(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 115767107 A
(43)申请公布日 2023.03.07

(21)申请号 202211573846.5

(22)申请日 2020.03.24

(66)本国优先权数据
PCT/CN2019/079395 2019.03.24 CN

(62)分案原申请数据
202080023977.9 2020.03.24

(71)申请人 北京字节跳动网络技术有限公司
地址 100041 北京市石景山区实兴大街30号院3号楼2层B-0035房间
申请人 字节跳动有限公司

(72)发明人 张莉 张凯 刘鸿彬 王悦

(74)专利代理机构 北京市柳沈律师事务所 11105
专利代理师 张亮

(51)Int.Cl.
H04N 19/80(2014.01)
H04N 19/117(2014.01)
H04N 19/42(2014.01)

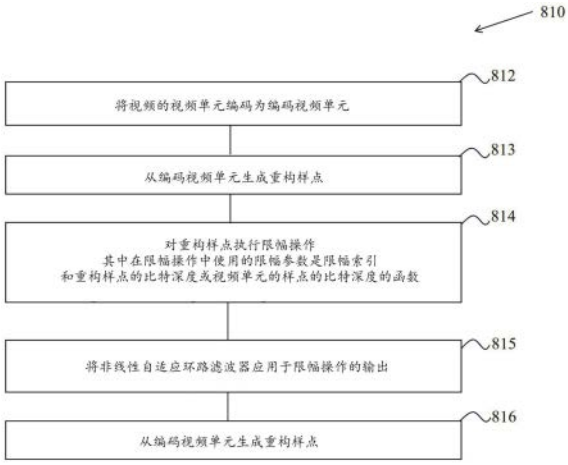
权利要求书5页 说明书24页 附图9页

(54)发明名称

视频处理中的非线性自适应环路滤波

(57)摘要

描述了视频处理中的非线性自适应环路滤波。在一个示例性方面，一种处理视频数据的方法，包括：确定非线性自适应环路滤波操作应用于视频的当前视频区域；为当前视频区域推导至少一个第一滤波索引并基于至少一个第一滤波索引推导一个或多个第一滤波参数；基于至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；基于一个或多个第一限幅参数来执行限幅操作，限幅操作是非线性自适应环路滤波操作的一部分，其中一个或多个第一限幅参数被用于限幅当前视频区域的当前样点值的参考样点值与在应用一个或多个第一滤波参数之前的当前样点值之间的差；以及基于非线性自适应环路滤波操作来执行当前视频区域与视频的比特流之间的转换。



CN 115767107 A

1.一种处理视频数据的方法,包括:

确定非线性自适应环路滤波操作应用于视频的当前视频区域;

为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数;

基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数;

基于所述一个或多个第一限幅参数来执行限幅操作,所述限幅操作是所述非线性自适应环路滤波操作的一部分,其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差;以及

基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

2.根据权利要求1所述的方法,其中,所述限幅参数集是基于与所述比特流中的字段相关联的比特深度和限幅索引来构建的。

3.根据权利要求2所述的方法,其中,对于所述限幅参数集,在特定限幅指数的给定值下,不同的限幅参数值被用于不同的比特深度。

4.根据权利要求2所述的方法,其中,与给定比特深度的第一限幅索引相对应的第一限幅参数值是基于与另一比特深度的第一限幅索引相对应的第二限幅参数值来推导的;并且

其中,所述第一限幅参数值是基于对所述第二参数限幅值应用移位操作来推导的。

5.根据权利要求1所述的方法,其中,所述至少一个第一滤波索引是基于不同方向的多个样点差来推导的,

其中,所述当前视频区域被分为多个M*M视频区域,并且对于每个M*M视频子区域推导不同方向的多个样点差,并且其中M等于2或4,并且其中所述当前视频区域是编解码树块或条带,并且

其中,不同方向的多个样点差是基于1:N子采样率来推导的,其中N大于1。

6.根据权利要求1所述的方法,其中,所述一个或多个第一滤波参数和所述一个或多个第一限幅参数是基于相同的自适应参数集标识来推导的。

7.根据权利要求1所述的方法,其中,所述一个或多个第一限幅参数是基于包括在所述比特流中的第一边信息和第一自适应参数集标识来推导的,并且

其中,所述一个或多个第一滤波参数是基于包括在所述比特流中的第二边信息和第一自适应参数集标识来推导的。

8.根据权利要求7所述的方法,其中,在对所述一个或多个第一滤波参数启用根据与所述第一自适应参数集标识相关联的数据信息的预测的情况下,与所述第一自适应参数集标识相关联的数据信息也被用于预测所述一个或多个第一限幅参数。

9.根据权利要求1所述的方法,其中,所述当前视频区域是编解码树单元、编解码树块或条带。

10.根据权利要求1所述的方法,其中,所述一个或多个第一限幅参数是进一步基于所述当前视频区域的比特深度值来推导的。

11.根据权利要求1所述的方法,其中,所述转换包括将所述当前视频区域编码成所述比特流。

12.根据权利要求1所述的方法,其中,所述转换包括从所述比特流解码所述当前视频区域。

13.一种用于处理视频数据的装置,包括处理器和其上具有指令的非暂时性存储器,其中所述指令在由所述处理器执行时使得所述处理器:

确定非线性自适应环路滤波操作应用于视频的当前视频区域;

为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数;

基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数;

基于所述一个或多个第一限幅参数来执行限幅操作,所述限幅操作是所述非线性自适应环路滤波操作的一部分,其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差;以及

基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

14.根据权利要求13所述的装置,其中,所述限幅参数集是基于与所述比特流中的字段相关联的比特深度和限幅索引来构建的,

其中,对于所述限幅参数集,在特定限幅指数的给定值下,不同的限幅参数值被用于不同的比特深度,

其中,与给定比特深度的第一限幅索引相对应的第一限幅参数值是基于与另一比特深度的第一限幅索引相对应的第二限幅参数值来推导的,并且

其中,所述第一限幅参数值是基于对所述第二参数限幅值应用移位操作来推导的。

15.根据权利要求13所述的装置,其中,所述至少一个第一滤波索引是基于不同方向的多个样点差来推导的,

其中,所述当前视频区域被分为多个M*M视频区域,并且对于每个M*M视频子区域推导不同方向的多个样点差,并且其中M等于2或4,并且其中所述当前视频区域是编解码树块或条带,并且

其中,不同方向的多个样点差是基于1:N子采样率来推导的,其中N大于1。

16.根据权利要求13所述的装置,其中,所述一个或多个第一滤波参数和所述一个或多个第一限幅参数是基于相同的自适应参数集标识来推导的,

其中,所述一个或多个第一限幅参数是基于包括在所述比特流中的第一边信息和第一自适应参数集标识来推导的,

其中,所述一个或多个第一滤波参数是基于包括在所述比特流中的第二边信息和第一自适应参数集标识来推导的,

其中,在对所述一个或多个第一滤波参数启用根据与所述第一自适应参数集标识相关联的数据信息的预测的情况下,与所述第一自适应参数集标识相关联的数据信息也被用于预测所述一个或多个第一限幅参数,

其中,所述当前视频区域是编解码树单元、编解码树块或条带,并且

其中,所述一个或多个第一限幅参数是进一步基于所述当前视频区域的比特深度值来推导的。

17.一种存储指令的非暂时性计算机可读存储介质,所述指令使得处理器:

确定非线性自适应环路滤波操作应用于视频的当前视频区域;

为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数;

基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数;

基于所述一个或多个第一限幅参数来执行限幅操作,所述限幅操作是所述非线性自适应环路滤波操作的一部分,其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差;以及

基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

18.根据权利要求17所述的非暂时性计算机可读存储介质,其中,所述限幅参数集是基于与所述比特流中的字段相关联的比特深度和限幅索引来构建的,

其中,对于所述限幅参数集,在特定限幅指数的给定值下,不同的限幅参数值被用于不同的比特深度,

其中,与给定比特深度的第一限幅索引相对应的第一限幅参数值是基于与另一比特深度的第一限幅索引相对应的第二限幅参数值来推导的,

其中,所述第一限幅参数值是基于对所述第二参数限幅值应用移位操作来推导的,

其中,所述至少一个第一滤波索引是基于不同方向的多个样点差来推导的,

其中,所述当前视频区域被分为多个M*M视频区域,并且对于每个M*M视频子区域推导不同方向的多个样点差,并且其中M等于2或4,并且其中所述当前视频区域是编解码树块或条带,

其中,不同方向的多个样点差是基于1:N子采样率来推导的,其中N大于1,

其中,所述一个或多个第一滤波参数和所述一个或多个第一限幅参数是基于相同的自适应参数集标识来推导的,

其中,所述一个或多个第一限幅参数是基于包括在所述比特流中的第一边信息和第一自适应参数集标识来推导的,

其中,所述一个或多个第一滤波参数是基于包括在所述比特流中的第二边信息和第一自适应参数集标识来推导的,

其中,在对所述一个或多个第一滤波参数启用根据与所述第一自适应参数集标识相关联的数据信息的预测的情况下,与所述第一自适应参数集标识相关联的数据信息也被用于预测所述一个或多个第一限幅参数,

其中,所述当前视频区域是编解码树单元、编解码树块或条带,并且

其中,所述一个或多个第一限幅参数是进一步基于所述当前视频区域的比特深度值来推导的。

19.一种存储由视频处理装置执行的方法生成的视频的比特流的非暂时性计算机可读记录介质,其中所述方法包括:

确定非线性自适应环路滤波操作应用于视频的当前视频区域;

为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引

推导一个或多个第一滤波参数；

基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；

基于所述一个或多个第一限幅参数来执行限幅操作，所述限幅操作是所述非线性自适应环路滤波操作的一部分，其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差；以及

基于所述限幅操作来生成所述视频的比特流。

20.根据权利要求19所述的非暂时性计算机可读记录介质，其中，所述限幅参数集是基于与所述比特流中的字段相关联的比特深度和限幅索引来构建的，

其中，对于所述限幅参数集，在特定限幅指数的给定值下，不同的限幅参数值被用于不同的比特深度，

其中，与给定比特深度的第一限幅索引相对应的第一限幅参数值是基于与另一比特深度的第一限幅索引相对应的第二限幅参数值来推导的，

其中，所述第一限幅参数值是基于对所述第二参数限幅值应用移位操作来推导的，

其中，所述至少一个第一滤波索引是基于不同方向的多个样点差来推导的，

其中，所述当前视频区域被分为多个M*M视频区域，并且对于每个M*M视频子区域推导不同方向的多个样点差，并且其中M等于2或4，并且其中所述当前视频区域是编解码树块或条带，

其中，不同方向的多个样点差是基于1:N子采样率来推导的，其中N大于1，

其中，所述一个或多个第一滤波参数和所述一个或多个第一限幅参数是基于相同的自适应参数集标识来推导的，

其中，所述一个或多个第一限幅参数是基于包括在所述比特流中的第一边信息和第一自适应参数集标识来推导的，

其中，所述一个或多个第一滤波参数是基于包括在所述比特流中的第二边信息和第一自适应参数集标识来推导的，

其中，在对所述一个或多个第一滤波参数启用根据与所述第一自适应参数集标识相关联的数据信息的预测的情况下，与所述第一自适应参数集标识相关联的数据信息也被用于预测所述一个或多个第一限幅参数，

其中，所述当前视频区域是编解码树单元、编解码树块或条带，并且

其中，所述一个或多个第一限幅参数是进一步基于所述当前视频区域的比特深度值来推导的。

21.一种用于存储视频的比特流的方法，包括：

确定非线性自适应环路滤波操作应用于视频的当前视频区域；

为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数；

基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；

基于所述一个或多个第一限幅参数来执行限幅操作，所述限幅操作是所述非线性自适应环路滤波操作的一部分，其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值

之间的差；

基于所述限幅操作来生成所述视频的比特流；以及

将所述比特流存储到非暂时性计算机可读记录介质中。

视频处理中的非线性自适应环路滤波

[0001] 相关应用的交叉引用

[0002] 本申请是申请日为2020年3月24日、申请号为202080023977.9的中国发明专利申请的分案申请,其根据巴黎公约适用的专利法和/或规则,本申请旨在及时要求2019年3月24日提交的国际专利申请号PCT/CN2019/079395的优先权和利益。出于法律规定的所有目的,上述申请的全部公开内容通过引用而并入作为本申请公开内容的一部分。

技术领域

[0003] 本专利文档涉及视频处理技术、设备和系统。

背景技术

[0004] 尽管在视频压缩方面有所进步,但是数字视频仍然在互联网和其他数字通信网络上占用最大的带宽使用。随着能够接收和显示视频的联网用户设备的数量增加,预计数字视频使用的带宽需求将继续增长。

发明内容

[0005] 描述了与数字视频处理相关的设备、系统和方法,例如,与用于视频处理的非线性自适应环路滤波相关的设备、系统和方法。所描述的方法可以应用于现有的视频编解码标准(例如,高效视频编解码(High Efficiency Video Coding,HEVC))和未来的视频编解码标准(例如,通用视频编解码(Versatile Video Coding,VVC))或编解码器。

[0006] 在一个代表性方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:将视频的视频单元编码为编码视频单元;从编码视频单元生成重构样点;对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的样点的比特深度的函数;将非线性自适应环路滤波器应用于限幅操作的输出;以及使用编码视频单元生成视频的编解码表示。

[0007] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:解析视频的编解码表示以得到表示视频的视频单元的编码视频单元;从编码视频单元生成视频单元的重构样点;对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的比特深度的函数;以及将非线性自适应环路滤波器应用于限幅操作的输出,以生成最终的解码视频单元。

[0008] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且其中,该确定是基于视频和/或视频区域和/或视频单元的编解码信息的。

[0009] 在另一方面,所公开的技术可用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并

且其中,限幅参数是色彩表示格式的函数。

[0010] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且其中,限幅参数取决于是否基于在第一域和第二域中视频单元的表示和/或色度视频单元的缩放色度残差来应用环路整形(ILR)对视频单元进行重构。

[0011] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括提供限幅参数的第一边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波;其中,第一边信息与指示在非线性自适应环路滤波器中使用的滤波器系数的第二边信息一起被信令通知。

[0012] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括指示多组限幅参数的边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波。

[0013] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;其中,编解码表示包括提供一个或多个限幅参数的边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的色度视频单元的重构进行滤波,其中,一个或多个限幅参数取决于色彩格式。

[0014] 在另一方面,所公开的技术可以用于提供一种用于视频处理的方法。该方法包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括提供限幅参数的边信息,该限幅参数用于使用自适应环路滤波器对视频区域的视频单元的重构进行滤波,其中,该执行包括通过将限幅操作应用于视频区域级别的样点差来生成滤波视频单元。

[0015] 在另一方面,所公开的技术可以用于提供一种处理视频数据的方法,包括:确定非线性自适应环路滤波操作应用于视频的当前视频区域;为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数;基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数;基于所述一个或多个第一限幅参数来执行限幅操作,所述限幅操作是所述非线性自适应环路滤波操作的一部分,其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差;以及基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

[0016] 在另一方面,所公开的技术可以用于提供一种用于处理视频数据的装置,包括处理器和其上具有指令的非暂时性存储器,其中所述指令在由所述处理器执行时使得所述处理器:确定非线性自适应环路滤波操作应用于视频的当前视频区域;为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数;基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数;基于所述一个或多个第一限幅参数来执行限幅操作,所述限幅操作是所述非线性自适应环路滤波操作的一部分,其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前

样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差；以及基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

[0017] 在另一方面，所公开的技术可以用于提供一种存储指令的非暂时性计算机可读存储介质，所述指令使得处理器：确定非线性自适应环路滤波操作应用于视频的当前视频区域；为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数；基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；基于所述一个或多个第一限幅参数来执行限幅操作，所述限幅操作是所述非线性自适应环路滤波操作的一部分，其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差；以及基于所述非线性自适应环路滤波操作来执行所述当前视频区域与所述视频的比特流之间的转换。

[0018] 在另一方面，所公开的技术可以用于提供一种存储由视频处理装置执行的方法生成的视频的比特流的非暂时性计算机可读记录介质，其中所述方法包括：确定非线性自适应环路滤波操作应用于视频的当前视频区域；为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数；基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；基于所述一个或多个第一限幅参数来执行限幅操作，所述限幅操作是所述非线性自适应环路滤波操作的一部分，其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差；以及基于所述限幅操作来生成所述视频的比特流。

[0019] 在另一方面，所公开的技术可以用于提供一种用于存储视频的比特流的方法，包括：确定非线性自适应环路滤波操作应用于视频的当前视频区域；为所述当前视频区域推导至少一个第一滤波索引并基于所述至少一个第一滤波索引推导一个或多个第一滤波参数；基于所述至少一个第一滤波索引从限幅参数集推导一个或多个第一限幅参数；基于所述一个或多个第一限幅参数来执行限幅操作，所述限幅操作是所述非线性自适应环路滤波操作的一部分，其中所述一个或多个第一限幅参数被用于限幅所述当前视频区域的当前样点值的参考样点值与在应用所述一个或多个第一滤波参数之前的当前样点值之间的差；基于所述限幅操作来生成所述视频的比特流；以及将所述比特流存储到非暂时性计算机可读记录介质中。

[0020] 在又一代表性方面，上述方法以处理器可执行代码的形式来实现，并且存储在计算机可读程序介质中。

[0021] 在又一代表性方面，公开了一种被配置或可操作来执行上述方法的设备。该设备可以包括被编程来实施该方法的处理器。

[0022] 在又一代表性方面，上述公开的方法中的一个或多个可以是编码器侧实施或解码器侧实施。

[0023] 在附图、说明书和权利要求书中更详细地描述了所公开技术的上述和其他方面和特征。

附图说明

[0024] 图1示出了用于视频编解码的编码器框图的示例。

[0025] 图2A、图2B和图2C示出了基于几何变换的自适应环路滤波器(geometry transformation-based adaptive loop filter,GALF)滤波器形状的示例。

[0026] 图3示出了GALF编码器决定的流程图的示例。

[0027] 图4A至图4D示出了用于自适应环路滤波器(adaptive loop filter,ALF)分类的示例子采样拉普拉斯计算。

[0028] 图5示出了亮度滤波器形状的示例。

[0029] 图6示出了宽视频图形阵列(Wide Video Graphic Array,WVGA)序列的区域划分的示例。

[0030] 图7示出了具有整形的解码流程的示例性流程图。

[0031] 图8A至图8C示出了根据所公开技术的一些实施方式的视频处理的示例方法的流程图。

[0032] 图9示出了根据所公开技术的一些实施方式的视频处理的示例方法的流程图。

[0033] 图10A和图10B是用于实施本文档中描述的视频处理技术的硬件平台的示例的框图。

具体实施方式

[0034] 由于对更高分辨率视频的需求日益增长,视频编解码方法和技术在现代技术中无处不在。视频编解码器通常包括对数字视频进行压缩或解压缩的电子电路或软件,并且正在不断改进以提供更高的编解码效率。视频编解码器将未压缩的视频转换为压缩格式,反之亦然。视频质量、用于表示视频的数据量(由比特率所确定)、编码和解码算法的复杂性、对数据丢失和错误的敏感性、编辑的简易性、随机存取和端到端延迟(延时)之间存在复杂的关系。压缩格式通常符合标准视频压缩规范,例如,高效视频编解码(HEVC)标准(也称为H.265或MPEG-H第2部分)、待定稿的通用视频编解码(VVC)标准或其他当前和/或未来的视频编解码标准。

[0035] 在一些实施例中,使用称为联合勘探模型(Joint Exploration Model,JEM)的参考软件来探索未来的视频编解码技术。在JEM中,基于子块的预测被采用于若干编解码工具中,诸如仿射预测、可选时域运动矢量预测(alternative temporal motion vector prediction,ATMVP)、时空运动矢量预测(spatial-temporal motion vector prediction,STMVP)、双向光流(directional optical flow,BIO)、帧速率上转换(Frame-Rate Up Conversion,FRUC)、局部自适应运动矢量分辨率(Locally Adaptive Motion Vector Resolution,LAMVR)、重叠块运动补偿(Overlapped Block Motion Compensation,OBMC)、局部照明补偿(Local Illumination Compensation,LIC)和解码器侧运动矢量细化(Decoder-side Motion Vector Refinement,DMVR)。

[0036] 所公开的技术的实施例可以应用于现有的视频编解码标准(例如,HEVC、H.265)和未来的标准,以提高运行时性能。在本文档中使用章节标题来提高描述的可读性,并且不以任何方式将讨论或实施例(和/或实施方式)仅限制于各个章节。

[0037] 1色彩空间和色度子采样的示例

[0038] 色彩空间,也称为色彩模型(或色彩系统),是一个抽象的数学模型,它简单地将色彩的范围描述为数字的元组,通常为3个或4个值或色彩分量(例如,RGB)。基本上来说,色彩空间是坐标系和子空间的细化。

[0039] 对于视频压缩,最常用的色彩空间是YCbCr和RGB。

[0040] YCbCr、Y'CbCr、或Y Pb/Cb Pr/Cr,也称为YBCCR或Y'CBCR,是一个色彩空间的家族,其在视频和数字摄影系统中用作色彩图像管线的一部分。Y'是亮度分量,而CB和CR是蓝差和红差色度分量。Y'(具有prime)不同于Y,Y是辉亮度(luminance),其意味着光强是基于伽马校正的RGB原色进行非线性编码的。

[0041] 色度子采样是通过对色度信息实施比对亮度信息更少分辨率来对图像进行编码的实践,这利用了人类视觉系统对色差的敏感度低于对辉亮度的敏感度。

[0042] 1.1 4:4:4色彩格式

[0043] 三个Y'CBcr分量中的每一个都具有相同的采样率,因此不存在色度子采样。这种方案有时用于高端胶片扫描仪和电影后期制作。

[0044] 1.2 4:2:2色彩格式

[0045] 两个色度分量以亮度的采样率的一半进行采样,例如,水平色度分辨率减半。这将未压缩视频信号的带宽减少了三分之一,几乎没有视觉差异。

[0046] 1.3 4:2:0色彩格式

[0047] 在4:2:0模式下,与4:1:1相比,水平采样增加了一倍,但在该方案中,Cb和Cr通道仅在每条交替线上采样,因此垂直分辨率减半。因此,数据速率是相同的。Cb和Cr在水平和垂直方向上均以2为因子进行子采样。有三种4:2:0方案的变体,具有不同的水平和垂直选址。

[0048] -在MPEG-2中,Cb和Cr是水平共址(cosite)的。Cb和Cr在垂直方向上位于像素之间(位于间隙中)。

[0049] -在JPEG/JFIF、H.261和MPEG-1中,Cb和Cr位于间隙中,在交替的亮度样点之间的中间。

[0050] -在4:2:0DV中,Cb和Cr在水平方向上共址。在垂直方向上,它们在交替线上共址。

[0051] 2典型视频编解码器的编解码流程的示例

[0052] 图1示出了VVC的编码器框图的示例。如图1所示,视频编码的流程可以从经历帧内预测和/或运动估计/运动补偿(motion estimation/motion compensation,ME/MC)的输入视频开始。这些操作使用来自视频的先前编解码部分的重构副本的反馈。帧内预测和/或ME/MC的输出通过变换操作(transform,T)进行差分处理,随后是量化操作(quantization,Q),量化操作被熵编解码成输出编解码表示。在反馈环路中,编码表示(Q块的输出)可以经历逆量化操作(inverse quantization,IQ),随后是逆变换(inverse transform,IT)操作,以生成编码视频块的重构样点。

[0053] 可以通过各种"环路"滤波操作来进一步处理重构样点,以生成用于进一步编码的参考样点或参考块或参考图片。环路滤波处理链包含三个环路滤波块:去块滤波器(deblocking filter,DF)、样点自适应偏移(sample adaptive offset,SAO)和ALF。不同于使用预定义滤波器的DF,SAO和ALF利用当前图片的原始样点,分别通过添加偏移和通过应用有限脉冲响应(finite impulse response,FIR)滤波器来减少原始样点和重构样点之间

的均方误差，其中编解码的边信息信令通知偏移和滤波器系数。ALF位于每个图片的最后处理阶段，并且可以被看作是尝试捕捉和修复先前阶段产生的伪像的工具。

[0054] 在解码器侧，以相反的顺序执行若干编码操作，以生成解码和重构的视频样点。例如，参考图1，解码器可以解析作为熵编解码的输出的编解码表示，并且获得视频的编码单元或块，然后将视频的编码单元或块经过逆量化(IQ)操作和逆变换(IT)操作以生成视频单元的重构样点。最终的解码视频单元可以通过应用如上关于视频编码器的反馈环路所述的环路滤波操作来生成。

[0055] 3JEM中基于几何变换的自适应环路滤波器的示例

[0056] 在JEM中，应用基于几何变换的自适应环路滤波器(GALF)与基于块的滤波器自适应。对于亮度分量，基于局部梯度的方向和活动性，为每个2×2块选择25个滤波器当中的一个。

[0057] 3.1滤波器形状示例

[0058] 在JEM中，可以为亮度分量选择多达三种菱形滤波器形状(如图2A、图2B和图2C所示，分别为5×5菱形、7×7菱形和9×9菱形)。在图像级别信令通知索引，以指示用于亮度分量的滤波器形状。对于图片中的色度分量，总是使用5×5菱形。

[0059] 3.1.1块分类

[0060] 每个2×2块被分类为25类之一。分类索引C是基于其方向性D和活动性的量化值$\hat{A}$而推导出的，如下所示：

[0061] $$C = 5D + \hat{A} \qquad (1)$$

[0062] 为了计算D和$\hat{A}$，首先使用1-D拉普拉斯算子来计算水平、垂直和两个对角方向的梯度：

[0063] $$g_v = \sum_{k=i-2}^{i+3} \sum_{l=j-2}^{j+3} V_{k,l},\ V_{k,l} = |2R(k,l) - R(k,l-1) - R(k,l+1)| \qquad (2)$$

[0064] $$g_h = \sum_{k=i-2}^{i+3} \sum_{l=j-2}^{j+3} H_{k,l},\ H_{k,l} = |2R(k,l) - R(k-1,l) - R(k+1,l)| \qquad (3)$$

[0065] $$g_{d1} = \sum_{k=i-2}^{i+3} \sum_{l=j-3}^{j+3} D1_{k,l}, D1_{k,l} \qquad (4)$$
$$= |2R(k,l) - R(k-1,l-1) - R(k+1,l+1)|$$

[0066] $$g_{d2} = \sum_{k=i-2}^{i+3} \sum_{j=j-2}^{j+3} D2_{k,l},\ D2_{k,l} \qquad (5)$$
$$= |2R(k,l) - R(k-1,l+1) - R(k+1,l-1)|$$

[0067] 索引i和j是指2×2块中左上角样点的坐标，并且R(i,j)是指坐标(i,j)处的重构样点。

[0068] 然后水平和垂直方向梯度的D最大值和最小值被设置为：

[0069] $$g_{h,v}^{max} = max(g_h, g_v), \quad g_{h,v}^{min} = min(g_h, g_v), \qquad (6)$$

[0070] 并且两个对角方向的梯度的最大值和最小值被设置为：

[0071] $$g_{d0,d1}^{max} = max(g_{d0}, g_{d1}), \quad g_{d0,d1}^{min} = min(g_{d0}, g_{d1}), \qquad (7)$$

[0072] 为了推导出方向性D的值，将这些值相互比较，并且与两个阈值t_1和t_2进行比较：

[0073] 步骤1.如果$g_{h,v}^{max} \le t_1 \cdot g_{h,v}^{min}$和$g_{d0,d1}^{max} \le t_1 \cdot g_{d0,d1}^{min}$都为真，则将D设置为0。

[0074] 步骤2.如果$g_{h,v}^{max}/g_{h,v}^{min} > g_{d0,d1}^{max}/g_{d0,d1}^{min}$，则从步骤3继续；否则从步骤4继续。

[0075] 步骤3.如果$g_{h,v}^{max} > t_2 \cdot g_{h,v}^{min}$，则将D设置为2；否则将D设置为1。

[0076] 步骤4.如果$g_{d0,d1}^{max} > t_2 \cdot g_{d0,d1}^{min}$，则将D设置为4；否则将D设置为3。

[0077] 活动性值A计算如下：

[0078] $$A = \sum_{k=i-2}^{i+3} \sum_{l=j-2}^{j+3} (V_{k,l} + H_{k,l}) \qquad (8)$$

[0079] A被进一步量化到0至4(包括端点)的范围，并且量化值被表示为$\hat{A}$。对于图片中的两个色度分量，不应用分类方法，即对于每个色度分量应用单组ALF系数。

[0080] 3.1.2滤波器系数的几何变换

[0081] 在对每个2×2块进行滤波之前，取决于对于该块所计算的梯度值，对滤波系数f(k,l)应用几何变换，诸如旋转或对角翻转和垂直翻转。这相当于将这些变换应用于滤波器支持区域中的样点。这个想法是通过对齐应用了ALF的不同的块的方向性，使这些不同的块更加相似。

[0082] 引入三种几何变换，包括对角翻转、垂直翻转和旋转：

[0083] 对角翻转：$f_D(k,l) = f(l,k)$，

[0084] 垂直翻转：$f_V(k,l) = f(k,K-l-1)$， (9)

[0085] 旋转：$f_R(k,l) = f(K-l-1,k)$。

[0086] 这里，K是滤波器的尺寸，并且$0 \le k,l \le K-1$是系数坐标，使得位置(0,0)位于左上角，并且位置(K-1,K-1)位于右下角。取决于对于该块所计算的梯度值，将变换应用于滤波器系数f(k,l)。表1总结了变换和四个方向的四个梯度之间的关系。

[0087] 表1：对于一个块所计算的梯度和变换的映射

[0088]

梯度值	变换
$g_{d2} < g_{d1}$且$g_h < g_v$	无变换
$g_{d2} < g_{d1}$且$g_v < g_h$	对角翻转
$g_{d1} < g_{d2}$且$g_h < g_v$	垂直翻转
$g_{d1} < g_{d2}$且$g_v < g_h$	旋转

[0089] 3.1.3滤波器参数的信令

[0090] 在JEM中，对于第一CTU来信令通知GALF滤波器参数，即在条带头之后和第一CTU的SAO参数之前。可以信令通知多达25个亮度滤波器系数集。为了减少比特开销，可以合并不

同分类的滤波器系数。并且,参考图片的GALF系数被存储并且被允许作为当前图片的GALF系数而重新使用。当前图片可以选择使用为参考图片存储的GALF系数,并且旁路(bypass)GALF系数信令。在这种情况下,仅信令通知参考图片之一的索引,并且为当前图片继承所指示的参考图片的存储的GALF系数。

[0091] 为了支持GALF时域预测,维护GALF滤波器组的候选列表。在开始对新序列进行解码时,候选列表是空的。在对一个图片进行之后,可以将对应的一组滤波器添加到候选列表中。一旦候选列表的尺寸达到最大允许值(即在当前JEM中为6),新的一组滤波器将按解码次序覆盖最旧的一组滤波器,也就是说,应用先进先出(first-in-first-out,FIFO)规则来更新候选列表。为了避免重复,只有当对应的图片不使用GALF时域预测时,才能将一组滤波器添加到列表中。为了支持时域缩放性,有多个滤波器组的候选列表,并且每个候选列表与时域层相关联。更具体地,由时域层索引(TempIdx)分配的每个阵列可以组成先前解码的图片的滤波器组,其TempIdx等于较低TempIdx。例如,第k个阵列被分配为与等于k的TempIdx相关联,并且它仅包含来自TempIdx小于或等于k的图片的滤波器组。在对特定图片进行编解码之后,与该图片相关联的滤波器组将被用于更新与等于或更高TempIdx相关联的那些阵列。

[0092] GALF系数的时域预测被用于帧间编解码帧,以最小化信令开销。对于帧内帧,时域预测不可用,并且一组16个固定滤波器被分配给每个类。为了指示固定滤波器的使用,信令通知每个类的标志,并且如果需要,还信令通知所选择的固定滤波器的索引。即使当为给定类选择固定滤波器时,仍然可以针对该类发送自适应滤波器的系数f(k,l),在这种情况下,将被应用于重构图像的滤波器的系数是两组系数的和。

[0093] 可以在CU级别控制亮度分量的滤波过程。信令通知一个标志以指示GALF是否被应用于CU的亮度分量。对于色度分量,仅在图片级别指示是否应用GALF。

[0094] 3.1.4滤波过程

[0095] 在解码器侧,当对块启用GALF时,对块内每个样点R(i,j)进行滤波,得到样点值R'(i,j),如下所示,其中L表示滤波器长度,$f_{m,n}$表示滤波器系数,并且f(k,l)表示解码滤波器系数。

[0096] $$R'(i,j)=\sum_{k=-L/2}^{L/2}\sum_{l=-L/2}^{L/2}f(k,l)\times R(i+k,j+l) \qquad (10)$$

[0097] 3.1.5编码器侧滤波器参数的确定过程

[0098] 图3示出了GALF的整个编码器决定过程。对于每个CU的亮度样点,编码器决定是否应用GALF,并且条带头中包括适当的信令标志。对于色度样点,应用滤波器的决定是基于图片级别而不是CU级别做出的。此外,只有在对图片启用亮度GALF时,才检查图片的色度GALF。

[0099] 4VVC中基于几何变换的自适应环路滤波器的示例

[0100] 与JEM相比,VVC中的GALF的当前设计有以下主要改变:

[0101] 1)自适应滤波器形状被移除。对于亮度分量仅允许7x7滤波器形状,而对于色度分量允许5x5滤波器形状。

[0102] 2)ALF参数的时域预测和来自固定滤波器的预测都被移除。

[0103] 3)对于每个CTU,信令通知一个比特标志以指示ALF是启用还是禁用。

[0104] 4)类索引的计算是在4x4级别而不是2x2级别执行的。另外,如JVET-L0147中所提

出的，利用用于ALF分类的子采样拉普拉斯计算方法。更具体地，不需要为一个块内的每个样点计算水平/垂直/45度对角/135度梯度。相反，利用1:2子采样。

[0105] 5AVS2中基于区域的自适应环路滤波器的示例

[0106] ALF是环路滤波的最后阶段。在该过程中有两个阶段。第一阶段是滤波器系数推导。为了训练滤波器系数，编码器将亮度分量的重构像素分类为16个区域，并且使用wiener-hopf方程为每个类别训练一组滤波器系数，以最小化原始帧和重构帧之间的均方误差。为了减少这16组滤波器系数之间的冗余，编码器将基于率失真性能来自适应地合并它们。处于最大值时，可以为亮度分量分配16个不同的滤波器组，而仅为色度分量分配一个滤波器组。第二阶段是滤波器决定，包括帧级别和LCU级别。首先，编码器决定是否执行帧级别自适应环路滤波。如果帧级别ALF开启，则编码器进一步决定是否执行LCU级别ALF。

[0107] 5.1滤波器形状

[0108] 在AVS-2中采用的滤波器形状是叠加了3×3正方形的7×7十字形，正如图5中针对亮度和色度分量所示。图5中的每个正方形对应于一个样点。因此，总共17个样点被用于推导位置C8的样点的滤波值。考虑到传输系数的开销，利用仅剩下九个系数({C0,C1,…,C8})的点对称滤波器，这将滤波器系数的数量以及滤波中的乘法次数减少了一半。点对称滤波器还可以减少针对一个滤波样点的一半计算，例如，针对一个滤波样点只有9次乘法和14次加法运算。

[0109] 5.2基于区域的自适应合并

[0110] 为了适应不同的编解码误差，AVS-2对亮度分量采用了基于区域的多自适应环路滤波器。亮度分量被划分成16个尺寸大致相等的基本区域，其中每个基本区域与最大编解码单元(largest coding unit,LCU)边界对齐，如图6所示，并且为每个区域推导出一个维纳滤波器。使用的滤波器越多，失真减少得越多，但是用于对这些系数进行编码的比特随着滤波器数量的增加而增加。为了获得最佳的率失真性能，可以将这些区域合并成更少更大的区域，这些更少更大的区域共享相同的滤波器系数。为了简化合并过程，基于图像先验相关性，根据修改的希尔伯特次序为每个区域分配索引。具有连续索引的两个区域可以基于率失真成本进行合并。

[0111] 应该向解码器信令通知区域之间的映射信息。在AVS-2中，基本区域的数量被用来表示合并结果，并且滤波器系数根据其区域次序被顺序地压缩。例如，当{0,1}、{2,3,4}、{5,6,7,8,9}和左侧基本区域分别合并为一个区域时，只有三个整数被编解码以表示该合并图，即2、3、5。

[0112] 5.3边信息的信令

[0113] 还使用了多个开关(switch)标志。序列开关标志adaptive_loop_filter_enable用于控制自适应环路滤波器是否应用于整个序列。图像开关标志picture_alf_enble[i]控制对于对应的第i个图像分量是否应用ALF。只有当启用picture_alf_enble[i]时，才传输该色彩分量的对应的LCU级别标志和滤波器系数。LCU级别标志lcu_alf_enable[k]控制对于对应的第k个LCU是否启用ALF，并且被交织到条带数据中。不同级别调节的标志的决定都基于率失真成本。这种高灵活性进一步使得ALF大大提高了编解码效率。

[0114] 在一些实施例中，对于亮度分量，可以有多达16组滤波器系数。

[0115] 在一些实施例中，对于每个色度分量(Cb和Cr)，可以传输一组滤波器系数。

[0116] 6 VTM-4中的GALF

[0117] 在VTM4.0中,自适应环路滤波器的滤波过程执行如下:

[0118] $O(x,y)=\sum_{(i,j)}w(i,j).I(x+i,y+j)$ (11)

[0119] 其中,样点I(x+i,y+j)是输入样点,O(x,y)是滤波后的输出样点(即滤波结果),并且w(i,j)表示滤波器系数。实际上,在VTM4.0中,使用整数算法来实施定点精度计算:

[0120] $$O(x,y)=\left(\sum_{i=-\frac{L}{2}}^{\frac{L}{2}}\sum_{j=-\frac{L}{2}}^{\frac{L}{2}}w(i,j).I(x+i,y+j)+64\right)\gg 7 \qquad (12)$$

[0121] 其中,L表示滤波器长度,并且其中,w(i,j)是定点精度的滤波器系数。

[0122] 7非线性自适应环路滤波(ALF)

[0123] 7.1滤波重构

[0124] 方程(11)可以在不影响编解码效率的情况下用以下表达式进行修订:

[0125] $O(x,y)=I(x,y)+\sum_{(i,j)\neq(0,0)}w(i,j).(I(x+i,y+j)-I(x,y))$ (13)

[0126] 这里,w(i,j)是与方程(11)中的滤波器系数相同的滤波器系数(除了w(0,0),它在方程(13)中等于1,而它在方程(11)中等于$1-\sum_{(i,j)\neq(0,0)}w(i,j)$]。

[0127] 7.2修改的滤波器

[0128] 使用上述(13)的滤波器公式,当临近样点值(I(x+i,y+j))与正在被滤波的当前样点值(I(x,y))相差太大时,我们可以通过使用简单的限幅函数来减少这些临近样点值的影响,从而引入简单的非线性,使得AFL更高效。

[0129] 在本提案中,ALF滤波器被修改如下:

[0130] $O'(x,y)=I(x,y)+\sum_{(i,j)\neq(0,0)}w(i,j).K(I(x+i,j+j)-I(x,y),k(i,j))$ (14)

[0131] 这里,K(d,b)=min(b,max(b,d))是限幅函数,而k(i,j)是限幅参数,它取决于(i,j)滤波器系数。编码器执行优化以找到最佳k(i,j)。

[0132] 在JVET-NO242实施方式中,为每个ALF滤波器指定限幅参数k(i,j),对于每个滤波器系数信令通知一个限幅值。这意味着对于每个亮度滤波器可以在比特流中信令通知多达12个限幅值,并且对于色度滤波器可以信令通知多达6个限幅值。

[0133] 为了限制信令成本和编码器复杂度,我们将限幅值的评估限制在一个小的可能值的集合内。在该提案中,我们只使用4个固定值,这4个固定值对于INTER和INTER片组是相同的。

[0134] 因为亮度的局部差异的方差通常高于色度,所以我们对亮度和色度滤波器使用两个不同的集合。我们还在每个集合中包括最大样点值(这里,对于10比特的比特深度为1024),因此如果没有必要,则可以禁用限幅。

[0135] 表2提供了JVET-NO242测试中使用的限幅值的集合。这4个值是通过在对数域中对亮度的样点值的整个范围(以10位编解码)和色度的从4到1024的范围进行大致相等的划分来选择的。

[0136] 更准确地,限幅值的亮度表通过以下公式获得:

[0137] $\mathrm{AlfClip_L}=\left\{\mathrm{round}\left(\left((M)^{\frac{1}{N}}\right)^{N-n+1}\right)\text{对于}n\in 1..N\right\}$,其中$M=2^{10}$且$N=4$。

[0138] 类似地,限幅值的色度表根据以下公式获得:

[0139] $\text{AlfClip}_C=\left\{\text{round}\left(A.\left(\left(\frac{M}{A}\right)^{\frac{1}{N-1}}\right)^{N-n}\right)\text{对于}n\in 1..N\right\}$,其中$M=2^{10}$,$N=4$且$A=4$。

[0140] 表2:授权的限幅值

[0141]

	帧内/帧间片组
亮度	{1024, 181, 32, 6}
色度	{1024, 161, 25, 4}

[0142] 通过使用与上表2中的限幅值的索引相对应的Golomb编码方案,在“alf_data”语法元素中对所选择的限幅值进行编解码。该编码方案与滤波器索引的编码方案相同。

[0143] 8JVET-M0427中的环路整形(ILR)

[0144] 环路整形(in-loop reshaping,ILR)也称为利用色度缩放的亮度映射(Luma Mapping with Chroma Scaling,LMCS)。

[0145] 环路整形(ILR)的基本思想是将原始(在第一域)信号(预测/重构信号)转换到第二域(整形域)。

[0146] 环路亮度整形器被实施为一对查找表(look-up table,LUT),但是两个LUT中只有一个需要被信令通知,因为另一LUT可以从信令通知的LUT计算出。每个LUT都是一维的、10比特的、1024条目的映射表(1D-LUT)。一个LUT是正向LUT,FwdLUT,它将输入亮度码值Y_i映射到变更值Y_r:$Y_r=\text{wdLUT}[Y_i]$。另一LUT是一个逆向LUT,InvLUT,它将变更码值Y_r映射到$\hat{Y}_i$:$\hat{Y}_i=InvLUT[Y_r]$。($\hat{Y}_i$表示Y_i的重构值。)

[0147] 8.1PWL模型

[0148] 概念上,分段线性(piece-wise linear,PWL)以如下方式来实施:

[0149] 设x1、x2为两个输入枢轴点,并且y1、y2为它们对于一个段(piece)的对应的输出枢轴点。x1和x2之间的任何输入值x的输出值y可以通过以下等式进行插值:

[0150] y=((y2-y1)/(x2-x1))*(x-x1)+y1

[0151] 在定点实施方式中,等式可以改写为:

[0152] y=((m*x+2FP_PREC-1)>>FP_PREC)+c

[0153] 这里,m是标量,c是偏移,FP_PREC是指定精度的常数值。

[0154] 注意,在CE-12软件中,PWL模型用于预先计算1024条目的FwdLUT和InvLUT映射表;但是PWL模型也允许在不预先计算LUT的情况下即时计算相同的映射值的实施方式。

[0155] 8.2在第四次VVC会议中的测试CE12-2

[0156] 8.2.1亮度整形

[0157] 一种环路亮度整形的测试2(即提案中的CE12-2)提供了较低复杂度的管线,该管线还消除了帧间条带重构中逐块帧内预测的解码延时。帧内预测在帧间和帧内条带两者的整形域中执行。

[0158] 无论条带类型如何,帧内预测总是在整形域中执行。在这种布置下,帧内预测可以在先前的TU重构完成之后立即开始。这种布置还可以为帧内模式提供统一的过程,而不是依赖于条带的。图7示出了基于模式的CE12-2解码过程的框图。

[0159] CE12-2还针对亮度和色度残差缩放测试了16段分段线性(PWL)模型,而不是CE12-1的32段PWL模型。

[0160] CE12-2中利用环路亮度整形器的帧间条带重构(较浅阴影块指示整形域中的信号:亮度残差;帧内亮度预测的;以及帧内亮度重构的)

[0161] 8.2.2依赖于亮度的色度残差缩放

[0162] 依赖于亮度的色度残差缩放是一个用定点整数运算实施的乘法过程。色度残差缩放补偿亮度信号与色度信号的相互作用。色度残差缩放应用于TU级别。更具体地,以下情况适用:

[0163] -对于帧内,对重构亮度进行平均。

[0164] -对于帧间,对预测亮度进行平均。

[0165] 平均用于表示PWL模型中的索引。索引表示缩放因子cScaleInv。色度残差乘以该数。

[0166] 注意,色度缩放因子是根据正向映射的预测亮度值而不是重构亮度值计算出的。

[0167] 8.2.3ILR边信息的信令

[0168] 参数(当前)在片组头中发送(类似于ALF)。这些据称需要40-100比特。以下规范基于JVET-L1001的第9版。下面用斜体突出显示了添加的语法。

[0169] 在7.3.2.1中的序列参数集RBSP语法

[0170]

seq_parameter_set_rbsp( ) {	**描述符**
sps_seq_parameter_set_id	ue(v)
…	

[0171]

sps_triangle_enabled_flag	u(1)
sps_ladf_enabled_flag	u(1)
if (sps_ladf_enabled_flag) {	
sps_num_ladf_intervals_minus2	u(2)
sps_ladf_lowest_interval_qp_offset	se(v)
for(i = 0; i < sps_num_ladf_intervals_minus2 + 1; i++) {	
sps_ladf_qp_offset[i]	se(v)
sps_ladf_delta_threshold_minus1[i]	ue(v)
}	
}	
sps_reshaper_enabled_flag	*u(1)*
rbsp_trailing_bits( )	
}	

[0172] 在7.3.3.1中的通用片组头语法

[0173]

tile_group_header( ) {	描述符
…	
if(num_tiles_in_tile_group_minus1 > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < num_tiles_in_tile_group_minus1; i++)	
entry_point_offset_minus1[i]	u(v)
}	
if (sps_reshaper_enabled_flag) {	
tile_group_reshaper_model_present_flag	*u(1)*
if (tile_group_reshaper_model_present_flag)	
tile_group_reshaper_model ( )	
tile_group_reshaper_enable_flag	*u(1)*
if (tile_group_reshaper_enable_flag && (!(qtbtt_dual_tree_intra_flag && tile_group_type == I)))	
tile_group_reshaper_chr oma_residual_scale_flag	*u(1)*
}	
byte_alignment( )	
}	

[0174] 添加新的语法表片组整形器模型：

[0175]

tile_group_reshaper_model () {	描述符
reshaper_model_min_bin_idx	*ue(v)*
reshaper_model_delta_max_bin_idx	*ue(v)*
reshaper_model_bin_delta_abs_cw_prec_minus1	*ue(v)*
for (i = reshaper_model_min_bin_idx; i <= reshaper_model_max_bin_idx; i++) {	
reshape_model_bin_delta_abs_CW [i]	*u(v)*
if (reshaper_model_bin_delta_abs_CW[i]) > 0)	
reshaper_model_bin_delta_sign_CW_flag[i]	*u(1)*
}	
}	

[0176] 在通用序列参数集RBSP语义中，添加以下语义：

[0177] sps_reshaper_enabled_flag等于1指定在编解码视频序列(coded video sequence，CVS)中使用整形器。sps_reshaper_enabled_flag等于0指定在CVS中不使用整形器。

[0178] 在片组头语法中，添加以下语义：

[0179] tile_group_reshaper_model_present_flag等于1指定在片组头中存在tile_group_reshaper_model()。tile_group_reshaper_model_present_flag等于0指定在片组头中不存在tile_group_reshaper_model()。当不存在tile_group_reshaper_model_present_flag时，它被推断为等于0。

[0180] tile_group_reshaper_enabled_flag等于1指定对当前片组启用整形器。tile_group_reshaper_enabled_flag等于0指定不对当前片组启用整形器。当tile_group_reshaper_enable_flag不存在时，它被推断为等于0。

[0181] tile_group_reshaper_chroma_residual_scale_flag等于1指定对当前片组启用色度残差缩放。tile_group_reshaper_chroma_residual_scale_flag等于0指定不对当前片组启用色度残差缩放。当tile_group_reshaper_chroma_residual_scale_flag不存在

时,它被推断为等于0。

[0182] 添加tile_group_reshaper_model()语法

[0183] reshape_model_min_bin_idx指定要在整形器构建过程中使用的最小二进制位(或段)索引。reshape_model_min_bin_idx的值应在0至MaxBinIdx(包括端点)的范围内。MaxBinIdx的值应等于15。

[0184] reshape_model_delta_max_bin_idx指定最大允许二进制位(或段)索引MaxBinIdx减去要在整形器构建过程中使用的最大二进制位索引。reshape_model_max_bin_idx的值被设置为等于MaxBinIDx-reshape_model_delta_max_bin_idx。

[0185] reshaper_model_bin_delta_abs_cw_prec_minus1加1指定用于表示语法reshape_model_bin_delta_abs_CW[i]的比特数。

[0186] reshape_model_bin_delta_abs_CW[i]指定第i个二进制位的绝对增量码字值。

[0187] reshaper_model_bin_delta_sign_CW_flag[i]指定reshape_model_bin_delta_abs_CW[i]的符号,如下所示:

[0188] -如果reshape_model_bin_delta_sign_CW_flag[i]等于0,则对应的变量RspDeltaCW[i]为正值。

[0189] -否则(reshape_model_bin_delta_sign_CW_flag[i]不等于0),则对应的变量RspDeltaCW[i]为负值。

[0190] 当reshape_model_bin_delta_sign_CW_flag[i]不存在时,它被推断为等于0。

[0191] 变量RspDeltaCW[i]＝(1 2*reshape_model_bin_delta_sign_CW[i])*reshape_model_bin_delta_abs_CW[i];

[0192] 变量RspCW[i]按以下步骤进行推导:

[0193] 变量OrgCW被设置为等于(1<<BitDepthY)/(MaxBinIdx+1)。

[0194] -如果reshaper_model_min_bin_idx<＝i<＝reshaper_model_max_bin_idx,

[0195] 则RspCW[i]＝OrgCW+RspDeltaCW[i]。

[0196] -否则,RspCW[i]＝0。

[0197] 如果$BitDepth_Y$的值等于10,则RspCW[i]的值应在32至2*OrgCW-1的范围内。

[0198] 变量InputPivot[i](i在0至MaxBinIdx+1范围内(包括端点))被推导如下:

[0199] InputPivot[i]＝i*OrgCW

[0200] 变量ReshapePivot[i](i在0至MaxBinIdx+1范围内(包括端点))以及变量ScaleCoef[i]和InvScaleCoeff[i](i在0至MaxBinIdx范围内(包括端点))被推导如下:

```
shiftY = 14
ReshapePivot[ 0 ] = 0;
for( i = 0; i <= MaxBinIdx ; i++) {
    ReshapePivot[ i + 1 ] = ReshapePivot[ i ] + RspCW[ i ]
    ScaleCoef[ i ]=( RspCW[ i ] * (1 << shiftY) + (1 <<
(Log2(OrgCW) - 1))) >> (Log2(OrgCW))
    if ( RspCW[ i ] == 0 )
      InvScaleCoeff[ i ] = 0
    else
      InvScaleCoeff[ i ] = OrgCW * (1 << shiftY) / RspCW[ i ]
}
```

[0201]

[0202] 变量ChromaScaleCoef[i](i在0至MaxBinIdx范围内(包括端点))被推导如下：

[0203] ChromaResidualScaleLut[64]＝{16384,16384,16384,16384,16384,16384,16384,8192,8192,8192,8192,5461,5461,5461,5461,4096,4096,4096,4096,3277,3277,3277,3277,2731,2731,2731,2731,2341,2341,2341,2048,2048,2048,1820,1820,1820,1638,1638,1638,1638,1489,1489,1489,1489,1365,1365,1365,1365,1260,1260,1260,1260,1170,1170,1170,1170,1092,1092,1092,1092,1024,1024,1024,1024}；

[0204] shiftC＝11

[0205] -if(RspCW[i]＝＝0)

[0206] ChromaScaleCoef[i]＝(1<<shiftC)

[0207] -Otherwise(RspCW[i]!＝0),ChromaScaleCoef[i]＝ChromaResidualScaleLut[RspCW[i]>>1]

[0208] 8.2.4 ILR的使用

[0209] 在编码器侧,每个图片(或片组)首先被转换到整形域。并且所有的编解码过程都是在整形域中执行的。对于帧内预测,临近块在整形域中;对于帧间预测,(从来自解码图像缓冲区的原始域生成的)参考块首先被转换到整形域。然后生成残差并将其编解码成比特流。

[0210] 在整个图片(或片组)完成编码/解码之后,整形域中的样点被转换到原始域,然后应用去方块滤波器和其他滤波器。

[0211] 对于以下情况,禁用对预测信号的正向整形：

[0212] -当前块是帧内编解码的

[0213] -当前块被编解码为CPR(当前图片参考,也称为帧内块复制(intra block copy,IBC))

[0214] -当前块被编解码为组合的帧间帧内模式(combined inter-intra mode,CIIP),并且对帧内预测块禁用正向整形

[0215] 9现有实施方式的缺点

[0216] JVET-N0242中的非线性ALF(non-linear ALF,NLALF)设计有以下问题：

[0217] (1)它是针对4:2:0色彩格式而设计的。对于4:4:4色彩格式,亮度和色度分量可能具有相似的重要性。如何更好地应用NLALF是未知的。

[0218] (2)限幅值是针对10比特的情况而设计的。如何针对其他比特深度定义NLALF尚未研究过。

[0219] (3)环路整形方法与NLALF的相互作用尚未研究过。

[0220] 10改进非线性自适应环路滤波的示例性方法

[0221] 当前公开的技术的实施例克服了现有实施方式的缺点,从而提供具有更高编解码效率的视频编解码。基于所公开的技术的非线性自适应环路滤波可以增强现有和未来的视频编解码标准,这将在以下针对各种实施方式描述的示例中进行阐述。下面提供的公开技术的示例解释了一般概念,并且不意味着解释为限制。在一个示例中,除非明确指出相反的情况,否则这些示例中描述的各种特征可以被组合。

[0222] 1.提出在NLALF中使用的参数(例如,表2中定义的限幅参数)可以取决于编解码信息。

[0223] a.提出在NLALF中使用的参数(例如,表2中定义的限幅参数)

[0224] 可以取决于时域层索引/低延迟检查标志/参考图片。

[0225] 2.可以定义或信令通知多组NLALF参数。

[0226] a.可替换地,此外,当多组NLALF参数被信令通知时,它们可以在数据单元中被信令通知,诸如自适应参数集(Adaptation ParameterSet,APS)/片组头/视频数据单元。

[0227] b.在一个示例中,以预测方式信令通知NLALF参数。

[0228] i.例如,在一个数据单元(诸如APS或片组或条带)中信令通知的一组NLALF参数由在同一数据单元中信令通知的另一组NLALF参数预测。

[0229] ii.例如,在一个数据单元(诸如APS或片组或条带)中信令通知的一组NLALF参数由在另一数据单元中信令通知的另一组NLALF参数预测。

[0230] 3.提出在NLALF中使用的参数(例如,表2中定义的限幅参数)可以取决于在应用NLALF之前重构样点的比特深度。

[0231] a.可替换地,提出在NLALF中使用的参数(例如,表2中定义的限幅参数)可以取决于在编码/解码之前样点的输入比特深度。

[0232] b.在一个示例中,一个给定比特深度的参数可以从为另一比特深度分配的参数中推导出。

[0233] i.在一个示例中,可以应用根据比特深度的移位操作来推导一个给定比特深度的参数。

[0234] 4.提出在NLALF中使用的参数(例如,表2中定义的限幅参数)可以取决于色彩表示格式。

[0235] a.在一个示例中,对于RGB情况,针对G色彩分量和针对B/R色彩分量,参数具有相同索引。

[0236] 5.提出在NLALF中使用的参数(例如,表2中定义的限幅参数)可以取决于是否应用环路整形(ILR)方法。

[0237] a.在一个示例中,参数在启用或禁用ILR时可以不同。

[0238] 6.提出将滤波器参数(诸如滤波器系数)和NLALF参数(诸如限幅参数)存储在一

起。

[0239] a.在一个示例中,它们都可以存储在APS中。

[0240] b.在一个示例中,当一个视频数据单元(例如,CTU/区域/片组)使用与一个APS相关联的滤波器系数时,也可以利用相关联的NLALF参数。

[0241] c.可替换地,对于编码/解码一个视频数据单元(例如,CTU/区域/片组),当启用根据与一个APS相关联的滤波器系数的预测时,相关联的NLALF参数也可以用于根据同一APS来预测该一个视频数据单元的NLALF参数。

[0242] 7.如何针对色度色彩分量处理NLALF可以取决于色彩格式。

[0243] a.在一个示例中,对于一种给定色彩格式(诸如4:4:4),两个色度分量可以使用不同的NLALF参数。

[0244] 8.提出可以在序列级别、图片级别、条带别级、片组级别、片级别、CTU级别、CU级别或块级别开启或关闭ALF中的限幅。

[0245] a.例如,可以向解码器信令通知是否在ALF中开启限幅,诸如在SPS、PPS、条带头、片组头、片、CTU、CU或块中信令通知。

[0246] 上述示例可以结合在下述方法的上下文中,例如方法810至840,其可以在视频解码器或视频编解码器处实施。

[0247] 图8A示出了用于视频处理的示例性方法的流程图。方法810包括,在步骤812,将视频的视频单元编码为编码视频单元。方法810还包括,在步骤813,从编码视频单元生成重构样点。方法810还包括,在步骤814,对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的样点的比特深度的函数。方法810还包括,在步骤815,将非线性自适应环路滤波器应用于限幅操作的输出。方法810还包括,在步骤816,使用编码视频单元生成视频的编解码表示。

[0248] 图8B示出了用于视频处理的示例性方法的流程图。方法820包括,在步骤822,解析视频的编解码表示以得到表示视频的视频单元的编码视频单元。方法820还包括,在步骤823,从编码视频单元生成视频单元的重构样点。方法820包括,在步骤824,对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的比特深度的函数。方法820还包括,在步骤825,将非线性自适应环路滤波器应用于限幅操作的输出,以生成最终的解码视频单元。

[0249] 图8C示出了用于视频处理的示例性方法的流程图。方法830包括,在步骤832,执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换。方法830还包括,在步骤834,确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波。在一些实施方式中,该确定是基于视频和/或视频区域和/或视频单元的编解码信息的。在一些实施方式中,限幅参数是色彩表示格式的函数。在一些实施方式中,限幅参数取决于是否基于在第一域和第二域中视频单元的表示和/或色度视频单元的缩放色度残差来应用环路整形(ILR)对视频单元进行重构。

[0250] 图9示出了用于视频处理的示例性方法的流程图。方法840包括执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换。在一些实施方式中,编解码表示包括提供限幅参数的第一边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且第一边信息与指示在非线性自适应环路滤波器中使用的滤

波器系数的第二边信息一起被信令通知。在一些实施方式中，编解码表示包括指示多组限幅参数的边信息，该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波。在一些实施方式中，编解码表示包括提供一个或多个限幅参数的边信息，该限幅参数用于使用非线性自适应环路滤波器对视频区域的色度视频单元的重构进行滤波，其中，一个或多个限幅参数取决于色彩格式。在一些实施方式中，编解码表示包括提供限幅参数的边信息，该限幅参数用于使用自适应环路滤波器对视频区域的视频单元的重构进行滤波，其中，执行包括通过将限幅操作应用于视频区域级别的样点差来生成滤波后的视频单元。

[0251] 11所公开技术的示例实施方式

[0252] 图10A是视频处理装置900的框图。装置900可以用于实施本文描述的方法中的一个或多个。装置900可以体现在智能手机、平板电脑、计算机、物联网(IoT)接收器等中。装置900可以包括一个或多个处理器902、一个或多个存储器904和视频处理硬件906。(多个)处理器902可以被配置为实施本文档中描述的一个或多个方法(包括但不限于方法800)。(多个)存储器904可以用于存储用于实施本文描述的方法和技术的数据和代码。视频处理硬件906可以用于在硬件电路中实施本文档中描述的一些技术。

[0253] 图10B是其中可以实施所公开技术的视频处理系统的框图的另一示例。图10B是示出其中可以实施本文公开的各种技术的示例视频处理系统4100的框图。各种实施方式可以包括系统4100的一些或全部组件。系统4100可以包括用于接收视频内容的输入4102。视频内容可以以原始或未压缩格式接收，例如，8或10比特多分量像素值，或者可以以压缩或编码格式接收。输入4102可以代表网络接口、外设总线接口或存储接口。网络接口的示例包括有线接口(诸如以太网、无源光网络(passive optical network，PON)等)和无线接口(诸如Wi-Fi或蜂窝接口)。

[0254] 系统4100可以包括编解码组件4104，其可以实施本文档中描述的各种编解码(coding)或编码(encoding)方法。编解码组件4104可以降低从输入端4102到编解码组件4104的输出端的视频的平均比特率，以产生视频的编解码表示。因此，编解码技术有时被称为视频压缩或视频转码技术。编解码组件4104的输出可以被存储，或者经由连接的通信进行传输，如组件4106所示。组件4108可以使用在输入端4102接收的所存储或传送的视频的比特流(或编解码)表示来生成发送到显示接口4110的像素值或可显示视频。从比特流表示生成用户可观看的视频的过程有时被称为视频解压缩。此外，虽然某些视频处理操作被称为“编解码”操作或工具，但是应当理解，编解码工具或操作在编码器处使用，并且将由解码器来执行与编码结果相反的对应的解码工具或操作。

[0255] 外围总线接口或显示接口的示例可以包括通用串行总线(USB)或高清晰度多媒体接口(HDMI)或显示端口等。存储接口的示例包括SATA(串行高级技术附件)、PCI、IDE接口等。本文档中描述的技术可以体现在各种电子设备中，诸如移动电话、膝上型电脑、智能手机或能够执行数字数据处理和/或视频显示的其他设备。

[0256] 所公开的技术的一些实施例包括做出启用视频处理工具或模式的决定或确定。在一个示例中，当启用视频处理工具或模式时，编码器将在视频的块的处理中使用或实施该工具或模式，但是不一定基于该工具或模式的使用来修改得到的比特流。也就是说，从视频的块到视频的比特流表示的转换将在启用视频处理工具或模式时使用该视频处理工具或

模式。在另一示例中，当启用视频处理工具或模式时，解码器将在知道比特流已经基于视频处理工具或模式被修改的情况下处理比特流。也就是说，将使用基于决定或确定而启用的视频处理工具或模式来执行从视频的比特流表示到视频的块的转换。

[0257] 所公开的技术的一些实施例包括做出禁用视频处理工具或模式的决定或确定。在一个示例中，当禁用视频处理工具或模式时，编码器将不会在视频的块到视频的比特流表示的转换中使用该工具或模式。在另一示例中，当禁用视频处理工具或模式时，解码器将在知道比特流没有被使用基于决定或确定而禁用的视频处理工具或模式修改的情况下处理比特流。

[0258] 在本文档中，术语“视频处理”可以指视频编码、视频解码、视频压缩或视频解压缩。例如，视频压缩算法可以在从视频的像素表示到对应的比特流表示的转换期间被应用，反之亦然。当前视频块的比特流表示可以例如对应于比特流中共置或分布在不同位置的比特，如语法所定义的。例如，宏块可以根据变换和编解码后的误差残差值进行编码，并且还可以使用比特流中的报头和其他字段中的比特进行编码。

[0259] 在一些实施例中，视频编解码方法可以使用如参考图10A或图10B所描述的在硬件平台上实施的装置来实施。

[0260] 可以使用以下基于条款的格式来描述各种技术和实施例。

[0261] 第一组条款描述了在先前章节中列出的公开技术的某些特征和方面。

[0262] 1.一种用于视频处理的方法，包括：基于当前视频块的编解码信息，确定当前视频块的一组参数；以及基于使用该一组参数执行非线性滤波操作，从对应的比特流表示重构当前视频块。

[0263] 2.根据条款1所述的方法，其中，非线性滤波操作包括非线性自适应环路滤波。

[0264] 3.根据条款1或2所述的方法，其中，该组参数包括当前视频块的亮度分量或色度分量的至少一个限幅值。

[0265] 4.根据条款3所述的方法，其中，非线性滤波操作是基于色度分量的色彩格式的。

[0266] 5.根据条款1至3中任一项所述的方法，其中，编解码信息包括时域层索引、低延迟检查标志或一个或多个参考图片。

[0267] 6.根据条款1至3中任一项所述的方法，其中，编解码信息包括非线性滤波操作之前的重构样点的比特深度。

[0268] 7.根据条款1至3中任一项所述的方法，其中，编解码信息包括色彩表示格式。

[0269] 8.根据条款1至3中任一项所述的方法，其中，编解码信息包括应用环路整形(ILR)方法的指示。

[0270] 9.根据条款1至3中任一项所述的方法，其中，对应的比特流表示包括包含该一组参数的多组参数，并且其中，该多组参数是在自适应参数集(APS)、片组头或一个或多个视频数据单元中被信令通知的。

[0271] 10.根据条款1至3中任一项所述的方法，其中，对应的比特流表示包括自适应参数集(APS)，自适应参数集包括该一组参数和与非线性滤波操作相关联的一个或多个滤波器系数。

[0272] 11.根据条款1或2所述的方法，其中，该一组参数包括一个或多个限幅值，并且其中，非线性滤波操作在序列级别、图片级别、条带级别、片组级别、片级别、编解码树单元

(CTU)级别、编解码单元(CU)级别或块级别执行。

[0273] 12.一种视频系统中的装置,包括处理器和其上具有指令的非暂时性存储器,其中,该指令在由处理器运行时使处理器实施根据条款1至11中任一项所述的方法。

[0274] 13.一种存储在非暂时性计算机可读介质上的计算机程序产品,该计算机程序产品包括用于执行根据条款1至11中任一项所述的方法的程序代码。

[0275] 第二组条款描述了在先前章节中列出的公开技术的某些特征和方面,包括例如示例实施方式1和3-5。

[0276] 1.一种视频处理方法,包括:将视频的视频单元编码为编码视频单元;从编码视频单元生成重构样点;对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的样点的比特深度的函数;将非线性自适应环路滤波器应用于限幅操作的输出;以及使用编码视频单元生成视频的编解码表示。

[0277] 2.根据条款1所述的方法,其中,限幅索引是在编解码表示中被信令通知的。

[0278] 3.一种视频处理方法,包括:解析视频的编解码表示以得到表示视频的视频单元的编码视频单元;从编码视频单元生成视频单元的重构样点;对重构样点执行限幅操作,其中,在限幅操作中使用的限幅参数是限幅索引和重构样点的比特深度或视频单元的比特深度的函数;以及将非线性自适应环路滤波器应用于限幅操作的输出,以生成最终的解码视频单元。

[0279] 4.根据条款3所述的方法,其中,限幅索引是至少基于编解码表示中的字段来确定的。

[0280] 5.根据条款3所述的方法,其中,限幅索引是使用预定义规则来确定的。

[0281] 6.根据条款1或3所述的方法,其中,限幅索引和重构样点的比特深度或视频单元的比特深度的函数这样的函数,即基于重构样点的比特深度或视频单元的比特深度,针对限幅索引的给定值返回不同的值的函数。

[0282] 7.根据条款1或3所述的方法,其中,限幅索引和限幅参数之间的映射取决于重构样点的比特深度或视频单元的比特深度。

[0283] 8.根据条款1或3所述的方法,其中,基于另一比特深度的与第二限幅索引相对应的第二限幅值,推导给定比特深度的与第一限幅索引相对应的第一限幅值。

[0284] 9.根据条款8所述的方法,其中,使用另一比特深度的移位操作被应用来推导给定比特深度的限幅参数。

[0285] 10.根据条款1至9中任一项所述的方法,其中,编解码表示包括限幅参数,该限幅参数控制在非线性自适应环路滤波器中使用的两个样点差的上限或下限。

[0286] 11.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且其中,该确定是基于视频和/或视频区域和/或视频单元的编解码信息的。

[0287] 12.根据条款11所述的方法,其中,编解码信息包括时域层索引。

[0288] 13.根据条款11所述的方法,其中,编解码信息包括低延迟检查标志。

[0289] 14.根据条款11所述的方法,其中,编解码信息包括一个或多个参考图片。

[0290] 15.根据条款11至14中任一项所述的方法,其中,视频区域包括视频图片。

[0291] 16.根据条款11至14中任一项所述的方法,其中,视频单元包括编解码单元。

[0292] 17.根据条款11至16中任一项所述的方法,其中,限幅参数控制在非线性自适应环路滤波器中使用的两个样点差的上限或下限。

[0293] 18.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且其中,限幅参数是色彩表示格式的函数。

[0294] 19.根据条款18所述的方法,其中,对于RGB色彩格式,限幅参数针对绿色分量和针对蓝色或红色分量具有相同的索引。

[0295] 20.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;以及确定限幅参数,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波,并且其中,限幅参数取决于是否基于在第一域和第二域中视频单元的表示和/或色度视频单元的缩放色度残差来应用环路整形(ILR)对视频单元进行重构。

[0296] 21.根据条款1至21中任一项所述的方法,其中,限幅参数对应于视频单元的亮度分量或色度分量的限幅值。

[0297] 22.根据条款1至21中任一项所述的方法,其中,该方法还包括,在转换期间,通过将非线性自适应环路滤波器应用于视频单元的重构来生成滤波后的视频单元,并且使用滤波后的视频单元来确定视频的另一视频单元的预测。

[0298] 23.根据条款1至22中任一项所述的方法,其中,执行转换包括从视频生成编解码表示。

[0299] 24.根据条款1至22中任一项所述的方法,其中,执行转换包括从编解码表示生成视频。

[0300] 25.一种视频系统中的装置,包括处理器和其上具有指令的非暂时性存储器,其中,该指令在由处理器运行时使处理器实施根据条款1至24中的一项或多项所述的方法。

[0301] 26.一种存储在非暂时性计算机可读介质上的计算机程序产品,计算机程序产品包括用于执行根据条款1至24中的一项或多项所述的方法的程序代码。

[0302] 第三组条款描述了在先前章节中列出的公开技术的某些特征和方面,包括例如示例实施方式2和6-8。

[0303] 1.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括提供限幅参数的第一边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波;其中,第一边信息与指示非线性自适应环路滤波器中使用的滤波器系数的第二边信息一起被信令通知。

[0304] 2.根据条款1的所述的方法,其中,第一边信息和第二边信息是在相同的自适应参数集中被信令通知的。

[0305] 3.根据条款1所述的方法,其中,在视频数据单元使用与自适应参数集相关联的至少一些滤波器系数的情况下,视频数据单元也使用与自适应参数集相关联的限幅参数。

[0306] 4.根据条款1所述的方法,其中,在对视频数据单元的转换启用根据与自适应参数集相关联的至少一些滤波器系数的预测的情况下,与自适应参数集相关联的参数用于根据自适应参数集预测另一视频数据单元的另一参数。

[0307] 5.根据条款3或4所述的方法,其中,视频数据单元是编解码树单元、视频区域或片组。

[0308] 6.根据条款1至5中任一项所述的方法,其中,参数对应于视频单元的亮度分量或色度分量的限幅值。

[0309] 7.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括指示多组限幅参数的边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波。

[0310] 8.根据条款7所述的方法,其中,边信息包括该多组限幅参数。

[0311] 9.根据条款7所述的方法,其中,该多组限幅参数对于编码器和解码器是已知的,并且边信息包括该多组限幅参数中的一个或多个限幅参数的索引。

[0312] 10.根据条款7所述的方法,其中,该多组限幅参数被包括在视频数据单元、或视频单元的头中。

[0313] 11.根据条款10所述的方法,其中,视频数据单元包括自适应参数集、片组或条带。

[0314] 12.根据条款8所述的方法,其中,在数据单元中信令通知的多组限幅参数中的一组限幅参数由在该数据单元中信令通知的另一组限幅参数预测。

[0315] 13.根据条款7所述的方法,其中,在数据单元中信令通知的多组限幅参数中的一组限幅参数由在另一数据单元中信令通知的另一组限幅参数预测。

[0316] 14.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换;其中,编解码表示包括提供一个或多个限幅参数的边信息,该限幅参数用于使用非线性自适应环路滤波器对视频区域的色度视频单元的重构进行滤波,其中,一个或多个限幅参数取决于色彩格式。

[0317] 15.根据条款14所述的方法,其中,对于特定色彩格式,两个色度分量使用不同的限幅参数。

[0318] 16.根据条款15所述的方法,其中,特定色彩格式是4:4:4。

[0319] 17.根据条款1至16中任一项所述的方法,其中,该方法还包括,在转换期间,通过将非线性自适应环路滤波器应用于视频单元的重构来生成滤波后的视频单元,并且使用滤波后的视频单元来确定视频的另一视频单元的预测。

[0320] 18.一种视频处理方法,包括:执行包括一个或多个视频区域的视频的编解码表示和视频之间的转换,其中,编解码表示包括提供限幅参数的边信息,该限幅参数用于使用自适应环路滤波器对视频区域的视频单元的重构进行滤波,其中,该执行包括通过将限幅操作应用于视频区域级别的样点差来生成滤波后的视频单元。

[0321] 19.根据条款18所述的方法,其中,视频区域级别是序列级别、图片级别、条带级别、片组级别、片级别、编解码树单元级别、编解码单元级别或块级别。

[0322] 20.根据条款18所述的方法,其中,对启用限幅操作的指示是在序列参数集(SPS)、图片参数集(PPS)、条带头、片组头、片、编解码树单元、编解码单元或块中被信令通知的。

[0323] 21.根据条款1至20中任一项所述的方法,其中,视频区域是视频图片。

[0324] 22.根据条款1至20中任一项所述的方法,其中,视频单元是编解码单元或变换单元或条带或编解码树或编解码树行。

[0325] 23.根据条款1至22中任一项所述的方法,其中,执行转换包括从当前块生成编解

码表示。

[0326] 24.根据条款1至22中任一项所述的方法,其中,执行转换包括从编解码表示生成当前块。

[0327] 25.一种视频系统中的装置,包括处理器和其上具有指令的非暂时性存储器,其中,指令在由处理器运行时使处理器实施根据条款1至24中任一项所述的方法。

[0328] 26.一种存储在非暂时性计算机可读介质上的计算机程序产品,该计算机程序产品包括用于执行根据条款1至24中任一项所述的方法的程序代码。

[0329] 根据前述内容,应当理解,出于说明的目的,本文已经描述了当前公开的技术的具体实施例,但是在不脱离本发明的范围的情况下,可以进行各种修改。因此,除了所附权利要求之外,当前公开的技术不受限制。

[0330] 本专利文档中描述的主题和功能操作的实施方式可以在各种系统、数字电子电路或计算机软件、固件或硬件中实施,包括本说明书中公开的结构及其结构等同物,或者它们中的一个或多个的组合。本说明书中描述的主题的实施方式可以被实施为一个或多个计算机程序产品,即,编码在有形和非暂时性计算机可读介质上的计算机程序指令的一个或多个模块,用于由数据处理装置执行或控制数据处理装置的操作。计算机可读介质可以是机器可读存储设备、机器可读存储基底、存储器设备、实现机器可读传播信号的合成物质或它们中的一个或多个的组合。术语“数据处理单元”或“数据处理装置”涵盖用于处理数据的所有装置、设备和机器,例如包括可编程处理器、计算机、或者多个处理器或计算机。除了硬件之外,该装置可以包括为所讨论的计算机程序创建运行环境的代码,例如,构成处理器固件、协议栈、数据库管理系统、操作系统或它们中的一个或多个的组合的代码。

[0331] 计算机程序(也称为程序、软件、软件应用、脚本或代码)可以以任何形式的编程语言进行编写,包括编译或解释语言,并且它可以以任何形式部署,包括作为独立程序或作为模块、组件、子例程或适于在计算环境中使用的其他单元。计算机程序不一定对应于文件系统中的文件。程序可以存储在保存其他程序或数据的文件的一部分中(例如,存储在标记语言文档中的一个或多个脚本),存储在专用于所讨论的程序的单个文件中,或者存储在多个协调文件中(例如,存储一个或多个模块、子程序或部分代码的文件)。计算机程序可以被部署为在一个计算机或位于一个站点或分布在多个站点并通过通信网络互连的多个计算机上运行。

[0332] 本说明书中描述的过程和逻辑流程可以由运行一个或多个计算机程序的一个或多个可编程处理器来执行,以通过对输入数据进行操作并生成输出来执行功能。过程和逻辑流程也可以由专用逻辑电路来执行,并且设备也可以被实施为专用逻辑电路,例如,现场可编程门阵列(FPGA)或专用集成电路(ASIC)。

[0333] 举例来说,适于运行计算机程序的处理器包括通用和专用微处理器,以及任何类型的数字计算机的任何一个或多个处理器。通常,处理器将从只读存储器或随机存取存储器或两者接收指令和数据。计算机的基本元件是用于执行指令的处理器和用于存储指令和数据的一个或多个存储器设备。通常,计算机还将包括用于存储数据的一个或多个大容量存储设备,例如磁盘、磁光盘或光盘,或者可操作地耦合到这样的一个或多个大容量存储设备,以从其接收数据,或向其传送数据,或两者兼有。然而,计算机不需要这样的设备。适合于存储计算机程序指令和数据的计算机可读介质包括所有形式的非易失性存储器、介质和

存储器设备，包括例如半导体存储器设备，例如，电EPROM、EEPROM和闪存设备。处理器和存储器可以由专用逻辑电路来补充或并入其中。

[0334] 本说明书和附图仅被认为是示例性的，其中示例性意味着示例。如本文所使用的，单数形式“一”、“一个”和“该”也旨在包括复数形式，除非上下文中明确指示。此外，“或”的使用旨在包括“和/或”，除非上下文另有明确指示。

[0335] 尽管本专利文档包含许多细节，但是这些细节不应被解释为对任何发明的范围或可能要求保护的内容的限制，而是对特定发明的特定实施例所特有的特征的描述。本专利文档中在单独实施例的上下文中描述的某些特征也可以在单个实施例中以组合来实施。相反，在单个实施例的上下文中描述的各种特征也可以在多个实施例中单独或以任何合适的子组合来实施。此外，尽管特征可以在上面被描述为在某些组合中起作用，甚至最初也是这样要求保护的，但是在某些情况下，来自所要求保护的组合的一个或多个特征可以从该组合中删除，并且所要求保护的组合可以针对子组合或子组合的变体。

[0336] 类似地，尽管在附图中以特定的顺序描述了操作，但是这不应该被理解为要求以所示的特定的次序或顺序的次序来执行这些操作，或者执行所有示出的操作，以获得期望的结果。此外，在本专利文档中描述的实施例中的各种系统组件的分离不应该被理解为在所有实施例中都需要这样的分离。

[0337] 仅描述了几个实施方式和示例，并且可以基于本专利文档中描述和示出的内容进行其他实施、增强和变化。

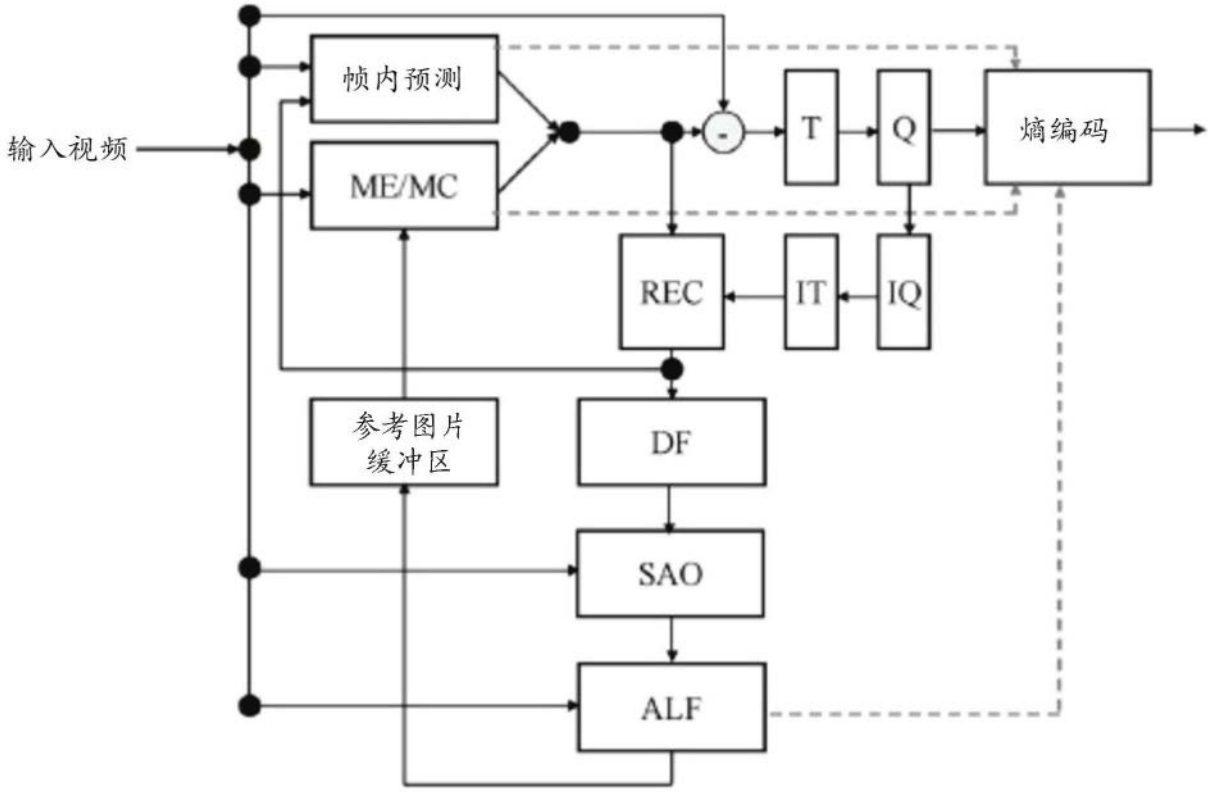

输入视频
帧内预测
ME/MC
T
Q
熵编码
REC
IT
IQ
参考图片
缓冲区
DF
SAO
ALF

图1

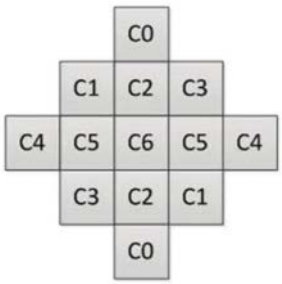

C0
C1
C2
C3
C4
C5
C6
C5
C4
C3
C2
C1
C0

图2A

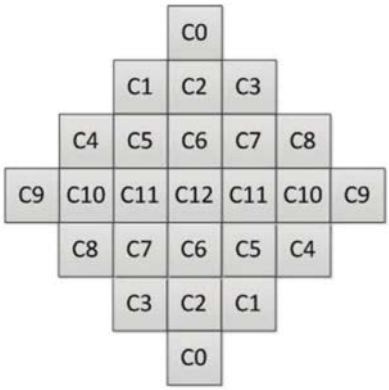

C0
C1 C2 C3
C4 C5 C6 C7 C8
C9 C10 C11 C12 C11 C10 C9
C8 C7 C6 C5 C4
C3 C2 C1
C0

图2B

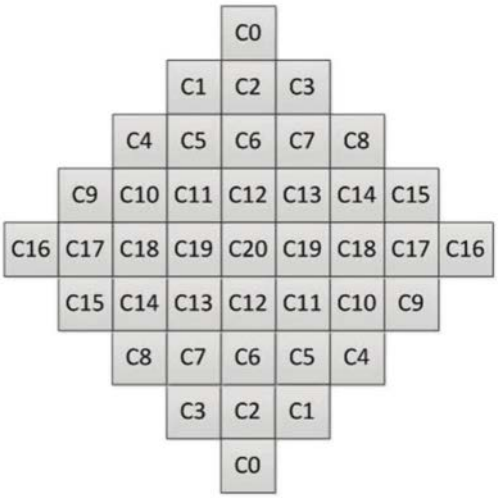

C0
C1 C2 C3
C4 C5 C6 C7 C8
C9 C10 C11 C12 C13 C14 C15
C16 C17 C18 C19 C20 C19 C18 C17 C16
C15 C14 C13 C12 C11 C10 C9
C8 C7 C6 C5 C4
C3 C2 C1
C0

图2C

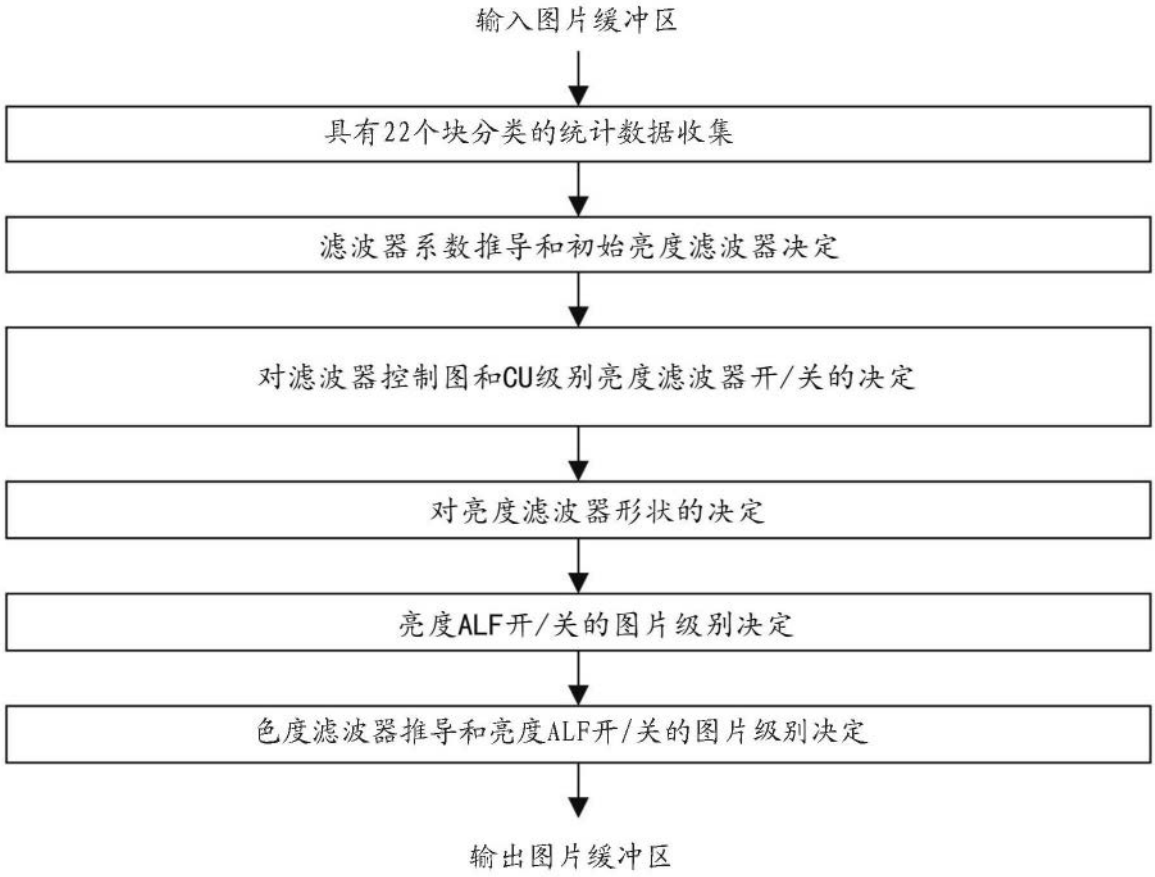

输入图片缓冲区
具有22个块分类的统计数据收集
滤波器系数推导和初始亮度滤波器决定
对滤波器控制图和CU级别亮度滤波器开/关的决定
对亮度滤波器形状的决定
亮度ALF开/关的图片级别决定
色度滤波器推导和亮度ALF开/关的图片级别决定
输出图片缓冲区

图3

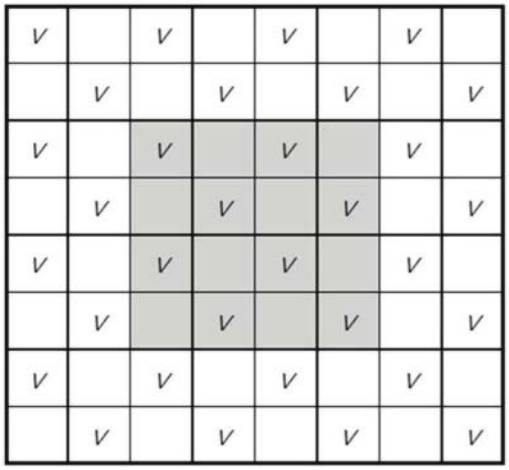


图4A

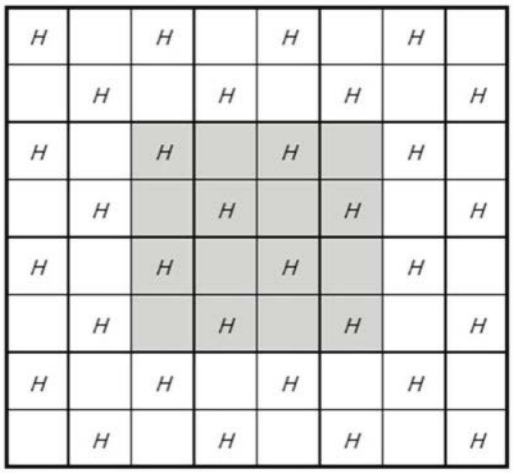


图4B

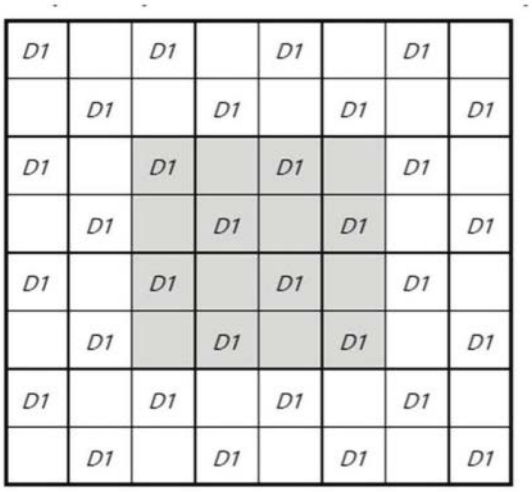


图4C

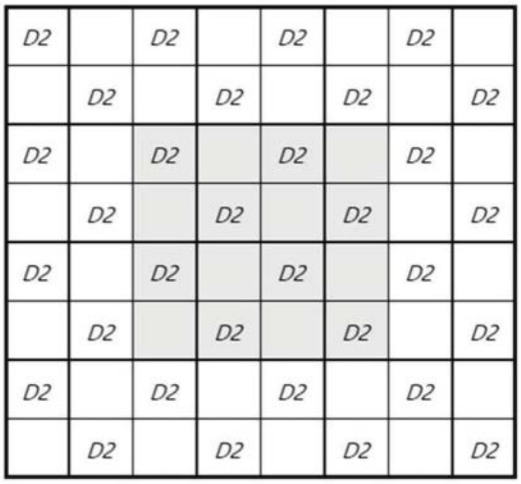


图4D

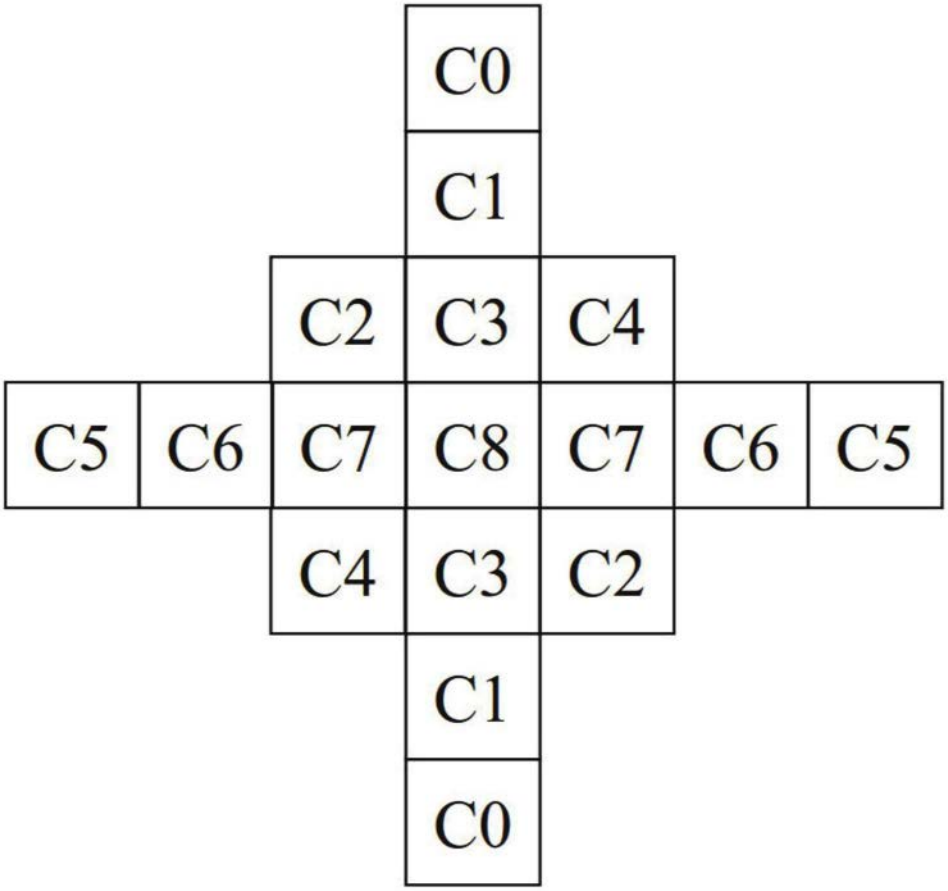


图5

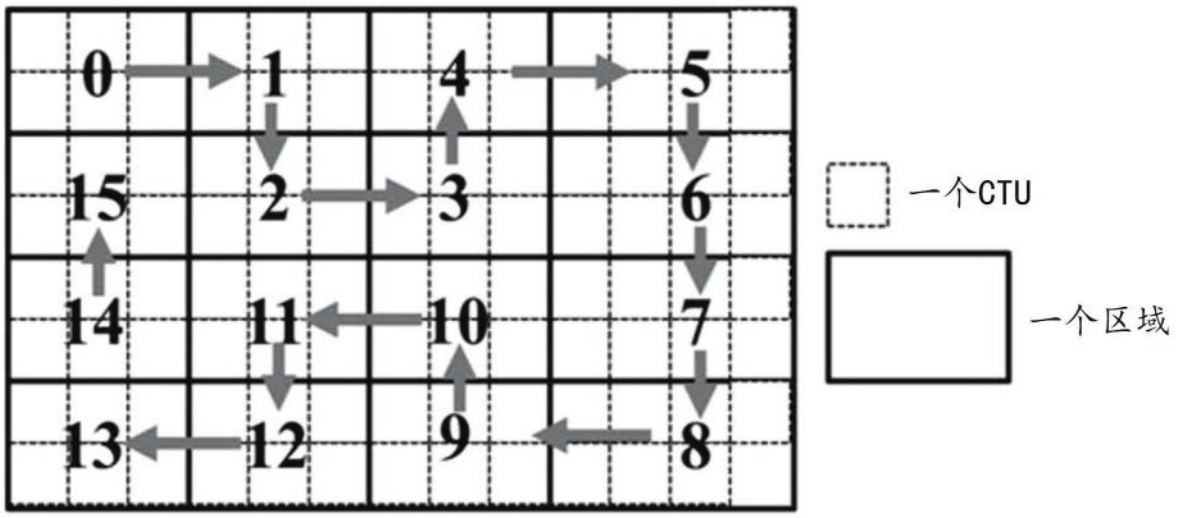

0
1
4
5
15
2
3
6
14
11
10
7
13
12
9
8
一个CTU
一个区域

图6

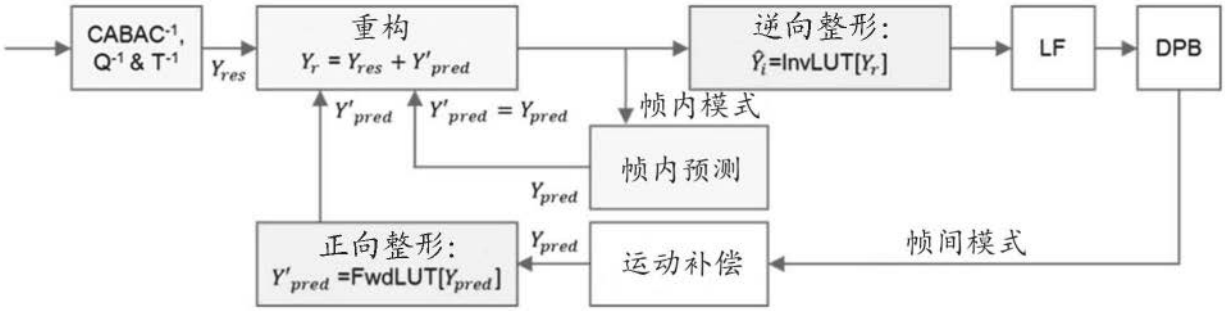

CABAC-1, Q-1 & T-1
Yres
重构
Yr = Yres + Y'pred
逆向整形:
Ŷi=InvLUT[Yr]
LF
DPB
Y'pred
Y'pred = Ypred
帧内模式
帧内预测
Ypred
正向整形:
Y'pred =FwdLUT[Ypred]
Ypred
运动补偿
帧间模式

图7

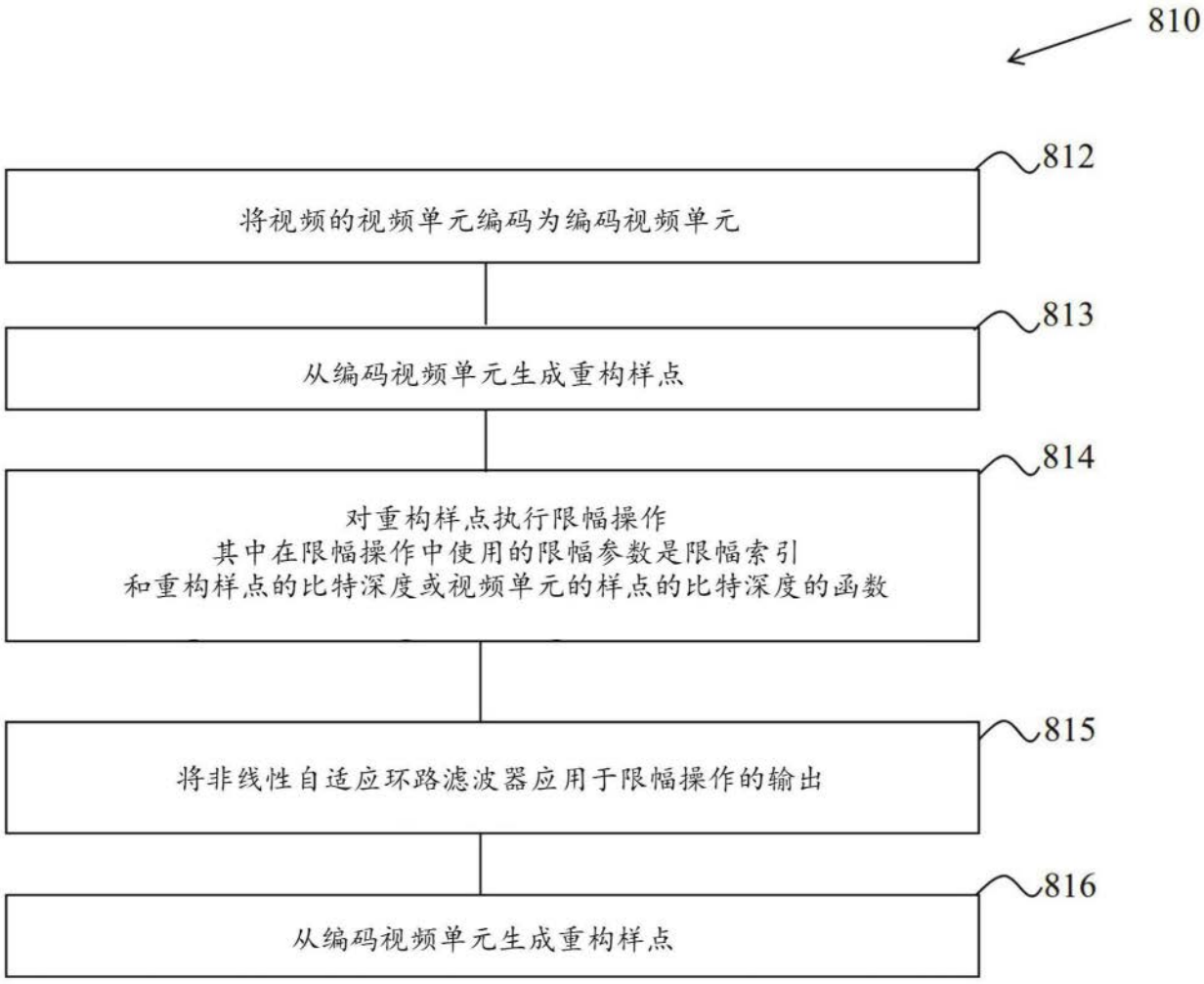

810
812
将视频的视频单元编码为编码视频单元
813
从编码视频单元生成重构样点
814
对重构样点执行限幅操作
其中在限幅操作中使用的限幅参数是限幅索引
和重构样点的比特深度或视频单元的样点的比特深度的函数
815
将非线性自适应环路滤波器应用于限幅操作的输出
816
从编码视频单元生成重构样点

图8A

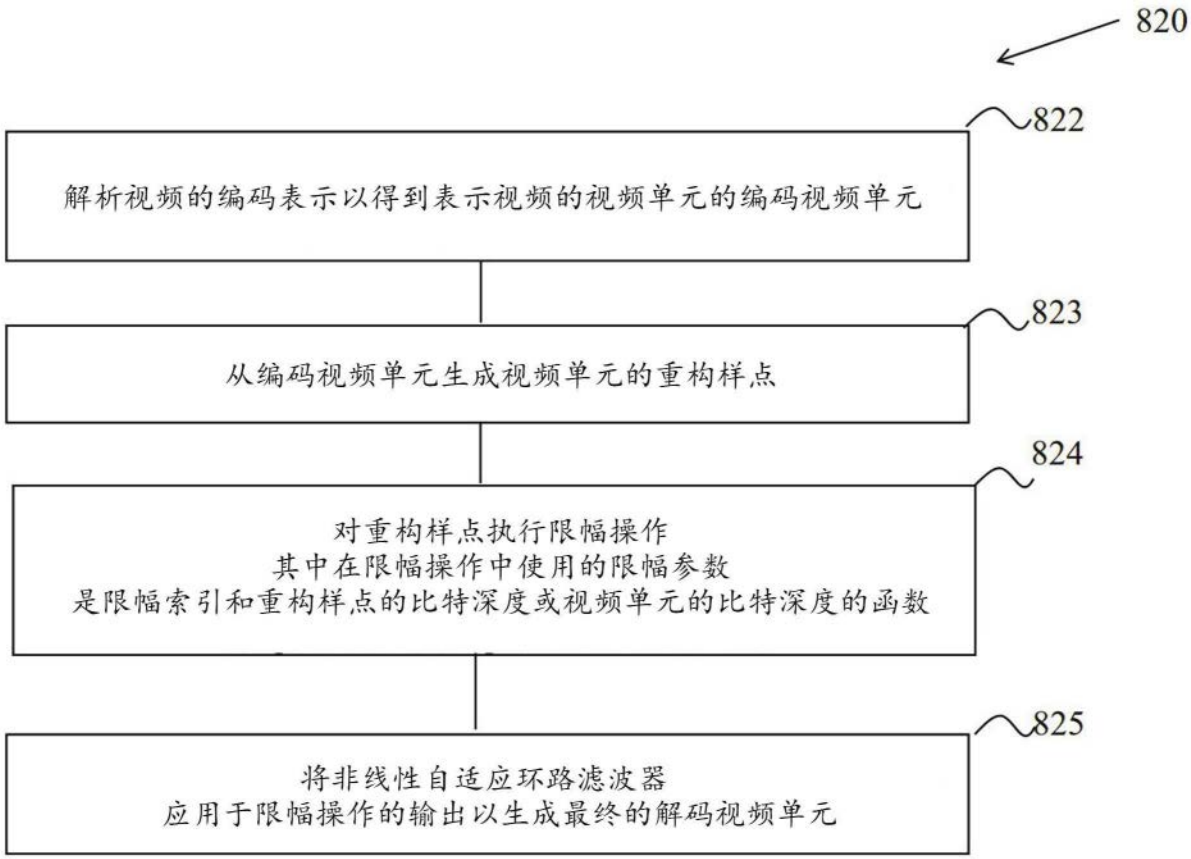

820
822
解析视频的编码表示以得到表示视频的视频单元的编码视频单元
823
从编码视频单元生成视频单元的重构样点
824
对重构样点执行限幅操作
其中在限幅操作中使用的限幅参数
是限幅索引和重构样点的比特深度或视频单元的比特深度的函数
825
将非线性自适应环路滤波器
应用于限幅操作的输出以生成最终的解码视频单元

图8B

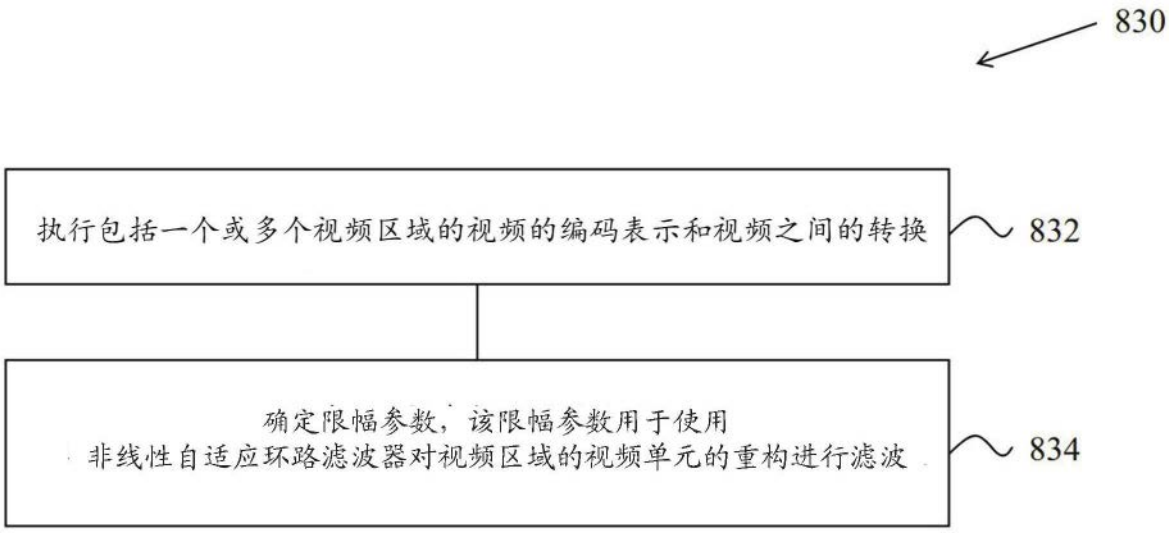

830
执行包括一个或多个视频区域的视频的编码表示和视频之间的转换
832
确定限幅参数，该限幅参数用于使用
非线性自适应环路滤波器对视频区域的视频单元的重构进行滤波
834

图8C

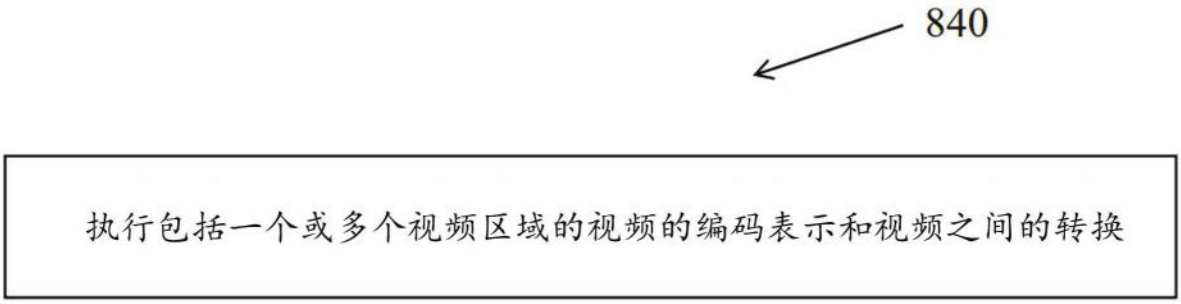

840
执行包括一个或多个视频区域的视频的编码表示和视频之间的转换

图9

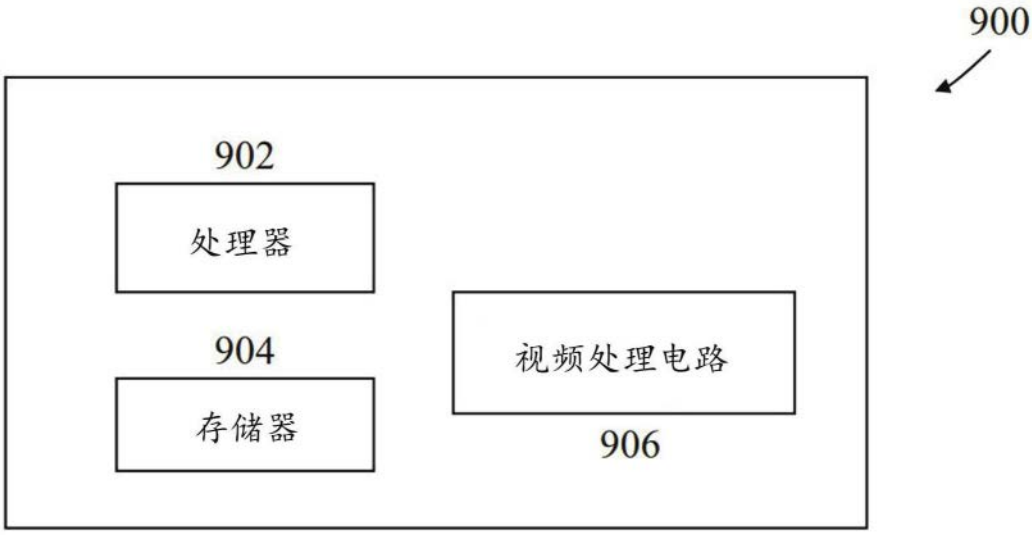

900
902
处理器
904
存储器
视频处理电路
906

图10A

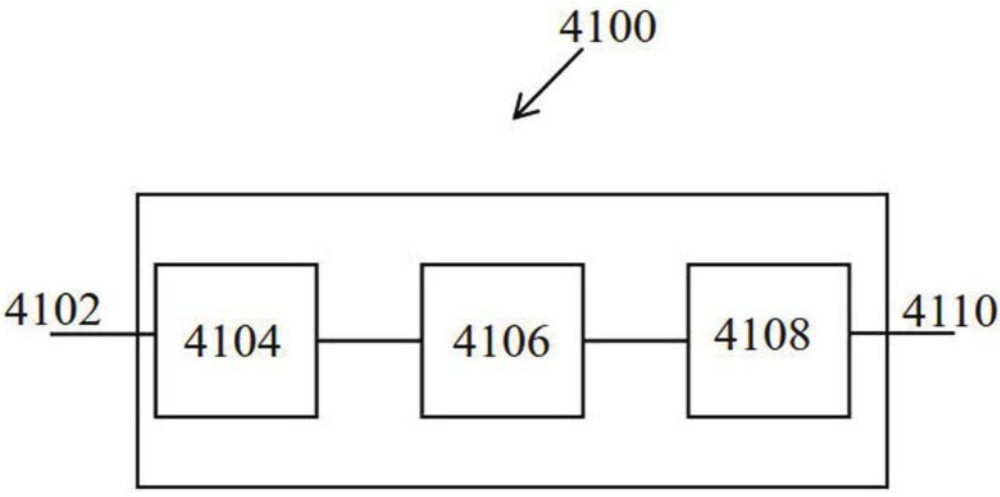

4100
4102
4104
4106
4108
4110

图10B