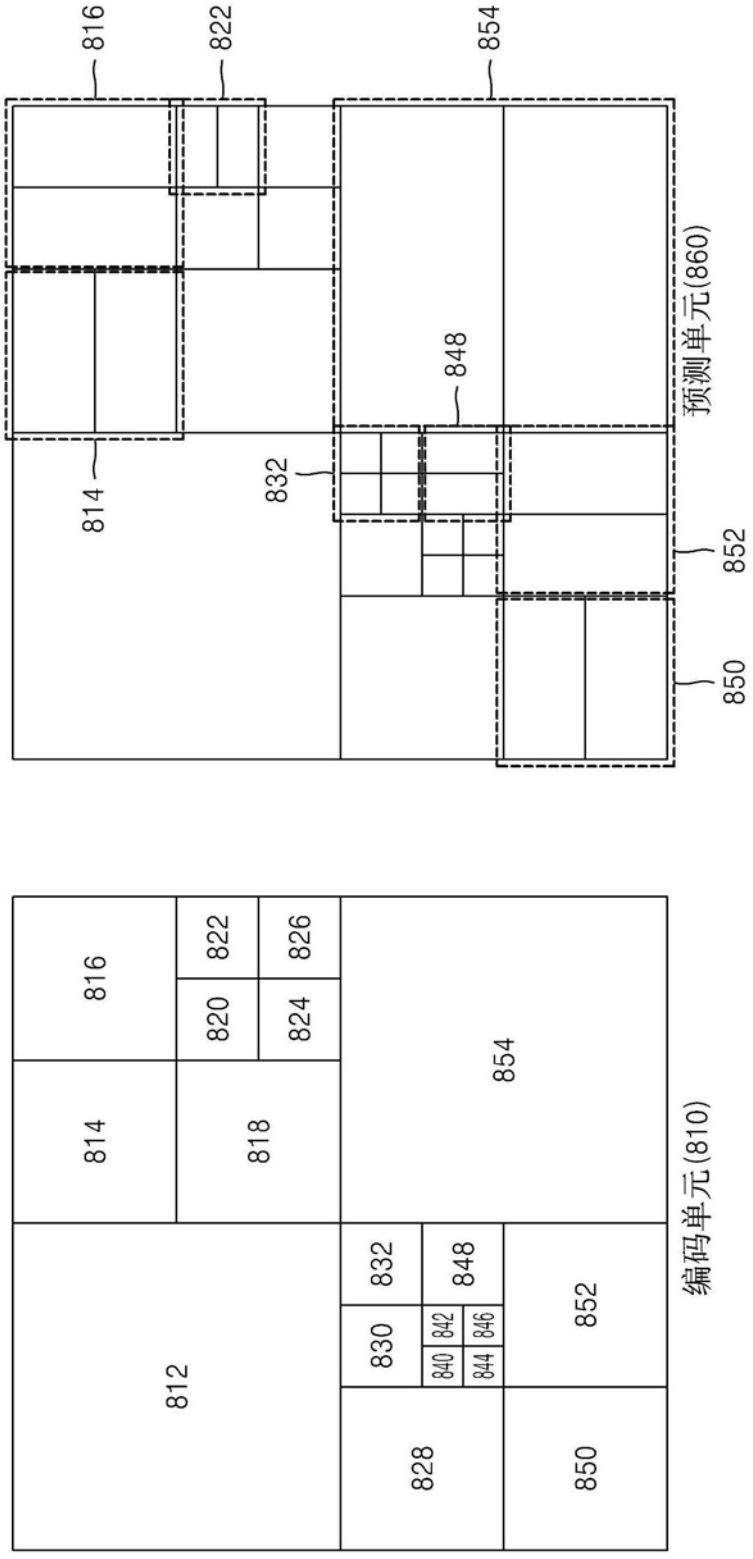

816
822
854
848
预测单元(860)
832
814
852
850
816
822
826
820
824
854
814
818
832
848
830
842
846
840
844
852
812
828
850
编码单元(810)

图8A

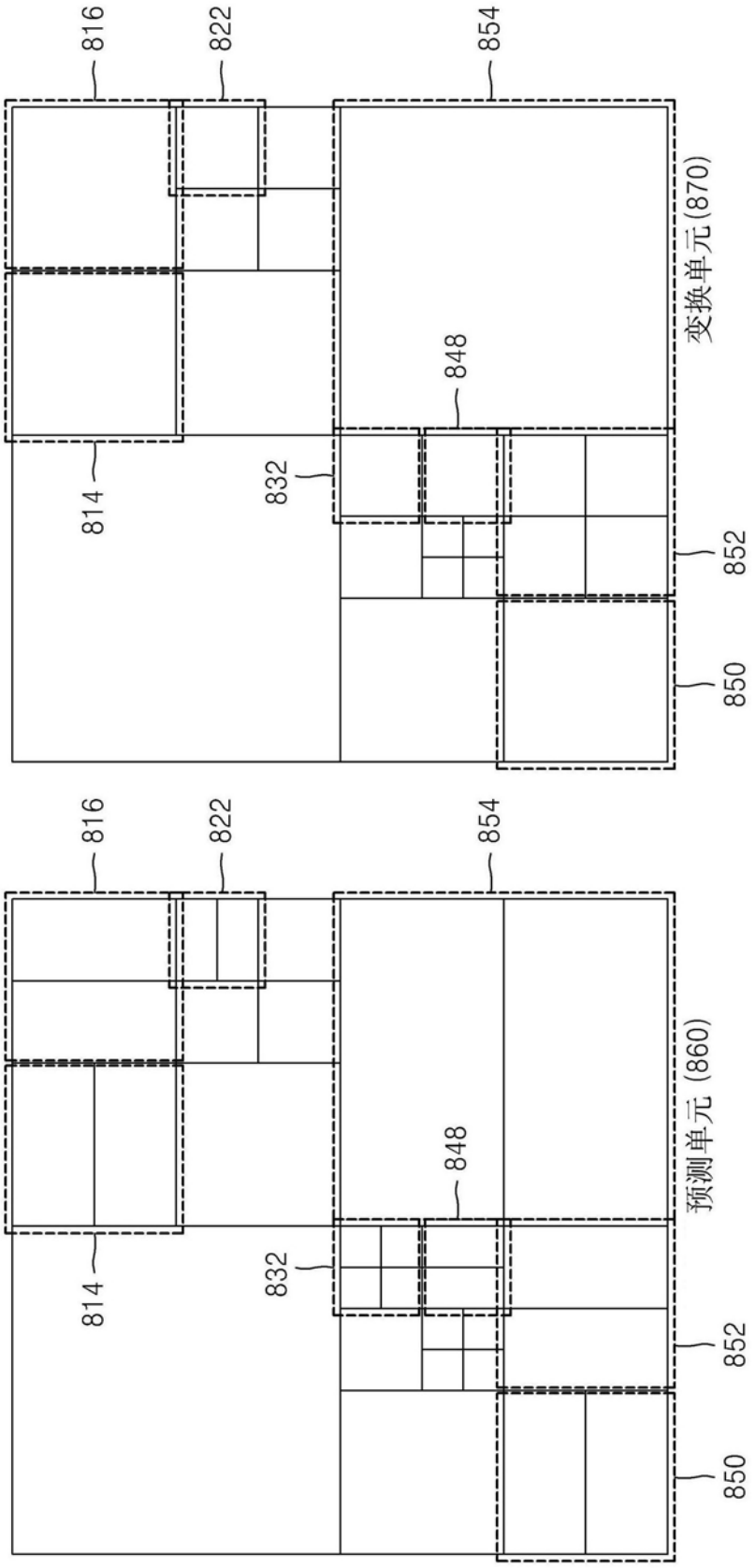

816
822
854
848
变换单元(870)
814
832
852
850
816
822
854
848
预测单元 (860)
814
832
852
850

图8B

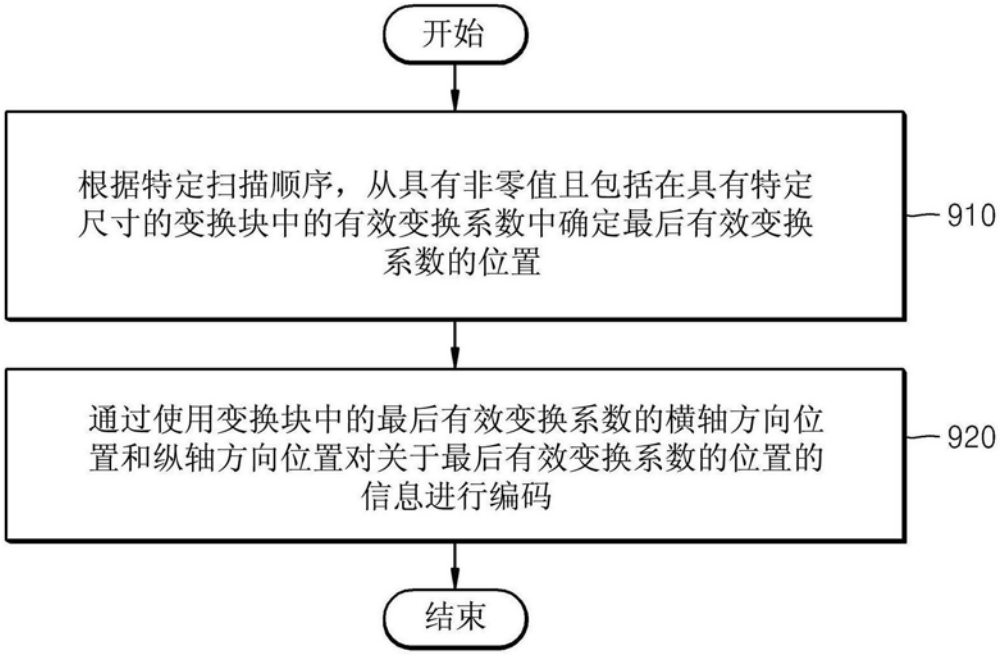

开始
根据特定扫描顺序，从具有非零值且包括在具有特定尺寸的变换块中的有效变换系数中确定最后有效变换系数的位置
910
通过使用变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置对关于最后有效变换系数的位置的信息进行编码
920
结束

图9

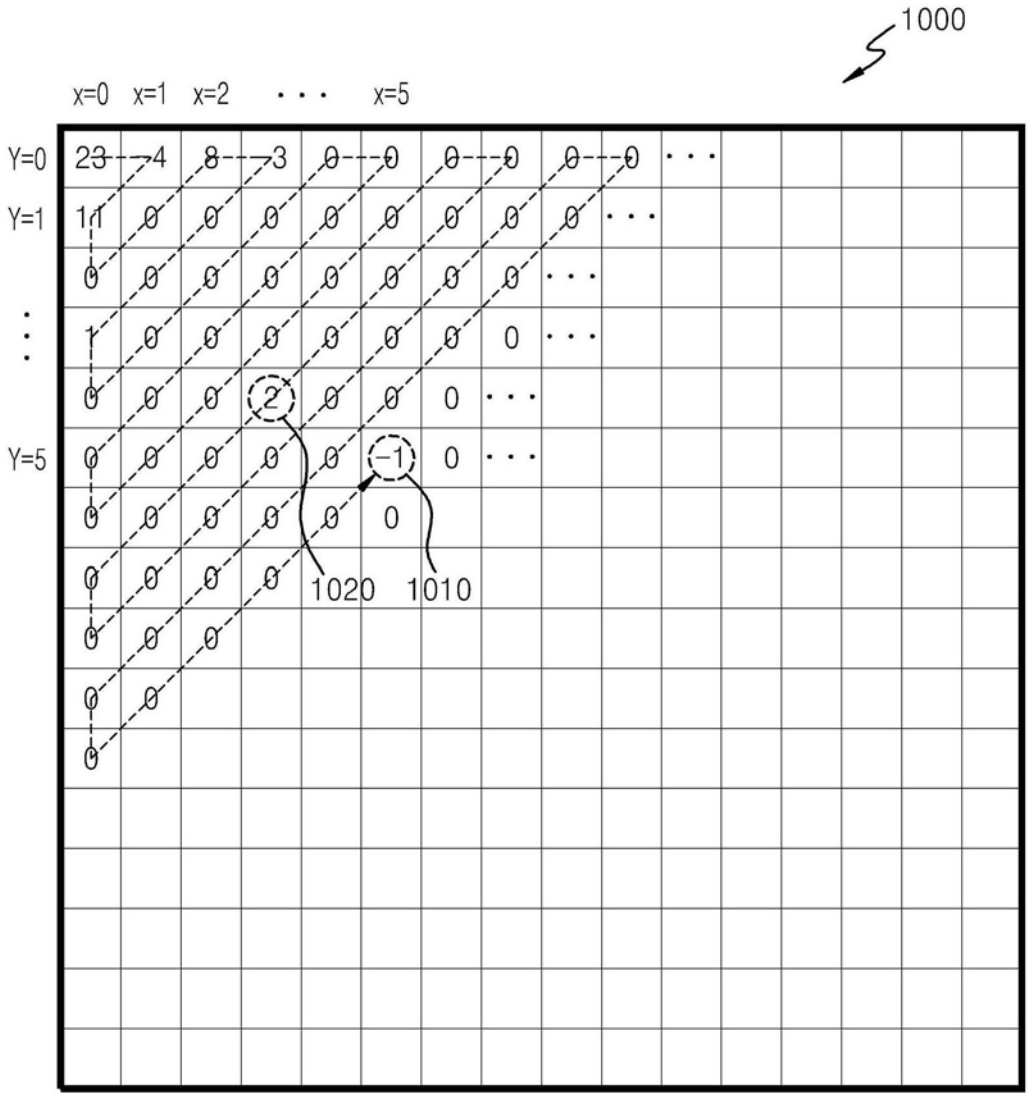

1000
x=0 x=1 x=2 · · · x=5
Y=0
Y=1
Y=5
1020
1010

图10

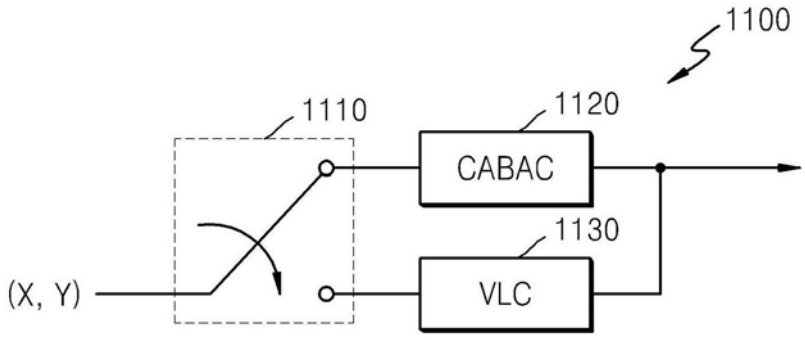

1100
1110
1120
CABAC
1130
VLC
(X, Y)

图11

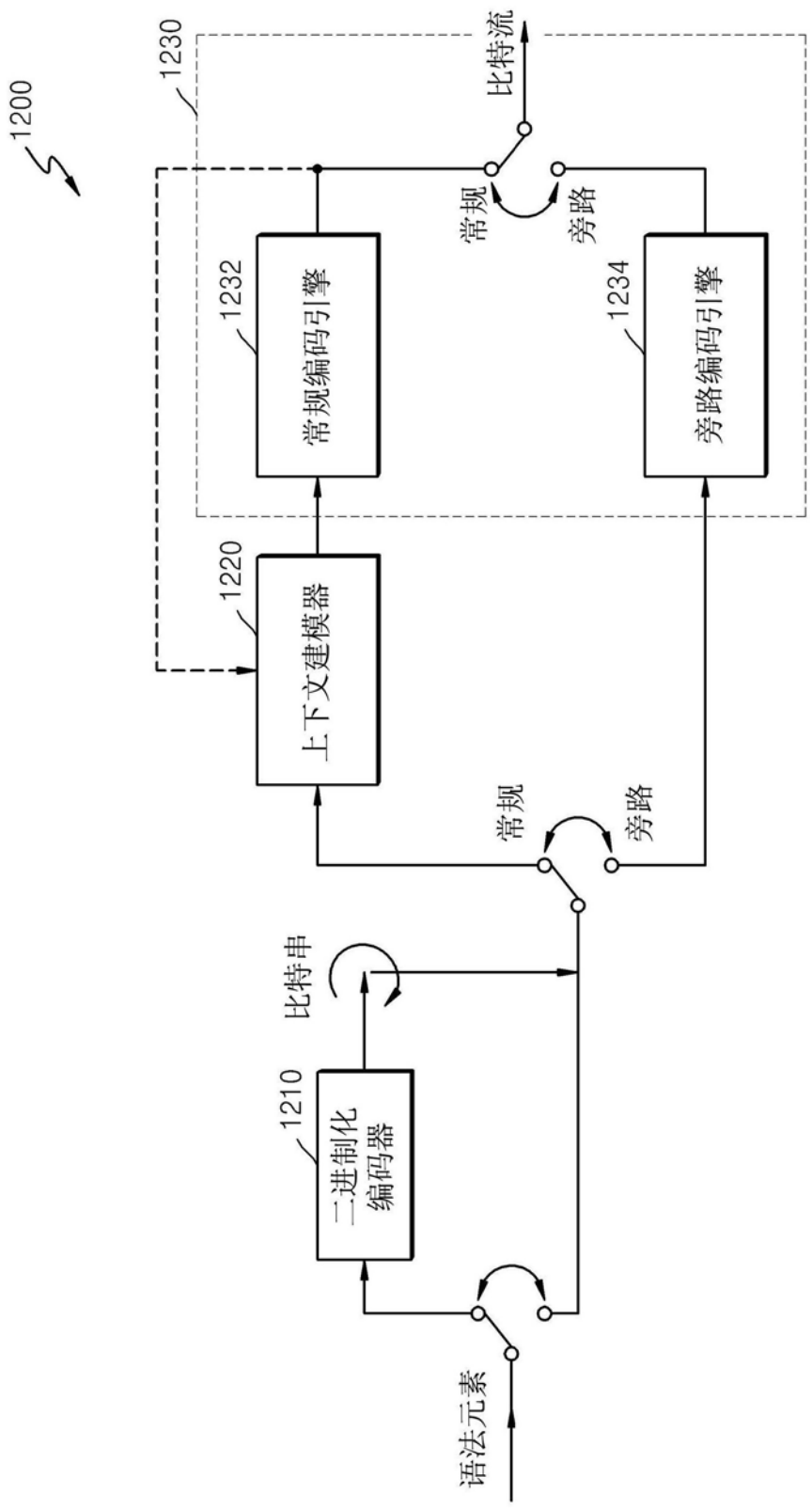

1200
1230
比特流
常规
旁路
1232
常规编码引擎
1234
旁路编码引擎
1220
上下文建模器
常规
旁路
比特串
1210
二进制化
编码器
语法元素

图12

4×4：	0	1	2													
8×8：	0	1	2	3	4	5	6									
8×8：	0	1	2	3	4	5	6	7	7	8	8	9	9	10	10	11

图13

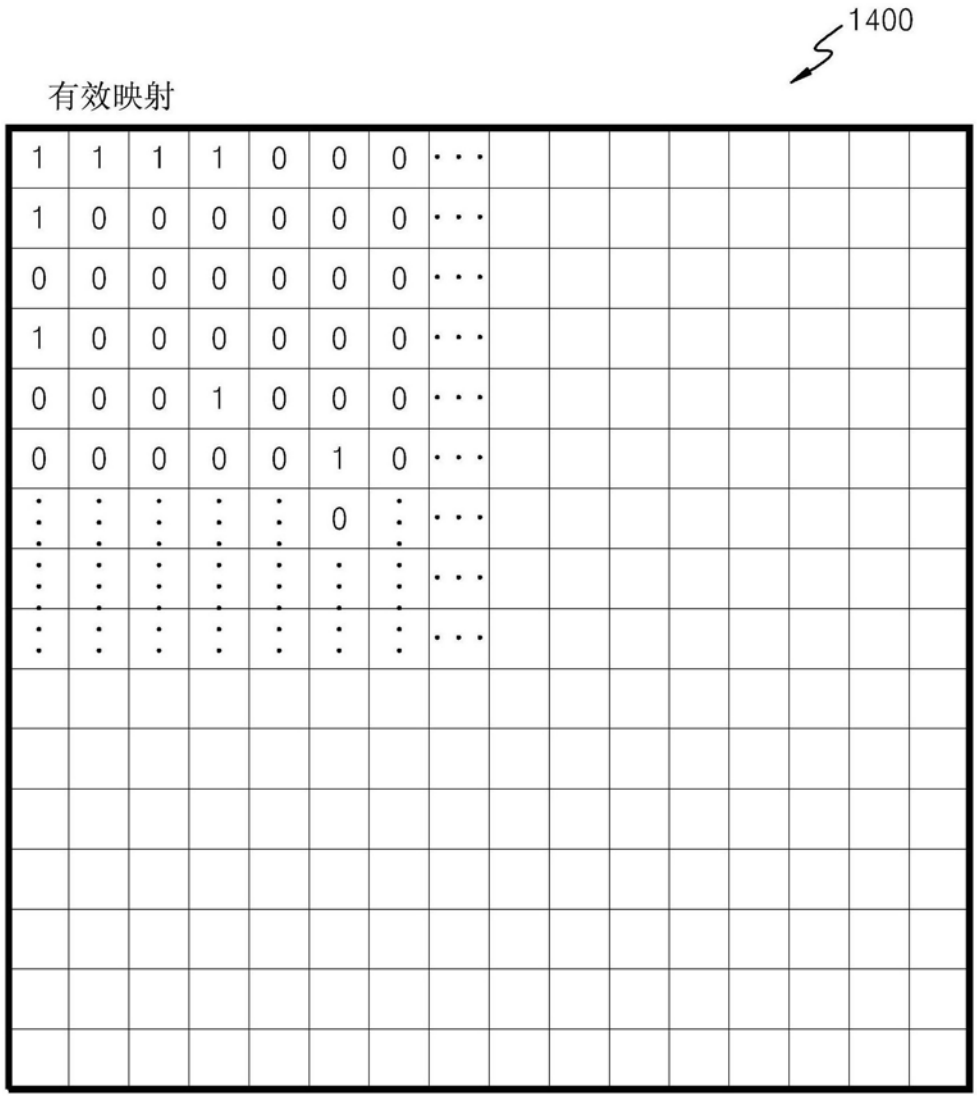


图14

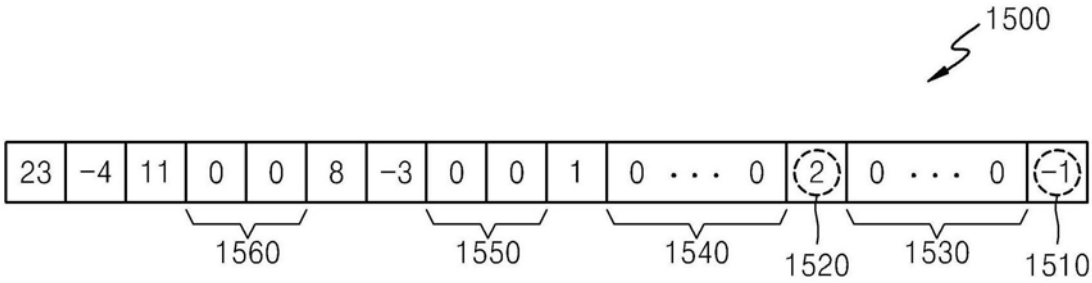


图15

VLC0	VLC1	VLC2	VLC3	VLC4	VLC5	VLC6	VLC7	VLC8
1	1x	1xx	1xxx	1xxxx	1x	1xx	1xxx	1
01	01x	01xx	01xxx	01xxxx	01x	01xx	01xxx	01
001	001x	001xx	001xxx	001xxxx	001x	001xx	001xxx	00
0001	0001x	0001xx	0001xxx	0001xxxx	. . .	. . .	. . .	. . .
00001	00001x	00001xx	00001xxx	00001xxxx				
0000001	0000001x	0000001xx	0000001xxx	0000001xxxx				
00000001	00000001x	00000001xx	00000001xxx	00000001xxxx				
000000001x	000000001xx	000000001xxx	000000001xxxx	000000001xxxxx				
0000000001xx	0000000001xxx	0000000001xxx	0000000001xxxxx	0000000001xxxxxx				
. . .	. . .	. . .	. . .	. . .				

图16

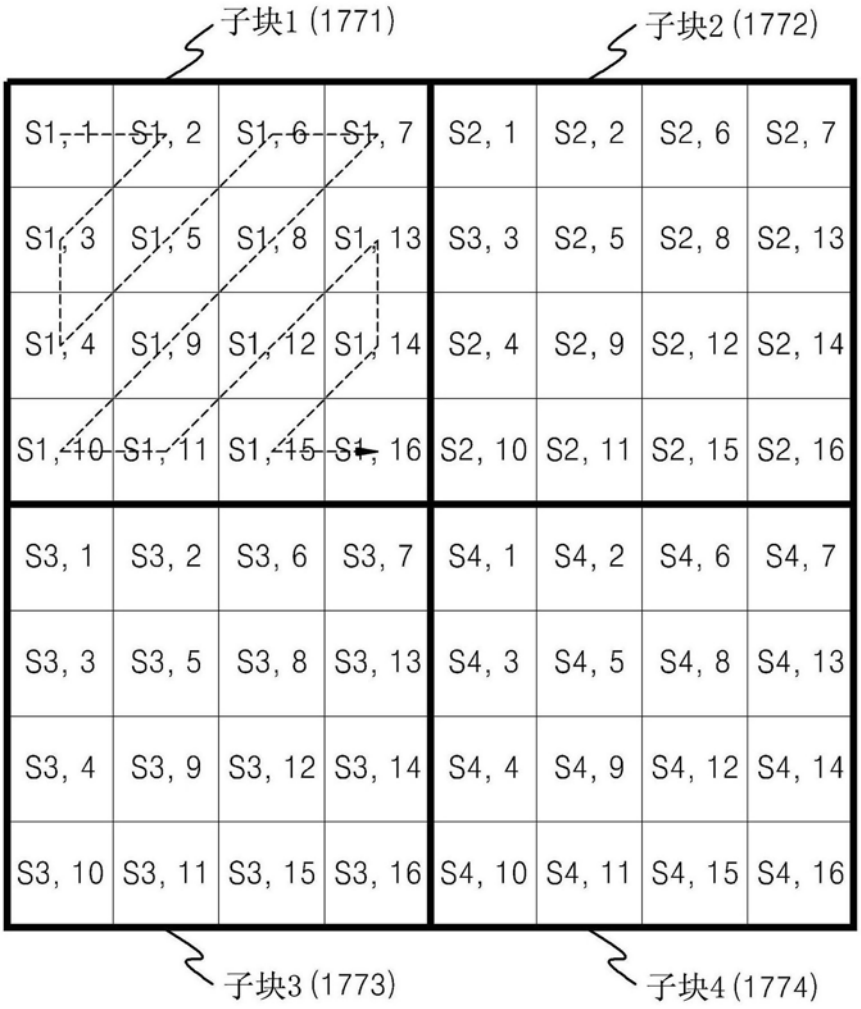

子块1 (1771)
子块2 (1772)
S1, 1
S1, 2
S1, 6
S1, 7
S2, 1
S2, 2
S2, 6
S2, 7
S1, 3
S1, 5
S1, 8
S1, 13
S3, 3
S2, 5
S2, 8
S2, 13
S1, 4
S1, 9
S1, 12
S1, 14
S2, 4
S2, 9
S2, 12
S2, 14
S1, 10
S1, 11
S1, 15
S1, 16
S2, 10
S2, 11
S2, 15
S2, 16
S3, 1
S3, 2
S3, 6
S3, 7
S4, 1
S4, 2
S4, 6
S4, 7
S3, 3
S3, 5
S3, 8
S3, 13
S4, 3
S4, 5
S4, 8
S4, 13
S3, 4
S3, 9
S3, 12
S3, 14
S4, 4
S4, 9
S4, 12
S4, 14
S3, 10
S3, 11
S3, 15
S3, 16
S4, 10
S4, 11
S4, 15
S4, 16
子块3 (1773)
子块4 (1774)

图17

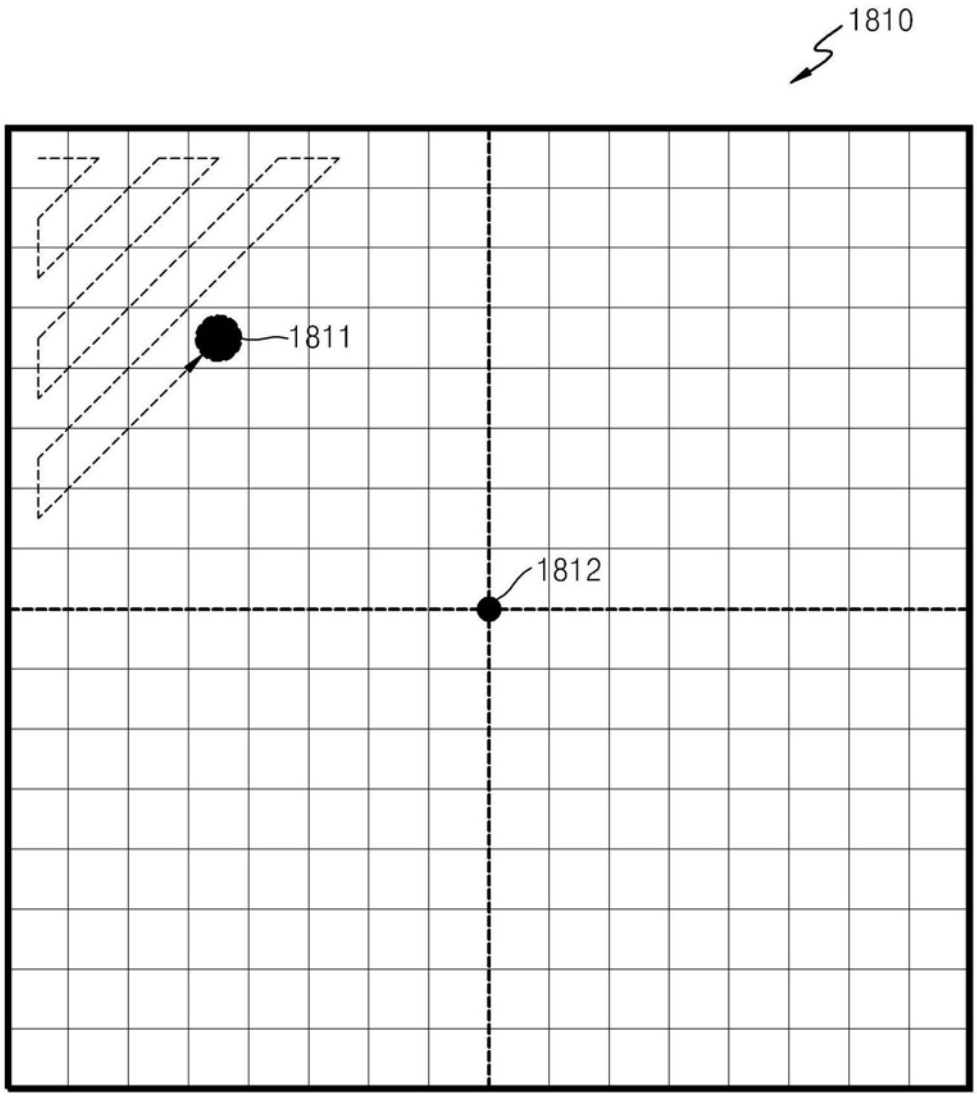


图18A

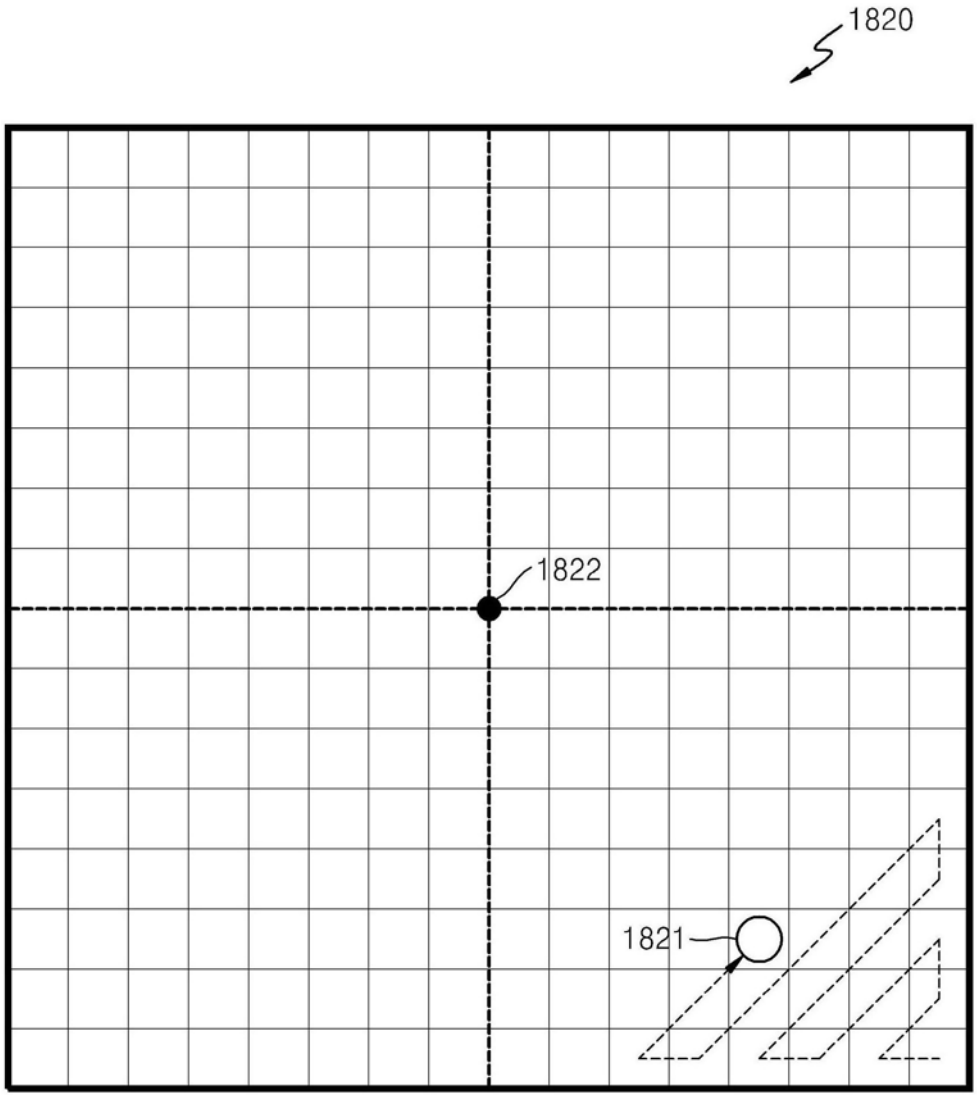


图18B

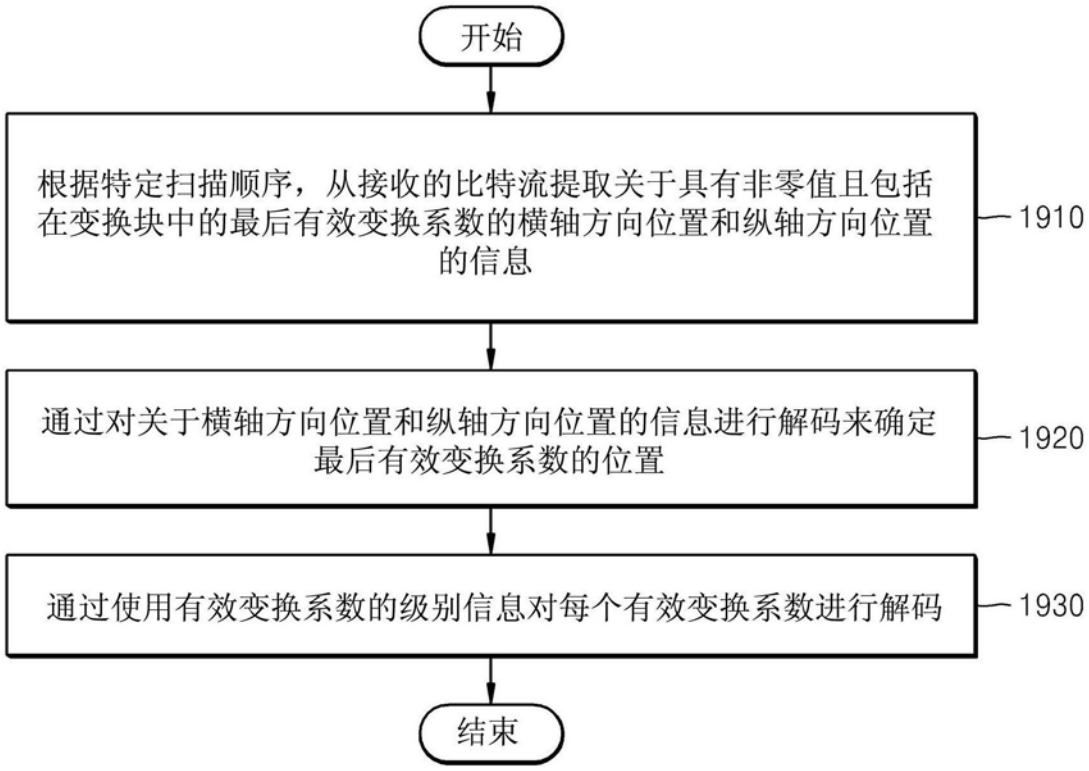

开始
根据特定扫描顺序，从接收的比特流提取关于具有非零值且包括在变换块中的最后有效变换系数的横轴方向位置和纵轴方向位置的信息
1910
通过对关于横轴方向位置和纵轴方向位置的信息进行解码来确定最后有效变换系数的位置
1920
通过使用有效变换系数的级别信息对每个有效变换系数进行解码
1930
结束

图19

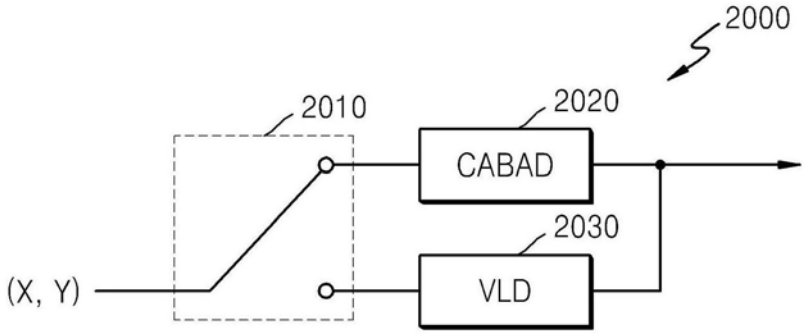

2000
2010
2020
CABAD
2030
VLD
(X, Y)

图20