



(12)发明专利

(10)授权公告号 CN 104782125 B

(45)授权公告日 2019.03.15

(21)申请号 201380058648.8

(22)申请日 2013.09.27

(65)同一申请的已公布的文献号
申请公布号 CN 104782125 A

(43)申请公布日 2015.07.15

(30)优先权数据
2012247040 2012.11.08 AU

(85)PCT国际申请进入国家阶段日
2015.05.08

(86)PCT国际申请的申请数据
PCT/AU2013/001117 2013.09.27

(87)PCT国际申请的公布数据
W02014/071439 EN 2014.05.15

(73)专利权人 佳能株式会社
地址 日本东京都大田区下丸子3丁目30番2号

(72)发明人 克里斯托弗·詹姆斯·罗斯沃恩

(74)专利代理机构 北京魏启学律师事务所
11398

代理人 魏启学

(51)Int.Cl.
H04N 19/176(2014.01)
H04N 19/119(2014.01)
H04N 19/122(2014.01)
H04N 19/61(2014.01)
H04N 19/96(2014.01)
H04N 19/136(2014.01)
H04N 19/186(2014.01)

(56)对比文件
US 2012177116 A1,2012.07.12,
US 2012183080 A1,2012.07.19,
Bross B ET AL.High Efficiency Video Coding (HEVC) text specification draft 8.《JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16》.2012,

审查员 刘欣

权利要求书3页 说明书34页 附图26页

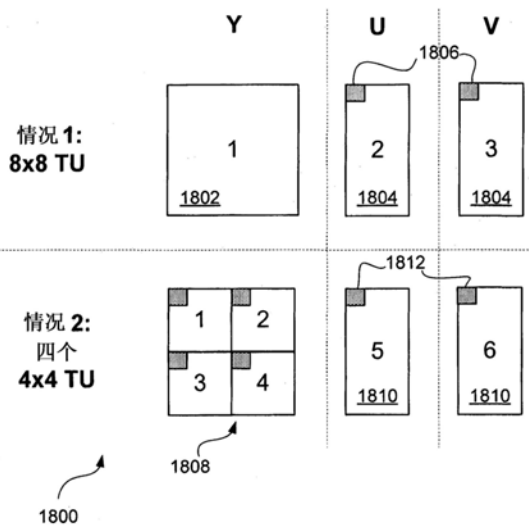
(54)发明名称

对编码单位的变换单位编码和解码的方法、设备和系统

(57)摘要

公开了一种用于从视频位流解码亮度变换和多个色度变换的方法。色度变换包含一个颜色通道的色度数据。该方法确定针对亮度变换的亮度变换跳过标志的值,其中该亮度变换标志表示是否将亮度变换的数据作为空间域表示编码在视频位流中。针对多个色度变换中的第一色度变换来确定色度变换跳过标志的值,其中该色度变换跳过标志表示是否将色度变换的数据作为空间域表示编码在视频位流中。该方法根据所确定出的亮度变换跳过标志来对亮度变换进行解码,并且根据所确定出的针对第一色度变换的色度变换跳过标志来对多个色度变换进行解码。

CN 104782125 B



1. 一种用于从针对4:2:2色度格式所配置的视频位流来对多个残差系数阵列进行逆变换的方法,所述方法包括以下步骤:

对四个亮度残差系数阵列进行解码,其中各亮度残差系数阵列与四个 4×4 亮度块中的相应的一个 4×4 亮度块相对应,所述四个 4×4 亮度块整体占据 8×8 亮度区域,

所述方法的特征在于还包括以下步骤:

在对所述四个亮度残差系数阵列进行解码之后,对第一颜色通道的两个色度残差系数阵列进行解码,其中所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行解码之后,连续地对所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行解码;

在对所述第一颜色通道的两个色度残差系数阵列进行解码之后,对第二颜色通道的两个色度残差系数阵列进行解码,其中所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行解码之后,连续地对所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行解码;以及

向各解码后的所述四个亮度残差系数阵列、各解码后的所述第一颜色通道的两个色度残差系数阵列和各解码后的所述第二颜色通道的两个色度残差系数阵列应用 4×4 逆变换。

2. 根据权利要求1所述的方法,其中,一个残差系数阵列包括对一个 4×4 块进行逆变换所需的所有系数。

3. 一种用于将多个残差系数阵列正变换成针对4:2:2色度格式所配置的视频位流的方法,所述方法包括以下步骤:

向四个亮度残差系数阵列中的各个、第一颜色通道的两个色度残差系数阵列中的各个和第二颜色通道的两个色度残差系数阵列中的各个应用 4×4 正变换;以及

对所述四个亮度残差系数阵列进行编码,其中各亮度残差系数阵列与四个 4×4 亮度块中的相应的一个 4×4 亮度块相对应,所述四个 4×4 亮度块整体占据 8×8 亮度区域,

所述方法的特征在于还包括以下步骤:

在对所述四个亮度残差系数阵列进行编码之后,对所述第一颜色通道的两个色度残差系数阵列进行编码,其中所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行编码之后,连续地对所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行编码;以及

在对所述第一颜色通道的两个色度残差系数阵列进行编码之后,对所述第二颜色通道的两个色度残差系数阵列进行编码,其中所述第二颜色通道的色度残差系数阵列中的第一

个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行编码之后,连续地对所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行编码。

4. 一种解码器,用于从针对4:2:2色度格式所配置的视频位流逆变换多个残差系数阵列,所述解码器包括:

第一解码部,用于对四个亮度残差系数阵列进行解码,其中各亮度残差系数阵列与四个 4×4 亮度块中的相应的一个 4×4 亮度块相对应,所述四个 4×4 亮度块整体占据 8×8 亮度区域,

所述解码器的特征在于还包括:

第二解码部,用于在所述四个亮度残差系数阵列的解码之后,对第一颜色通道的两个色度残差系数阵列进行解码,其中所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行解码之后,连续地对所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行解码;

第三解码部,用于在所述第一颜色通道的两个色度残差系数阵列的解码之后,对第二颜色通道的两个色度残差系数阵列进行解码,其中所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块相对应,并且所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行解码之后,连续地对所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行解码;以及

变换部,用于向各解码后的所述四个亮度残差系数阵列、各解码后的所述第一颜色通道的两个色度残差系数阵列和各解码后的所述第二颜色通道的两个色度残差系数阵列应用 4×4 逆变换。

5. 根据权利要求4所述的解码器,其中,一个残差系数阵列包括对一个 4×4 块进行逆变换所需的所有系数。

6. 一种编码器,用于将多个残差系数阵列正变换成针对4:2:2色度格式所配置的视频位流,所述编码器包括:

变换部,用于向四个亮度残差系数阵列中的各个、第一颜色通道的两个色度残差系数阵列中的各个和第二颜色通道的两个色度残差系数阵列中的各个应用 4×4 正变换;以及

第一编码部,用于对所述四个亮度残差系数阵列进行编码,其中各亮度残差系数阵列与四个 4×4 亮度块中的相应的一个 4×4 亮度块相对应,所述四个 4×4 亮度块整体占据 8×8 亮度区域,

所述编码器的特征在于还包括:

第二编码部,用于在所述四个亮度残差系数阵列的编码之后,对所述第一颜色通道的

两个色度残差系数阵列进行编码,其中所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第一颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行编码之后,连续地对所述第一颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行编码;以及

第三编码部,用于在所述第一颜色通道的两个色度残差系数阵列的编码之后,对所述第二颜色通道的两个色度残差系数阵列进行编码,其中所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列对应于与所述 8×8 亮度区域的上部相关联的 4×4 色度块,并且所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列对应于与所述 8×8 亮度区域的下部相关联的 4×4 色度块,以及在对所述第二颜色通道的色度残差系数阵列中的第一个色度残差系数阵列进行编码之后,连续地对所述第二颜色通道的色度残差系数阵列中的第二个色度残差系数阵列进行编码。

对编码单位的变换单位编码和解码的方法、设备和系统

[0001] 相关申请的交叉引用

[0002] 本申请要求2012年11月8日提交的澳大利亚专利申请2012247040的优先权,其如同在此进行了充分陈述而通过引用全部包括于此。该申请是2012年9月28日提交的澳大利亚专利申请2012232992的分案申请,其如同在此进行了充分陈述而通过引用全部包括于此。

技术领域

[0003] 本发明通常涉及数字视频信号处理,尤其涉及用于对变换单位(TU)的残差系数进行编码和解码的方法、设备和系统,其中:该变换单位(TU)包括一个或多个变换单位(TU)并且可以是针对包括4:2:2色度格式的多个色度格式所配置的,以及该变换单位(TU)的残差系数可以表示频域或空间域中的数据。

背景技术

[0004] 当前存在包括用于传输和存储视频数据的应用程序的许多视频编码用的应用程序。还开发了许多视频编码标准并且其它视频编码标准当前正在开发中。视频编码标准化的最新进展已导致形成被称为“视频编码联合小组”(JCT-VC)的组。该视频编码联合小组(JCT-VC)包括:已知为视频编码专家组(VCEG)的国际电信联盟(ITU)的电信标准化部门(ITU-T)的研究组16、问题6(SG16/Q6)的成员;以及还已知为运动图片专家组(MPEG)的国际标准化组织/国际电工委员会联合技术委员会1/小组委员会29/工作组11(ISO/IECJTC1/SC29/WG11)的成员。

[0005] 视频编码联合小组(JCT-VC)的目标是产生新的视频编码标准以显著优于目前现有的已知为“H.264/MPEG-4 AVC”的视频编码标准。该H.264/MPEG-4 AVC标准本身是针对诸如MPEG-4和ITU-T H.263等的以前的视频编码标准的大幅改进。已将开发中的新视频编码标准命名为“高效率视频编码(HEVC)”。视频编码联合小组JCT-VC还考虑由于针对高效率视频编码(HEVC)所提出的技术而产生的、在对该标准的实现进行缩放从而以高分辨率实时或以高帧频进行工作的情况下产生困难的实现挑战。一个实现挑战是用于支持使视频数据在频域和空间域之间进行变换所用的多个“变换”大小的逻辑的复杂度和大小。

发明内容

[0006] 本发明的目的是大致克服或至少改进现有配置的一个或多个缺点。

[0007] 根据本发明的一个方面,提供一种用于从视频位流解码亮度变换和多个色度变换的方法,所述多个色度变换包含单个颜色通道的色度数据,所述方法包括以下步骤:

[0008] 确定针对所述亮度变换的亮度变换跳过标志的值,其中所述亮度变换跳过标志表示是否将所述亮度变换的数据作为空间域表示而编码在所述视频位流中;

[0009] 确定针对所述多个色度变换中的第一色度变换的色度变换跳过标志的值,其中所述色度变换跳过标志表示是否将该色度变换的数据作为空间域表示而编码在所述视频位

流中;以及

[0010] 根据所确定出的所述亮度变换跳过标志的值来对所述亮度变换进行解码,并且根据所确定出的针对所述第一色度变换的所述色度变换跳过标志的值来对所述多个色度变换进行解码。

[0011] 根据另一方面,提供一种用于从视频位流解码具有亮度变换和两个色度变换的变换单位的方法,所述两个色度变换根据4:2:2色度格式包含单个颜色通道的色度数据,所述方法包括以下步骤:

[0012] 确定针对所述亮度变换的亮度变换跳过标志的值,其中所述亮度变换跳过标志表示是否将所述亮度变换的数据作为空间域表示而编码在所述视频位流中;

[0013] 确定针对所述两个色度变换中的第一色度变换的色度变换跳过标志的值,其中所述色度变换跳过标志表示是否将该色度变换的数据作为空间域表示而编码在所述视频位流中;以及

[0014] 根据所确定出的所述亮度变换跳过标志的值来对所述亮度变换进行解码,并且根据所确定出的针对所述第一色度变换的所述色度变换跳过标志的值来对所述两个色度变换进行解码。

[0015] 根据又一方面,提供一种用于从视频位流解码亮度变换和多个色度变换的方法,所述多个色度变换包含单个颜色通道的色度数据,所述方法包括以下步骤:

[0016] 分割步骤,用于将变换中的至少一个矩形变换分割成多个正方形变换;以及

[0017] 解码步骤,用于对所述正方形变换进行解码。

[0018] 期望地,所述分割步骤包括:将所有矩形变换分割成正方形变换,以使得所述解码步骤仅在正方形变换的情况下进行工作。

[0019] 根据另一方面,提供一种用于从视频位流解码包含色度残差系数的变换单位的方法,所述变换单位包含与单个色度通道相关联的至少一个色度残差系数阵列,所述方法包括以下步骤:

[0020] 确定所述变换单位的大小,其中所述大小与所述变换单位在相应的编码单位中的层级有关;

[0021] 针对所述变换单位的色度通道,使用预定的最大数量的变换来从所述视频位流解码所述至少一个色度残差系数阵列;

[0022] 针对解码后的色度残差系数阵列选择逆变换,其中所述逆变换是从逆变换的预定集合中所选择的;以及

[0023] 向各个色度残差系数阵列应用所选择的逆变换,以针对所述变换单位的色度通道来解码色度残差样本。

[0024] 在又一方面中,公开了一种用于对视频位流中所编码的颜色通道的变换单位(TU)中的区域的残差数据进行解码的方法,所述方法包括以下步骤:

[0025] 第一判断步骤,用于根据所述视频位流判断为启用变换跳过标志;

[0026] 第二判断步骤,用于判断所述区域是否是所述颜色通道中以及编码块标志(CBF)的值为1的变换单位(TU)中的第一区域,并且如果判断为所述区域是所述第一区域,则对所述变换跳过标志的值进行解码并存储,而如果判断为所述区域不是所述第一区域,则检索所述变换跳过标志的值;以及

[0027] 使用所述变换跳过标志的值来对所述区域的残差数据进行解码。

[0028] 这里,优选地,所述第一判断步骤还包括:判断为编码单位变换量化绕过标志没有启用并且变换大小为 4×4 。

[0029] 根据本发明的另一方面,提供一种用于从针对 $4:2:2$ 色度格式所配置的视频位流来对多个残差系数阵列进行逆变换的方法,所述方法包括以下步骤:

[0030] 对多个亮度残差系数阵列进行解码,其中各亮度残差系数阵列与多个 4×4 亮度块中的一个 4×4 亮度块相对应,各 4×4 亮度块配置有多个 4×4 变换单位中的一个 4×4 变换单位,多个 4×4 亮度块整体占据 8×8 亮度区域;

[0031] 在对所述多个亮度残差系数阵列进行解码之后,对第一颜色通道的多个色度残差系数阵列进行解码,其中各色度残差系数阵列与 4×4 色度块相对应,并且所述第一颜色通道的各 4×4 色度块配置有所述多个 4×4 变换单位中的两个 4×4 变换单位;

[0032] 在对所述第一颜色通道的多个色度残差系数阵列进行解码之后,对第二颜色通道的多个色度残差系数阵列进行解码,其中各色度残差系数阵列与 4×4 色度块相对应,并且所述第二颜色通道的各色度块配置有所述多个 4×4 变换单位中的两个 4×4 变换单位;以及

[0033] 向解码后的所述多个亮度残差系数阵列、解码后的所述第一颜色通道的多个色度残差系数阵列和解码后的所述第二颜色通道的多个色度残差系数阵列各自应用逆变换。

[0034] 优选地,所述多个亮度残差系数阵列中的亮度残差系数阵列的数量为4。期望地,多个色度残差系数阵列中的色度残差系数阵列的数量为2。有利地,一个残差系数阵列包括对一个 4×4 块进行逆变换所需的所有系数。

[0035] 根据另一方面,公开了一种用于将多个残差系数阵列正变换成针对 $4:2:2$ 色度格式所配置的视频位流的方法,所述方法包括以下步骤:

[0036] 向多个亮度残差系数阵列、第一颜色通道的多个色度残差系数阵列和第二颜色通道的多个色度残差系数阵列各自应用正变换;

[0037] 对所述多个亮度残差系数阵列进行编码,其中各亮度残差系数阵列与多个 4×4 亮度块中的一个 4×4 亮度块相对应,各 4×4 亮度块配置有多个 4×4 变换单位中的一个 4×4 变换单位,多个 4×4 亮度块整体占据 8×8 亮度区域;

[0038] 在对所述多个亮度残差系数阵列进行编码之后,对所述第一颜色通道的多个色度残差系数阵列进行编码,其中各色度残差系数阵列与 4×4 色度块相对应,并且所述第一颜色通道的各 4×4 色度块配置有所述多个 4×4 变换单位中的两个 4×4 变换单位;以及

[0039] 在对所述第一颜色通道的多个色度残差系数阵列进行编码之后,对所述第二颜色通道的多个色度残差系数阵列进行编码,其中各色度残差系数阵列与 4×4 色度块相对应,并且所述第二颜色通道的各色度块配置有所述多个 4×4 变换单位中的两个 4×4 变换单位。

[0040] 还公开了包括互补编码器的其它方面。

附图说明

[0041] 现在将参考以下附图来说明本发明的至少一个实施例,其中:

[0042] 图1是示出视频编码和解码系统的示意框图;

[0043] 图2A和2B构成可以实践图1的视频编码和解码系统的其中一个或这两者的通用计算机系统的示意框图;

- [0044] 图3是示出视频编码器的功能模块的示意框图；
- [0045] 图4是示出视频解码器的功能模块的示意框图；
- [0046] 图5A和5B示意性示出用于表示帧数据的色度格式；
- [0047] 图6A是编码单位的示例性变换树的示意表示；
- [0048] 图6B是配置在亮度样本网格上的示例性变换树的示意表示；
- [0049] 图6C是配置在色度样本网格上的示例性变换树的示意表示；
- [0050] 图7是表示示例性变换树的亮度通道的数据结构的数据结构的示意例示；
- [0051] 图8示出表示示例性变换树的色度通道的数据结构；
- [0052] 图9A和9B示意性示出对示例性变换树进行编码的位流结构；
- [0053] 图9C、9D和9E示意性示出对示例性变换树进行编码的替代位流结构；
- [0054] 图10是示出用于对示例性变换树进行编码的方法的示意流程图；
- [0055] 图11是示出用于对示例性变换树进行解码的方法的示意流程图
- [0056] 图12A~12C示意性示出 4×8 变换单位的残差扫描模式；
- [0057] 图13是示出用于对示例性变换单位进行编码的方法的示意流程图；
- [0058] 图14是示出用于对示例性变换单位进行解码的方法的示意流程图；
- [0059] 图15示意性示出针对 4×4 和 8×8 的变换单位(TU)的 4×4 变换的可能配置；
- [0060] 图16示意性示出实现所用的示例性色度区域；
- [0061] 图17是示出用于对示例性变换单位的残差数据进行解码的方法的示意流程图；以及
- [0062] 图18示意性示出应用于利用 4×8 (非正方形)变换的 4×8 色度区域的变换跳过操作。

具体实施方式

[0063] 本发明在任一个或多个附图中参考具有相同附图标记的步骤和/或特征的情况下,除非出现相反意图,否则这些步骤和/或特征为了本说明书的目的而具有相同的功能或操作。

[0064] 图1是示出可以利用用于对代表如下推断子分割的句法元素进行编码的技术的视频编码和解码系统100的功能模块的示意框图,其中该推断子分割针对色度通道来将变换单位子分割成多个变换。系统100包括源装置110和目的地装置130。通信信道120用于从源装置110向目的地装置130通信编码视频信息。在一些情况下,源装置110和目的地装置130分别可以包括移动电话手机,其中在这种情况下,通信信道120是无线信道。在其它情况下,源装置110和目的地装置130可以包括视频会议设备,其中在这种情况下,通信信道120通常是诸如因特网接线等的有线信道。此外,源装置110和目的地装置130可以包括范围宽的任意装置,其中这些装置包括支持空中电视广播的装置、有线电视应用、因特网视频应用并且包括在一些存储介质或文件服务器上捕获编码视频的应用。

[0065] 如图所示,源装置110包括视频源112、视频编码器114和发送器116。视频源112通常包括诸如摄像传感器等的所拍摄视频帧数据的源、存储在非瞬态记录介质上的先前拍摄到的视频序列、或者来自远程摄像传感器的视频提供。可以包括摄像传感器作为视频源112的源装置110的示例包括智能电话、视频摄录机和网络视频照相机等。视频编码器114将来

自视频源112的所拍摄帧数据转换成编码视频数据并且以下将参考图3来进一步说明。该编码视频数据通常由发送器116经由通信信道120作为编码视频信息进行发送。还可以将该编码视频数据存储在诸如“闪速 (Flash)”存储器或硬盘驱动器等的一些存储装置中,直到随后经由通信信道120进行发送为止。

[0066] 目的地装置130包括接收器132、视频解码器134和显示装置136。接收器132从通信信道120接收编码视频信息并将所接收到的视频数据传递至视频解码器134。然后视频解码器134将解码后的帧数据输出至显示装置136。显示装置136的示例包括诸如智能电话、平板计算机和计算机监视器或者单机型电视机等内的阴极射线管和液晶显示器。还可以将源装置110和目的地装置130各自的功能嵌入单个装置中。

[0067] 尽管以上说明了示例性装置,但源装置110和目的地装置130各自通常经由硬件组件和软件组件的组合可以配置在通用计算系统内。图2A示出如下的这种计算机系统200,其中该计算机系统200包括:计算机模块201;诸如键盘202、鼠标指示器装置203、扫描器226、可被配置为视频源112的照相机227、以及麦克风280等的输入装置;以及包括打印机215、可被配置为显示装置136的显示装置214、以及扬声器217等的输出装置。计算机模块201可以使用外部调制器-解调器(调制解调器)收发器装置216来经由接线221与通信网络220进行通信。可以表示通信信道120的通信网络220可以是广域网(WAN),诸如因特网、蜂窝远程通信网络或私有WAN等。在接线221是电话线的情况下,调制解调器216可以是传统的“拨号上网”调制解调器。可选地,在接线221是高容量(例如,线缆)接线的情况下,调制解调器216可以是宽带调制解调器。还可以使用无线调制解调器来进行向通信网络220的无线连接。收发器装置216可以提供发送器116和接收器132的功能,并且通信信道120可以嵌入接线221中。

[0068] 计算机模块201通常包括至少一个处理器单元205和存储器单元206。例如,存储器单元206可以具有半导体随机存取存储器(RAM)和半导体只读存储器(ROM)。计算机模块201还包括多个输入/输出(I/O)接口,其中这多个输入/输出(I/O)接口包括:音频-视频接口207,其连接至视频显示器214、扬声器217和麦克风280;I/O接口213,其连接至键盘202、鼠标203、扫描器226、照相机227以及可选的操纵杆或其它人机接口装置(未示出);以及外部调制解调器216和打印机215所用的接口208。在一些实现中,调制解调器216可以内置于计算机模块201内,例如内置于接口208内。计算机模块201还具有本地网络接口211,其中该本地网络接口211允许计算机系统200经由接线223连接至已知为局域网(LAN)的局域通信网络222。如图2A所示,局域通信网络222还可以经由接线224连接至广域网220,其中该局域通信网络222通常包括所谓的“防火墙”装置或具有相似功能的装置。本地网络接口211可以包括以太网(Ethernet™)电路卡、蓝牙(Bluetooth™)无线配置或IEEE 802.11无线配置;然而,对于接口211,可以实践多种其它类型的接口。本地网络接口211还可以提供发送器116和接收器132的功能,并且通信信道120也可以嵌入局域通信网络222中。

[0069] I/O接口208和213可以提供串行连接和并行连接中的任一个或这两者,其中前者通常根据通用串行总线(USB)标准来实现并且具有相应的USB连接器(未示出)。设置有存储装置209,并且存储装置209通常包括硬盘驱动器(HDD) 210。还可以使用诸如软盘驱动器和磁带驱动器(未示出)等的其它存储装置。通常设置有光盘驱动器212以用作数据的非易失性源。作为向计算机系统200的数据的适当源,可以使用例如光盘(例如,CD-ROM、DVD、蓝光盘(Blu-ray Disc™))、USB-RAM、便携型外部硬盘驱动器和软盘等的便携型存储器装置。通

常,HDD 210、光盘驱动器212、网络220和222中的任意还可被配置为用作视频源112、或者用作为了经由显示器214进行再现所要存储的解码视频数据的目的地。

[0070] 计算机模块201的组件205~213通常经由互连总线204并且以得到相关领域技术人员已知的计算机系统200的传统操作模式的方式进行通信。例如,处理器205使用接线218连接至系统总线204。同样,存储器206和光盘驱动器212通过接线219连接至系统总线204。可以实践所述配置的计算机的示例包括IBM-PC和兼容机、Sun SPARCstation、Apple MacTM或相似的计算机系统。

[0071] 在适当或期望的情况下,可以使用计算机系统200来实现视频编码器114和视频解码器134以及以下所述的方法,其中可以将视频编码器114和视频解码器134以及要说明的图10~13的处理作为计算机系统200内可执行的一个或多个软件应用程序233来实现。特别地,利用在计算机系统200内执行的软件233中的指令231(参见图2B)来实现视频编码器114、视频解码器134和所述方法的步骤。可以将软件指令231形成为各自用于进行一个或多个特定任务的一个或多个代码模块。还可以将软件分割成两个单独部分,其中第一部分和相应的代码模块进行所述方法,并且第二部分和相应的代码模块管理第一部分和用户之间的用户界面。

[0072] 例如,可以将软件存储在包括以下所述的存储装置的计算机可读介质中。将软件从计算机可读介质载入计算机系统200,然后由计算机系统200来执行。记录有这种软件或计算机程序的计算机可读介质是计算机程序产品。优选地,在计算机系统200中使用该计算机程序产品实现了用于实现视频编码器114、视频解码器134和所述方法的有利设备。

[0073] 通常将软件233存储在HDD 210或存储器206中。将该软件从计算机可读介质载入计算机系统200,并且由计算机系统200来执行。因而,例如,可以将软件233存储在光盘驱动器212所读取的光学可读盘存储介质(例如,CD-ROM) 225上。

[0074] 在一些示例中,将应用程序233以编码在CD-ROM 225上并且经由相应的驱动器212进行读取的方式供给至用户,或者可选地,可以由用户从网络220或222读取应用程序233。此外,还可以将软件从其它计算机可读介质载入计算机系统200。计算机可读存储介质是指将所记录的指令和/或数据提供至计算机系统200以供执行和/或处理的任何非瞬态有形存储介质。这种存储介质的示例包括软盘、磁带、CD-ROM、DVD、蓝光盘、硬盘驱动器、ROM或集成电路、USB存储器、磁光盘、或者诸如PCMCIA卡等的计算机可读卡,而与这些装置在计算机模块201的内部还是外部无关。还可以参与将软件、应用程序、指令和/或视频数据或编码视频数据提供至计算机模块401的瞬态或非有形计算机可读传输介质的示例包括:无线电或红外线传输通道及向着其它计算机或联网装置的网络接线、以及具有电子邮件发送和网站上所记录的信息等的因特网或内联网。

[0075] 可以执行上述的应用程序233的第二部分和相应的代码模块来实现要渲染或呈现在显示器214上的一个或多个图形用户界面(GUI)。通过典型地对键盘202和鼠标203进行操作,计算机系统200的用户和应用程序可以在功能上可改变的方式对界面进行操作,以将控制命令和/或输入提供至与这些GUI相关联的应用程序。还可以实现在功能上可改变的其它形式的用户界面,诸如利用经由扬声器217所输出的语音提示和经由麦克风280所输入的用户声音命令的音频界面等。

[0076] 图2B是处理器205和“存储器”234的详细示意框图。存储器234表示图2A中的计算

机模块201可以访问的(包括HDD 209和半导体存储器206的)所有存储器模块的逻辑聚合。

[0077] 在初始对计算机模块201通电的情况下,执行上电自检(power-on self-test, POST)程序250。通常将POST程序250存储在图2A的半导体存储器206的ROM 249中。有时将诸如存储有软件的ROM 249等的硬件装置称为固件。POST程序250检查计算机模块201内的硬件以确保适当工作,并且通常检查处理器205、存储器234(209,206)和通常还存储在ROM 249中的基本输入-输出系统软件(BIOS)模块251,以进行正确操作。一旦POST程序250成功运行,BIOS 251启动图2A的硬盘驱动器210。启动硬盘驱动器210使得经由处理器205执行驻留在硬盘驱动器210上的引导装入程序252。这样将操作系统253载入RAM存储器206,其中在该RAM存储器206上,操作系统253开始工作。操作系统253是处理器205可执行的系统级应用程序,以实现包括处理器管理、存储器管理、装置管理、存储管理、软件应用程序接口和通用用户界面等的各种高级功能。

[0078] 操作系统253管理存储器234(209,206),以确保计算机模块201上运行的各处理或应用程序具有在不会与分配至其它处理的内存冲突的情况下执行的充足内存。此外,必须适当使用图2A的计算机系统200中可用的不同类型的存储器,以使得各处理可以高效地运行。因此,聚合存储器234并不意图例示如何分配存储器的特定区段(除非另外说明),而是提供计算机系统200可访问的存储器的概述图以及如何使用该存储器。

[0079] 如图2B所示,处理器205包括多个功能模块,其中这多个功能模块包括控制单元239、运算逻辑单元(ALU)240和有时称为高速缓冲存储器的本地或内部存储器248。高速缓冲存储器248在寄存器区段中通常包括多个存储寄存器244~246。一个或多个内部总线241从功能上使这些功能模块相互连接。处理器205通常还具有用于使用接线218来经由系统总线204与外部装置进行通信的一个或多个接口242。存储器234使用接线219连接至总线204。

[0080] 应用程序233包括可以包含条件分支指令和循环指令的指令序列231。程序233还可以包括执行程序233时所使用的的数据232。将指令231和数据232分别存储在存储器位置228、229、230和235、236、237中。根据指令231和存储器位置228~230的相对大小,如存储器位置230中示出的指令所描述的,可以将特定指令存储在单个存储器位置中。可选地,如存储器位置228和229中示出的指令段所描述的,可以将指令分割成各自被存储在单独的存储器位置的多个部分。

[0081] 通常,向处理器205赋予一组指令,其中在该处理器205内执行该组指令。处理器205等待下一输入,其中处理器205通过执行另一组指令来对该下一输入作出反应。可以从一个或多个源提供各输入,其中该输入包括一个或多个输入装置202、203所生成的数据、从外部源经由网络220、202其中之一所接收到的数据、从存储装置206、209其中之一所检索到的数据或者从插入相应的读取器212内的存储介质225所检索到的数据(所有这些均在图2A中示出)。执行一组指令在一些情况下可能会导致输出数据。执行还可能涉及将数据或变量存储至存储器234。

[0082] 视频编码器114、视频解码器134和所述方法可以使用存储在存储器234内的相应存储器位置255、256、257中的输入变量254。视频编码器114、视频解码器134和所述方法产生存储在存储器234内的相应存储器位置262、263、264中的输出变量261。可以将中间变量258存储在存储器位置259、260、266和267中。

[0083] 参考图2B的处理器205,寄存器244、245、246、运算逻辑单元(ALU)240和控制单元

239一起工作以进行微操作序列,其中这些微操作序列是针对构成程序233的指令集中的每个指令进行“提取、解码和执行”周期所需的。各提取、解码和执行周期包括以下操作:

[0084] (a) 提取操作,用于从存储器位置228、229、230提取或读取指令231;

[0085] (b) 解码操作,其中在该解码操作中,控制单元239判断提取了哪个指令;以及

[0086] (c) 执行操作,其中在该执行操作中,控制单元239和/或ALU 240执行该指令。

[0087] 之后,可以执行针对下一指令的进一步提取、解码和执行周期。同样,可以进行存储周期,其中在该存储周期中,控制单元239将值存储至或写入存储器位置232。

[0088] 要说明的图10~13的处理中的各步骤或子处理与程序233的一个或多个区段相关联,并且通常通过处理器205中的寄存器部244、245、247、ALU 240和控制单元239一起工作以针对程序233的所述区段的指令集中的每个指令进行提取、解码和执行周期,来进行该步骤或子处理。

[0089] 图3是示出视频编码器114的功能模块的示意框图。图4是示出视频解码器134的功能模块的示意框图。如图2A和2B所示,可以使用通用计算机系统200来实现视频编码器114和视频解码器134,其中可以利用计算机系统200内的专用硬件、利用计算机系统200内可执行的软件(诸如驻留在硬盘驱动器205上并且由处理器205控制其执行的软件应用程序233的一个或多个软件代码模块等)、或者可选地由计算机系统200内可执行的专用硬件和软件的组合来实现各种功能模块。可选地,可以在诸如进行所述方法的功能或子功能的一个或多个集成电路等的专用硬件中实现视频编码器114、视频解码器134和所述方法。这种专用硬件可以包括图形处理器、数字信号处理器、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或者一个或多个微处理器和关联存储器。特别地,视频编码器114包括模块320~344,并且视频解码器134包括模块420~434,其中这些模块各自可被实现为软件应用程序233的一个或多个软件代码模块。

[0090] 尽管图3的视频编码器114是高效率视频编码(HEVC)视频编码流水线的示例,但模块320~344所进行的处理阶段对于诸如VC-1或H.264/MPEG-4 AVC等的其它视频编解码器是共通的。视频编码器114接收诸如所拍摄帧数据等的所拍摄帧数据作为一系列帧,其中各帧包括一个或多个颜色通道。各帧针对各颜色通道包括一个样本网格。使用诸如推荐ITU-R BT.709(‘YUV’)等的“颜色空间”来表示颜色信息,尽管其它颜色空间也是可以的。在使用YUV颜色空间的情况下,颜色通道包括亮度通道(“Y”)和两个色度通道(“U”和“V”)。此外,根据图像的采样或者通过应用滤波以对所拍摄帧数据进行重采样,在各颜色通道的样本网格中可能包括不同的信息量。存在已知为“色度格式”的若干采样方法,其中将参考图5A和5B来说明这些采样方法中的一些方法。

[0091] 视频编码器114将诸如帧数据310等的所拍摄帧数据的各帧分割成通常被称为“编码树块”(CTB)的区域。各编码树块(CTB)包括将帧的一部分子分割成“编码单位”(CU)的集合的层级四叉树子分割。编码树块(CTB)通常占据 64×64 个亮度样本的区域,尽管诸如 16×16 或 32×32 等的其它大小也是可以的。在一些情况下,可以使用诸如 128×128 等的甚至更大的大小。可以通过分割将编码树块(CTB)子分割成四个大小相等的区域以创建新的层级。可以递归地应用分割,从而得到四叉树层级。由于编码树块(CTB)的边尺寸始终是2的幂并且四叉树分割始终使宽度和高度减半,因此区域的边尺寸也始终为2的幂。在不进行区域的进一步分割的情况下,认为在该区域内存在“编码单位”(CU)。在编码树块的顶级不进行分

割的情况下,占据整个编码树块的区域包含通常被称为“最大编码单位”(LCU)的一个编码单位(CU)。针对各编码单位还存在诸如被 8×8 个亮度样本所占据的区域等的最小大小,尽管其它最小大小也是可以的。通常将该大小的编码单位称为“最小编码单位”(SCU)。作为该四叉树层级的结果,编码树块(CTB)整体被一个或多个编码单位(CU)所占据。

[0092] 视频编码器114针对各编码单位(CU)产生通常被称为“预测单位”(PU)的样本的一个或多个阵列。各编码单位(CU)中的预测单位(PU)的各种配置均是可以的,其中要求是预测单位(PU)没有重叠并且编码单位(CU)整体被一个或多个预测单位(PU)所占据。该方案确保了预测单位(PU)覆盖整个帧区域。

[0093] 视频编码器114通过从多路复用器模块340输出预测单位(PU)382来进行工作。差模块344输出预测单位(PU)382和来自帧数据310的编码树块(CTB)的编码单位(CU)的空间域中的数据样本的相应2D阵列之间的差,其中该差已知为“残差样本阵列”360。可以在变换模块320中将残差样本阵列360变换成频域,或者残差样本阵列360可以保持在空间域中,其中利用在变换跳过控制模块346的控制下进行工作并且使用变换跳过标志386进行信号通知的多路复用器321来进行这两个操作之间的选择。变换跳过控制模块346确定变换跳过标志386,其中该变换跳过标志386表示是否使用变换模块320来将残差样本阵列360变换成残差系数阵列362、或者是否跳过变换模块320的使用。跳过变换模块320被称为“变换跳过”。在不跳过变换的情况下,来自差模块344的残差样本阵列360被变换模块320接收到,其中该变换模块320通过应用“正变换”将残差样本阵列360从空间表示转换(或“编码”)为频域表示。变换模块320在通常被称为“变换树”的编码单位(CU)向一个或多个变换单位(TU)的层级子分割中,创建针对变换单位(TU)的各变换的被配置为残差变换阵列362的变换系数。在进行变换跳过的情况下,在编码位流312中在空间域内表示残差样本阵列360,并且绕过变换模块320,这样使得残差样本阵列360经由在变换跳过标志386的控制下进行工作的多路复用器321被直接传递至缩放和量化模块322。变换跳过控制模块346可以针对变换跳过标志386的各值(即,变换跳过或正常变换操作)测试编码位流312中所需的位率。变换跳过控制模块346可以选择在编码位流312中得到较低位率的变换跳过标志386的值,由此实现更高的压缩效率。变换跳过控制模块346所进行的各测试增加了视频编码器114的复杂度,因而期望减少变换跳过控制模块346对选择变换跳过的益处大于进行测试的成本的位率进行测试的情况的数量。例如,这可以通过将变换跳过局限于特定变换大小和块类型(诸如开发中的高效率视频编码(HEVC)标准中的针对(如以下进一步说明的)帧内预测块的仅 4×4 变换等)来实现。变换跳过功能特别用于对包含大量“高频”信息的残差样本阵列360进行编码。高频信息通常存在于包含许多清晰边缘的帧数据310中,诸如字母数字字符嵌入帧数据310的情况等。诸如计算机生成图形等的其它源的帧数据310也可以包含大量高频信息。变换模块320的DCT状变换对于包含大部分为低频信息的帧数据310(诸如从拍摄自然图像的摄像传感器所获得的帧数据等)而言最佳。因而,变换跳过功能的存在向与开发中的高效率视频编码(HEVC)标准有关的应用提供了相当大的编码效率增益。对于视频编码器114,支持变换跳过功能的一个缺陷是需要测试变换跳过模块386的两个可能模式。如以下所论述的,如参考帧内预测模块336所述,针对 4×4 个样本的残差样本阵列360大小、并且在残差样本阵列360与帧内预测块相对应的情况下,支持变换跳过功能。然而,期望针对各颜色通道单独用信号通知变换跳过标志386,因而可以利用变换跳过控制模块346来针对各颜色通道进

行单独测试。由于高频信息可以集中于一个或这两个色度通道(因而适合变换跳过)、而亮度通道可以具有最少的高频信息因而受益于使用变换,因此针对各颜色通道的单独信令是有利的。例如,有色背景上的有色文本将会得到该方案。

[0094] 针对开发中的高效率视频编码(HEVC)标准,使用改进的离散余弦变换(DCT)来实现向频域表示的转换,其中在该改进的DCT中,对传统的DCT进行修改以使用移位和加法来实现。根据所支持的变换大小,残差样本阵列360和变换系数362的各种大小均是可行的。在开发中的高效率视频编码(HEVC)标准中,针对具有诸如 32×32 、 16×16 、 8×8 和 4×4 等的特定大小的样本的2D阵列进行变换。因而,可以认为存在视频编码器114可利用的变换大小的预定集合。此外,如以上所预示的,变换大小的集合在亮度通道和色度通道之间可能有所不同。二维变换通常被配置为“可分离的”,从而能够实现为沿一个方向(例如,沿着行)在样本的2D阵列上进行工作的1D变换的第一集合,之后是沿另一方向(例如,沿着列)在从1D变换的第一集合输出的样本的2D阵列上进行工作的1D变换的第二集合。通常将宽度和高度相同的变换称为“正方形变换”。宽度和高度不同的其它变换也是可行的,并且通常被称为“非正方形变换”。这些变换的最优实现可以将行方向和列方向的一维变换组合到诸如 4×4 变换模块或 8×8 变换模块等的特定硬件或软件模块中。尺寸更大的变换需要更大量的电路来实现,尽管这些变换可能不经常使用。因此,在开发中的高效率视频编码(HEVC)标准中,存在最大变换大小 32×32 。代替重新使用从相应的正方形变换呈现的现有一维变换逻辑,由于所支持的非正方形变换大小通常将需要全新的硬件来实现,因此变换实现的集成性还引入了优先减少这些所支持的非正方形变换大小的数量。

[0095] 向亮度通道和色度通道这两者应用变换。针对变换单位(TU)在亮度通道和色度通道的处理之间存在差异,并且以下将参考图5A和5B来进行论述。各变换树占据一个编码单位(CU),并且被定义为编码单位(CU)向着在变换树(四叉树)层级的各叶节点处包含一个变换单位(TU)的层级的四叉树分解,其中各变换单位(TU)能够利用所支持的变换大小的变换。与编码树块(CTB)相似,需要编码单位(CU)整体被一个或多个变换单位(TU)占据。在变换树四叉树层级的各等级处,“编码块标志值”用信号通知在不存在进一步分割的情况下在当前层级中在各颜色通道中可能存在变换,或者用信号通知较低层级可能包含由此得到的变换单位(TU)中的至少一个变换。在编码块标志值为0的情况下,针对无论是当前层级等级或较低的层级等级处的变换树的任何变换单位(TU)的相应颜色通道,均不进行变换。在编码块标志值为1的情况下,区域包含必须具有至少一个非零残差系数的变换。这样,针对各颜色通道,0个或更多个变换可以覆盖从无直至编码单位(CU)整体的编码单位(CU)的区域的一部分。针对各颜色通道存在单独的编码块标志值。由于存在仅具有一个可能的编码块标志值的情况,因此无需对各编码块标志值进行编码。

[0096] 因而,多路复用器321的输出是残差样本阵列360和变换系数阵列362其中之一,并且在图3中简单地标记为阵列363。将阵列363输入至缩放和量化模块322,其中在该缩放和量化模块322中,根据所确定的量化参数384来对阵列363的样本值进行缩放和量化以产生残差数据阵列364。根据所确定的量化参数384的值,该缩放和量化处理导致精度损失。所确定的量化参数384的值越高,则从残差数据中丢失的信息越多。这样以降低来自视频解码器134的输出的可视质量为代价提高了视频编码器114所实现的压缩。所确定的量化参数384可以在对帧数据310的各帧进行编码期间改变,或者可以针对帧数据310的一部分(诸如整

个帧)固定。所确定的量化参数384的其它改变也是可以的,诸如以单独值量化不同的残差系数等。将残差数据阵列364和所确定的量化参数384视为向逆缩放模块326的输入,其中该逆缩放模块326反转缩放和量化模块322所进行的缩放以产生作为残差数据阵列364的重新缩放版本的重新缩放数据阵列366。开发中的高效率视频编码(HEVC)标准还支持“无损”编码模式。在无损编码处于使用中的情况下,绕过变换模块320以及缩放和量化模块322这两者,这导致将残差样本阵列360直接输入至熵编码器324。在无损模式中,还绕过了逆缩放模块326和逆变换模块328。利用熵编码器324将无损编码模式(与通常的“有损”模式相对)的选择编码在编码位流312中。图3没有示出用以实现无损模式的旁路的逻辑。绕过缩放和量化模块322导致没有对残差系数阵列362或残差样本阵列360进行量化,并且利用熵编码器324将帧数据310的准确表示编码在编码位流312中。无损编码模式导致视频编码器114的压缩效率低,因此通常仅用于诸如医学应用等的无损编码十分必要的应用中。

[0097] 还将残差数据阵列364、所确定的量化参数384和变换跳过标志386视为向熵编码器模块324的输入,其中该熵编码器模块324将残差数据阵列364的值编码在编码位流312(或“视频位流”)中。将各变换单位(TU)中的残差数据阵列364编码在通常已知为“子块”的组中。子块应优选具有相同的尺寸而与变换的大小无关,因为这样允许重新使用与子块处理有关的逻辑。通常将一个子块内的残差数据称为“数据组”(或“系数组”,即使在应用变换跳过并且“系数组”包括空间域表示而不是频域表示的情况下),并且针对各数据组,通常对数据组标志进行编码以表示该数据组内的至少一个残差数据值是否为非零。在一些情况下,可以推断出数据组标志,因而不进行编码。针对属于数据组标志值为1的数据组的各残差数据值来对标志进行编码,以表示残差数据值为非零(“有效”)还是零(“非有效”)。由于缩放和量化模块322所引起的精度损失,因此重新缩放数据阵列366与阵列363的原始值不相同。然后将来自逆缩放模块326的重新缩放数据阵列366输出至逆变换模块328。逆变换模块328进行从频域向着空间域的逆变换,以产生与视频解码器134处所产生的空间域表示相同的重新缩放变换系数阵列366的空间域表示368。多路复用器369被配置为补充多路复用器321的操作。多路复用器369被配置为接收重新缩放数据阵列366和(变换后的)空间域表示368各自作为输入,并且在变换跳过标志386的控制下,选择输入366和368其中之一作为向求和模块342的输入。

[0098] 运动估计模块338通过将帧数据310与来自(通常配置在存储器206内的)帧缓冲器模块332中所存储的帧的一个或多个集合中的先前帧数据进行比较,来产生运动矢量374。帧的这些集合已知为“参考图片列表”。然后将这些运动矢量374输入至运动补偿模块334,其中该运动补偿模块334考虑到源自于运动矢量374的空间偏移,通过对帧缓冲器模块332中所存储的样本进行滤波来产生帧间预测的预测单位(PU)376。尽管图3没有示出,但也将运动矢量374作为句法元素传递至熵编码器模块324以编码在编码位流312中。帧内预测模块336使用从求和模块342获得的样本370来产生帧内预测的预测单位(PU)378,其中该求和模块342对来自多路复用器模块340的预测单位(PU)382和多路复用器369的空间域输出进行求和。帧内预测模块336还产生帧内预测模式380,其中该帧内预测模式380被发送至熵编码器324以编码在编码位流312中。

[0099] 可以使用帧内预测或帧间预测方法来生成预测单位(PU)。帧内预测方法利用与先前进行了解码的预测单位(PU)邻接(通常位于该预测单位的上方和左方)的样本,从而在该

预测单位 (PU) 内生成参考样本。被称为“帧内预测模式”的各种方向的帧内预测都是可以的。帧间预测方法利用运动矢量以参考来自所选择的参考帧中的块。由于块可以与子样本精度 (例如, 样本的1/8) 向下对准, 因此需要进行滤波以针对预测单位 (PU) 创建参考样本块。根据由此得到的编码位流312的期望位率和由于帧内预测或帧间预测方法而引入的图像质量失真量之间的率失真权衡来判断使用哪一个方法。如果使用帧内预测, 则还根据率失真权衡来从一组帧内预测可能模式中选择一个帧内预测模式。多路复用器模块340根据通过率失真算法所作出的决定, 来选择来自帧内预测模块336的帧内预测的参考样本378或来自运动补偿块334的帧间预测的预测单位 (PU) 376。求和模块342产生总和370, 其中将该总和370输入至去块滤波器模块330。去块滤波器模块330沿着块边界进行滤波, 从而产生写入存储器206内所配置的帧缓冲器模块332的去块样本372。帧缓冲器模块332是具有用以保持来自一个或多个过去帧的数据作为参考图片列表的一部分以供将来参考的充足容量的缓冲器。

[0100] 对于开发中的高效率视频编码 (HEVC) 标准, 将熵编码器324所产生的编码位流312描绘成网络抽象层 (NAL) 单位。通常, 帧的各片包含在一个NAL单位中。熵编码器324通过进行上下文自适应二进制算术编码 (CABAC) 算法将统称为“句法元素”的残差数据阵列364、帧内预测模式380、运动矢量和其它参数编码在编码位流312中。将句法元素一起组成“句法结构”, 这些分组可以包含用以描述层级结构的递归。除诸如帧内预测模式等的序数值或者诸如运动矢量等的整数值以外, 句法元素还包括诸如用以表示四叉树分割等的标志。运动估计模块338和运动补偿模块334在精度为亮度样本的1/8的运动矢量374上进行工作, 从而使得能够进行帧数据310中的帧之间的运动的精确建模。

[0101] 尽管参考高效率视频编码 (HEVC) 视频解码流水线来描述图4的视频解码器134, 但模块420~434所进行的处理阶段对于采用诸如H.264/MPEG-4 AVC、MPEG-2和VC-1等的熵编码的其它视频编解码器而言是共通的。还可以从存储器206、硬盘驱动器210、CD-ROM、蓝光 (Blu-ray™) 盘或其它计算机可读存储介质读取编码视频信息。可选地, 可以从诸如连接至通信网络220的服务器或射频接收器等的外部源接收编码视频信息。

[0102] 如从图4看出, 将诸如编码位流312等的所接收到的视频数据输入至视频解码器134。可以从存储器206、硬盘驱动器210、CD-ROM、蓝光 (Blu-ray™) 盘或其它计算机可读存储介质读取编码位流312。可选地, 可以从诸如连接至通信网络220的服务器或射频接收器等的外部源来接收编码位流312。编码位流312包含表示要解码的所拍摄帧数据的编码句法元素。

[0103] 将编码位流312输入至熵解码器模块420, 其中该熵解码器模块420从编码位流312中提取句法元素, 并且将这些句法元素的值传递至视频解码器134中的其它块。熵解码器模块420应用上下文自适应二进制算术编码 (CABAC) 算法以从编码位流312解码句法元素。使用解码后的句法元素来在视频解码器134内重建参数。参数包括零以上的残差数据阵列450、运动矢量452、预测模式454和变换跳过标志468。将残差数据阵列450传递至逆缩放模块421, 将运动矢量452传递至运动补偿模块434并且将预测模式454传递至帧内预测模块426和多路复用器428。逆缩放模块421对残差数据进行逆缩放以创建重建数据455。在变换跳过标志468为0的情况下, 逆缩放模块421将重建数据455输出至逆变换模块422。逆变换模块422应用“逆变换”以将重建数据 (在这种情况下为变换系数) 从频域表示转换 (或“解码”)

为空间域表示,从而经由多路复用器模块423输出残差样本阵列456。在变换跳过标志468的值为1的情况下,经由多路复用器模块423输出(在这种情况下在空间域内的)重建数据455作为残差样本阵列456。逆变换模块422进行与逆变换328相同的操作。因此,逆变换模块422必须被配置为提供对符合开发中的高效率视频编码(HEVC)标准的编码位流312进行解码所需的变换大小的预定集合。在编码位流312中的信令表示使用无损模式的情况下,视频解码器134被配置为绕过逆缩放模块421和逆变换模块422(图4中未示出),这导致将残差数据阵列450直接输入至求和模块424。

[0104] 运动补偿模块434将来自熵解码器模块420的运动矢量452与来自存储器206内所配置的帧缓冲器块432的参考帧数据460相组合使用,以产生作为输出解码帧数据的预测结果的针对预测单位(PU)的帧间预测的预测单位(PU)462。在预测模式454表示使用帧内预测对当前编码单位进行编码的情况下,帧内预测模块426使用在空间上邻接预测单位(PU)的样本和还由预测模式454提供的预测方向来产生针对预测单位(PU)的帧内预测的预测单位(PU)464。根据从求和模块424输出的总和458来获得在空间上邻接的样本。多路复用器模块428根据当前预测模式454来选择针对预测单位(PU)466的帧内预测的预测单位(PU)464或帧间预测的预测单位(PU)462。利用求和模块424将从多路复用器模块428输出的预测单位(PU)466与来自逆缩放和变换模块422的残差样本阵列456相加,以产生总和458,然后将该总和458分别输入至去块滤波器模块430和帧内预测模块426。去块滤波器模块430沿着诸如变换单位(TU)边界等的数据块边界进行滤波以使可见的伪影平滑化。将去块滤波器模块430的输出写入存储器206内所配置的帧缓冲器模块432。帧缓冲器模块432提供用以保持一个或多个解码帧以供将来参考的充足存储量。还将解码帧412从帧缓冲器模块432输出至诸如显示装置136等的显示装置。

[0105] 图5A和5B各自示出分别使用4:2:0色度格式和4:2:2色度格式进行编码后的帧部分500和帧部分510的样本网格。将色度格式作为配置参数指定到视频编码器114,并且视频编码器114将“chroma_format_idc”句法元素编码在用于指定该色度格式的编码位流312中。视频解码器134从编码位流312解码“chroma_format_idc”句法元素以确定使用中的色度格式。例如,在4:2:0色度格式处于使用中的情况下,chroma_format_idc的值为1,在4:2:2色度格式处于使用中的情况下,chroma_format_idc的值为2,并且在4:4:4色度格式处于使用中的情况下,chroma_format_idc的值为3。在图5A和5B中,使用“×”符号来例示诸如亮度样本位置501等的亮度样本位置,并且使用“○”符号来例示诸如色度样本位置502等的色度样本位置。通过在所示出的点处对帧部分500进行采样,在应用4:2:0色度格式的情况下,针对各颜色通道获得样本网格。在各亮度样本位置×处,对亮度通道(“Y”)进行采样,并且在各色度样本位置○处,对色度通道(“U”和“V”)这两者进行采样。如图5A所示,针对各色度样本位置,存在亮度样本位置的2×2排列。通过对帧部分510中所示出的亮度样本位置处的亮度样本和色度样本位置处的色度样本进行采样,在应用4:2:2色度格式的情况下,针对各颜色通道获得样本网格。对帧部分510进行与帧部分500相同的样本向颜色通道的分配。与帧部分500相反,在帧部分510中存在两倍多的色度样本位置。在帧部分510中,色度样本位置每隔一个亮度样本位置进行配置。因此,在图5B中,针对各色度样本位置,存在亮度样本位置的2×1排列。

[0106] 以上以亮度样本为单位说明了变换单位的各种容许尺寸。因而,通过针对亮度通

道应用的变换所覆盖的区域将具有与变换单位尺寸相同的尺寸。由于变换单位还对色度通道进行编码,因此针对各色度通道所应用的变换将具有根据使用中的特定色度格式而改变的尺寸。例如,在4:2:0色度格式处于使用中的情况下,16×16变换单位(TU)将针对亮度通道使用16×16变换并且针对各色度通道将使用8×8变换。一个特殊情况为:在将4×4变换用于亮度通道的情况下,不存在可用于色度通道的相应可用的2×2变换(应用4:2:0色度格式的情况)或可用的4×2变换(应用4:2:2色度格式的情况)。在该特殊情况下,针对各色度通道的4×4变换可以覆盖被多个亮度变换所占据的区域。

[0107] 图6A是帧的编码树块(CTB)600内的(利用粗边框示出的)编码单位(CU)602的示例性变换树的示意表示。一个四叉树子分割将编码树块(CTB)600分割成诸如编码单位(CU)602等的四个32×32编码单位(CU)。在编码单位(CU)602内存在示例性变换树。示例性变换树包括若干四叉树子分割,从而得到例如变换单位#9(TU)604的如图6A那样编号的10个变换单位(TU)。变换单位#1~#10覆盖编码单位(CU)602整体。各四叉树子分割将区域从空间上分割成四个象限,从而得到四个较小区域。各变换单位(TU)具有与该变换单位(TU)在变换树内的层级相对应的变换深度值。层级表示在四叉树子分割终止之前所进行的四叉树子分割的数量,从而得到占据相应区域的变换单位(TU)的实例。例如,变换单位#9(TU)604占据编码单位(CU)602的区域的1/4,因此其变换深度为1。各变换单位(TU)具有通常被描述为包含变换单位(TU)的区域在亮度样本网格上的尺寸的关联大小(或“变换大小”)。该大小依赖于编码单位(CU)大小和变换深度。变换深度为0的变换单位(TU)的大小等于相应编码单位(CU)的大小。变换深度的递增使得存在于变换树中的给定变换深度处的变换单位(TU)的大小减半。由于帧包括亮度通道和色度通道,因此编码单位(CU)602占据亮度样本网格和色度样本网格这两者上的区域,因而各变换单位(TU)包括描述亮度样本网格上的亮度样本和色度样本网格上的色度样本这两者的信息。各变换单位(TU)的信息的性质依赖于视频编码器114或视频解码器134的处理阶段。在向变换模块320的输入以及逆缩放和变换模块422的输出处,残差样本阵列360和456分别包含空间域中的各变换单位(TU)的信息。由于亮度通道和色度通道之间的处理差异,因而可以将残差样本阵列360和456进一步分割成“色度残差样本阵列”和“亮度残差样本阵列”。在来自缩放和量化模块322的输出以及逆缩放和变换模块422的输入处,残差数据阵列364和450分别包含频域中的各变换单位(TU)的信息。由于亮度通道和色度通道之间的处理差异,因而可以将残差数据阵列364和450进一步分割成“色度残差数据阵列”和“亮度残差数据阵列”。

[0108] 图6B示出针对包含一组变换单位(TU)并且占据编码单位(CU)602的32×32编码单位(CU)的亮度通道的与图6A的示例性变换树相对应的示例性变换树630,其中该示例性变换树630占据亮度样本网格上的32×32亮度样本阵列。图7示出表示示例性变换树630的数据结构700。在图6B中,编号为1~10的框表示(利用几个变换单位(TU)640例示的)存在于区域632内的变换单位,并且各框包含在(虚线边框的框所示出的)无法进行进一步子分割的区域中。

[0109] 在图6B中,编号为1和9的框包含针对亮度通道的16×16变换,编号为2、3和8的框包含针对亮度通道的8×8变换,并且编号为4~7的框包含针对亮度通道的4×4变换。针对这些框各自的相应区域(虚线框)的编码块标志值为1,以表示存在变换。

[0110] 如以下将论述的,利用在位流的编码和解码各自中使用的但无需在位流中进行传

输的单独编码块标志值来指定针对各颜色通道的变换的有无。结果,从熵解码器420输出的残差系数阵列450的数量依赖于编码块标志值。在任何颜色通道中均不存在有效系数(即,所有系数都为0)的情况下,从熵解码器420输出的残差数据(系数)阵列450的数量为0。

[0111] 在图7中,圆形表示在相应圆形内示出分割变换标志值的分割变换标志值。在图7中,三角形表示在相应三角形内示出编码块标志值的编码块标志值。正方形表示对各变换进行编号以与图6B中存在的变换编号一致的变换单位。

[0112] 示例性变换树630的最高层级包含占据 32×32 编码单位(CU)的区域632。分割变换标志值702表示将区域632子分割成诸如区域634等的四个 16×16 区域,由此定义示例性变换树630的“非叶”节点。针对各 16×16 区域,诸如分割变换标志值704等的进一步分割变换标志值表示应将各个 16×16 区域进一步子分割成四个 8×8 区域。例如,如分割变换标志值704(0)所示,没有对区域634进行进一步子分割,由此定义示例性变换树630的“叶”节点。作为对比,如分割变换标志值712(1)所示,将区域638进一步子分割成四个 4×4 区域(诸如区域636等)。变换树630中所存在的递归分割结构与编码树块(CTB)中所存在的二叉树分割类似。对于亮度通道,在二叉树的“叶”节点处,利用编码块标志值来用信号通知在变换单位(TU)中存在变换,例如编码块标志值708(1)表示在区域634中存在变换710。

[0113] 由于可以使用变换来表示各区域中的残差数据,因此不允许区域小于所支持的最小变换大小(诸如针对亮度通道的 4×4 个亮度样本等)。另外,对于大于最大可用变换大小的区域,推断出分割变换标志值为1。例如,对于顶层等级为 64×64 编码单位的变换树,在所支持的最大变换大小为 32×32 个亮度样本的情况下发生向四个 32×32 区域的自动子分割(即,在编码位流312中没有用信号通知)。

[0114] 右下方的 16×16 区域642包含针对亮度通道无变换的变换单位(TU)(编号为10(十)并且带阴影),因此其相应编码块标志值716为0。

[0115] 图6C和8示出针对色度通道的与图6A的示例性变换树相对应的示例性变换树630,其中示例性变换树630是针对4:2:2色度格式所配置的,包含与针对亮度通道的变换树630相对应的针对色度通道的一组变换,并且由数据结构800来表示。由于在亮度通道和色度通道之间变换树层级凭借图6A的结构是共通的,因此在数据结构700和800之间共用分割变换标志值。与数据结构700相对比,数据结构800包括各变换分割标志值为1的编码块标志值(即,在变换树的非叶节点上)。例如,编码块标志值802(1)与变换分割标志702相关联。如果变换树的非叶节点上的编码块标志值为0,则推断出子节点上的编码块标志值为0(并且没有将相应的编码块标志编码在编码位流312中)。尽管在亮度通道中可能存在有效残差系数,但如果任何子区域中都不存在有效残差系数,则非叶区域处的编码块标志值使得能够终止针对各色度通道的变换树的较低等级处的编码块标志的编码。由于大多数信息存在于亮度通道中,因此这是典型的所拍摄帧数据的共通情形。

[0116] 在针对4:4:4色度格式配置视频编码器114和视频解码器134的情况下,大小并非预定集合的变换单位(TU)大小其中之一的任何给定变换单位(TU)的各色度通道的色度区域具有与该给定变换单位(TU)(即,在没有发生推断分割的情况下)的亮度区域相同的尺寸。在针对4:4:4色度格式配置视频编码器114和视频解码器134的情况下,大小是预定集合的变换单位(TU)大小其中之一的任何给定变换单位(TU)的各色度通道的色度区域具有小于该给定变换单位(TU)(即,在发生推断分割的情况下)的亮度区域的尺寸。

[0117] 在4:2:2色度格式处于使用中的情况下,这样得到包括针对各色度通道的色度样本中的图6C的 16×32 区域662并由此占据色度样本网格上的 16×32 区域的编码单位(CU)602。图6C示出绘制为色度样本的阵列的色度样本网格上的区域,其中各色度样本沿水平方向和垂直方向均等地间隔开(与图5B相对比)。由于使用了4:2:2色度格式,因此图6C的各色度区域相对于图6B的相应亮度区域看似为沿水平方向被压缩。分割变换标志值702(1)将与编码单位(CU)602相对应的 16×32 区域662分割成诸如 8×16 区域664等的四个 8×16 区域。 8×16 区域664具有非正方形形状,并且其大小大于诸如 4×8 区域670等的图6C所示的其它非正方形区域。对于各 8×16 区域,以与存在于针对亮度样本阵列的变换树630中的二叉树分割相类似的方式,诸如分割变换标志值704等的分割变换标志值表示是否应将相应的 8×16 区域进一步子分割成四个更小的 4×8 区域。将右上方的 8×16 区域672进一步子分割成四个 4×8 区域。编码块标志值804(1)表示四个 4×8 区域各自可以包含有效残差系数。因而,针对各 4×8 区域的编码块标志需要表示针对该相应区域存在变换。在这四个 4×8 区域中,左下方的 4×8 区域674(带阴影)包含变换单位(TU)但不包含变换,因此具有编码块标志值814(0)。诸如区域670等的其余的 4×8 区域各自具有变换,因此具有相应的编码块标志值(1)。将左上方的 8×16 区域子分割成两个相等大小的 8×8 区域。与二叉树子分割相对比,在编码位流312中不存在相应的分割变换标志。

[0118] 将在编码位流312中不存在信令的情况下将变换单位(TU)中的诸如色度通道等的通道的区域分割成(各自可以具有变换的)多个区域称为“推断分割”。该推断分割使得不必引入支持针对该情况(8×16)的非正方形变换的硬件。作为代替,使用诸如第一个 8×8 变换666等的变换。由于通过推断分割而产生的各个区域可以包含所有的零残差信息,因此需要指定在通过推断分割所产生的各区域中存在变换。因此,针对通过推断分割所产生的各区域需要单独的编码块标志值。在这种情况下,编码块标志值806和808分别与第一个 8×8 变换666和第二个 8×8 变换668相对应。对于没有发生推断分割的变换单位(TU),针对各色度通道的编码块标志值指定针对被色度通道的变换单位(TU)占据的区域的变换的有无。在发生推断分割的情况下,针对由此产生的各个区域需要单独的编码块标志值(图8中未示出),然而实现可以保留归因于整个变换单位(TU)的编码块标志值。在所有情况下都可以将单独的编码块标志值推断为“1”,或者可以通过对分割所产生的各区域的编码块标志值进行逻辑“OR(或)”运算来确定单独的编码块标志值。如果根据通过分割所产生的各区域的编码块标志值来确定单独的编码块标志值,则单独的编码块标志值可以由熵编码器324编码在编码位流312中并且由熵解码器420从编码位流312解码为附加编码块标志(图9中未示出)。在这种情况下,在单独的编码块标志值为0的情况下,可以推断出来自分割的各区域的编码块标志值为0,并且在单独的编码块标志值为1的情况下,来自分割的各区域的编码块标志由熵编码器324编码在编码位流312中并且由熵解码器420从编码位流312解码。

[0119] 16×32 区域662的左下方的 8×16 的区域680示出在上方的 8×8 推断区域682中存在 8×8 变换但在下方的 8×8 推断区域684中不存在 8×8 变换的推断分割。右下方的 8×16 阵列676(带阴影)包含变换单位(TU),但在通过推断分割所产生的任一正方形 8×8 区域中不包含变换,因此其编码块标志值810、812为0。

[0120] 存在两个色度通道得到图6C所示的结构的重叠,其中单独的编码块标志值用于指定针对各色度通道存在变换。在该实现中,针对色度的除大小 4×8 以外的区域大小来推断

分割,从而使用诸如(区域670中所包含的) 4×8 变换816等的 4×8 矩形变换,并且使得能够在其它情况(例如, 8×8 、 16×16)下重新使用现有的正方形变换。因而,可以认为存在分割成两个区域的预定区域大小(诸如 8×16 和 16×32 等)的集合,因而可以使用(大小为 8×8 和 16×16)两个变换。发生推断分割的区域大小的预定集合的不同定义也是可以的,并且将允许使用现有的正方形变换和矩形变换的不同组合。针对特定实现,还可以始终进行分割的推断,其中在这种情况下,针对色度4:2:2颜色通道没有引入矩形变换。在这种情况下,发生推断分割的区域大小的预定集合包含所有可能的色度区域大小(例如,针对4:2:2色度格式的 4×8 、 8×16 和 16×32 或者针对4:4:4色度格式的 4×4 、 8×8 、 16×16 和 32×32)。

[0121] 图16是示出针对4:2:2色度格式的所有可能色度区域大小(4×8 、 8×16 和 16×32)的“始终”推断分割的实现所用的“无矩形变换”的示例的示意表示。如图16所示,针对各色度区域的标记1(一)和2(二)是通过推断分割而产生的。

[0122] 在4:2:0色度格式处于使用中的情况下,针对变换单位(TU)中的任一色度区域没有发生推断分割,因此针对各色度通道的变换的最大数量始终为1(针对各色度通道的编码块标志值控制是否发生色度变换)。

[0123] 尽管独立于亮度通道和色度通道之间的差来对视频编码器114和视频解码器134进行说明,但由于色度格式而产生的不同样本网格使得模块的差异成为必要。实际实现针对亮度通道和色度通道可以具有单独的“处理路径”。因而,这种实现可以使亮度样本的处理和色度样本的处理解耦。由于编码位流312针对亮度通道和色度通道这两者是单个位流,因此不使熵编码器324和熵解码器420解耦。另外,诸如帧缓冲器332、432等的单个帧缓冲器保持亮度样本和色度样本,因而没有解耦。然而,模块322~330和334~340以及模块422~430和434可以使亮度处理和色度处理解耦,从而使得能够进行针对亮度和色度具有单独逻辑的实现,由此创建“亮度处理路径”和“色度处理路径”。

[0124] 特定实现可以推断变换单位(TU)的色度通道的 16×32 区域向两个 16×16 区域的分割,但无法推断针对 8×16 情况和 4×8 情况的分割。代替能够依赖于现有技术中已成熟的4、8或16点变换逻辑,这种实现使得不必向色度处理路径引入32点变换逻辑。

[0125] 图9A和9B示出可用于编码或表示变换树的层级的句法结构。在变换树的非叶节点处,根据诸如数据结构700和800等的的数据结构递归地扩展句法结构900,以定义存在于编码位流312的与变换树相对应的部分中的句法元素。在变换树的(在变换树中没有发生进一步子分割的)叶节点处,句法结构930定义了存在于编码位流312的该部分中的句法元素。通常,存在针对亮度的一个数据结构和针对色度的两个数据结构,尽管诸如用于对阿尔法(alpha)通道或深度映射进行编码等的其它数据结构也是可以的。可选地,诸如在色度通道共用一个数据结构并且在色度通道之间能够共用编码块标志值的情况下等,可以利用较少的数据结构。变换树非叶节点句法结构902定义了针对诸如变换树630等的变换树的一个层级的编码。分割变换标志910对诸如分割变换标志值702等的分割变换标志值(1)进行编码。该值表示:变换树非叶节点句法结构902包括包含变换树非叶节点句法结构902或变换树叶节点句法结构932的附加实例或者“子节点”的较低层级。编码块标志912对针对“U”色度通道的编码块标志值802(1)进行编码,并且编码块标志914对针对“V”色度通道的另一编码块标志值进行编码。如果变换树非叶节点句法结构902定义变换树层级的顶层等级,则存在编码块标志912、914。如果变换树非叶节点句法结构902没有定义变换树层级的顶层等级,则

在存在变换树层级的母等级中的相应编码块标志并且这些编码块标志的值为1的情况下,仅存在编码块标志912、914。由于在变换树630中(相对于顶层层级)存在下层层级,因此发生四叉树子分割。该子分割得到包括在变换树非叶节点句法结构902中的(利用编号为0~3的变量“blkIdx”(块索引)标识的)四个变换树句法结构916、918、920、922。

[0126] 句法结构930定义了针对变换树叶节点932的叶节点的编码(即,没有发生进一步子分割)。分割变换标志940对诸如分割变换标志值704等的分割变换标志值(0)进行编码。

[0127] 在相应区域大于最小大小的情况下,仅对分割变换标志进行编码。例如,针对(与所支持的最小亮度变换大小相对应的)4×4个亮度样本的区域,区域636具有最小容许大小,因而推断出变换分割标志值714为0,并且针对相应的变换树句法结构不进行分割变换标志的编码。

[0128] 对于区域636,使用4×8色度变换来对色度残差样本进行变换,因而不存在推断变换分割。可以存在诸如编码块标志942和编码块标志946等的编码块标志,以用信号通知存在针对各个色度通道的变换。编码块标志950用信号通知存在针对亮度通道的变换。在变换单位(TU)句法结构952中存在针对亮度通道和色度通道(在存在的情况下)的残差系数。如果编码块标志950的值为1,则在编码位流312中存在用于对亮度变换所用的残差系数或在跳过该变换的情况下的残差样本进行编码的亮度变换跳过标志964和亮度残差数据块954。亮度变换跳过标志964的值表示是使用(在正常操作中)还是绕过(在变换跳过操作中)视频编码器114中的变换模块320和视频解码器134中的逆变换模块422。如果针对各色度通道的编码块标志的值为1,则在编码位流312中存在相应的色度变换跳过标志966和968以及色度残差块956和960。变换跳过标志966用信号通知针对色度残差块956的变换跳过模式,并且变换跳过标志968用信号通知针对色度残差块960的变换跳过模式。在没有发生推断变换分割的情况下,在编码位流312中不存在编码块标志944和948以及色度残差块958和962。在没有发生推断变换分割的情况下,针对各色度通道的变换跳过标志由此用信号通知区域636整体中的针对相应色度通道的变换跳过模式。

[0129] 对于区域664,使用两个8×8色度变换来对色度残差样本进行变换,因而存在推断变换分割。编码块标志942和946(在存在的情况下)用信号通知针对第一个8×8变换666的各色度通道存在8×8变换。编码块标志944和948(在存在的情况下)用信号通知针对第二个8×8变换668的各色度通道存在8×8变换。如果编码块标志944的值为1,则在编码位流312中存在色度残差块958。如果编码块标志948的值为1,则在编码位流312中存在色度残差块962。变换跳过标志966用信号通知针对色度残差块956和958的变换跳过模式,并且变换跳过标志968用信号通知针对色度残差块960和962的变换跳过模式。在存在推断变换标志的情况下,根据在不存在推断变换分割的情况下的表现方式,针对各色度通道的变换跳过标志由此用信号通知区域664整体中的针对相应色度通道的变换跳过模式。

[0130] 如图9B所示的句法结构930示出针对推断变换分割以邻接方式进行编码后的各色度通道的第一变换和第二变换。可选地,可以使用以邻接方式对各色度通道的句法元素进行编码或者对交错有其它句法元素的各色度通道的句法元素进行编码等的其它配置。

[0131] 图9C、9D和9E示出可用于编码或表示变换树的层级的替代句法结构9100。在变换树的非叶节点处,根据诸如数据结构700和800等的数据结构递归地扩展替代句法结构9100,以定义存在于编码位流312的与变换树相对应的部分中的句法元素。针对包括叶节点

的变换树中的各节点,存在替代句法结构9100的实例,其中这些叶节点各自包含变换单位(TU)。在发生“推断分割”以针对各色度通道对变换单位(TU)进行子分割的情况下,句法结构9130针对通过推断分割而产生的第一个子区域(例如,在4:2:2色度格式处于使用中的情况下的色度区域的上半部分或在4:4:4色度格式处于使用中的情况下的色度区域的左上1/4部分),定义了存在于编码位流312的该部分中的句法元素。此外,句法结构9160针对通过推断分割而产生的后续子区域(例如,在4:2:2色度格式处于使用中的情况下的色度区域的下半部分的又一子区域或者在4:4:4色度格式处于使用中的情况下的色度区域的其余三个子区域),定义了存在于编码位流312的该部分中的句法元素。“第一个”子区域和“后续”子区域(例如,第二个以及可能的第三个或第四个子区域)的概念隐含在四叉树内的区域中的子区域的扫描顺序中。该扫描顺序是:首先从左向右其次从上向下穿过这些子区域。通常,存在针对亮度的一个数据结构和针对色度的两个数据结构,尽管诸如用于对阿尔法通道或深度映射进行编码等的附加数据结构也是可以的。可选地,诸如在色度通道共用一个数据结构并且在色度通道之间能够共用编码块标志值的情况下等,可以利用较少的数据结构。变换树句法结构9102定义了针对诸如变换树630等的变换树的一个层级的编码。

[0132] 对于诸如变换树630等的变换树的非叶节点处的变换树句法结构9102的实例,分割变换标志9110对诸如分割变换标志值702等的分割变换标志值(1)进行编码。该值表示变换树句法结构9102的实例包括包含变换树句法结构9102的附加实例或“子节点”的较低层级。编码块标志9112根据编码块标志912的描述来对编码块标志值进行编码。编码块标志9114根据编码块标志914的描述来对编码块标志值进行编码。由于在变换树630(相对于顶层层级)中存在较低的层级,因此发生四叉树子分割。该子分割得到变换树节点句法结构9102中所包括的(利用编号为0~3的“blkIdx”变量标识的)四个变换树句法结构9116、9118、9120、9122。各个变换树句法结构9116、9118、9120、9122是变换树句法结构9102的另一实例。在变换树句法结构9102中将不存在用于对亮度变换所用的残差系数或在跳过该变换的情况下的残差样本进行编码的编码块标志9124和亮度变换单位部9126。

[0133] 实现还可以对变换树句法结构9102进行配置,以使得将编码块标志9124和亮度变换单位部9126(在存在的情况下)放置在变换树句法结构9102中的较前位置,诸如放置在编码块标志9114和变换树句法结构9116之间等。

[0134] 对于诸如变换树630等的变换树的叶节点处的变换树句法结构9102的实例,分割变换标志9110对诸如分割变换标志值704等的分割变换标志值(0)进行编码。因而,变换树句法结构9102的实例与变换树930中的变换单位(TU)相对应。变换单位(TU)具有根据诸如编码单位(CU)602等的包含变换单位(TU)的编码单位(CU)以及变换深度所确定的大小。编码块标志9112对编码块标志值(1)进行编码,以表示通过针对“U”色度通道的推断分割所产生的任何色度区域均可以具有编码块标志值(1)。如果编码块标志9112对值0进行编码,则通过针对“U”色度通道的推断分割所产生的各色度区域的编码块标志值具有被推断为0的编码块标志值。即使在编码块标志9112对值1进行编码的情况下,实现仍可以对通过推断分割所产生的各色度区域的值为0的编码块标志进行编码。因此,实现可以将编码块标志9112从编码位流312中省略,代替地始终针对所省略的编码块标志9112推断编码块标志值(1)。编码块标志9114以与编码块标志9112相同的方式对“V”色度通道的另一编码块标志值进行编码。对于与发生向四个色度区域的推断分割(色度残差系数阵列的最大数量为4)的大小

一致的变换单位 (TU) 大小,在变换树节点句法结构9102中包括(利用“blkIdx”0~3标识的)四个变换树句法结构9116、9118、9120、9122。对于与发生向两个色度区域的推断分割(色度残差系数阵列的最大数量为2)的大小一致的变换单位 (TU) 大小,在变换树节点句法结构9102中包括诸如(利用“blkIdx”0和1标识的)变换树句法结构9116、9118等的两个变换树句法结构。各个变换树句法结构9116、9118、9120、9122是针对色度句法结构9132的变换树的实例。编码块标志9124对诸如编码块标志值708等的编码块标志值进行编码,从而指定针对变换单位 (TU) 的亮度通道的变换的有无。变换单位9126的亮度部分将亮度变换跳过标志编码为变换跳过标志9127并且将亮度残差系数阵列编码为亮度残差句法元素9128。

[0135] 在发生推断分割的情况下仅针对第一个色度区域(或“子区域”)而存在的色度句法结构9132的变换树包括变换树句法结构930的句法的缩小集合。编码块标志9142对色度区域的“U”色度通道的编码块标志值进行编码。编码块标志9144对色度区域的“V”色度通道的编码块标志值进行编码。变换单位 (TU) 9146的色度部分对变换单位 (TU) 句法结构952的子集进行编码。变换单位 (TU) 9146的色度部分对包含一个颜色通道的色度数据的色度变换进行编码。如果编码块标志9142的值为1,则针对“U”色度通道将色度变换以色度残差系数阵列的形式编码为色度残差句法元素9150,并且如果编码块标志9144的值为1,则针对“V”色度通道将色度变换以色度残差系数阵列的形式编码为色度残差句法元素9152(统称为“色度变换”所用的残差系数阵列)。针对通过推断分割而产生的各色度区域,变换跳过标志9148与色度残差句法元素9150相关联并且对“U”色度通道的变换跳过标志值进行编码。针对通过推断分割而产生的各色度区域,变换跳过标志9151与色度残差句法元素9152相关联并且对“V”色度通道的变换跳过标志值进行编码。该关联是将变换跳过标志编码在包括相应的残差句法元素的“残差编码”句法结构中。

[0136] 在发生推断分割的情况下仅针对除第一个色度区域(或“子区域”)以外的色度区域而存在的色度句法结构9162的变换树包括变换树句法结构930的句法的缩小集合。编码块标志9172对色度区域的“U”色度通道的编码块标志值进行编码。编码块标志9174对色度区域的“V”色度通道的编码块标志值进行编码。变换单位 (TU) 9176的色度部分对变换单位 (TU) 句法结构952的子集进行编码。如果编码块标志9172的值为1,则变换单位 (TU) 9176的色度部分针对“U”色度通道将色度残差系数阵列编码为色度残差句法元素9180。如果编码块标志9174的值为1,则变换单位 (TU) 9176的色度部分针对“V”色度通道将色度残差系数阵列编码为色度残差句法元素9182。根据变换跳过标志9148来确定针对与各色度残差句法元素9180相对应的区域的变换跳过模式。根据变换跳过标志9151来确定针对与各色度残差句法元素9182相对应的区域的变换跳过模式。这些实现可以利用诸如寄存器246等的硬件寄存器或者存储器206来存储来自第一个色度区域的变换跳过标志以用于后续的子区域。

[0137] 如图9D和9E所示的句法结构9130和9160针对推断变换分割示出紧挨各色度通道的第一和第二色度残差系数阵列之后进行编码得到的第一和第二编码块标志。可选地,可以使用诸如针对各色度通道以邻接方式对编码块标志和色度残差系数阵列进行编码等的其它配置。

[0138] 尽管通过将 8×16 区域664分割成两个 8×8 区域来例示推断变换分割,但替代实现可以针对其它区域进行该分割。例如,一些实现可以推断 16×32 区域向两个 16×16 区域的分割。这些实现有利地使得不必进行色度处理路径中的32点1D变换。由于在应用4:2:0色度

格式时针对色度处理路径不要求32点1D变换,因此将针对32点1D变换的要求从色度处理路径完全去除。由此,使用单独的处理电路以使亮度通道和色度通道解耦的实现可以在色度处理电路中实现较低的实现成本。

[0139] 存在针对各亮度样本位置具有一个色度样本位置的4:4:4色度格式。因此,利用该格式,针对色度格式和亮度格式的变换可以具有相同大小。在亮度处理路径中最大变换大小为 32×32 的情况下,这需要将 32×32 变换引入色度处理路径以进行解耦实现。特定实现可以推断针对各色度通道的用以将 32×32 区域分割成四个 16×16 区域的分割,从而使得能够将现有的 16×16 变换重新用在色度处理路径中。由于在针对4:4:4色度格式的色度处理路径中将仅使用 32×32 变换,因此推断针对各色度通道的用以将 32×32 区域分割成四个 16×16 区域的分割将使得能够将 32×32 变换从色度处理路径去除,从而减少所需的处理电路。这些实现针对各色度通道将需要四个编码块标志值,并且由此将多达四个编码块标志在编码位流312中编码在针对各色度通道的句法结构930中。

[0140] 支持4:2:2色度格式的实现还可以推断针对各色度通道的用以将 32×16 区域分割成四个 8×16 区域的分割。这些实现针对各色度通道需要四个编码块标志值,由此将四个编码块标志在编码位流312中编码在针对各色度通道的句法结构930中,因而在变换单位(TU)句法结构952中可以引入“CU3”、“CU4”、“CV3”和“CV4”编码块标志(图9B中未示出)。这些实现避免了将32点变换逻辑引入色度处理路径,并且在不对 8×16 区域进行子分割的情况下,这些实现可以将要求变换的(亮度通道中的)大小为 16×16 的变换单位(TU)所需的 8×16 变换逻辑重新用于针对色度通道的大小为 8×16 的变换。

[0141] 图10是示出用于通过对变换树非叶节点句法结构902和变换树叶节点句法结构932进行编码来对变换单位(TU)进行编码的方法1000的示意流程图。参考变换单位(TU)的色度通道来说明方法1000,然而方法1000还可应用于变换单位(TU)的任何色度通道。由于变换树非叶节点句法结构902和变换树叶节点句法结构932描述变换树中的一个节点,因此方法1000将变换树的一个节点编码在编码位流312中。例如,可以以硬件或利用在处理器205上可执行的软件来实现方法1000。最初针对变换树的顶层等级调用方法1000,并且方法1000能够(递归地)调用自身以对变换树的子节点进行编码。确定变换单位大小的步骤1002根据包含变换树的编码单位(CU)大小和变换单位(TU)的变换深度值来确定变换树中的变换单位(TU)的大小。在变换树的顶层等级调用方法1000的情况下,将变换深度值设置为0,否则利用方法1000的母实例来提供变换深度值。如果变换深度值小于最大容许变换深度,则将诸如分割变换标志值702等的分割变换标志值作为分割变换标志910编码在编码位流312中。

[0142] 在分割变换标志值为1时,仅在变换树层级的母节点具有相应的编码块标志值(1)的情况下,才针对各色度通道进行色度编码块标志912和914的编码。然后,方法1000针对变换树的(利用变换树句法结构916、918、920和922在编码位流312的一部分中所表示的)各子节点调用方法1000的新实例。针对子节点所调用的方法1000的各实例设置有与(本方法1000实例的变换深度值+1)相等的变换深度值。

[0143] 在分割变换标志值为0的情况下,识别最大数量的正变换的步骤1004确定针对编码中的区域各色度通道的最大数量(n)的变换。在没有发生推断分割的情况下,该数量n将为1。在4:2:2色度格式处于使用中、遇到诸如 8×16 区域664等的色度通道的矩形区域并

且该区域大小是区域大小的预定集合(诸如 16×32 和 8×16 等)其中之一)的情况下,发生推断分割并且变换的最大数量将为2(否则变换数量将为1)。另一方面(区域大小不是区域大小的预定集合其中之一),变换的最大数量将为1。例如,如果 4×8 不是区域大小的预定集合其中之一,则变换的最大数量将为1。在4:4:4色度格式处于使用中并且所遇到的区域大小是区域大小的预定集合其中之一(诸如 32×32 区域等)的情况下,发生推断分割并且变换的最大数量将为4。另一方面(区域大小不是区域大小的预定集合其中之一),最大数量将为1。例如,如果 8×8 不是区域大小的预定集合其中之一,则变换的最大数量将为1。尽管区域大小的预定集合包括 8×16 ,但区域大小的其它预定集合也是可以的,诸如在4:2:2色度格式处于使用中的情况下仅为 16×32 或者在4:4:4色度格式处于使用中的情况下为 32×32 等。

[0144] 针对各色度通道,如果母节点的编码块标志值为1,则针对各个n,将编码块标志编码在编码位流312中。例如,在变换数量等于2的情况下,编码块标志942和944表示针对通过分割所推断出的两个区域各自存在变换。选择正变换的步骤1006基于变换单位(TU)大小来针对最大数量的各变换从正变换的预定集合中选择正变换,而该变换单位(TU)大小依赖于变换深度,因而与最大编码单位中的变换单位的层级有关。在变换深度等于0的情况下,变换单位(TU)大小等于编码单位(CU)大小。对于变换深度的各递增,变换单位(TU)大小减半。对于 32×32 编码单位(CU)大小、变换深度为0并且使用4:2:2色度格式,变换单位(TU)大小由此将是 32×32 并且针对色度的变换大小由此将为 16×32 。例如,在变换的最大数量为2并且针对色度的区域大小为 16×32 的情况下,针对通过推断分割而产生的色度的各个 16×16 区域选择 16×16 正变换。

[0145] 应用正变换的步骤1008针对编码块标志值为1的相应区域上的最大数量的各变换进行正变换。对色度残差样本阵列进行编码的步骤1008通常由变换模块320来进行。这样使得将各色度残差样本阵列(空间域表示)转换成色度残差系数阵列(频域表示)。

[0146] 对色度残差系数阵列进行编码的步骤1010将编码块标志值为1的各色度通道的最大数量的变换区域各自的色度残差系数阵列编码在编码位流312中。针对给定色度通道的给定变换单位进行编码的色度残差系数阵列的数量依赖于各变换的编码块标志值,因而将在0~(最多为)变换的最大数量的范围内改变。例如,在变换数量为2、并且这两个色度通道针对各个计数值的编码块标志值均为1的情况下,将色度残差块956、958、960和962编码在编码位流312中。如果针对给定色度通道的各变换的编码块标志值为0,则针对该色度通道不将色度残差块编码在编码位流312中。对色度残差系数阵列进行编码的步骤1010通常由熵编码器324来进行。

[0147] 图11是示出用于通过对变换树非叶节点句法结构902和变换树叶节点句法结构932进行解码来对变换单位(TU)进行解码的方法1100的示意流程图。参考变换单位(TU)的色度通道来说明方法1100,然而方法1100还可应用于变换单位(TU)的任何色度通道。由于变换树非叶节点句法结构902和变换树叶节点句法结构932描述变换树中的一个节点,因此方法1100从编码位流312解码变换树的一个节点。可以以适当的硬件或者可选地以例如处理器205可执行的软件来进行方法1100。最初针对变换树的顶层等级调用方法1100,并且方法1100能够(递归地)调用自身以对变换树的子节点进行解码。确定变换单位(TU)大小的步骤1102以与确定变换单位大小的步骤1002相同的方式确定变换单位(TU)大小。确定变换单位大小的步骤1102根据包含变换树的编码单位(CU)大小和变换单位(TU)的变换深度值来

确定变换树中的变换单位 (TU) 的大小。在变换树的顶层等级调用方法1100的情况下,将变换深度值设置为0,否则将利用方法1100的母实例来提供变换深度值。在变换深度值小于最大容许变换深度的情况下,从编码位流312解码诸如分割变换标志值702等的分割变换标志值作为分割变换标志910。

[0148] 在分割变换标志值为1的情况下,仅在变换树层级的母节点具有相应的编码块标志值(1)的情况下,才针对各色度通道进行色度编码块标志912和914的解码。然后,方法1100针对变换树的(利用变换树句法结构916、918、920和922在编码位流312的一部分中所表示的)各子节点调用方法1100的新实例。针对子节点所调用的方法1100的各实例设置有与(本发明1100实例的变换深度值+1)相等的变换深度值。

[0149] 在分割变换标志值为0的情况下,识别最大数量的逆变换的步骤1104以与识别最大数量(n)的正变换的步骤1004相同的方式,确定针对存在于解码中的区域的各种色度通道的至少一个色度残差系数阵列各自的变换的(最大)数量(n)。在没有发生推断分割的情况下,该数量n将为1。在4:2:2色度格式处于使用中、遇到诸如 8×16 区域664等的色度通道的矩形区域并且区域大小是区域大小的预定集合(诸如 16×32 和 8×16 等)其中之一(的情况下,发生推断分割并且变换的最大数量将为2(否则,变换数量将为1)。另一方面(区域大小不是区域大小的预定集合其中之一),变换的最大数量将为1。例如,如果 4×8 不是区域大小的预定集合其中之一,则变换的最大数量将为1。在4:4:4色度格式处于使用中并且所遇到的区域大小是区域大小的预定集合其中之一(诸如 32×32 区域等)的情况下,发生推断分割并且变换的最大数量将为4。另一方面(区域大小不是区域大小的预定集合其中之一),最大数量将为1。例如,如果 8×8 不是区域大小的预定集合其中之一,则变换的最大数量将为1。尽管区域大小的预定集合包括 8×16 ,但区域大小的其它预定集合也是可以的,诸如在4:2:2色度格式处于使用中的情况下仅为 16×32 或者在4:4:4色度格式处于使用中的情况下为 32×32 等。针对各色度通道,如果母节点的编码块标志值为1,则针对(n)个变换中的每一个变换,在编码位流312中解码编码块标志。例如,在变换的最大数量等于2的情况下,编码块标志942和944表示针对通过推断分割出的两个区域各自存在变换。

[0150] 然后,对色度残差系数阵列进行解码的步骤1106从编码块标志值为1的编码位流312中解码针对各色度通道的最大数量的变换区域各自的残差系数阵列。针对给定色度通道的给定变换单位进行解码的残差系数阵列的数量依赖于各变换的编码块标志值,因而将在0~(最多为)“变换数量(n)”的范围内改变。例如,在变换数量为2并且这两个色度通道针对各个计数值的编码块标志均为1的情况下,从编码位流312解码色度残差块956、958、960和962。对色度残差系数阵列进行解码的步骤1106通常由熵解码器420针对编码块标志值为1的各色度残差系数阵列进行。

[0151] 然后,选择逆变换的步骤1108对于针对各色度通道的编码块标志值为1的最大数量的各变换,从逆变换的预定集合中选择逆变换。例如,在变换的最大数量为2、区域大小为 16×32 并且针对这两个变换各自的编码块标志值为1的情况下,针对通过推断分割而产生的 16×16 区域各自选择 16×16 逆变换。

[0152] 然后,应用逆变换的步骤1110针对编码块标志值为1的相应区域上的最大数量的变换区域各自进行逆变换。这样使得将各色度残差系数阵列(频域表示)转换成表示解码视频帧的色度残差样本阵列(空间域表示)。应用逆变换的步骤1110通常由逆缩放和变换模块

422来进行。

[0153] 针对 4×8 变换单位1200,图12A示出对角扫描模式1201,图12B示出水平扫描模式1202,并且图12C示出垂直扫描模式1203。使用所例示的扫描模式来对 4×8 变换单位1200进行扫描的这些实现具有将残差系数分组成已知为“子块”的 4×4 的块的性质。因此,可以使用存在于编码位流312中的“系数组”标志来针对各子块表示存在至少一个有效(非零)残差系数。对 4×8 变换应用 4×4 子块大小实现了与存在于系数始终被分组成子块的其它变换大小中的扫描模式的一致性。

[0154] 特定实现可以应用系数组标志以用信号通知在各子块中存在至少一个非零残差系数。有利地,这些扫描模式通过将子块处理重新用于所有变换大小,允许重新使用对残差系数进行处理的控制软件或数字电路。可以根据诸如所配置的预测单位(PU)的帧内预测方向等的标准来选择所使用的特定扫描模式。在变换对 $4:2:2$ 色度格式样本网格上的色度样本进行编码的情况下,由于将各色度样本映射到亮度样本的非正方形(2×1)阵列,因此帧内预测方向和扫描模式之间的关系改变,从而影响帧内预测模式的“方向”或角度。在图12A~12C中的位于变换单位(TU)的左上角的结束于DC系数的“后向”方向上示出扫描。此外,不要求从变换单位(TU)的右下角开始扫描。由于在变换单位(TU)的左上方区域中非零残差系数占主导,因此扫描可以从“最末有效系数位置”开始并且沿后方向进行直至到达左上方系数为止。

[0155] 其它实现可以向给定区域应用一次扫描以对残差系数进行编码,然后向这些残差系数应用一个以上的变换。在这种情况下,将仅一个编码块标志用于该区域,因此用于扫描模式所覆盖的所有变换。在任何扫描中存在至少一个有效残差系数的情况下,将编码块标志设置为1。例如,可以应用图12A~12C的 4×8 扫描模式以对两个 4×4 变换的残差系数进行编码。残差系数的两个 4×4 阵列可以相连接以形成适合扫描模式的 4×8 阵列。由于对阵列进行一次扫描,因此将一个“最末有效系数”位置编码在扫描模式的位流中,并且对于该阵列,一个编码块标志值就足够了。改进的离散余弦变换(DCT)的能量压缩性质给其它方案提供了优点,诸如使沿着扫描模式的路径的各正方形变换的系数交错至矩形系数阵列内等。这样提供了在组合后的 4×8 阵列中各 4×4 残差系数阵列中的残差系数值的密度大致相等的优点,从而使得熵编码器324能够创建更高的压缩效率,以供熵解码器420进行后续解码。

[0156] 特定实现的编码色度颜色通道可以使用用以对与 $4:2:0$ 色度样本网格相对应的色度样本位置处的残差样本进行编码的第一变换以及用以对相对于 $4:2:0$ 色度样本网格而在 $4:2:2$ 色度样本网格中引入的附加色度样本位置处的残差样本进行编码的第二变换。这些实现可以有利地使用诸如Hadamard变换等的第二变换的简化变换,其中在该Hadamard变换中,将第二变换的输出与第一变换的残差样本相加(或相组合)以产生第二变换的残差样本。有利地,可以使用实现诸如Haar变换等的变换的预处理阶段来将 $4:2:2$ 色度格式的色度样本网格采样成 $4:2:0$ 色度格式的色度样本网格。这些结构必须从预处理阶段发送附加残差系数作为边信息,其中将这种残差应用于在最大编码单位(LCU)等级处应用预处理变换的情况下的各最大编码单位(LCU)。

[0157] 针对给定区域具有多个变换的实现可以使用覆盖整个区域的单个组合扫描或针对各变换的单独扫描。如果将针对多个变换的扫描组合成一个扫描,则针对扫描中的各区域仅需一个编码块标志。使用单个组合扫描的这些实现可以通过使各变换的残差系数交错

(诸如以系数为单位进行交错等),来实现残差系数的更高压缩,从而配置来自谱特性相似的各种变换的残差系数。

[0158] 图13是示出用于对变换单位进行编码的方法1300的示意框图。视频编码器114所进行的方法1300对变换单位的亮度通道和色度通道进行编码。在确定亮度变换跳过标志值的步骤1302中,变换跳过控制模块346通常通过测试在空间域(进行变换跳过)和频域(不进行变换跳过)这两者中对残差样本阵列360进行编码的成本,来针对亮度通道确定诸如变换跳过标志964或9127等的变换跳过标志的值。在确定色度变换跳过标志值的步骤1304中,变换跳过控制模块346针对要应用于通过推断分割而产生的并且属于相同色度通道的所有子区域的色度通道其中之一,确定或设置诸如变换跳过标志966或9148等的变换跳过标志的值。变换跳过控制模块346可以应用与针对亮度通道的逻辑相同的逻辑;然而,位率成本确定在确定对色度通道(或“颜色通道”)中的所有色度残差样本阵列进行变换跳过或者不对色度通道中的色度残差样本阵列进行变换跳过的成本时,必须考虑通过推断分割而产生的各个色度残差样本阵列。针对各色度通道重复确定色度变换跳过标志值的步骤1304,从而确定其它色度通道的变换跳过标志(诸如变换跳过标志968或9151等)。对亮度变换和色度变换进行编码的步骤1306使用熵编码器324来将亮度残差样本阵列编码在编码位流312中,并且使用熵编码器324来将色度通道的色度残差样本阵列编码在编码位流312中。通过在变换模块320中将残差样本阵列变换成残差系数阵列、或者在利用视频编码器114进行变换跳过时绕过变换模块320,来根据亮度变换跳过标志确定亮度残差样本阵列。随后,将残差阵列363传递至缩放和量化模块322以创建残差数据阵列364。在残差数据阵列364中的值中的至少一个值为非零的情况下,利用熵编码器324将残差数据阵列364的值(以残差数据块的形式,诸如残差数据块954、956、958、960或962等)编码在编码位流312中,并且将相应的编码块标志设置为1。除不同于第一个色度残差样本阵列的色度残差样本阵列与该第一个色度残差样本阵列共用变换跳过标志以外,以与亮度残差样本阵列相同的方式确定色度残差样本阵列。针对各色度通道重复步骤1306中的色度残差样本阵列的编码。

[0159] 图14是示出用于对变换单位进行解码的方法1400的示意流程图。视频解码器134所进行的方法1400对变换单位的亮度通道和色度通道进行解码。确定亮度变换跳过标志值的步骤1402通过使用熵解码器420从编码位流312解码诸如变换跳过标志964或9127等的变换跳过标志,来确定针对亮度通道的变换跳过标志的值。确定色度变换跳过标志值的步骤1404针对要应用于色度通道内并且处于相同的变换单位(TU)中的所有色度残差样本阵列的色度通道内的一个色度残差样本阵列,确定变换跳过标志的值。步骤1404使用熵解码器420来从编码位流312解码诸如变换跳过标志966或9148等的变换跳过标志。使变换跳过标志与第一个色度残差样本阵列相关联的实现使得不必在根据随后的残差系数阵列确定变换跳过标志之前缓冲先前的残差样本阵列(其中,该随后的残差系数阵列之后将会用于继续对先前的残差样本阵列进行处理,由此引入附加内部缓冲)。步骤1404诸如通过使用熵解码器420从编码位流312解码变换跳过标志968或9151等,还可以确定针对附加色度通道的变换跳过标志。对亮度变换和色度变换进行解码的步骤1406使熵解码器420在诸如编码块标志950等的相应编码块标志为1的情况下,对诸如亮度残差数据块954等的亮度残差系数阵列进行解码,并且在诸如编码块标志942和944等的各相应编码块标志为1的情况下,对诸如色度残差系数阵列956和958等的与特定色度通道相关联的色度残差系数阵列进行解码。

在对亮度变换进行解码的情况下,如果不进行变换跳过,则仅使亮度残差系数阵列通过逆变换模块422,否则亮度残差系数阵列绕过逆变换模块422。在对色度变换进行解码的情况下,针对变换单位中的各色度残差样本阵列,应用存在于编码位流312中并且与第一个色度残差样本阵列相关联的变换跳过标志。

[0160] 在发生推断分割的情况下,方法1300和1400的说明是指可以包含针对给定色度通道的多个色度残差样本阵列的“变换单位”。这符合句法结构930。在句法结构9100、9130和9160处于使用中的情况下,将通过推断分割而产生的各色度区域例示为单独的变换单位(TU)(在图9C、9D和9E中标记为色度变换单位(CTU))。为了方法1300和1400的目的,色度变换单位(CTU)仅是使用变换树句法结构9100来对色度区域进行分割的假象。在图9C中,亮度变换单位(LTU)9126所占据的空间区域可被视为“变换单位”,这是因为该亮度变换单位(LTU)9126占据与变换单位952相同的空间区域。色度变换单位(CTU)9116、9118和9120~9122(在存在的情况下)可被视为通过推断分割而产生的色度子区域。

[0161] 有利地,方法1300和1400这两者使得针对各颜色通道来对一个变换跳过标志进行编码,而与(在4:2:2色度格式和4:4:4色度格式处于使用中的情况下适用的)推断分割操作的存在与否无关。该特性得到与4:2:0色度格式一致的表现方式,其中:针对各残差系数阵列存在一个变换跳过格式,并且针对给定变换单位的各颜色通道仅存在一个残差系数阵列。例如,采用4:2:0的 8×8 变换单位将具有针对亮度的 8×8 变换和针对各色度通道的 4×4 色度变换。在这种情况下,针对各色度通道将存在一个变换跳过标志。在4:2:2的情况下,通过推断分割,在各色度通道中将存在两个 4×4 色度变换。通过第一个 4×4 色度变换进行了编码但应用于这两个 4×4 色度变换的变换跳过标志将控制针对与4:2:0的情况相同的空间区域的变换跳过状况。该一致表现方式得到针对与4:2:0情况向后兼容的4:2:2的变换跳过处理(即,由于在4:2:2的情况下支持变换跳过,因而在4:2:0的情况下没有发生句法元素的重新配置)。针对所有色度具有共通的变换跳过,这得到避免为了指定变换跳过的目的而人为地将变换单位分割成上半部分和下半部分的推断分割。

[0162] 图15是示出针对视频编码器114和视频解码器134的 4×4 和 8×8 的变换单位中的 4×4 变换的可能配置的示意表示。在图15中沿着列示出颜色通道Y、U和V,并且沿着行示出三种情况。在所有的所示情况中,视频编码器114和视频解码器134被配置为使用4:2:2色度格式。此外,在所有情况中,视频编码器114和视频解码器134支持 4×8 色度区域向两个 4×4 色度区域的推断分割,因而针对各颜色通道示出两个 4×4 色度变换。示出如下三个情况:

[0163] 情况1: 8×8 变换单位(上行);

[0164] 情况2:具有第一排序(顺序1)的变换的四个 4×4 变换单位(中间行);以及

[0165] 情况3:具有第二排序(顺序2)的变换的四个 4×4 变换单位(下行)。

[0166] 针对各情况,按变换在编码位流312中出现的顺序来对这些变换进行编号。情况1针对各色度通道示出具有 8×8 亮度变换和两个 4×4 变换的变换单位(TU)。由于亮度变换为 8×8 ,因此亮度变换不具有变换跳过标志。情况2和3还示出四个 4×4 变换单位得到横跨多个变换单位(TU)的针对各色度变换的色度区域。在情况2和3中,如开发中的高效率视频编码(HEVC)标准所使用的,将四个变换单位(TU)编号为0~3并且向这四个变换单位(TU)编入索引“blkIdx”变量。对于图15所示的各变换,如果支持变换跳过,则在该变换的左上角中包括框。对于始终明确对变换跳过标志进行编码的变换,该框带阴影(诸如阴影框1502等)。无

阴影框(诸如无阴影框1504等)示出根据先前(诸如上变换等)推导出针对本次变换的变换跳过标志的情况。不支持该推导的实现针对具有无阴影框的变换,将变换跳过标志明确编码在编码位流312中。在情况2和情况3中,调用四次诸如变换单位句法结构952等的变换单位句法结构(“blkIdx”的值从0递增至3),其中针对各 4×4 变换单位进行一次。因而,在编码位流312中存在变换单位句法结构的四个实例。在各调用时,如果诸如编码块标志950等的相应编码块标志的值为1,则在编码位流312中存在诸如亮度残差数据块954等的亮度残差块。在情况2中,在第四次调用(“blkIdx”等于3)时,(如果诸如编码块标志942、944、946、948等的相应编码块标志的值为1,则)将诸如色度残差块956、958、960、962等的针对色度通道的色差残差块编码在编码位流312中。图9B中的亮度残差块和色度残差块的排序与情况2中所呈现的变换的排序相对应。在情况3中,排序由于以下而改变:在变换单位句法结构的第二次调用时(即,在“blkIdx”等于1的情况下),对上半部分的色度残差块(诸如色度残差块956、960等)进行处理,并且在变换单位句法结构的第四次调用时(即,在“blkIdx”等于3的情况下),对下半部分的色度残差块(诸如色度残差块958、962等)进行处理。

[0167] 图15中没有示出的另一情况是在 $4:2:0$ 色度格式处于使用中的情况下的 4×4 变换单位,其中将针对色度的一个 4×4 变换应用于相同的四叉树层级(整体占据亮度样本网格上的 8×8 区域)处的与针对亮度的四个 4×4 变换单位相对应的色度样本网格上的区域。在 4×8 变换可用于色度的情况下,如以下参考图18所述,(除 4×4 变换外)还将针对 $4:2:2$ 情况的变换跳过应用于 4×8 变换。在 4×8 变换不可用于色度并且 $4:2:2$ 色度格式处于使用中的情况下,这些实现必须使用针对各色度通道的两个 4×4 变换,并且可以对针对一个 4×4 变换(诸如上方的 4×4 变换等)的变换跳过标志进行编码,但将编码后的变换跳过标志应用于给定色度通道的这两个 4×4 变换。

[0168] 图17是示出详述图14的方法1400的各方面的用于对变换单位(TU)的残差数据进行解码的方法1700的示意图。方法1700确定针对给定区域的变换跳过标志并且对该区域的残差数据进行解码。在针对变换单位(TU)的亮度通道调用方法1700的情况下,仅存在一个区域。对于变换单位(TU)的一个色度通道并且在发生推断分割的情况下,存在两个区域并且针对编码块标志值为1的各区域调用方法1700。方法1700从变换跳过支持测试步骤1702开始。该步骤1702测试变换跳过启用标志和编码单位变换量化绕过标志以及本区域的变换大小。编码在编码位流312中的变换跳过启用标志表示在编码位流312中是否可利用变换跳过功能。编码在编码位流312中的编码单位变换量化绕过标志表示利用视频编码器114是否选择了“无损”编码模式,由此绕过了变换模块320和量化模块322这两者,因而视频编码器114以无损模式进行工作,从而使得视频解码器134能够准确地再现来自视频源112的所拍摄帧数据。将开发中的高效率视频编码(HEVC)标准中的“ \log_2 TrafoSize”变量所示的本区域的变换大小定义为正方形变换的边尺寸的 \log_2 。在变换跳过标志为真(即,启用)、编码单位变换量化绕过标志为假(即,未启用)并且变换大小为 4×4 (即, \log_2 TrafoSize等于2)的情况下,控制进入颜色通道中的第一个为真的编码块标志(CBF)区域的测试步骤1704,否则控制进入对残差数据进行解码的步骤1712。测试步骤1704判断本区域是否是颜色通道中(以及编码块标志(CBF)值为1的变换单位(TU)中)的第一个区域。由于在本区域的编码块标志的值为1的情况下仅调用方法1700,因此可能存在两种情况。如果针对推断分割的第一个色度区域(在 $4:2:2$ 色度格式处于使用中的情况下的上部区域,例如图6C中的区域682或

666) 调用方法1700, 则测试步骤1704评价为真并且控制进入对变换跳过标志进行解码的步骤1706。如果针对推断分割的后续色度区域(在4:2:2色度格式处于使用中的情况下的下部区域, 例如图6C中的区域684或668) 调用方法1700, 则测试步骤1704在先前针对第一个色度区域(针对本变换单位) 调用方法1700的情况下评价为假, 并且在先前没有针对第一个色度区域(针对本变换单位) 调用方法1700的情况下评价为真。在测试步骤1704评价为真的情况下, 控制进入对变换跳过标志进行解码的步骤1706。在步骤1706中, 熵解码器420从编码位流312解码变换跳过标志以确定变换跳过标志值。存储变换跳过标志值的步骤1708将变换跳过标志值存储在诸如硬件寄存器或寄存器246等的存储器中, 以供随后在方法1700的后续调用时使用。如果测试步骤1704评价为假, 则控制进入检索变换跳过标志值的步骤1710, 其中在该步骤1710中, 从诸如硬件寄存器或寄存器246等的存储器检索方法1700的前一调用时确定并存储的变换跳过标志值。在对残差数据进行解码的步骤1712中, 利用熵解码器420从编码位流312解码诸如残差数据块954、956、958、960或962等的残差数据块。如以上参考多路复用器423所述, 将所确定的变换跳过标志值作为变换跳过标志值468进行传递以控制变换跳过操作。在针对亮度通道调用方法1700的情况下, 步骤1702~1710与图14的步骤1402相对应, 并且在针对色度通道调用方法1700的情况下, 步骤1702~1710与图14的步骤1404相对应。对残差数据进行解码的步骤1712与图14的步骤1406的亮度残差解码和图14的步骤1406的色度残差解码相对应。如开发中的高效率视频编码(HEVC) 标准中所定义的, 方法1700还与“残差编码”句法结构相对应。

[0169] 图18是示出应用于针对各颜色通道的(利用 4×8 的非正方形变换的) 4×8 色度区域的变换跳过操作的示意表示1800。在图18中示出亮度通道(“Y”)和各色度通道(“U”和“V”)。图18示出两个情况。

[0170] 情况1: “ 8×8 TU”(图18的上行) 示出 8×8 变换单位(TU), 其中针对亮度通道为 8×8 变换1802并且针对各色度通道为 4×8 (非正方形或者矩形) 变换1804。在支持变换跳过操作的变换的左上角利用阴影框示出变换跳过标志。在这种情况下, 如图18所示, (除 4×4 变换的情况外) 在 4×8 变换的情况中也支持变换跳过操作, 因而 4×8 变换各自包括变换跳过标志1806。

[0171] 情况2: “四个 4×4 TU”(图18的下行) 示出四个 4×4 变换单位(TU), 其中针对亮度通道为四个 4×4 变换1808并且针对各色度通道为 4×8 (非正方形或者矩形) 变换1810。针对各色度通道的 4×8 变换(在色度样本网格上) 配置有(亮度样本网格上的) 亮度变换, 并且在四个 4×4 变换单位(TU) 之间共用。在该实现中, 如图18所示, (除 4×4 变换的情况外) 在 4×8 变换的情况中也支持变换跳过操作, 因而 4×8 变换包括变换跳过标志1812。

[0172] 对于支持图18的情况1和情况2的实现, 利用视频解码器134来进行方法1700的修改后的测试步骤1702以及步骤1706和1712。除包括了 4×8 的变换大小(以及变换大小 4×4) 作为支持变换跳过操作的可能变换大小以外, 修改后的测试步骤1702作为图17的测试步骤1702进行工作, 由此使得修改后的测试步骤1702在 4×4 变换和 4×8 变换的这两个情况下都能够评价为真。

[0173] 附录A示出与句法结构900和句法结构930有关的开发中的高效率视频编码(HEVC) 标准的可用“文本”。将附录A中的transform_tree() 函数的各实例描述为在图9A和9C中标记为“TT”的句法结构的一部分, 并且将附录A中的transform_unit() 函数的各实例描述为

在图9A和9B中标记为“TU”的句法结构的一部分。附录A中所提供的文本是符合句法结构900和930的文本的一个示例,并且其它示例也是可以的。符合句法结构900和930的文本表明了视频编码器114进行用以对位流进行编码的方法1000并且视频解码器134进行用以对位流进行解码的方法1100。

[0174] 附录B示出与句法结构9100和句法结构9130有关的开发中的高效率视频编码(HEVC)标准的可用文本。将附录B中的transform_tree()函数的各实例描述为图9C、9D和9E中标记为“TT”的句法结构的一部分,并且将附录A中的transform_unit()函数的各实例描述为图9C、9D和9E中标记为“TU”的句法结构的一部分。附录B中所提供的文本是符合句法结构9100和9130的文本的一个示例,并且其它示例也是可以的。符合句法结构9100和9130的文本表明了视频编码器114进行用以对位流进行编码的方法1000并且视频解码器134进行用以对位流进行解码的方法1100。

[0175] 附录A和附录B中的文本得到如下实现:在针对4:4:4色度格式所配置的大小为 32×32 的变换单位(TU)中遇到的 32×32 色度区域使得应用四个(最大数量) 16×16 色度变换,并且在针对4:2:2色度格式所配置的大小为 32×32 的变换单位(TU)中遇到的 16×32 色度区域使得应用两个(最大数量) 16×16 色度变换。通过附录A和附录B中的文本所得到的实现在应用于大小较小并且针对4:2:2色度格式所配置的变换单位(TU)的情况下,应用一个(最大)色度变换。例如,将 8×16 变换应用于 8×16 色度区域并且将 4×8 变换应用于 4×8 色度区域。

[0176] 产业上的可利用性

[0177] 所述的配置适用于计算机和数据处理行业,特别适用于对诸如视频信号等的信号进行编码解码的数字信号处理。

[0178] 前述仅说明本发明的一些实施例,并且可以在没有背离本发明的范围和精神的情况下对本发明进行修改和/或改变,其中这些实施例仅是示例性而非限制性的。

[0179] (仅限澳大利亚)在本说明书的上下文中,词语“包括”意味着“主要但未必仅包括”或“具有”或“包含”,而不是“仅由...组成”。词语“包括(comprising)”的诸如“comprise”和“comprises”等的词尾变化具有相应的变化含义。

[0180] 附录A

[0181] transform_tree()和transform_unit()使用循环结构来实现推断色度分割

[0182] 7.3.11变换树句法

[0183]

transform_tree(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx) {	描述符
if(log2TrafoSize <= Log2MaxTrafoSize && log2TrafoSize > Log2MinTrafoSize && trafoDepth < MaxTrafoDepth && !(IntraSplitFlag && trafoDepth == 0))	
split_transform_flag[x0][y0][trafoDepth]	ae(v)
if(trafoDepth == 0 log2TrafoSize > 2) {	
if(trafoDepth == 0 cbf_cb[xBase][yBase][trafoDepth - 1]) {	
for(tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
cbf_cb[x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt)][y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt)][trafoDepth + (TrafoCrCbCnt > 1)]	ae(v)
}	
cbf_cb[x0][y0][trafoDepth] = (TrafoCrCbCnt > 1)	
}	
if(trafoDepth == 0 cbf_cr[xBase][yBase][trafoDepth - 1]) {	
for(tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
cbf_cr[x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt)][y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt)][trafoDepth + (TrafoCrCbCnt > 1)]	ae(v)
}	
cbf_cr[x0][y0][trafoDepth] = (TrafoCrCbCnt > 1)	
}	
if(split_transform_flag[x0][y0][trafoDepth]) {	
x1 = x0 + ((1 << log2TrafoSize) >> 1)	
y1 = y0 + ((1 << log2TrafoSize) >> 1)	
transform_tree(x0, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 0)	
transform_tree(x1, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 1)	
transform_tree(x0, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 2)	
transform_tree(x1, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 3)	
} else {	
if(PredMode[x0][y0] == MODE_INTRA trafoDepth != 0 cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth])	
cbf_luma[x0][y0][trafoDepth]	ae(v)

[0184]

transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx)	
}	
}	

[0185] 7.3.12变换单位句法

[0186]

transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx) {	描述符
if(cbf_luma[x0][y0][trafoDepth] cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) {	
if(cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded) {	
cu_qp_delta_abs	ae(v)
if(cu_qp_delta_abs)	
cu_qp_delta_sign	ae(v)
}	
if(cbf_luma[x0][y0][trafoDepth])	
residual_coding(x0, y0, log2TrafoSize, 0)	
if(log2TrafoSize > 2) {	
if(cbf_cb[x0][y0][trafoDepth])	
for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt), log2TrafoSize, 1)	
}	
if(cbf_cr[x0][y0][trafoDepth])	
for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt), log2TrafoSize, 2)	
}	
} else if(blkIdx == 3) {	
if(cbf_cb[xBase][yBase][trafoDepth])	
residual_coding(xBase, yBase, log2TrafoSize, 1)	
if(cbf_cr[xBase][yBase][trafoDepth])	
residual_coding(xBase, yBase, log2TrafoSize, 2)	
}	
}	
}	

[0187] 7.4.8.1通用编码单位语义

[0188] 按照如下推导变量TrafoCrCbHorCnt和TrafoCrCbVertCnt:

[0189] -如果log2TrafoSize等于5并且split_transform_flag等于0,则按照如下推导TransformIdxMax:

[0190] -如果chroma_format_idc等于1,则TrafoCrCbHorCnt和TrafoCrCbVertCnt等于1。

[0191] -如果chroma_format_idc等于2,则TrafoCrCbHorCnt等于1并且TrafoCrCbVertCnt等于2。

[0192] -否则,如果chroma_format_idc等于3,则TrafoCrCbHorCnt和TrafoCrCbVertCnt等于2。

[0193] -否则, TrafoCrCbHorCnt和TrafoCrCbVertCnt等于1。

[0194] 推导出TrafoCrCbCnt作为TrafoCrCbHorCnt*TrafoCrCbVertCnt。

[0195] 按照如下推导变量log2CrCbTrafoHorSize和log2CrCbTrafoVertSize。

[0196] - 如果 `chroma_format_idc` 等于 1, 则 `log2CrCbTrafoHorSize` 和 `log2CrCbTrafoVertSize` 等于 `log2TrafoSize-1`。

[0197] - 否则, 如果 `chroma_format_idc` 等于 2, 则 `log2CrCbTrafoHorSize` 等于 `log2TrafoSize`, 并且 `log2CrCbTrafoVertSize` 等于 `min(log2TrafoSize-1, 4)`。

[0198] - 否则, 如果 `chroma_format_idc` 等于 3, 则 `log2CrCbTrafoHorSize` 和 `log2CrCbTrafoVertSize` 等于 `min(log2TrafoSize, 4)`。

[0199] 附录A结束

[0200] 附录B

[0201] 对于通过推断分割所产生的各色度变换, 针对每对色度通道调用一次 `transform_tree()`

[0202] 7.3.11变换树句法

[0203]

transform_tree(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx, chromaOnly) {	描述符
if(log2TrafoSize <= Log2MaxTrafoSize && log2TrafoSize > Log2MinTrafoSize && trafoDepth < MaxTrafoDepth && !(IntraSplitFlag && trafoDepth == 0) && !chromaOnly)	
split_transform_flag[x0][y0][trafoDepth]	ae(v)
if(trafoDepth == 0 log2TrafoSize > 2) {	
if(trafoDepth == 0 cbf_cb[xBase][yBase][trafoDepth - 1])	
if(TrafoCrCbCnt > 1) {	
cbf_cb[x0][y0][trafoDepth] = 1	
} else {	
cbf_cb[x0][y0][trafoDepth]	ae(v)
}	
if(trafoDepth == 0 cbf_cr[xBase][yBase][trafoDepth - 1])	
if(TrafoCrCbCnt > 1) {	
cbf_cr[x0][y0][trafoDepth] = 1	
} else {	
cbf_cr[x0][y0][trafoDepth]	ae(v)
}	
}	
if(split_transform_flag[x0][y0][trafoDepth] TrafoCrCbCnt > 1) {	
x1 = x0 + ((1 << log2TrafoSize) >> 1)	
y1 = y0 + ((1 << log2TrafoSize) >> 1)	
transform_tree(x0, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 0, TrafoCrCbCnt > 1)	
if(chroma_format_idc != 2) {	
transform_tree(x1, y0, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 1, TrafoCrCbCnt > 1)	
}	
transform_tree(x0, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 2, TrafoCrCbCnt > 1)	
if(chroma_format_idc != 2) {	
transform_tree(x1, y1, x0, y0, log2TrafoSize - 1, trafoDepth + 1, 3, TrafoCrCbCnt > 1)	

[0204]

}	
}	
else-if(!split_transform_flag[x0][y0][trafoDepth] && TrafoCrCbCnt > 1) {	
if((PredMode[x0][y0] == MODE_INTRA trafoDepth != 0 cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) && !chromaOnly)	
cbf_luma[x0][y0][trafoDepth]	ae(v)
transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx, chromaOnly)	
}	
}	

[0205] 7.3.12变换单位句法

[0206]

transform_unit(x0, y0, xBase, yBase, log2TrafoSize, trafoDepth, blkIdx) {	描述符
if(cbf_luma[x0][y0][trafoDepth] cbf_cb[x0][y0][trafoDepth] cbf_cr[x0][y0][trafoDepth]) {	
if(cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded) {	
cu_qp_delta_abs	ae(v)
if(cu_qp_delta_abs)	
cu_qp_delta_sign	ae(v)
}	
if(cbf_luma[x0][y0][trafoDepth])	
residual_coding(x0, y0, log2TrafoSize, 0)	
if(log2TrafoSize > 2) {	
if(cbf_cb[x0][y0][trafoDepth])	
for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt), log2TrafoSize, 1)	
}	
if(cbf_cr[x0][y0][trafoDepth])	
for (tIdx = 0; tIdx < TrafoCrCbCnt; tIdx++) {	
residual_coding(x0 + ((1 << log2CrCbTrafoHorSize) * (tIdx mod TrafoCrCbHorCnt), y0 + (1 << log2CrCbTrafoVertSize) * (tIdx div TrafoCrCbVertCnt), log2TrafoSize, 2)	
}	
} else if(blkIdx == 3) {	
if(cbf_cb[xBase][yBase][trafoDepth])	
residual_coding(xBase, yBase, log2TrafoSize, 1)	
if(cbf_cr[xBase][yBase][trafoDepth])	
residual_coding(xBase, yBase, log2TrafoSize, 2)	
}	
}	
}	
}	

[0207] 7.4.8.1通用编码单位语义

[0208] 按照如下推导变量TrafoCrCbHorCnt和TrafoCrCbVertCnt:

- [0209] -如果 $\log_2\text{TrafoSize}$ 等于5并且 $\text{split_transform_flag}$ 等于0,则按照如下推导 TransformIdxMax:
- [0210] -如果 chroma_format_idc 等于1,则 TrafoCrCbHorCnt 和 TrafoCrCbVertCnt 等于1。
- [0211] -如果 chroma_format_idc 等于2,则 TrafoCrCbHorCnt 等于1并且 TrafoCrCbVertCnt 等于2。
- [0212] -否则,如果 chroma_format_idc 等于3,则 TrafoCrCbHorCnt 和 TrafoCrCbVertCnt 等于2。
- [0213] -否则, TrafoCrCbHorCnt 和 TrafoCrCbVertCnt 等于1。
- [0214] 推导出 TrafoCrCbCnt 作为 $\text{TrafoCrCbHorCnt} * \text{TrafoCrCbVertCnt}$ 。
- [0215] 附录B结束

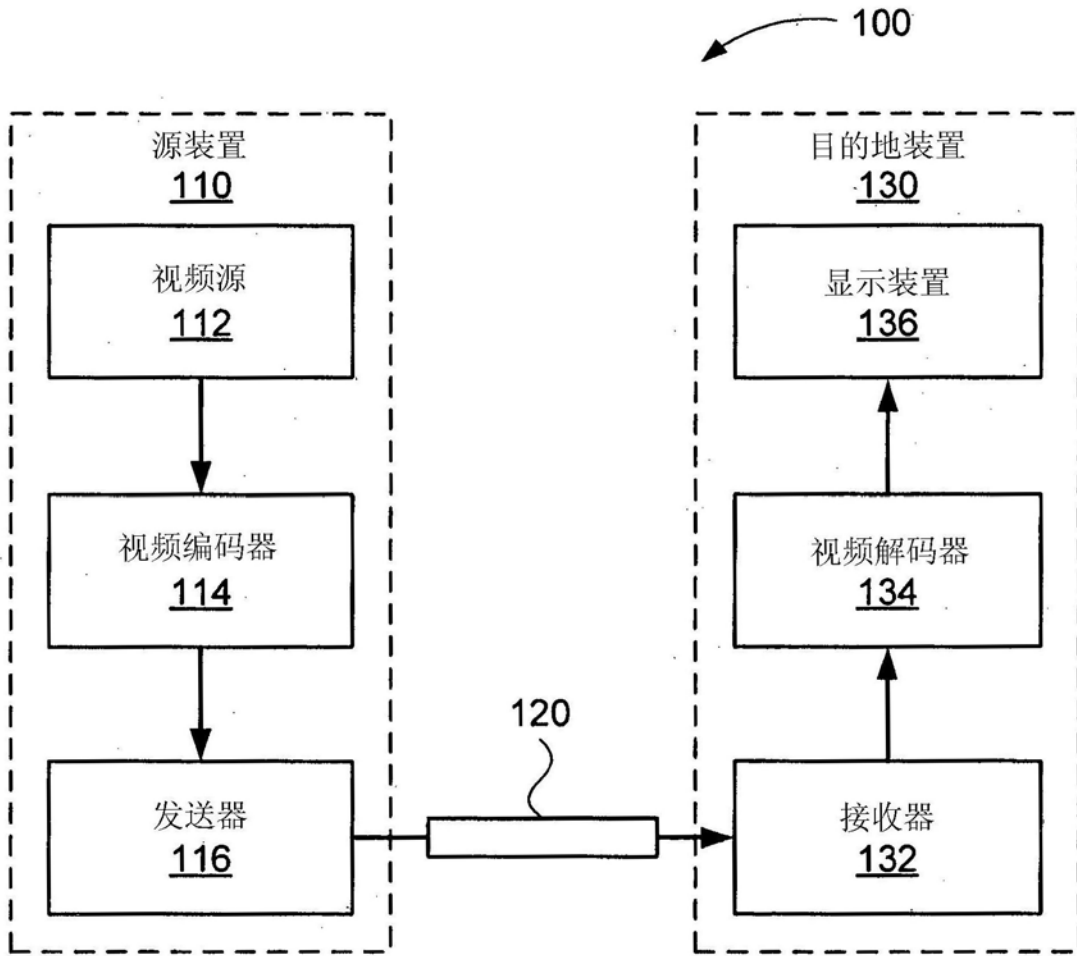


图1

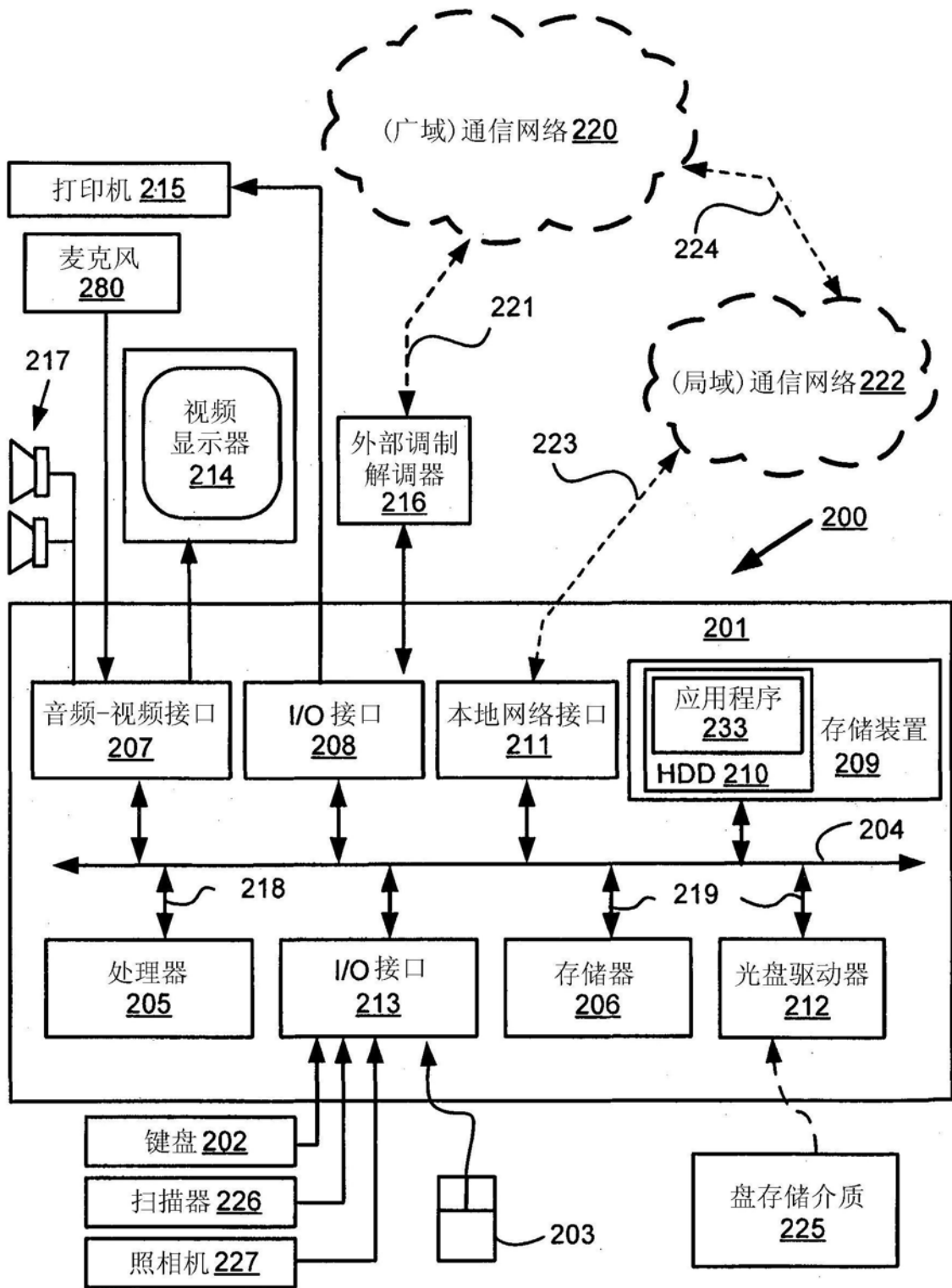


图2A

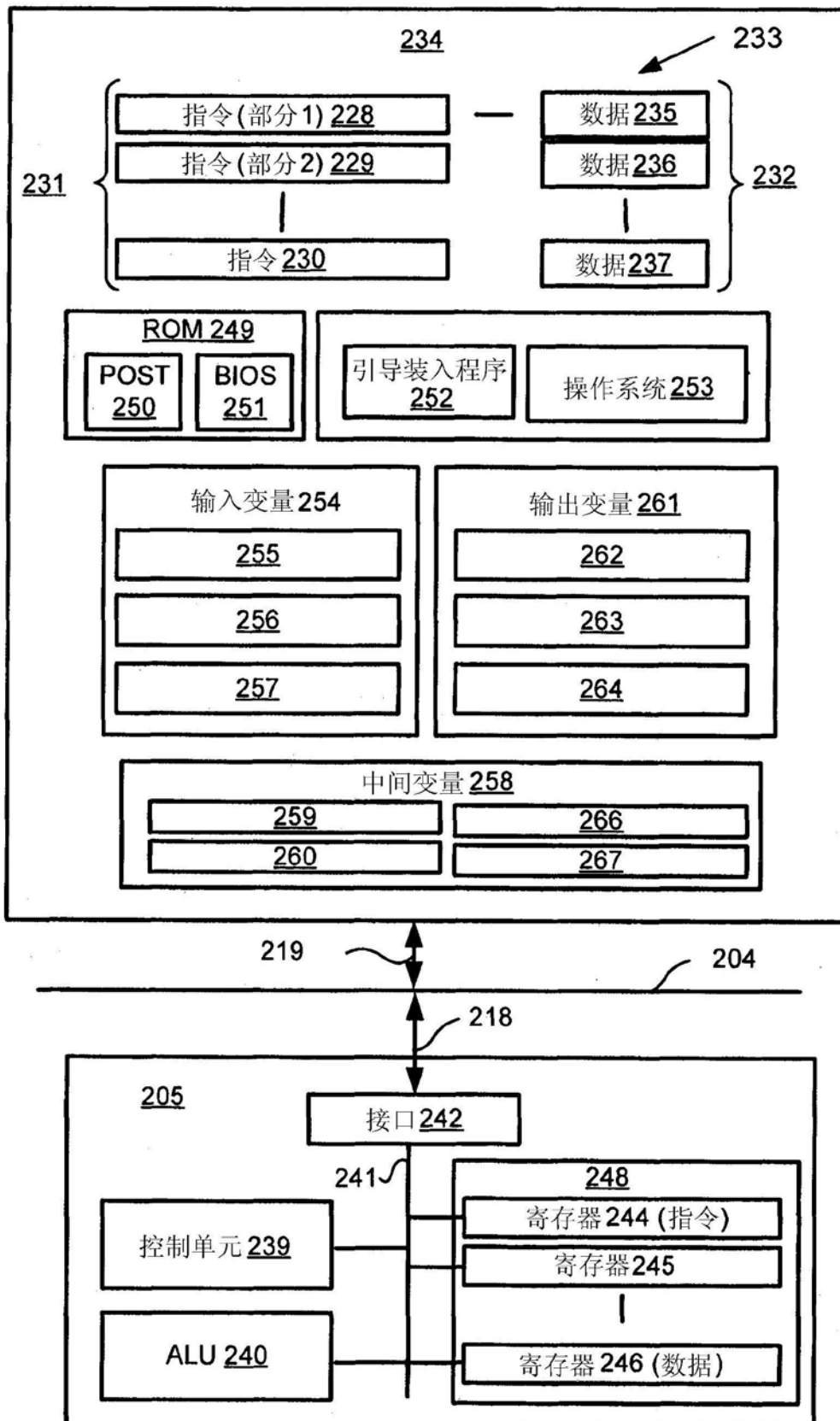


图2B

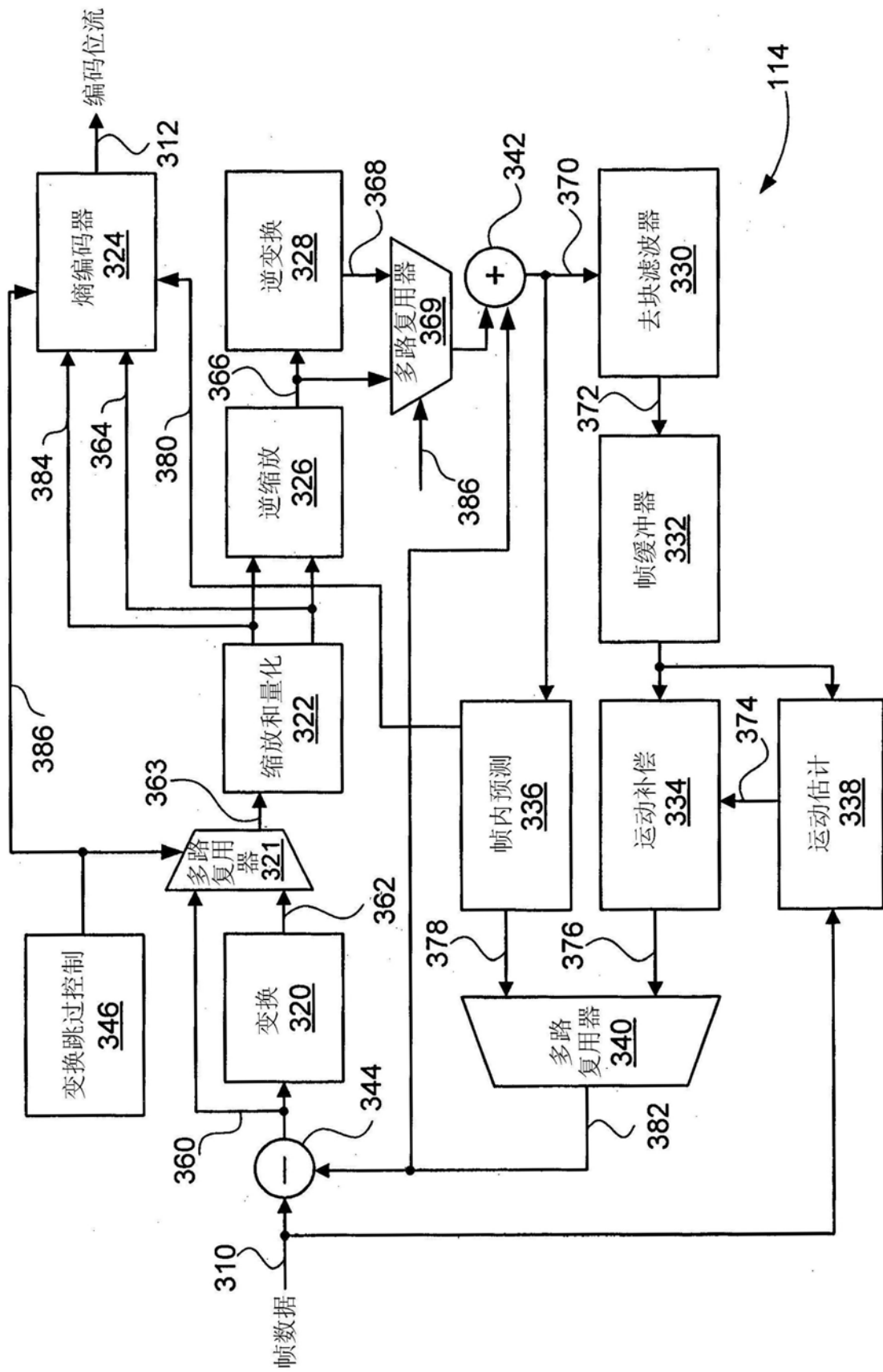


图3

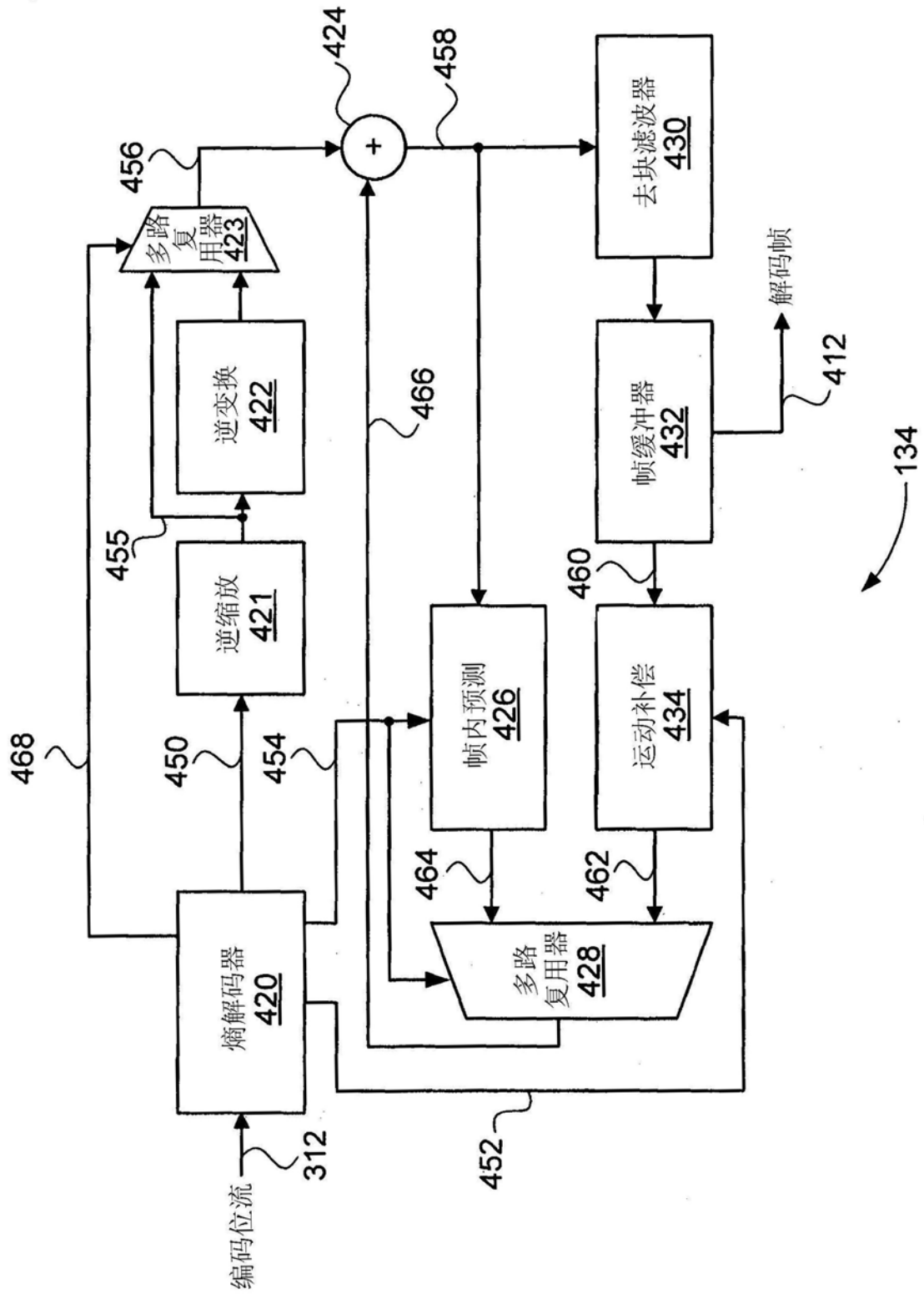


图4

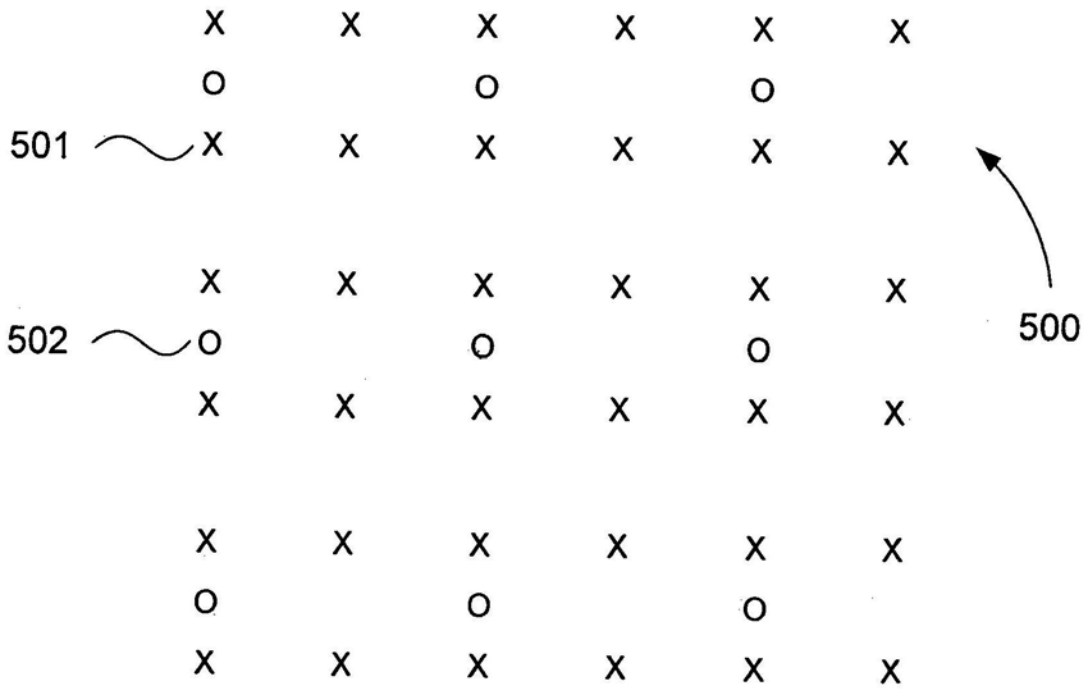


图5A

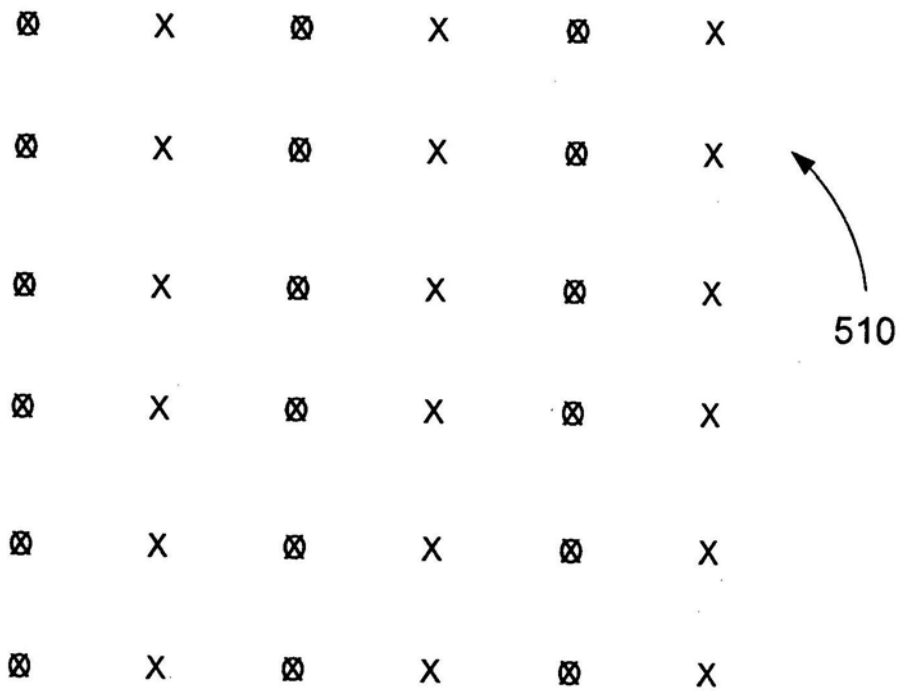


图5B

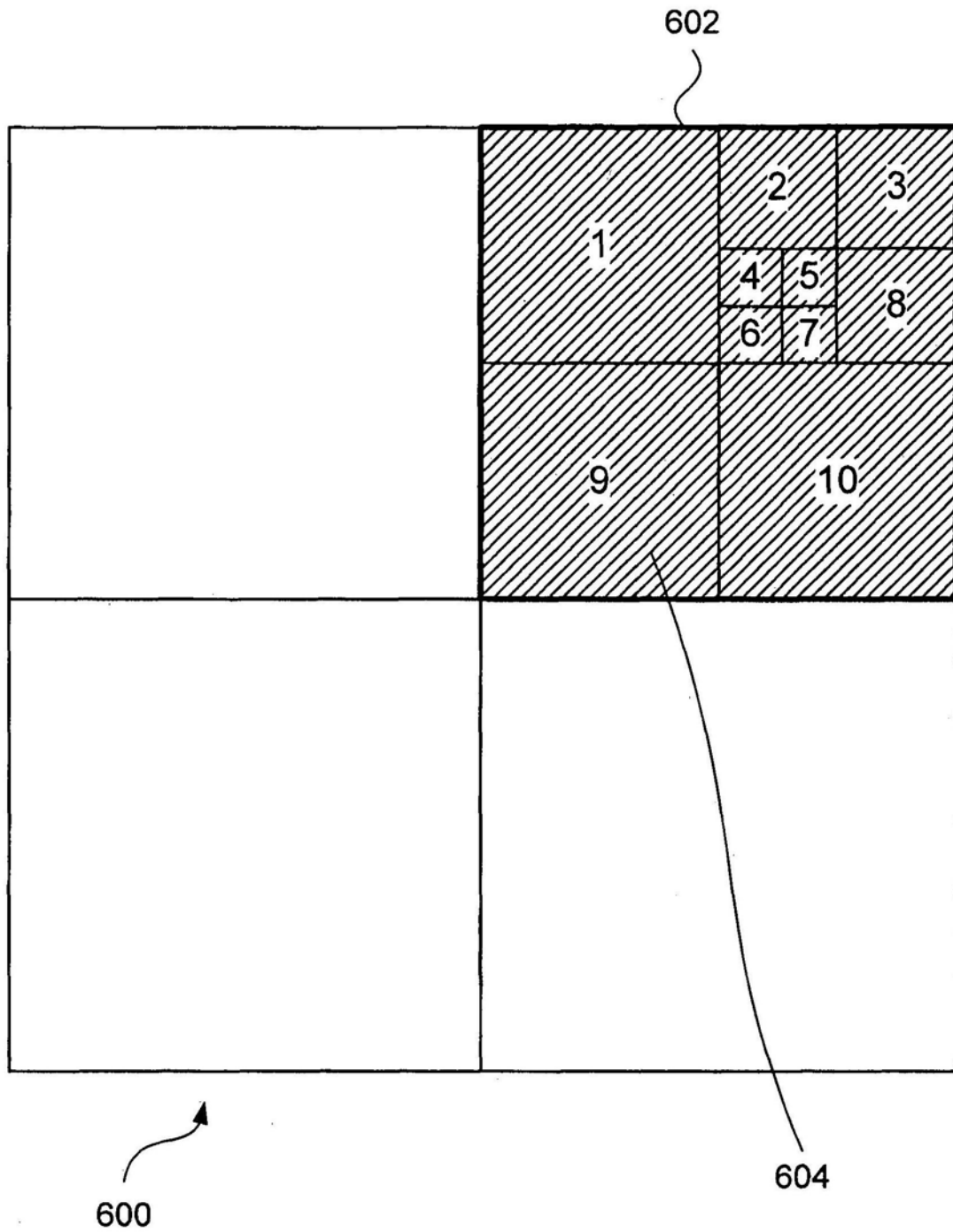


图6A

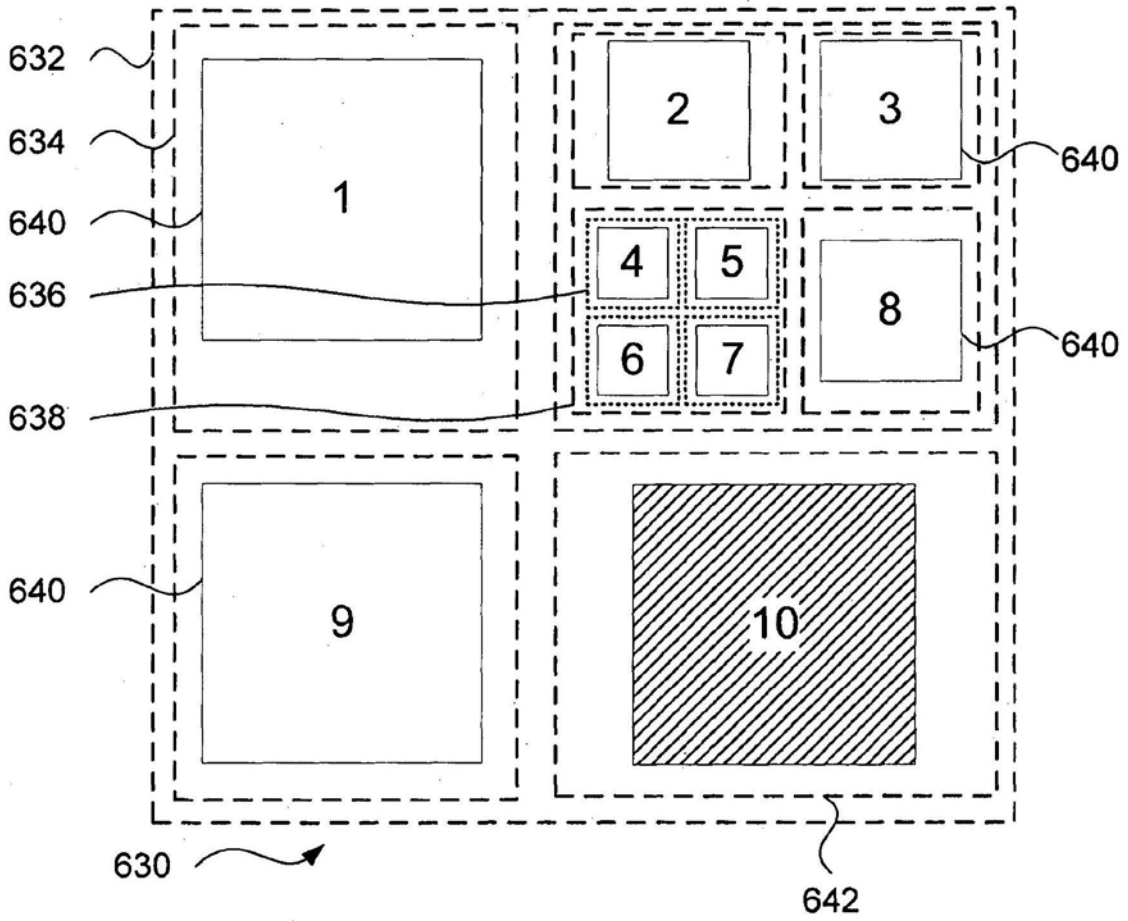


图6B

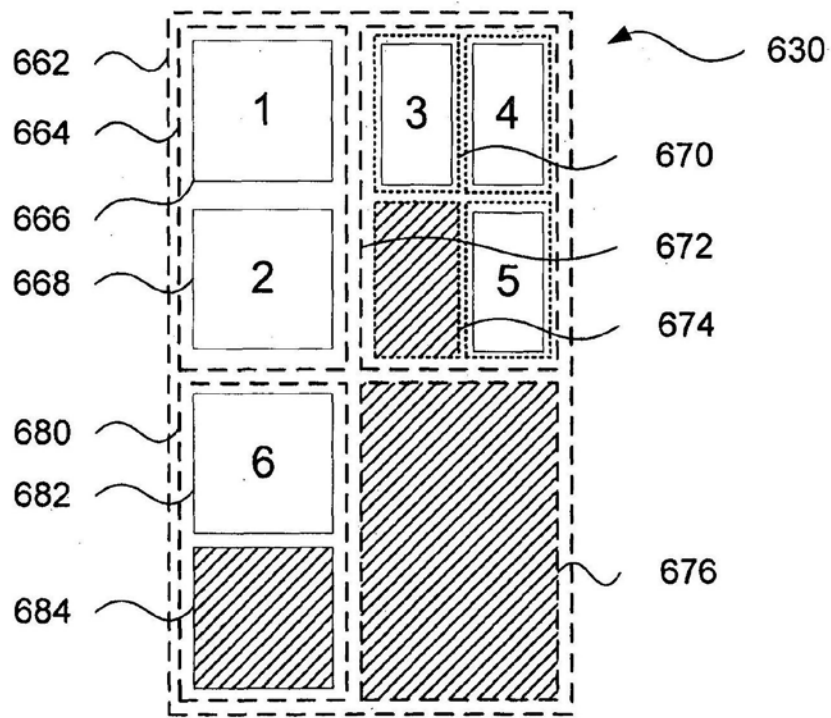


图6C

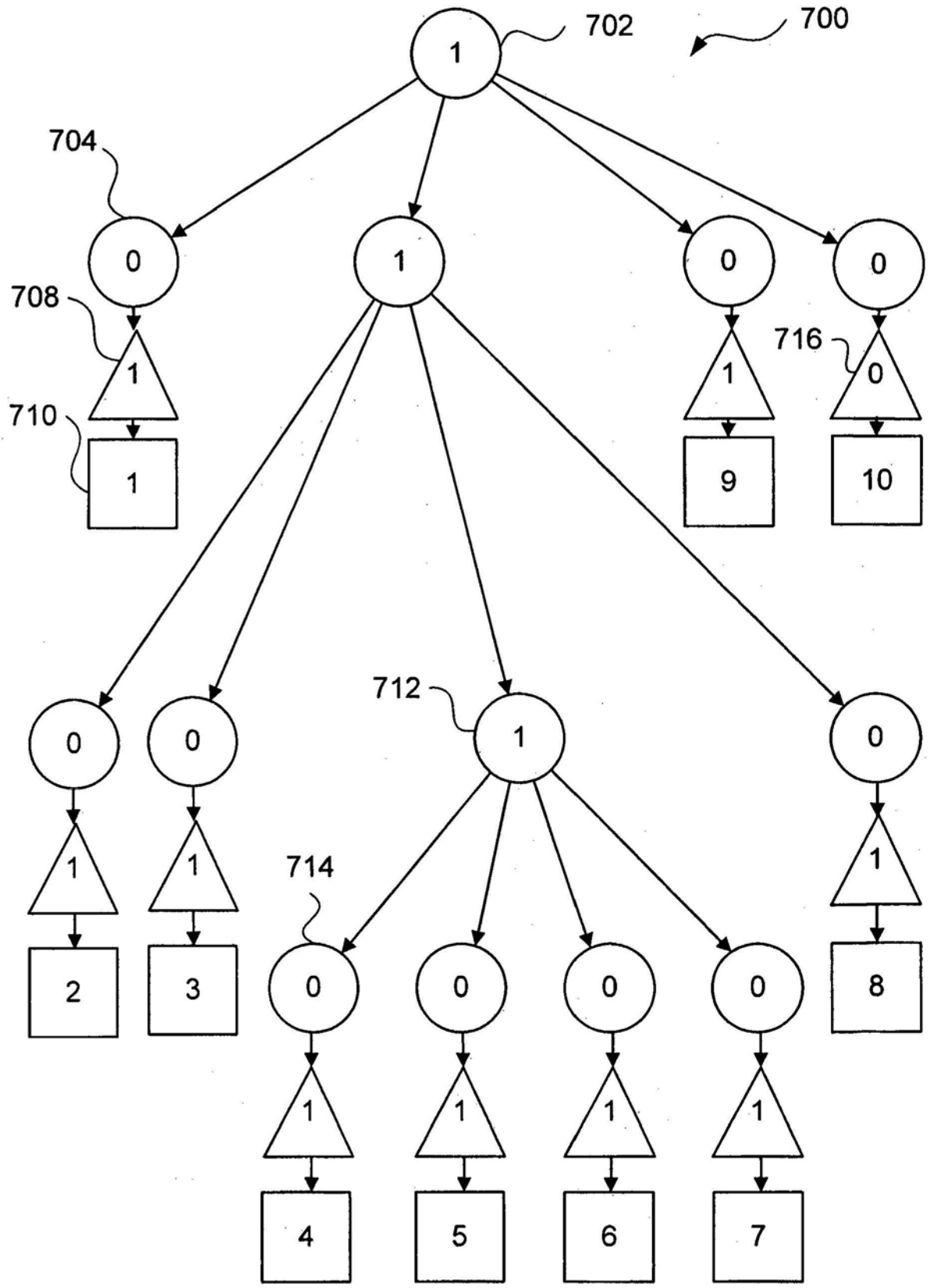


图7

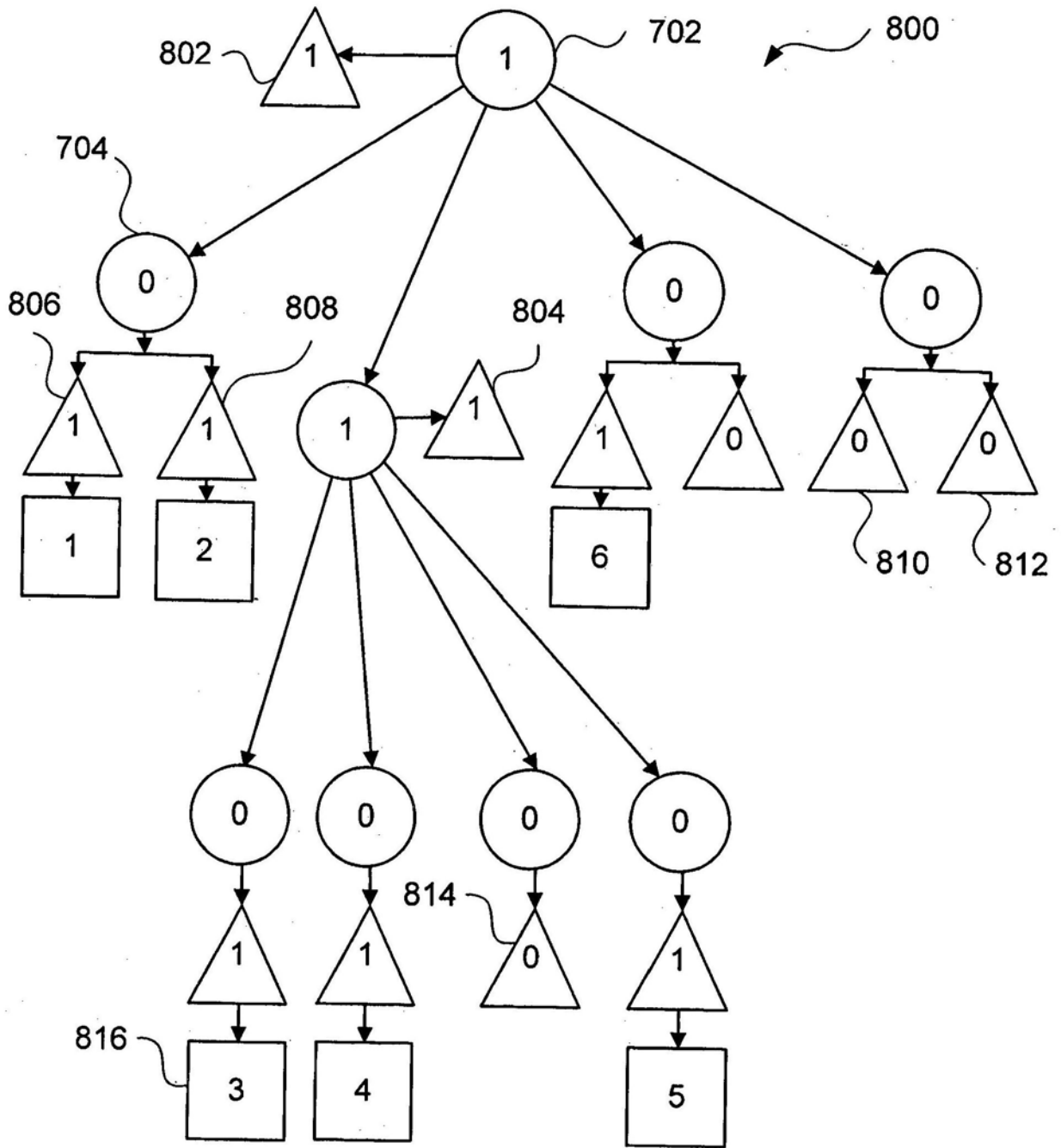


图8

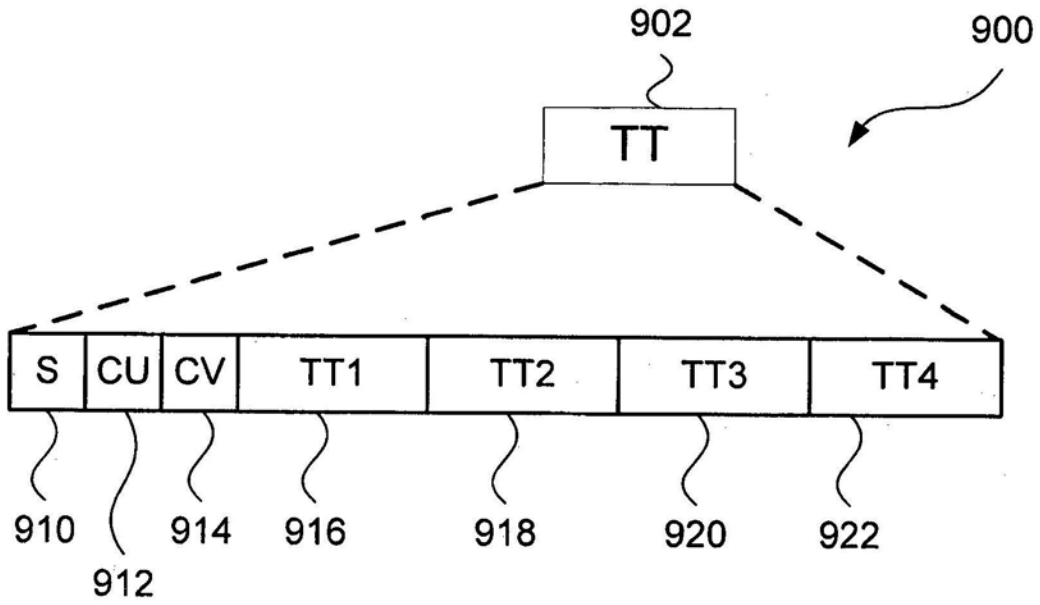


图9A

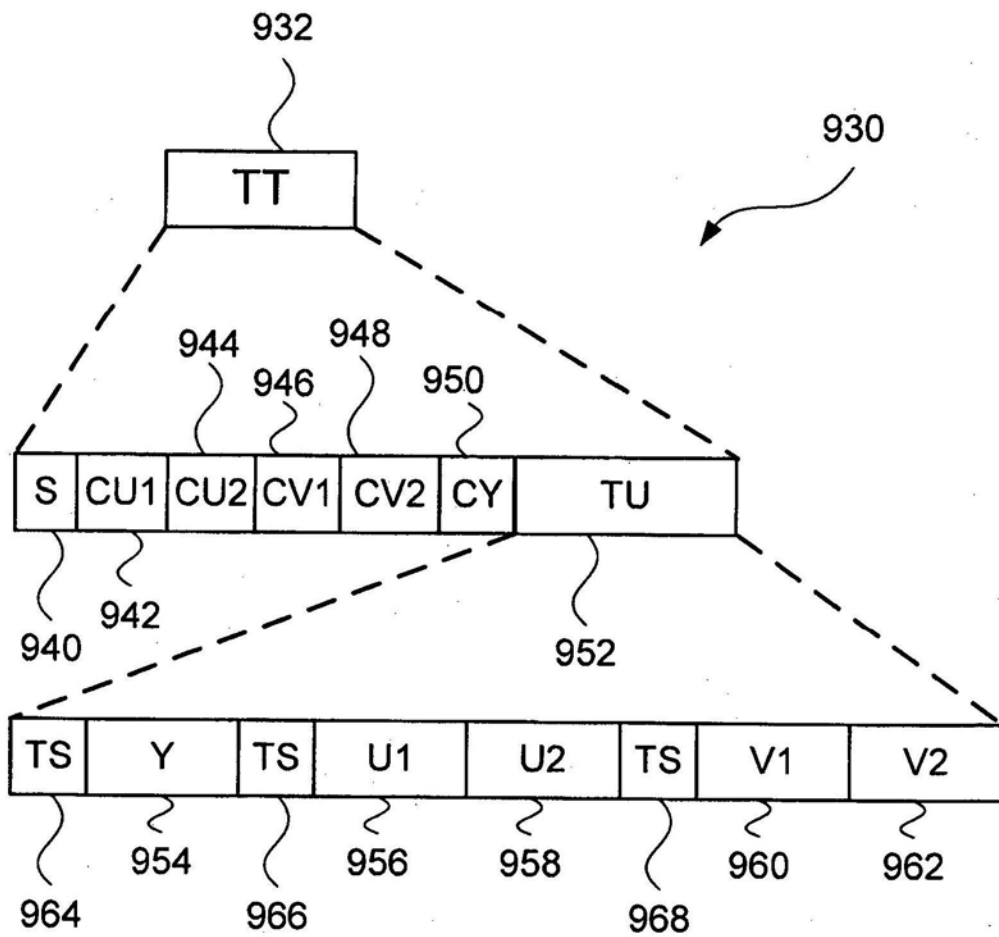


图9B

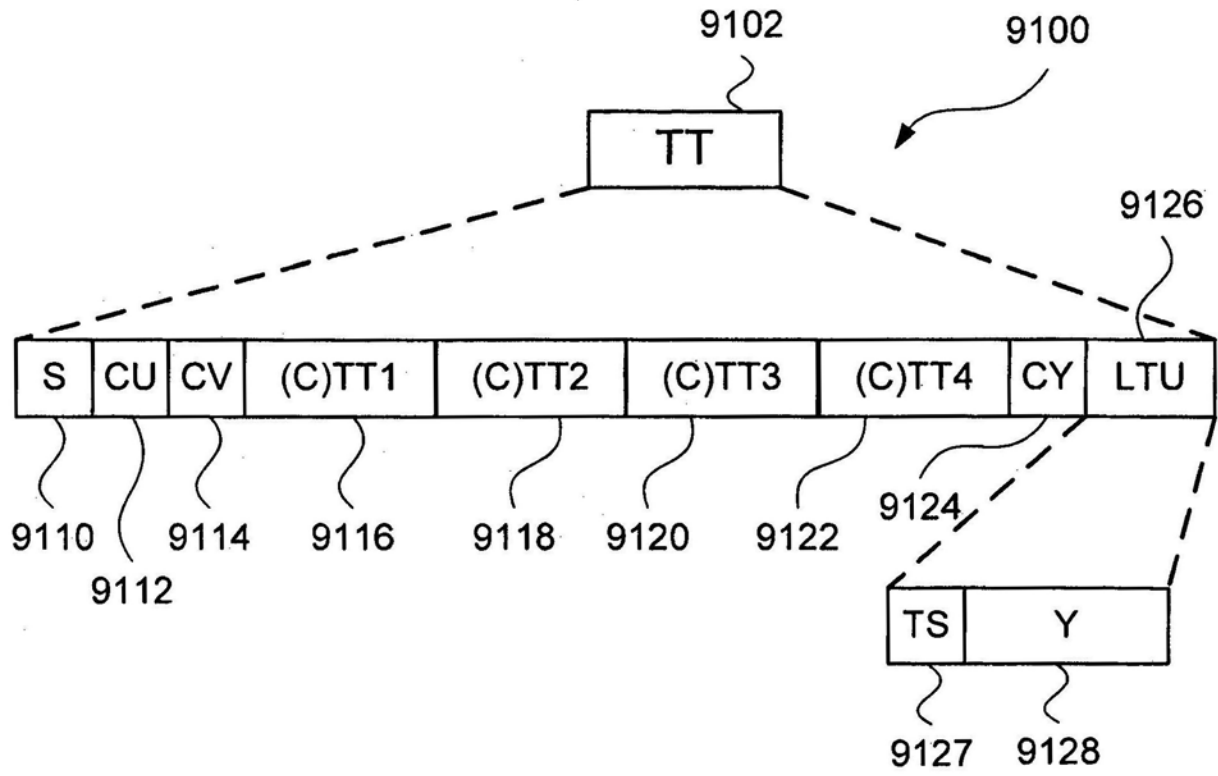


图9C

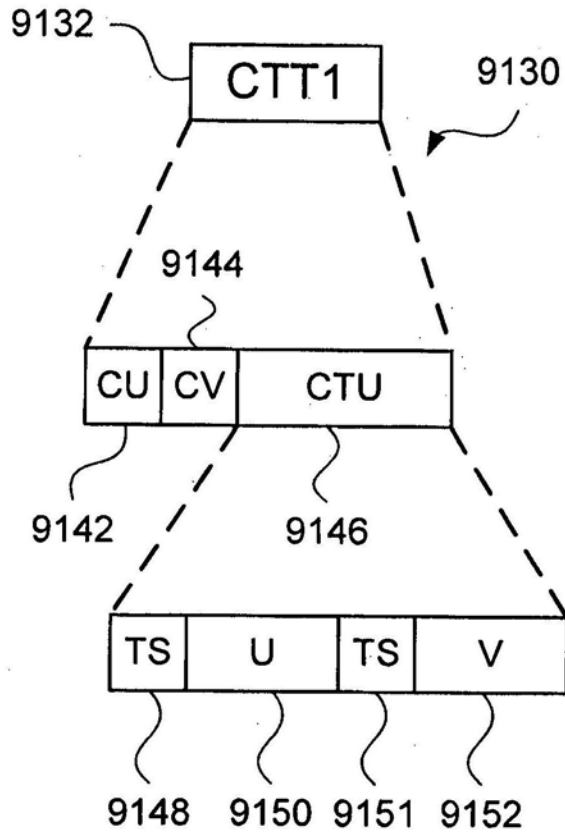


图9D

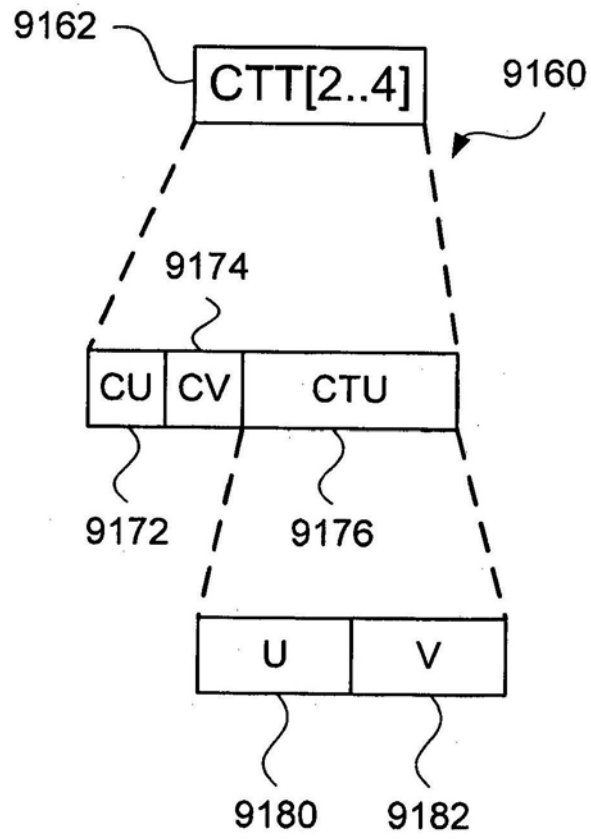


图9E

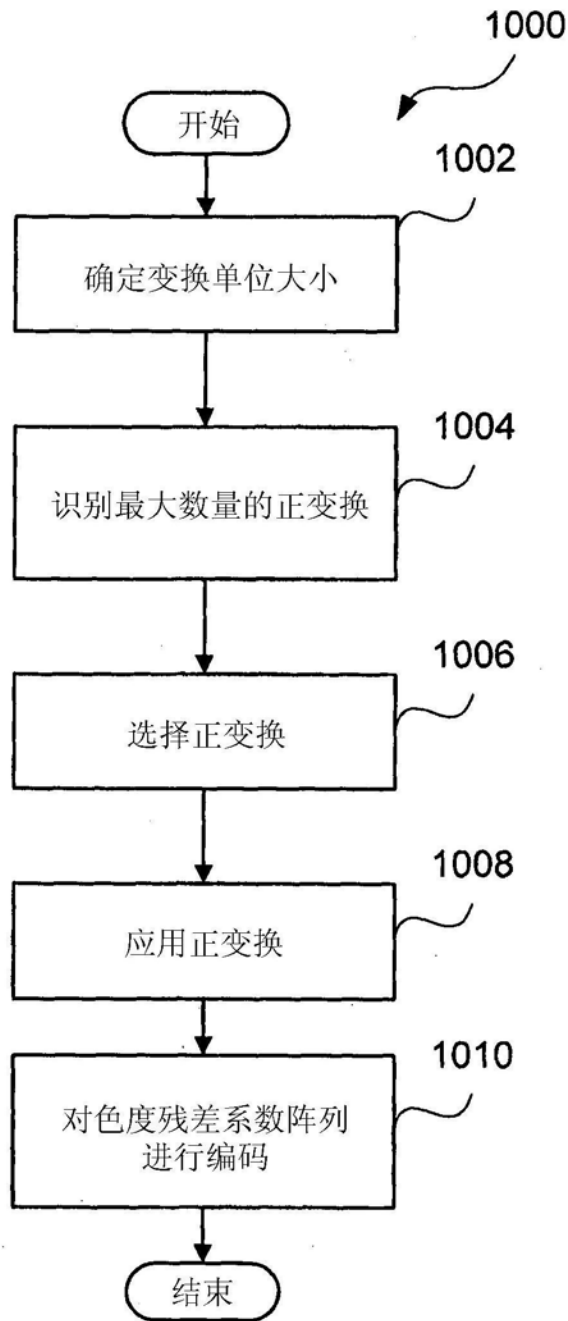


图10

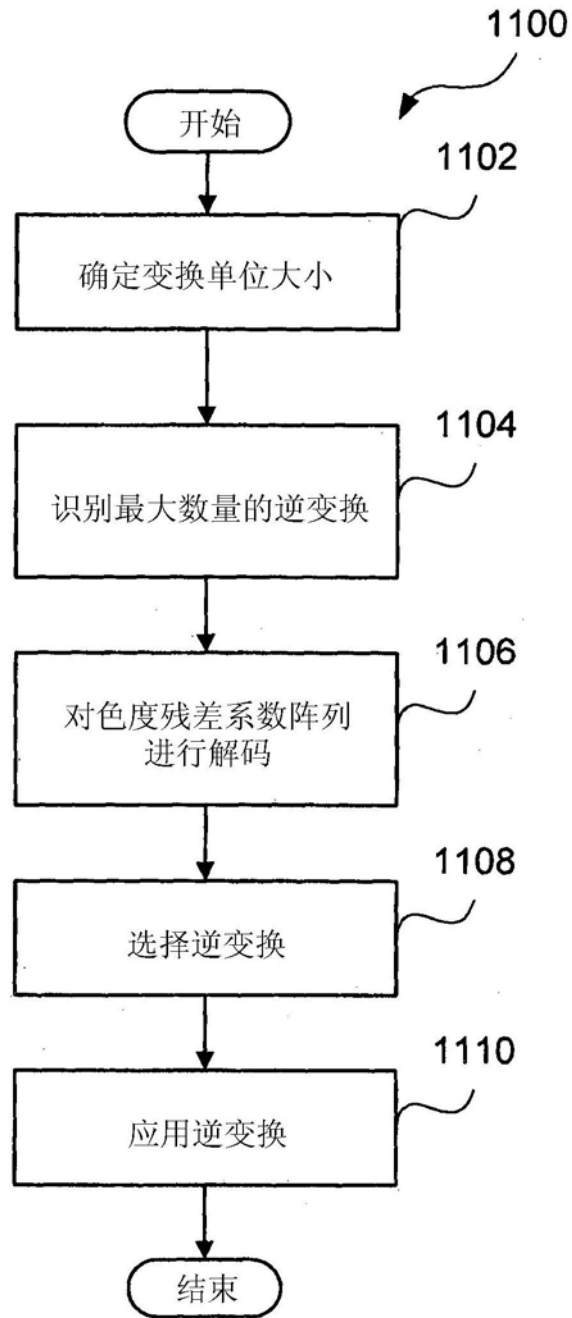


图11

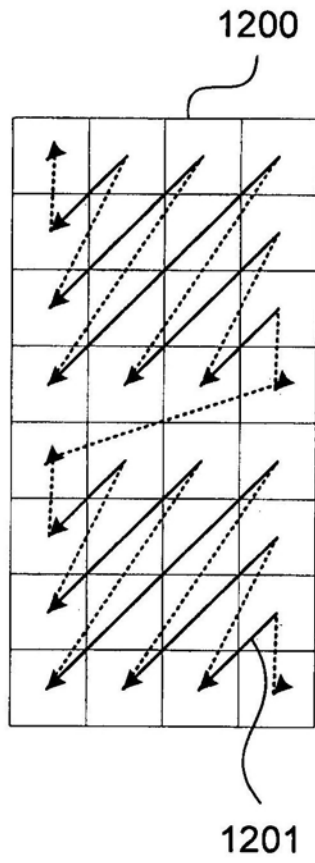


图12A

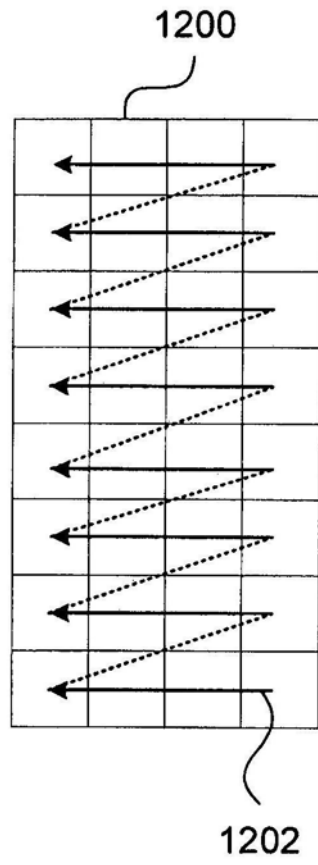


图12B

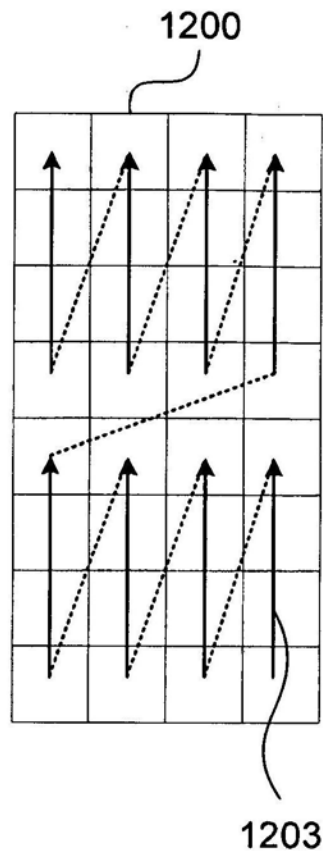


图12C

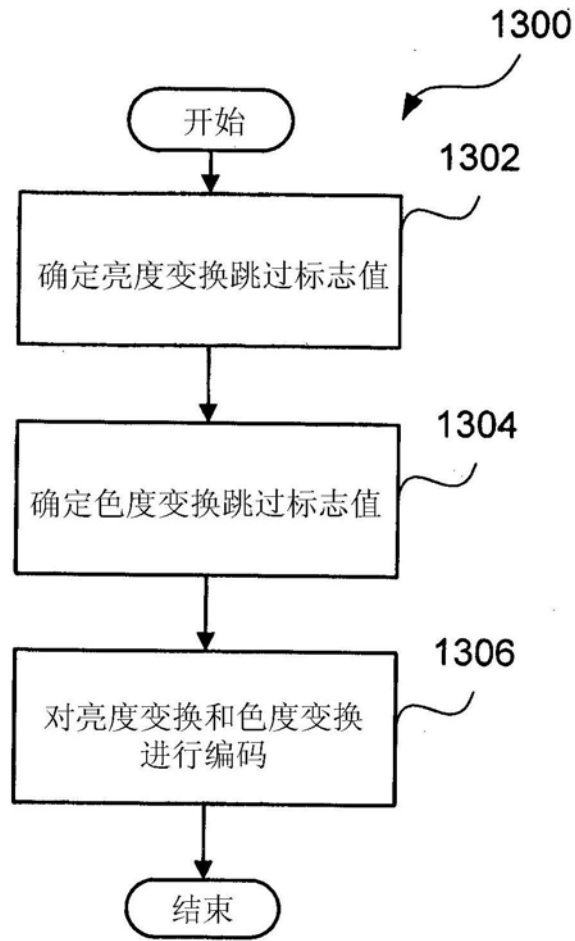


图13

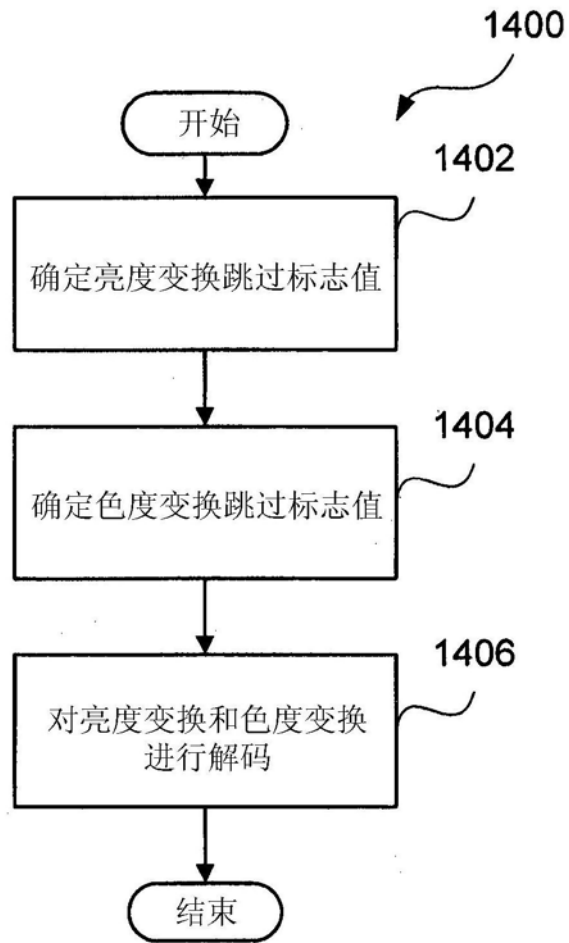


图14

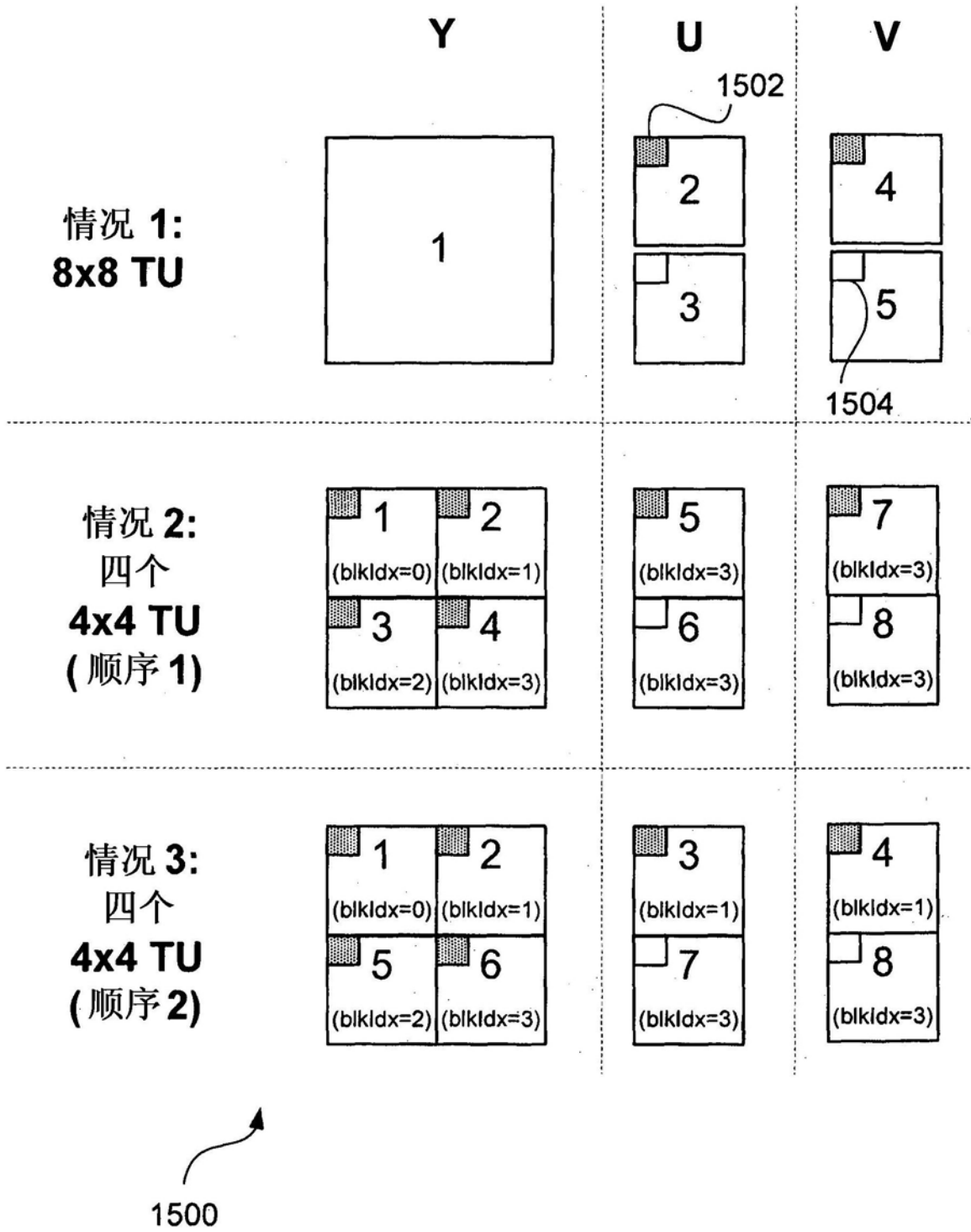


图15

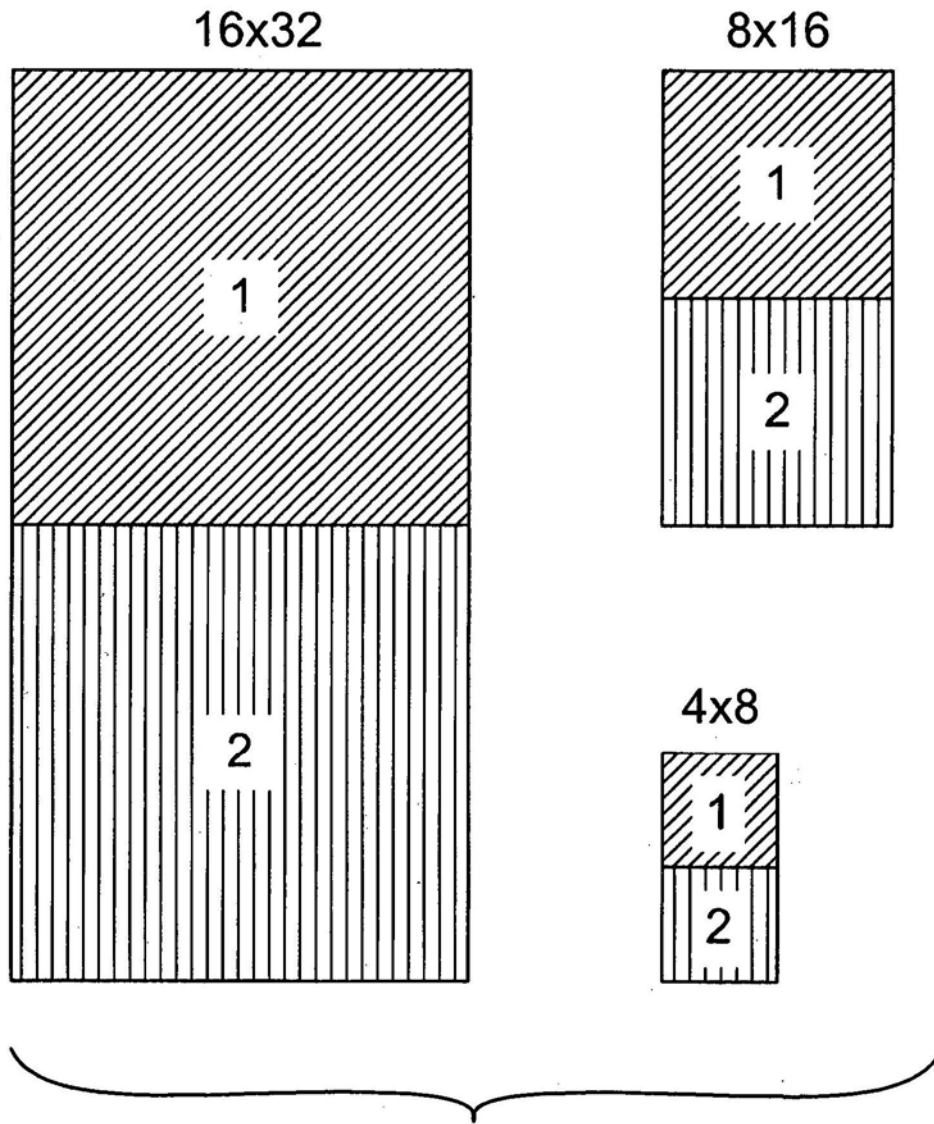


图16

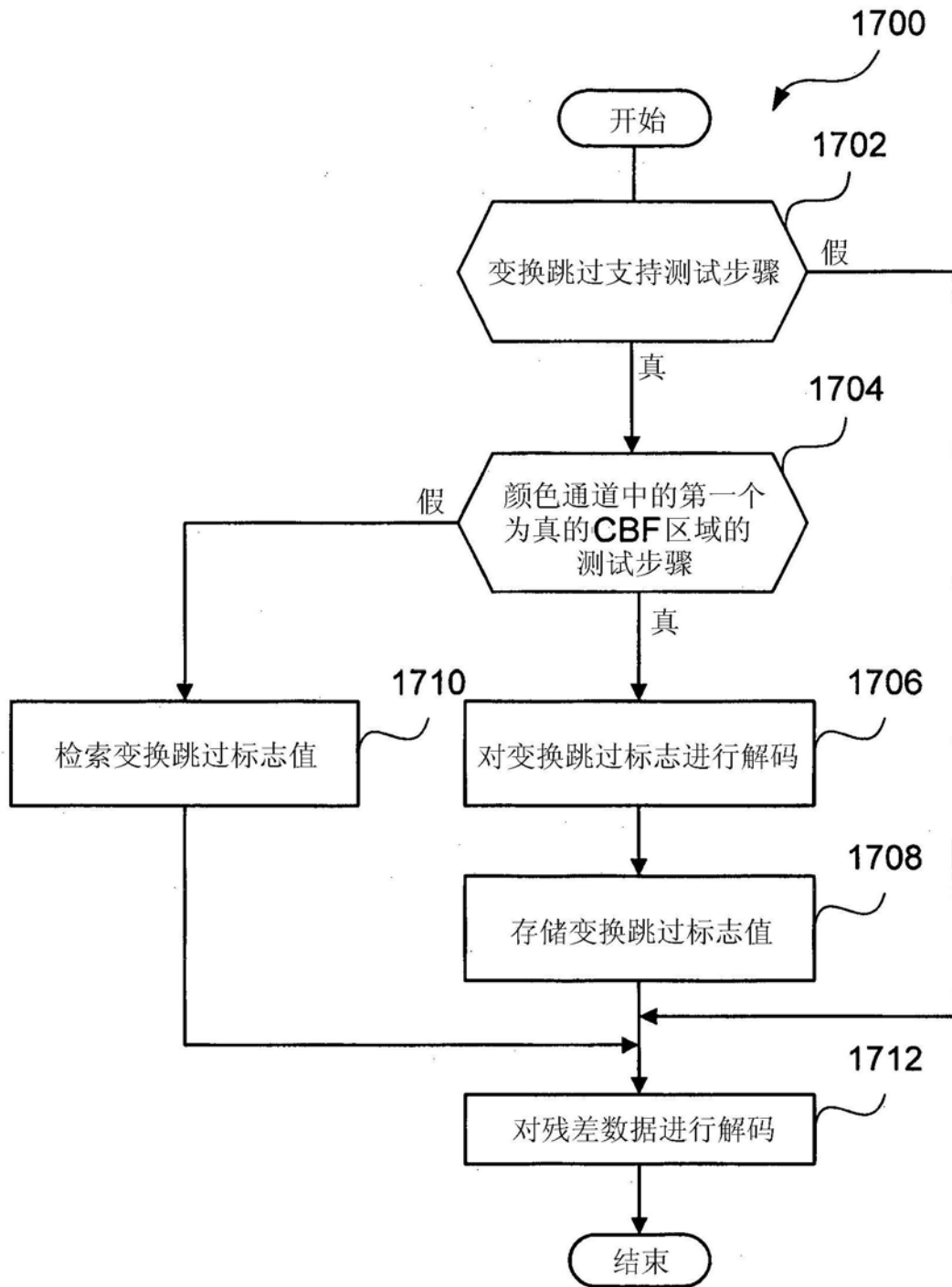


图17

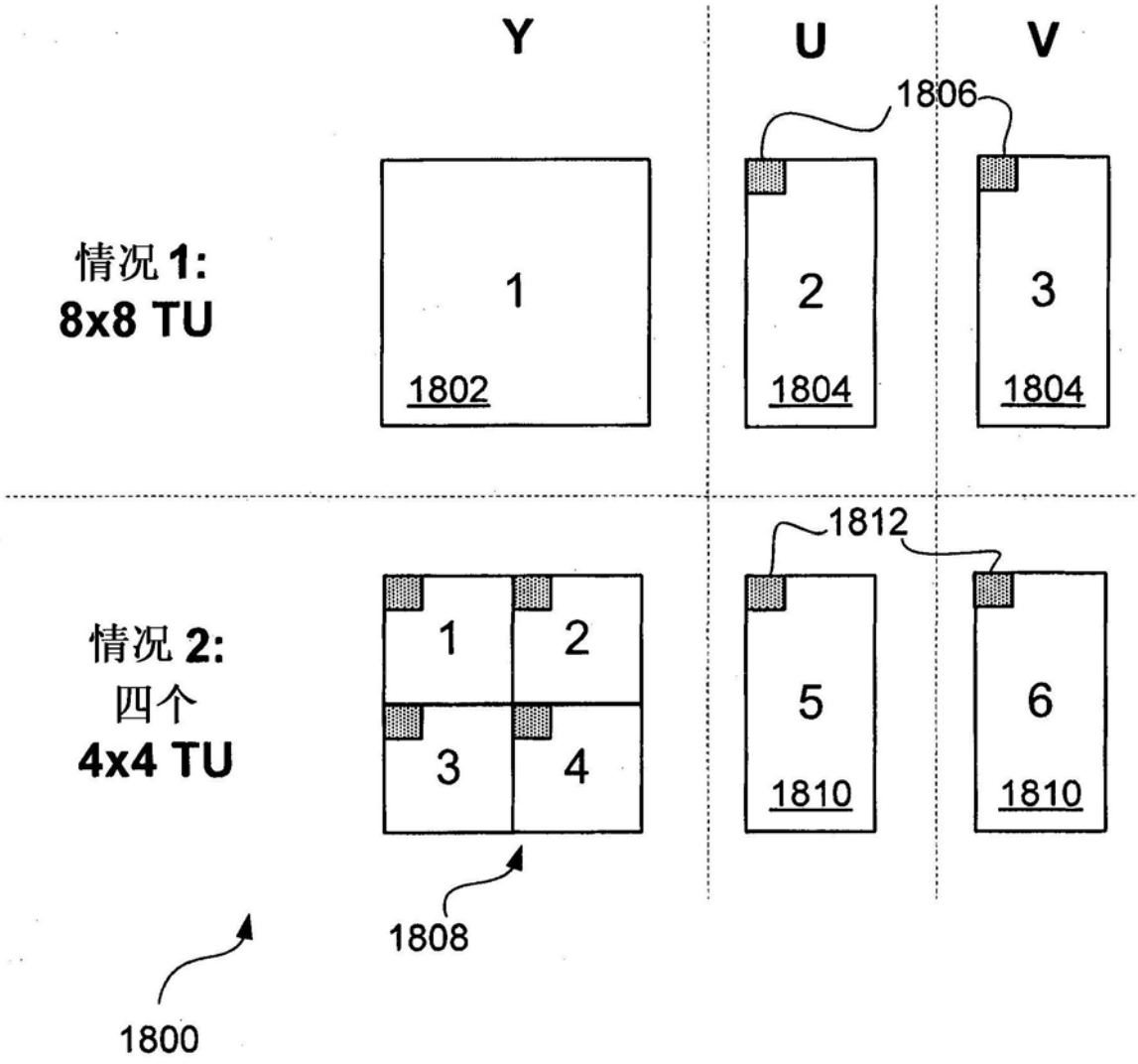


图18