



(12) 发明专利申请

(10) 申请公布号 CN 118317083 A

(43) 申请公布日 2024.07.09

(21) 申请号 202410357561.0

H04N 19/126 (2014.01)

(22) 申请日 2020.02.28

H04N 19/60 (2014.01)

(30) 优先权数据

H04N 19/61 (2014.01)

2019-044274 2019.03.11 JP

H04N 19/176 (2014.01)

(62) 分案原申请数据

H04N 19/157 (2014.01)

202080020010.5 2020.02.28

H04N 19/44 (2014.01)

(71) 申请人 佳能株式会社

H04N 19/119 (2014.01)

地址 日本东京都大田区下丸子3丁目30番2号

H04N 19/136 (2014.01)

H04N 19/132 (2014.01)

(72)发明人 志摩真悟

(74) 专利代理机构 北京魏启学律师事务所

11398

专利代理师 魏启学 王小香

(51) Int.Cl.

H04N 19/124 (2014.01)

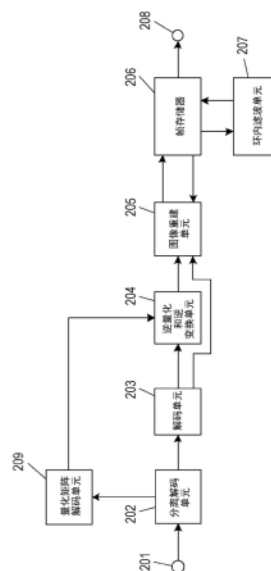
权利要求书2页 说明书17页 附图14页

(54) 发明名称

图像解码设备和图像解码方法

(57) 摘要

本发明涉及一种图像解码设备和图像解码方法。解码单元从位流中解码与 $N \times M$ 阵列的量化系数相对应的数据。逆量化单元通过至少使用量化矩阵来从 $N \times M$ 阵列的量化系数导出正交变换系数。逆正交变换单元对逆量化单元所生成的正交变换系数进行逆正交变换,以生成与 $P \times Q$ 阵列的像素的块相对应的预测残差。逆量化单元通过至少使用 $N \times M$ 阵列的元素的量化矩阵来导出正交变换系数,并且逆正交变换单元生成针对具有比 $N \times M$ 阵列大的大小的 $P \times Q$ 阵列的像素的预测残差。



1. 一种图像解码设备,其能够通过使用多个块来从位流中解码图像,所述多个块包括 $P \times Q$ 的像素阵列的第一块,其中 P 和 Q 是整数,所述图像解码设备包括:

解码单元,其被配置为从所述位流中解码与量化变换系数相对应的数据;

逆量化单元,其被配置为通过使用量化矩阵来从所述量化变换系数导出变换系数,所述变换系数表示频率分量;

逆变换单元,其被配置为通过对所述变换系数进行逆变换处理,来从所述变换系数导出预测残差;以及

图片重建单元,其被配置为基于所述预测残差和预测图像数据来重建图片数据,所述预测图像数据能够通过使用帧间预测来导出,

其中,在对象块是所述第一块的情况下,

(i) 所述逆量化单元被配置为通过使用 $N \times M$ 的元素阵列的量化矩阵来从 $N \times M$ 的量化变换系数阵列导出 $N \times M$ 的变换系数阵列,其中, N 是满足 $N < P$ 的整数,且 M 是满足 $M < Q$ 的整数,以及

(ii) 所述逆变换单元被配置为通过进行所述 $N \times M$ 的变换系数阵列与 $M \times Q$ 阵列的矩阵的相乘来导出 $N \times Q$ 的中间值阵列,并且通过进行 $P \times N$ 阵列的矩阵与所述 $N \times Q$ 的中间值阵列的相乘来导出针对所述第一块的 $P \times Q$ 的预测残差阵列,以及

其中,在对象块是比所述第一块小的第二块的情况下,

(i) 所述逆量化单元被配置为通过使用针对所述第二块的量化矩阵来从针对所述第二块的量化变换系数的阵列导出针对所述第二块的变换系数的阵列,其中,针对所述第二块的变换系数的阵列的大小、针对所述第二块的量化变换系数的阵列的大小、以及针对所述第二块的量化矩阵的大小与所述第二块的大小相同,以及

(ii) 所述逆变换单元被配置为通过至少使用针对所述第二块的变换系数的阵列和大小与所述第二块的大小相同的矩阵进行相乘,来导出大小与所述第二块的大小相同的预测残差的阵列。

2. 一种图像解码方法,其中能够通过使用多个块来从位流中解码图像,所述多个块包括 $P \times Q$ 的像素阵列的第一块,其中 P 和 Q 是整数,所述图像解码方法包括:

从所述位流中解码与量化变换系数相对应的数据;

通过使用量化矩阵来从所述量化变换系数导出变换系数,所述变换系数表示频率分量;

通过对所述变换系数进行逆变换处理,来从所述变换系数导出预测残差;以及

基于所述预测残差和预测图像数据来重建图片数据,所述预测图像数据能够通过使用帧间预测来导出,

其中,在对象块是所述第一块的情况下,

(i) 通过使用 $N \times M$ 的元素阵列的量化矩阵来从 $N \times M$ 的量化变换系数阵列导出 $N \times M$ 的变换系数阵列,其中, N 是满足 $N < P$ 的整数,且 M 是满足 $M < Q$ 的整数,以及

(ii) 通过进行所述 $N \times M$ 的变换系数阵列与 $M \times Q$ 阵列的矩阵的相乘来导出 $N \times Q$ 的中间值阵列,并且通过进行 $P \times N$ 阵列的矩阵与所述 $N \times Q$ 的中间值阵列的相乘来导出针对所述第一块的 $P \times Q$ 的预测残差阵列,以及

其中,在对象块是比所述第一块小的第二块的情况下,

(i) 通过使用针对所述第二块的量化矩阵来从针对所述第二块的量化变换系数的阵列导出针对所述第二块的变换系数的阵列,其中,针对所述第二块的变换系数的阵列的大小、针对所述第二块的量化变换系数的阵列的大小、以及针对所述第二块的量化矩阵的大小与所述第二块的大小相同,以及

(ii) 通过至少使用针对所述第二块的变换系数的阵列和大小与所述第二块的大小相同的矩阵进行相乘,来导出大小与所述第二块的大小相同的预测残差的阵列。

图像解码设备和图像解码方法

[0001] (本申请是申请日为2020年2月28日、申请号为2020800200105、发明名称为“图像解码设备、图像解码方法和存储介质”的申请的分案申请。)

技术领域

[0002] 本发明涉及图像解码技术。

背景技术

[0003] 已知有高效率视频编码 (HEVC) 方法 (以下称为HEVC) 作为对运动图像进行压缩的编码方法。HEVC采用具有比传统宏块 (16×16 阵列的像素) 大的大小的基本块以提高编码效率。该具有大大小小的基本块被称为编码树单元 (CTU), 并且大小最大为 64×64 阵列的像素。CTU被进一步分割成作为用于进行预测和变换的单位的子块。

[0004] 在HEVC中, 使用量化矩阵来根据频率分量向进行了正交变换的系数 (以下称为正交变换系数) 指派权重。量化矩阵的使用可以通过与具有低频分量的数据相比, 更多地减少具有对于人类视觉而言劣化不太明显的高频分量的数据, 来在维持图像质量的同时提高压缩效率。日本特开2013-38758论述了用于对指示这种量化矩阵的信息进行编码的技术。

[0005] 最近已经开始作为HEVC的后继者的更加高效编码方法的国际标准化。具体地, 由国际标准化组织和国际电工委员会 (ISO/IEC) 以及ITU电信标准化部门 (ITU-T) 建立的联合视频专家组 (JVET) 已经正在推进通用视频编码 (VVC) 编码方法 (以下称为VVC) 的标准化。对于这种标准化, 已经正在研究通过在要进行正交变换的块大小大的情况下将高频分量的正交变换系数强制设置为0来减少代码量以提高编码效率的新技术 (以下将这种新技术称为归零)。

[0006] 现有技术文献

[0007] 专利文献

[0008] 专利文献1: 日本特开2013-38758

发明内容

[0009] 在VCC中, 如HEVC那样, 量化矩阵的引入也正在检讨中。然而, HEVC中的量化矩阵以使用具有与传统正交变换的大小相同的大小的量化矩阵的量化方法为前提, 并且不支持作为将正交变换系数的一部分设置为零的新技术的归零。因此, 不能对进行了归零的正交变换系数进行基于频率分量的量化控制, 并且不能提高主观图像质量, 这已成为问题。

[0010] 用于解决问题的方案

[0011] 为了即使在通过使得能够进行使用与将正交变换系数的一部分强制设置为0的技术相对应的量化矩阵的量化处理来使用该技术的条件下, 也能提高主观图像质量, 例如, 本发明提供以下的结构。更具体地, 一种图像解码设备, 其能够以多个块为单位从位流中解码图像, 所述多个块包括 $P \times Q$ 阵列的像素的块, 其中 P 和 Q 是整数, 所述图像解码设备包括: 解码单元, 其被配置为从所述位流中解码与 $N \times M$ 阵列的量化系数相对应的数据, 其中, N 是满

足 $N < P$ 的整数,且 M 是满足 $M < Q$ 的整数;逆量化单元,其被配置为通过至少使用量化矩阵来从所述 $N \times M$ 阵列的量化系数生成正交变换系数;以及逆正交变换单元,其被配置为对所述逆量化单元所生成的正交变换系数进行逆正交变换,以生成与所述 $P \times Q$ 阵列的像素的块相对应的预测残差。所述逆量化单元通过至少使用 $N \times M$ 阵列的元素的量化矩阵来从所述 $N \times M$ 阵列的量化系数生成所述正交变换系数,并且所述逆正交变换单元对所述正交变换系数进行逆正交变换,以生成针对具有比 $N \times M$ 阵列大的大小的所述 $P \times Q$ 阵列的像素的预测残差。

附图说明

- [0012] 图1是示出根据第一典型实施例的图像编码设备的结构的框图。
- [0013] 图2是示出根据第二典型实施例的图像解码设备的结构的框图。
- [0014] 图3是示出根据第一典型实施例的图像编码设备中的图像编码处理的流程图。
- [0015] 图4是示出根据第二典型实施例的图像解码设备中的图像解码处理的流程图。
- [0016] 图5是示出可应用于本发明的图像编码设备和图像解码设备的计算机的硬件结构的示例的框图。
- [0017] 图6A是示出根据第一典型实施例所要输出的位流的示例的图。
- [0018] 图6B是示出根据第一典型实施例所要输出的位流的示例的图。
- [0019] 图7A是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0020] 图7B是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0021] 图7C是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0022] 图7D是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0023] 图7E是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0024] 图7F是示出第一典型实施例和第二典型实施例中所使用的子块分割的示例的图。
- [0025] 图8A是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的示例的图。
- [0026] 图8B是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的示例的图。
- [0027] 图8C是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的示例的图。
- [0028] 图9是示出用于扫描第一典型实施例和第二典型实施例中所使用的量化矩阵的各元素的方法的图。
- [0029] 图10是示出第一典型实施例和第二典型实施例中所生成的量化矩阵的差值矩阵的图。
- [0030] 图11A是示出用于对量化矩阵的差值进行编码的编码表的示例的图。
- [0031] 图11B是示出用于对量化矩阵的差值进行编码的编码表的示例的图。
- [0032] 图12A是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的另一示例的图。
- [0033] 图12B是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的另一示例的图。
- [0034] 图12C是示出第一典型实施例和第二典型实施例中所使用的量化矩阵的另一示例的图。

具体实施方式

[0035] 以下将参考附图来说明本发明的典型实施例。以下典型实施例的结构将被视为例示性的,并且本发明不限于以下典型实施例的结构。在各典型实施例中方便地使用诸如基本块、子块、量化矩阵和基量化矩阵等的术语,并且在含义保持不变的范围内也可以使用其它术语。例如,基本块和子块可被分别称为基本单元和子单元,或者分别简称为块和单元。在以下的说明中,根据一般定义,矩形是指四个内角为直角且两条对角线的长度相等的四边形。根据一般定义,正方形是指四边形中的四个角相等且四条边的长度相等的四边形。换句话说,正方形是矩形的一种。

[0036] <第一典型实施例>

[0037] 以下将参考附图来说明本发明的典型实施例。

[0038] 以下将更详细地说明归零。如上所述,归零是指将编码对象块中的正交变换系数的一部分强制设置为0的处理。例如,假定输入图像(图片)中的 64×64 阵列的像素的块是编码对象块。在这种情况下,正交变换系数也具有 64×64 阵列的大小。在归零时,即使作为正交变换的结果, 64×64 阵列的正交变换系数的一部分具有非零值,也通过将这些正交变换系数的值视为0来进行编码。例如,在二维的正交变换系数中,与左上方的预定范围相对应的包括DC分量的低频分量未被强制设置为0,而与比低频分量高的频率分量相对应的正交变换系数始终设置为0。

[0039] 以下将说明根据本典型实施例的图像编码设备。图1是示出根据本典型实施例的图像编码设备的框图。参考图1,终端101输入图像数据。

[0040] 块分割单元102将输入图像分割成多个基本块,并且将以基本块为单位的图像输出到后级。

[0041] 量化矩阵存储单元103生成并存储量化矩阵。量化矩阵用于根据频率分量来向对正交变换系数的量化处理指派权重。例如,通过进行基于基准参数值(量化参数)的标度值(量化标度)与量化矩阵中的各元素的值的相乘,来向(以下所述的)量化处理中的针对各正交变换系数的量化步长指派权重。

[0042] 用于生成量化矩阵存储单元103所要存储的量化矩阵的方法不限于特定方法。例如,用户可以输入指示量化矩阵的信息,或者图像编码设备可以基于输入图像的特性来计算量化矩阵。可选地,可以使用预先指定为初始值的量化矩阵。除图8A所示的 8×8 阵列的基量化矩阵之外,在本典型实施例中,还通过扩大该基量化矩阵来生成图8B和图8C所示的两个不同类型的 32×32 阵列的二维量化矩阵,然后存储所生成的这两个不同的量化矩阵。图8B的量化矩阵是通过使图8A的 8×8 阵列的基量化矩阵变为四倍(即,通过将该基量化矩阵的各元素在垂直和水平方向上重复四次)所生成的 32×32 阵列的量化矩阵。另一方面,图8C的量化矩阵是通过扩大图8A的基量化矩阵的左上方 4×4 部分(即,通过将该部分中的各元素在垂直和水平方向上重复八次)所生成的 32×32 阵列的量化矩阵。

[0043] 如上所述,基量化矩阵不仅用于对 8×8 阵列的像素的子块进行量化,而且还用于生成具有比基量化矩阵大的大小的量化矩阵。尽管所使用的基量化矩阵的大小是 8×8 阵列,但该大小不限于此。此外,根据子块的大小,可以使用另一基量化矩阵。例如,当使用 8×8 、 16×16 和 32×32 阵列的三个不同子块时,也可应用各自与这些子块中的不同子块相对应的三个不同基量化矩阵。

[0044] 预测单元104确定针对以基本块为单位的图像数据的子块分割。更具体地,预测单元104确定是否将基本块分割成子块,并且如果要进行向子块的分割,则确定如何对基本块进行分割。如果不进行向子块的分割,则子块具有与基本块的大小相同的大小。子块可以是正方形或除正方形以外的矩形(非正方形)。

[0045] 预测单元104以子块为单位进行作为帧内预测(intra-frame prediction)的帧内预测(intra prediction)或作为帧间预测(inter-frame prediction)的帧间预测(inter prediction)等,以生成预测图像数据。

[0046] 例如,预测单元104从帧内预测和帧间预测中选择要对一个子块进行的预测方法,并进行所选择的预测以生成该子块的预测图像数据。然而,要使用的预测方法不限于此,而且也可应用组合帧内预测和帧间预测的预测方法。

[0047] 预测单元104进一步基于所输入的图像数据和预测图像数据来计算预测残差,并输出这些预测残差。例如,预测单元104计算子块的各像素值与通过对该子块的预测所生成的预测图像数据的各像素值之间的差,并且将这些差计算为预测残差。

[0048] 预测单元104还将预测所需的信息(例如,指示子块分割状态的信息、以及与指示子块的预测方法的预测模式有关且与运动矢量有关的信息)连同预测残差等一起输出。在以下的说明中,预测所需的信息被统称为预测信息。

[0049] 变换和量化单元105以子块为单位对预测单元104所计算出的预测残差进行正交变换,以获取表示预测残差的各频率分量的正交变换系数。变换和量化单元105进一步通过使用量化参数和量化矩阵存储单元103中所存储的量化矩阵来对正交变换系数进行量化,以获取作为量化后的正交变换系数的量化系数。正交变换的功能和量化的功能可以是单独配置的。

[0050] 逆量化和逆变换单元106通过使用量化系数和量化矩阵存储单元103中所存储的量化矩阵来对从变换和量化单元105输出的量化系数进行逆量化,以重建正交变换系数。然后,逆量化和逆变换单元106进一步对正交变换系数进行逆正交变换以重建预测残差。通过使用量化矩阵和量化参数来重建(导出)正交变换系数的处理被称为逆量化。进行逆量化的功能和进行逆变换的功能可以是单独配置的。图像解码设备导出量化参数所使用的信息由编码单元110编码在位流中。

[0051] 帧存储器108存储重建的图像数据。

[0052] 图像重建单元107基于从预测单元104输出的预测信息通过适当地参考帧存储器108来生成预测图像数据,基于该预测图像数据和所输入的预测残差来生成重建对象图像数据,并且输出该重建对象图像数据。

[0053] 环内滤波单元109对重建对象图像进行诸如解块滤波和采样自适应偏移等的环内滤波处理,并输出滤波处理后的图像。

[0054] 编码单元110对从变换和量化单元105输出的量化系数以及从预测单元104输出的预测信息进行编码以生成代码数据,并输出该代码数据。

[0055] 量化矩阵编码单元113对从量化矩阵存储单元103输出的基量化矩阵进行编码,以生成图像解码设备导出基量化矩阵所要使用的量化矩阵代码数据,并输出该量化矩阵代码数据。

[0056] 整合编码单元111通过使用从量化矩阵编码单元113输出的量化矩阵代码数据来

生成头代码数据。整合编码单元111进一步连同从编码单元110输出的代码数据一起形成位流,并输出结果。

[0057] 终端112将由整合编码单元111生成的位流输出到外部。

[0058] 以下将说明利用图像编码设备的图像编码操作。在本典型实施例的结构中,以帧为单位输入运动图像数据。在本典型实施例的说明中,尽管块分割单元102将运动图像数据分割成 64×64 阵列的像素的基本块,但本发明不限于此。例如,可以使用 128×128 阵列的像素的基本块作为基本块,或者可以使用 32×32 阵列的像素的基本块作为基本块。

[0059] 在图像编码之前,图像编码设备生成量化矩阵并对该量化矩阵进行编码。在以下的说明中,在例如量化矩阵800和各块中,水平方向被称为x坐标,垂直方向被称为y坐标,其中水平方向的右方向和垂直方向的下方向被称为正。量化矩阵800的最左上方元素的坐标值是(0,0)。更具体地, 8×8 阵列的基量化矩阵的最右下方元素的坐标值是(7,7)。 32×32 阵列的量化矩阵的最右下方元素的坐标值是(31,31)。

[0060] 首先,量化矩阵存储单元103生成量化矩阵。根据子块的大小、要量化的正交变换系数的大小和预测方法的类型来生成量化矩阵。本典型实施例生成要用于生成(以下所述的)量化矩阵的 8×8 阵列的基量化矩阵(图8A)。然后,量化矩阵存储单元103扩大该基量化矩阵以生成图8B和图8C所示的 32×32 阵列的两个不同的量化矩阵。图8B的量化矩阵是通过使图8A中的 8×8 阵列的基量化矩阵变为四倍(即,通过将该基量化矩阵的各元素在垂直和水平方向上重复四次)所生成的 32×32 阵列的量化矩阵。

[0061] 更具体地,在图8B所示的示例中,向 32×32 阵列的量化矩阵中的x坐标为0~3且y坐标为0~3的范围中的各元素指派基量化矩阵的最左上方元素的值1。向 32×32 阵列的量化矩阵中的x坐标为28~31且y坐标为28~31的范围中的各元素指派基量化矩阵的最右下方元素的值15。在图8B的示例中,基量化矩阵中的元素的值各自均被指派给 32×32 阵列的量化矩阵中的元素的一部分。

[0062] 另一方面,图8C的量化矩阵是通过将图8A的基量化矩阵的左上方 4×4 部分中的各元素在垂直和水平方向上重复八次而放大得到的 32×32 阵列的量化矩阵。

[0063] 更具体地,在图8C所示的示例中,向 32×32 阵列的量化矩阵中的x坐标为0~7且y坐标为0~7的范围中的各元素指派基量化矩阵的左上方 4×4 部分中的最左上方元素的值1。向 32×32 阵列的量化矩阵中的x坐标为24~31且y坐标为24~31的范围中的各元素指派基量化矩阵的左上方 4×4 部分中的最右下方元素的值7。在图8C的示例中,基量化矩阵中的元素的值中的仅与左上方 4×4 部分(x坐标为0~3且y坐标为0~3的范围)相对应的元素的值被指派给 32×32 阵列的量化矩阵的元素。

[0064] 注意,要生成的量化矩阵不限于此。在要量化的正交变换系数具有除 32×32 阵列以外的大小的情况下,可以生成诸如 16×16 、 8×8 和 4×4 阵列等的、与要量化的正交变换系数的大小相对应的量化矩阵。没有特别限制确定构成基量化矩阵和量化矩阵的各元素的方法。例如,可以使用预定的初始值或者可以单独设置各元素。可以根据图像特性来生成量化矩阵。

[0065] 量化矩阵存储单元103存储这样生成的基量化矩阵和量化矩阵。图8B和8C分别示出针对与(以下所述的) 32×32 阵列和 64×64 阵列的子块相对应的正交变换系数的量化所要使用的量化矩阵的示例。粗框800表示量化矩阵。为了简化说明,量化矩阵800包括1024

(32×32) 个像素,并且粗框800中的各正方形表示构成量化矩阵800的各元素。在本典型实施例中,尽管图8B和图8C所示的三个不同量化矩阵是以二维形式存储的,但量化矩阵中的元素的形式不限于此。根据要量化的正交变换系数的大小或者根据编码对象是亮度块还是色度块,可以针对相同的预测方法存储多个量化矩阵。通常,为了实现符合人类视觉特性的量化处理,如图8B和图8C所示,与量化矩阵的左上部分相对应的低频部分的元素具有小的值,并且与其右下部分相对应的高频部分的元素具有大的值。

[0066] 量化矩阵编码单元113从量化矩阵存储单元103顺次读取以二维形式存储的基量化矩阵的各元素,扫描各元素,计算差,并将各差值布置在一维阵列中。根据本典型实施例,对于图8A所示的基量化矩阵,通过使用图9所示的扫描方法来按扫描顺序针对各元素计算与紧前元素的差。例如,通过图9所示的扫描方法来扫描图8A所示的 8×8 阵列的基量化矩阵。在扫描最左上方的第一个元素1之后,扫描位于元素1下方的元素2,并且计算出差+1。为了对量化矩阵的第一个元素(根据本第一典型实施例为元素1)进行编码,计算与预定初始值(例如,8)的差,但本发明不限于此。可以使用与某个值的差或者第一元素自身的值。

[0067] 以这种方式,本典型实施例基于图8A的量化矩阵通过使用图9的扫描方法来生成图10所示的差分矩阵。量化矩阵编码单元113进一步对该差分矩阵进行编码以生成量化矩阵代码数据。在本典型实施例中,尽管量化矩阵编码单元113通过使用图11A所示的编码表来进行编码,但编码表不限于此,而且可以使用图11B所示的编码表。以这种方式生成的量化矩阵代码数据被输出到后级的整合编码单元111。

[0068] 返回参考图1,整合编码单元111对图像数据编码所需的头信息进行编码,并将量化矩阵的代码数据与编码后的头信息整合。

[0069] 随后,整合编码单元111对图像数据进行编码。从终端101输入的一帧的图像数据被输入到块分割单元102。

[0070] 块分割单元102将所输入的图像数据分割成多个基本块,然后将以基本块为单位的图像输出到预测单元104。根据本典型实施例,块分割单元102输出以 64×64 阵列的像素的基本块为单位的图像。

[0071] 图7A至图7F示出子块分割方法的示例。粗框700表示基本块。为了简化说明,基本子块700具有 64×64 阵列的像素,并且粗框700中的各正方形表示各子块。图7B示出四叉树的正方形子块分割的示例。将 64×64 阵列的像素的基本块分割成 32×32 阵列的像素的子块。图7C至图7F示出矩形子块分割的示例。参考图7C,基本块被分割成 32×64 阵列的像素的纵长矩形子块。参考图7D,基本块被分割成 64×32 阵列的像素的横长矩形子块。参考图7E和图7F,基本块按1:2:1的比被分割成不同的矩形子块。这样,块分割单元102通过不仅使用正方形子块而且还使用非正方形的矩形子块来进行编码处理。可以将基本块进一步分割成多个正方形块,然后将这些分割正方形块分割成子块。换句话说,基本块的大小不限于 64×64 阵列的像素,并且可以使用具有多个大小的基本块。

[0072] 在本典型实施例中,尽管仅使用如图7A所示未分割的 64×64 阵列的像素的基本块和如图7B所示的四叉树分割,但子块分割方法不限于此。可以使用图7E和图7F所示的三叉树分割或者图7C和图7D所示的二叉树分割。在使用除图7A和图7B中的分割以外的子块分割的情况下,生成与量化矩阵存储单元103所要使用的子块相对应的量化矩阵。在还生成与所生成的量化矩阵相对应的新的基量化矩阵的情况下,该新的基量化矩阵将由量化矩阵编码

单元113进行编码。

[0073] 以下将更详细地说明利用根据本典型实施例的预测单元104的预测方法。作为示例,本典型实施例使用帧内预测和帧间预测这两个不同的预测方法。在帧内预测方法中,通过使用在空间上位于编码对象块周围的已编码图像来生成编码对象块的预测像素,并且还生成指示包括水平预测、垂直预测和IDC预测的帧内预测方法中的所使用的帧内预测方法的帧内预测模式信息。在帧间预测方法中,通过使用在时间上与编码对象块不同的帧的已编码图像来生成编码对象块的预测像素,并且还生成指示要参考的帧或运动矢量的运动信息。如上所述,预测单元104可以使用组合帧内预测和帧间预测的预测方法。

[0074] 预测单元104基于所确定的预测模式和已编码图像来生成预测图像数据,根据所输入的图像数据和该预测图像数据来生成预测残差,并将这些预测残差输出到变换和量化单元105。将诸如子块分割和预测模式等的信息作为预测信息输出到编码单元110和图像重建单元107。

[0075] 变换和量化单元105对所输入的预测残差进行正交变换和量化以生成量化系数。变换和量化单元105首先根据子块的大小对预测残差进行正交变换处理以生成正交变换系数,然后通过使用量化矩阵存储单元103中所存储的量化矩阵根据预测模式对所生成的正交变换系数进行量化来生成量化系数。以下将更详细地说明正交变换和量化处理。

[0076] 在选择图7B所示的 32×32 的子块分割的情况下,变换和量化单元105使用 32×32 的正交变换矩阵来对 32×32 阵列的预测残差进行正交变换,以生成 32×32 阵列的正交变换系数。更具体地,变换和量化单元105进行以离散余弦变换(DCT)为代表的 32×32 阵列的正交变换矩阵与 32×32 阵列的预测残差的相乘,以计算 32×32 阵列的中间系数。变换和量化单元105进一步进行 32×32 阵列的中间系数与前述 32×32 阵列的正交变换矩阵的转置矩阵的相乘,以生成 32×32 阵列的正交变换系数。变换和量化单元105通过使用量化参数和图8B所示的 32×32 阵列的量化矩阵来对所生成的 32×32 阵列的正交变换系数进行量化以生成 32×32 阵列的量化系数。由于 64×64 阵列的基本块包括 32×32 阵列的四个子块,因此变换和量化单元105将上述处理重复四次。

[0077] 在图7A所示的 64×64 分割状态(未分割)的情况下,将通过对 64×64 阵列的正交变换矩阵中的奇数编号的行(以下称为奇数行)进行间隔剔除所生成的 64×32 阵列的正交变换矩阵用于 64×64 阵列的预测残差。更具体地,变换和量化单元105使用通过对奇数行进行间隔剔除所生成的 64×32 阵列的正交变换矩阵来对预测残差进行正交变换,以生成 32×32 阵列的正交变换系数。

[0078] 更具体地,变换和量化单元105首先通过从 64×64 阵列的正交变换矩阵中对奇数行进行间隔剔除来生成 64×32 阵列的正交变换矩阵。然后,变换和量化单元105进行 64×32 阵列的正交变换矩阵与 64×64 阵列的预测残差的相乘,以生成 64×32 阵列的中间系数。变换和量化单元105进行 64×32 阵列的中间系数与通过对上述 64×32 阵列的正交变换矩阵进行转置所生成的 32×64 的转置矩阵的相乘,以生成 32×32 阵列的正交变换系数。然后,变换和量化单元105通过将所生成的 32×32 阵列的正交变换系数设置为 64×64 阵列的正交变换系数的左上部分(x坐标为0~31且y坐标为0~31的范围)的系数、并将其它部分设置为0,来进行归零。

[0079] 这样,在本典型实施例中,通过使用 64×32 阵列的正交变换矩阵和通过对 64×32

阵列的正交变换矩阵进行转置所生成的 32×64 阵列的转置矩阵,来对 64×64 阵列的预测残差进行正交变换。变换和量化单元105通过这样生成 32×32 阵列的正交变换系数来进行归零。这使得能够通过即使通过 64×64 的正交变换所生成的 64×64 阵列的正交变换系数的一部分的值不为0、也将该一部分强制设置为0的方法相比更少的计算量来生成 32×32 阵列的正交变换系数。更具体地,与通过使用 64×64 阵列的正交变换矩阵来进行正交变换、然后通过不论如此得到的要进行归零的正交变换系数是否为0都将这些系数视为0来对这些系数进行编码的情况相比,可以减少正交变换中的计算量。尽管使用用于根据 64×64 阵列的预测残差基于正交变换系数来计算 32×32 阵列的正交变换系数的方法可以减少计算量,但归零方法不限于此,而且可应用各种方法。

[0080] 在进行归零的情况下,可以对指示归零对象范围中的正交变换系数为0的信息、或者对仅仅指示进行了归零的信息(诸如标志等)进行编码。通过对这些信息进行解码,图像解码设备可以通过将归零对象视为0来对各块进行解码。

[0081] 然后,变换和量化单元105通过使用量化参数和图8C所示的 32×32 阵列的量化矩阵来对由此生成的 32×32 阵列的正交变换系数进行量化,以生成 32×32 阵列的量化系数。

[0082] 在本典型实施例中,对于与 32×32 阵列的子块相对应的 32×32 阵列的正交变换系数,使用图8B的量化矩阵,并且对于与 64×64 阵列的子块相对应的 32×32 阵列的正交变换系数,使用图8C的量化矩阵。更具体地,对于未进行归零的 32×32 阵列的正交变换系数,使用图8B的量化矩阵,并且对于进行了归零且与 64×64 阵列的子块相对应的 32×32 阵列的正交变换系数,使用图8C的量化矩阵。注意,要使用的量化矩阵不限于此。所生成的量化系数被输出到编码单元110以及逆量化和逆变换单元106。

[0083] 逆量化和逆变换单元106通过使用量化参数和量化矩阵存储单元103中所存储的量化矩阵来对所输入的量化系数进行逆量化,以重建正交变换系数。然后,逆量化和逆变换单元106进一步对重建的正交变换系数进行逆正交变换以重建预测残差。如变换和量化单元105那样,使用与编码对象子块的大小相对应的量化矩阵来进行逆量化处理。以下将更详细地说明利用逆量化和逆变换单元106的逆量化和逆正交变换处理。

[0084] 在选择图7B所示的 32×32 的子块分割,逆量化和逆变换单元106通过使用图8B的量化矩阵来对变换和量化单元105所生成的 32×32 阵列的量化系数进行逆量化,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元106进行前述 32×32 阵列的转置矩阵与 32×32 阵列的正交变换系数的相乘,以计算 32×32 阵列的中间系数。然后,逆量化和逆变换单元106进行 32×32 阵列的中间系数与前述 32×32 阵列的正交变换矩阵的相乘,以重建 32×32 阵列的预测残差。逆量化和逆变换单元106对 32×32 阵列的各个子块进行类似的处理。另一方面,在如图7A所示选择未分割的情况下,逆量化和逆变换单元106通过使用图8C的量化矩阵来对变换和量化单元105所生成的 32×32 阵列的量化系数进行逆量化,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元106进行 32×64 阵列的转置矩阵与 32×32 阵列的正交变换系数的相乘,以计算 32×64 阵列的中间系数。逆量化和逆变换单元106进行这些 32×64 阵列的中间系数与 64×32 阵列的正交变换矩阵的相乘,以重建 64×64 阵列的预测残差。本典型实施例根据子块的大小来通过使用与变换和量化单元105所使用的量化矩阵相同的量化矩阵来进行逆量化处理。重建的预测残差被输出到图像重建单元107。

[0085] 图像重建单元107基于从预测单元104输入的预测信息,适当地参考帧存储器108中所存储的预测图像的重建所需的数据并重建预测图像。然后,图像重建单元107根据重建的预测图像以及从逆量化和逆变换单元106输入的重建的预测残差来重建图像数据,并将该图像数据输入和存储到帧存储器108中。

[0086] 环内滤波单元109从帧存储器108读取重建对象图像,并进行诸如解块滤波等的环内滤波处理。然后,环内滤波单元109输入滤波处理后的图像并将该图像存储到帧存储器108中。

[0087] 编码单元110以块为单位对变换和量化单元105所生成的量化系数以及从预测单元104输入的预测信息进行熵编码,以生成代码数据。尽管没有特别指定熵编码方法,但可以使用Golomb编码、算术编码和霍夫曼编码。所生成的代码数据被输出到整合编码单元111。

[0088] 整合编码单元111将从编码单元110输入的代码数据连同上述的头代码数据一起复用以生成位流。最终,该位流经由端子112被输出到外部。

[0089] 图6A示出在第一典型实施例中要输出的位流的示例。序列头包括基量化矩阵的代码数据,该代码数据包括各元素的编码结果。注意,对基量化矩阵上的代码数据等进行编码的位置不限于此,并且将理解,位流可被配置成使得将这种数据编码在图片头部或其它头部中。如果在序列内切换量化矩阵,则可以对新的基量化矩阵进行编码以进行更新。在这种情况下,可以重写所有的量化矩阵,或者可以通过指定与要重写的量化矩阵相对应的量化矩阵的子块大小来改变这些量化矩阵的一部分。

[0090] 图3是示出利用根据第一典型实施例的图像编码设备的编码处理的流程图。

[0091] 在步骤S301中,在图像编码之前,量化矩阵存储单元103生成并存储二维量化矩阵。在本典型实施例中,量化矩阵存储单元103生成图8A所示的基量化矩阵以及根据该基量化矩阵的图8B和图8C所示的量化矩阵,并且存储该基量化矩阵和这些量化矩阵。

[0092] 在步骤S302中,量化矩阵编码单元113扫描用于在步骤S301中生成量化矩阵的基量化矩阵,并计算按扫描顺序邻接的元素之间的差,以生成一维差分矩阵。根据本典型实施例,对于图8A的基量化矩阵,通过使用图9的扫描方法来生成图10所示的差分矩阵。量化矩阵编码单元113进一步对所生成的差分矩阵进行编码以生成量化矩阵代码数据。

[0093] 在步骤S303中,整合编码单元111将图像数据编码所需的头信息连同所生成的量化矩阵代码数据一起编码并输出。

[0094] 在步骤S304中,块分割单元102以 64×64 阵列的像素的基本块为单位对以帧为单位的输入图像进行分割。

[0095] 在步骤S305中,预测单元104通过使用上述预测方法对在步骤S304中生成的以基本块为单位的图像数据进行预测处理,来生成诸如子块分割信息和预测模式等的预测信息、以及预测图像数据。在本典型实施例中,使用两个类型的块大小,即图7B所示的 32×32 阵列的像素的子块的块大小和图7A所示的 64×64 阵列的像素的子块的块大小。预测单元104进一步根据所输入的图像数据和预测图像数据来计算预测残差。

[0096] 在步骤S306中,变换和量化单元105对在步骤S305中计算出的预测残差进行正交变换以生成正交变换系数。然后,变换和量化单元105进一步通过使用量化参数和在步骤S301中生成并存储的量化矩阵对正交变换系数进行量化来生成量化系数。更具体地,变换

和量化单元105进行图7B的 32×32 阵列的像素的子块中的预测残差与 32×32 阵列的正交变换矩阵及其转置矩阵的相乘,以生成 32×32 阵列的正交变换系数。另一方面,变换和量化单元105进行图7A中的 64×64 阵列的像素的子块中的预测残差与 64×32 阵列的正交变换矩阵及其转置矩阵的相乘,以生成 32×32 阵列的正交变换系数。在本典型实施例中,变换和量化单元105通过将图8B的量化矩阵用于图7B的 32×32 阵列的子块中的正交变换系数、并且通过将图8C的量化矩阵用于与图7A的 64×64 阵列的子块相对应的正交变换系数,来对 32×32 阵列的正交变换系数进行量化。

[0097] 在步骤S307中,逆量化和逆变换单元106通过使用量化参数和在步骤S301中生成并存储的量化矩阵来对在步骤S306中生成的量化系数进行逆量化,以重建正交变换系数。此外,逆量化和逆变换单元106对正交变换系数进行逆正交变换以重建预测残差。在该步骤中,逆量化和逆变换单元106通过使用与步骤S306中所使用的量化矩阵相同的量化矩阵来进行逆量化处理。更具体地,逆量化和逆变换单元106通过使用图8B的量化矩阵来对与图7B的 32×32 阵列的像素的子块相对应的 32×32 阵列的量化系数进行逆量化处理,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元106通过使用 32×32 阵列的正交变换矩阵及其转置矩阵来对 32×32 阵列的正交变换系数进行相乘,以重建 32×32 阵列的像素的预测残差。另一方面,逆量化和逆变换单元106通过使用图8C的量化矩阵来对与图7A的 64×64 阵列的像素子块相对应的 32×32 阵列的量化系数进行逆量化处理,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元106进行 32×32 阵列的正交变换系数与 64×32 阵列的正交变换矩阵及其转置矩阵的相乘,以重建 64×64 阵列的像素的预测残差。

[0098] 在步骤S308中,图像重建单元107基于在步骤S305中生成的预测信息来重建预测图像。图像重建单元107进一步根据重建的预测图像和在步骤S307中生成的预测残差来重建图像数据。

[0099] 在步骤S309中,编码单元110对在步骤S305中生成的预测信息和在步骤S306中生成的量化系数进行编码以生成代码数据。编码单元110通过还包括其它代码数据来生成位流。

[0100] 在步骤S310中,图像编码设备判断是否针对帧中的所有基本块完成了编码。如果编码完成(步骤S310中为“是”),则处理进入步骤S311。如果编码未完成(步骤S310中为“否”),则处理以下一基本块作为对象返回到步骤S304。

[0101] 在步骤S311中,环内滤波单元109对在步骤S308中重建的图像数据进行环内滤波处理以生成滤波处理后的图像。然后,处理结束。

[0102] 通过上述结构和操作,可以在减少计算量的同时控制各频率分量的量化以提高主观图像质量。特别地,在步骤S306中,通过减少正交变换系数的数量并使用与减少的正交变换系数相对应的量化矩阵进行量化处理,可以在减少计算量的同时控制各频率分量的量化以提高主观图像质量。此外,在减少正交变换系数的数量以仅对低频部分进行量化和编码的情况下,可以通过使用如图8C那样的通过仅扩大基量化矩阵的低频部分所获得的量化矩阵来实现对低频部分最佳的量化控制。在图8C的示例中,低频部分在这里是指x坐标为0~3且y坐标为0~3的范围。

[0103] 在根据本典型实施例的结构中,为了减少代码量,仅对在生成图8B和图8C的量化矩阵时共同使用的图8A的基量化矩阵进行编码。可选地,可以对图8B和图8C的量化矩阵自

身进行编码。由于可以针对量化矩阵的各个频率分量设置独特值,因此这使得能够对各频率分量进行更精细的量化控制。可选地,作为另一结构,可以对图8B和图8C的量化矩阵设置各自不同的基量化矩阵,并且可以对各个基量化矩阵进行编码。在这种情况下,可以通过对 32×32 阵列的正交变换系数和 64×64 阵列的正交变换系数进行各自不同的量化控制,来实现主观图像质量的更周密的控制。此外,在这种情况下,可以通过代替使 8×8 阵列的基量化矩阵的左上方 4×4 部分变为八倍、而是使 8×8 阵列的整个基量化矩阵变为四倍来获得与 64×64 阵列的正交变换系数相对应的量化矩阵。这也使得能够对 64×64 阵列的正交变换系数进行更精细的量化控制。

[0104] 此外,在根据本典型实施例的结构中,尽管针对使用了归零的 64×64 阵列的子块的量化矩阵是唯一地确定的,但可以引入标识符以使得能够进行选择。例如,图6B示出如下的位流:新引入量化矩阵编码方法信息代码,以使已归零的 64×64 阵列的子块的量化矩阵编码是可选的。例如,在量化矩阵编码方法信息代码指示0的情况下,将作为独立量化矩阵的图8C用于与使用了归零的 64×64 阵列的像素的子块相对应的正交变换系数。在编码方法信息代码指示1的情况下,将作为针对不要归零的普通子块的量化矩阵的图8B用于使用了归零的 64×64 阵列的像素的子块。在编码方法信息代码指示2的情况下,代替 8×8 阵列的基量化矩阵的元素,对针对使用了归零的 64×64 阵列的像素的子块所要使用的量化矩阵的所有元素进行编码。这样可以以可选择的方式实现量化矩阵代码的量减少和对使用了归零的子块的独立量化控制。

[0105] 在本典型实施例中,通过归零所处理的子块仅是 64×64 阵列的子块。然而,通过归零所要处理的子块不限于此。例如,在与图7C或图7D所示的 32×64 或 64×32 阵列的子块相对应的正交变换系数中,可以将下半部分或右半部分的 32×32 的正交变换系数强制设置为0。在这种情况下,仅对上半部分或左半部分的 32×32 阵列的正交变换系数进行量化和编码。通过使用与图8B的量化矩阵不同的量化矩阵来进行对上半部分或左半部分的 32×32 阵列的正交变换系数的量化处理。

[0106] 此外,可以进行配置使得:以与 8×8 阵列的基量化矩阵的元素的值分开的方式设置和编码与所生成的正交变换系数中的假定对图像质量的影响最大的左上方的DC系数相对应的各量化矩阵的值。图12B和图12C示出与DC分量相对应的最左上方元素的值从图8B和图8C中的量化矩阵改变的示例。在这种情况下,可以通过除对与图8A的基量化矩阵有关的信息进行编码以外、还附加地对DC部分的位置处的指示“2”的信息进行编码,来设置图12B和图12C所示的量化矩阵。这使得能够针对对图像质量的影响最大的正交变换系数的DC分量进行更精细的量化控制。

[0107] <第二典型实施例>

[0108] 图2是示出根据本发明第二典型实施例的图像解码设备的结构的框图。在第二典型实施例中,以下将说明对在第二典型实施例中生成的代码数据进行解码的图像解码设备作为示例。

[0109] 终端201用于输入编码后的位流。

[0110] 分离解码单元202从位流中分离与解码处理有关的信息和与系数有关的代码数据,并对位流的头部分中所包括的代码数据进行解码。在本典型实施例中,分离出量化矩阵代码并将该量化矩阵代码输出到后级。分离解码单元202进行图1中的整合编码单元111的

逆操作。

[0111] 量化矩阵解码单元209从位流中解码量化矩阵代码以重建基量化矩阵,并且进一步进行根据该基量化矩阵来生成量化矩阵的处理。

[0112] 解码单元203对从分离解码单元202输出的代码数据进行解码,以重建(导出)量化系数和预测信息。

[0113] 逆量化和逆变换单元204与图1的逆量化和逆变换单元106一样,通过使用重建的量化矩阵和量化参数来对量化系数进行逆量化以获取正交变换系数,并且进一步对这些正交变换系数进行逆正交变换以重建预测残差。用于导出量化参数的信息也是由解码单元203从位流中解码得到的。进行逆量化的功能和进行逆变换的功能可以是单独配置的。

[0114] 帧存储器206存储与重建的图片有关的图像数据。

[0115] 图像重建单元205基于所输入的预测信息通过适当地参考帧存储器206来生成预测图像数据。然后,图像重建单元205根据该预测图像数据和由逆量化和逆变换单元204重建的预测残差来生成重建对象图像数据,并输出该重建对象图像数据。

[0116] 环内滤波单元207与图1的109一样对重建对象图像进行诸如解块滤波等的环内滤波处理,并输出滤波处理后的图像。

[0117] 端子208将重建的图像数据输出到外部。

[0118] 以下将说明利用图像解码设备的图像解码操作。在根据本典型实施例的结构中,在第一典型实施例中生成的位流是以帧为单位(以图片为单位)输入的。

[0119] 参考图2,从端子201输入的1帧的位流被输入到分离解码单元202。分离解码单元202从位流中分离出与解码处理有关的信息和与系数有关的代码数据,并且对位流的头部中所包括的代码数据进行解码。更具体地,分离解码单元202重建量化矩阵代码数据。在本典型实施例中,从图6A所示的位流的序列头提取量化矩阵代码数据,并将该量化矩阵代码数据输出到量化矩阵解码单元209。在本典型实施例中,提取出与图8A所示的基量化矩阵相对应的量化矩阵代码数据,并输出该量化矩阵代码数据。随后,重建图片数据的以基本块为单位的代码数据,并将该代码数据输出到解码单元203。

[0120] 量化矩阵解码单元209首先对所输入的量化矩阵代码数据进行解码,以重建图10所示的一维差分矩阵。在本典型实施例中,与第一典型实施例一样,尽管量化矩阵解码单元209通过使用图11A所示的编码表来进行解码,但编码表不限于此。可以使用与第一典型实施例中的编码表相同的任何其它编码表。此外,量化矩阵解码单元209根据重建的一维差分矩阵来重建二维量化矩阵。这里,量化矩阵解码单元209进行与根据第一典型实施例的量化矩阵编码单元113的操作相反的操作。更具体地,在本典型实施例中,通过使用图9所示的扫描方法来根据图10所示的差分矩阵重建图8A所示的基量化矩阵,并且分别存储重建的基量化矩阵。更具体地,量化矩阵解码单元209通过将差分矩阵中的各差值顺次地与上述初始值相加来重建量化矩阵的各元素。然后,量化矩阵解码单元209通过根据图9所示的扫描方法将各个重建的一维元素与二维量化矩阵的相应元素顺次关联来重建二维量化矩阵。

[0121] 此外,量化矩阵解码单元209如在第一典型实施例中那样扩大重建的基量化矩阵以生成图8B和图8C所示的两个类型的 32×32 阵列的量化矩阵。图8B的量化矩阵是通过使图8A的 8×8 阵列的基量化矩阵变为四倍(即,通过将该基量化矩阵的各元素在垂直和水平方向上重复四次)所生成的 32×32 阵列的量化矩阵。

[0122] 相比之下,图8C的量化矩阵是通过放大图8A的基量化矩阵的左上方 4×4 部分(即,通过将该部分的各元素在垂直和水平方向上重复八次)所生成的 32×32 阵列的量化矩阵。然而,所生成的量化矩阵不限于此。如果存在除 32×32 阵列以外的、在后级要逆量化的量化系数的任何其它大小,则可以生成诸如 16×16 、 8×8 和 4×4 阵列等的与要逆量化的量化系数的大小相对应的阵列量化矩阵。这些所生成的量化矩阵被存储并用于后级的逆量化处理。

[0123] 解码单元203从位流中解码代码数据,以重建量化系数和预测信息。解码单元203基于解码得到的预测信息来确定解码对象子块的大小,将重建的量化系数输出到逆量化和逆变换单元204,并将重建的预测信息输出到图像重建单元205。根据本典型实施例,不论解码对象子块的大小如何(即,大小是如图7A那样的 64×64 阵列还是如图7B那样的 32×32 阵列),对于各子块都重建 32×32 阵列的量化系数。

[0124] 逆量化和逆变换单元204通过使用量化参数和由量化矩阵解码单元209重建的量化矩阵来对所输入的量化系数进行逆量化以生成正交变换系数,并且进一步对这些正交变换系数进行逆正交变换以重建预测残差。以下将更详细地说明逆量化和逆正交变换处理。

[0125] 在选择如图7B那样的 32×32 的子块分割的情况下,逆量化和逆变换单元204通过使用图8B的量化矩阵来对由解码单元203重建的 32×32 阵列的量化系数进行逆量化,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元204进行 32×32 阵列的转置矩阵与 32×32 阵列的正交变换系数的相乘以计算 32×32 阵列的中间系数。逆量化和逆变换单元204进行 32×32 阵列的中间系数与前述 32×32 的正交变换矩阵的相乘以重建 32×32 阵列的预测残差。逆量化和逆变换单元204对 32×32 阵列的各个子块进行类似的处理。

[0126] 另一方面,在如图7A所示选择未分割的情况下,逆量化和逆变换单元204通过使用图8C的量化矩阵来对由解码单元203重建的 32×32 阵列的量化系数进行逆量化,以重建 32×32 阵列的正交变换系数。然后,逆量化和逆变换单元204进行 32×64 阵列的转置矩阵与 32×32 阵列的正交变换系数的相乘,以计算 32×64 阵列的中间系数。逆量化和逆变换单元204进行 32×64 阵列的中间系数与 64×32 的正交变换矩阵的相乘以重建 64×64 阵列的预测残差。

[0127] 重建的预测残差被输出到图像重建单元205。在本典型实施例中,根据通过由解码单元203重建的预测信息所确定的解码对象子块的大小来确定在逆量化处理中要使用的量化矩阵。更具体地,在逆量化处理中,对于图7B中的 32×32 阵列的各个子块,使用图8B的量化矩阵,并且对于图7A中的 64×64 阵列的子块,使用图8C的量化矩阵。然而,要使用的量化矩阵不限于此,并且可以使用与第一典型实施例中的变换和量化单元105以及逆量化和逆变换单元106所使用的量化矩阵相同的量化矩阵。

[0128] 图像重建单元205基于从解码单元203输入的预测信息,通过适当地参考帧存储器206来获取预测图像重建所需的数据,并且重建预测图像。在本典型实施例中,图像重建单元205与第一典型实施例的预测单元104一样使用帧内预测和帧间预测这两个不同的预测方法。如上所述,可以使用组合帧内预测和帧间预测的预测方法。与第一典型实施例一样,以子块为单位进行预测处理。

[0129] 具体预测处理与第一典型实施例中的预测单元104所使用的预测处理类似,因此将省略对预测处理的冗余说明。图像重建单元205根据通过预测处理所生成的预测图像和

从逆量化和逆变换单元204输入的预测残差来重建图像数据。更具体地,图像重建单元205通过将预测图像和预测残差相加来重建图像数据。重建的图像数据被适当地存储在帧存储器206中。在其它子块的预测时适当地参考所存储的图像数据。

[0130] 与图1的环内滤波单元109一样,环内滤波单元207从帧存储器206读取重建对象图像,并进行诸如解块滤波等的环内滤波处理。然后,滤波处理后的图像再次被输入到帧存储器206。

[0131] 帧存储器206中所存储的重建图像最终从端子208被输出到外部。例如,重建对象图像被输出到外部显示设备。

[0132] 图4是示出利用根据第二典型实施例的图像解码设备的图像解码处理的流程图。

[0133] 在步骤S401中,分离解码单元202从位流中分离出与解码处理有关的信息和与系数有关的代码数据,并对头部分中的代码数据进行解码。更具体地,分离解码单元202重建量化矩阵代码数据。

[0134] 在步骤S402中,量化矩阵解码单元209首先对在步骤S401中重建的量化矩阵代码数据进行解码,以重建图10所示的一维差分矩阵。接着,量化矩阵解码单元209根据重建的一维差分矩阵来重建二维基量化矩阵。量化矩阵解码单元209进一步扩大重建的二维基量化矩阵以生成量化矩阵。

[0135] 更具体地,在本典型实施例中,量化矩阵解码单元209基于图10所示的差分矩阵通过使用图9所示的扫描方法来重建图8A所示的基量化矩阵。量化矩阵解码单元209进一步扩大重建的基量化矩阵以生成图8B和图8C所示的量化矩阵,并存储这些量化矩阵。

[0136] 在步骤S403中,解码单元203对在步骤S401中分离出的代码数据进行解码,以重建量化系数和预测信息。解码单元203进一步基于解码得到的预测信息来确定解码对象子块的大小。在本典型实施例中,不论子块的大小如何(即,大小是如图7A那样的 64×64 阵列还是如图7B那样的 32×32 阵列),针对各解码对象子块都重建 32×32 阵列的量化系数。

[0137] 在步骤S404中,逆量化和逆变换单元204通过使用在步骤S402中重建的量化矩阵来对量化系数进行逆量化以获取正交变换系数,并且对这些正交变换系数进行逆正交变换以重建预测残差。在本典型实施例中,根据由在步骤S403中重建的预测信息确定的解码对象子块的大小来确定在逆量化处理中要使用的量化矩阵。更具体地,在逆量化处理中,对于图7B的 32×32 阵列的各个子块,使用图8B的量化矩阵,并且对于图7A的 64×64 阵列的子块,使用图8C的量化矩阵。然而,要使用的量化矩阵不限于此,只要使用与第一典型实施例的步骤S306和S307中使用量化矩阵相同的量化矩阵即可。

[0138] 在步骤S405中,图像重建单元205根据在步骤S403中生成的预测信息来重建预测图像。在本典型实施例中,与第一典型实施例的步骤S305一样,使用帧内预测和帧间预测这两个类型的预测方法。图像重建单元205进一步根据重建的预测图像和在步骤S404中生成的预测残差来重建图像数据。

[0139] 在步骤S406中,图像解码设备判断是否针对帧中的所有基本块都完成了解码。如果针对所有基本块都完成了解码,则处理进入步骤S407。如果并非针对所有基本块都完成了解码,则处理以下一基本块作为对象返回到步骤S403。

[0140] 在步骤S407中,环内滤波单元207对在步骤S405中重建的图像数据进行环内滤波处理以生成滤波处理后的图像。然后,处理结束。

[0141] 对于在第一典型实施例中生成的仅进行了低频的正交变换系数的量化和编码的子块,上述的结构和操作使得能够通过使用量化矩阵来控制针对各频率分量的量化,以对主观图像质量有所提高的位流进行解码。此外,对于仅低频的正交变换系数被量化和编码的子块,可以通过使用诸如图8C的量化矩阵等的、通过仅扩大基量化矩阵的低频部分所获得的量化矩阵,来对最佳量化控制已应用于低频部分的位流进行解码。

[0142] 在本典型实施例的结构中,尽管为了减少代码量、仅对共同地用于生成图8B和图8C的量化矩阵的图8A的基量化矩阵进行解码,但可以对图8B和图8C的量化矩阵自身进行解码。由于可以针对量化矩阵的各个频率分量设置独特值,因此这使得能够对针对各频率分量实现更精细的量化控制的位流进行解码。

[0143] 在该结构中,可以针对图8B和图8C的各个量化矩阵设置单独的基量化矩阵,并且可以对各基量化矩阵进行编码。在这种情况下,可以对 32×32 阵列的正交变换系数和 64×64 阵列的正交变换系数进行各自不同的量化控制,以对实现了主观图像质量的更周密的控制的位流进行解码。此外,在这种情况下,可以通过代替使 8×8 阵列的基量化矩阵的左上方 4×4 部分变为八倍、而使 8×8 阵列的整个基量化矩阵变为四倍,来生成与 64×64 阵列的正交变换系数相对应的量化矩阵。这使得能够还对 64×64 阵列的正交变换系数进行更精细的量化控制。

[0144] 另外,在本典型实施例的结构中,尽管针对使用了归零的 64×64 阵列的子块的量化矩阵是唯一地确定的,但可以引入标识符以使量化矩阵可选择。例如,图6B示出如下的位流:通过新引入量化矩阵编码方法信息代码,使针对使用了归零的 64×64 阵列的子块的量化矩阵编码可选择。例如,在量化矩阵编码方法信息代码指示0的情况下,则将如图8C那样的独立量化矩阵用于与使用了归零的 64×64 阵列的子块相对应的正交变换系数。在量化矩阵编码方法信息代码指示1的情况下,将如图8B那样的针对不要归零的通常子块的量化矩阵用于使用了归零的 64×64 阵列的子块。在量化矩阵编码方法信息代码指示2的情况下,代替 8×8 阵列的基量化矩阵的元素,对针对使用了归零的 64×64 阵列的子块所要使用的量化矩阵的所有元素进行编码。这样使得能够对如下的位流进行解码,其中对于该位流,可以以可选择的方式实现量化矩阵代码的量减少和对使用了归零的子块的独立量化控制。

[0145] 在本典型实施例中,尽管通过归零所处理的子块仅是 64×64 阵列的子块,但通过归零所处理的子块不限于此。例如,量化矩阵解码单元209可被配置为在与图7C或图7D所示的 32×64 或 64×32 阵列的子块相对应的正交变换系数中,在无需对下半部分或右半部分的 32×32 阵列的正交变换系数进行解码的情况下,仅对上半部分或左半部分的量化系数进行解码。在这种情况下,仅要对上半部分或左半部分的 32×32 阵列的正交变换系数进行解码和逆量化。通过使用与图8B的量化矩阵不同的量化矩阵来进行对上半部分或左半部分的 32×32 阵列的正交变换系数的量化处理。

[0146] 此外,可以进行配置,使得:以与 8×8 的基量化矩阵的元素的值分开的方式解码和设置与所生成的正交变换系数中的假定对图像质量的影响最大的左上方的DC系数相对应的各个量化矩阵的值。图12B和图12C示出与DC分量相对应的最左上方元素的值从图8B和图8C中的值改变的示例。在这种情况下,可以通过除对与图8A的基量化矩阵有关的信息进行解码以外、还对DC部分的位置处的指示“2”的信息进行解码,来设置图12B和图12C所示的量化矩阵。这使得能够对如下的位流进行解码,其中对于该位流,针对对图像质量的影响最大

的正交变换系数的DC分量应用了更精细的量化控制。

[0147] <第三实施例>

[0148] 上述典型实施例基于图1和图2所示的各处理单元由硬件装置构成这一前提。然而,图1和图2中的各处理单元所要进行的处理可以由计算机程序构成。

[0149] 图5是示出可应用于根据上述各个典型实施例的图像编码设备和图像解码设备的计算机的硬件结构的示例的框图。

[0150] 中央处理单元(CPU) 501通过使用随机存取存储器(RAM) 502和只读存储器(ROM) 503中所存储的计算机程序和数据来控制整个计算机,并且同时进行由根据上述各个典型实施例的设备进行的上述处理。更具体地,CPU 501用作图1和图2所示的各处理单元。

[0151] RAM 502包括用于临时存储从外部存储装置506加载的计算机程序和数据、以及经由接口(I/F) 507从外部获取到的数据的区域。RAM 502还包括CPU 501进行各种处理所使用的工作区。更具体地,例如,RAM 502可被分配作为帧存储器,并且可以适当地提供其它各种区域。

[0152] ROM 503存储计算机的设置数据和引导程序。操作单元504包括键盘和鼠标。计算机的用户可以对操作单元504进行操作以将各种指令输入到CPU 501。输出单元505输出CPU 501所进行的处理的结果。输出单元505例如包括液晶显示器(LCD)。

[0153] 外部存储装置506是由硬盘驱动设备代表的大容量存储信息存储装置。外部存储装置506存储操作系统(OS)、以及用于使CPU 501实现图1和图2所示的各处理单元的功能的计算机程序。外部存储装置506还可以存储要处理的图像数据。

[0154] 外部存储装置506中所存储的计算机程序和数据在CPU 501的控制下被适当地加载到RAM 502中,然后由CPU 501进行处理。诸如局域网(LAN)和因特网等的网络、投影设备、显示设备、以及其它设备可以连接到I/F 507。计算机可以经由I/F 507获取和发送出各种类型的信息。总线508连接上述处理单元。

[0155] 为了实现如此配置的处理单元的操作,CPU 501用作用以控制以上参考流程图所述的操作的中心。

[0156] (其它典型实施例)

[0157] 在将记录有用于实现上述功能的计算机程序代码的存储介质供给至系统、并且该系统读取并执行这些计算机程序代码的情况下,也可实现各个典型实施例。在这种情况下,从存储介质读取的计算机程序代码实现上述典型实施例的功能,并且存储计算机程序代码的存储介质构成本发明。本发明包括以下情况:在计算机上运行的OS基于程序代码的指令来执行实际处理的一部分或全部,并且通过该处理来实现上述功能。

[0158] 本发明也可以通过以下形式来实现。更具体地,从存储介质读取计算机程序代码,然后将这些计算机程序代码写入插入到计算机中的功能扩展卡或连接到计算机的功能扩展单元中所包括的存储器中。本发明包括以下情况:该功能扩展卡或功能扩展单元中所包括的CPU等基于计算机程序代码的指令来执行实际处理的一部分或全部,以实现上述功能中的任何功能。

[0159] 在将本发明应用于上述存储介质的情况下,存储介质存储与上述流程图的处理相对应的计算机程序的代码。

[0160] 根据前述典型实施例,在通过使得能够进行使用与将正交变换系数的一部分强制

设置为0的技术相对应的量化矩阵的量化处理、来使用该技术的条件下,可以提高主观图像质量。

[0161] 本发明不限于上述典型实施例,而且可以在不背离本发明的精神和范围的情况下以各种方式进行修改和改变。因此,添加了所附权利要求以公开本发明的范围。

[0162] 本申请要求2019年3月11日提交的日本专利申请2019-044274的权益,其通过引用而被全部包含于此。

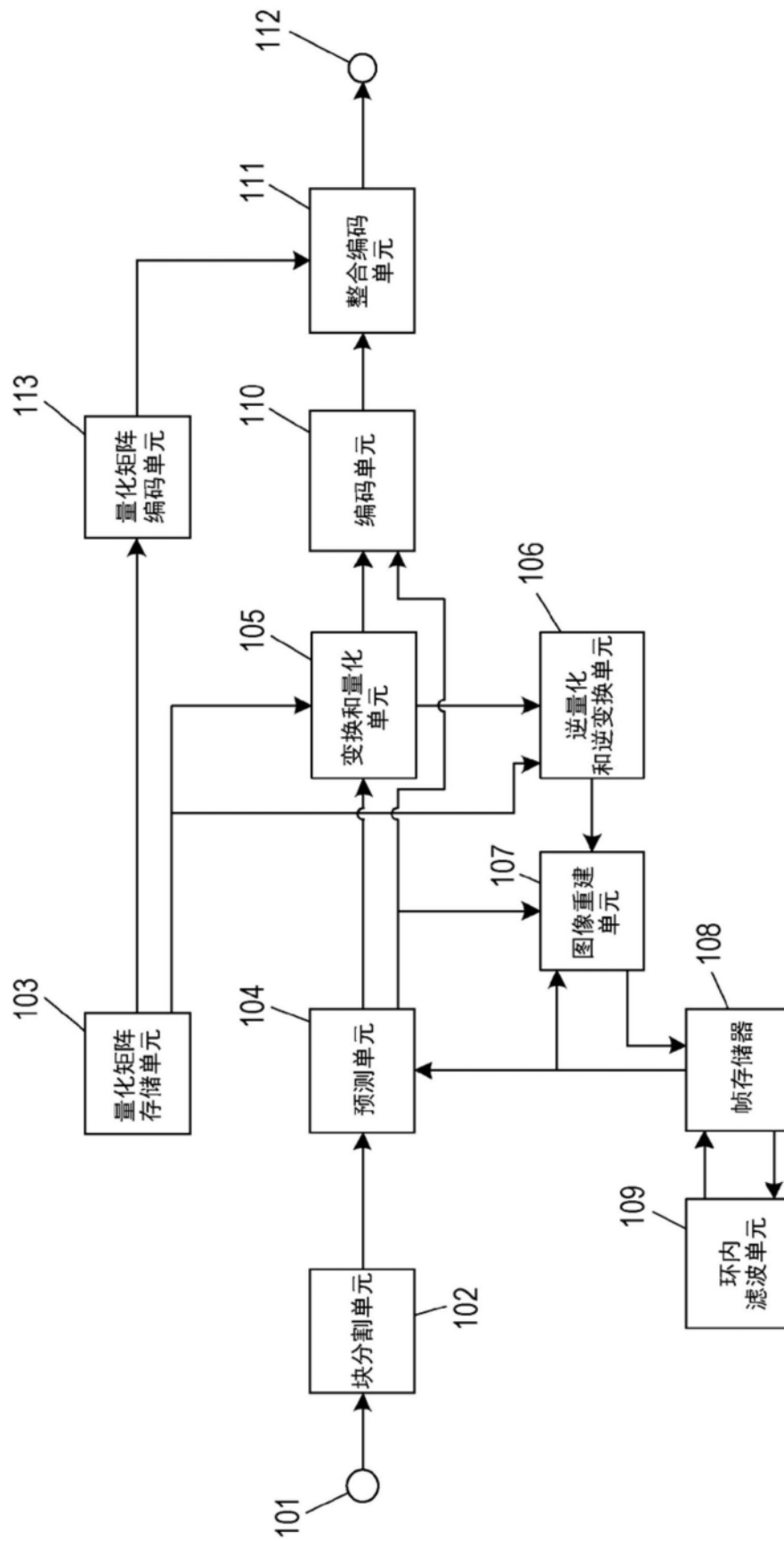


图1

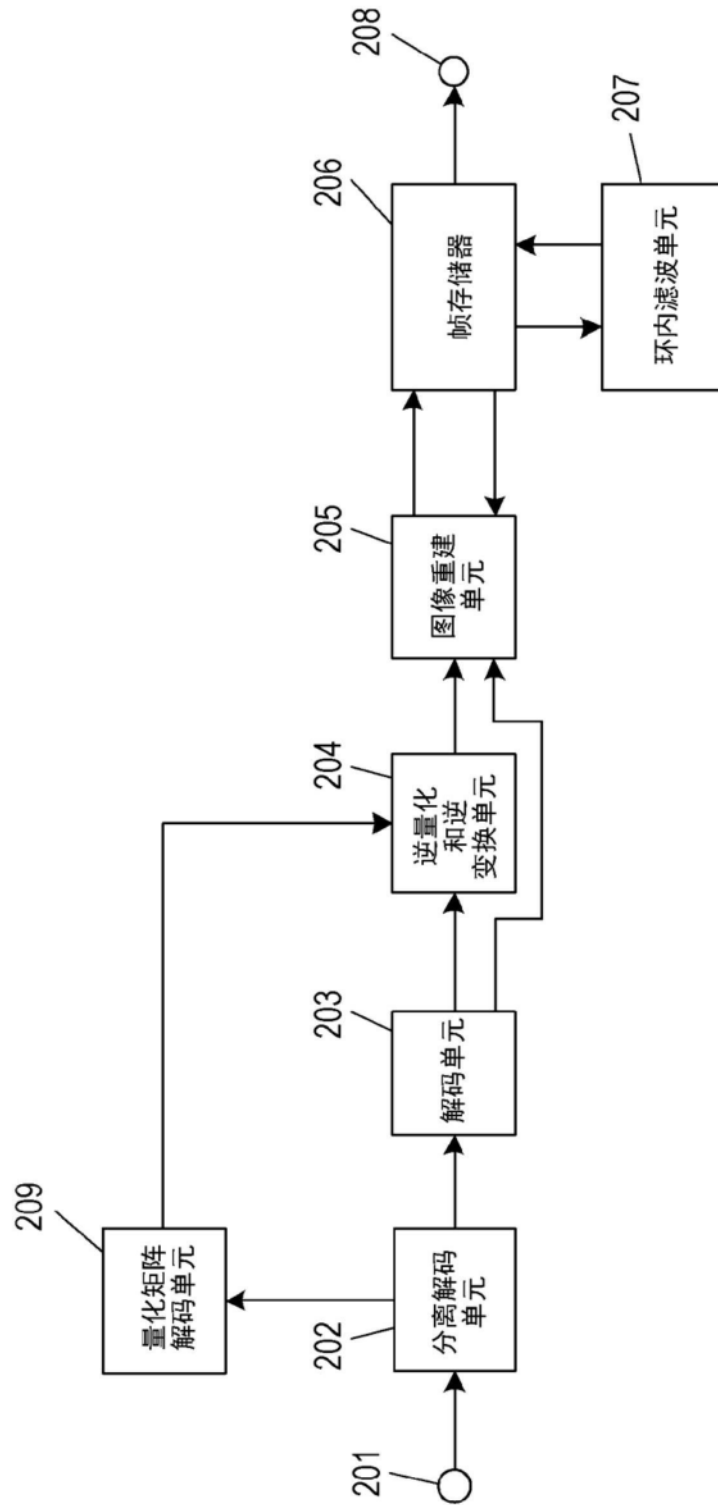


图2

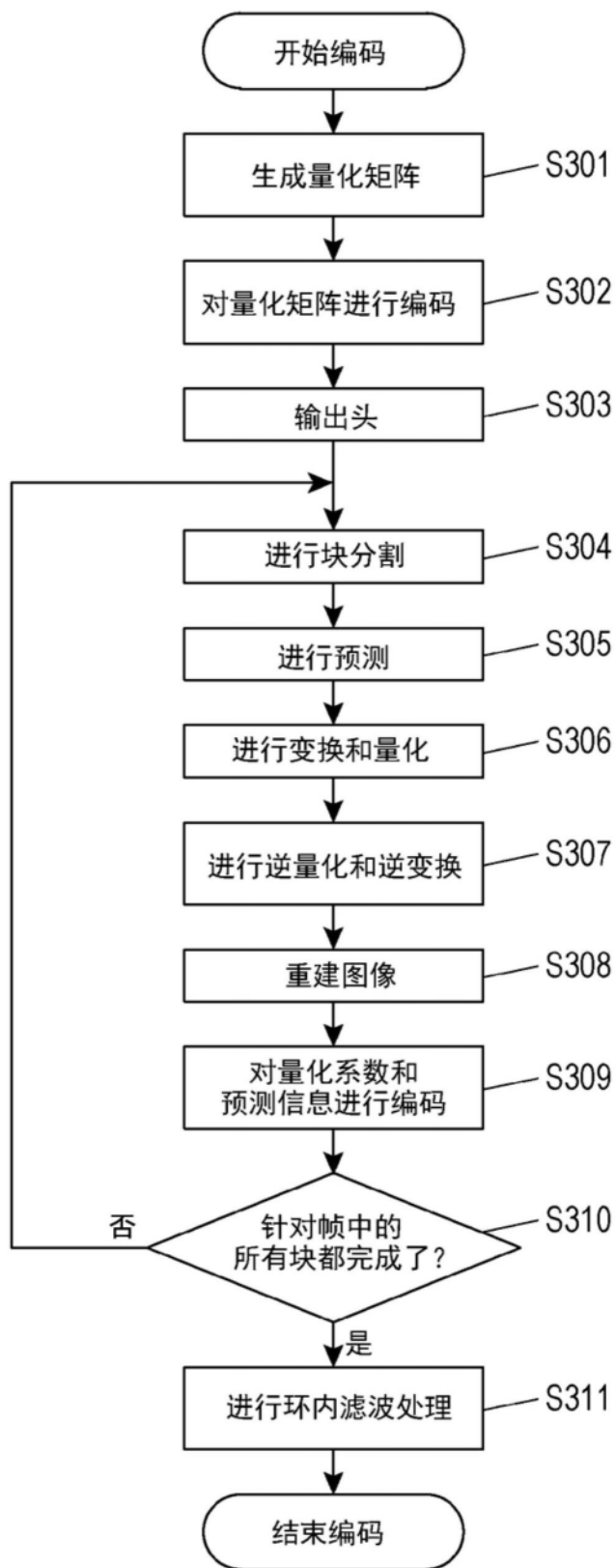


图3

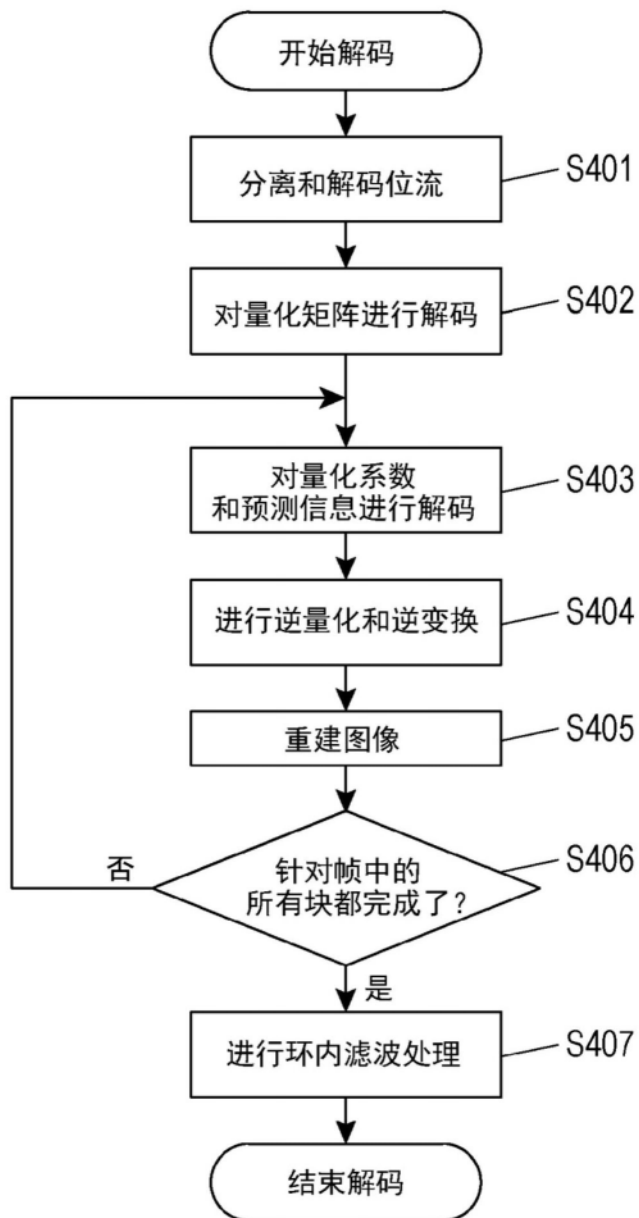


图4

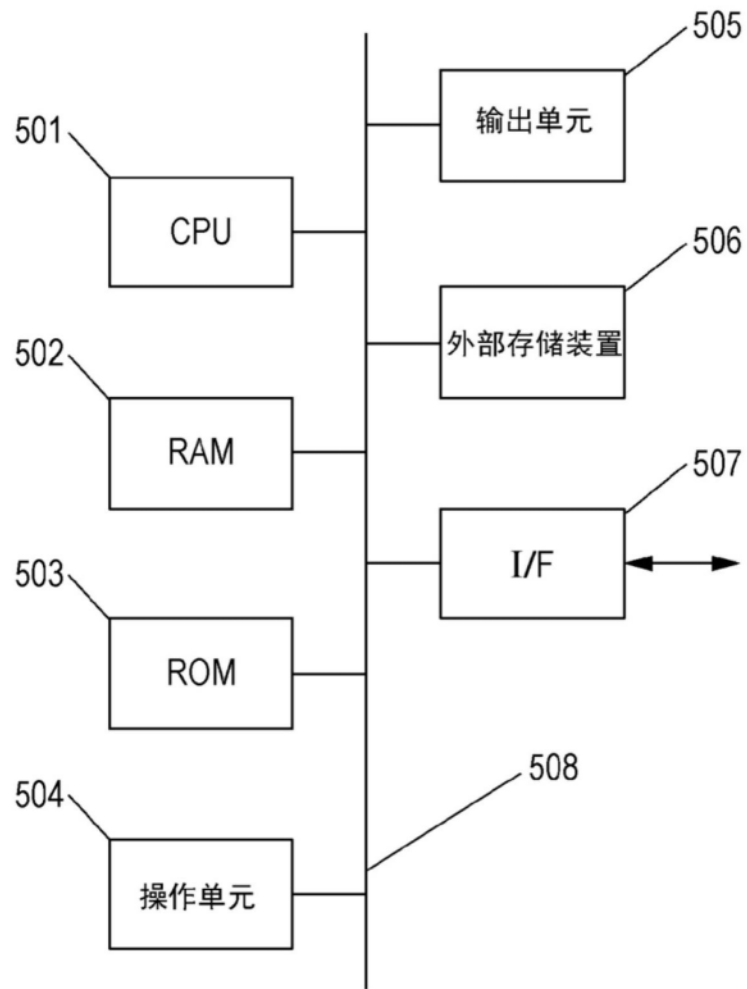


图5

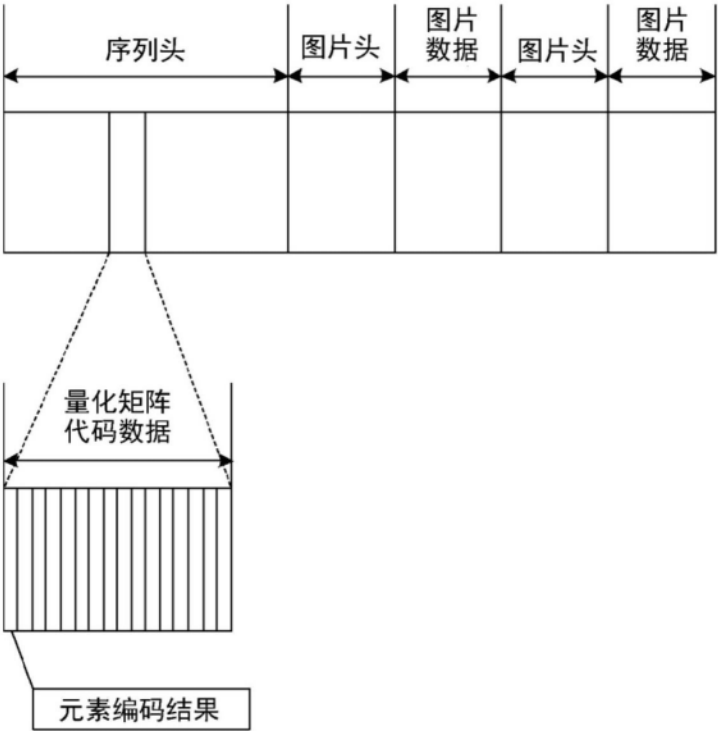


图6A

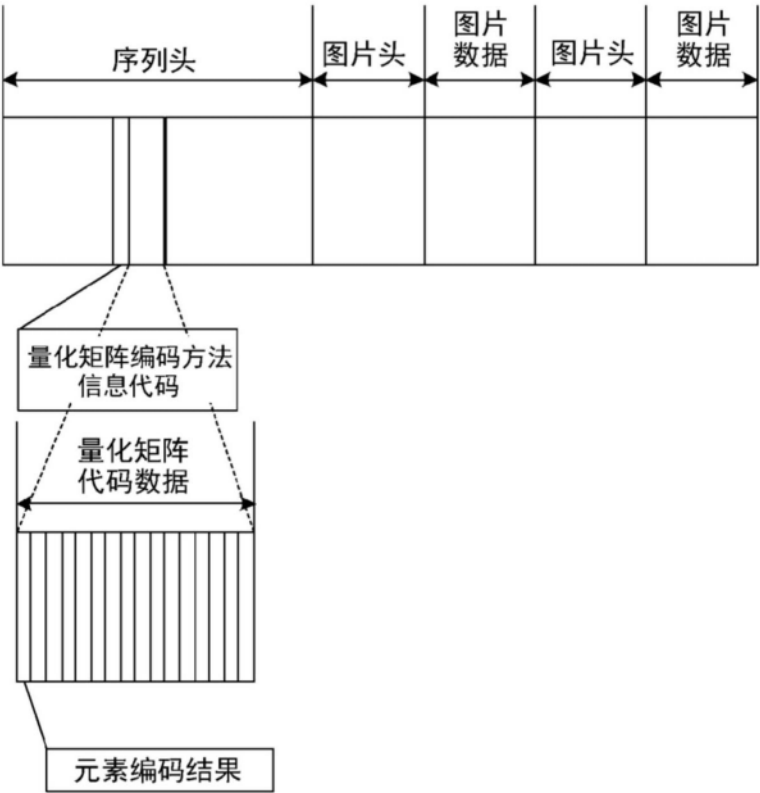


图6B

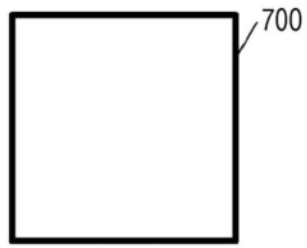


图7A

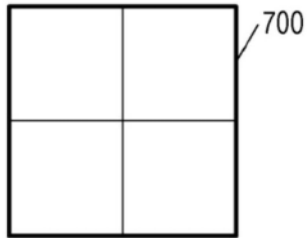


图7B

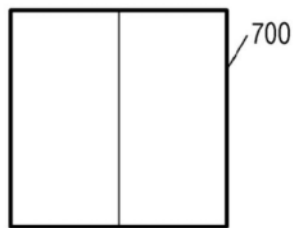


图7C

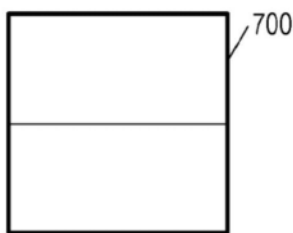


图7D

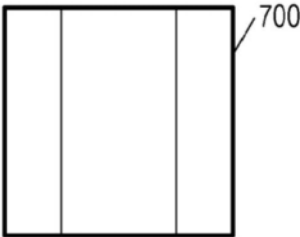


图7E

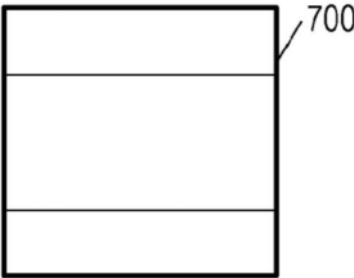


图7F

1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10
4	5	6	7	8	9	10	11
5	6	7	8	9	10	11	12
6	7	8	9	10	11	12	13
7	8	9	10	11	12	13	14
8	9	10	11	12	13	14	15

A label '800' with a pointer line indicates the right side of the table.

图8A

800

29

[illegible]

图8C

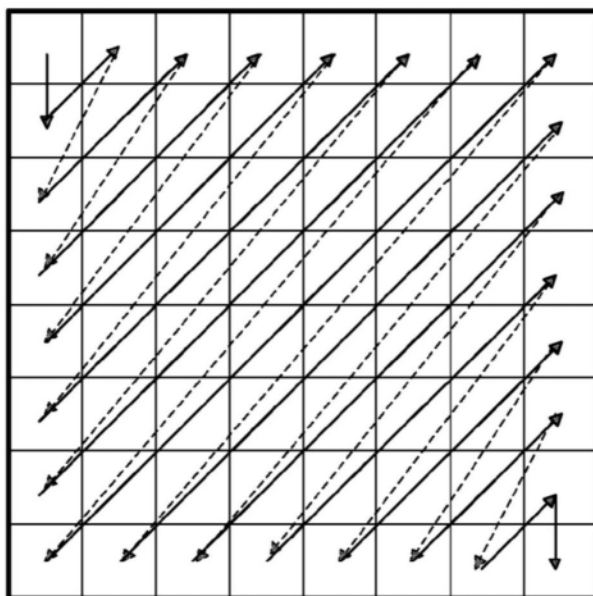


图9

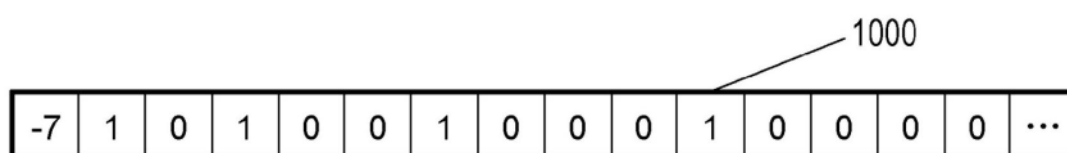


图10

编码对象值	二进制代码
...	...
-5	0001011
-4	0001001
-3	00111
-2	00101
-1	011
0	1
1	010
2	00100
3	00110
4	0001000
5	0001010
...	...

图11A

编码对象值	二进制代码
...	...
-5	0001011
-4	0001001
-3	00111
-2	0010
-1	011
0	11
1	10
2	010
3	00110
4	0001000
5	0001010
...	...

图11B

1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10
4	5	6	7	8	9	10	11
5	6	7	8	9	10	11	12
6	7	8	9	10	11	12	13
7	8	9	10	11	12	13	14
8	9	10	11	12	13	14	15

800

图12A

[illegible]

图12B

800

34