



(12) 发明专利

(10) 授权公告号 CN 110708547 B

(45) 授权公告日 2024. 04. 09

(21) 申请号 201910614873.4

(22) 申请日 2019.07.09

(65) 同一申请的已公布的文献号  
申请公布号 CN 110708547 A

(43) 申请公布日 2020.01.17

(30) 优先权数据  
62/696,266 2018.07.10 US  
16/121,444 2018.09.04 US

(73) 专利权人 三星显示有限公司  
地址 韩国京畿道

(72) 发明人 维贾雅拉加万·蒂鲁玛莱

(74) 专利代理机构 北京德琦知识产权代理有限公司 11018  
专利代理师 史迎雪 康泉

(51) Int.Cl.

- H04N 19/122 (2014.01)
- H04N 19/13 (2014.01)
- H04N 19/184 (2014.01)
- H04N 19/186 (2014.01)

(56) 对比文件

- CN 108141591 A, 2018.06.08
- US 2004240556 A1, 2004.12.02
- US 2009304292 A1, 2009.12.10
- US 2010008417 A1, 2010.01.14
- US 6278735 B1, 2001.08.21
- US 9350988 B1, 2016.05.24

审查员 田亚平

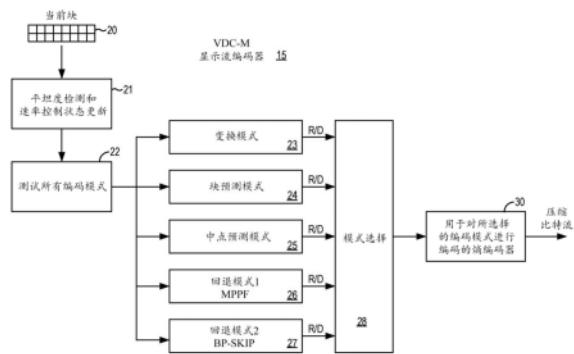
权利要求书4页 说明书13页 附图14页

(54) 发明名称

针对变换模式的有效熵编码分组方法

(57) 摘要

本发明涉及针对变换模式的有效熵编码分组方法。一种在以变换模式进行操作的熵编码器中形成熵编码组的系统和方法,包括接收第一数量的量化变换系数的块以作为当前块的样本值并且使用一个或多个分组因子评估当前块的样本值。响应于确定当前块满足第一分组条件,选择第一分组方法,其中第一分组方法形成具有一个样本值的第一熵编码组,该样本值是DC变换系数。响应于确定当前块满足第二分组条件,选择第二分组方法,其中第二分组方法形成具有至少两个样本值的第一熵编码组,样本值中一个样本值是DC变换系数。



1. 一种在以变换模式操作的熵编码器中形成熵编码组的方法,所述方法包括:  
接收由所述变换模式生成的第一数量的量化变换系数的块,以作为当前块的样本值;  
使用至少一个分组因子评估所述当前块的样本值,其中所述至少一个分组因子包括以下因子中的一个或多个:块在切片或视频帧内的位置;在变换模式处理期间选取的帧内预测模式;和输入视频源的颜色分量;

基于所述当前块的所述评估,选择包括第一分组方法和第二分组方法的多种分组方法中的一个;

使用所选择的分组方法来形成第二数量的熵编码组,所述熵编码组包括第一熵编码组;

将每个熵编码组中的所述样本值转换为基于比特的表示;并且

提供所述第二数量的熵编码组的所述基于比特的表示作为输出,

其中,选择所述多种分组方法中的所述一个包括:

响应于基于所述至少一个分组因子确定所述当前块满足第一分组条件而针对所述当前块选择所述第一分组方法,其中所述第一熵编码组包括一个样本值,所述一个样本值包括DC变换系数;并且

响应于基于所述至少一个分组因子确定所述当前块满足第二分组条件而针对所述当前块选择所述第二分组方法,其中所述第一熵编码组包括至少两个样本值,所述至少两个样本值中的一个样本值包括所述DC变换系数。

2. 根据权利要求1所述的方法,其中,将每个熵编码组中的所述样本值转换为基于比特的表示包括:

使用公共前缀熵代码,将每个熵编码组中的所述样本值转换为二的补码的基于比特的表示。

3. 根据权利要求1所述的方法,其中,将每个熵编码组中的所述样本值转换为基于比特的表示包括:

使用公共前缀熵代码,将每个熵编码组中的所述样本值转换为符号幅度的基于比特的表示。

4. 根据权利要求1所述的方法,其中使用至少一个分组因子评估所述当前块的样本值包括:

评估所述当前块的样本值,以确定所述当前块在视频帧或视频切片内的位置。

5. 根据权利要求4所述的方法,其中,响应于基于所述至少一个分组因子确定所述当前块满足第一分组条件而针对所述当前块选择所述第一分组方法包括:

响应于确定所述当前块位于所述视频帧或所述视频切片的第一行中,针对所述当前块选择所述第一分组方法。

6. 根据权利要求5所述的方法,其中,响应于基于所述至少一个分组因子确定所述当前块满足第二分组条件而针对所述当前块选择所述第二分组方法包括:

响应于确定所述当前块位于所述视频帧或所述视频切片的除了所述第一行之外的位置,针对所述当前块选择所述第二分组方法。

7. 根据权利要求1所述的方法,其中,

所述第一数量的量化变换系数包括针对给出的视频内容格式的十六个量化变换系数,

并且所述第二数量的熵编码组包括四个熵编码组。

8. 根据权利要求1所述的方法, 其中,

在所述第二分组方法中形成的所述至少两个样本值包括至少一个其他样本值, 所述至少一个其他样本值包括AC变换系数。

9. 根据权利要求1所述的方法, 其中, 使用至少一个分组因子评估所述当前块的样本值包括:

评估所述当前块的样本值, 以确定所述当前块在视频帧或视频切片内的位置, 并且确定在所述变换模式中使用的帧内预测器。

10. 根据权利要求9所述的方法, 其中, 响应于基于所述至少一个分组因子确定所述当前块满足第一分组条件而针对所述当前块选择所述第一分组方法包括:

响应于确定所述当前块位于所述视频帧或所述视频切片的第一行中或者所述当前块在所述变换模式中使用除了DC预测器之外的帧内预测器, 针对所述当前块选择所述第一分组方法。

11. 根据权利要求10所述的方法, 其中, 响应于基于所述至少一个分组因子确定所述当前块满足第二分组条件而针对所述当前块选择所述第二分组方法包括:

响应于确定所述当前块位于所述视频帧或所述视频切片的除了所述第一行之外的位置并且所述当前块在所述变换模式中使用所述DC预测器作为所述帧内预测器, 针对所述当前块选择所述第二分组方法。

12. 根据权利要求10所述的方法, 其中, 所述量化变换系数的块是从具有多个分量的像素值块导出的, 所述量化变换系数的块与所述像素值块的一个分量相关联, 所述方法进一步包括:

对于与其他分量中的每个分量相关联的每个量化变换系数的块, 使用所选择的分组方法形成所述第二数量的熵编码组。

13. 一种用于在以变换模式操作的熵编码器中形成熵编码组的系统, 包括:

处理器; 以及

存储器, 与所述处理器耦接, 其中所述存储器被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

接收由所述变换模式生成的第一数量的量化变换系数的块, 以作为当前块的样本值;

使用至少一个分组因子评估所述当前块的样本值, 其中所述至少一个分组因子包括以下因子中的一个或多个: 块在切片或视频帧内的位置; 在变换模式处理期间选取的帧内预测模式; 和输入视频源的颜色分量;

基于所述当前块的所述评估, 选择包括第一分组方法和第二分组方法的多种分组方法中的一个;

使用所选择的分组方法来形成第二数量的熵编码组, 所述熵编码组包括第一熵编码组;

将每个熵编码组中的所述样本值转换为基于比特的表示; 并且

提供所述第二数量的熵编码组的所述基于比特的表示作为输出,

其中, 所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

响应于基于所述至少一个分组因子确定所述当前块满足第一分组条件而针对所述当前块选择所述第一分组方法,其中所述第一熵编码组包括一个样本值,所述一个样本值包括DC变换系数;并且

响应于基于所述至少一个分组因子确定所述当前块满足第二分组条件而针对所述当前块选择所述第二分组方法,其中所述第一熵编码组包括至少两个样本值,所述至少两个样本值中的一个样本值包括所述DC变换系数。

14. 根据权利要求13所述的系统,其中,所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

评估所述当前块的样本值,以确定所述当前块在视频帧或视频切片内的位置;并且

响应于确定所述当前块位于所述视频帧或所述视频切片的第一行中,针对所述当前块选择所述第一分组方法。

15. 根据权利要求14所述的系统,其中,所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

响应于确定所述当前块位于所述视频帧或所述视频切片的除了所述第一行之外的位置,针对所述当前块选择所述第二分组方法。

16. 根据权利要求13所述的系统,其中,所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

评估所述当前块的样本值,以确定所述当前块在视频帧或视频切片内的位置,并且确定在所述变换模式中使用的帧内预测器;并且

响应于确定所述当前块位于所述视频帧或所述视频切片的第一行中或者所述当前块在所述变换模式中使用除了DC预测器之外的帧内预测器,针对所述当前块选择所述第一分组方法。

17. 根据权利要求16所述的系统,其中所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

响应于确定所述当前块位于所述视频帧或所述视频切片的除了所述第一行之外的位置并且所述当前块在所述变换模式中使用所述DC预测器作为所述帧内预测器,针对所述当前块选择所述第二分组方法。

18. 根据权利要求13所述的系统,其中,所述存储器进一步被配置为向所述处理器提供在被执行时使所述处理器执行以下操作的指令:

使用公共前缀熵代码,将每个熵编码组中的所述样本值转换为二的补码或符号幅度的基于比特的表示。

19. 根据权利要求13所述的系统,其中,

在所述第二分组方法中形成的所述至少两个样本值包括至少一个其他样本值,所述至少一个其他样本值包括AC变换系数。

20. 一种对压缩比特流进行解码的方法,所述方法包括:

接收压缩比特流,所述压缩比特流在变换模式操作中在利用多种分组方法的熵编码器中使用编码方法被编码;

处理所接收的压缩比特流中的当前块的样本值;

使用一个或多个分组因子评估所述当前块的样本值,其中所述至少一个分组因子包括

以下因子中的一个或多个:块在切片或视频帧内的位置;在变换模式处理期间选取的帧内预测模式;和输入视频源的颜色分量;

从被用于对所述当前块进行编码的所述多种分组方法中确定一分组方法,所述多种分组方法包括:第一分组方法,形成第二数量的熵编码组,其中第一熵编码组包括一个样本值,所述一个样本值包括DC变换系数;以及第二分组方法,形成所述第二数量的熵编码组,其中第一熵编码组包括至少两个样本值,所述至少两个样本值中的一个样本值包括所述DC变换系数;

响应于所述确定,使用所确定的分组方法对所述当前块进行解码;并且生成针对每个分量的量化变换系数,以作为输出。

## 针对变换模式的有效熵编码组分组方法

[0001] 相关申请的交叉引用

[0002] 本申请要求2018年7月10日提交的题为“针对变换模式的有效熵编码组分组方法”的第62/696,266号美国临时专利申请的优先权,该申请出于所有目的通过引用合并于此。

### 技术领域

[0003] 本公开涉及一种使用变换模式的熵编码的方法,特别地,涉及一种在高吞吐量熵编码器中针对变换模式的熵编码组分组方法。

### 背景技术

[0004] VESA(视频电子标准协会)针对移动或智能电话显示器或手持设备而开发出一种被称为VESA显示压缩-移动或VDC-M的新的显示接口压缩标准。VDC-M以压缩比高于显示流压缩(DSC)为目标,同时以增加复杂性为代价保持了如DSC的视觉无损质量。VDC-M适用于使用显示链路(例如MIPI显示串行接口(DSI))的移动设备。VDC-M显示接口压缩标准在Jacobson等人的“VESA正在开发的一种新的显示流压缩标准(A new display stream compression standard under development in VESA)”,Proc.SPIE 10396,数字图像处理XL的应用,103960U(2017年9月19日)中描述。

[0005] 特别地,VDC-M压缩标准是固定速率编解码器(编码器-解码器)并且针对具有每个颜色分量为8比特的RGB 4:4:4信号源支持降至6比特/像素的压缩率。VDC-M是基于块的编解码器,块大小为 $8 \times 2$ 个像素。在操作中,VDC-M编解码器通过对每个块应用或测试多个编码模式并且基于速率-失真成本为该块选择最佳编码模式,来进行操作。可以为每个块选择不同的编码模式。用所选择的编码模式,使用熵编码器对数据样本进行编码以生成压缩比特流。实现子流复用,以使压缩比特流的并行解析成为可能,从而实现高的解码器吞吐量。

### 发明内容

[0006] 本公开公开了一种用于形成熵编码组的设备和方法,该熵编码组大致如以下所示和/或如在以下所描述的(例如结合至少一张附图)那样,如在权利要求中更完整地阐述的那样。

[0007] 根据以下的描述和附图,将更加全面地理解本公开的这些和其他优点、方面及新颖特征、以及本公开的所示实施例的细节。

[0008] 在一个实施例中,一种在以变换模式操作的熵编码器中形成熵编码组的方法包括:接收由变换模式生成的第一数量的量化变换系数的块,以作为当前块的样本值;使用一个或多个分组因子评估当前块的样本值;响应于基于至少一个分组因子确定当前块满足第一分组条件,针对当前块选择第一分组方法,该第一分组方法形成第二数量的熵编码组,其中第一熵编码组包括一个样本值,该样本值是DC变换系数;响应于基于至少一个分组因子确定当前块满足第二分组条件,针对当前块选择第二分组方法,该第二分组方法形成第二数量的熵编码组,其中第一熵编码组包括至少两个样本值,样本值中一个样本值是DC变换

系数;使用所选择的分组方法来形成第二数量的熵编码组;将每个熵编码组中的样本值转换为基于比特的表示;并且提供第二数量的熵编码组的基于比特的表示作为输出。

[0009] 在另一实施例中,一种用于在以变换模式操作的熵编码器中形成熵编码组的系统包括:处理器;以及存储器,与处理器耦接,其中存储器被配置为向处理器提供指令,该指令在被执行时使处理器:接收由变换模式生成的第一数量的量化变换系数的块,以作为当前块的样本值;使用一个或多个分组因子评估当前样本值块;响应于基于至少一个分组因子确定当前块满足第一分组条件,针对当前块选择第一分组方法,该第一分组方法形成第二数量的熵编码组,其中第一熵编码组包括一个样本值,该样本值是DC变换系数;响应于基于至少一个分组因子确定当前块满足第二分组条件,针对当前块选择第二分组方法,该第二分组方法形成第二数量的熵编码组,其中第一熵编码组包括至少两个样本值,样本值中的一个样本值是DC变换系数;使用所选择的分组方法形成第二数量的熵编码组;将每个熵编码组中的样本值转换为基于比特的表示;以及提供第二数量的熵编码组的基于比特的表示作为输出。

[0010] 在另一实施例中,一种对压缩比特流进行解码的方法包括:接收压缩比特流,该压缩比特流在变换模式操作中在利用多种分组方法的熵编码器中使用编码方法被编码;处理所接收的压缩比特流中的当前块的样本值;使用一个或多个分组因子评估当前块的样本值;从被用于对当前块进行编码的多种分组方法中确定一分组方法,该多种分组方法包括:第一分组方法,形成第二数量的熵编码组,其中第一熵编码组包括一个样本值,该样本值包括DC变换系数;以及第二分组方法,形成第二数量的熵编码组,其中第一熵编码组包括至少两个样本值,样本值中的一个样本值包括DC变换系数;响应于该确定,使用所确定的分组方法对当前块进行解码;并且生成针对每个分量的量化变换系数,以作为输出。

## 附图说明

[0011] 本公开的各种实施例在以下详细描述和附图中被公开。

[0012] 图1是示出一些示例中用于实现显示接口编解码器的系统配置的示意图。

[0013] 图2是示出一些示例中显示流编码器的操作的过程图。

[0014] 图3是示出一些示例中在显示流编码器中的变换模式操作的过程图。

[0015] 图4示出一些示例中可被用在变换编码模式下的一组8个帧内预测器。

[0016] 图5是示出一些示例中熵编码器的操作的过程图。

[0017] 包括图6(a)至图6(d)的图6示出一些示例中的熵编码组结构以及公共前缀熵代码(CPEC)的输出。

[0018] 包括图7(a)和图7(b)的图7示出传统分组方法,该传统分组方法被用于针对通过变换模式被编码的样本值块中的一个分量形成熵编码组。

[0019] 包括图8(a)和图8(b)的图8示出传统分组方法,该传统分组方法被用于针对通过变换模式被编码的8个样本值块中的一个分量形成熵编码组。

[0020] 包括图9(a)至图9(d)的图9示出一些实施例中的三种分组方法,该三种分组方法可被应用以针对变换模式形成熵编码组。

[0021] 包括图10(a)至图10(c)的图10示出一些实施例中的两种分组方法,该两种分组方法可被应用以针对4:2:2的视频内容的变换模式的色度分量形成熵编码组。

[0022] 图11是示出一些实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。

[0023] 图12是示出在替代实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。

[0024] 图13是示出在替代实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。

[0025] 图14是示出一些实施例中在显示流解码器中实现的解码方法的流程图。

### 具体实施方式

[0026] 根据本公开的实施例,显示接口编解码器使用变换编码模式来执行视频信号压缩以对像素值块进行编码,并通过应用两种或更多种分组方法来实现熵编码,用以针对从像素值块中的对应分量生成的量化变换系数的块中的每个分量形成熵编码组(ECG)。在一些实施例中,显示接口编解码器包括显示流编码器,该显示流编码器实现多种分组方法以针对量化变换系数的块中的每个分量形成熵编码组(ECG)。在一些实施例中,分组方法包括仅用DC系数形成第一熵编码组的第一分组方法以及用DC系数和一些AC系数形成第一熵编码组的第二分组方法。显示流编码器中的熵编码器应用一个或多个因子来选择分组方法以用于每个块,使得由此形成的熵编码组内的样本值将具有相似范围内的样本值。以这种方式,使解码器操作更有效并且可以实现高的解码器吞吐量。

[0027] 在本公开的实施例中,显示接口编解码器包括显示流解码器,该显示流解码器实现编码器熵编码组分组方法以对压缩比特流进行解码。具体地,显示流解码器接收在编码器处使用多种分组方法生成的压缩比特流。解码器应用被用于在编码器处选择分组方法的一个或多个因子,来确定被用于压缩比特流中的每个块的分组方法。然后,解码器基于由编码器使用的分组方法来对每个块进行解码。

[0028] 在本公开的实施例中,显示接口编解码器基于被称为VESA显示压缩移动(或VDC-M)的用于移动或智能电话显示器的显示接口压缩标准来执行视频信号压缩。VDC-M是固定速率编解码器(编码器-解码器),并且针对RGB 4:4:4源支持降至6比特/像素的压缩率。图1是示出用于一些示例中实现显示接口编解码器的系统配置的示意图。参考图1,电子设备10包括通过显示链路13与显示驱动器集成电路16通信的应用处理器12。应用处理器12包括显示引擎14,该显示引擎14接收视频内容并实现显示流编码器15,例如基于VDC-M显示接口压缩标准的显示流编码器。编码后的比特流通过显示链路13从显示流编码器15被发送到显示驱动器集成电路16。显示链路13可以是物理链路或无线链路。显示驱动器集成电路16包括帧缓冲器17,该帧缓冲器17用于存储输入的比特流并将接收到的比特流提供给显示流解码器18,例如基于VDC-M显示接口压缩标准的显示流解码器。

[0029] 在一些实施例中,显示流编码器15和显示流解码器18都被实现为硬件部件,例如半导体集成电路。在一些实施例中,显示流编码器15和显示流解码器18被实现为专用集成电路(ASIC)。在其他实施例中,显示流解码器18以硬件实现,而显示流编码器15在主机系统上以应用处理器12中的软件或固件实现。

[0030] 图2是示出一些示例中显示流编码器的操作的过程图。VDC-M显示接口压缩标准是基于块的编解码器,块大小为 $8 \times 2$ 个像素。因此,显示流编码器15以 $8 \times 2$ 个像素(例如,在8



列和2行中)的块来对视频数据进行操作。参考图2,显示流编码器15接收 $8 \times 2$ 像素的当前块(20)的像素值。每个像素块包括针对一个或多个颜色分量的像素值。例如,每个像素块通常包括针对三个颜色分量的像素值。显示流编码器15执行平坦度检测以及速率控制状态更新(21)。然后,显示流编码器15针对当前像素块的像素值而并行地测试所有组的编码模式(22)。在本示例中,应用了编码模式23-27。对于每个编码模式,显示流编码器15确定该模式的速率R(是该模式所需的所有语法位的总和)以及失真D。显示流编码器15确定每个编码模式的RD成本。显示流编码器15选择具有受制于速率控制约束的最小RD成本的编码模式(28)。然后,显示流编码器15使用熵编码器30对所选择的编码模式进行编码。显示流编码器15生成可被发送到显示链路13的压缩比特流。然后,显示流编码器15移动至下一个像素块。在操作中,针对视频内容的帧中的每个像素块可以选择不同的编码模式。

[0031] 在本说明书中,“分量”或“颜色分量”指的是被用于表示视频信号的颜色分量,“分量”或“颜色分量”可以是RGB或YCbCr或YUV。

[0032] 显示流编码器15中的每个编码模式具有针对特定类型的视频内容而定制的属性。在本示例中,显示流编码器15实现以下的编码模式:

[0033] (1) 变换模式(23) - 变换模式对于捕获输入视频源中的自然内容是有用的。变换模式的示例是离散余弦变换。

[0034] (2) 块预测(BP)模式(24) - 块预测模式对于捕获输入视频源中的图形和纹理内容是有用的。

[0035] (3) 中点预测(MPP)模式(25) - 中点预测模式在捕获输入视频源中的非相关噪声内容时是有用的。

[0036] (4) 回退模式(26、27) - 提供一个或多个回退模式以在其他模式不可承受时保证固定速率压缩。在本示例中,提供了中点预测回退(MPPF)模式(26)和块预测跳过(BP-Skip)模式(27)。

[0037] 变换模式

[0038] 现在将更详细地描述显示流编码器中的变换编码模式。图3是示出一些示例中在显示流编码器中的变换模式操作的过程图。变换模式23将帧内预测应用于当前像素值块以生成预测像素值块(32)。具体地,在变换编码模式中,显示流编码器15针对当前块测试一组帧内预测器,用以确定哪个帧内预测器产生最小的RD成本。从该组帧内预测器中选择最佳的帧内预测器。用选择出的帧内预测器,在变换模式下的显示流编码器15生成残差块,该残差块是当前块的像素值与由所选择的帧内预测器生成的预测块的像素值之间的差(34)。然后,显示流编码器15将离散余弦变换(DCT)应用于该残差块的像素值以生成变换系数的块(36)。对变换系数的块进行量化,以生成量化变换系数的块(38)。当变换模式是由显示流编码器15选择出的编码模式时,量化变换系数随后由熵编码器30编码并被发送到显示链路13。

[0039] 图4示出了一些示例中可被用在变换编码模式下的一组8个帧内预测器。参考图4,可在变换编码模式下被应用以执行帧内预测的一组8个帧内预测器包括DC、垂直、垂直左、垂直右、对角线左、对角线右、水平左和水平右。在本示例中,使用带有修改的与在H.264视频编码器中使用的帧内预测器相类似的帧内预测器。具体地,该修改可以包括不使用紧邻左侧块中的样本以便允许低复杂度和低延迟。

[0040] 对于每个像素块,每个帧内预测器使用来自前一行的像素值(在图4中被表示为像素 $A_{-4}$ 至 $A_{-11}$ )来生成预测像素值块。在当前块位于帧或切片中的第一行的情况下,当前块没有可用的邻域。在那种情况下,使用DC预测器。也就是说,将等于源数据的动态范围的一半的DC值用作针对当前块的预测像素值。例如,为128的DC值将被用于每通道8比特(bpc)的视频源。对于不在第一行中的块,可以使用其他的帧内预测器(例如垂直或垂直左),其中前一行中的像素值被用于预测当前块中的像素值。

[0041] 熵编码器

[0042] 返回图2,用所选择的编码模式,显示流编码器15使用熵编码器30对样本进行编码。现在将描述熵编码器30的操作。图5是示出一些示例中熵编码器的操作的过程图。熵编码器40被设计为通过将块的每个颜色分量中的样本划分为单独的组来促成在解码器处的高吞吐量。参考图5,熵编码器40接收由所选择的编码模式生成的每个颜色分量16个数据值的当前块。在本说明书中,针对每个颜色分量的当前块的数据值也被称为样本,并且熵编码器可以被描述为接收针对每个颜色分量的16个样本的块。对于与相应颜色分量相关联的每个数据值块,熵编码器40将一个或多个样本分组为熵编码组(ECG)(42)。具体地,针对每个分量的当前块中的样本被分布在预定数量的熵编码组中。例如,对于4:4:4的视频内容,可以将针对一分量的16个样本的块分布在4个熵编码组中。然后,熵编码器40将每个组中的样本值转换为预定的基于比特的表示(44)。例如,熵编码器40可以将数据值转换为二的补码表示。可替代地,熵编码器40可以将数据值转换为符号幅度表示。然后,熵编码器40确定针对该块的每个组的前缀值(46)。利用由此针对每个组形成的前缀和后缀,熵编码器40将熵编码组打包成要被发送的压缩比特流(48)。以这种方式,针对视频内容的每个颜色分量的编码后的数据将在压缩比特流中作为一组熵编码组被发送。

[0043] 由熵编码器40编码的数据值是所选择的编码模式的函数。在一个示例中,当选择了变换模式23时,熵编码器40对量化变换系数进行编码。也就是说,数据值是量化变换系数。在另一示例中,当选择了块预测模式24时,熵编码器40对量化的预测残差进行编码。

[0044] 在以上描述中,假设视频内容是4:4:4的视频内容。在那种情况下,针对每个颜色分量(例如RGB)存在每块16个样本。在其他示例中,视频内容可以是4:2:2的视频内容。对于4:2:2的视频内容,亮度分量包含16个样本的块,并且两个色度分量仅包含8个样本或 $4 \times 2$ 的块大小。在又一示例中,视频内容可以是4:2:0的视频内容。对于4:2:0的视频内容,亮度分量包含16个样本的块,并且两个色度分量仅包含4个样本或 $2 \times 2$ 的块大小。

[0045] 熵编码器40接收针对当前数据值块中的每个颜色分量的16个样本,并且将这些样本划分成熵编码组。然后将每个熵编码组中的样本值编码成基于比特的表示。在本说明书中,使用公共前缀熵代码(CPEC)将熵编码组转换为基于比特的表示。包括图6(a)至6(d)的图6示出了一些示例中的熵编码组结构以及公共前缀熵代码(CPEC)的输出。参考图6(a),熵编码器40将样本划分为熵编码组,其中每个熵编码组由可变长度的前缀以及一个或多个固定长度的后缀组成。在图6(a)所示的示例中,熵编码组包括与4个样本相关联的4个后缀。

[0046] 对于每个熵编码组,应用CPEC以将样本值转换为基于比特的表示。具体地,具有组大小N或N个样本的熵编码组包括可变长度的前缀,后跟N个固定长度的后缀。使用CPEC,前缀使用可变长度代码(例如,一元代码)被编码并且表示用于对每个后缀进行编码的比特的数量。组内的实际样本值在后缀中表示,针对组内的每个样本有一个后缀。组内的所有N个

后缀都被分配了相同的比特数,即固定长度代码。例如,样本可以以二的补码表示或符号幅度表示的形式来表示。特别地,使用CPEC的熵编码组中的每个后缀与被编码为给定的基于比特的表示的一个样本值相对应。

[0047] 图6(b)示出了针对具有4个样本的熵编码组的示例CPEC结构,该4个样本具有样本值{1, -2, -1, 0}。在该示例中,二的补码表示被用于生成熵编码组的比特。忠实地重建该组所需的比特数为2。特别地,在二的补码表示中,需要n比特来表示在范围 $[-2^{(n-1)}, 2^{(n-1)} - 1]$ 中的数据。在图6(b)的示例中,前缀将以信号通知值2。对于样本值组{1, -2, -1, 0},由CPEC操作输出的比特将是前缀110(2的一元代码)及4个后缀,其中每个样本使用2个比特(例如“01 10 11 00”)来进行编码,如图6(b)所示。图6(b)的示例中的CPEC操作的输出仅是说明性的,并且CPEC操作的实际输出可以根据实际实现而不同。

[0048] 如此构造,熵编码器对块的每个分量的样本进行划分以生成一组熵编码组,然后在压缩比特流中发送该组熵编码组。在熵编码器处,当前组的可变长度的前缀被解析,并且当前组中剩余数量的后缀比特被获知。因此,熵编码器可以跳转并开始解析下一组的前缀,而解码器对前一组的后缀进行解码,如图6(c)所示。由于可变长度代码的有限的数量(对于大小为 $8 \times 2$ 的块中的一个颜色分量而言,最多为4),因此解码器可以实现高吞吐量。

[0049] 图6(b)所示的示例使用了二的补码表示。诸如符号幅度的其他的表示也可被用在CPEC操作中以生成用于每个熵编码组的比特。使用符号幅度表示的CPEC操作的输出如图6(d)所示。参考图6(d),符号幅度表示被采用,每个样本需要n个比特,其中组中的所有样本的绝对值(即幅度)在 $[0, 2^n - 1]$ 的数据范围内。在符号幅度表示中,符号位仅以信号通知非零值。对于样本值{1, -2, -1, 0}的输入组,以符号幅度表示的CPEC操作的输出将是前缀110(2的一元代码),后跟后缀及最后的符号位。后缀使用2个比特(例如,01 10 01 00)来对每个样本的绝对值进行编码,并且符号位可以是1 0 0,假设1用于正的符号并且0用于负的符号。注意到,输入组的最后一个样本值0的符号值未被以信号通知。

[0050] 当组中的所有样本都具有零值时,针对整个组仅以信号通知一个比特,这在本公开中被称为组跳过。组跳过适用于使用符号幅度和二的补码表示的CPEC。

[0051] 熵编码器40确定将样本分组为针对样本块中的每个分量的熵编码组。每个ECG的大小(即每个ECG中要包括的样本的数量)将取决于编码模式、颜色分量和色度采样格式。熵编码组内的样本分布也取决于编码模式。

[0052] 包括图7(a)和图7(b)的图7示出了传统分组方法,该传统分组方法被用于针对由变换模式编码的样本值块中的一个分量形成熵编码组。参考图7(a),在变换编码模式的情况下,针对一个分量的每个样本值块包含 $8 \times 2$ 或16个量化变换系数S0至S15。熵编码器将16个样本分布在4个熵编码组ECG0至ECG3中,其中每个组使用CPEC结构被编码,如图7(b)所示。特别地,对于变换模式,采用了非均匀分组方法,其中针对每个熵编码组的样本的大小或数量并不相同。非均匀样本分布被用于变换模式,使得变换域中具有相似频率的样本被分组在一起。参考图7(b),针对变换模式中的样本的传统分组方法形成大小为1、3、5和7的熵编码组。DC值S0被认为是单独的组ECG0。组ECG1包含三个样本S1、S2和S8。组ECG2包含五个样本S3、S4、S9、S10和S11。组ECG3包含七个样本S5、S6、S7、S12、S13、S14和S15。特别地,分配给每个组的样本可以在 $8 \times 2$ 的块内按Z字形图案选择出。

[0053] 在4:4:4的视频内容的情况下,每个颜色分量(例如RGB)包含相同数量的样本,即,

每块16个样本。因此,图7中的分组方法可被用于所有的颜色分量。在4:2:2的视频内容的情况下,亮度分量包含16个样本的块,并且因此可以使用图7中的分组方法。但是,色度分量仅包含8个样本或 $4 \times 2$ 的块大小。在那种情况下,应用不同的分组方法。包括图8(a)和图8(b)的图8示出了传统分组方法,该传统分组方法被用于针对由变换模式编码的8个样本值块中的一个分量形成熵编码组。参考图8(a),对于4:2:2的视频内容,色度分量由 $4 \times 2$ 的块大小或8个样本来表示。在那种情况下,传统的分组方法可以形成分别具有1个样本和7个样本的两个熵编码组ECG0和ECG1,如图8(b)所示。

[0054] 为了描述的完整性,在块预测模式的情况下,采用了均匀分组方法,其中每个熵编码组包含相同数量的量化残差样本。例如,分组方法可以每组放置4个样本,以用于每个分量的 $8 \times 2$ 个的样本值块。在一个示例中,第一组ECG0可以包含样本{S0,S1,S8,S9},第二组ECG1可以包含样本{S2,S3,S10,S11},等等。

[0055] 返回图7,传统的熵编码器对在变换模式中生成并使用CPEC被编码的样本应用相同的分组方法。当组中的所有样本处于相似范围内时,因为所有的后缀长度具有相同的大小,所以传统分组方法将是有效的。然而,当组中的样本值在大范围内展开时,传统分组方法变得低效。例如,当组中的一个样本值远大于其余样本值时,将需要大量的比特。

[0056] 以下示例说明了要使用CPEC结构被编码的两组样本值。在第一个示例中,组中的样本值在相同范围内。在第二个示例中,一个样本值与其他样本值在不同范围内。

	场景 1	场景 2
[0057]	样本值{1,-2,-1,0} 每个后缀需要的比特: 2 前缀长度: 3 后缀长度: $4 \times 2 = 8$ 总长度: 11	样本值: {1,-15,-1,0} 每个后缀需要的比特: 5 前缀长度: 6 后缀长度: $4 \times 5 = 20$ 总长度: 26

[0058] 如以上示例中所示,当样本值在相同范围内时(场景1),可以使用11比特的组长度来表示该组。但是,当样本值不在相同范围内时(场景2),需要26比特的组长度来表示该组。较长的组长度在熵编码器处引入低效率。

[0059] 在传统的熵编码器中,在变换模式中使用相同的分组方法以形成熵编码组,而与块在切片或帧中的位置无关,进而与用于变换模式的帧内预测模式无关。使用相同的分组方法来构造熵编码组对于在切片的第一行(FLS)中或者不在FLS(NFLS)中的块而言不是有效的方法。

[0060] 在本说明书中,视频内容由视频帧表示,其中帧是构成完整运动画面的许多静止图像中的一个。此外,在本说明书中,切片指的是帧的空间上不同的区域,该区域可以与同一帧中的其他区域分开编码。在本说明书中,像素或像素块有时指的是在“切片的第一行”中。如本文所使用的,引用“切片的第一行”可以指在视频帧内的切片的第一行中的块或位于视频帧的第一行中的块。

[0061] 在视频帧中,在帧的第一行或在切片的第一行(FLS)中的块不具有前一行视频数

据。也就是说,对于切片的第一行中的块,相邻像素值是不可用的或受限的。在那种情况下,对于变换模式中的帧内预测操作,DC帧内预测器是唯一的选择。用于帧内预测器的DC值通常是动态范围的一半(例如,对于8比特的图像,DC值是128)。DC预测值不真实地反映当前块的实际DC值。使用DC帧内预测器导致更高的DC系数值。因此,最好在分组过程中将DC视为单独的组。

[0062] 另一方面,对于不在帧的第一行或切片的第一行(NFLS)中的块,相邻像素值是可用的。因此,从邻域正确地估计块的实际DC值是可能的。在这种情况下,预期DC系数为零或接近于零。在这种情况下,在单独的组中考虑DC系数是没有效率的。将DC系数与一些AC系数分组在一起实际上更有效。

[0063] 在本公开的实施例中,显示接口编解码器使用变换编码模式并且通过应用多种分组方法以形成熵编码组来实现熵编码,来执行视频信号压缩。更具体地,显示流编码器合并熵编码器,该熵编码器针对变换模式实现多种分组方法。基于一个或多个因子来针对块选择特定的分组方法。显示接口编解码器在不增加编解码器的复杂性的情况下提高了压缩性能,同时在解码器处保持了相同的吞吐量。

[0064] 如上所述,现有技术针对变换模式使用固定分组方案来形成针对所有块的熵编码组。也就是说,使用相同的分组方法来形成针对所有块的熵编码组。相反,本公开的显示流编码器提供针对变换模式的多种分组方法,并且基于一个或多个因子来针对块选择特定的分组方法。

[0065] 在一些实施例中,基于以下因子中的一个或多个来针对每个块选择分组方法:

[0066] (1) 块在切片或视频帧内的位置;

[0067] (2) 在变换模式处理期间选取的帧内预测模式;或者

[0068] (3) 输入视频源的颜色分量。

[0069] 在一个实施例中,显示流编码器实现M种分组方法,以将块中的分量的样本分组为针对变换模式的熵编码组。包括图9(a)至9(d)的图9示出了三种分组方法,该三种分组方法在一些实施例中可被应用于针对变换模式形成熵编码组。参考图9,根据变换模式处理,针对被表示为样本值S0至S15的每个分量生成16个量化变换系数的块,如图9(a)所示。显示流编码器实现各种分组方法以将16个样本分布在4个熵编码组中。每种分组方法在每个熵编码组的大小上可以不同,但是所有的分组方法都生成相同总数量的熵编码组,即,4个熵编码组。

[0070] 在本示例中,显示流编码器实现第一分组方法,该第一分组方法形成仅具有DC系数的第一熵编码组ECG0(S0),如图9(b)所示。也就是说,第一熵编码组ECG0的大小为1并且仅包括样本S0。其他熵编码组ECG1、ECG2和ECG3的大小分别为3、5和7。分配给组ECG1、ECG2和ECG3的样本可以在 $8 \times 2$ 的块内按Z字形图案选择,如图9(b)所示。

[0071] 显示流编码器实现第二分组方法,该第二分组方法形成具有DC系数和多个AC系数的第一熵编码组ECG0,如图9(c)所示。在本示例中,第一熵编码组ECG0的大小为3并且包括样本S0、S1和S2。其他熵编码组ECG1、ECG2和ECG3的大小分别为3、4和6。分配给组ECG1、ECG2和ECG3的样本可以在 $8 \times 2$ 的块内按Z字形图案选择,如图9(c)所示。

[0072] 最后,在一些实施例中,显示流编码器可以实现第三分组方法,该第三分组方法形成具有DC系数和一个AC系数的第一熵编码组ECG0,如图9(d)所示。在本示例中,第一熵编码

组ECG0的大小为2并且包括样本S0和S1。其他熵编码组ECG1、ECG2和ECG3的大小分别为4、4、6。分配给组ECG1、ECG2和ECG3的样本可以在 $8 \times 2$ 的块内按Z字形图案选择出,如图9(d)所示。

[0073] 显示流编码器中的熵编码器应用一个或多个因子来选择用于每个块的分组方法,使得由此形成的ECG组内的样本值将在相似的范围。图9中所示的三种分组方法仅是说明性的,并不旨在进行限制。在其他实施例中,可以实现两种或更多种分组方法,每个熵编码组可以具有相同数量或不同数量的样本,并且可以改变样本在组中的分布。所有的分组方法产生相同数量的组,例如4个,但是在4个组中的样本分布可以变化。

[0074] 上面的图9示出了针对4:4:4的视频内容的分组方法,其中块中的所有的三个分量都包括16个样本。针对4:2:2的视频内容,色度分量仅包含8个样本,并且亮度分量包含16个样本。包括图10(a)至10(c)的图10示出了一些实施例中的两种分组方法,该两种分组方法可被应用于针对4:2:2的视频内容的变换模式的色度分量形成熵编码组。参考图10(a),根据变换模式处理,8个量化变换系数的块被提供为针对色度分量的样本值S0至S7。显示流编码器实现第一分组方法,该第一分组方法形成仅具有DC系数的第一熵编码组ECG0,如图10(b)所示。在本示例中,第一熵编码组ECG0的大小为1并且仅包括样本S0。另一个熵编码组ECG1的大小为7。

[0075] 显示流编码器实现第二分组方法,该第二分组方法形成具有DC系数和多个AC系数的第一熵编码组ECG0,如图10(c)所示。在本示例中,第一熵编码组ECG0的大小为3并且包括样本S0、S1和S2。另一个熵编码组ECG1的大小为5。显示流编码器中的熵编码器应用一个或多个因子来选择用于每个块的分组方法,使得由此形成的ECG组内的样本值将在相似的范围。

[0076] 图11是说明在一些实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。在一些实施例中,熵编码分组方法可以在图2的显示流编码器15中的熵编码器30中实现。在变换编码模式中,针对像素值块的每个分量生成量化变换系数的块。参考图11,熵编码分组方法1100接收针对当前像素值块的每个分量由变换编码模式处理生成的S个量化变换系数的块(1102)。方法1100使用一个或多个分组因子评估样本(1104)。例如,方法1100可以确定当前块在切片或视频帧内的位置。可替代地,方法1100可以确定在变换模式处理期间选取的帧内预测模式。最后,方法1100可以确定正被处理的样本的颜色分量。

[0077] 基于分组因子的评估,方法1100可以应用包括仅具有DC系数的第一熵编码组的第一分组方法来形成熵编码组(1106)。也就是说,第一熵编码组的大小为1并且仅包括块的与DC系数相关联的第一样本值。在一个示例中,第一分组方法是图9(b)中所示的分组方法。

[0078] 可替代地,基于分组因子的评估,方法1100可以应用包括具有DC系数和一个或多个AC系数的第一熵编码组的第二分组方法来形成熵编码组(1108)。也就是说,第一熵编码组的大小大于1并且包括块的与DC系数相关联的第一样本值和与AC系数相关联的一个或多个其他样本值。在一个示例中,第二分组方法是图9(c)中所示的分组方法。

[0079] 使用所选择的分组方法,方法1100将块中的S数量个样本分布在G数量个熵编码组中。两种分组方法生成相同数量的G个熵编码组。分组方法的不同在于样本在G个熵编码组中的每一组中的分布。

[0080] 利用如此形成的熵编码组,方法1100使用CPEC将每个熵编码组中的样本值转换为

针对当前块的当前分量的基于比特的表示(1110)。使用CPEC,每个熵编码组中的样本值被使用所选择的基于比特的表示(例如符号幅度表示或二的补码表示)来表示。

[0081] 用如此形成的一组G个熵编码组,方法1100提供一组熵编码组作为输出(1112),并且方法1100选择下一个样本值块(1114)。方法1100在1102处重复,其中方法1100接收与新的像素值块的分量相关联的下一个量化变换系数的块。值得注意的是,块的所有颜色分量将使用同一分组方法。也就是说,同一分组方法将被应用于块中的针对该块的所有颜色分量的样本值。

[0082] 在一个示例中,方法1100接收要被分成4个组的16个样本的块。第一分组方法包括大小为1的第一熵编码组。也就是说,第一熵编码组仅具有DC系数的一个样本。第二分组方法包括大小大于1的第一熵编码组。也就是说,第一熵编码组具有DC系数和至少一个AC系数的两个或更多个样本。

[0083] 在一些实施例中,熵编码分组方法使用当前块在视频帧或视频切片内的位置来确定要使用的分组方法。图12是示出在替代实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。在一些实施例中,熵编码分组方法可以在图2的显示流编码器15中的熵编码器30中实现。在变换编码模式中,针对每个像素块的每个分量来生成量化变换系数的块。参考图12,熵编码分组方法1200接收由针对当前像素值块的分量由变换编码模式处理生成的S数量个量化变换系数的块(1202)。然后,方法1200评估当前块在视频帧或视频切片内的位置(1204)。具体地,方法1200确定当前块是否在帧的第一行或切片的第一行中(1206)。

[0084] 响应于确定当前块在帧或切片的第一行中,方法1200应用包括仅具有DC系数的第一熵编码组的第一分组方法来形成熵编码组(1208)。也就是说,第一熵编码组的大小为1并且仅包括块的与DC系数相关联的第一样本值。在一个示例中,第一分组方法是图9(b)中所示的分组方法。

[0085] 响应于确定当前块不在帧或切片的第一行中,方法1200应用包括具有DC系数和一个或多个AC系数的第一熵编码组的第二分组方法来形成熵编码组(1210)。也就是说,第一熵编码组的大小大于1并且包括块的与DC系数相关联的第一样本值和与AC系数相关联的一个或多个其他样本值。在一个示例中,第二分组方法是图9(c)中所示的分组方法。

[0086] 使用所选择的分组方法,方法1200将块中的S数量个样本分布在G数量个熵编码组中。两种分组方法生成相同数量的G个熵编码组。分组方法的不同在于样本在G个熵编码组中的每一组中的分布。

[0087] 用由此形成的熵编码组,方法1200使用CPEC将每个熵编码组中的样本值转换为针对当前块的当前分量的基于比特的表示(1212)。使用CPEC,每个熵编码组中的样本值使用所选择的基于比特的表示(例如,符号幅度表示或二的补码表示)来表示。

[0088] 用由此形成的一组G个熵编码组,方法1200提供一组熵编码组作为输出(1214),并且方法1200选择下一块(1216)。方法在1202处重复,其中方法1200接收与新的像素值块的分量相关联的下一个量化变换系数的块。值得注意的是,块的所有颜色分量将使用同一分组方法。也就是说,同一分组方法将被应用于块中的针对该块的所有颜色分量的样本值。

[0089] 在一个示例中,方法接收要被分成4个组的16个样本的块。第一分组方法包括大小为1的第一熵编码组。也就是说,第一熵编码组仅具有DC系数的一个样本。第二分组方法包

括具有大小大于1的第一熵编码组。也就是说,第一熵编码组具DC系数和至少一个AC系数的两个或更多个样本。

[0090] 如此构造,熵编码分组方法1200将第一分组方法用于视频帧或切片的第一行,并将第二分组方法用于视频帧和切片的所有其他行。相对于传统分组方法,熵编码分组方法1200实现了效率提高,在传统分组方法中,同一分组方法被应用于所有的像素块,而不管块在帧或切片内的位置。特别地,针对不在帧或切片的第一行中的块,可以从邻域的样本正确地估计块的实际DC值。因此,DC系数可以与其他AC变换系数处于相同的范围内是可能的。因此,不必将DC系数分离成单独的熵编码组。相反,DC系数可以与一个或多个AC系数组合。在构造第一熵编码组ECG0之后,将剩余的系数分隔成组(可以是均匀的或非均匀的),使得第一分组方法和第二分组方法之间的组的总数量相同。

[0091] 在一些实施例中,当两个条件(当前块不在帧或切片的第一行中以及变换模式帧内预测器是DC预测器)被满足时,熵编码分组方法应用第二分组方法(图9(c))。否则,使用第一分组方法(图9(b))。也就是说,第一分组方法被用于视频帧或切片的第一行中的块或者当变换模式帧内预测器不是DC预测器时。

[0092] 图13是示出在替代实施例中针对变换模式的在熵编码器中实现的熵编码分组方法的流程图。在一些实施例中,熵编码分组方法可以在图2的显示流编码器15中的熵编码器30中实现。在变换编码模式中,针对每个像素块的每个分量生成量化变换系数的块。参考图13,熵编码分组方法1300接收针对当前像素值块的当前分量由变换编码模式处理生成的S数量个量化变换系数的块(1302)。然后,方法1300评估当前块在视频帧或视频切片内的位置(1304)。具体地,方法1300确定当前块是否在帧的第一行或切片的第一行中(1304)。

[0093] 响应于确定当前块在帧或切片的第一行中,方法1300应用包括仅具有DC系数的第一熵编码组的第一分组方法来形成熵编码组(1308)。也就是说,第一熵编码组的大小为1并且仅包括块的与DC系数相关联的第一样本值。在一个示例中,第一分组方法是图9(b)中所示的分组方法。

[0094] 响应于确定当前块不在帧或切片的第一行中,方法1300继续确定被用于当前块的变换模式帧内预测器是否是DC预测器(1306)。响应于确定当前块在变换模式中不使用DC帧内预测器,方法1300应用第一分组方法来形成熵编码组(1308)。

[0095] 响应于确定被用于当前块的变换模式帧内预测器是DC预测器,方法1300应用包括具有DC系数和一个或多个AC系数的第一熵编码组的第二分组方法来形成熵编码组(1310)。也就是说,第一熵编码组的大小大于1并且包括块的与DC系数相关联的第一样本值和与AC系数相关联的一个或多个其他样本值。在一个示例中,第二分组方法是图9(c)中所示的分组方法。以这种方式,第二分组方法仅应用于不在帧或切片的第一行中并且使用DC帧内预测器的块。

[0096] 使用所选择的分组方法,方法1300将块中的S数量个样本分布在G数量个熵编码组中。两种分组方法生成相同数量的G个熵编码组。分组方法的不同在于样本在G个熵编码组中的每一组中的分布。

[0097] 利用由此形成的熵编码组,方法1300使用CPEC将每个熵编码组中的样本值转换为针对当前块的当前分量的基于比特的表示(1312)。使用CPEC,每个熵编码组中的样本值被使用所选择的基于比特的表示来表示。在一些实施例中,使用二的补码表示来表示样本值。



在其他实施例中,使用符号幅度表示来表示样本值。

[0098] 用由此形成的一组G个熵编码组,方法1300提供一组熵编码组作为输出(1314),并且方法1300选择下一块(1316)。该方法在1302处重复,其中方法1300接收与新的像素值块的分量相关联的下一个量化变换系数的块。值得注意的是,块的所有颜色分量将使用同一分组方法。也就是说,同一分组方法将被应用于块中的针对该块的所有颜色分量的样本值。

[0099] 在一个示例中,方法1300接收要被分成4个组的16个样本的块。第一分组方法包括大小为1的第一熵编码组。也就是说,第一熵编码组仅具有DC系数的一个样本。第二分组方法包括具有大小大于1的第一熵编码组。也就是说,第一熵编码组具有DC系数和至少一个AC系数的两个或更多个样本。

[0100] 如此构造,当所使用的帧内预测器不是DC预测器时,熵编码分组方法1300将第一分组方法用于视频帧或切片的第一行或视频帧或切片的其他行。当所使用的帧内预测器是DC预测器时,方法1300将第二分组方法用于视频帧和切片的所有其他行。相对于传统分组方法,熵编码分组方法1300实现了效率提高,在传统分组方法中,同一分组方法被应用于所有的像素块,而不管块在帧或片内的位置或者不管所使用的帧内预测器。特别地,对于选择DC预测器作为其最佳帧内预测器的块,预期是DC值与其他系数在相同的范围内。因此,在这种情况下,构造具有DC系数和一个或多个AC系数的第一熵编码组是更为有效的。

[0101] 可替代地,对于帧或切片的第一行中的所有块,或者对于不在第一行中但是使用了除DC预测器之外的帧内预测器的块,使用第一ECG方法,其中DC系数在其自己的组中被分离。

[0102] 上述熵编码分组方法仅是说明性的。在本公开的其他实施例中,可以使用其他的分组方法来形成熵编码组。在上述实施例中,熵编码分组方法被应用于 $8 \times 2$ 或 $4 \times 2$ 个像素的块大小。在其他实施例中,使用了多种分组方法的变换模式技术可被应用于任何块大小。

[0103] 此外,在上述实施例中,变换模式下的CPEC操作被描述为使用针对样本值的二的补码表示。在其他实施例中,变换模式的CPEC操作可以使用针对样本值的二的补码表示或符号幅度表示或者这两者。

[0104] 返回图1,在由此通过显示流编码器15形成的压缩比特流的情况下,该压缩比特流通过显示链路13被发送到显示驱动器集成电路16。在显示驱动器集成电路16处,显示流解码器18执行压缩比特流的解码。在本公开的实施例中,显示流解码器18被配置为实现由显示流编码器15使用的熵编码分组方法,以便对接收到的压缩比特流进行解码。图14是示出一些实施例中在显示流解码器中实现的解码方法的流程图。参考图14,解码方法1400接收压缩比特流(1402)。解码方法1400被配置为对使用上述熵编码分组方法被编码的压缩比特流进行解码,其中所接收的数据包括已在变换模式下使用多种分组方法被编码的块。

[0105] 解码方法1400处理来自压缩比特流的每个样本值块(1404)。对于正被处理的当前样本值块,解码方法1400使用一个或多个分组因子来评估当前块(1406)。特别地,解码方法1400使用与被用于在编码器处对块进行编码的一组分组因子相同的一组分组因子。例如,解码方法1400可以确定当前块在切片或视频帧内的位置。可替代地,方法1400可以确定在变换模式处理期间选取的帧内预测模式。最后,解码方法1400可以确定输入视频源的颜色分量。

[0106] 基于分组因子的评估,解码方法1400确定被用于对块进行编码的特定分组方法

(1408)。例如,解码方法1400确定该块是否已使用上述第一分组方法或第二分组方法被编码。例如,第一分组方法包括仅具有DC系数的第一熵编码组,并且第二分组方法包括具有DC系数和一个或多个AC系数的第一熵编码组。

[0107] 用所确定的分组方法,解码方法1400继续前进到使用所确定的分组方法解码当前块(1410)。特别地,解码方法1400对与块中的分量的量化变换系数相关联的比特进行解析,并对每个组的样本值进行解码。然后,通过应用在编码器侧执行的分组操作的逆,使用每个组的样本值来构造针对每个分量的 $8 \times 2$ 逐块量化变换系数。例如,执行图9(a)和图9(c)的分组操作的逆,以将熵编码组的样本值恢复到 $8 \times 2$ 逐块顺序。结果,获得了针对当前块的分量的 $8 \times 2$ 量化变换系数的块。

[0108] 然后,解码方法1400处理 $8 \times 2$ 的量化变换系数以获得当前块的像素值(1412)。例如,解码方法1400应用图3中的操作的逆,以从量化变换系数的块获得像素值块。例如,解码方法1400将像素数据提供为视频数据输出,其中像素数据可以被提供给显示器以用于视频回放。然后,解码方法1400选择下一块(1414),并且该方法在1404处重复,在1404处,处理所选择的块。

[0109] 在一个示例中,解码方法使用块在帧或切片内的位置来评估当前块。对于在帧或切片的第一行中的块,解码方法确定第一分组方法被用于对该块进行编码,并且解码方法使用第一分组方法的分组方案来对当前块进行解码。对于不在帧或切片的第一行中的块,解码方法确定第二分组方法被用于对该块进行编码,并且解码方法使用第二分组方法的分组方案来对当前块进行解码。

[0110] 本公开可以以多种方式来实现,包括作为过程、装置、系统、物质的组成、体现在计算机可读存储介质上的计算机程序产品、和/或处理器,例如被配置为执行被存储在耦接到处理器的存储器上的和/或由耦接到处理器的存储器提供的指令的硬件处理器或处理设备。在本说明书中,这些实现或本公开可以采用的任何其他形式可以被称为技术。通常,可以在本公开的范围内改变所公开的处理的步骤的顺序。除非另有说明,否则诸如描述为被配置为执行任务的处理器或存储器的部件可以被实现为被临时配置为在给定时间执行任务的通用部件或者被制造为执行任务的特定部件。如本文所使用的,术语“处理器”指的是被配置为处理数据(例如计算机程序指令)的一个或多个设备、电路和/或处理核。

[0111] 以上提供了本公开的一个或多个实施例的详细描述以及示出本公开的原理的附图。结合这些实施例描述了本公开,但是本公开不限于任何实施例。本公开的范围仅由权利要求限定,并且本公开包含许多替代、修改和等同物。在以上描述中阐述了许多具体细节,以便提供对本公开的透彻理解。提供这些细节是出于示例的目的,并且可以在没有这些具体细节中的一些或全部的情况下根据权利要求来实践本公开。出于清楚的目的,没有详细描述与本公开相关的技术领域中的已知的技术材料,从而使本公开不会不必要地被混淆。

[0112] 以上详细描述被提供以说明本公开的具体实施例,而并不旨在进行限制。在本公开的范围内的许多修改和变化都是可行的。本公开由所附权利要求来限定。

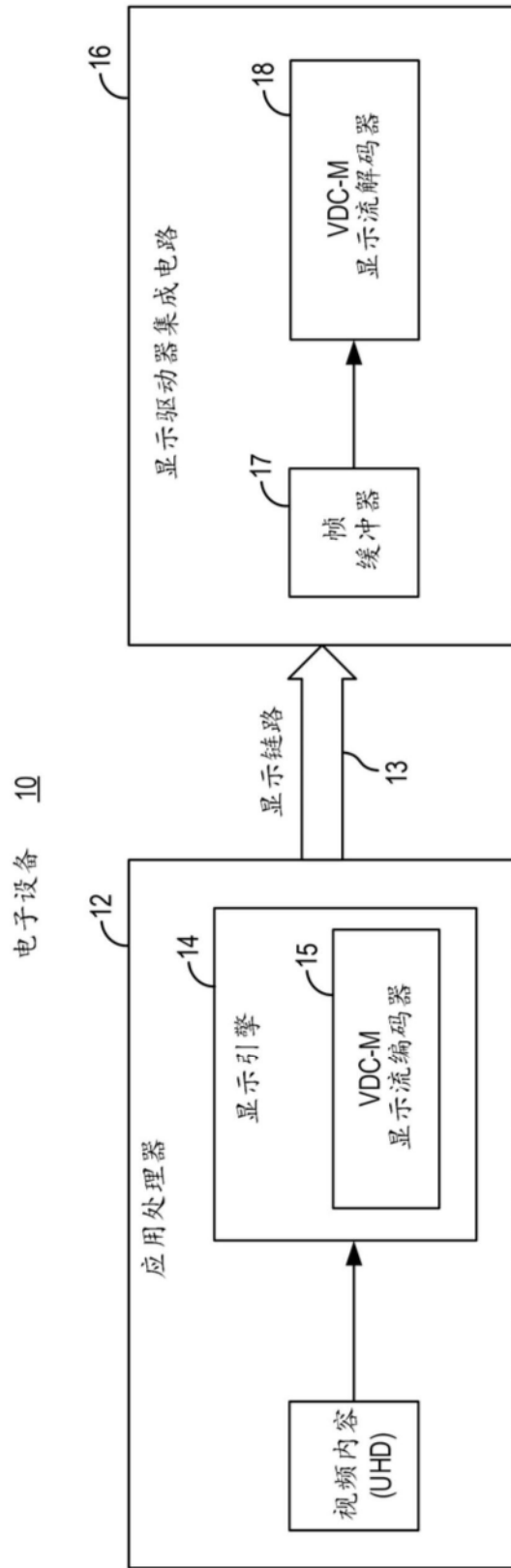


图1

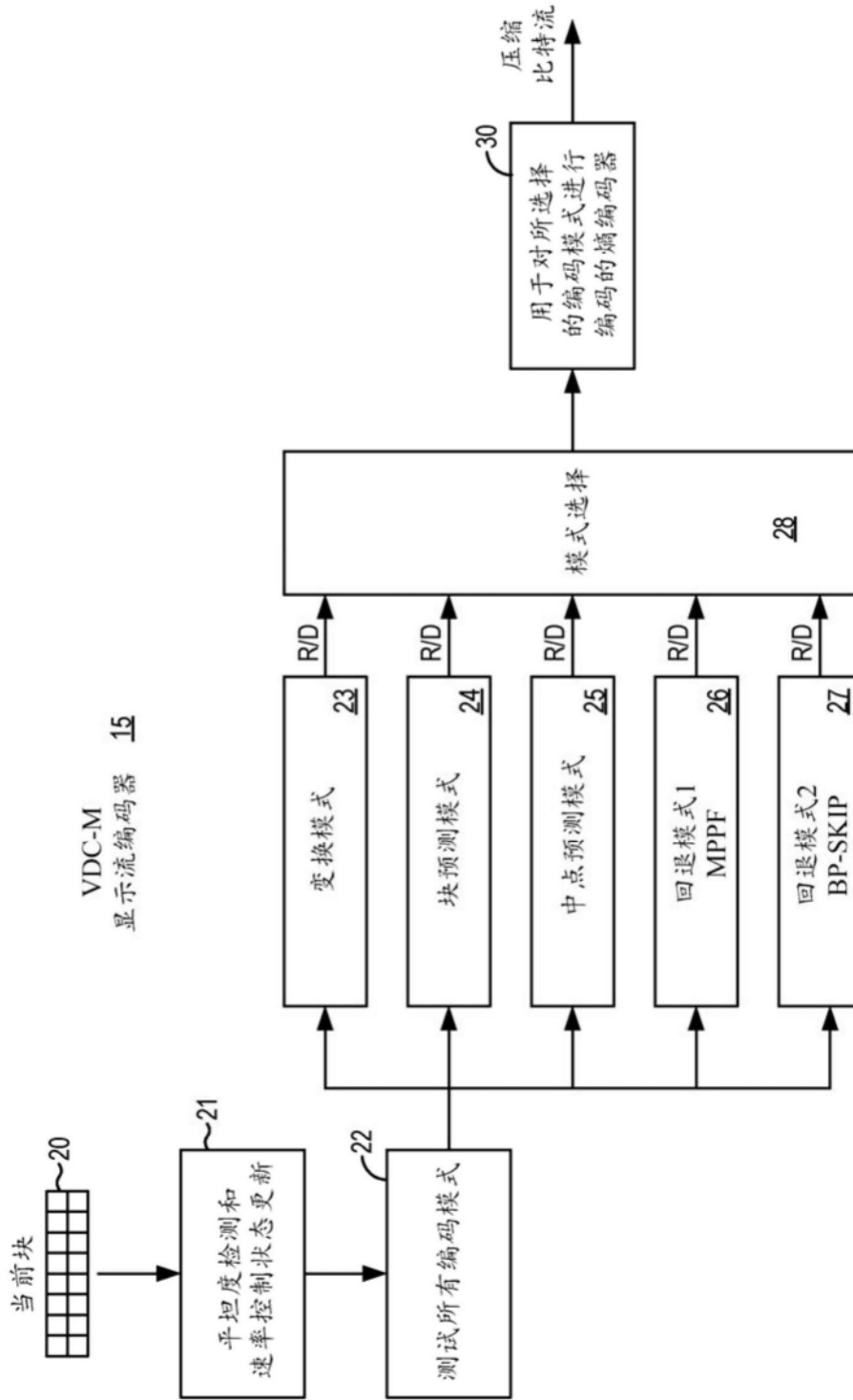


图2

变换模式 23

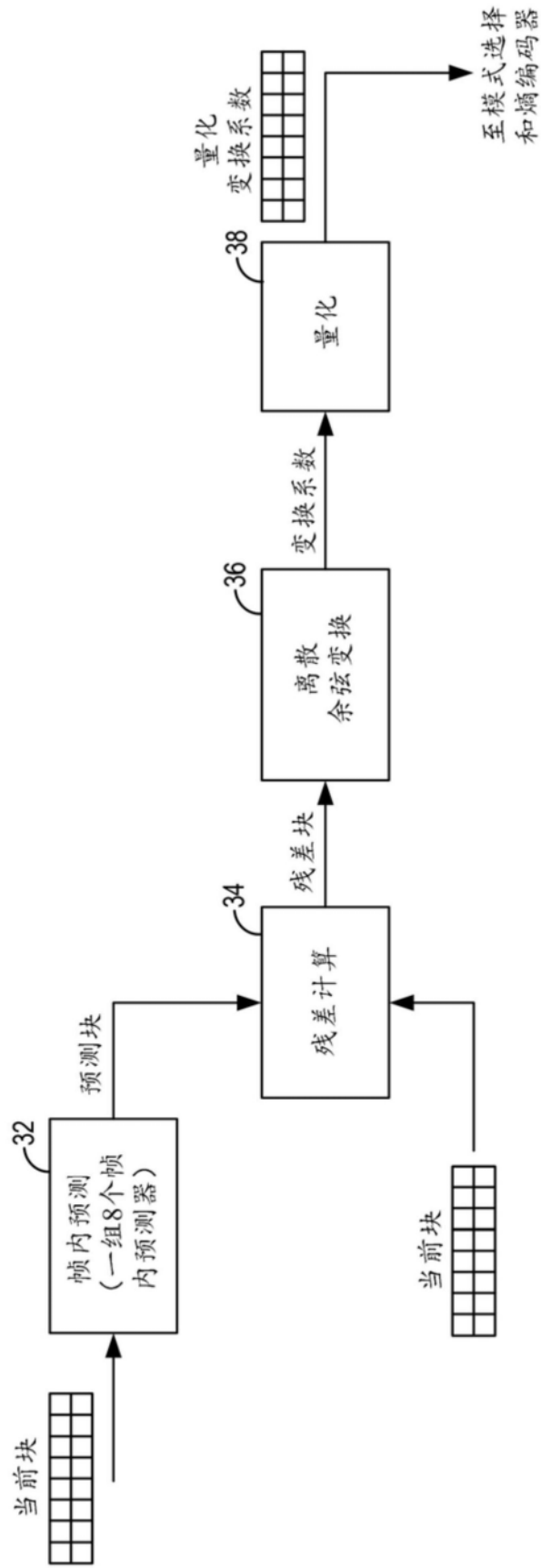


图3

8个帧内预测器

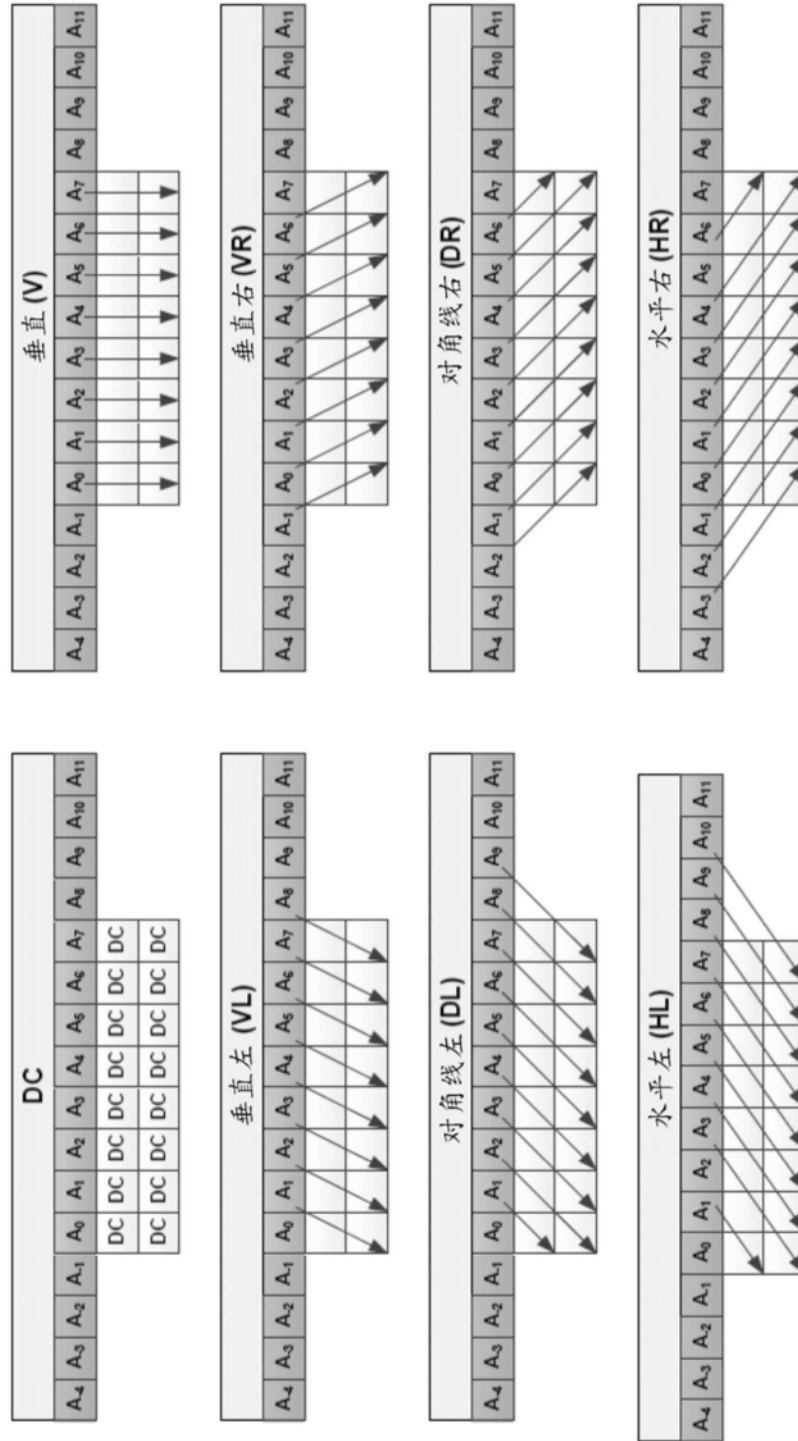


图4

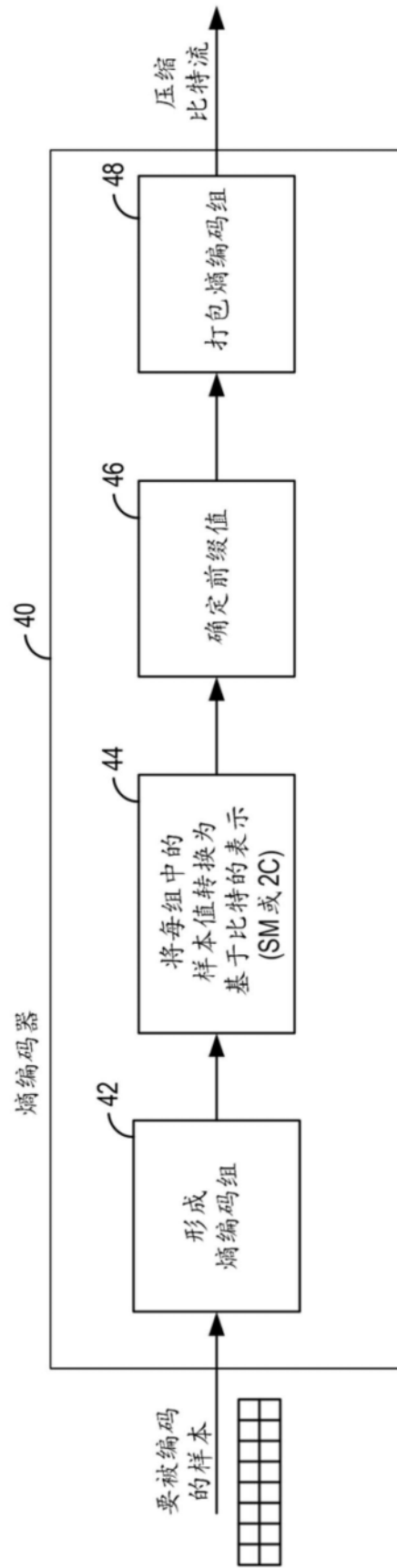


图5





样本块(4:4:4)



图 7(a)

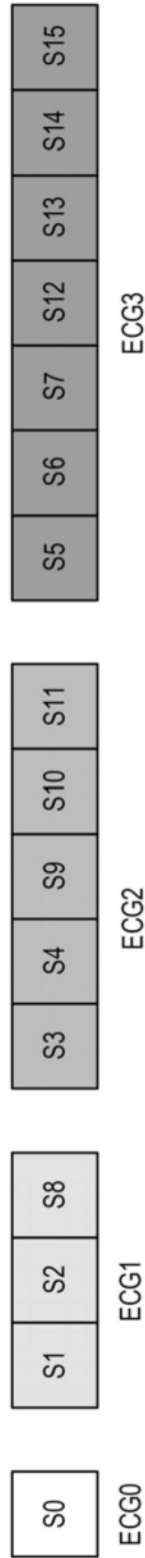


图 7(b)

图7

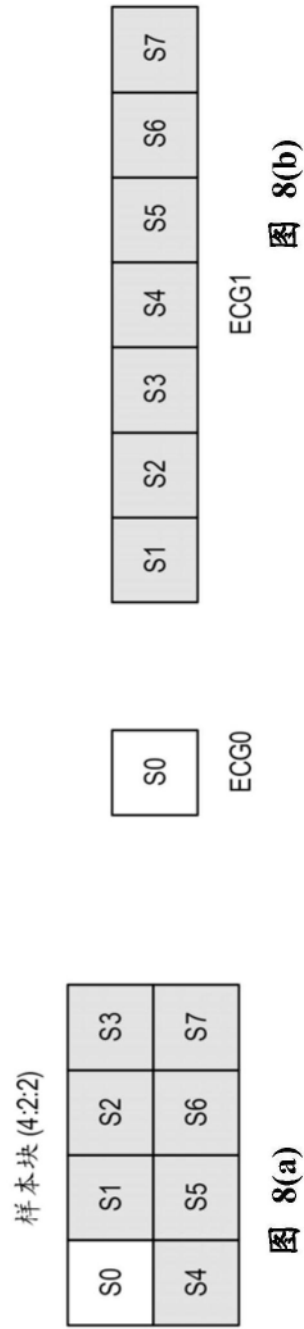


图8

样本块 (4:4:4)

S0	S1	S2	S3	S4	S5	S6	S7
S8	S9	S10	S11	S12	S13	S14	S15

图 9(a)

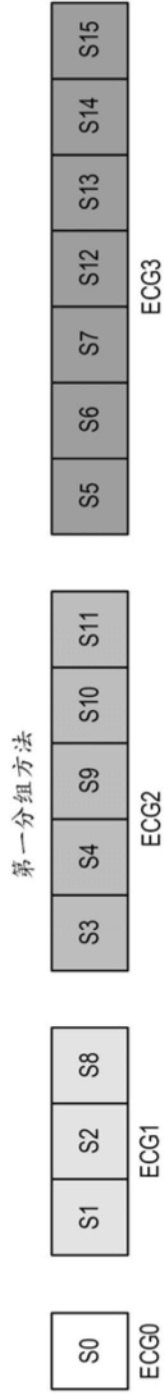


图 9(b)

第二分组方法

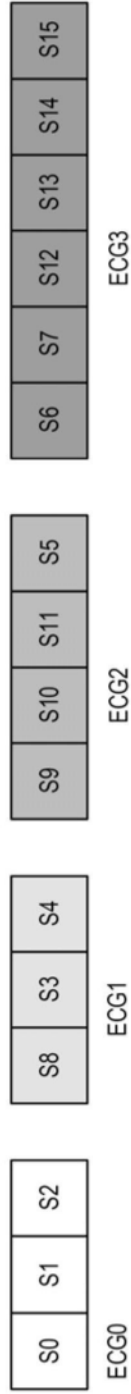


图 9(c)

第三分组方法

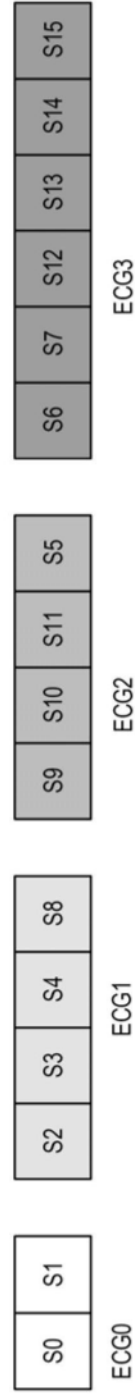


图 9(d)

图9

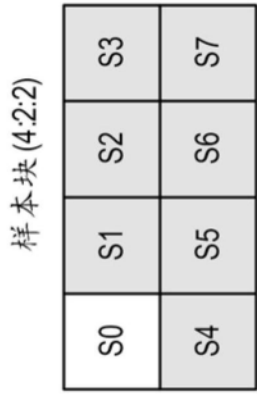


图 10(a)

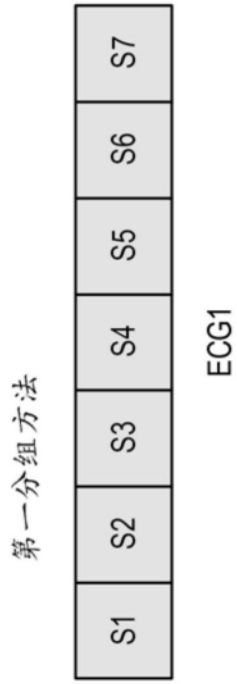


图 10(b)

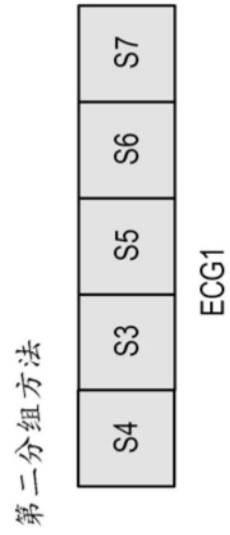


图 10(c)

图10

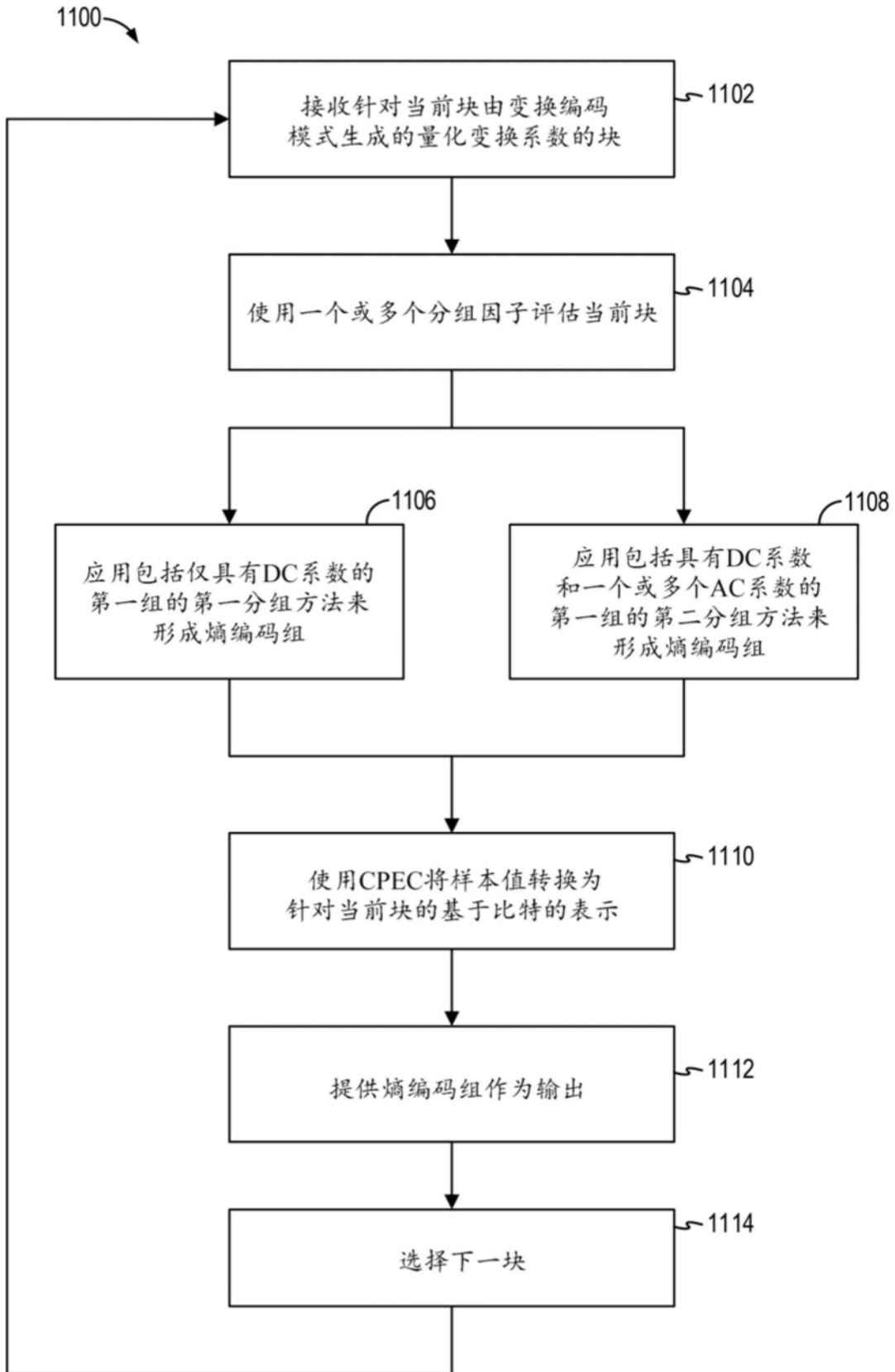


图11

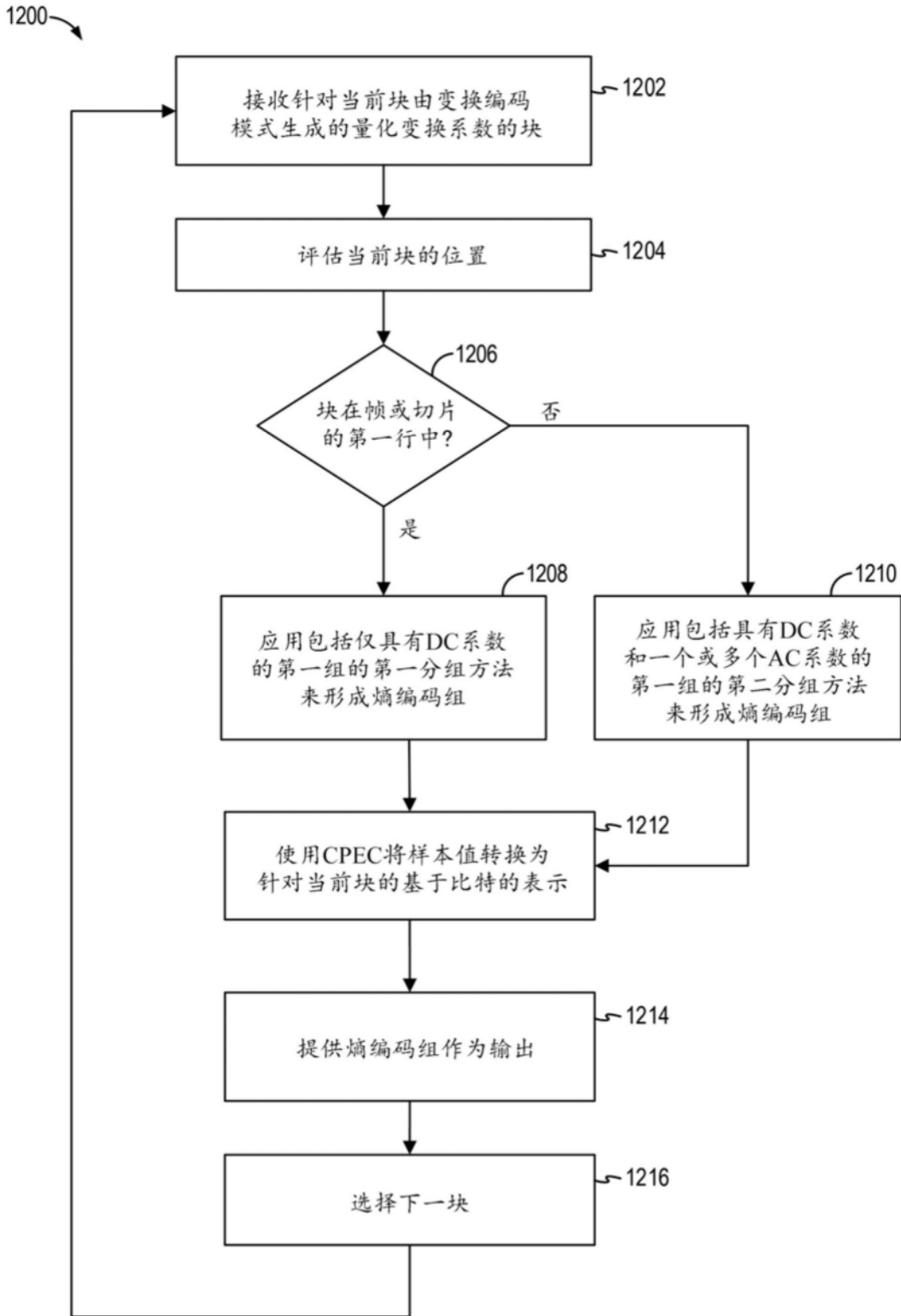


图12

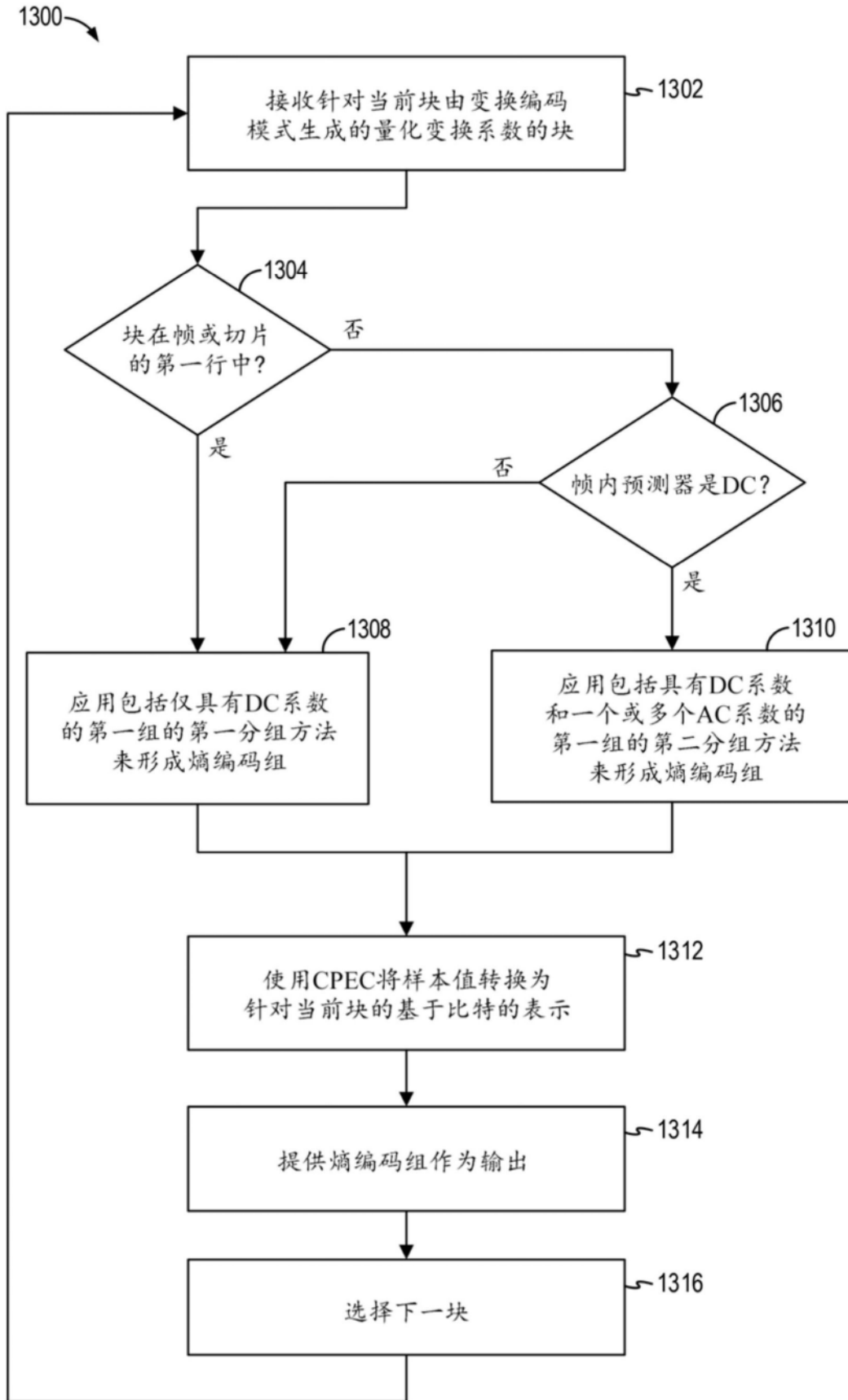


图13

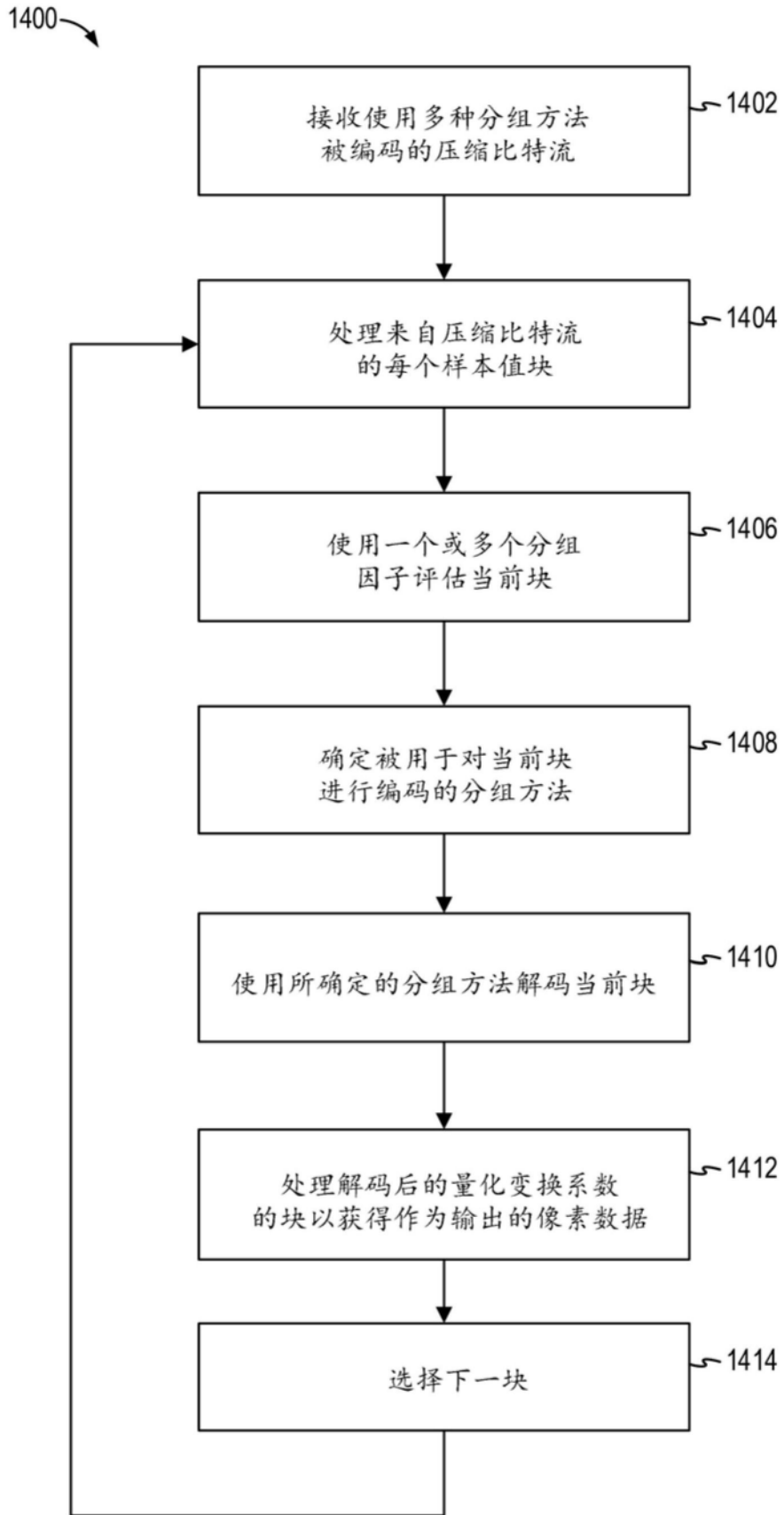


图14