(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 113545058 B
(45)授权公告日 2025.06.24

(21)申请号 202080018661.0
(22)申请日 2020.03.11
(65)同一申请的已公布的文献号
申请公布号 CN 113545058 A
(43)申请公布日 2021.10.22
(30)优先权数据
62/816,745 2019.03.11 US
62/850,453 2019.05.20 US
16/814,654 2020.03.10 US
(85)PCT国际申请进入国家阶段日
2021.09.03
(86)PCT国际申请的申请数据
PCT/US2020/022065 2020.03.11
(87)PCT国际申请的公布数据
WO2020/185875 EN 2020.09.17
(73)专利权人 高通股份有限公司
地址 美国加利福尼亚
(72)发明人 M·卡切维奇 M·Z·科班
王洪涛
(74)专利代理机构 永新专利商标代理有限公司 72002
专利代理师 戴开良
(51)Int.Cl.
H04N 19/13(2006.01)
H04N 19/157(2006.01)
H04N 19/18(2006.01)
H04N 19/182(2006.01)
H04N 19/70(2006.01)
H04N 19/91(2006.01)
(56)对比文件
CN 103999460 A,2014.08.20
审查员 林雪钰

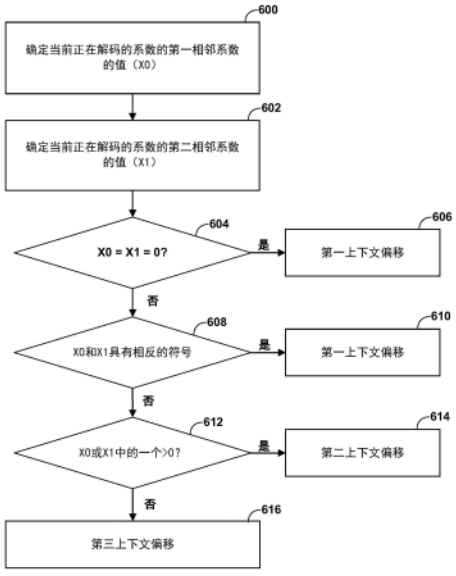
权利要求书6页 说明书30页 附图12页

(54)发明名称

用于变换跳过模式的系数译码

(57)摘要

一种用于对视频数据进行解码的设备,其针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;确定用于当前正在解码的系数的第二相邻系数的值;基于用于第一相邻系数的值和用于第二相邻系数的值,来确定当前正在解码的系数的上下文偏移;并基于所确定的上下文偏移,对当前正在解码的系数的值进行解码。



CN 113545058 B

1.一种对视频数据进行解码的方法,所述方法包括:

针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;

确定用于当前正在解码的所述系数的第二相邻系数的值;

基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的上下文偏移,其中,基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的所述上下文偏移包括:

响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移;

响应于所述第一相邻系数和所述第二相邻系数都为正,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移;以及

响应于所述第一相邻系数和所述第二相邻系数都为负,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为负,选择第三上下文偏移,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

基于所确定的上下文偏移,对当前正在解码的所述系数的符号进行解码。

2.根据权利要求1所述的方法,其中,所述第一相邻系数包括当前正在解码的所述系数的上相邻系数或左相邻系数中的一个,而所述第二相邻系数包括所述上相邻系数或所述左相邻系数中的另一个。

3.根据权利要求1所述的方法,其中,基于所确定的上下文偏移,对当前正在解码的所述系数的所述符号进行解码包括:

基于所确定的上下文偏移来确定上下文;

接收一个或多个数据箱;以及

基于所确定的上下文,对所述一个或多个数据箱进行上下文解码以确定当前正在解码的所述系数的所述符号。

4.根据权利要求3所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定用于当前正在解码的所述系数的预测绝对值;

接收语法元素;

响应于所述语法元素具有等于一的值,确定当前正在解码的所述系数的绝对值等于所述预测绝对值。

5.根据权利要求3所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定用于当前正在解码的所述系数的预测绝对值;

接收语法元素;

响应于所述语法元素的值大于所述预测绝对值,确定当前正在解码的所述系数的绝对值等于所述语法元素的所述值加一。

6.根据权利要求3所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定用于当前正在解码的所述系数的预测绝对值;

接收语法元素;

响应于所述语法元素的值小于所述预测绝对值,确定当前正在解码的所述系数的绝对值等于所述语法元素的所述值。

7.根据权利要求1所述的方法,还包括:

在不进行逆变换的情况下,对所述用于当前正在解码的所述系数的值进行逆量化,以确定视频数据的所述残差块的残差值。

8.根据权利要求1所述的方法,还包括:

基于所述用于当前正在解码的所述系数的值,确定解码的残差块;

将所述解码的残差块与预测块相加以确定重构块;

对所述重构块执行一个或多个滤波操作,以确定视频数据的解码块;以及

输出包括视频数据的所述解码块的视频数据的解码图片。

9.一种对视频数据进行编码的方法,所述方法包括:

针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;

确定用于当前正在编码的所述系数的第二相邻系数的值;

基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移,其中,基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的所述上下文偏移包括:

响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移;

响应于所述第一相邻系数和所述第二相邻系数都为正,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移;以及

响应于所述第一相邻系数和所述第二相邻系数都为负,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为负,选择第三上下文偏移,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

基于所确定的上下文偏移,对当前正在编码的所述系数的符号进行编码。

10.根据权利要求9所述的方法,其中,所述第一相邻系数包括上相邻系数或左相邻系数中的一个,而所述第二相邻系数包括所述上相邻系数或所述左相邻系数中的另一个。

11.根据权利要求9所述的方法,其中,基于所确定的上下文偏移,对当前正在编码的所述系数的所述符号进行编码包括:

基于所确定的上下文偏移来确定上下文;

确定当前正在编码的所述系数的所述符号;以及

基于所确定的上下文,对一个或多个数据箱进行上下文编码以表示当前正在编码的所述系数的所述符号。

12.根据权利要求11所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的预测绝对值;

响应于当前正在编码的所述系数的绝对值等于所述预测绝对值,将语法元素编码为具有等于一的值。

13.根据权利要求11所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的预测绝对值;

响应于当前正在编码的所述系数的绝对值小于所述预测绝对值,将语法元素编码为具有等于当前正在编码的所述系数的所述绝对值的值。

14.根据权利要求11所述的方法,还包括:

基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的预测绝对值;

响应于当前正在编码的所述系数的绝对值大于所述预测绝对值,将语法元素编码为具有等于当前正在编码的所述系数的所述绝对值减一的值。

15.根据权利要求9所述的方法,还包括:

对视频数据的所述残差块的残差值进行量化,以确定当前正在编码的所述系数的值。

16.根据权利要求9所述的方法,还包括:

确定预测块;以及

将所述预测块与视频数据的原始块进行比较,以确定视频数据的所述残差块。

17.一种用于对视频数据进行解码的设备,所述设备包括:

存储器,其配置为存储视频数据;以及

在电路中实现并且被配置为进行以下操作的一个或多个处理器:

针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;

确定用于当前正在解码的所述系数的第二相邻系数的值;

基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的上下文偏移,其中,为了基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的所述上下文偏移,所述一个或多个处理器被配置为:

响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移;

响应于所述第一相邻系数和所述第二相邻系数都为正,或者

所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移;以及

响应于所述第一相邻系数和所述第二相邻系数都为负,或者

所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为负,选择第三上下文偏移,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

基于所确定的上下文偏移,对当前正在解码的所述系数的符号进行解码。

18.根据权利要求17所述的设备,其中,所述第一相邻系数包括上相邻系数或左相邻系

数中的一个,而所述第二相邻系数包括所述上相邻系数或所述左相邻系数中的另一个。

19.根据权利要求17所述的设备,其中,为了基于所确定的上下文偏移,对当前正在解码的所述系数的所述符号进行解码,所述一个或多个处理器还被配置为:

基于所确定的上下文偏移来确定上下文;

接收一个或多个数据箱;以及

基于所确定的上下文,对所述一个或多个数据箱进行上下文解码以确定当前正在解码的所述系数的所述符号。

20.根据权利要求17所述的设备,其中,所述一个或多个处理器还被配置为:

对所述用于当前正在解码的所述系数的值进行逆量化,以确定视频数据的所述残差块的残差值。

21.根据权利要求17所述的设备,其中,所述一个或多个处理器还被配置为:

基于所述用于当前正在解码的所述系数的值,确定解码的残差块;

将所述解码的残差块与预测块相加以确定重构块;

对所述重构块执行一个或多个滤波操作,以确定视频数据的解码块;以及

输出包括视频数据的所述解码块的视频数据的解码图片。

22.根据权利要求17所述的设备,其中,所述设备包括无线通信设备,所述无线通信设备包括被配置为接收编码的视频数据的接收器。

23.根据权利要求22所述的设备,其中,所述无线通信设备包括电话手持装置,并且其中,所述接收器被配置为:根据无线通信标准来解调包括所述编码的视频数据的信号。

24.根据权利要求17所述的设备,还包括:

被配置为显示解码的视频数据的显示器。

25.根据权利要求17所述的设备,其中,所述设备包括照相机、计算机、移动设备、广播接收器设备或机顶盒中的一个或多个。

26.一种用于对视频数据进行编码的设备,所述设备包括:

存储器,其配置为存储所述视频数据;以及

在电路中实现并且被配置为进行以下操作的一个或多个处理器:

针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;

确定用于当前正在编码的所述系数的第二相邻系数的值;

基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移,其中,为了基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的所述上下文偏移,所述一个或多个处理器被配置为:

响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移;

响应于所述第一相邻系数和所述第二相邻系数都为正,或者

所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移;以及

响应于所述第一相邻系数和所述第二相邻系数都为负,或者

所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所

述第二相邻系数中的另一者为负,选择第三上下文偏移,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

基于所确定的上下文偏移,对当前正在编码的所述系数的符号进行编码。

27.根据权利要求26所述的设备,其中,所述第一相邻系数包括上相邻系数或左相邻系数中的一个,而所述第二相邻系数包括所述上相邻系数或所述左相邻系数中的另一个。

28.根据权利要求26所述的设备,其中,为了基于所确定的上下文偏移,对当前正在编码的所述系数的所述符号进行编码,所述一个或多个处理器还被配置为:

基于所确定的上下文偏移来确定上下文;

确定当前正在编码的所述系数的所述符号;以及

基于所确定的上下文,对一个或多个数据箱进行上下文编码以表示当前正在编码的所述系数的所述符号。

29.根据权利要求26所述的设备,其中,所述一个或多个处理器还被配置为:

对视频数据的所述残差块的残差值进行量化,以确定当前正在编码的所述系数的值。

30.根据权利要求26所述的设备,其中,所述一个或多个处理器还被配置为:

确定预测块;以及

将所述预测块与视频数据的原始块进行比较,以确定视频数据的所述残差块。

31.一种用于对视频数据进行解码的装置,所述装置包括:

用于针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值的单元;

用于确定用于当前正在解码的所述系数的第二相邻系数的值的单元;

用于基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的上下文偏移的单元,其中,用于基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的所述上下文偏移的单元包括:

用于响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移的单元;

用于响应于所述第一相邻系数和所述第二相邻系数都为正,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移的单元;以及

用于响应于所述第一相邻系数和所述第二相邻系数都为负,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为负,选择第三上下文偏移的单元,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

用于基于所确定的上下文偏移,对当前正在解码的所述系数的符号进行解码的单元。

32.一种存储指令的非暂时性计算机可读存储介质,当所述指令被一个或多个处理器执行时,使所述一个或多个处理器用于:

针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;

确定用于当前正在解码的所述系数的第二相邻系数的值;

基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所

述系数的上下文偏移,其中,基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的所述上下文偏移包括:

响应于所述第一相邻系数和所述第二相邻系数都等于零或者具有相反符号,选择第一上下文偏移;

响应于所述第一相邻系数和所述第二相邻系数都为正,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为正,选择第二上下文偏移;以及

响应于所述第一相邻系数和所述第二相邻系数都为负,或者所述第一相邻系数和所述第二相邻系数中的一者等于零并且所述第一相邻系数和所述第二相邻系数中的另一者为负,选择第三上下文偏移,其中,所述第一上下文偏移、所述第二上下文偏移、所述第三上下文偏移是不同的偏移值;以及

基于所确定的上下文偏移,对当前正在解码的所述系数的符号进行解码。

用于变换跳过模式的系数译码

[0001] 本申请要求享受以下申请的优先权：

[0002] 2020年3月10日提交的美国专利申请16/814,654；

[0003] 2019年3月11日提交的美国临时专利申请62/816,745;以及

[0004] 2019年5月20日提交的美国临时专利申请62/850,453，

[0005] 故以引用方式将每份申请的全部内容并入本文。

技术领域

[0006] 本公开内容涉及视频编码和视频解码。

背景技术

[0007] 数字视频能力可以并入到各种各样的设备中,这些设备包括数字电视、数字直接广播系统、无线广播系统、个人数字助理(PDA)、膝上型计算机或桌面型计算机、平板计算机、电子书读取器、数码相机、数字记录装置、数字媒体播放器、视频游戏设备、视频游戏控制台、蜂窝或卫星无线电话、所谓的“智能电话”、视频电话会议设备、视频流设备等等。数字视频设备实现视频译码技术,例如,由MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4第10部分、高级视频译码(AVC)、ITU-T H.265/高效率视频译码(HEVC)所规定的标准、以及这些标准的扩展里所描述的那些技术。视频设备可以通过实现这些视频译码技术,更高效地发送、接收、编码、解码和/或存储数字视频信息。

[0008] 视频译码技术包括空间(帧内)预测和/或时间(帧间)预测以减少或消除视频序列中固有的冗余。对于基于块的视频译码,可以将视频片段(例如,视频图片或视频图片的一部分)划分为视频块,视频块还可以称为译码树单元(CTU)、译码单元(CU)和/或译码节点。使用相对于同一图片中相邻块中的参考样本的空间预测,对图片的帧内译码(I)片段中的视频块进行编码。图片的帧间译码(P或B)片段中的视频块可以使用相对于同一图片中相邻块中的参考样本的空间预测,或者使用相对于其它参考图片中的参考样本的时间预测。图片可以称为帧,而参考图片可以称为参考帧。

发明内容

[0009] 在一些译码场景中,视频编码器可以在其中不执行变换过程(即,跳过变换过程)的变换跳过模式下,对视频数据进行编码。因此,对于以变换跳过模式编码的块,不对残差数据进行变换。本公开内容描述了用于变换跳过模式的系数译码方案的技术。本公开内容的技术包括熵解码过程,其将系数的二进制表示转换为一系列非二进制值的量化系数。相应的熵编码过程通常是熵解码的逆过程,其也是本公开内容的一部分。

[0010] 在一个例子中,一种对视频数据进行解码的方法包括:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;确定用于当前正在解码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值来确定用于当前正在解码的所述系数的上下文偏移;并基于所确定的上下文

偏移,对用于当前正在解码的所述系数的值进行解码。

[0011] 在另一个例子中,一种对视频数据进行编码的方法包括:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;确定用于当前正在编码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移;并基于所确定的上下文偏移,对当前正在编码的所述系数的值进行编码。

[0012] 在另一个例子中,一种用于对视频数据进行解码的设备包括:被配置为存储视频数据的存储器、以及在电路中实现并且被配置为进行以下操作的一个或多个处理器:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;确定用于当前正在解码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值来确定用于当前正在解码的所述系数的上下文偏移;并基于所确定的上下文偏移,对用于当前正在解码的所述系数的值进行解码。

[0013] 在另一个例子中,一种用于对视频数据进行编码的设备包括:被配置为存储视频数据的存储器、以及在电路中实现并且被配置为进行以下操作的一个或多个处理器:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;确定用于当前正在编码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移;并基于所确定的上下文偏移,对当前正在编码的所述系数的值进行编码。

[0014] 在另一个例子中,一种用于对视频数据进行解码的装置包括:用于针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值的单元;用于确定用于当前正在解码的所述系数的第二相邻系数的值的单元;用于基于所述第一相邻系数的值和所述第二相邻系数的值,来确定用于当前正在解码的所述系数的上下文偏移的单元;用于基于所确定的上下文偏移,对用于当前正在解码的所述系数的值进行解码的单元。

[0015] 在另一个例子中,一种用于对视频数据进行编码的装置包括:用于针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值的单元;用于确定用于当前正在编码的所述系数的第二相邻系数的值的单元;用于基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移的单元;用于基于所确定的上下文偏移,对当前正在编码的所述系数的值进行编码的单元。

[0016] 在另一个例子中,一种计算机可读存储介质存储有指令,当所述指令被一个或多个处理器执行时,使所述一个或多个处理器进行以下操作:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;确定用于当前正在解码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值来确定用于当前正在解码的所述系数的上下文偏移;并基于所确定的上下文偏移,对用于当前正在解码的所述系数的值进行解码。

[0017] 在另一个例子中,一种计算机可读存储介质存储有指令,当所述指令被一个或多个处理器执行时,使所述一个或多个处理器进行以下操作:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;确定用于当

前正在编码的所述系数的第二相邻系数的值;基于所述第一相邻系数的值和所述第二相邻系数的值,确定当前正在编码的所述系数的上下文偏移;并基于所确定的上下文偏移,对当前正在编码的所述系数的值进行编码。

[0018] 在附图和下面的说明书中阐述了一个或多个例子的细节。根据说明书、附图以及权利要求书,其它特征、对象和优点将变得显而易见。

附图说明

[0019] 图1是示出可以执行本公开内容的技术的示例性视频编码和解码系统的框图。

[0020] 图2A和图2B是示出示例性四叉树二叉树(QTBT)结构和相应的译码树单元(CTU)的概念图。

[0021] 图3示出了当前正被编码或解码的系数的相邻系数的例子。

[0022] 图4是示出可以执行本公开内容的技术的示例性视频编码器的框图。

[0023] 图5是示出可以执行本公开内容的技术的示例性视频解码器的框图。

[0024] 图6A和图6B是示出二进制算术译码中的范围更新过程的概念图。

[0025] 图7是示出二进制算术译码中的输出过程的概念图。

[0026] 图8是示出视频编码器中的上下文自适应二进制算术译码(CABAC)译码器的框图。

[0027] 图9是示出视频解码器中的CABAC译码器的框图。

[0028] 图10是示出用于对视频数据进行编码的过程的流程图。

[0029] 图11是示出用于对视频数据进行解码的过程的流程图。

[0030] 图12是示出用于确定对残差块的系数的符号进行译码的上下文的过程的流程图。

具体实施方式

[0031] 视频译码(例如,视频编码和/或视频解码)通常涉及根据同一图片中已译码的视频数据块来预测视频数据块(例如,帧内预测),或者根据不同图片中已译码的视频数据块来预测视频数据块(例如,帧间预测)。在某些情况下,视频编码器还通过将预测块与原始块进行比较来计算残差数据。因此,残差数据代表预测块和原始块之间的差异。为了减少用信号发送残差数据所需要的比特数,视频编码器可以对残差数据进行变换和量化,并在编码比特流中用信号发送经变换和量化的残差数据。通过这种变换和量化过程实现的压缩可能是有损的,这意味着变换和量化过程可能在解码的视频数据中引入失真。

[0032] 视频解码器对残差数据进行解码,并将残差数据添加到预测块以产生与单独预测块相比更接近地匹配原始视频块的重构视频块。由于残差数据的变换和量化所引入的损失,因此第一重构块可能具有失真或伪影。一种常见类型的伪影或失真称为块效应,其中用于对视频数据进行译码的块的边界是可见的。

[0033] 为了进一步提高解码视频的质量,视频解码器可以对重构的视频块执行一个或多个滤波操作。这些滤波操作的例子包括:去块滤波、样本自适应偏移(SAO)滤波和自适应环路滤波(ALF)。用于这些滤波操作的参数可以由视频编码器确定并在编码视频比特流中显式地用信号通知,或者可以由视频解码器隐式地确定而无需在编码视频比特流中显式地用信号通知这些参数。

[0034] 在一些译码场景中,视频编码器可以在其中不执行上述变换过程(即,跳过变换过

程)的变换跳过模式下,对视频数据进行编码。因此,对于以变换跳过模式编码的块,不对残差数据进行变换。视频数据的使用变换跳过模式进行编码的残差块,也可以称为未变换的残差块。本公开内容描述了用于变换跳过模式的系数译码方案的技术。本公开内容的技术包含将二进制表示转换为一系列非二进制值量化系数的熵解码过程。相应的熵编码过程(其通常是熵解码的逆过程)也是本公开内容的一部分。本公开内容的技术可以应用于任何现有的视频编解码器(例如,高效视频编码(HEVC))、或者当前正在开发的标准(例如,通用视频编码(VVC)、以及其它未来的视频编码标准。

[0035] 本公开内容提出的技术包括:例如,基于用于第一相邻系数的值和用于第二相邻系数的值,确定当前正在解码的系数的上下文偏移,并基于所确定的上下文偏移,对当前正在解码的系数的值进行解码。由于与对于经变换的块相比,残差块中的相邻系数之间的系数值对于变换跳过的块而言倾向于更相关,因此本公开内容的技术可以导致改进的熵编码,这可以通过例如减少表示编码视频数据所需的比特开销来提高整体编码效率,而不会降低解码的视频数据的质量。

[0036] 图1是示出可以执行本公开内容的技术的示例性视频编码和解码系统100的框图。本公开内容的技术通常针对于视频数据的译码(编码和/或解码)。通常,视频数据包括用于处理视频的任何数据。因此,视频数据可以包括原始的、未译码的视频、编码的视频、解码的(例如,重构的)视频以及视频元数据(例如,信令数据)。

[0037] 如图1中所示,在该例子中,系统100包括源设备102,其提供要由目的地设备116进行解码和显示的经编码的视频数据。具体而言,源设备102经由计算机可读介质110,将视频数据提供给目的地设备116。源设备102和目的地设备116可以是或者包括广泛的设备中的任何设备,其包括桌面型计算机、笔记本(例如,膝上型)计算机、平板计算机或其它移动设备、机顶盒、电话手持装置、智能电话、电视、照相机、显示设备、数字媒体播放器、视频游戏控制台、视频流设备、广播接收器设备等等。在一些情况下,源设备102和目的地设备116可以被配备用于无线通信,因此可以称为无线通信设备。

[0038] 在图1的例子中,源设备102包括视频源104、存储器106、视频编码器200和输出接口108。目的地设备116包括输入接口122、视频解码器300、存储器120和显示设备118。根据本公开内容,源设备102的视频编码器200和目的地设备116的视频解码器300可以被配置为应用本公开内容中所描述的用于高效译码的技术。因此,源设备102表示视频编码设备的例子,而目的地设备116表示视频解码设备的例子。在其它例子中,源设备和目的地设备可以包括其它组件或布置。例如,源设备102可以从诸如外部照相机之类的外部视频源接收视频数据。同样,目的地设备116可以与外部显示设备对接,而不是包括集成显示设备。

[0039] 如图1中所示的系统100仅仅是一个例子。通常,任何数字视频编码和/或解码设备都可以执行本公开内容中所描述的用于高效译码的技术。源设备102和目的地设备116仅仅是这样的译码设备的例子,其中,源设备102生成用于向目的地设备116传输的经编码的视频数据。本公开内容将“译码”设备称为执行数据的译码(编码和/或解码)的设备。因此,视频编码器200和视频解码器300分别表示译码设备(具体而言,视频编码器和视频解码器)的例子。在一些例子中,设备102、116可以以基本上对称的方式操作,使得设备102、116中的每一个包括视频编码和解码组件。因此,系统100可以支持视频设备102、116之间的单向或双向视频传输,例如,用于视频流传输、视频回放、视频广播或视频电话。

[0040] 通常,视频源104代表视频数据(即,原始的、未编码的视频数据)的源,并将视频数据的图片的连续序列(也称为“帧”)提供给视频编码器200,其中视频编码器200对图片的数据进行编码。源设备102的视频源104可以包括视频捕获设备(例如,摄像机)、包含先前捕获的原始视频的视频档案和/或用于从视频内容提供商接收视频的视频馈送接口。作为另一替代方案,视频源104可以生成基于计算机图形的数据作为源视频、或者实时视频、存档视频和计算机生成的视频的组合。在每种情况下,视频编码器200对捕获的、预捕获的或计算机生成的视频数据进行编码。视频编码器200可以将图像从接收到的顺序(有时称为“显示顺序”)重新排列为用于译码的译码顺序。视频编码器200可以生成包括有编码的视频数据的比特流。然后,源设备102可以经由输出接口108将编码的视频数据输出到计算机可读介质110上,以通过例如目的地设备116的输入接口122进行接收和/或检索。

[0041] 源设备102的存储器106和目的地设备116的存储器120代表通用存储器。在一些例子中,存储器106、120可以存储原始视频数据,例如,来自视频源104的原始视频和来自视频解码器300的原始的经解码的视频数据。另外地或替代地,存储器106、120可以存储可由例如视频编码器200和视频解码器300分别执行的软件指令。尽管在该例子中,将存储器106和存储器120与视频编码器200和视频解码器300分开地示出,但应当理解,视频编码器200和视频解码器300还可以包括内部存储器,以实现功能上相似或等效的目的。此外,存储器106、120可以存储编码的视频数据(例如,从视频编码器200输出并输入到视频解码器300)。在一些例子中,可以将存储器106、120的一部分分配为一个或多个视频缓冲器,例如,用于存储原始的、解码和/或编码的视频数据。

[0042] 计算机可读介质110可以代表能够将编码的视频数据从源设备102传输到目的地设备116的任何类型的介质或设备。在一个例子中,计算机可读介质110代表用于使源设备102能够实时地例如经由射频网络或基于计算机的网络将编码的视频数据直接发送到目的地设备116的通信介质。输出接口108可以根据无线通信标准,对包括编码的视频数据的传输信号进行解调,并且输入接口122可以根据诸如无线通信协议之类的通信标准,对接收到的传输信号进行解调。通信介质可以包括任何无线或有线通信介质,例如射频(RF)频谱或一条或多条物理传输线。通信介质可以形成诸如局域网、广域网或全球网络(例如,互联网)的基于分组的网络的一部分。通信介质可以包括路由器、交换机、基站或者有助于从源设备102到目的地设备116的通信的任何其它设备。

[0043] 在一些例子中,计算机可读介质110可以包括存储设备112。源设备102可以将经编码的数据从输出接口108输出到存储设备112。类似地,目的地设备116可以经由输入接口122,从存储设备112访问经编码的数据。存储设备112可以包括多种分布式或本地访问的数据存储介质(例如,硬盘驱动器、蓝光光盘、DVD、CD-ROM、闪存、易失性或非易失性存储器)或者用于存储编码的视频数据的任何其它适当的数字存储介质中的任何一种。

[0044] 在一些例子中,计算机可读介质110可以包括文件服务器114或者能够存储由源设备102所生成的经编码视频数据的另一种中间存储设备。源设备102可以将编码的视频数据输出到文件服务器114或者可以存储源设备102所生成的编码的视频数据的另一个中间存储设备。目的地设备116可以经由流传输或下载,从文件服务器114访问所存储的视频数据。文件服务器114可以是能够存储编码的视频数据并将该编码的视频数据发送到目的地设备116的任何类型的服务器设备。文件服务器114可以代表网络服务器(例如,用于网站)、文件

传输协议(FTP)服务器、内容传送网络设备、或网络附加存储(NAS)设备。目的地设备116可以通过包括互连网连接的任何标准数据连接,从文件服务器114访问编码的视频数据。这可以包括适合于访问已存储在文件服务器114上的编码视频数据的无线信道(例如,Wi-Fi连接)、有线连接(例如,数字用户线(DSL)、电缆调制解调器等等)或二者的组合。文件服务器114和输入接口122可以被配置为根据流传输协议、下载传输协议或者其组合进行操作。

[0045] 输出接口108和输入接口122可以代表无线发射器/接收器、调制解调器、有线网络组件(例如,以太网卡)、根据各种IEEE 802.11标准中的任何一种进行操作的无线通信组件、或者其它物理组件。在其中输出接口108和输入接口122包括无线组件的例子中,输出接口108和输入接口122可以被配置为根据诸如4G、4G-LTE(长期演进)、高级LTE、5G等等之类的蜂窝通信标准,来传输诸如编码的视频数据之类的数据。在输出接口108包括无线发射器的一些例子中,输出接口108和输入接口122可以被配置为根据其它无线标准(例如,IEEE 802.11规范、IEEE 802.15规范(例如,ZigBee™)、Bluetooth™标准等等),来传输诸如编码的视频数据之类的数据。在一些例子中,源设备102和/或目的地设备116可以包括各自的片上系统(SoC)设备。例如,源设备102可以包括SoC设备以执行归属于视频编码器200和/或输出接口108的功能,而目的地设备116可以包括SoC设备以执行归属于视频解码器300和/或输入接口122的功能。

[0046] 本公开内容的技术可以应用于支持各种多媒体应用中的任何一种的视频编码,这些多媒体应用诸如为空中电视广播、有线电视传输、卫星电视传输、互联网流视频传输(如,基于HTTP的动态自适应流媒体(DASH)、编码到数据存储介质上的数字视频),对存储在数据存储介质上的数字视频的解码、或者其它应用。

[0047] 目的地设备116的输入接口122从计算机可读介质110(例如,通信介质、存储设备112、文件服务器114等等)接收编码的视频比特流。编码的视频比特流可以包括由视频编码器200定义的而且由视频解码器300使用的信令信息,例如,具有描述视频块或其它译码单元(例如,片段、图片、图片组、序列等等)的特性和处理的值的语法元素。显示设备118向用户显示经解码的视频数据的解码图片。显示设备118可以代表多种显示设备中的任何一种,例如阴极射线管(CRT)、液晶显示器(LCD)、等离子显示器、有机发光二极管(OLED)显示器、或者另一种类型的显示设备。

[0048] 虽然没有在图1中示出,但在一些例子中,视频编码器200和视频解码器300可以分别与音频编码器和/或音频解码器集成在一起,并且可以包括适当的MUX-DEMUX单元或者其它硬件和/或软件,以处理包括共同数据流中的音频和视频两者的复用流。如果适用的话,MUX-DEMUX单元可以符合ITU H.223多路复用器协议或其它协议(例如,用户数据报协议(UDP))。

[0049] 视频编码器200和视频解码器300各自可以实现为多种适当的编码器和/或解码器电路中的任何一种,例如一个或多个微处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)、离散逻辑、软件、硬件、固件或者其任意组合。当部分地以软件实现这些技术时,设备可以将用于软件的指令存储在适当的非临时性计算机可读介质中,并使用一个或多个处理器以硬件方式执行这些指令以执行本公开内容的技术。视频编码器200和视频解码器300中的每一个可以被包括在一个或多个编码器或解码器中,编码器或解码器中的任何一个可以集成为相应设备中的组合编码器/解码器(CODEC)的一部分。

包括视频编码器200和/或视频解码器300的设备可以包括集成电路、微处理器和/或无线通信设备(例如,蜂窝电话)。

[0050] 视频编码器200和视频解码器300可以根据诸如ITU-T H.265之类的视频编码标准(也称为高效视频编码(HEVC))或者其扩展(例如,多视图和/或可扩展的视频编码扩展)进行操作。替代地,视频编码器200和视频解码器300可以根据诸如联合勘探试验模型(JEM)或ITU-T H.266(也称为通用视频编码(VVC))之类的其它专有或工业标准来操作。在以下文献中描述了VVC标准的草案:Bross,et al.“Versatile Video Coding(Draft 4),”ITU-T SG 16 WP 3的联合视频专家组(JVET)和ISO/IEC JTC 1/SC 29/WG 11的第13次会议:2019年1月9日至18日于马里兰州马拉喀什的JVET-M1001-v5(以下简称“VVC草案4”)。在以下文献中描述了VVC标准的另一个草案:Bross,et al.“Versatile Video Coding(Draft 7),”ITU-T SG 16 WP 3的联合视频专家组(JVET)和ISO/IEC JTC 1/SC 29/WG 11的第16次会议:2019年10月1日至11日于瑞士日内瓦的JVET-P2001-v14(以下简称“VVC草案7”)。然而,本公开内容的技术并不限于任何特定的编码标准。

[0051] 通常,视频编码器200和视频解码器300可以执行图片的基于块的编码。术语“块”通常是指包括要处理的数据的结构(例如,在编码和/或解码过程中使用的编码、解码或其它方式)。例如,一个块可以包括亮度和/或色度数据的样本的二维矩阵。通常,视频编码器200和视频解码器300可以对以YUV(例如,Y、Cb、Cr)格式表示的视频数据进行编码。也就是说,不是对图片的样本的红色、绿色和蓝色(RGB)数据进行译码,而是视频编码器200和视频解码器300可以对亮度和色度分量进行译码,其中色度分量可以包括红色和蓝色色度分量。在一些例子中,视频编码器200在编码之前,将接收到的RGB格式的数据转换成YUV表示,而视频解码器300将YUV表示转换成RGB格式。替代地,预处理和后处理单元(没有示出)可以执行这些转换。

[0052] 本公开内容通常可以涉及图片的译码(例如,编码和解码),以包括对图片的数据进行编码或解码的过程。类似地,本公开内容可以涉及对图片的块的译码,以包括对这些块的数据进行编码或解码的过程(例如,预测和/或残余编码)。编码的视频比特流通常包括用于语法元素的一系列值,其中这些语法元素表示译码决策(例如,译码模式)以及将图片划分成块。因此,对图片或块进行译码的提及,通常应当被理解为用于形成该图片或块的语法元素的译码值。

[0053] HEVC定义了各种块,其包括译码单元(CU)、预测单元(PU)和变换单元(TU)。根据HEVC,视频译码器(例如,视频编码器200)根据四叉树结构,将译码树单元(CTU)划分为CU。也就是说,视频译码器将CTU和CU划分为四个相等的、不重叠的正方形,并且四叉树的每个节点具有零个或四个子节点。没有子节点的节点可以称为“叶节点”,并且这种叶节点的CU可以包括一个或多个PU和/或一个或多个TU。视频译码器可以进一步分割PU和TU。例如,在HEVC中,残余四叉树(RQT)表示TU的划分。在HEVC中,PU表示帧间预测数据,而TU表示残差数据。帧内预测的CU包括帧内预测信息(例如,帧内模式指示)。

[0054] 再举一个例子,视频编码器200和视频解码器300可以被配置为根据JEM或VVC进行操作。根据JEM或VVC,视频译码器(例如,视频编码器200)将图片划分为多个译码树单元(CTU)。视频编码器200可以根据诸如四叉树-二叉树(QTBT)结构或多类型树(MTT)结构之类的树结构,对CTU进行划分。QTBT结构消除了多个分区类型的概念,例如HEVC的CU、PU和TU之

间的分隔。QTBT结构包括两个层级:根据四叉树划分而划分的第一层级、根据二叉树划分而划分的第二层级。QTBT结构的根节点对应于CTU。二叉树的叶节点对应于译码单元(CU)。

[0055] 在MTT划分结构中,可以使用四叉树(QT)分割、二叉树(BT)分割和一种或多种类型的三叉树(TT)(也称为三叉型树(TT))分割,对块进行划分。三叉树或三叉型树分割是将一个块分割成三个子块的划分。在一些例子中,三叉树或三叉型树分割将一个块划分为三个子块,而不穿过中心来划分原始块。MTT中的划分类型(例如,QT、BT和TT)可以是对称的,也可以是不对称的。

[0056] 在一些例子中,视频编码器200和视频解码器300可以使用单个QTBT或MTT结构来表示亮度和色度分量中的每一个,而在其它例子中,视频编码器200和视频解码器300可以使用两个或更多QTBT或MTT结构,例如一个QTBT/MTT结构用于亮度分量,另一个QTBT/MTT结构用于两个色度分量(或者两个QTBT/MTT结构用于相应的色度分量)。

[0057] 视频编码器200和视频解码器300可以被配置为使用每个HEVC的四叉树划分、QTBT划分、MTT划分或其它划分结构。为了便于说明起见,相对于QTBT划分给出了本公开内容的技术的描述。但是,应当理解的是,本公开内容的技术还可以应用于被配置为使用四叉树划分或其它类型的划分的视频译码器。

[0058] 可以以各种方式,在图片中对块(例如,CTU或CU)进行分组。举一个例子,砖块(brick)可以指代图片中特定瓦片(tile)内的某个矩形区域的CTU行。瓦片可以是图片中的特定瓦片列和特定瓦片行内的CTU的矩形区域。瓦片列是指高度等于图片的高度、并且具有(例如,诸如在图片参数集中)由语法元素指定的宽度的矩形区域的CTU。瓦片行是指具有由语法元素指定的高度(例如,诸如在图片参数集中给出的)、并且宽度等于图片的宽度的矩形区域的CTU。

[0059] 在一些例子中,可以将一个瓦片(tile)划分成多个砖块,每个砖块可以包括该瓦片内的一个或多个CTU行。没有被划分为多个砖块的瓦片,也可以称为砖块。但是,作为瓦片的一个真实子集的砖块不能称为瓦片。

[0060] 还可以将图片中的砖块排列在片段中。片段可以是图片中能够专门地被包含在单个网络抽象层(NAL)单元中的整数数量砖块。在一些例子中,一个片段包括多个完整瓦片,或者仅包括一个瓦片的连续序列的砖块。

[0061] 本公开内容可以以垂直和水平维度,互换地使用“NxN”和“N乘N”来指代块(例如,CU或其它视频块)的样本尺寸,例如16x16样本或16乘16样本。通常,一个16x16 CU在垂直方向上将有16个样本(y=16),在水平方向上将有16个样本(x=16)。同样地,一个NxN CU通常在垂直方向上具有N个样本,在水平方向上具有N个样本,其中N表示非负整数值。可以按行和列来排列CU中的样本。此外,CU在水平方向上不必具有与垂直方向上相同数量的样本。举例来说,CU可以包含NxM个样本,其中M不一定等于N。

[0062] 视频编码器200对用于表示预测和/或残差信息以及其它信息的CU的视频数据进行编码。预测信息指示将如何预测CU,以便形成用于该CU的预测块。残差信息通常表示在编码之前的CU的样本与预测块之间的逐样本差异。

[0063] 为了预测CU,视频编码器200通常可以通过帧间预测或帧内预测来形成用于CU的预测块。帧间预测通常指代从先前编码的图片的数据中预测CU,而帧内预测通常指代从同一图片的先前编码的数据中预测CU。为了执行帧间预测,视频编码器200可以使用一个或多

个运动矢量来产生预测块。视频编码器200通常可以例如在CU与参考块之间的差异方面,执行运动搜索以识别与CU紧密匹配的参考块。视频编码器200可以使用绝对差之和(SAD)、平方差之和(SSD)、平均绝对差(MAD)、均方差(MSD)或其它此类差值计算来计算差值度量,以确定参考块是否紧密匹配当前CU。在一些例子中,视频编码器200可以使用单向预测或双向预测来预测当前CU。

[0064] JEM和VVC的一些例子还提供仿射运动补偿模式,其可以被认为是帧间预测模式。在仿射运动补偿模式中,视频编码器200可以确定表示非平移运动的两个或更多运动矢量,例如缩小或放大、旋转、透视运动或其它不规则运动类型。

[0065] 为了执行帧内预测,视频编码器200可以选择帧内预测模式来生成预测块。JEM和VVC的某些例子提供了67种帧内预测模式,其包括各种定向模式以及平面模式和DC模式。通常,视频编码器200选择帧内预测模式,该帧内预测模式描述了相对于当前块(例如,CU的块)的相邻样本,从相邻样本中预测当前块的样本。假设视频编码器200以光栅扫描顺序(从左到右、从上到下)对CTU和CU进行译码,则这些样本通常可以在与当前块相同的图片中的当前块的上方、左上方或左侧。

[0066] 视频编码器200对表示当前块的预测模式的数据进行编码。举例来说,对于帧间预测模式,视频编码器200可以对表示使用各种可用帧间预测模式中的哪一个模式的数据、以及对应模式的运动信息进行编码。对于单向或双向帧间预测,例如,视频编码器200可以使用高级运动矢量预测(AMVP)或合并模式,来对运动矢量进行编码。视频编码器200可以使用类似模式来对仿射运动补偿模式的运动矢量进行编码。

[0067] 在诸如块的帧内预测或帧间预测之类的预测之后,视频编码器200可以计算该块的残差数据。残差数据(例如,残差块)表示该块与使用相应预测模式形成的针对该块的预测块之间的逐样本差异。视频编码器200可以向残差块应用一个或多个变换,以在变换域而非样本域中产生经变换的数据。举例来说,视频编码器200可以向残差视频数据应用离散余弦变换(DCT)、整数变换、小波变换或概念上类似的变换。另外,视频编码器200可以在第一变换之后应用次级变换,例如依赖于模式的不可分离次级变换(MDNSST)、依赖于信号的变换、Karhunen-Loeve变换(KLT)等等。视频编码器200在应用所述一个或多个变换之后,产生变换系数。

[0068] 尽管以上描述了执行变换的例子,但是在一些例子中,可以跳过变换。例如,视频编码器200可以实现其中跳过变换操作的变换跳过模式。在跳过变换的例子中,视频编码器200可以输出与残差值相对应的系数而不是变换系数。在以下描述中,术语“系数”应当被解释为包括与残差值相对应的系数或者根据变换结果所生成的变换系数。

[0069] 如上所述,在进行任何变换或者跳过变换以产生系数之后,视频编码器200可以执行系数的量化。在一些例子中,当跳过变换时,也可以跳过量化。量化通常指代这样的过程:对系数进行量化,以可能地减少用于表示这些系数的数据量,从而提供进一步压缩。通过执行量化过程,视频编码器200可以减小与一些或所有这些系数相关联的比特深度。例如,视频编码器200可以在量化期间,将n比特值舍入为m比特值,其中n大于m。在一些例子中,为了执行量化,视频编码器200可以执行待量化的值的按比特右移。

[0070] 在量化之后,视频编码器200可以扫描系数(例如,根据变换的结果或者由于变换跳过而生成的系数),从而根据包括量化的系数的二维矩阵产生一维矢量。可以将该扫描设

计为将较高能量(因此频率较低)的系数放在矢量的前面,并将较低能量(因此频率较高)的系数放在矢量的后面。在跳过变换的例子中,扫描的结果可能不是较高的能量系数在矢量的前面和较低的能量系数在矢量的后面。在一些例子中,视频编码器200可以利用预定义的扫描顺序来扫描经量化的系数,以产生经串行化的矢量,然后对矢量的经量化的系数进行熵编码。在其它例子中,视频编码器200可以执行自适应扫描。在扫描量化的系数以形成一维矢量之后,视频编码器200可以例如根据上下文自适应二进制算术编码(CABAC)对一维矢量进行熵编码。视频编码器200还可以对语法元素的值进行熵编码,该语法元素描述与编码的视频数据相关联的元数据,以供视频解码器300在对视频数据进行解码时使用。

[0071] 如上所述,视频编码器200将对TU中的残差数据进行编码。根据TU中的残差数据的预期特性,视频编码器200可以以不同模式(例如,变换模式或变换跳过模式)对TU进行编码,其中不同的模式利用不同的系数译码方案。一些系数译码方案利用系数组来对TU进行编码。系数组通常是TU中的系数的一个子集。举例来说,视频编码器200可以将16x16 TU编码为四个4x4系数组。

[0072] 为了执行CABAC,视频编码器200可以将上下文模型内的上下文分配给要发送的符号。例如,该上下文可以涉及符号的相邻值是否为零值。概率确定可以基于分配给符号的上下文。

[0073] 视频编码器200还可以在例如图片头、块报头、片段头或其它语法数据(例如,序列参数集(SPS)、图片参数集(PPS)或视频参数集(VPS))中,生成针对视频解码器300的语法数据(例如,基于块的语法数据、基于图片的语法数据、以及基于序列的语法数据)。视频解码器300可以类似地对此类语法数据进行解码,以确定如何解码对应的视频数据。

[0074] 用此方式,视频编码器200可以生成包括编码的视频数据的比特流,例如,用于描述将图片划分成块(例如,CU)的语法元素、以及这些块的预测和/或残差信息的语法元素。最终,视频解码器300可以接收比特流,并对编码的视频数据进行解码。

[0075] 通常,视频解码器300执行与视频编码器200所执行的过程互易的过程,以解码比特流的编码的视频数据。举例来说,视频解码器300可以以与视频编码器200的CABAC编码过程实质上相似的方式(尽管互易),使用CABAC,对比特流的语法元素的值进行解码。语法元素可以定义用于将图片分割为CTU、以及根据对应的分割结构(诸如QTBT结构)对每个CTU进行分割的分割信息,以定义CTU的CU。这些语法元素可以进一步规定用于视频数据的块(例如,CU)的预测和残差信息。

[0076] 可以通过例如量化的系数来表示残差信息,其中该量化的系数表示残差值或者变换系数。视频解码器300可以对块的量化的变换系数进行逆量化和逆变换,以再现该块的残差块。在视频编码器200跳过变换操作的例子中(例如,变换跳过模式),视频编码器300可以跳过逆变换操作。视频解码器300使用用信号通知的预测模式(帧内或帧间预测)和相关的预测信息(例如,用于帧间预测的运动信息)来形成用于该块的预测块(即,预测块)。然后,视频解码器300可以组合预测块和残差块(在逐个样本的基础上)以再现原始块。视频解码器300可以执行其它的处理,例如执行解块处理以减少沿块的边界的视觉伪像。

[0077] 根据本公开内容的技术,视频编码器200和视频解码器300可以被配置为:基于用于第一相邻系数的值和用于第二相邻系数的值,确定当前正在解码的系数的上下文偏移,并基于所确定的上下文偏移,对当前正在解码的系数的值进行解码。由于与对于变换块相

比,残差块中的相邻系数之间的系数值对于变换跳过块更相关,因此本公开内容的技术可以导致改进的熵编码,这可以通过例如减少表示编码视频数据所需的比特开销来提高整体编码效率,而不会降低解码的视频数据的质量。

[0078] 本公开内容通常涉及“用信号通知”某些信息(例如,语法元素)。术语“用信号通知”通常可以指代传送用于语法元素的值和/或用于对编码的视频数据进行解码的其它数据。也就是说,视频编码器200可以在比特流中用信号通知用于语法元素的值。通常,用信号通知是指在比特流中生成值。如上所述,源设备102可以基本实时地或者不实时地,将比特流传输到目的地设备116,例如在将语法元素存储到存储设备112以便稍后由目的地设备116检索时,可能发生这种情形。

[0079] 图2A和图2B是示出示例性四叉树二叉树(QTBT)结构130以及对应的译码树单元(CTU)132的概念图。实线表示四叉树分割,而虚线表示二叉树分割。在二叉树的每个分割(即,非叶子)节点中,用信号通知一个标志以指示使用哪种分割类型(即,水平或垂直),在该例子中,0表示水平分割,而1表示垂直分割。对于四叉树分割,由于四叉树节点将一个块水平和垂直地分割为大小相等的4个子块,因此无需指示分割类型。因此,视频编码器200可以对QTBT结构130的区域树层级(即,第一层级)(即,实线)的语法元素(例如,分割信息)和QTBT结构130的预测树层级(即,第二层级)(即,虚线)的语法元素(例如,分割信息)进行编码,并且视频解码器300可以对其进行解码。视频编码器200可以对QTBT结构130的终端叶节点表示的CU的视频数据(例如,预测和变换数据)进行编码,并且视频解码器300可以对其进行解码。

[0080] 一般而言,图2B的CTU 132可以与规定块的大小的参数相关联,块的大小与第一层级和第二层级的QTBT结构130的节点相对应。这些参数可以包括CTU大小(其以样本来代表CTU 132的大小)、最小四叉树大小(MinQTSize,其代表最小允许的四叉树叶节点大小)、最大二叉树大小(MaxBTSize,其代表最大允许二叉树根节点大小)、最大二叉树深度(MaxBTDepth,其代表最大允许二叉树深度)和最小二叉树大小(MinBTSize,其代表最小允许二叉树叶子节点大小)。

[0081] 对应于CTU的QTBT结构的根节点可以在QTBT结构的第一层级具有四个子节点,可以根据四叉树划分来划分每个子节点。即,第一层级的节点是叶节点(没有子节点)或者具有四个子节点。QTBT结构130的例子表示这样的节点,例如包括具有用于分支的实线的父节点和子节点。如果第一层级的节点不大于最大允许的二叉树根节点大小(MaxBTSize),则可以通过相应的二叉树进一步对这些节点进行划分。可以迭代一个节点的二叉树分割,直到分割产生的节点达到最小允许的二叉树叶节点大小(MinBTSize)或最大允许的二叉树深度(MaxBTDepth)为止。QTBT结构130的例子将这样的节点表示为具有用于分支的虚线。二进制树叶节点称为译码单元(CU),其用于预测(例如,帧内或帧间预测)和变换,而无需任何进一步的划分。如上所述,CU也可以称为“视频块”或“块”。

[0082] 在QTBT划分结构的一个例子中,将CTU大小设置为128x128(亮度样本和两个相应的64x64色度样本),将MinQTSize设置为16x16,将MaxBTSize设置为64x64,将MinBTSize(对于宽度和高度二者)设置为4,将MaxBTDepth设置为4。首先,对CTU应用四叉树划分以生成四叉树叶节点。四叉树叶节点的大小可以从16x16(即,MinQTSize)到128x128(即,CTU大小)。如果四叉树叶节点为128x128,则由于该大小超过MaxBTSize(即,在该例子中为64x64),因

此不会通过二叉树进一步分割四叉树叶节点。否则,将通过二叉树,对四叉树叶节点进一步划分。因此,四叉树叶节点也是二叉树的根节点,并且二叉树的深度为0。当二叉树的深度达到MaxBTDepth(在该例子中为4)时,不允许进一步分割。二叉树节点的宽度等于MinBTSize(在该例子中为4),表示不允许进一步的水平分割。类似地,高度等于MinBTSize的二叉树节点意味着该二叉树节点不允许进一步的垂直分割。如上所述,将二叉树的叶节点称为CU,并且根据预测和变换对其进行进一步处理而无需进一步划分。

[0083] 当以变换跳过模式对视频数据的块进行编码时,视频编码器200在执行量化过程之前,跳过针对残差信号的变换过程。视频解码器300同样在执行去量化过程之后,跳过逆变换处理步骤。未变换残差信号的特性通常与变换信号的特性完全不同。例如,当与用于变换块的系数相比时,变换跳过块的系数往往与其相邻系数更相关。结果,当与残差数据的变换块的水平值和符号信息相比时,残差数据的变换跳过块中的相邻系数的水平值和符号信息往往更相关。

[0084] 以下文献提出了对以变换跳过模式编码的块执行残差编码的建议过程:B.Bross、T.Nguyen、P.Keydel、H.Schwarz、D.Marpe、T.Wiegand,“Non-CE8:Unified Transform Type Signaling and Residual Coding for Transform Skip”,JVET文档JVET-M0464,Marrackech,MA,2019年1月。为了在变换跳过模式中对层级和符号信息进行高效编码,可以修改JVET-M0464中提出的系数编码以利用信号特性进行更高效的编码。

[0085] 图3显示了来自残差数据的变换跳过块的三个系数的例子。变换跳过块还将包括图3中没有示出的其它系数。在图3的例子中,系数值X代表系数140的值,其表示当前正在编码的系数。系数值X0代表系数142的值,其是系数140的左相邻系数。系数值X1代表系数144的值,其是系数140的上相邻系数。在本公开内容中,上相邻也可以是称为上方相邻。

[0086] 视频编码器200和视频解码器300可以被配置为对变换跳过块中的系数执行符号译码。在JVET-M0464中描述的用于变换跳过残差译码的技术中,将系数从第一个(即,左上角)到最后一个(即,右下角)进行译码,而不是像对变换块那样从最后一个到第一个进行译码。如果实施JVET-M0464的技术,则视频编码器200和视频解码器300可以被配置为使用CABAC,对使用残差的信道类型作为上下文的符号信息进行上下文译码。也就是说,视频编码器200和视频解码器300可以将一个上下文用于亮度残差,并将另一个上下文用于色度残差。本公开内容描述了可以代替JVET-M0464中描述的技术或者除了JVET-M0464中描述的技术之外使用的符号译码技术。以下技术利用上相邻系数(例如,图3中的系数144)和左相邻系数(例如,图3中的系数142)的符号信息,来导出当前正在译码的系数(例如,图3中的系数140)的符号译码上下文偏移。

[0087] 参考图3,X0为左相邻系数值,X1为上相邻系数值。如果这两个相邻系数都为零或者均非零但具有相反符号,则视频编码器200和视频解码器300可以使用上下文偏移0(ctxOffset=0)。否则,如果这两个相邻系数都是非负的,则视频编码器200和视频解码器300可以使用上下文偏移1(ctxOffset=1)。对于所有其它情况,视频编码器200和视频解码器300可以使用上下文偏移2(ctxOffset=2)。这可以通过以下伪代码来描述:

```
if ((X0 == 0 && X1 == 0) || ((X0 * X1) < 0))
{
    ctxOffset = 0;
}
else if (X0 >= 0 && X1 >= 0)
{
    ctxOffset = 1;
}
else
{
    ctxOffset = 2;
}
```

[0088]

[0089]

[0090] 在一些例子中,如果两个相邻系数都为零或都为非零但具有相反的符号,则视频编码器200和视频解码器300可以使用上下文偏移0。否则(当这两个相邻系数都是正的,或这两个相邻系数都是负的,或者一个是0,另一个是非零),视频编码器200和视频解码器300可以使用上下文1。如果使用上下文1,则根据约定,发信号通知的0或1的符号值将意味着当前正在译码的系数的符号与非零邻居之一的符号相同。也可以将这种技术扩展到以下的译码场景:其中,只有先前译码的非零系数的值可以用于上下文推导,其中符号值为0或1表示系数的符号与先前译码的具有单一上下文的非零系数符号的符号相同。

[0091] 用于亮度和色度分量的分开的上下文集合可以与上面所描述的上下文偏移推导结合使用。

[0092] 视频编码器200和视频解码器300还可以被配置为执行水平映射。在JVET-M0464的变换跳过残差译码中,使用sig_coeff_flag、abs_level_gtX_flags、par_level_flag和abs_remainder值,来对系数绝对水平absCoeffLevel进行译码以形成最终的绝对变换系数值,其中X可以是1,..,5(或者某个其它临界值C)。因此,可以通过以下方式构造absCoeffLevel值:absCoeffLevel=1+abs_level_gt1_flag+par_level_flag+2*(abs_level_gt2_flag+abs_level_gt3_flag+…+abs_level_gtC_flag)+2*abs_remainder

[0093] 代替如在JVET-M0464中直接表示absCoeffLevel,视频编码器200可以被配置为将absCoeffLevel映射到修改的水平。视频解码器300可以被配置为执行逆映射。

[0094] 类似于上面所描述的符号译码上下文偏移推导技术,视频编码器200和视频解码器300可以使用左相邻系数和上相邻系数的absCoeffLevel信息,对绝对系数水平值进行编码和解码。在该例子中,使X_0表示当前正在译码的系数(例如,图3中的系数140)的左相邻系数(例如,图3中的系数142)的绝对系数水平,并且使X_1表示当前正在译码的系数(例如,图3中的系数140)的上相邻系数(例如,图3中的系数144)的绝对系数水平。为了表示具有绝对系数水平absCoeff的系数,可以对映射的absCoeffMod进行译码。

[0095] 视频编码器200用于导出absCoeffMod的值的操作,可以用以下伪代码来示出:

```
pred = max(X0, X1);
if (absCoeff == pred
{
    absCoeffMod = 1;
}
else
{
    absCoeffMod = (absCoeff < pred) ? absCoeff + 1 : absCoeff;
}
```

[0097] 在一些例子中,如果要译码的系数的绝对值(absCoeff)等于最大相邻预测值pred,则视频编码器200将修改的水平absCoeffMod设置为1。否则,如果absCoeff小于预测值,则视频编码器200将要译码的值增加1。否则,视频编码器200不修改absCoeff值。

[0098] 例如,视频编码器200可以基于第一相邻系数的值和第二相邻系数的值,确定当前正在编码的系数的预测水平值,并且响应于当前正在编码的系数的水平值等于预测水平值,将语法元素编码为具有等于一的值。在其它情况下,视频编码器200可以基于第一相邻系数的值和第二相邻系数的值,确定当前正在编码的系数的预测水平值,并且响应于当前正在编码的系数的水平值小于预测水平值,将语法元素编码为具有等于当前正在编码的系数的水平值的值。在其它实例中,视频编码器200可以基于第一相邻系数的值和第二相邻系数的值,确定当前正在编码的系数的预测水平值,并且响应于当前正在编码的系数的水平值大于预测水平值,将语法元素编码为具有等于当前正在编码的系数的水平值减一的值。

[0099] 视频解码器300用于导出absCoeff的值的操作,可以用以下伪代码来示出:

```
pred = max(X0, X1);
if (absCoeffMod == 1 && pred > 0)
{
    absCoeff = pred;
}
else
{
    absCoeff = absCoeffMod – (absCoeffMod <= pred);
}
```

[0101] 例如,视频解码器300可以基于第一相邻系数的值和第二相邻系数的值,来确定当前正在解码的系数的预测水平值,接收语法元素指示,并且响应于语法元素具有等于一的值,确定当前正在解码的系数的水平值等于预测水平值。在其它实例中,视频解码器300可

以基于第一相邻系数的值和第二相邻系数的值,来确定当前正在解码的系数的预测水平值,接收语法元素指示,并且响应于语法元素的值大于预测水平值,确定当前正在解码的系数的水平值等于语法元素的值加一。在其它实例中,视频解码器300可以基于第一相邻系数的值和第二相邻系数的值,来确定当前正在解码的系数的预测水平值,接收语法元素指示,并且响应于语法元素的值小于预测水平值,确定当前正在解码的系数的水平值等于语法元素的值。

[0102] 在一些例子中,视频编码器200和视频解码器300基于左相邻系数值和上相邻系数值是否为零,来确定或导出本文中标识为abs_level_gt1_flag的语法元素的上下文。语法元素abs_level_gt1_flag是用于对系数值进行译码的语法元素。abs_level_gt1_flag的值等于1,可以例如表示系数的绝对水平大于1。abs_level_gt1_flag的值等于0,可以例如表示系数的绝对水平不大于1。

[0103] 在一个例子中,所述上下文可以是多个(例如,三个)不同的上下文之一。对于左相邻系数和上相邻系数都具有非零值的情况,可以导出一种上下文。而对于只有左相邻系数或上相邻系数中的一个具有非零值的情况,可以导出另一种上下文。对于左相邻系数和上相邻系数两者都具有零值的情况,可以导出第三上下文。在一个例子中,这种上下文推导仅应用于非BDPCM(块差分脉冲编码调制)模式。

[0104] 在一些例子中,对于具有不存在或不可用的邻居值的译码场景(例如,当正在译码的系数在块的左边界上并且左邻居不存在时),视频编码器200和视频解码器300可以被配置为在导出上下文时使用零值。

[0105] 在一些例子中,可以如下所示地描述上下文推导:

```
ctxOffset = 0;
if (Exist(left_neighbor) && non-zero(left_neighbor))
{
   ctxOffset += 1;
}
```

[0106]

```
else if (Exist(above_neighbor) && non-zero(above_neighbor))
{
   ctxOffset += 1;
}
```

[0107] 在这样的例子中,对于不存在的/不可用的邻居值(例如,块的左边界上的值的左邻居),视频编码器200和视频解码器300可以被配置为在推导上下文时,将零值用于不可用的值。

[0108] 图4是示出可以执行本公开内容的技术的示例性视频编码器200的框图。提供图4以便于解释的目的,故其不应被认为是对本公开内容中广泛例示和描述的技术的限制。为了便于说明起见,本公开内容在诸如HEVC(H.265)视频编码标准和开发中的VVC(H.266)之类的视频编码标准上下文中,描述视频编码器200。但是,本公开内容的技术并不限于这些视频编码标准,并且一般性地适用于视频编码和解码。

[0109] 在图4的例子中,视频编码器200包括视频数据存储器230、模式选择单元202、残差生成单元204、变换处理单元206、量化单元208、逆量化单元210、逆变换处理单元212、重构单元214、滤波单元216、解码图片缓冲器(DPB)218和熵编码单元220。视频数据存储器230、

模式选择单元202、残差生成单元204、变换处理单元206、量化单元208、逆量化单元210、逆变换处理单元212、重构单元214、滤波单元216、DPB 218和熵编码单元220中的任何一个或全部,可以在一个或多个处理器中或者在处理电路中实现。此外,视频编码器200可以包括另外的或替代的处理器或处理电路,以执行这些功能和其它功能。

[0110] 视频数据存储器230可以存储将由视频编码器200的组件编码的视频数据。视频编码器200可以从例如视频源104(图1)接收存储在视频数据存储器230中的视频数据。DPB 218可以充当参考图片存储器,该参考图片存储器存储参考视频数据,以供视频编码器200预测后续视频数据时使用。视频数据存储器230和DPB 218可以由多种存储设备(例如,动态随机存取存储器(DRAM)(其包括同步DRAM(SDRAM))、磁阻RAM(MRAM)、电阻性RAM(RRAM)或其它类型的存储设备)中的任何一个形成。视频数据存储器230和DPB 218可以由相同的存储设备或不同的存储设备提供。在各个例子中,视频数据存储器230可以与视频编码器200的其它组件一起在芯片上,如图所示,或者相对于那些组件在芯片外。

[0111] 在本公开内容中,对视频数据存储器230的引用不应被解释为限于视频编码器200内部的存储器(除非如此具体地描述),也不应被解释为限于视频编码器200之外的存储器(除非如此具体地描述)。而是,对视频数据存储器230的引用应当被理解为存储视频数据的参考存储器,其中视频编码器200接收该视频数据(例如,要编码的当前块的视频数据)以进行编码。图1的存储器106还可以提供对来自视频编码器200的各个单元的输出的临时存储。

[0112] 示出了图4的各个单元以帮助理解由视频编码器200执行的操作。这些单元可以实现为固定功能电路、可编程电路或者其组合。固定功能电路指代提供特定功能、并在可以执行的操作上预先设置的电路。可编程电路指代可以被编程以执行各种任务,并且在可以执行的操作中提供灵活功能的电路。例如,可编程电路可以执行使可编程电路以软件或固件的指令所定义的方式进行操作的软件或固件。固定功能电路可以执行软件指令(例如,用于接收参数或输出参数),但是固定功能电路执行的操作的类型通常是不可变的。在一些例子中,这些单元中的一个或多个单元可以是不同的电路块(固定功能或可编程),并且在一些例子中,所述一个或多个单元可以是集成电路。

[0113] 视频编码器200可以包括由可编程电路形成的算术逻辑单元(ALU)、基本功能单元(EFU)、数字电路、模拟电路和/或可编程核。在使用由可编程电路执行的软件来执行视频编码器200的操作的例子中,存储器106(图1)可以存储视频编码器200接收并执行的软件的目标代码,或者视频编码器200中的另一个存储器(没有示出)可以存储这样的指令。

[0114] 视频数据存储器230被配置为存储接收到的视频数据。视频编码器200可以从视频数据存储器230检索视频数据的图片,并将视频数据提供给残差生成单元204和模式选择单元202。视频数据存储器230中的视频数据可以是将进行编码的原始视频数据。

[0115] 模式选择单元202包括运动估计单元222、运动补偿单元224和帧内预测单元226。模式选择单元202可以包括其它功能单元,以根据其它预测模式来执行视频预测。举例而言,模式选择单元202可以包括调色板单元、块内复制单元(其可以是运动估计单元222和/或运动补偿单元224的一部分)、仿射单元、线性模型(LM)单元等等。

[0116] 模式选择单元202通常协调多个编码通道,以测试编码参数的组合以及针对此类组合的最终率失真值。编码参数可以包括:CTU到CU的划分、用于CU的预测模式、用于CU的残差数据的变换类型、用于CU的残差数据的量化参数等等。模式选择单元202可以最终选择具

有比其它测试的组合更好的速率失真值的编码参数的组合。

[0117] 视频编码器200可以将从视频数据存储器230检索到的图片划分为一系列CTU,并将一个或多个CTU封装在片段中。模式选择单元202可以根据树结构(例如,上面所描述的HEVC的QTBT结构或四叉树结构)来划分图片的CTU。如上所述,视频编码器200可以根据树结构,通过划分CTU来形成一个或多个CU。这样的CU通常也可以称为“视频块”或“块”。

[0118] 通常,模式选择单元202还控制其组件(例如,运动估计单元222、运动补偿单元224和帧内预测单元226)以生成针对当前块(例如,当前CU、或者在HEVC中,PU和TU的重叠部分)的预测块。对于当前块的帧间预测,运动估计单元222可以执行运动搜索以识别一个或多个参考图片(例如,存储在DPB 218中的一个或多个先前编码的图片)中的一个或多个紧密匹配的参考块。具体而言,运动估计单元222可以例如根据绝对差之和(SAD)、平方差之和(SSD)、平均绝对差(MAD)、均方差(MSD)等等,来计算表示潜在参考块与当前块有多么相似的值。运动估计单元222通常可以使用当前块与正考虑的参考块之间的逐样本差异来执行这些计算。运动估计单元222可以识别具有由这些计算所产生的最小值的参考块,该参考块指示与当前块最紧密匹配的参考块。

[0119] 运动估计单元222可以形成一个或多个运动矢量(MV),这些MV定义参考图片中的参考块相对于当前图片中的当前块的位置。然后,运动估计单元222可以将运动矢量提供给运动补偿单元224。例如,对于单向帧间预测,运动估计单元222可以提供单个运动矢量,而对于双向帧间预测,运动估计单元222可以提供两个运动矢量。然后,运动补偿单元224可以使用运动矢量来生成预测块。例如,运动补偿单元224可以使用运动矢量来检索参考块的数据。再举一个例子,如果运动矢量具有分数采样精度,则运动补偿单元224可以根据一个或多个插值滤波器,对用于预测块的值进行插值。此外,对于双向帧间预测,运动补偿单元224可以例如通过逐样本平均或加权平均,来检索由相应的运动矢量识别的两个参考块的数据,并对检索到的数据进行组合。

[0120] 再举一个例子,对于帧内预测或帧内预测译码,帧内预测单元226可以根据与当前块相邻的样本来生成预测块。例如,对于定向模式,帧内预测单元226通常可以在数学上组合相邻样本的值,并在当前块上沿定义的方向填充这些计算的值以产生预测块。再举一个例子,对于DC模式,帧内预测单元226可以计算与当前块的相邻样本的平均值,并且生成预测块以包括针对预测块的每个样本的该所得平均值。

[0121] 模式选择单元202将预测块提供给残差生成单元204。残差生成单元204从视频数据存储器230接收当前块的原始未编码版本,并从模式选择单元202接收预测块。残差生成单元204计算当前块和预测块之间的逐样本差异。所得的逐样本差异定义了当前块的残差块。在一些例子中,残差生成单元204还可以确定残差块中的样本值之间的差,以使用残差差分脉冲编码调制(RDPCM)来生成残差块。在一些例子中,可以使用执行二进制减法的一个或多个减法器电路,来形成残差生成单元204。

[0122] 在模式选择单元202将CU划分为PU的例子中,每一个PU可以与亮度预测单元和对应的色度预测单元相关联。视频编码器200和视频解码器300可以支持具有各种大小的PU。如上所述,CU的大小可以指代CU的亮度译码块的大小,而PU的大小可以指代PU的亮度预测单元的大小。假设特定CU的大小为2Nx2N,则视频编码器200可以支持2Nx2N或NxN的PU大小来用于帧内预测,并支持2Nx2N、2NxN、Nx2N、NxN等等的对称PU大小来用于帧间预测。视频编

码器200和视频解码器300还可以对于帧间预测,支持PU大小为2NxnU、2NxnD、nLx2N和nRx2N的非对称划分。

[0123] 在模式选择单元不将CU进一步划分成PU的例子中,每个CU可以与亮度译码块和对应的色度译码块相关联。如上所述,CU的大小可以指代CU的亮度译码块的大小。视频编码器200和视频解码器300可以支持2Nx2N、2NxN或Nx2N的CU大小。

[0124] 对于其它视频译码技术(例如,块内复制模式编码、仿射模式编码和线性模型(LM)模式编码,仅举几个例子),模式选择单元202经由与译码技术相关联的各个单元,针对正在编码的当前块来生成预测块。在一些例子中(例如,调色板模式编码),模式选择单元202可以不生成预测块,而是生成语法元素,这些语法元素指示基于所选的调色板来重构块的方式。在这样的模式下,模式选择单元202可以将这些语法元素提供给熵编码单元220以进行编码。

[0125] 如上所述,残差生成单元204接收当前块和对应的预测块的视频数据。然后,残差生成单元204生成当前块的残差块。为了生成残差块,残差生成单元204计算预测块和当前块之间的逐样本差。

[0126] 变换处理单元206将一个或多个变换应用于残差块以生成变换系数的块(本文称为“变换系数块”)。变换处理单元206可以将各种变换应用于残差块以形成变换系数块。例如,变换处理单元206可以将离散余弦变换(DCT)、方向变换、Karhunen-Loeve变换(KLT)或者概念上类似的变换应用于残差块。在一些例子中,变换处理单元206可以对残差块执行多个变换(例如,主变换和次要变换(如,旋转变换))。在一些例子中,变换处理单元206不向残差块应用变换。在这样的实例中,变换处理单元206可以输出系数块,其中这些系数对应于残差值而非变换系数。

[0127] 量化单元208可以对变换系数块中的变换系数进行量化,以产生经量化的变换系数块。对于以变换跳过模式译码的块,量化单元208可以对系数块中的系数进行量化以产生经量化的系数块。量化单元208可以根据与当前块相关联的量化参数(QP)值,来量化系数或者变换系数。视频编码器200(例如,经由模式选择单元202)可以通过调整与CU相关联的QP值,来调整应用的量化程度。量化可能导致信息的丢失,因此,量化后的系数或变换系数的精度可能比变换处理单元206生成的原始系数或变换系数的精度低。

[0128] 逆量化单元210和逆变换处理单元212可以将逆量化和逆变换分别应用于量化的系数块,以根据系数块来重建残差块。重构单元214可以基于重构的残差块和由模式选择单元202生成的预测块,来产生与当前块相对应的重构块(尽管可能具有一定程度的失真)。例如,重构单元214可以将重构的残差块的样本添加到模式选择单元202所生成的预测块中的对应样本,以生成重构的块。

[0129] 滤波单元216可以对重构的块执行一个或多个滤波操作。例如,滤波单元216可以执行解块操作以减少沿着CU的边缘的块状伪影。在一些例子中,可以跳过滤波单元216的操作。

[0130] 视频编码器200将重构的块存储在DPB 218中。例如,在不执行滤波单元216的操作的例子中,重构单元214可以将重构的块存储到DPB 218中。在执行滤波单元216的操作的例子中,滤波单元216可以将滤波后的重构块存储到DPB 218中。运动估计单元222和运动补偿单元224可以从DPB 218检索参考图片,该参考图片由重构(并且可能滤波)的块形成,以对

随后编码的图片进行帧间预测。另外,帧内预测单元226可以使用当前图片的DPB 218中的重构块,对当前图片中的其它块进行帧内预测。

[0131] 通常,熵编码单元220可以对从视频编码器200的其它功能组件接收的语法元素进行熵编码。例如,熵编码单元220可以对来自量化单元208的量化的系数块进行熵编码。再举一个例子,熵编码单元220可以对来自模式选择单元202的预测语法元素(例如,用于帧间预测的运动信息或者用于帧内预测的帧内模式信息)进行熵编码。熵编码单元220可以对作为视频数据的另一个例子的语法元素执行一个或多个熵编码操作,以生成熵编码的数据。例如,熵编码单元220可以执行上下文自适应可变长度编码(CAVLC)操作、CABAC操作、变量至变量(V2V)长度编码操作、基于语法的上下文自适应二进制算术编码(SBAC)操作、概率间隔分割熵(PIPE)编码操作、指数格伦布编码操作、或者对数据的另一种类型的熵编码操作。在一些例子中,熵编码单元220可以在不对语法元素进行熵编码的旁通模式下操作。

[0132] 视频编码器200可以输出比特流,该比特流包括用于重构片段或图片的块所需要的经熵编码的语法元素。具体而言,熵编码单元220可以输出比特流。

[0133] 关于块描述了上面所描述的操作。这样的描述应当被理解为用于亮度译码块和/或色度译码块的操作。如上所述,在一些例子中,亮度译码块和色度译码块是CU的亮度和色度分量。在一些例子中,亮度译码块和色度译码块是PU的亮度和色度分量。

[0134] 在一些例子中,不需要针对色度译码块重复针对亮度译码块执行的操作。举一个例子,不需要重复用于识别亮度译码块的运动矢量(MV)和参考图片的操作,来识别用于色度块的MV和参考图片。相反,可以缩放用于亮度译码块的MV以确定用于色度块的MV,并且参考图片可以是相同的。再举一个例子,对于亮度译码块和色度译码块,帧内预测处理可以是相同的。

[0135] 视频编码器200表示用于对视频数据进行编码的设备的例子,该设备包括被配置为存储视频数据的存储器、以及在电路中实现并且被配置为执行以下操作的一个或多个处理单元:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在编码的系数的第一相邻系数的值;确定当前正在编码的系数的第二相邻系数的值;基于用于第一相邻系数的值和用于第二相邻系数的值,确定当前正在编码的系数的上下文偏移;并基于所确定的上下文偏移,对当前正在编码的系数的值进行编码。例如,第一相邻系数可以是上相邻系数或左相邻系数中的一个,而第二相邻系数可以是上相邻系数或左相邻系数中的另一个。

[0136] 为了基于所确定的上下文偏移,对当前正在编码的系数的值进行编码,视频编码器200可以被配置为:基于所确定的上下文偏移来确定上下文;确定当前正在编码的系数的符号;并基于所确定的上下文,对一个或多个数据箱(bin)进行上下文编码以表示当前正在编码的系数的符号。视频编码器200可以被配置为:对视频数据的残差块的残差值进行量化,以确定当前正在编码的系数的值。视频编码器200可以被配置为:确定预测块,并将该预测块与视频数据的原始块进行比较以确定视频数据的残差块。

[0137] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:基于第一相邻系数的值和第二相邻系数的值,从三个可用的上下文偏移中选择上下文偏移。这三个可用的上下文偏移可以包括:针对当第一相邻系数和第二相邻系数都等于零或者具有相反符号时的第一上下文偏移;针对当第

一相邻系数和第二相邻系数都为正,或者第一相邻系数和第二相邻系数中的一个等于零并且第一相邻系数和第二相邻系数中的另一个为正时的第二上下文偏移;以及针对当第一相邻系数和第二相邻系数都为负,或者第一相邻系数和第二相邻系数中的一个等于零并且第一相邻系数和第二相邻系数中的另一个为负时的第三上下文偏移。

[0138] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值等于零并且第二相邻系数的值等于零,将上下文偏移值设置为第一偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值是大于零或小于零中的一个,并且第二相邻系数的值是大于零或小于零中的另一个,将上下文偏移值设置为第一偏移值。

[0139] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值和第二相邻系数的值均为非零值且具有相反的符号,将上下文偏移值设置为第一偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值大于或等于零并且第二相邻系数的值大于或等于零,将上下文偏移值设置为第二偏移值,其中第一相邻系数的值或第二相邻系数的值中的至少一者大于或等于一。

[0140] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值和第二相邻系数的值均为非负值,将上下文偏移值设置为第二偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在编码的系数的上下文偏移,视频编码器200可以被配置为:响应于第一相邻系数的值小于或等于零并且第二相邻系数的值小于或等于零,将上下文偏移值设置为第三偏移值,其中第一相邻系数的值或者第二相邻系数的值小于或等于负一(-1)。第一偏移值、第二偏移值和第三偏移值可以是例如不同的偏移值。上下文偏移是用于确定上下文的值。因此,可以认为第一偏移值、第二偏移值和第三偏移值标识或者指代三个不同的上下文。

[0141] 视频编码器200还表示被配置为对视频数据进行编码的设备的例子,该设备包括被配置为存储视频数据的存储器、以及利用电路来实现的一个或多个处理单元,所述一个或多个处理单元被配置为:针对视频数据的使用变换跳过模式进行编码的残差块,确定当前正在编码的系数的第一相邻系数的绝对系数水平;确定当前正在编码的系数的第二相邻系数的绝对系数水平;确定当前正在编码的系数的绝对系数水平;并基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数水平,对指示当前正在编码的系数的绝对系数水平的一个或多个语法元素进行编码。例如,视频编码器200可以对视频数据的残差块的残差值进行量化,以确定当前正在编码的系数的值。

[0142] 在一些例子中,视频编码器200可以被配置为:基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数水平来确定预测值水平;并使用指示预测值水平等于当前正在编码的系数的绝对系数水平的语法元素的第一值、以及指示预测值水平不等于当前正在编码的系数的绝对系数水平的语法元素的第二值,对语法元素进行编码。在一些例子中,视频编码器200可以被配置为:基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数

水平来确定预测值水平;基于预测值水平和当前正在编码的系数的绝对系数水平,确定用于语法元素的值;对语法元素进行编码。为了确定预测值水平,视频编码器200可以被配置为:将预测值水平设置为第一相邻系数的绝对系数水平或第二相邻系数的绝对系数水平中的较大者。

[0143] 图5是示出可以执行本公开内容的技术的示例性视频解码器300的框图。提供图5以便于解释的目的,故其不应被认为是对本公开内容中广泛例示和描述的技术的限制。为了便于说明起见,本公开内容描述了根据JEM、VVC和HEVC的技术的视频解码器300。但是,本公开内容的技术可以由被配置为实现其它视频编码标准的视频编码设备来执行。

[0144] 在图5的例子中,视频解码器300包括编码图片缓冲器(CPB)存储器320、熵解码单元302、预测处理单元304、逆量化单元306、逆变换处理单元308、重构单元310、滤波单元312和解码图片缓冲器(DPB)314。CPB存储器320、熵解码单元302、预测处理单元304、逆量化单元306、逆变换处理单元308、重构单元310、滤波单元312和DPB 314中的任何一个或全部,可以在一个或多个处理器中或者在处理电路中实现。此外,视频解码器300可以包括另外的或替代的处理器或处理电路,以执行这些功能和其它功能。

[0145] 预测处理单元304包括运动补偿单元316和帧内预测单元318。预测处理单元304可以包括用于根据其它预测模式来执行预测的其它单元。举例而言,预测处理单元304可以包括调色板单元、块内复制单元(其可以形成运动补偿单元316的一部分)、仿射单元、线性模型(LM)单元等等。在其它例子中,视频解码器300可以包括更多、更少或者不同的功能组件。

[0146] CPB存储器320可以存储将由视频解码器300的组件解码的视频数据(例如,编码的视频比特流)。例如,可以从计算机可读介质110(图1)中获得存储在CPB存储器320中的视频数据。CPB存储器320可以包括存储来自编码视频比特流的编码视频数据(例如,语法元素)的CPB。而且,CPB存储器320可以存储除编码图片的语法元素之外的视频数据,例如,表示来自视频解码器300的各个单元的输出的临时数据。DPB 314通常存储解码的图片,视频解码器300在解码编码的视频比特流的后续数据或图片时,可以输出和/或使用该解码图片作为参考视频数据。CPB存储器320和DPB 314可以由诸如DRAM、SDRAM、MRAM、RRAM或其它类型的存储设备之类的各种存储设备中的任何一种来形成。CPB存储器320和DPB 314可以由相同的存储设备或不同的存储设备来提供。在各个例子中,CPB存储器320可以与视频解码器300的其它组件一起在芯片上,或者相对于那些组件在芯片外。

[0147] 另外地或替代地,在一些例子中,视频解码器300可以从存储器120(图1)中检索译码的视频数据。也就是说,存储器120可以如上文参照CPB存储器320所讨论的来存储数据。类似地,当视频解码器300的一些或全部功能利用由视频解码器300的处理电路执行的软件来实现时,存储器120可以存储将由视频解码器300执行的指令。

[0148] 示出了图5的各个单元以帮助理解由视频解码器300执行的操作。这些单元可以实现为固定功能电路、可编程电路或者其组合。类似于图4,固定功能电路指代提供特定功能、并在可以执行的操作上预先设置的电路。可编程电路指代可以被编程以执行各种任务,并且在可以执行的操作中提供灵活功能的电路。例如,可编程电路可以执行软件或固件,软件或固件使可编程电路以软件或固件的指令所定义的方式进行操作。固定功能电路可以执行软件指令(例如,用于接收参数或输出参数),但是固定功能电路执行的操作的类型通常是不可变的。在一些例子中,这些单元中的一个或多个单元可以是不同的电路块(固定功能或

可编程),并且在一些例子中,所述一个或多个单元可以是集成电路。

[0149] 视频解码器300可以包括ALU、EFU、数字电路、模拟电路和/或由可编程电路形成的可编程核。在通过在可编程电路上执行的软件来执行视频解码器300的操作的例子中,片上或片外存储器可以存储视频解码器300接收并执行的软件的指令(例如,目标代码)。

[0150] 熵解码单元302可以从CPB接收编码的视频数据,并且对视频数据进行熵解码以再现语法元素。预测处理单元304、逆量化单元306、逆变换处理单元308、重构单元310和滤波单元312可以基于从比特流中提取的语法元素来生成解码的视频数据。

[0151] 通常,视频解码器300在逐块的基础上重建图片。视频解码器300可以单独地对每个块执行重构操作(其中,当前正在重构(即,解码)的块可以称为“当前块”)。

[0152] 熵解码单元302可以对定义量化的系数块的量化系数的语法元素、以及诸如量化参数(QP)和/或变换模式指示之类的变换信息进行熵解码。逆量化单元306可以使用与量化的系数块相关联的QP来确定量化程度,并且同样地,确定用于逆量化单元306应用的逆量化程度。例如,逆量化单元306可以执行按比特左移运算以对变换系数进行逆量化。逆量化单元306可以由此形成包括系数的系数块。

[0153] 在逆量化单元306针对于进行变换的块来形成系数块之后,逆变换处理单元308可以将一个或多个逆变换应用于变换系数块以生成与当前块相关联的残差块。例如,逆变换处理单元308可以向这些变换系数块应用逆DCT、逆整数变换、逆Karhunen-Loeve变换(KLT)、逆旋转变换、逆方向变换或者另一种逆变换。对于以变换跳过模式译码的块,逆变换处理单元308可以不执行逆变换,并且在这些译码场景中,可以被视为不处理或改变系数块的直通单元。

[0154] 此外,预测处理单元304根据由熵解码单元302进行熵解码的预测信息语法元素,来生成预测块。例如,如果预测信息语法元素指示当前块是帧间预测的,则运动补偿单元316可以生成预测块。在这种情况下,预测信息语法元素可以指示DPB 314中的从其检索参考块的参考图片、以及标识参考图片中的参考块相对于当前图片中的当前块的位置的运动矢量。运动补偿单元316通常可以以与关于运动补偿单元224(图4)所描述的方式基本上相似的方式,来执行帧间预测处理。

[0155] 再举一个例子,如果预测信息语法元素指示当前块是帧内预测的,则帧内预测单元318可以根据由预测信息语法元素指示的帧内预测模式来生成预测块。再次,帧内预测单元318通常可以以与关于帧内预测单元226(图4)所描述的方式基本上相似的方式来执行帧内预测处理。帧内预测单元318可以从DPB 314检索当前块的相邻样本的数据。

[0156] 重构单元310可以使用预测块和残差块来重构当前块。例如,重构单元310可以将残差块的样本添加到预测块的对应样本以重构当前的块。

[0157] 滤波单元312可以对重构的块执行一个或多个滤波操作。例如,滤波单元312可以执行解块操作,以减少沿着重构块的边缘的块状伪像。不一定在所有示例中都执行滤波单元312的操作。

[0158] 视频解码器300可以将重构的块存储在DPB 314中。例如,在不执行滤波单元312的操作的例子中,重构单元310可以将重构的块存储到DPB 314中。在执行滤波单元312的操作的例子中,滤波单元312可以将滤波的重构块存储到DPB 314中。如上所述,DPB 314可以向预测处理单元304提供参考信息,例如用于帧内预测的当前图片的样本以及用于后续运动

补偿的先前解码的图片。此外,视频解码器300可以从DPB输出解码的图片,以便随后在诸如图1的显示设备118之类的显示设备上呈现。

[0159] 视频解码器300表示视频解码设备的例子,该视频解码设备包括被配置为存储视频数据的存储器、以及在电路中实现并且被配置为执行以下操作的一个或多个处理单元:针对视频数据的使用变换跳过模式进行编码的残差块,确定用于当前正在解码的系数的第一相邻系数的值;确定用于当前正在解码的系数的第二相邻系数的值;基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移;并基于所确定的上下文偏移,对当前正在解码的系数的值进行解码。例如,第一相邻系数可以是上相邻系数或左相邻系数中的一个,而第二相邻系数可以是上相邻系数或左相邻系数中的另一个。如上所述,对于以变换跳过模式译码的残差块,所述系数值可以对应于残差值,例如,经量化的残差值或者非量化的残差值。

[0160] 为了基于所确定的上下文偏移,对当前正在解码的系数的值进行解码,视频解码器300可以被配置为:基于所确定的上下文偏移来确定上下文;接收一个或多个数据箱;并基于所确定的上下文,对所述一个或多个数据箱进行上下文解码以确定当前正在解码的系数的符号。视频解码器300可以被配置为:对当前正在解码的系数的值进行逆量化,以确定视频数据的残差块的残差值。视频解码器300可以被配置为:基于当前正在解码的系数的值,确定解码的残差块;将解码的残差块与预测块相加以确定重构块;对重构块执行一个或多个滤波操作,以确定视频数据的解码块;并输出包括视频数据的解码块的视频数据的解码图片。

[0161] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:基于第一相邻系数的值和第二相邻系数的值,从三个可用的上下文偏移中选择上下文偏移。这三个可用的上下文偏移可以包括:当第一相邻系数和第二相邻系数都等于零或者具有相反符号时的第一上下文偏移;当第一相邻系数和第二相邻系数都为正,或者第一相邻系数和第二相邻系数中的一个等于零并且第一相邻系数和第二相邻系数中的另一个为正时的第二上下文偏移;以及当第一相邻系数和第二相邻系数都为负,或者第一相邻系数和第二相邻系数中的一个等于零并且第一相邻系数和第二相邻系数中的另一个为负时的第三上下文偏移。

[0162] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值等于零并且第二相邻系数的值等于零,将上下文偏移值设置为第一偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值是大于零或小于零中的一个,并且第二相邻系数的值是大于零或小于零中的另一个,将上下文偏移值设置为第一偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值和第二相邻系数的值均为非零值且具有相反的符号,将上下文偏移值设置为第一偏移值。

[0163] 为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值大于或等于零并且第二相邻系数的值大于或等于零,将上下文偏移值设置为第二偏移值,其中第一相邻系数的

值或第二相邻系数的值中的至少一者大于或等于一。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值和第二相邻系数的值均为非负值,将上下文偏移值设置为第二偏移值。为了基于第一相邻系数的值和第二相邻系数的值来确定当前正在解码的系数的上下文偏移,视频解码器300可以被配置为:响应于第一相邻系数的值小于或等于零并且第二相邻系数的值小于或等于零,将上下文偏移值设置为第三偏移值,其中第一相邻系数的值或者第二相邻系数的值小于或等于负一(-1)。

[0164] 视频解码器300还表示视频解码设备的例子,该设备包括被配置为存储视频数据的存储器、以及利用电路来实现的一个或多个处理单元,所述一个或多个处理单元被配置为:针对视频数据的使用变换跳过模式进行编码的残差块,确定当前正在解码的系数的第一相邻系数的绝对系数水平;确定当前正在解码的系数的第二相邻系数的绝对系数水平;基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数水平,确定当前正在解码的系数的绝对系数水平。例如,视频解码器300可以对当前正在解码的系数进行逆量化,以确定视频数据的残差块的残差值。

[0165] 为了确定当前正在解码的系数的绝对系数水平,视频解码器300可以被配置为:基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数水平,确定预测值(predictor)水平;接收语法元素,其中用于该语法元素的第一值指示预测值水平等于当前正在解码的系数的绝对系数水平,用于该语法元素的第二值指示预测值水平不等于当前正在解码的系数的绝对系数水平。为了确定当前正在解码的系数的绝对系数水平,视频解码器300可以被配置为:基于第一相邻系数的绝对系数水平和第二相邻系数的绝对系数水平,确定预测值水平;接收语法元素;并基于预测值水平和该语法元素,确定当前正在解码的系数的绝对系数水平。为了确定预测值水平,视频解码器300可以将预测值水平设置为第一相邻系数的绝对系数水平或第二相邻系数的绝对系数水平中的较大者。

[0166] 图6A和6B示出了箱n处的CABAC过程的示例。在图6A的例子400中,箱n处的范围包括在给定特定上下文状态(σ)时,由最不可能符号(LPS)的概率(p_σ)给出的RangeMPS和RangeLPS。示例400示出了当箱n的值等于最可能符号(MPS)时,箱n+1处的范围的更新。在该例子中,该范围的下值保持不变,但箱n+1处的范围值减少为箱n处的RangeMPS的值。图6B的例子402示出了当箱n的值不等于MPS(即,等于LPS)时,在箱n+1处的范围的更新。在该例子中,该范围的下值移动到箱n处的RangeLPS的较低范围值。此外,箱n+1处的范围值减少为箱n处的RangeLPS的值。

[0167] 在HEVC视频译码过程的一个例子中,使用9比特来表示范围,并且使用10比特来表示下值。存在重新归一化过程,以保持足够精确的范围和下值。每当范围小于256时,就会发生重新归一化。因此,在重新归一化后,该范围始终等于或大于256。根据范围的值和下值,二进制算术译码器(BAC)向比特流输出“0”或“1”,或者更新内部变量(称为BO:比特突出)以保持未来的输出。图7显示了取决于范围的BAC输出的例子。例如,当范围和下值高于某个阈值(例如,512)时,将‘1’输出到比特流。当范围和下值低于某个阈值(例如,512)时,向比特流输出‘0’。当范围和下值在特定阈值之间时,不会向比特流输出任何内容。相反,递增BO值,对下一个箱进行编码。

[0168] 在H.264/AVC的CABAC上下文模型中以及在HEVC的一些例子中,存在128个状态。存

在64种可能的LPS概率(通过状态σ来表示),其可以是从0到63。每个MPS可以是零或一。因此,128个状态是64种状态概率乘以MPS的2个可能值(0或1)。因此,可以用7比特来索引该状态。

[0169] 为了减少推导LPS范围(rangeLPS$_\sigma$)的计算量,可以预先计算所有情况的结果,并且将其作为近似值存储在查找表中。因此,通过使用简单的查表,无需任何乘法即可获得LPS范围。避免乘法对于某些设备或应用程序可能很重要,这是因为这种操作可能在许多硬件架构中导致显著的延迟。

[0170] 可以使用4列预先计算的LPS范围表来代替乘法。将该范围划分为四个段。可以通过式(range>>6)&3导出段索引。实际上,通过从实际范围中移位和删除比特来导出段索引。下表的1显示了可能的范围及其相应的索引。

[0171] 表1-范围索引

[0172]

范围	256-319	320-383	384-447	448-511
(range>>6)&3	0	1	2	3

[0173] LPS范围表于是具有64个条目(每个概率状态一个条目)乘以4(每个范围索引一个)。每个条目是RangeLPS,即范围乘以LPS概率的值。在下面的表2中显示了该表的一部分的例子。表2描述了概率状态9-12。在HEVC的一项提议中,概率状态的范围可以从0到63。

[0174] 表2-RangeLPS

[0175]

概率状态(σ)	RangeLPS 索引 0	索引 1	索引 2	索引 3
…	…	…	…	…
9	90	110	130	150
10	85	104	123	142
11	81	99	117	135
12	77	94	111	128
…	…	…	…	…

[0176] 在每个段(即,范围值)中,每个概率状态σ的LPS范围是预定义的。换句话说,将概率状态σ的LPS范围量化为四个值(即,每个范围索引的一个值)。在给定点使用的特定LPS范围取决于该段所属的范围。该表中使用的可能LPS范围的数量是表列数(即,可能的LPS范围值的数量)和LPS范围精度之间的权衡。一般来说,更多的列导致LPS范围值的量化误差更小,但也会增加使用更多存储器来存储表的需求。更少的列会增加量化误差,但也会减少存储表所需要的存储器。

[0177] 如上所述,每个LPS概率状态具有对应的概率。将每个状态的概率p导出如下:

[0178] $p_\sigma = \alpha p_{\sigma-1}$

[0179] 其中,状态σ是从0到63。常数α表示每个上下文状态之间的概率变化量。在一个例子中,α=0.9493,或者更准确地说,$\alpha = (0.01875/0.5)^{1/63}$。状态σ=0的概率等于0.5(即,$p_0 = 1/2$)。也就是说,在上下文状态0下,LPS和MPS的概率相等。通过将前一个状态乘以α来导出每个连续状态的概率。因此,LPS在上下文状态α=1下发生的概率为p_0*0.9493(0.5*0.9493=.47465)。因此,随着状态索引α的增加,LPS发生的概率下降。

[0180] CABAC是自适应的,这是因为对概率状态进行更新以便遵循信号统计(即,先前译码的箱的值)。更新过程如下。对于给定的概率状态,更新取决于状态索引和标识为LPS或MPS的编码符号的值。作为更新过程的结果,导出新的概率状态,其包括潜在修改的LPS概率估计和修改的MPS值(如果需要的话)。

[0181] 在箱值等于MPS的情况下,可以将给定状态索引增加1。这适用于所有状态,除了当MPS发生在状态索引62时,其中LPS概率已经处于其最小值(或者等效地,达到最大MPS概率)。在这种情况下,状态索引62保持固定,直到看到LPS,或者对最后一个箱值进行编码为止(状态63用于最后一个箱值的特殊情况)。当LPS发生时,通过将状态索引减少一定量来改变状态索引,如下面的等式所示。这种规则一般适用于LPS的每次发生,但有以下例外。假设已在索引$\sigma=0$的状态下编码了LPS,这对应于等概率情况,状态索引保持固定,但MPS值将被切换,使得LPS和MPS的值将互换。在所有其它情况下,无论编码了哪个符号,MPS值都不会改变。LPS概率的转换规则的推导是基于给定LPS概率$p_{旧}$与其更新的对应物$p_{新}$之间的以下关系:

[0182] 如果发生MPS,$p_{新}=\max(\alpha p_{旧},p_{62})$

[0183] 如果发生LPS,$p_{新}=(1-\alpha)+\alpha p_{旧}$

[0184] 关于CABAC中的概率估计处理的实际实现,重要的是要注意,可以通过至多两个表来实现所有转换规则,每个表都具有63条的6位无符号整数值。在一些例子中,可以使用单个表TransIdxLPS来确定状态转换,其中对于给定的状态索引σ,表TransIdxLPS确定在观察到LPS的情况下的新的更新状态索引TransIdxLPS[σ]。可以通过将状态索引简单(饱和)地增加固定值1来获得MPS驱动的转换,从而产生更新的状态索引$\min(\sigma+1,62)$。下面的表3是一个局部TransIdxLPS表的例子。

[0185] 表3-TransIdxLPS

[0186]

概率状态(σ)	新状态TransIdxLPS[σ]
…	…
9	6
10	8
11	8
12	8
…	…

[0187] 上面关于图6A、6B和图7所描述的技术仅代表CABAC的一种示例性实施方式。应当理解的是,本公开内容的技术并不仅限于所描述的CABAC的实现。例如,在较旧的BAC方法中(例如,在H.264/AVC中使用的BAC方法),针对低分辨率视频调整了RangeLPS和TransIdxLPS表(即,通用中间格式(CIF)和四分之一CIF(QCIF)视频)。使用HEVC和未来的编解码器(例如,VVC),大量的视频内容是高清(HD)的,并且在一些情况下比HD更高。高清或高于高清分辨率的视频内容往往与用于开发H.264/AVC的10年前的QCIF序列具有不同的统计数据。因此,来自H.264/AVC的RangeLPS和TransIdxLPS表可能会以过快的方式导致状态之间的适应。也就是说,概率状态之间的转换(尤其是当LPS发生时)对于更流畅、更高分辨率的高清视频内容来说可能太大了。因此,根据传统技术使用的概率模型对于高清和超高清内容可能不那么准确。此外,由于HD视频内容包括更大范围的像素值,因此H.264/AVC表不包含足

够的条目来说明高清内容中可能存在的更极端值。

[0188] 这样,对于HEVC和诸如VVC之类的未来编码标准,可以修改RangeLPS和TransIdxLPS表以考虑到这种新内容的特性。具体而言,HEVC和未来编码标准的BAC过程可以使用允许较慢的适应过程的表,并且可能考虑更极端的情况(即,偏斜化的概率)。因此,举一个例子,可以修改RangeLPS和TransIdxLPS表,以便与在具有H.264/AVC或HEVC的BAC中使用的概率状态和范围相比,通过包括更多的概率状态和范围来实现这些目标。

[0189] 图8是例如形成如图4中所示的视频编码器200的一部分的示例性熵编码单元220的框图,其可以被配置为根据本公开内容的技术来执行CABAC。将语法元素418输入到熵编码单元220中。如果语法元素已经是二进制值语法元素(即,只有值0和1的语法元素),则可以跳过二值化的步骤。如果语法元素是非二进制值语法元素(例如,通过多个比特表示的语法元素,例如系数水平),则通过二进制化器420来二进制化非二进制值语法元素。二进制化器420执行非二进制值语法元素到二进制决策序列的映射。这些二进制决策通常称为“箱”。例如,对于系数水平,该水平的值可以分解为连续的箱,每个箱指示系数水平的绝对值是否大于某个值。例如,箱0(有时称为显著性标志)指示系数水平的绝对值是否大于0。箱1指示系数水平的绝对值是否大于1,以此类推。可以针对每个非二进制值语法元素,开发唯一映射。

[0190] 由二进制化器420产生的每个箱馈送到熵编码单元220的二进制算术译码端。也就是说,对于预定集合的非二进制值语法元素,在下一个箱类型(例如,箱1)之前,对每个箱类型(例如,箱0)进行译码。可以在常规模式或旁路模式下执行译码。在旁路模式下,旁路编码引擎426使用固定概率模型(例如,使用格伦布-莱斯或指数格伦布译码)来执行算术编码。旁路模式通常用于更可预测的语法元素。

[0191] 常规模式下的编码涉及执行CABAC。常规模式CABAC用于对箱值进行译码,其中给定先前译码的箱的值,箱的值的概率是可预测的。箱是LPS的概率由上下文建模器422来确定。上下文建模器422输出箱值和上下文模型(例如,概率状态σ)。上下文模型可以是用于一系列箱的初始上下文模型,或者可以基于先前译码的箱的译码值来确定。如上所述,上下文建模器可以基于先前译码的箱是MPS还是LPS来更新状态。

[0192] 在上下文建模器422确定上下文模型和概率状态σ之后,常规编码引擎424对箱值执行BAC。根据本公开内容的技术,常规编码引擎424使用TransIdxLPS表430来执行BAC,其中TransIdxLPS表430包括超过64个概率状态σ。在一个例子中,概率状态的数量是128。在前一个箱(箱n)是LPS时,使用TransIdxLPS来确定下一箱(箱n+1)使用哪种概率状态。给定特定概率状态σ,常规编码引擎424还可以使用RangeLPS表428来确定LPS的范围值。但是,根据本公开内容的技术,不是使用TransIdxLPS表430的所有可能的概率状态σ,而是将概率状态索引σ映射到分组索引以便在RangeLPS表中使用。也就是说,到RangeLPS表428中的每个索引可以表示全部数量的概率状态中的两个或更多个。概率状态索引σ到经分组索引的映射可以是线性的(例如,除以二),也可以是非线性的(例如,对数函数或映射表)。

[0193] 在本公开内容的其它例子中,可以通过将参数α设置为大于0.9493来使连续概率状态之间的差异更小。在一个例子中,α=0.9689。在本公开内容的另一个例子中,可以将LPS发生的最高概率(p_0)设置为低于0.5。在一个例子中,p_0可以等于0.493。

[0194] 根据本公开内容的一种或多种技术,与在二进制算术编码过程中使用用于更新概

率状态的变量的相同值(例如,窗大小、缩放因子(α)和概率更新速度中的一个或多个)相反,熵编码单元220可以针对不同的上下文模型和/或不同的语法元素使用变量的不同值。例如,熵编码单元220可以针对多个上下文模型中的上下文模型,确定在二进制算术编码过程中用于更新概率状态的变量的值,并基于所确定的值来更新概率状态。

[0195] 图9是例如形成如图5中所示的视频解码器300的一部分的示例性熵解码单元302的框图,其可以被配置为根据本公开内容的技术执行CABAC。图9的熵解码单元302以与图8中描述的熵编码单元220相反的方式来执行CABAC。将来自比特流448的译码比特输入到熵解码单元302。基于译码比特是使用旁路模式还是常规模式进行熵编码,将译码比特馈送到上下文建模器450或旁路解码引擎452。如果译码比特是以旁路模式进行译码的,则旁路解码引擎452可以例如使用Golomb-Rice或指数Golomb解码来检索二进制值语法元素或非二进制语法元素的箱。

[0196] 如果译码比特是以常规模式译码的,则上下文建模器450可以确定译码比特的概率模型,并且常规解码引擎454可以对译码比特进行解码以产生非二进制值语法元素(或者如果是二进制值,则是语法元素本身)的箱。在上下文建模器450确定上下文模型和概率状态σ之后,常规解码引擎454对箱值执行BAC。根据本公开内容的技术,常规解码引擎454使用TransIdxLPS表458来执行BAC,其中TransIdxLPS表458包括超过64种概率状态σ。在一个例子中,概率状态的数量是128个,但是可以定义其它数量的概率状态,这与本公开内容的技术一致。在前一个箱(箱n)是LPS时,使用TransIdxLPS表458来确定哪个概率状态用于下一箱(箱n+1)。常规解码引擎454还可以在给定特定概率状态σ的情况下,使用RangeLPS表456来确定LPS的范围值。然而,根据本公开内容的技术,不是使用TransIdxLPS表458的所有可能的概率状态σ,而是将概率状态索引σ映射到经分组索引以便在RangeLPS表456中使用。也就是说,到RangeLPS表456中的每个索引可以表示全部数量的概率状态中的两个或更多个。概率状态索引σ到分组索引的映射可以是线性的(例如,除以二),也可以是非线性的(例如,对数函数或映射表)。

[0197] 在本公开内容的其它例子中,可以通过将参数α设置为大于0.9493来使连续概率状态之间的差异更小。在一个例子中,α=0.9689。在本公开内容的另一个例子中,可以将LPS发生的最高概率(p_0)设置为低于0.5。在一个例子中,p_0可以等于0.493。

[0198] 在常规解码引擎454对箱进行解码之后,逆二进制化器460可以执行反向映射以将箱转换回非二进制值语法元素的值。

[0199] 图10是示出用于对当前块进行编码的示例方法的流程图。当前块可以包括当前CU。虽然相对于视频编码器200(图1和4)进行了描述,但应当理解,其它设备可以被配置为执行与图10类似的方法。

[0200] 在该例子中,视频编码器200在初始时预测当前块(550)。举例来说,视频编码器200可以形成针对当前块的预测块。然后,视频编码器200可以针对当前块来计算残差块(552)。为了计算残差块,视频编码器200可以计算原始未译码的块与当前块的预测块之间的差。随后,视频编码器200可以对残差块进行变换,并量化残差块的系数(554)。在诸如变换跳过模式之类的一些译码模式中,视频编码器200可以跳过变换,并仅仅对残差数据进行量化。接下来,视频编码器200可以扫描残差块的经量化的系数(556)。在扫描期间或者在扫描之后,视频编码器200可以对系数进行熵编码(558)。例如,视频编码器200可以使用CAVLC

或CABAC,对系数进行编码。然后,视频编码器200可以输出这些系数的经熵编码的数据(560)。

[0201] 图11是示出用于对视频数据的当前块进行解码的示例方法的流程图。当前块可以包括当前CU。虽然相对于视频解码器300(图1和5)进行了描述,但应当理解,其它设备可以被配置为执行与图11类似的方法。

[0202] 视频解码器300可以接收当前块的经熵编码的数据(例如,经熵编码的预测信息和用于对应于当前块的残差块的系数的经熵编码的数据)(570)。视频解码器300可以对熵编码的数据进行熵解码,以确定当前块的预测信息并再现残差块的系数(572)。视频解码器300可以例如使用如针对当前块的预测信息所指示的帧内或帧间预测模式来预测当前块(574),以计算针对当前块的预测块。然后,视频解码器300可以对再现的系数进行逆扫描(576),以创建量化的系数的块。然后,视频解码器300可以对这些系数进行逆量化,并将逆变换应用于这些系数以产生残差块(578)。在诸如变换跳过模式之类的一些译码模式中,视频解码器300可以跳过逆变换,并仅仅对系数进行逆量化。视频解码器300可以通过组合预测块和残差块,来最终解码当前块(580)。

[0203] 图12是示出用于确定对残差块的系数的符号进行译码(例如,编码或解码)的上下文的示例方法的流程图。例如,残差块可以是跳过变换的残差块。将关于通用视频译码器来描述图12的技术,其中该通用视频译码器可以对应于诸如视频编码器200之类的视频编码器或者诸如视频解码器300之类的视频解码器。然而,应当理解的是,其它设备也可以被配置为执行类似于图12的方法。

[0204] 视频译码器确定当前正在译码的系数的第一相邻系数的值(X0)(600)。视频译码器确定当前正在译码的系数的第二相邻系数的值(X1)(602)。响应于X0和X1二者都等于0(604,是),视频译码器将用于译码当前正在译码的系数的符号的上下文设置为第一上下文(606)。响应于X0或X1中的至少一者不等于零(604,否),但X0和X1具有相反的符号(608,是),视频译码器还将用于译码当前正在译码的系数的符号的上下文设置为第一上下文(610)。

[0205] 响应于X0或X1中的至少一者不等于零(604,否),X0和X1不具有相反的符号(608,否),并且X0或X1之一大于零(612,是),视频译码器将用于译码当前正在译码的系数的符号的上下文设置为不同于第一上下文的第二上下文(614)。响应于X0或X1中的至少一者不等于零(604,否),X0和X1不具有相反的符号(608,否),并且X0和X1都不大于零(612,否),视频译码器将用于译码当前正在译码的系数的符号的上下文设置为不同于第一上下文或第二上下文的第三上下文(616)。

[0206] 应当认识到,根据示例,本文所描述的任何技术的某些动作或事件可以以不同的顺序执行、可以进行添加、合并或者完全省略(例如,并非所有描述的动作或事件都是实施该技术所必需的)。此外,在某些例子中,可以例如通过多线程处理、中断处理或多个处理器并发地而不是顺序地执行动作或事件。

[0207] 在一个或多个例子中,所描述的功能可以利用硬件、软件、固件或者其任意组合来实现。当利用软件实现时,可以将这些功能存储在计算机可读介质上,或者作为计算机可读介质上的一个或多个指令或代码进行传输,并由基于硬件的处理单元来执行。计算机可读介质可以包括计算机可读存储介质,计算机可读存储介质对应于诸如数据存储介质或通信

介质之类的有形介质,其中通信介质包括有助于例如根据通信协议,将计算机程序从一个地方传送到另一个地方的任何介质。用此方式,计算机可读介质通常可以对应于:(1)非临时性的有形计算机可读存储介质;或者(2)诸如信号或载波波形之类的通信介质。数据存储介质可以是一个或多个计算机或者一个或多个处理器能够进行访问以获取用于实现本公开内容中描述的技术的指令、代码和/或数据结构的任何可用介质。计算机程序产品可以包括计算机可读介质。

[0208] 举例而言,但非做出限制,这种计算机可读存储介质可以包括RAM、ROM、EEPROM、CD-ROM或者其它光盘存储器、磁盘存储器或其它磁存储设备、闪存或者能够用于存储具有指令或数据结构形式的期望的程序代码并能够由计算机进行存取的任何其它介质。此外,可以将任何连接适当地称作计算机可读介质。举例而言,如果指令是使用同轴电缆、光纤光缆、双绞线、数字用户线路(DSL)或者诸如红外线、无线和微波之类的无线技术,从网站、服务器或其它远程源传输的,那么所述同轴电缆、光纤光缆、双绞线、DSL或者诸如红外线、无线和微波之类的无线技术包括在所述介质的定义中。但是,应当理解的是,计算机可读存储介质和数据存储介质并不包括连接、载波波形、信号或者其它临时介质,而是针对于非临时的有形存储介质。如本文所使用的,磁盘和光盘包括压缩光盘(CD)、激光光盘、光盘、数字通用光盘(DVD)、软盘和蓝光光盘,其中磁盘通常磁性地复制数据,而光盘则用激光来光学地复制数据。上述的组合也应当包括在计算机可读介质的保护范围之内。

[0209] 指令可以由诸如一个或多个数字信号处理器(DSP)、通用微处理器、专用集成电路(ASIC)、现场可编程门阵列(FPGA)之类的一个或多个处理器或者其它等同的集成或分立逻辑电路来执行。因此,如本文所使用的,术语“处理器”和“处理电路”可以指代前述的结构或者适合于实现本文所描述的技术的任何其它结构中的任何一种。此外,在一些方面,本文所描述的功能可以提供在被配置为实现编码和解码的专用硬件和/或软件模块中,或者并入到组合的编解码器中。此外,这些技术可以在一个或多个电路或逻辑元件中完全实现。

[0210] 本公开内容的技术可以使用多种多样的设备或装置来实现,其包括使用无线手持装置、集成电路(IC)或者一组IC(例如,芯片集)。本公开内容中描述了各种组件、模块或单元,以强调被配置为执行所公开的技术的设备的功能方面,但不一定需要由不同的硬件单元来实现。相反,如上所述,各个单元可以组合在编解码器硬件单元中,或者通过协作的硬件单元集合(其包括如上所述的一个或多个处理器)结合适当的软件和/或固件来提供。

[0211] 描述了各个示例。这些和其它示例落入所附权利要求的保护范围之内。

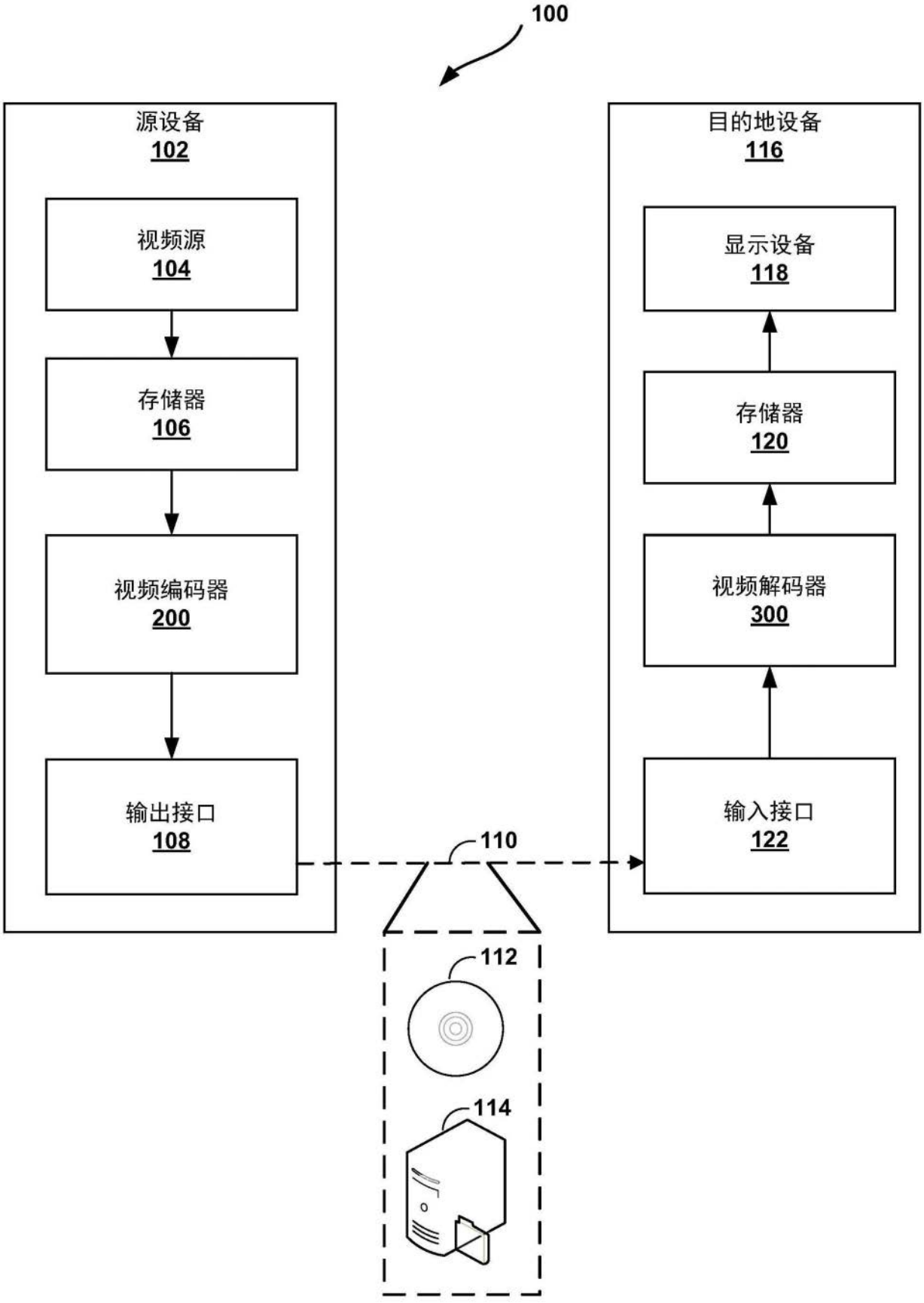

100
源设备
102
视频源
104
存储器
106
视频编码器
200
输出接口
108
目的地设备
116
显示设备
118
存储器
120
视频解码器
300
输入接口
122
110
112
114

图1

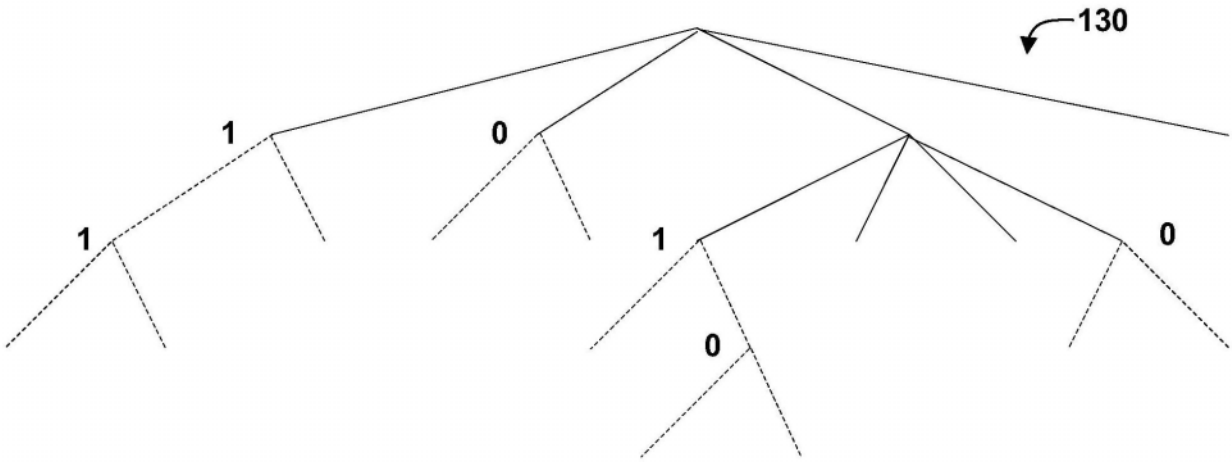


图2A

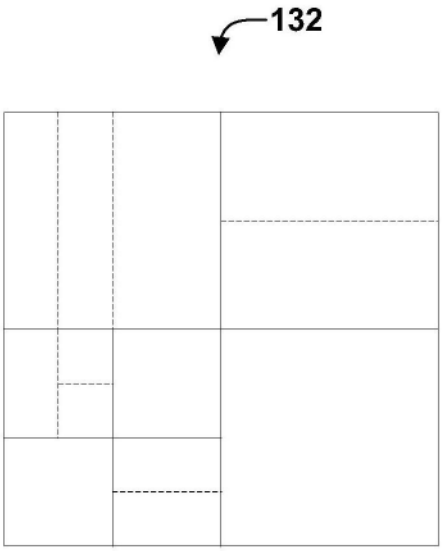


图2B

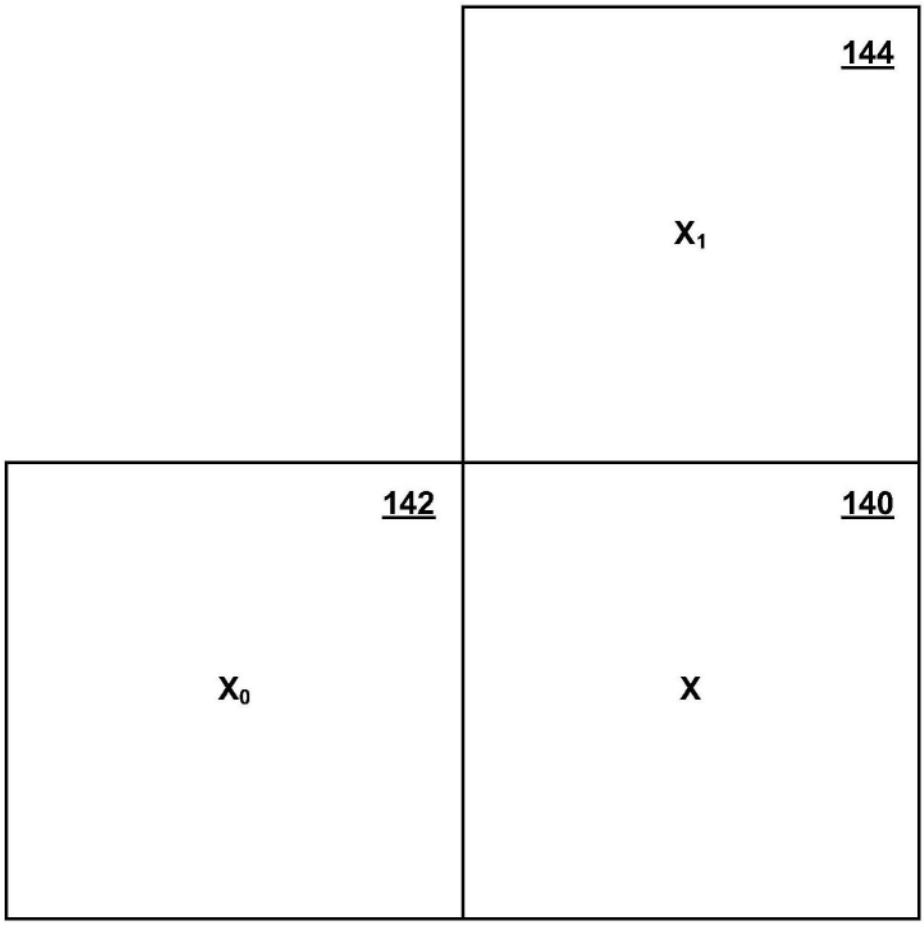


图3

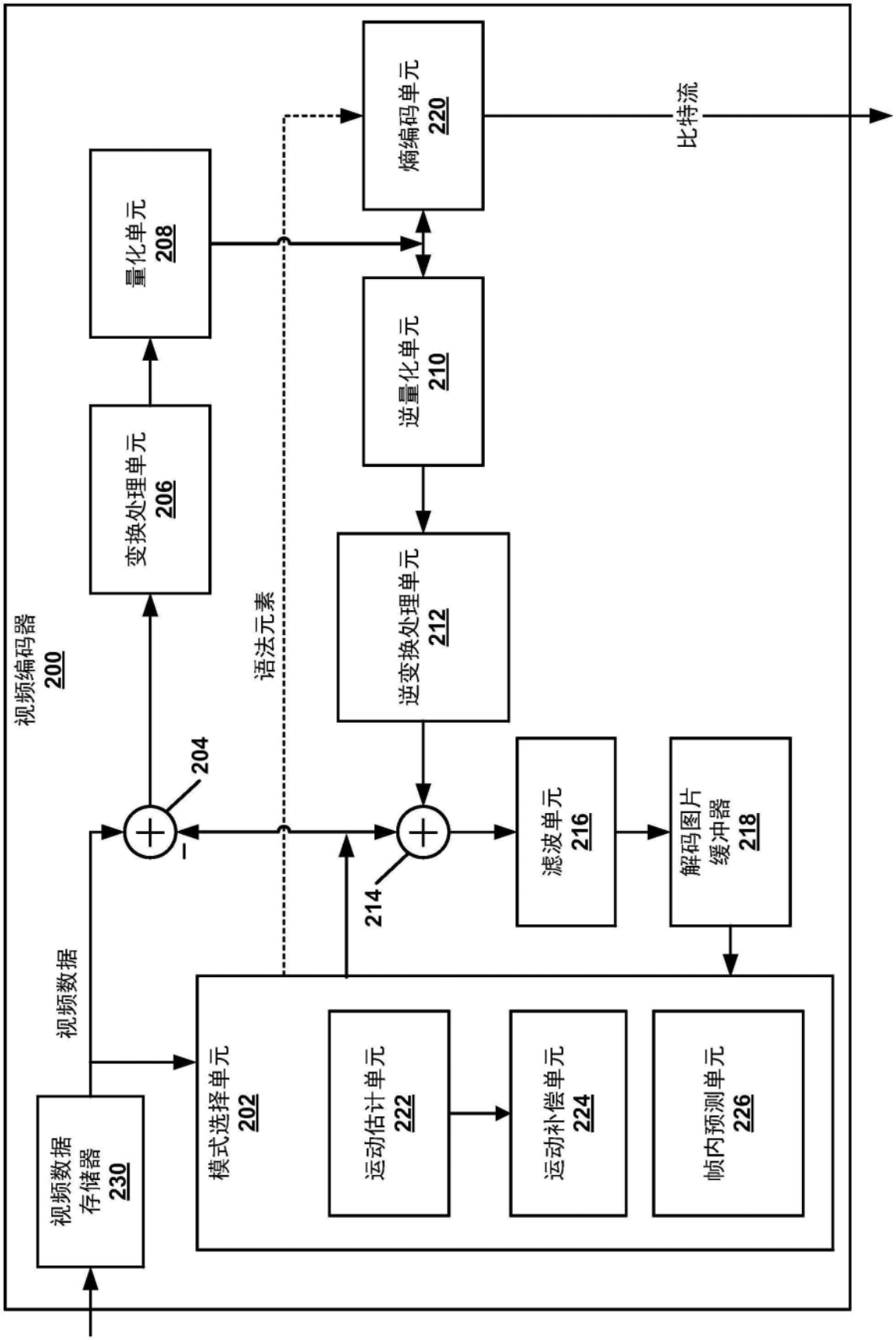

视频编码器
200
视频数据存储器
230
模式选择单元
202
运动估计单元
222
运动补偿单元
224
帧内预测单元
226
视频数据
204
变换处理单元
206
量化单元
208
逆量化单元
210
逆变换处理单元
212
214
滤波单元
216
解码图片缓冲器
218
语法元素
熵编码单元
220
比特流

图4

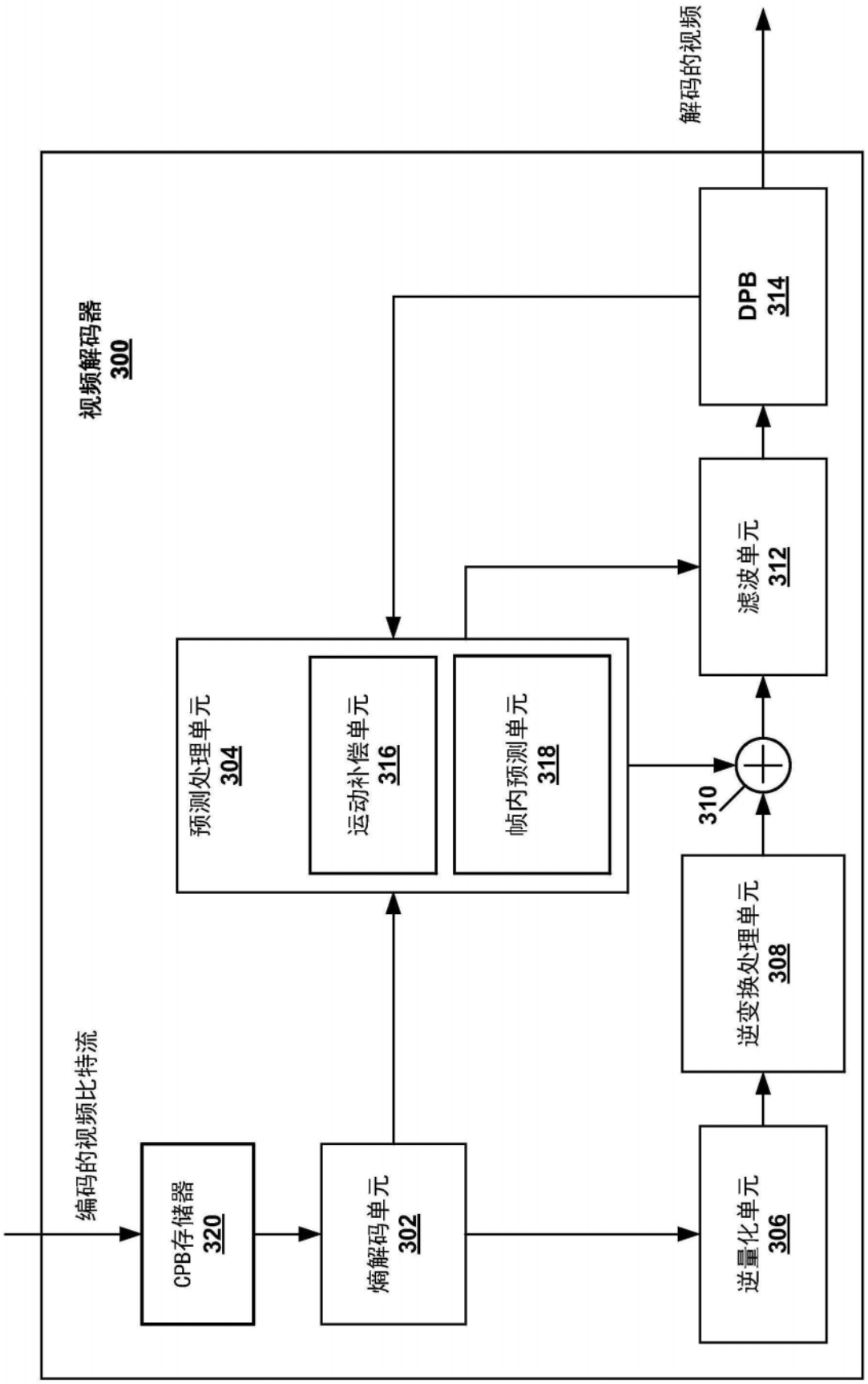

视频解码器
300
编码的视频比特流
CPB存储器
320
熵解码单元
302
预测处理单元
304
运动补偿单元
316
帧内预测单元
318
逆量化单元
306
逆变换处理单元
308
310
滤波单元
312
DPB
314
解码的视频

图5

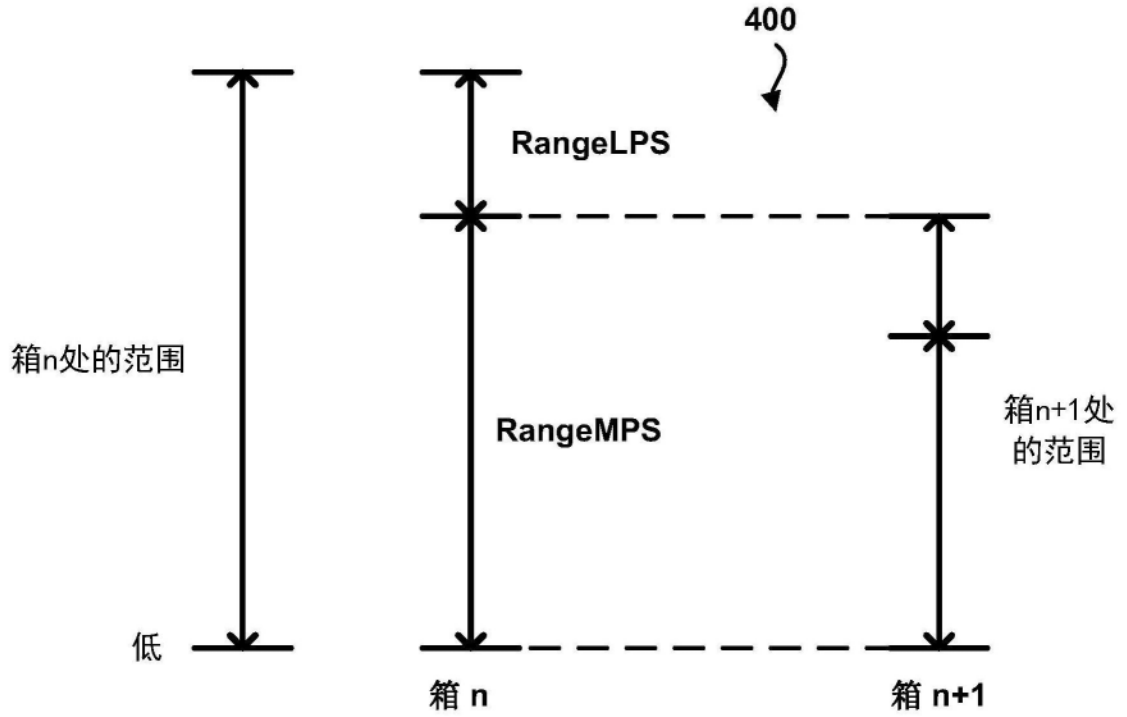

400
RangeLPS
箱n处的范围
RangeMPS
箱n+1处
的范围
低
箱 n
箱 n+1

图6A

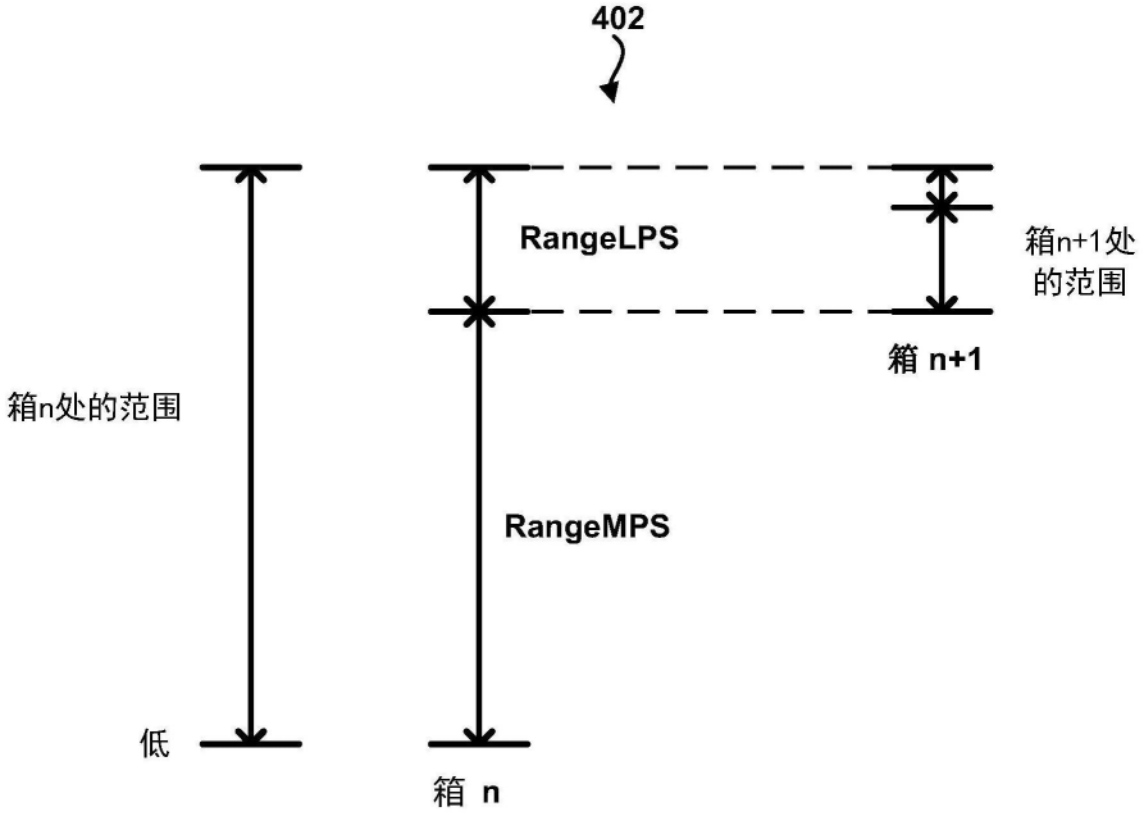

402
RangeLPS
箱n+1处
的范围
箱 n+1
箱n处的范围
RangeMPS
低
箱 n

图6B

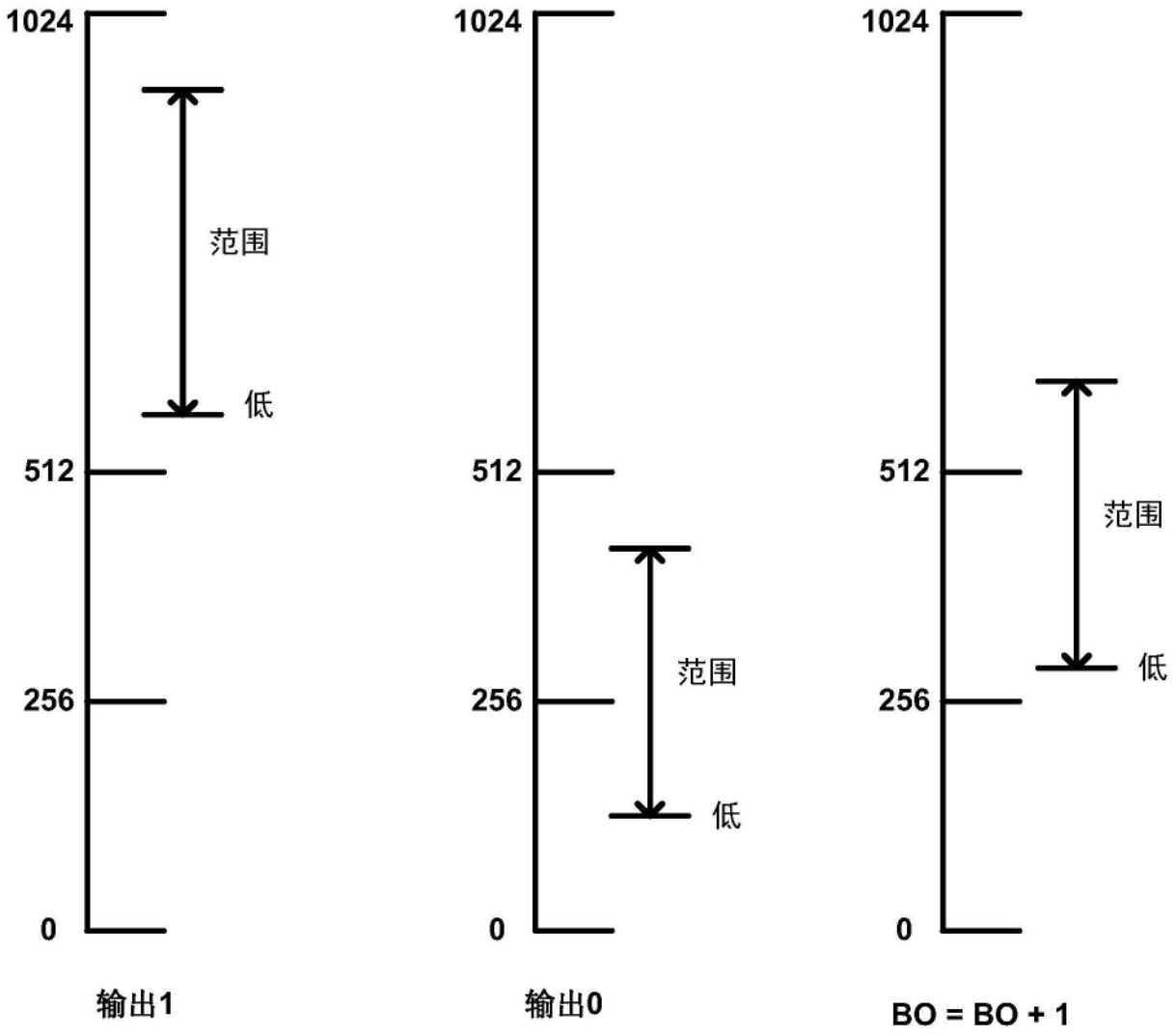

1024
范围
低
512
256
0
输出1
1024
512
范围
256
低
0
输出0
1024
512
范围
低
256
0
BO = BO + 1

图7

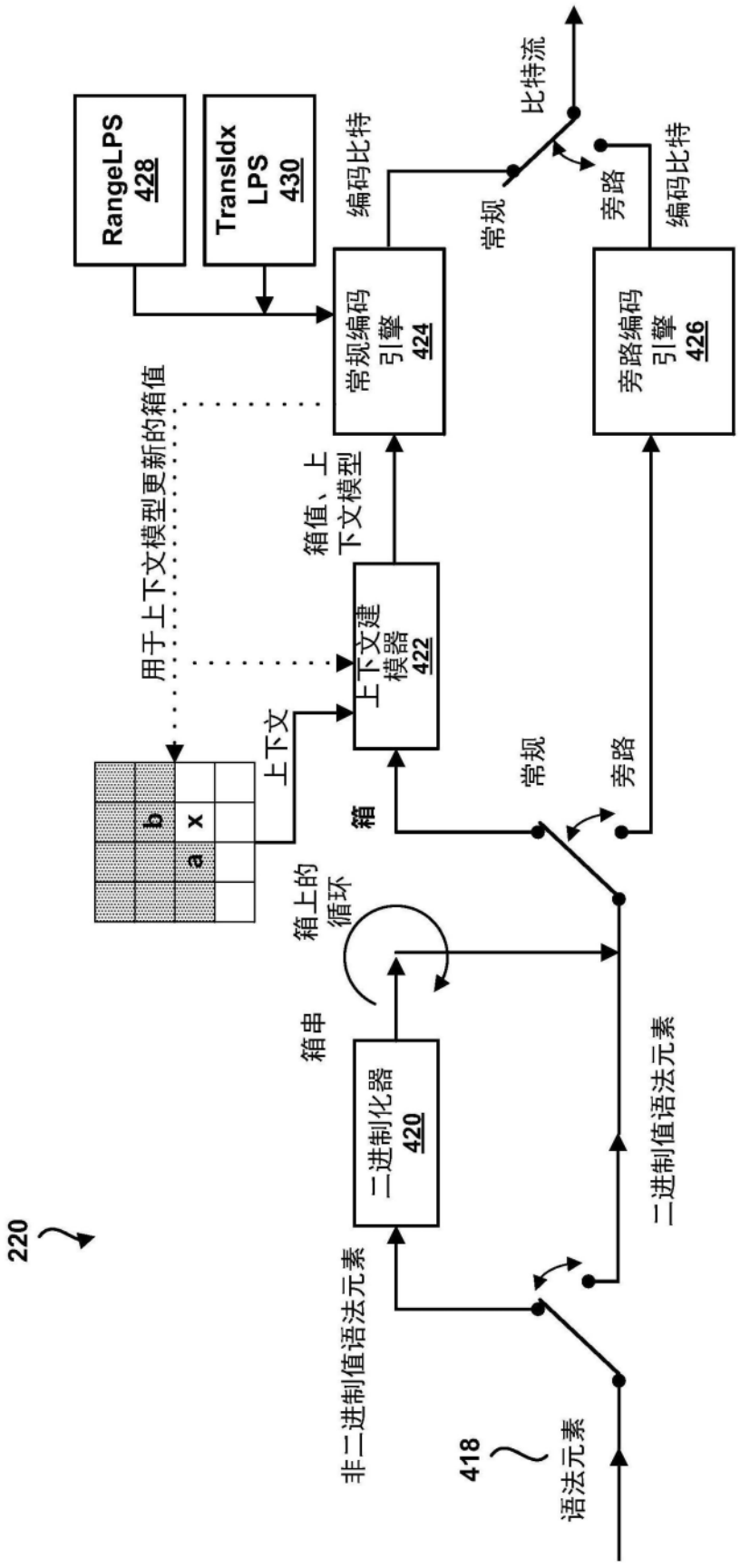

220
418
语法元素
非二进制值语法元素
二进制化器
420
二进制值语法元素
箱串
箱上的
循环
箱
常规
旁路
上下文
上下文建
模器
422
用于上下文模型更新的箱值
箱值、上
下文模型
常规编码
引擎
424
旁路编码
引擎
426
RangeLPS
428
TransIdx
LPS
430
编码比特
编码比特
常规
旁路
比特流
a
b
x

图8

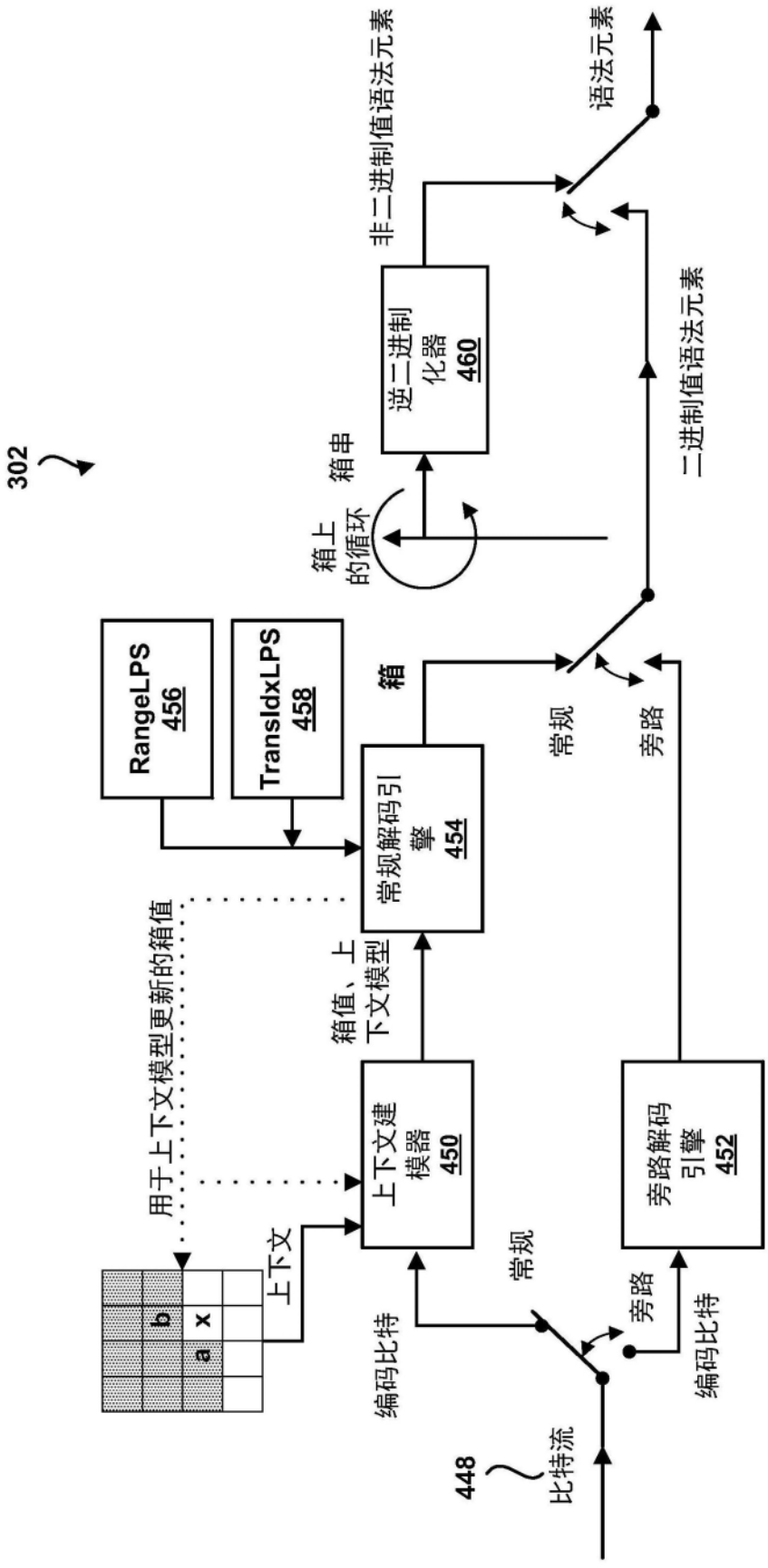

302
448
比特流
编码比特
上下文
用于上下文模型更新的箱值
a
b
x
上下文建模器 450
箱值、上下文模型
常规解码引擎 454
RangeLPS 456
TransIdxLPS 458
箱
常规
旁路
旁路解码引擎 452
编码比特
常规
旁路
箱上的循环
箱串
逆二进制化器 460
非二进制值语法元素
语法元素
二进制值语法元素

图9

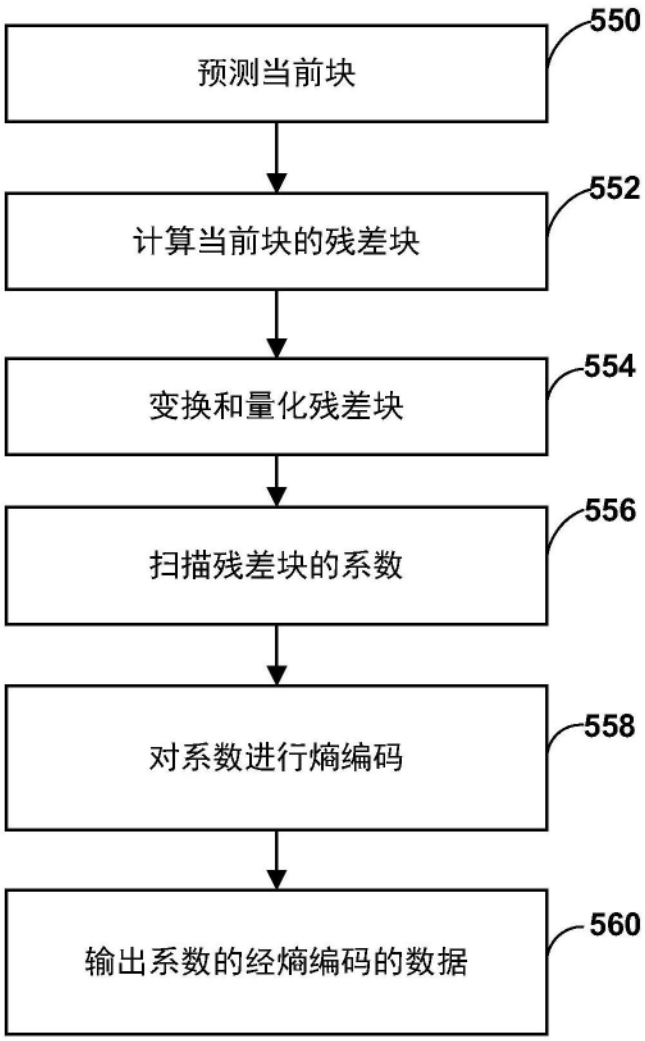

550
预测当前块
552
计算当前块的残差块
554
变换和量化残差块
556
扫描残差块的系数
558
对系数进行熵编码
560
输出系数的经熵编码的数据

图10

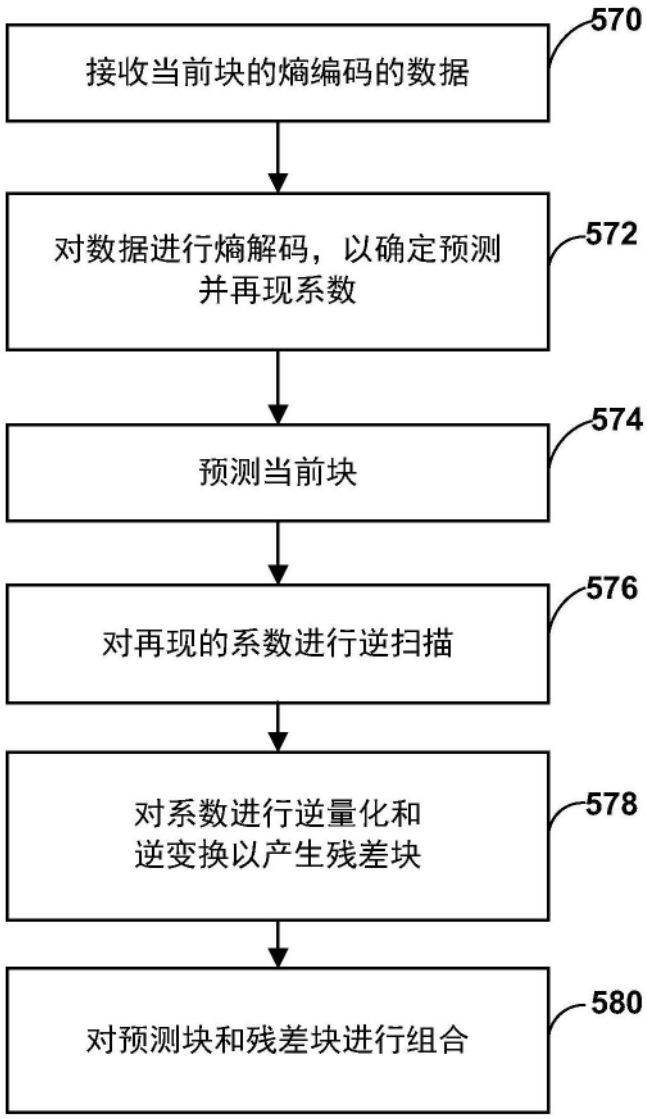

570
接收当前块的熵编码的数据
572
对数据进行熵解码，以确定预测并再现系数
574
预测当前块
576
对再现的系数进行逆扫描
578
对系数进行逆量化和逆变换以产生残差块
580
对预测块和残差块进行组合

图11

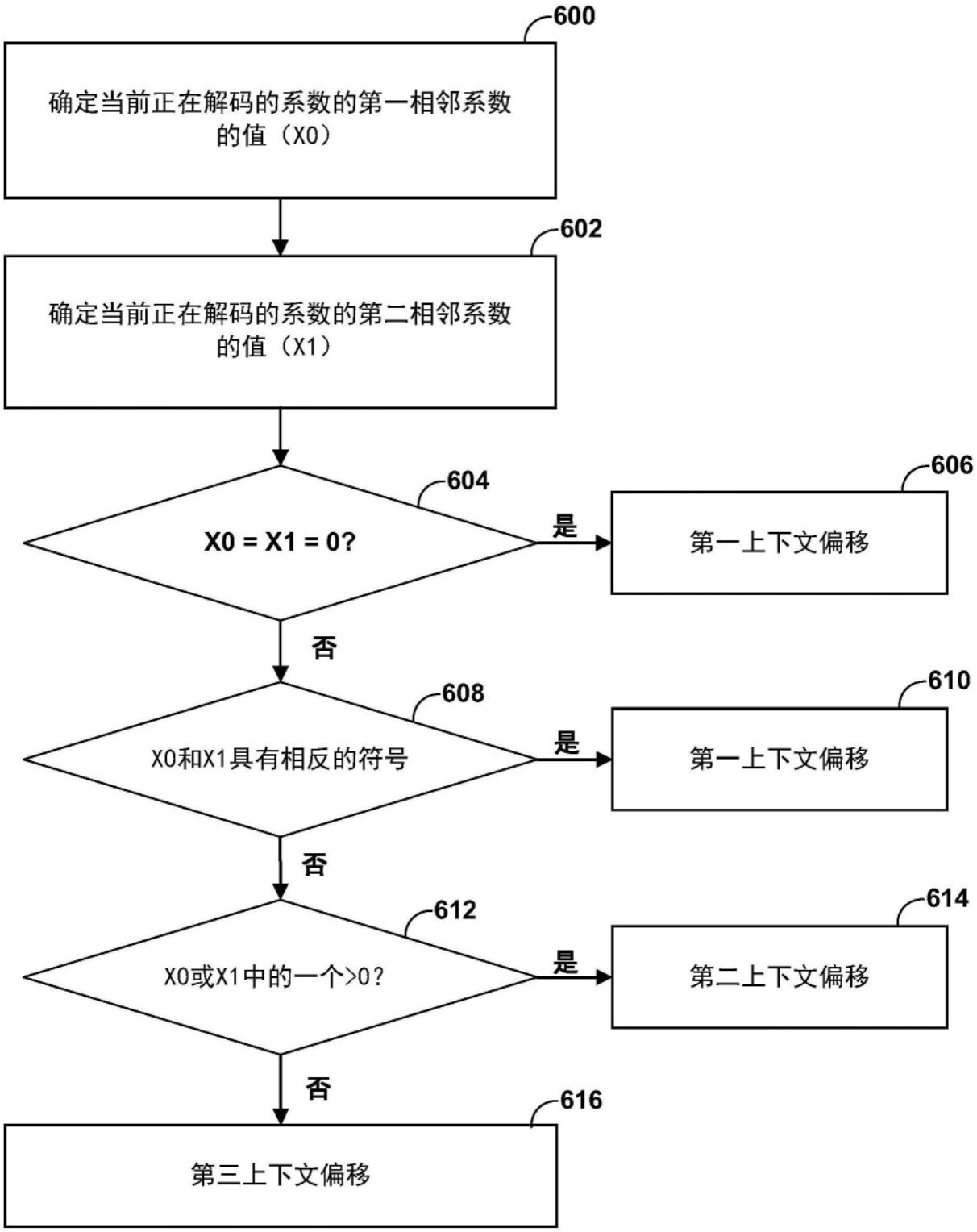

600
确定当前正在解码的系数的第一相邻系数的值（X0）
602
确定当前正在解码的系数的第二相邻系数的值（X1）
604
X0 = X1 = 0?
是
606
第一上下文偏移
否
608
X0和X1具有相反的符号
是
610
第一上下文偏移
否
612
X0或X1中的一个>0?
是
614
第二上下文偏移
否
616
第三上下文偏移

图12