



(10) 授权公告号 CN 113557527 B

(45) 授权公告日 2024. 07. 23

(21) 申请号 202080020181.8

(22) 申请日 2020.03.12

(65) 同一申请的已公布的文献号  
申请公布号 CN 113557527 A

(43) 申请公布日 2021.10.26

(30) 优先权数据  
62/817,500 2019.03.12 US(85) PCT国际申请进入国家阶段日  
2021.09.10(86) PCT国际申请的申请数据  
PCT/US2020/022435 2020.03.12(87) PCT国际申请的公布数据  
W02020/186084 EN 2020.09.17(73) 专利权人 腾讯美国有限责任公司  
地址 美国加利福尼亚州帕洛阿尔托公园大道2747号

(72) 发明人 赵欣 许晓中 李翔 刘杉

(74) 专利代理机构 北京德琦知识产权代理有限公司 11018  
专利代理师 焦方佼 王琦

(51) Int.Cl.

H04N 19/52 (2014.01)

H04N 19/124 (2014.01)

H04N 19/18 (2014.01)

H04N 19/44 (2014.01)

H04N 19/85 (2014.01)

H04N 19/186 (2014.01)

(56) 对比文件

CN 108464006 A, 2018.08.28

US 2017318301 A1, 2017.11.02

审查员 谢小诗

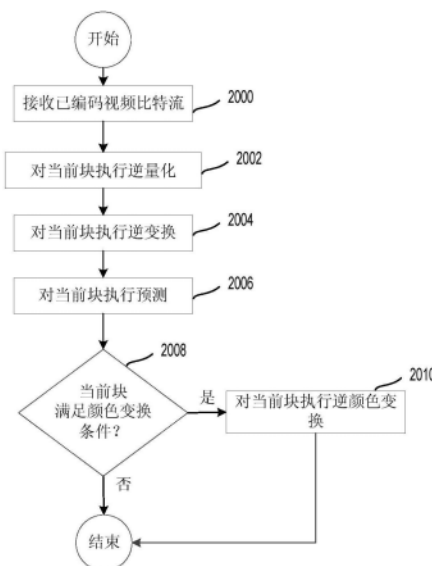
权利要求书2页 说明书24页 附图18页

## (54) 发明名称

视频解码方法、视频解码器及介质

## (57) 摘要

本公开实施例公开了一种视频解码方法、视频解码器及介质。所述方法包括接收包括当前图片的已编码视频比特流。所述方法进一步包括对所述当前图片中的当前块执行逆量化。所述方法进一步包括,在执行所述逆量化之后,对所述当前块执行逆变换。所述方法进一步包括,在执行所述逆变换之后,对所述当前块执行预测过程。所述方法进一步包括,对所述当前块执行预测过程之后,确定是否满足预定条件。所述方法进一步包括,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。



1. 一种视频解码方法,其特征在于,所述方法包括:  
接收包括当前图片的已编码视频比特流;  
对所述当前图片中的当前块执行逆量化;  
在执行所述逆量化之后,对所述当前块执行逆颜色变换;  
当确定对所述逆颜色变换的不同的颜色分量使用相同的编码单元分割树而不应用双树DualTree时,对所述当前块执行逆颜色变换,并且所述逆颜色变换应用于帧内编码块。
2. 根据权利要求1所述的方法,其特征在于,所述逆颜色变换是逆自适应颜色变换ACT,并且执行所述逆颜色变换将当前块从颜色空间变换转换为RGB格式。
3. 根据权利要求1所述的方法,其特征在于,执行所述逆颜色变换包括将常数减去所述逆颜色变换的一个或多个颜色分量。
4. 根据权利要求3所述的方法,其特征在于,所述常数是对输入样本的比特深度减1,再执行左移操作导出的。
5. 根据权利要求1所述的方法,其特征在于,所述方法进一步包括:  
对所述当前块执行所述逆颜色变换之前;  
确定是否满足预定条件;  
当满足所述预定条件时,确定发信号通知当前块进行颜色变换。
6. 根据权利要求5所述的方法,其特征在于,所述方法进一步包括:为具有最大编码单元CU的每个编码树单元CTU,发信号通知所述颜色变换。
7. 根据权利要求1所述的方法,其特征在于,所述方法进一步包括:当确定将双树应用于帧内切片时,将所述逆颜色变换应用于帧间切片。
8. 根据权利要求1所述的方法,其特征在于,进一步包括:  
确定是否满足预定条件;  
当确定满足所述预定条件时,不将跨分量线性模式CCLM应用于所述当前块的色度单元。
9. 一种视频解码器,其特征在于,包括处理电路,其中,所述处理电路用于:  
接收包括当前图片的已编码视频比特流;  
对所述当前图片中的当前块执行逆量化;  
在执行所述逆量化之后,对所述当前块执行逆颜色变换;  
当确定对所述逆颜色变换的不同的颜色分量使用相同的编码单元分割树而不应用双树DualTree时,对所述当前块执行逆颜色变换,并且所述逆颜色变换应用于帧内编码块。
10. 根据权利要求9所述的视频解码器,其特征在于,所述逆颜色变换是逆自适应颜色变换ACT,并且执行所述逆颜色变换将当前块从颜色空间变换转换为RGB格式。
11. 根据权利要求9所述的视频解码器,其特征在于,执行所述逆颜色变换包括所述处理电路将常数减去所述逆颜色变换的一个或多个颜色分量。
12. 根据权利要求11所述的视频解码器,其特征在于,所述常数是对输入样本的比特深度减1,再执行左移操作导出的。
13. 根据权利要求9所述的视频解码器,其特征在于,所述处理电路进一步用于:在对所述当前块执行逆颜色变换之前,确定是否满足预定条件;  
当确定满足所述预定条件时,发信号通知当前块进行颜色变换。

14. 根据权利要求13所述的视频解码器, 其特征在于, 所述处理电路进一步用于: 为具有最大编码单元CU的每个编码树单元CTU, 发信号通知所述颜色变换。

15. 根据权利要求9所述的视频解码器, 其特征在于, 所述处理电路进一步用于: 当确定将双树应用于帧内切片时, 将所述逆颜色变换应用于帧间切片。

16. 根据权利要求9至15任意一项所述的视频解码器, 其特征在于, 所述处理电路进一步用于: 确定是否满足预定条件; 当确定满足所述预定条件时, 不将跨分量线性模式CCLM应用于所述当前块的色度单元。

17. 一种视频编码方法, 其特征在于, 所述方法包括:

对当前图片中的当前块执行逆量化;

在执行所述逆量化之后, 当确定对逆颜色变换的不同的颜色分量使用相同的编码单元分割树而不应用双树DualTree时, 对所述当前块执行逆颜色变换, 并且所述逆颜色变换应用于帧内编码块。

18. 一种视频解码器, 其特征在于, 包括处理电路, 其中, 所述处理电路用于执行如权利要求1-8任一项所述的方法。

19. 一种计算机设备, 其特征在于, 包括一个或多个处理器和一个或多个存储器, 所述一个或多个存储器中存储有至少一条指令, 所述至少一条指令由所述一个或多个处理器加载并执行, 以实现如权利要求1至8和权利要求17任一项所述的方法。

20. 一种非易失性计算机可读介质, 其特征在于, 所述非易失性计算机可读介质存储指令, 当所述指令被视频解码器中的处理器执行时, 使得所述视频解码器执行如权利要求1至8任意一项所述的方法。

21. 一种存储计算机程序的计算机可读存储介质, 所述计算机程序被处理器执行如权利要求17所述的视频编码的方法以形成比特流, 所述比特流存储在所述计算机可读存储介质中。

## 视频解码方法、视频解码器及介质

[0001] 引用并入

[0002] 本公开要求于2020年3月12日提交的美国申请第16/817,028号“通用视频编解码中颜色转换的方法和装置”的优先权,该美国申请要求于2019年3月12日提交的美国临时申请第62/817,500号“通用视频编解码中色彩转换”的优先权。在先申请的全部公开内容通过引用整体并入本文。

### 技术领域

[0003] 本公开描述总体上涉及视频编解码的实施例。

### 背景技术

[0004] 本文所提供的背景描述旨在整体呈现本公开的背景。在背景技术部分以及本说明书的各个方面中所描述的目前已署名的发明人的工作所进行的程度,并不表明其在本公开提交时作为现有技术,且从未明示或暗示其被承认为本公开的现有技术。

[0005] 通过具有运动补偿的帧间图片预测技术,可以进行视频编码和解码。未压缩的数字视频可包括一系列图片,每个图片具有例如 $1920 \times 1080$ 亮度样本及相关色度样本的空间维度。所述系列图片具有固定的或可变的图片速率(也非正式地称为帧率),例如每秒60个图片或60Hz。未压缩的视频具有非常大的比特率要求。例如,每个样本8比特的1080p60 4:2:0的视频( $1920 \times 1080$ 亮度样本分辨率,60Hz帧率)要求接近1.5Gbit/s带宽。一小时这样的视频就需要超过600GB的存储空间。

[0006] 视频编码和解码的一个目的,是通过压缩减少输入视频信号的冗余信息。视频压缩可以帮助降低对上述带宽或存储空间的要求,在某些情况下可降低两个或更多数量级。无损和有损压缩,以及两者的组合均可采用。无损压缩是指从压缩的原始信号中重建原始信号精确副本的技术。当使用有损压缩时,重建信号可能与原始信号不完全相同,但是原始信号和重建信号之间的失真足够小,使得重建信号可用于预期应用。有损压缩广泛应用于视频。容许的失真量取决于应用。例如,相比于电视应用的用户,某些消费流媒体应用的用户可以容忍更高的失真。可实现的压缩比反映出:较高的允许/容许失真可产生较高的压缩比。

[0007] 视频编码器和解码器可利用几大类技术,例如包括:运动补偿、变换、量化和熵编码。

[0008] 视频编解码器技术可包括已知的帧内编码技术。在帧内编码中,在不参考先前重建的参考图片的样本或其它数据的情况下表示样本值。在一些视频编解码器中,图片在空间上被细分为样本块。当所有的样本块都以帧内模式编码时,该图片可以为帧内图片。帧内图片及其衍生(例如独立解码器刷新图片)可用于复位解码器状态,并且因此可用作编码视频比特流和视频会话中的第一图片,或用作静止图像。帧内块的样本可用于变换,且可在熵编码之前量化变换系数。帧内预测可以是使预变换域中的样本值最小化的技术。在某些情形下,变换后的DC值越小,且AC系数越小,则在给定的量化步长尺寸下需要越少的比特来表

示熵编码之后的块。

[0009] 如同从诸如MPEG-2代编码技术中所获知的,传统帧内编码不使用帧内预测。然而,一些较新的视频压缩技术包括:试图从例如周围样本数据和/或元数据中得到数据块的技术,其中周围样本数据和/或元数据是在空间相邻的编码/解码期间、且在解码顺序之前获得的。这种技术后来被称为“帧内预测”技术。需要注意的是,至少在某些情形下,帧内预测仅使用正在重建的当前图片的参考数据,而不使用参考图片的参考数据。

[0010] 可以存在许多不同形式的帧内预测。当在给定的视频编码技术中可以使用超过一种这样的技术时,所使用的技术可以按帧内预测模式进行编码。在某些情形下,模式可具有子模式和/或参数,且这些模式可单独编码或包含在模式码字中。将哪个码字用于给定模式/子模式/参数组合会通过帧内预测影响编码效率增益,因此用于将码字转换成比特流的熵编码技术也会出现这种情况。

[0011] H.264引入了一种帧内预测模式,其在H.265中进行了改进,且在诸如联合开发模型(JEM)/通用视频编码(VVC)/基准集合(BMS)的更新的编码技术中进一步被改进。通过使用属于已经可用的样本的相邻样本值可以形成预测块。在一些示例中,将相邻样本的样本值按照某一方向复制到预测块中。对所使用方向的引用可以被编码在比特流中,或者本身可以被预测。

[0012] 参照图1A,右下方描绘了来自H.265的33个可能的预测方向(对应于35个帧内模式的33个角模式)中已知的九个预测方向的子集。箭头会聚的点(101)表示正在被预测的样本。箭头表示样本正在被预测的方向。例如,箭头(102)表示根据右上方与水平方向成45度角的一个或多个样本,预测样本(101)。类似地,箭头(103)表示根据左下方与水平方向成22.5度角的一个或多个样本,预测样本(101)。

[0013] 仍然参考图1A,在左上方示出了一个包括 $4 \times 4$ 个样本的正方形块(104)(由粗虚线表示)。正方形块(104)包括16个样本,每个样本用“S”、以及其在Y维度上的位置(例如,行索引)和在X维度上的位置(例如,列索引)来标记。例如,样本S21是Y维度上的第二个样本(从顶部开始)和X维度上的第一个样本(从左侧开始)。类似地,样本S44在Y维度和X维度上都是块(104)中的第四个样本。由于该块为 $4 \times 4$ 大小的样本,因此S44位于右下角。还示出了遵循类似编号方案的参考样本。参考样本用R、以及其相对于块(104)的Y位置(例如,行索引)和X位置(例如,列索引)来标记。在H.264与H.265中,预测样本与正在重建的块相邻,因此不需要使用负值。

[0014] 通过从信号通知的预测方向所占用的相邻样本来复制参考样本值,可以进行帧内图片预测。例如,假设编码视频比特流包括信令,对于该块,该信令指示与箭头(102)一致的预测方向,即,根据右上方与水平方向成45度角的一个或多个预测样本来预测样本。在这种情况下,根据同一参考样本R05来预测样本S41、S32、S23和S14。然后,根据参考样本R08来预测样本S44。

[0015] 在某些情况下,例如通过内插,可以合并多个参考样本的值,以便计算参考样本,尤其是当方向不能被45度整除时。

[0016] 随着视频编码技术的发展,可能的方向的数量已经增加了。在H.264(2003年)中,可以表示九种不同的方向。在H.265(2013年)和JEM/VVC/BMS中增加到了33个,而在此申请时,可以支持多达65个方向。已经进行了实验来识别最可能的方向,并且熵编码中的某些技

术被用于使用少量比特来表示那些可能的方向,对于较不可能的方向则接受某些代价。此外,有时可以根据在相邻的、已经解码的块中所使用的相邻方向来预测方向本身。

[0017] 在图1B中示出了在HEVC中使用的多个帧内预测模式。在HEVC中,共有35种帧内预测模式。其中,模式10是水平模式。模式26是垂直模式。并且,模式2,18和34是对角线模式。帧内预测模式由三种最可能的模式(Most Probable Mode,MPM)和其余32种模式发出信号。

[0018] 图1C示出了在VVC中使用的多个帧内预测模式。在VVC中,如图1C所示,共有95种帧内预测模式。其中,模式18是水平模式。模式50是垂直模式。并且,模式2,34和66是对角线模式。模式-1~14和模式67~80被称为广角帧内预测(Wide-Angle Intra Prediction,WAIP)模式。

[0019] 表示方向的已编码视频比特流中的帧内预测方向比特的映射可以因视频编码技术的不同而不同,并且,例如可以从对帧内预测模式到码字的预测方向的简单直接映射,到包括最可能的模式(Most Probable Modes,MPMs)和类似技术的复杂的自适应方案。然而,在所有情况下,视频内容中可能存在某些方向,其在统计学上比其它方向更不可能出现。由于视频压缩的目的是减少冗余,所以在运行良好的视频编码技术中,与更可能的方向相比,那些不太可能的方向将使用更多数量的比特来表示。

[0020] 通过具有运动补偿的帧间图片预测技术,可以进行视频编码和解码。未压缩的数字视频可包括一系列图片,每个图片具有例如 $1920 \times 1080$ 亮度样本及相关色度样本的空间维度。所述系列图片具有固定的或可变的图片速率(也非正式地称为帧率),例如每秒60个图片或60Hz。未压缩的视频具有非常大的比特率要求。例如,每个样本8比特的1080p60 4:2:0的视频( $1920 \times 1080$ 亮度样本分辨率,60Hz帧率)要求接近1.5Gbit/s带宽。一小时这样的视频就需要超过600GB的存储空间。

[0021] 运动补偿可以是一种有损压缩技术,且可涉及如下技术:来自先前重建的图片或重建图片一部分(参考图片)的样本数据块在空间上按运动矢量(下文称为MV)指示的方向移位后,用于新重建的图片或图片部分的预测。在某些情况下,参考图片可与当前正在重建的图片相同。MV可具有两个维度X和Y,或者三个维度,其中第三个维度表示使用中的参考图片(后者间接地可为时间维度)。

[0022] 在一些视频压缩技术中,应用于某个样本数据区域的MV可根据其它MV来预测,例如根据与正在重建的区域空间相邻的另一个样本数据区域相关的、且按解码顺序在该MV前面的那些MV。这样做可以大大减少编码MV所需的数据量,从而消除冗余信息并增加压缩量。MV预测可以有效地进行,例如,当对从相机导出的输入视频信号(称为自然视频)进行编码时,存在一种统计上的可能性,即面积大于单个MV适用区域的区域,会朝着类似的方向移动,因此,在某些情况下,可以用相邻区域的MV导出的相似运动矢量进行预测。这导致针对给定区域发现的MV与根据周围MV预测的MV相似或相同,并且在熵编码之后,又可以用比直接编码MV时使用的比特数更少的比特数来表示。在某些情况下,MV预测可以是对从原始信号(即样本流)导出的信号(即MV)进行无损压缩的示例。在其它情况下,MV预测本身可能是有损的,例如由于根据几个周围MV计算预测值时产生的取整误差。

[0023] H.265/HEVC(ITU-T H.265建议书,“高效视频编解码(High Efficiency Video Coding)”,2016年12月)中描述了各种MV预测机制。在H.265提供的多种MV预测机制中,本公开描述的是下文称作“空间合并”的技术。

[0024] 请参考图1D,当前块(101)包括在运动搜索过程期间已由编码器发现的样本,根据已产生空间偏移的相同大小的先前块,可预测所述样本。另外,可从一个或多个参考图片相关联的元数据中导出所述MV,而非对MV直接编码。例如,使用关联于A0、A1和B0、B1、B2(分别对应102到106)五个周围样本中的任一样本的MV,(按解码次序)从最近的参考图片的元数据中导出所述MV。在H.265中,MV预测可使用相邻块也正在使用的相同参考图片的预测值。形成候选列表的顺序可以是A0→B0→B1→A1→B2。

## 发明内容

[0025] 根据示例性实施例,提出了一种在视频解码器中执行的视频解码方法。所述方法包括接收包括当前图片的已编码视频比特流。所述方法进一步包括对所述当前图片中的当前块执行逆量化。所述方法进一步包括,在执行所述逆量化之后,对所述当前块执行逆变换。所述方法进一步包括,在执行所述逆变换之后,对所述当前块执行预测过程。所述方法进一步包括,对所述当前块执行预测过程之后,确定是否满足预定条件。所述方法进一步包括,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

[0026] 根据示例性实施例,提出了一种用于视频解码的视频解码器,其中,该视频解码器包括处理电路,所述处理电路用于接收包括当前图片的已编码视频比特流。所述处理电路进一步用于对所述当前图片中的当前块执行逆量化。所述处理电路进一步用于在执行所述逆量化之后,对所述当前块执行逆变换。所述处理电路进一步用于在执行所述逆变换之后,对所述当前块执行预测过程。所述处理电路进一步用于,对所述当前块执行预测过程之后,确定是否满足预定条件。所述处理电路进一步用于,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

[0027] 根据示例性实施例,提出了一种非易失性计算机可读介质,所述非易失性计算机可读介质存储指令,当所述指令被视频解码器中的处理器执行时,使得所述视频解码器执行一个方法,其中,该方法包括:接收包括当前图片的已编码视频比特流。所述方法进一步包括,对所述当前图片中的当前块执行逆量化。所述方法进一步包括,在执行所述逆量化之后,对所述当前块执行逆变换。所述方法进一步包括,在执行所述逆变换之后,对所述当前块执行预测过程。所述方法进一步包括,对所述当前块执行预测过程之后,确定是否满足预定条件。所述方法进一步包括,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

## 附图说明

[0028] 根据以下详细描述和附图,所公开的主题的其他特征、性质和各种优点将进一步明确,其中:

[0029] 图1A示出了帧内预测模式的示例性子集的示意图;

[0030] 图1B示出了示例性帧内预测方向的示意图。

[0031] 图1C示出了示例性帧内预测方向的示意图。

[0032] 图1D示出了一个示例中的当前块及其周围的空间合并候选的示意图。

[0033] 图2示出了根据一实施例的通信系统的简化框图的示意图。

[0034] 图3示出了根据另一实施例的通信系统的简化框图的示意图。

- [0035] 图4示出了根据一实施例的解码器的简化框图的示意图。
- [0036] 图5示出了根据一实施例的编码器的简化框图的示意图。
- [0037] 图6示出了根据另一实施例的编码器的框图。
- [0038] 图7示出了根据另一实施例的解码器的框图。
- [0039] 图8A示出了根据实施例的块分区。
- [0040] 图8B示出了根据实施例的块分割树。
- [0041] 图9A示出了根据实施例的垂直中心侧三叉树分区。
- [0042] 图9B示出了根据实施例的水平中心侧三叉树分区。
- [0043] 图10A-10D示出了根据各种实施例的多个不同色度格式。
- [0044] 图11示出了根据实施例的示例编码器。
- [0045] 图12示出了根据实施例的示例解码器。
- [0046] 图13示出了最小亮度值和最大亮度值之间的直线。
- [0047] 图14A和图14B示出了根据实施例的LT\_CCLM中用于推导 $\alpha$ 和 $\beta$ 的样本的位置。
- [0048] 图15A和图15B示出了根据实施例的T\_CCLM中用于推导 $\alpha$ 和 $\beta$ 的样本的位置。
- [0049] 图16A和图16B示出了根据实施例的L\_CCLM中用于推导 $\alpha$ 和 $\beta$ 的样本的位置。
- [0050] 图17示出了根据实施例将相邻样本分为两组的示例。
- [0051] 图18示出了根据实施例的编码器和解码器的示意图。
- [0052] 图19示出了由编码器执行的过程的实施例示意图。
- [0053] 图20示出了由编码器执行的过程的实施例示意图。
- [0054] 图21示出了根据本公开实施例的计算机系统的示意图。

## 具体实施方式

[0055] 图2是根据本公开实施例的通信系统(200)的简化框图。通信系统(200)包括多个终端装置,所述多个终端装置可通过例如网络(250)彼此通信。举例来说,通信系统(200)包括通过网络(250)互连的第一对终端装置(210)和终端装置(220)。在图2的示例中,第一对终端装置(210)和终端装置(220)执行单向数据传输。举例来说,终端装置(210)可对视频数据(例如由终端装置(210)采集的视频图片流)进行编码,以通过网络(250)传输到另一个终端装置(220)。已编码的视频数据以一个或多个已编码视频码流形式传输。终端装置(220)可从网络(250)接收已编码视频数据,对已编码视频数据进行解码以恢复视频数据,并根据恢复的视频数据显示视频图片。单向数据传输在媒体服务等应用中是较常见的。

[0056] 在另一实施例中,通信系统(200)包括执行已编码视频数据的双向传输的第二对终端装置(230)和终端装置(240),所述双向传输可例如在视频会议期间发生。对于双向数据传输,在一个示例中,终端装置(230)和终端装置(240)中的每个终端装置可对视频数据(例如由终端装置采集的视频图片流)进行编码,以通过网络(250)传输到终端装置(230)和终端装置(240)中的另一终端装置。终端装置(230)和终端装置(240)中的每个终端装置还可接收由终端装置(230)和终端装置(240)中的另一终端装置传输的已编码视频数据,且可对所述已编码视频数据进行解码以恢复视频数据,且可根据恢复的视频数据在可访问的显示装置上显示视频图片。

[0057] 在图2的示例中,终端装置(210)、终端装置(220)、终端装置(230)和终端装置



(240)可为服务器、个人计算机和智能电话,但本公开的原理可不限于此。本公开的实施例适用于膝上型计算机、平板电脑、媒体播放器和/或专用视频会议设备。网络(250)表示在终端装置(210)、终端装置(220)、终端装置(230)和终端装置(240)之间传送已编码视频数据的任何数目的网络,包括例如有线(连线的)和/或无线通信网络。通信网络(250)可在电路交换和/或分组交换信道中交换数据。该网络可包括电信网络、局域网、广域网和/或互联网。出于本公开的目的,除非在下文中有所解释,否则网络(250)的架构和拓扑对于本公开的操作来说可能是无关紧要的。

[0058] 作为实施例,图3示出视频编码器和视频解码器在流式传输环境中的放置方式。本公开所公开主题可同等地适用于其它支持视频的应用,包括例如视频会议、数字TV、在包括CD、DVD、存储棒等的数字介质上存储压缩视频等等,以此类推。

[0059] 流式传输系统可包括采集子系统(313),所述采集子系统可包括数码相机等视频源(301),所述视频源创建未压缩的视频图片流(302)。在实施例中,视频图片流(302)包括由数码相机拍摄的样本。相较于已编码的视频数据(304)(或已编码的视频码流),视频图片流(302)被描绘为粗线以强调高数据量的视频图片流,视频图片流(302)可由电子装置(320)处理,所述电子装置(320)包括耦接到视频源(301)的视频编码器(303)。视频编码器(303)可包括硬件、软件或软硬件组合以实现或实施如下文更详细地描述的所公开主题的各方面。相较于视频图片流(302),已编码的视频数据(304)(或已编码的视频码流(304))被描绘为细线以强调较低数据量的已编码的视频数据(304)(或已编码的视频码流(304)),其可存储在流式传输服务器(305)上以供将来使用。一个或多个流式传输客户端子系统,例如图3中的客户端子系统(306)和客户端子系统(308),可访问流式传输服务器(305)以检索已编码的视频数据(304)的副本(307)和副本(309)。客户端子系统(306)可包括例如电子装置(330)中的视频解码器(310)。视频解码器(310)对已编码的视频数据的传入副本(307)进行解码,且产生可在显示器(312)(例如显示屏)或另一呈现装置(未描绘)上呈现的输出视频图片流(311)。在一些流式传输系统中,可根据某些视频编码/压缩标准对已编码的视频数据(304)、视频数据(307)和视频数据(309)(例如视频码流)进行编码。该些标准的实施例包括ITU-T H.265。在实施例中,正在开发的视频编码标准非正式地称为下一代视频编码(Versatile Video Coding,VVC),本公开可用于VVC标准的上下文中。

[0060] 应注意,电子装置(320)和电子装置(330)可包括其它组件(未示出)。举例来说,电子装置(320)可包括视频解码器(未示出),且电子装置(330)还可包括视频编码器(未示出)。

[0061] 图4是根据本公开的实施例的视频解码器(410)的框图。视频解码器(410)可设置在电子装置(430)中。电子装置(430)可包括接收器(431)(例如接收电路)。视频解码器(410)可用于代替图3实施例中的视频解码器(310)。

[0062] 接收器(431)可接收将由视频解码器(410)解码的一个或多个已编码视频序列;在同一实施例或另一实施例中,一次接收一个已编码视频序列,其中每个已编码视频序列的解码独立于其它已编码视频序列。可从信道(401)接收已编码视频序列,所述信道可以是通向存储已编码的视频数据的存储装置的硬件/软件链路。接收器(431)可接收已编码的视频数据以及其它数据,例如,可转发到它们各自的使用实体(未标示)的已编码音频数据和/或辅助数据流。接收器(431)可将已编码视频序列与其它数据分开。为了防止网络抖动,缓冲

存储器(415)可耦接在接收器(431)与熵解码器/解析器(420)(此后称为“解析器(420)”)之间。在某些应用中,缓冲存储器(415)是视频解码器(410)的一部分。在其它情况下,所述缓冲存储器(415)可设置在视频解码器(410)外部(未标示)。而在其它情况下,视频解码器(410)的外部设置缓冲存储器(未标示)以例如防止网络抖动,且在视频解码器(410)的内部可配置另一缓冲存储器(415)以例如处理播出定时。而当接收器(431)从具有足够带宽和可控性的存储/转发装置或从等时同步网络接收数据时,也可能不需要配置缓冲存储器(415),或可以将所述缓冲存储器做得较小。当然,为了在互联网等业务分组网络上使用,也可能需要缓冲存储器(415),所述缓冲存储器可相对较大且可具有自适应性大小,且可至少部分地实施于操作系统或视频解码器(410)外部的类似元件(未标示)中。

[0063] 视频解码器(410)可包括解析器(420)以根据已编码视频序列重建符号(421)。这些符号的类别包括用于管理视频解码器(410)的操作的信息,以及用以控制显示装置(412)(例如,显示屏)等显示装置的潜在信息,所述显示装置不是电子装置(430)的组成部分,但可耦接到电子装置(430),如图4中所示。用于显示装置的控制信息可以是辅助增强信息(Supplemental Enhancement Information, SEI消息)或视频可用性信息(Video Usability Information, VUI)的参数集片段(未标示)。解析器(420)可对接收到的已编码视频序列进行解析/熵解码。已编码视频序列的编码可根据视频编码技术或标准进行,且可遵循各种原理,包括可变长度编码、霍夫曼编码(Huffman coding)、具有或不具有上下文灵敏度的算术编码等等。解析器(420)可基于对应于群组的至少一个参数,从已编码视频序列提取用于视频解码器中的像素的子群中的至少一个子群的子群参数集。子群可包括图片群组(Group of Pictures, GOP)、图片、图块、切片、宏块、编码单元(Coding Unit, CU)、块、变换单元(Transform Unit, TU)、预测单元(Prediction Unit, PU)等等。解析器(420)还可从已编码视频序列提取信息,例如变换系数、量化器参数值、运动矢量等等。

[0064] 解析器(420)可对从缓冲存储器(415)接收的视频序列执行熵解码/解析操作,从而创建符号(421)。

[0065] 取决于已编码视频图片或一部分已编码视频图片(例如:帧间图片和帧内图片、帧间块和帧内块)的类型以及其它因素,符号(421)的重建可涉及多个不同单元。涉及哪些单元以及涉及方式可由解析器(420)从已编码视频序列解析的子群控制信息控制。为了简洁起见,未描述解析器(420)与下文的多个单元之间的此类子群控制信息流。

[0066] 除已经提及的功能块以外,视频解码器(410)可在概念上细分成如下文所描述的数个功能单元。在商业约束下运行的实际实施例中,这些单元中的许多单元彼此紧密交互并且可以彼此集成。然而,出于描述所公开主题的目的,概念上细分成下文的功能单元是适当的。

[0067] 第一单元是缩放器/逆变换单元(451)。缩放器/逆变换单元(451)从解析器(420)接收作为符号(421)的量化变换系数以及控制信息,包括使用哪种变换方式、块大小、量化因子、量化缩放矩阵等。缩放器/逆变换单元(451)可输出包括样本值的块,所述样本值可输入到聚合器(455)中。

[0068] 在一些情况下,缩放器/逆变换单元(451)的输出样本可属于帧内编码块;即:不使用来自先前重建的图片的预测性信息,但可使用来自当前图片的先前重建部分的预测性信息的块。此类预测性信息可由帧内图片预测单元(452)提供。在一些情况下,帧内图片预测

单元(452)采用从当前图片缓冲器(458)提取的已重建信息生成大小和形状与正在重建的块相同的周围块。举例来说,当前图片缓冲器(458)缓冲部分重建的当前图片和/或完全重建的当前图片。在一些情况下,聚合器(455)基于每个样本,将帧内预测单元(452)生成的预测信息添加到由缩放器/逆变换单元(451)提供的输出样本信息中。

[0069] 在其它情况下,缩放器/逆变换单元(451)的输出样本可属于帧间编码和潜在运动补偿块。在此情况下,运动补偿预测单元(453)可访问参考图片存储器(457)以提取用于预测的样本。在根据符号(421)对提取的样本进行运动补偿之后,这些样本可由聚合器(455)添加到缩放器/逆变换单元(451)的输出(在这种情况下被称作残差样本或残差信号),从而生成输出样本信息。运动补偿预测单元(453)从参考图片存储器(457)内的地址获取预测样本可受到运动矢量控制,且所述运动矢量以所述符号(421)的形式而供运动补偿预测单元(453)使用,所述符号(421)例如是包括X、Y和参考图片分量。运动补偿还可包括在使用子样本精确运动矢量时,从参考图片存储器(457)提取的样本值的内插、运动矢量预测机制等等。

[0070] 聚合器(455)的输出样本可在环路滤波器单元(456)中被各种环路滤波技术采用。视频压缩技术可包括环路内滤波器技术,所述环路内滤波器技术受控于包括在已编码视频序列(也称作已编码视频码流)中的参数,且所述参数作为来自解析器(420)的符号(421)可用于环路滤波器单元(456)。然而,在其他实施例中,视频压缩技术还可响应于在解码已编码图片或已编码视频序列的先前(按解码次序)部分期间获得的元信息,以及响应于先前重建且经过环路滤波的样本值。

[0071] 环路滤波器单元(456)的输出可以是样本流,所述样本流可输出到显示装置(412)以及存储在参考图片存储器(457),以用于后续的帧间图片预测。

[0072] 一旦完全重建,某些已编码图片就可用作参考图片以用于将来预测。举例来说,一旦对应于当前图片的已编码图片被完全重建,且已编码图片(通过例如解析器(420))被识别为参考图片,则当前图片缓冲器(458)可变为参考图片存储器(457)的一部分,且可在开始重建后续已编码图片之前重新分配新的当前图片缓冲器。

[0073] 视频解码器(410)可根据例如ITU-T H.265标准中的预定视频压缩技术执行解码操作。在已编码视频序列遵循视频压缩技术或标准的语法以及视频压缩技术或标准中记录的配置文件的意义下,已编码视频序列可符合所使用的视频压缩技术或标准指定的语法。具体地说,配置文件可从视频压缩技术或标准中可用的所有工具中选择某些工具作为在所述配置文件下可供使用的仅有工具。对于合规性,还要求已编码视频序列的复杂度处于视频压缩技术或标准的层级所限定的范围内。在一些情况下,层级限制最大图片大小、最大帧率、最大重建取样率(以例如每秒兆(mega)个样本为单位进行测量)、最大参考图片大小等。在一些情况下,由层级设定的限制可通过假想参考解码器(Hypothetical Reference Decoder,HRD)规范和在已编码视频序列中用信号表示的HRD缓冲器管理的元数据来进一步限定。

[0074] 在实施例中,接收器(431)可连同已编码视频一起接收附加(冗余)数据。所述附加数据可以是已编码视频序列的一部分。所述附加数据可由视频解码器(410)用以对数据进行适当解码和/或较准确地重建原始视频数据。附加数据可呈例如时间、空间或信噪比(signal noise ratio,SNR)增强层、冗余切片、冗余图片、前向纠错码等形式。

[0075] 图5是根据本公开实施例的视频编码器(503)的框图。视频编码器(503)设置于电子装置(520)中。电子装置(520)包括传输器(540)(例如传输电路)。视频编码器(503)可用于代替图3实施例中的视频编码器(303)。

[0076] 视频编码器(503)可从视频源(501)(并非图5实施例中的电子装置(520)的一部分)接收视频样本,所述视频源可采集将由视频编码器(503)编码的视频图像。在另一实施例中,视频源(501)是电子装置(520)的一部分。

[0077] 视频源(501)可提供将由视频编码器(503)编码的呈数字视频样本流形式的源视频序列,所述数字视频样本流可具有任何合适位深度(例如:8位、10位、12位……)、任何色彩空间(例如BT.601Y CrCb、RGB……)和任何合适取样结构(例如Y CrCb 4:2:0、Y CrCb 4:4:4)。在媒体服务系统中,视频源(501)可以是存储先前已准备的视频的存储装置。在视频会议系统中,视频源(501)可以是采集本地图像信息作为视频序列的相机。可将视频数据提供为多个单独的图片,当按顺序观看时,这些图片被赋予运动。图片自身可构建为空间像素阵列,其中取决于所用的取样结构、色彩空间等,每个像素可包括一个或多个样本。所属领域的技术人员可以很容易理解像素与样本之间的关系。下文侧重于描述样本。

[0078] 根据实施例,视频编码器(503)可实时或在由应用所要求的任何其它时间约束下,将源视频序列的图片编码且压缩成已编码视频序列(543)。施行适当的编码速度是控制器(550)的一个功能。在一些实施例中,控制器(550)控制如下文所描述的其它功能单元且在功能上耦接到这些单元。为了简洁起见,图中未标示耦接。由控制器(550)设置的参数可包括速率控制相关参数(图片跳过、量化器、率失真优化技术的 $\lambda$ 值等)、图片大小、图片群组(group of pictures, GOP)布局,最大运动矢量搜索范围等。控制器(550)可用于具有其它合适的功能,这些功能涉及针对某一系统设计优化的视频编码器(503)。

[0079] 在一些实施例中,视频编码器(503)在编码环路中进行操作。作为简单的描述,在实施例中,编码环路可包括源编码器(530)(例如,负责基于待编码的输入图片和参考图片创建符号,例如符号流)和嵌入于视频编码器(503)中的(本地)解码器(533)。解码器(533)以类似于(远程)解码器创建样本数据的方式重建符号以创建样本数据(因为在本公开所考虑的视频压缩技术中,符号与已编码视频码流之间的任何压缩是无损的)。将重建的样本流(样本数据)输入到参考图片存储器(534)。由于符号流的解码产生与解码器位置(本地或远程)无关的位精确结果,因此参考图片存储器(534)中的内容在本地编码器与远程编码器之间也是按比特位精确对应的。换句话说,编码器的预测部分“看到”的参考图片样本与解码器将在解码期间使用预测时所“看到”的样本值完全相同。这种参考图片同步性基本原理(以及在例如因信道误差而无法维持同步性的情况下产生的漂移)也用于一些相关技术。

[0080] “本地”解码器(533)的操作可与例如已在上文结合图4详细描述视频解码器(410)的“远程”解码器相同。然而,另外简要参考图4,当符号可用且熵编码器(545)和解析器(420)能够无损地将符号编码/解码为已编码视频序列时,包括缓冲存储器(415)和解析器(420)在内的视频解码器(410)的熵解码部分,可能无法完全在本地解码器(533)中实施。

[0081] 此时可以观察到,除存在于解码器中的解析/熵解码之外的任何解码器技术,也必定以基本上相同的功能形式存在于对应的编码器中。出于此原因,本公开侧重于解码器操作。可简化编码器技术的描述,因为编码器技术与全面地描述的解码器技术互逆。仅在某些区域中需要更详细的描述,并且在下文提供。

[0082] 在操作期间,在一些实施例中,源编码器(530)可执行运动补偿预测编码。参考来自视频序列中被指定为“参考图片”的一个或多个先前已编码图片,所述运动补偿预测编码对输入图片进行预测性编码。以此方式,编码引擎(532)对输入图片的像素块与参考图片的像素块之间的差异进行编码,所述参考图片可被选作所述输入图片的预测参考。

[0083] 本地视频解码器(533)可基于源编码器(530)创建的符号,对可指定为参考图片的已编码视频数据进行解码。编码引擎(532)的操作可为有损过程。当已编码视频数据可在视频解码器(图5中未示)处被解码时,重建的视频序列通常可以是带有一些误差的源视频序列的副本。本地视频解码器(533)复制解码过程,所述解码过程可由视频解码器对参考图片执行,且可使重建的参考图片存储在参考图片高速缓存(534)中。以此方式,视频编码器(503)可在本地存储重建的参考图片的副本,所述副本与将由远端视频解码器获得的重建参考图片具有共同内容(不存在传输误差)。

[0084] 预测器(535)可针对编码引擎(532)执行预测搜索。即,对于将要编码的新图片,预测器(535)可在参考图片存储器(534)中搜索可作为所述新图片的适当预测参考的样本数据(作为候选参考像素块)或某些元数据,例如参考图片运动矢量、块形状等。预测器(535)可基于样本块逐像素块操作,以找到合适的预测参考。在一些情况下,根据预测器(535)获得的搜索结果,可确定输入图片可具有从参考图片存储器(534)中存储的多个参考图片取得的预测参考。

[0085] 控制器(550)可管理源编码器(530)的编码操作,包括例如设置用于对视频数据进行编码的参数和子群参数。

[0086] 可在熵编码器(545)中对所有上述功能单元的输出进行熵编码。熵编码器(545)根据例如霍夫曼编码、可变长度编码、算术编码等技术对各种功能单元生成的符号进行无损压缩,从而将所述符号转换成已编码视频序列。

[0087] 传输器(540)可缓冲由熵编码器(545)创建的已编码视频序列,从而为通过通信信道(560)进行传输做准备,所述通信信道可以是通向将存储已编码的视频数据的存储装置的硬件/软件链路。传输器(540)可将来自视频编码器(503)的已编码视频数据与要传输的其它数据合并,所述其它数据例如是已编码音频数据和/或辅助数据流(未示出来源)。

[0088] 控制器(550)可管理视频编码器(503)的操作。在编码期间,控制器(550)可以为每个已编码图片分配某一已编码图片类型,但这可能影响可应用于相应的图片的编码技术。例如,通常可将图片分配为以下任一种图片类型:

[0089] 帧内图片(I图片),其可以是不将序列中的任何其它图片用作预测源就可被编码和解码的图片。一些视频编解码器容许不同类型的帧内图片,包括例如独立解码器刷新(Independent Decoder Refresh,“IDR”)图片。所属领域的技术人员了解I图片的变体及其相应的应用和特征。

[0090] 预测性图片(P图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,所述帧内预测或帧间预测使用至多一个运动矢量和参考索引来预测每个块的样本值。

[0091] 双向预测性图片(B图片),其可以是可使用帧内预测或帧间预测进行编码和解码的图片,所述帧内预测或帧间预测使用至多两个运动矢量和参考索引来预测每个块的样本值。类似地,多个预测性图片可使用多于两个参考图片和相关联元数据以用于重建单个块。

[0092] 源图片通常可在空间上细分成多个样本块(例如,4×4、8×8、4×8或16×16个样

本的块),且逐块进行编码。这些块可参考其它(已编码)块进行预测编码,根据应用于块的相应图片的编码分配来确定所述其它块。举例来说,I图片的块可进行非预测编码,或所述块可参考同一图片的已经编码的块来进行预测编码(空间预测或帧内预测)。P图片的像素块可参考一个先前编码的参考图片通过空间预测或通过时域预测进行预测编码。B图片的块可参考一个或两个先前编码的参考图片通过空间预测或通过时域预测进行预测编码。

[0093] 视频编码器(503)可根据例如ITU-T H.265建议书的预定视频编码技术或标准执行编码操作。在操作中,视频编码器(503)可执行各种压缩操作,包括利用输入视频序列中的时间和空间冗余的预测编码操作。因此,已编码视频数据可符合所用视频编码技术或标准指定的语法。

[0094] 在实施例中,传输器(540)可在传输已编码的视频时传输附加数据。源编码器(530)可将此类数据作为已编码视频序列的一部分。附加数据可包括时间/空间/SNR增强层、冗余图片和切片等其它形式的冗余数据、SEI消息、VUI参数集片段等。

[0095] 采集到的视频可作为呈时间序列的多个源图片(视频图片)。帧内图片预测(常常简化为帧内预测)利用给定图片中的空间相关性,而帧间图片预测则利用图片之间的(时间或其它)相关性。在实施例中,将正在编码/解码的特定图片分割成块,正在编码/解码的特定图片被称作当前图片。在当前图片中的块类似于视频中先前已编码且仍被缓冲的参考图片中的参考块时,可通过称作运动矢量的矢量对当前图片中的块进行编码。所述运动矢量指向参考图片中的参考块,且在使用多个参考图片的情况下,所述运动矢量可具有识别参考图片的第三维度。

[0096] 在一些实施例中,双向预测技术可用于帧间图片预测中。根据双向预测技术,使用两个参考图片,例如按解码次序都在视频中的当前图片之前(但按显示次序可能分别是过去和将来)第一参考图片和第二参考图片。可通过指向第一参考图片中的第一参考块的第一运动矢量和指向第二参考图片中的第二参考块的第二运动矢量对当前图片中的块进行编码。具体来说,可通过第一参考块和第二参考块的组合来预测所述块。

[0097] 此外,合并模式技术可用于帧间图片预测中以改善编码效率。

[0098] 根据本公开的一些实施例,帧间图片预测和帧内图片预测等预测的执行以块为单位。举例来说,根据HEVC标准,将视频图片序列中的图片分割成编码树单元(coding tree unit,CTU)以用于压缩,图片中的CTU具有相同大小,例如 $64 \times 64$ 像素、 $32 \times 32$ 像素或 $16 \times 16$ 像素。一般来说,CTU包括三个编码树块(coding tree block,CTB),所述三个编码树块是一个亮度CTB和两个色度CTB。更进一步的,还可将每个CTU以四叉树拆分为一个或多个编码单元(coding unit,CU)。举例来说,可将 $64 \times 64$ 像素的CTU拆分为一个 $64 \times 64$ 像素的CU,或4个 $32 \times 32$ 像素的CU,或16个 $16 \times 16$ 像素的CU。在实施例中,分析每个CU以确定用于CU的预测类型,例如帧间预测类型或帧内预测类型。此外,取决于时间和/或空间可预测性,将CU拆分为一个或多个预测单元(prediction unit,PU)。通常,每个PU包括亮度预测块(prediction block,PB)和两个色度PB。在实施例中,编码(编码/解码)中的预测操作以预测块为单位来执行。以亮度预测块作为预测块为例,预测块包括像素值(例如,亮度值)的矩阵,例如 $8 \times 8$ 像素、 $16 \times 16$ 像素、 $8 \times 16$ 像素、 $16 \times 8$ 像素等等。

[0099] 图6是根据本公开另一实施例的视频编码器(603)的图。视频编码器(603)用于接收视频图片序列中的当前视频图片内的样本值的处理块(例如预测块),且将所述处理块编

码到作为已编码视频序列的一部分的已编码图片中。在本实施例中,视频编码器(603)用于代替图3实施例中的视频编码器(303)。

[0100] 在HEVC实施例中,视频编码器(603)接收用于处理块的样本值的矩阵,所述处理块为例如 $8 \times 8$ 样本的预测块等。视频编码器(603)使用例如率失真(rate-distortion, RD)优化来确定是否使用帧内模式、帧间模式或双向预测模式来编码所述处理块。当在帧内模式中编码处理块时,视频编码器(603)可使用帧内预测技术以将处理块编码到已编码图片中;且当在帧间模式或双向预测模式中编码处理块时,视频编码器(603)可分别使用帧间预测或双向预测技术将处理块编码到已编码图片中。在某些视频编码技术中,合并模式可以是帧间图片预测子模式,其中,在不借助预测值外部的已编码运动矢量分量的情况下,从一个或多个运动矢量预测值导出运动矢量。在某些其它视频编码技术中,可存在适用于主题块的运动矢量分量。在实施例中,视频编码器(603)包括其它组件,例如用于确定处理块模式的模式决策模块(未示出)。

[0101] 在图6的实施例中,视频编码器(603)包括如图6所示的耦接到一起的帧间编码器(630)、帧内编码器(622)、残差计算器(623)、开关(626)、残差编码器(624)、通用控制器(621)和熵编码器(625)。

[0102] 帧间编码器(630)用于接收当前块(例如处理块)的样本、比较所述块与参考图片中的一个或多个参考块(例如先前图片和后来图片中的块)、生成帧间预测信息(例如根据帧间编码技术的冗余信息描述、运动矢量、合并模式信息)、以及基于帧间预测信息使用任何合适的技术计算帧间预测结果(例如已预测块)。在一些实施例中,参考图片是基于已编码的视频信息解码的已解码参考图片。

[0103] 帧内编码器(622)用于接收当前块(例如处理块)的样本、在一些情况下比较所述块与同一图片中已编码的块、在变换之后生成量化系数、以及在一些情况下还(例如根据一个或多个帧内编码技术的帧内预测方向信息)生成帧内预测信息。在实施例中,帧内编码器(622)还基于帧内预测信息和同一图片中的参考块计算帧内预测结果(例如已预测块)。

[0104] 通用控制器(621)用于确定通用控制数据,且基于所述通用控制数据控制视频编码器(603)的其它组件。在实施例中,通用控制器(621)确定块的模式,且基于所述模式将控制信号提供到开关(626)。举例来说,当所述模式是帧内模式时,通用控制器(621)控制开关(626)以选择供残差计算器(623)使用的帧内模式结果,且控制熵编码器(625)以选择帧内预测信息且将所述帧内预测信息添加在码流中;以及当所述模式是帧间模式时,通用控制器(621)控制开关(626)以选择供残差计算器(623)使用的帧间预测结果,且控制熵编码器(625)以选择帧间预测信息且将所述帧间预测信息添加在码流中。

[0105] 残差计算器(623)用于计算所接收的块与选自帧内编码器(622)或帧间编码器(630)的预测结果之间的差(残差数据)。残差编码器(624)用于基于残差数据操作,以对残差数据进行编码以生成变换系数。在实施例中,残差编码器(624)用于将残差数据从时域转换到频域,且生成变换系数。变换系数接着经由量化处理以获得量化的变换系数。在各种实施例中,视频编码器(603)还包括残差解码器(628)。残差解码器(628)用于执行逆变换,且生成已解码残差数据。已解码残差数据可适当地由帧内编码器(622)和帧间编码器(630)使用。举例来说,帧间编码器(630)可基于已解码残差数据和帧间预测信息生成已解码块,且帧内编码器(622)可基于已解码残差数据和帧内预测信息生成已解码块。适当处理已解码



块以生成已解码图片,且在一些实施例中,所述已解码图片可在存储器电路(未示出)中缓冲并用作参考图片。

[0106] 熵编码器(625)用于将码流格式化以产生已编码的块。熵编码器(625)根据HEVC标准等合适标准产生各种信息。在实施例中,熵编码器(625)用于获得通用控制数据、所选预测信息(例如帧内预测信息或帧间预测信息)、残差信息和码流中的其它合适的信息。应注意,根据所公开的主题,当在帧间模式或双向预测模式的合并子模式中对块进行编码时,不存在残差信息。

[0107] 图7是根据本公开另一实施例的视频解码器(710)的图。视频解码器(710)用于接收作为已编码视频序列的一部分的已编码图像,且对所述已编码图像进行解码以生成重建的图片。在实施例中,视频解码器(710)用于代替图3实施例中的视频解码器(310)。

[0108] 在图7实施例中,视频解码器(710)包括如图7中所示耦接到一起的熵解码器(771)、帧间解码器(780)、残差解码器(773)、重建模块(774)和帧内解码器(772)。

[0109] 熵解码器(771)可用于根据已编码图片来重建某些符号,这些符号表示构成所述已编码图片的语法元素。此类符号可包括例如用于对所述块进行编码的模式(例如帧内模式、帧间模式、双向预测模式、后两者的合并子模式或另一子模式)、可分别识别供帧内解码器(772)或帧间解码器(780)用以进行预测的某些样本或元数据的预测信息(例如帧内预测信息或帧间预测信息)、呈例如量化的变换系数形式的残差信息等等。在实施例中,当预测模式是帧间或双向预测模式时,将帧间预测信息提供到帧间解码器(780);以及当预测类型是帧内预测类型时,将帧内预测信息提供到帧内解码器(772)。残差信息可经由逆量化并提供到残差解码器(773)。

[0110] 帧间解码器(780)用于接收帧间预测信息,且基于所述帧间预测信息生成帧间预测结果。

[0111] 帧内解码器(772)用于接收帧内预测信息,且基于所述帧内预测信息生成预测结果。

[0112] 残差解码器(773)用于执行逆量化以提取解量化的变换系数,且处理所述解量化的变换系数,以将残差从频域转换到空间域。残差解码器(773)还可能某些控制信息(用以获得量化器参数QP),且所述信息可由熵解码器(771)提供(未标示数据路径,因为这仅仅是低量控制信息)。

[0113] 重建模块(774)用于在空间域中组合由残差解码器(773)输出的残差与预测结果(可由帧间预测模块或帧内预测模块输出)以形成重建的块,所述重建的块可以是重建的图片的一部分,所述重建的图片继而可以是重建的视频的一部分。应注意,可执行解块操作等其它合适的操作来改善视觉质量。

[0114] 应注意,可使用任何合适的技术来实施视频编码器(303)、视频编码器(503)和视频编码器(603)以及视频解码器(310)、视频解码器(410)和视频解码器(710)。在实施例中,可使用一个或多个集成电路来实施视频编码器(303)、视频编码器(503)和视频编码器(603)以及视频解码器(310)、视频解码器(410)和视频解码器(710)。在另一实施例中,可使用执行软件指令的一个或多个处理器来实施视频编码器(303)、视频编码器(503)和视频编码器(603)以及视频解码器(310)、视频解码器(410)和视频解码器(710)。

[0115] 根据一些实施例,使用表示为编码树的四叉树二叉树(Quad Tree Binary Tree,



QTBT) 结构,将编码树单元(Coding Tree Unit,CTU)划分为多个编码单元(Coding Unit,CU),以适应多个CU中包括的各个块的各种局部特性。可以在CU级执行一个决定,即,是否使用帧间图片(时间)、或帧内图片(空间)预测对图片区域进行编码。根据PU分区类型,可以将每个CU进一步分割成一个、两个或四个预测单元(Prediction Unit,PU)。在一些实施例中,在一个PU内,应用相同的预测过程,并且基于PU将相关信息发送到解码器。基于PU分区类型,在应用预测过程获得残差块之后,根据与用于CTU的编码树的二叉树结构类似的另一二叉树结构,将CU划分为多个变换单元(Transform Unit,TU)。在一些其它实施例中,PU仅包含一个TU,其中,该TU与PU的形状相同。

[0116] CTU的编码树可以包括多个分区类型,包括CU、PU和TU。在一些实施例中,CU或TU仅为正方形形状,而PU可以为用于帧间预测块的正方形或矩形形状。在其它实施例中,允许矩形形状的CU、PU和TU。在图片边界处,可以应用隐式二叉树分割,使得块将保持二叉树分割,直到分割块的大小适合该图片边界。根据一些实施例,隐式分割意味着不发信号通知分割标志,而是暗示分割标志。例如,隐式QT意味着对于图片边界块仅允许QT分割。这样,在图像边界处不发信号通知分割标志。作为另一示例,当在图片边界处仅允许BT分割时,隐式分割是二进制分割。在一些实施例中,当在图片边界处允许QT和BT时,不存在隐式分割,并且显式方法是明确的发信号通知。

[0117] 根据一些实施例,QTBT结构不包括多个分区类型(例如,QTBT不包括CU、PU和TU的分离),并且支持CU分区形状的更多灵活性。例如,在QTBT块结构中,CU可以具有正方形或矩形形状。图8A示出了由QTBT结构划分的示例CTU(800)。例如,将CTU(800)划分为四个大小相等的子CU(A)、(B)、(C)和(D)。图8B示出了对应的编码树,该编码树示出了对应于子CU(A)、(B)、(C)和(D)的分支。实线指示二叉树分割,并且虚线指示二叉树分割。二叉树结构可以包括两种分割类型:(i)对称水平分割和(ii)对称垂直分割。在二叉树的每个分割(即,非叶)节点中,可以发信号通知一个标志,以指示使用了哪种分割类型(即,水平或垂直),其中,0指示水平分割,并且1指示垂直分割,反之亦然。对于二叉树分割,不指示分割类型,因为二叉树分割水平地和垂直地分割块,以产生相等大小的4个子块。

[0118] 如图8A和图8B所示,首先通过垂直分割,将子CU(A)划分为两个子块,其中,使用另一个垂直分割,再次划分左子块。使用水平分割,进一步划分子CU(B)。使用另一个四分割分区(quad split partition),进一步划分子CU(C)。使用垂直分割,划分子CU(C)的左上子块,然后,使用水平分割,划分子CU(C)的左上子块。此外,使用水平分割,划分子CU(C)的右下子块。不进一步划分子CU(C)的右上和左下子块。不进一步划分子CU(D),因此在“D”分支下面的编码树中不包括任何额外的叶节点。

[0119] 可以将二叉树叶节点称为CU,其中,二进制分割可以用于预测和变换处理,而无需任何进一步划分,这意味着在QTBT编码块结构中CU、PU和TU具有相同的块大小。CU可以包括不同颜色分量的编码块(Coding Block,CB)。例如,在4:2:0色度格式的P切片和B切片的情况下,一个CU可以包含一个亮度CB和两个色度CB,并且有时包含单个分量的CB(例如,在帧内图片或I切片的情况下,一个CU仅包含一个亮度CB、或仅包含两个色度CB)。在一些实施例中,在帧内图片或I切片中,将TU宽度或高度约束为不超过给定限制(例如,针对亮度为64,且针对色度为32)。如果CB宽度或高度大于限制,则进一步分割TU,直到TU的大小不超过限制。

[0120] 根据一些实施例,所述QTBT分区方案包括以下参数:

[0121] CTU大小:四叉树的根节点大小

[0122] MinQTSIZE:允许的最小四叉树叶节点大小

[0123] MaxBTSIZE:允许的最大二叉树根节点大小

[0124] MaxBTDepth:允许的最大二叉树深度

[0125] MinBTSIZE:允许的最小二叉树叶节点大小

[0126] 在QTBT分区结构的一个示例中,将CTU大小设定为 $128 \times 128$ 亮度样本,其中,该 $128 \times 128$ 亮度样本具有两个相应的 $64 \times 64$ 色度样本块。将MinQTSIZE设定为 $16 \times 16$ 。将MaxBTSIZE设定为 $64 \times 64$ 。将MinBTSIZE(针对宽度和高度)设定为 $4 \times 4$ 。并且,将MaxBTDepth设定为4。首先将QTBT分区结构应用于CTU,以产生四叉树叶节点。四叉树叶节点可以具有从 $16 \times 16$ (即,MinQTSIZE)到 $128 \times 128$ (即,CTU大小)的大小。如果叶四叉树节点是 $128 \times 128$ ,则不会通过二叉树进一步划分叶四叉树节点,因为该叶四叉树节点大小超过MaxBTSIZE(即, $64 \times 64$ )。否则,通过二叉树进一步划分叶四叉树节点。因此,四叉树叶节点也是二叉树的根节点,并且四叉树叶的二叉树深度为0。当二叉树深度达到MaxBTDepth(即,4)时,不执行进一步的分割。当二叉树节点的宽度等于MinBTSIZE(即,4)时,不执行进一步的水平分割。类似地,当二叉树节点的高度等于MinBTSIZE时,不执行进一步的垂直分割。通过预测和变换处理,进一步处理二叉树的叶节点,而无需任何进一步的划分。在一些实施例中,最大CTU大小是 $256 \times 256$ 的亮度样本。

[0127] QTBT分区结构可以进一步支持亮度和色度分量各自具有分离的QTBT结构的能力。例如,对于P切片和B切片,一个CTU中的亮度CTB和色度CTB可以共享相同的QTBT结构。然而,对于I切片,通过QTBT结构,将亮度CTB划分为多个CU。并且通过另一QTBT结构,将色度CTB划分为色度CU。因此,在该示例中,I切片中的CU包含亮度分量的编码块、或两个色度分量的编码块,并且P切片或B切片中的CU包含所有三个颜色分量的编码块。

[0128] 在一些实施例中,限制小块的帧间预测,以减少运动补偿的内存访问要求,使得 $4 \times 8$ 块和 $8 \times 4$ 块不支持双向预测,并且 $4 \times 4$ 块不支持帧间预测。在其它实施例中,QTBT分区方案不包括这些限制。

[0129] 根据一些实施例,多类型树(Multi-type-tree,MTT)结构包括(i)四叉树分割、(ii)二叉树分割,以及(iii)水平和垂直中心侧三叉树。图9A示出了垂直中心侧三叉树的实施例,并且图9B示出了水平中心侧三叉树的示例。与QTBT结构相比,因为MTT允许额外的结构,MTT可以是更灵活的树结构。

[0130] 三叉树分区包括显著有利的特征,例如,提供对四叉树和二叉树分区的补充,其中,三叉树分区能够捕获位于块中心的对象,而四叉树和二叉树沿着块中心分割。作为三叉树分区的另一优点,所提出的三叉树分区的宽度和高度是2的幂,从而不需要额外的变换。两级树提供了降低复杂度的优点。作为示例,遍历树的复杂度是 $T^D$ ,其中,T表示分割类型的数量,并且D是树的深度。

[0131] 存在不同的YUV格式,或色度格式,如图10A至图10D所示。每个色度格式可以定义不同颜色分量的不同下采样网格。

[0132] 可以用不同颜色格式(例如,YCbCr或RGB)表示视频样本的颜色。在RGB格式中,三个分量(即,R、G和B)具有强相关性,导致这三个颜色分量之间存在统计冗余。可以使用线性

变换,将视频样本的颜色表示转换成不同的颜色空间。将RGB颜色空间转换为YUV颜色空间可以如下执行:

[0133] 等式(1):  $Y = ((66 * R + 129 * G + 25 * B + 128) \gg 8) + 16$

[0134] 等式(2):  $U = ((-38 * R - 74 * G + 112 * B + 128) \gg 8) + 128$

[0135] 等式(3):  $V = ((112 * R - 94 * G - 18 * B + 128) \gg 8) + 128$

[0136] 将RGB颜色空间转换为YUV颜色空间可以如下执行:

[0137] 等式(4):  $Y = \text{round}(0.256788 * R + 0.504129 * G + 0.097906 * B) + 16$

[0138] 等式(5):  $U = \text{round}(-0.148223 * R - 0.290993 * G + 0.439216 * B) + 128$

[0139] 等式(6):  $V = \text{round}(0.439216 * R - 0.367788 * G - 0.071427 * B) + 128$

[0140] 为了对RGB视频内容进行有效编码,开发了环路内自适应颜色变换(Adaptive Colour Transform, ACT),其中,ACT在剩余域中操作。可以发信号通知CU级标志,以指示使用ACT。可以如下执行正向ACT颜色变换(例如,在编码器处执行的变换):

[0141] 等式(7):

$$[0142] \quad \begin{bmatrix} Y \\ C_g \\ C_o \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ -1 & 2 & -1 \\ 2 & 0 & -2 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

[0143] 可以如下执行反向ACT颜色变换(例如,在解码器处执行的逆变换):

[0144] 等式(8):

$$[0145] \quad \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & -1 \end{bmatrix} \times \begin{bmatrix} Y \\ C_g \\ C_o \end{bmatrix}$$

[0146] 图11示出了用于执行颜色空间变换的示例编码器1100。在图11中,在执行颜色空间变换之前,执行预测。例如,对当前块执行帧间预测或帧内预测,以产生残差信号。将残差信号提供给正向颜色空间变换单元(1102),以执行正向变换,例如等式(7)中的变换。将正向颜色空间变换的输出提供给交叉分量产生(Cross Component Production, CCP)单元(1102)。将CCP单元(1102)的输出提供给变换(Transform, T)单元(1106),以执行例如离散余弦变换(Discrete Cosine Transform, DCT)类型的变换。将变换单元(1106)的输出提供给量化器(quantizer, Q) (1108)以产生系数。将系数提供给熵编码器单元(1110),以提供比特流。熵编码器单元(1110)可以接收模式/mv信号,以选择熵编码器的特定操作模式。

[0147] 编码器(1100)还可以包括将比特流转换为残差信号的分量。例如,可以将熵编码器(1110)产生的比特流提供给逆量化器(Inverse Quantizer, IQ)单元(1112)。可以将逆量化器单元(IQ)的输出提供给逆变换(Inverse Transform, IT)单元(1114)。可以将逆变换(IT)单元(1114)的输出提供给逆CCP单元(1116)。可以将逆CCP单元(1116)的输出提供给逆颜色空间变换(1118),其中,可以执行逆颜色变换(例如,等式(8)中示出的变换)。

[0148] 图12示出了用于将比特流转换为残差信号的示例解码器(1200)。图12所示的比特流可以是由熵编码器(1110)(图11)产生的比特流。可以将比特流提供给熵解码器单元(1202)。可以将熵解码器单元(1202)的输出提供给逆量化器(IQ)单元(1204)。可以将逆量化器单元(IQ)的输出提供给逆变换(inverse transform, IT)单元(1206)。可以将逆变换(IT)单元(1206)的输出提供给逆CCP单元(1208)。可以将逆CCP单元(1208)的输出提供给逆

颜色空间变换(1210),其中,可以执行逆颜色变换(例如,等式(8)中示出的变换),以产生残差信号。可以对残差信号执行帧内预测或帧间预测,以对当前块进行解码。图11和图12中公开的单元可以通过处理器、或设计为执行每个单元功能的电路(例如,专用集成电路),以软件形式实现

[0149] 对于帧内PU的色度分量,编码器可以在包括平面、DC、水平、垂直、来自亮度分量的帧内预测模式(DM)的直接副本、左和上跨分量线性模式(Left and Top Cross-component Linear Mode,LT\_CCLM)、左跨分量线性模式(Left Cross-component Linear Mode,L\_CCLM)和上跨分量线性模式(Top Cross-component Linear Mode,T\_CCLM)的8种模式中选择最佳色度预测模式。可以将LT\_CCLM、L\_CCLM和T\_CCLM归类成跨分量线性模式(Cross-component Linear Mode,CCLM)的组。这3种模式之间的差别在于相邻样本的不同区域可以用于导出参数 $\alpha$ 和 $\beta$ 。对于LT\_CCLM,左相邻样本和上相邻样本都可以用于导出参数 $\alpha$ 和 $\beta$ 。对于L\_CCLM,在一些示例中,仅使用左相邻样本导出参数 $\alpha$ 和 $\beta$ 。对于T\_CCLM,在一些示例中,仅使用上相邻样本导出参数 $\alpha$ 和 $\beta$ 。

[0150] 跨分量线性模型(CCLM)预测模式可用于减少跨分量冗余,其中,使用如下示例的线性模型,基于相同CU的已重建亮度样本,预测色度样本:

[0151] 等式(9):  $\text{pred}_c(i, j) = \alpha \cdot \text{rec}_L'(i, j) + \beta$ ,

[0152] 其中,  $\text{pred}_c(i, j)$  表示CU中的预测色度样本,并且  $\text{rec}_L(i, j)$  表示相同CU的下采样已重建亮度样本。可以由直线方程导出参数 $\alpha$ 和 $\beta$ ,也可以将该直线方程称为最大-最小(max-min)方法。由于可以作为解码过程的一部分,执行该计算过程,而不仅仅作为编码器搜索操作,因此不使用语法来传达 $\alpha$ 和 $\beta$ 值。

[0153] 对于色度4:2:0格式,CCLM预测可以应用六抽头插值滤波器,以得到对应于图13所示的色度样本的下采样亮度样本。基于公式,从已重建亮度样本计算下采样亮度样本  $\text{Rec}'_L[x, y]$ 。

[0154] 下采样亮度样本可以用于找到最大和最小样本点。2个点(亮度和色度对)(A,B)可以是如图13所描绘的相邻亮度样本集内的最小值和最大值。

[0155] 根据以下等式,可以获得线性模型参数 $\alpha$ 和 $\beta$ :

[0156] 等式(10):  $\alpha = \frac{y_B - y_A}{x_B - x_A}$

[0157] 等式(11):  $\beta = y_A - \alpha x_A$

[0158] 使用乘法和移位运算有利地避免了除法运算。可以使用一个查找表(Look-up Table,LUT)存储预先计算的值,并且,可以使用最大亮度样本和最小亮度样本之间的绝对差值,指定LUT的条目索引。LUT的大小可以是512。

[0159] 图14A和图14B示出了用于导出LT\_CCLM中 $\alpha$ 和 $\beta$ 的样本的示例位置。在T\_CCLM模式中,在一些示例中,仅使用上相邻样本(包括2\*W个样本)计算线性模型系数。图15A和图15B示出了用于导出T\_CCLM中的 $\alpha$ 和 $\beta$ 的样本的示例位置。

[0160] 在L\_CCLM模式中,在一些示例中,仅使用左相邻样本(包括2\*H个样本)计算线性模型系数。图16A和图16B示出了用于导出L\_CCLM中 $\alpha$ 和 $\beta$ 的样本的示例位置。

[0161] CCLM预测模式还可以包括两个色度分量之间(即,从Cb分量预测Cr分量)的预测。可以在残差域中应用CCLM Cb到Cr预测,而不是使用已重建样本信号。可以将加权的已重建

Cb残差添加到原始Cr帧内预测,形成最终Cr预测,从而实现CCLM Cb到Cr预测

[0162] 等式 (12) :  $pred_{Cr}^*(i,j) = pred_{Cr}(i,j) + \alpha \cdot resi_{Cb}'(i,j)$

[0163] 可以添加CCLM亮度到色度预测模式作为一个额外的色度帧内预测模式。在编码器侧,对色度分量再添加一个率失真(Rate Distortion, RD)成本检查,以选择色度帧内预测模式。当除了CCLM亮度到色度预测模式之外的帧内预测模式用于CU的色度分量时,CCLM Cb到Cr预测用于预测Cr分量。

[0164] 多模型CCLM(Multiple Model CCLM, MMLM)是CCLM的另一扩展,其中可以有一个以上的模型(例如,两个或更多个模型)。在MMLM中,可以将当前块的相邻亮度样本和相邻色度样本划分为两个组,其中,每个组可以用作训练集,以导出线性模型(即,针对特定组导出特定 $\alpha$ 和 $\beta$ )。此外,基于与相邻亮度样本的相同分类规则,对当前亮度块的样本进行分类。

[0165] 图17示出了将相邻样本分为两个组的示例。可以将图17示出的阈值计算为相邻已重建亮度样本的平均值。将 $Rec' L[x,y] \leq \text{阈值}$ 的相邻样本分类为组1;而将 $Rec' L[x,y] > \text{阈值}$ 的相邻样本分类为组2。

[0166] 等式 (13) : 
$$\begin{cases} Pred_c[x,y] = \alpha_1 \times Rec'_L[x,y] + \beta_1 & \text{if } Rec'_L[x,y] \leq \text{Threshold} \\ Pred_c[x,y] = \alpha_2 \times Rec'_L[x,y] + \beta_2 & \text{if } Rec'_L[x,y] > \text{Threshold} \end{cases}$$

[0167] 为了启用VVC中的环路内颜色变换,以有效编码RGB格式的输入视频,需要处理VVC中颜色变换与若干编解码工具之间的交互,例如,跨分量线性模型和双树分区。本公开实施例提供了利用VVC中的编解码工具来处理颜色变换的显著有利的特征。

[0168] 本公开实施例可以单独使用或以任何顺序组合使用。此外,根据本公开实施例,每个方法、编码器和解码器可以由处理电路(例如,一个或多个处理器、或一个或多个集成电路)实现。在一个示例中,一个或多个处理器执行存储在非易失性计算机可读介质中的程序。根据本公开的实施例,可以将术语块解释为预测块、编码块或编码单元(即,CU)。根据本公开的实施例,术语亮度分量可以指按编码顺序编码为第一分量的任何颜色分量(例如,红或绿颜色分量)。此外,根据本公开的实施例,术语色度分量可以指按编码顺序未编码为第一分量的任何颜色分量。

[0169] 根据一些实施例,在编码器执行预测过程之前,以及在解码器执行重建过程之后,应用例如ACT的颜色变换。在编码器处,可以在预测(例如,帧间预测、帧内预测)之前执行ACT,并且,如果将ACT应用于当前CU,那么可以将参考样本和输入原始样本映射到不同颜色空间。对于解码器处的像素重建,如果将ACT应用于重建中的块,则可以在将参考样本用于预测之前,将参考样本映射到可选的颜色空间,然后,可以将已重建样本映射回到原始颜色空间。

[0170] 图18示出了使用ACT的编码器和解码器处理的实施例。图18中公开的单元可以通过处理器、或电路,例如设计成执行图18中公开的每个单元的功能的专用集成电路,以软件实现。在编码器处,ACT单元(1800)和(1804)分别对参考信号和输入信号执行ACT变换。由ACT单元(1800)和(1804)在编码器处执行的ACT变换可以是等式(7)中公开的ACT变换。将ACT(1800)的输出提供给预测(P)单元(1802)。此外,将参考信号提供给预测(P)单元(1806)。预测(P)单元(1802)和(1806)可以执行帧间预测或帧内预测。变换(T)单元1808接收(i)和(ii)中的一个,其中,(i)预测(P)单元(1802)的输出与ACT单元(1804)的输出之间

的差, (ii) 预测 (P) 单元 (1806) 的输出与输入信号之间的差。变换 (T) 单元 (1808) 可以执行变换操作, 例如, 离散余弦变换 (Discrete Cosine Transform, DCT)。将变换 (T) 单元 1808 的输出提供给量化器单元 (quantizer, Q) (1810), 执行量化操作, 产生系数集。

[0171] 在解码器处, 逆量化器 (Inverse Quantizer, IQ) 单元 (1812) 接收系数, 以执行逆量化过程。将逆量化器 (IQ) 单元 (1812) 的输出提供给逆变换 (Inverse Transform, IT) 单元 (1814), 以执行逆变换。ACT 单元 (1820) 接收预测 (P) 单元 (1818) 的输出和逆变换 (IT) 单元 (1814) 的输出之和。ACT 单元 (1816) 接收预测 (P) 单元 (1818) 的输出。ACT 单元 (1816) 和 (1820) 可以执行逆颜色变换, 例如等式 (8) 中公开的逆颜色变换。预测 (P) 单元 (1818) 和 (1822) 可以执行帧间预测或帧内预测。ACT 单元 (1816) 的输出提供已重建参考信号, 并且 ACT 单元 (1820) 的输出提供已重建原始信号。

[0172] 根据一些实施例, 在 ACT 过程中, 在颜色变换之后, 为了正向变换, 将第二颜色分量进一步偏移常数  $c$ 。在颜色变换之前, 为了反向变换, 将第三颜色分量进一步偏移常数  $c$ 。等式 (14) 示出了修改的正向变换, 并且等式 (15) 示出了修改的反向 (即, 逆) 变换。

[0173] 等式 (14): 
$$\begin{bmatrix} Y \\ C_g \\ C_o \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ -1 & 2 & -1 \\ 2 & 0 & -2 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ c \\ c \end{bmatrix}$$

[0174] 等式 (15): 
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & -1 \end{bmatrix} \times \begin{bmatrix} Y \\ C_g - c \\ C_o - c \end{bmatrix}$$

[0175] 在一些实施例中, 将常数  $c$  导出为  $1 \ll (\text{bitDepth} - 1)$ , 其中,  $\text{bitDepth}$  是指输入样本的位深度。

[0176] 根据一些实施例, 仅在使用相同的变换单元分割树对不同颜色分量进行编码时, 应用颜色变换。在一个实施例中, 当在帧内切片上应用双树 (DualTree) 时, 仅对帧间切片应用颜色变换。

[0177] 根据一些实施例, 当应用颜色变换时, 不应用或发信号通知 CCLM 模式, 因为从一个分量生成残差样本取决于另一分量的重建。在另一实施例中, 当使用 CCLM 模式时, 不应用或发信号通知颜色变换。在一个实施例中, 当对帧内残差样本应用颜色变换时, 不应用或发信号通知 CCLM 模式, 因为从一个分量生成残差样本取决于另一分量的重建。在一个实施例中, 当在残差样本上应用颜色变换且使用 CCLM 模式时, 不应用或发信号通知颜色变换。

[0178] 根据一些实施例, 针对作为最大编码单元 (CU) 的每个 CTU, 发信号通知颜色变换。

[0179] 在一些实施例中, 仅对帧内编码块、或仅对帧间编码块, 发信号通知和应用颜色变换。在一些实施例中, 当应用颜色变换时, 不应用双树 (DualTree) (即, 不同的颜色分量共享相同的变换单元分区)。

[0180] 图 19 示出了由例如编码器 (603) 的编码器执行的过程的实施例。该过程可以开始于步骤 (S1900), 其中, 确定是否满足颜色变换条件。例如, 颜色变换条件可以为一个标志, 该标志指示是否针对当前块启用颜色变换。作为另一示例, 颜色变换条件可以为一个标志, 该标志指示针对 CTU 中的每一块启用颜色变换。如果满足颜色变换条件, 则过程前进到步骤 (S1902), 其中, 对当前块执行颜色变换。例如, 颜色变换可以是等式 (7) 所示的 ACT 操作。

[0181] 过程从步骤 (S1902) 前进到步骤 (S1904), 以对颜色变换的当前块执行预测。该预

测可以是帧间预测或帧内预测。如果在步骤 (S1900) 不满足颜色变换条件,则过程从步骤 (S1900) 前进到步骤 (S1904)。过程从步骤 (S1904) 前进到步骤 (S1906),以对预测的当前块执行例如DCT的变换过程。过程前进到步骤 (S1908),以对变换的当前块执行量化过程。量化过程的输出可以是系数集,该系数集包含在发送到解码器的比特流中。图19所示的过程可以在执行步骤 (S1908) 之后终止。

[0182] 图20示出了由例如解码器 (710) 的解码器执行的过程的实施例。该过程可以在步骤 (S2000) 处开始,其中,接收已编码视频比特流。该比特流可以包括步骤 (S1908) (图19) 产生的系数。过程前进到步骤 (S2002),其中,对与当前块相对应的系数集执行逆量化。过程前进到步骤 (S2004),其中,对当前块的逆量化的输出执行逆变换。过程前进到步骤 (S2006),其中,对与当前块相对应的逆变换的输出执行预测,比如,帧间预测或帧内预测。在步骤 (S2008),确定当前块是否满足颜色变换条件。例如,颜色变换条件可以为一个标志,该标志指示是否针对当前块启用颜色变换。作为另一示例,颜色变换条件可以为一个标志,该标志指示针对CTU中的每一块启用颜色变换。如果满足颜色变换条件,则过程进行到步骤 (S2010),对当前块执行逆颜色变换。例如,逆颜色变换可以是等式 (8) 所示的ACT操作。如果不满足颜色变换条件,则完成图20所示的过程。图20所示的过程也可以在执行步骤 (S2010) 之后完成。

[0183] 上述技术可以通过计算机可读指令实现为计算机软件,并且物理地存储在一个或多个计算机可读介质中。例如,图21示出了计算机系统 (2100),其适于实现所公开主题的某些实施例。

[0184] 所述计算机软件可通过任何合适的机器代码或计算机语言进行编码,通过汇编、编译、链接等机制创建包括指令的代码,所述指令可由一个或多个计算机中央处理单元 (CPU),图形处理单元 (GPU) 等直接执行或通过译码、微代码等方式执行。

[0185] 所述指令可以在各种类型的计算机或其组件上执行,包括例如个人计算机、平板电脑、服务器、智能手机、游戏设备、物联网设备等。

[0186] 图21所示的用于计算机系统 (2100) 的组件本质上是示例性的,并不用于对实现本公开实施例的计算机软件的使用范围或功能进行任何限制。也不应将组件的配置解释为与计算机系统 (2100) 的示例性实施例中所示的任一组件或其组合具有任何依赖性要求。

[0187] 计算机系统 (2100) 可以包括某些人机界面输入设备。这种人机界面输入设备可以通过触觉输入 (如:键盘输入、滑动、数据手套移动)、音频输入 (如:声音、掌声)、视觉输入 (如:手势)、嗅觉输入 (未示出),对一个或多个人类用户的输入做出响应。所述人机界面设备还可用于捕获某些媒体,气与人类有意识的输入不必直接相关,如音频 (例如:语音、音乐、环境声音)、图像 (例如:扫描图像、从静止影像相机获得的摄影图像)、视频 (例如二维视频、包括立体视频的三维视频)。

[0188] 人机界面输入设备可包括以下中的一个或多个 (仅绘出其中一个):键盘 (2101)、鼠标 (2102)、触控板 (2103)、触摸屏 (2110)、数据手套 (未示出)、操纵杆 (2105)、麦克风 (2106)、扫描仪 (2107)、照相机 (2108)。

[0189] 计算机系统 (2100) 还可以包括某些人机界面输出设备。这种人机界面输出设备可以通过例如触觉输出、声音、光和嗅觉/味觉来刺激一个或多个人类用户的感受。这样的人机界面输出设备可包括触觉输出设备 (例如通过触摸屏 (2110)、数据手套 (未示出) 或操纵

杆(2105)的触觉反馈,但也可以有不用作输入设备的触觉反馈设备)、音频输出设备(例如,扬声器(2109)、耳机(未示出))、视觉输出设备(例如,包括阴极射线管屏幕、液晶屏幕、等离子屏幕、有机发光二极管屏的屏幕(2110),其中每一个都具有或没有触摸屏输入功能、每一个都具有或没有触觉反馈功能——其中一些可通过诸如立体画面输出的手段输出二维视觉输出或三维以上的输出;虚拟现实眼镜(未示出)、全息显示器和放烟箱(未示出))以及打印机(未示出)。

[0190] 计算机系统(2100)还可以包括人可访问的存储设备及其相关介质,如包括具有CD/DVD的高密度只读/可重写式光盘(CD/DVD ROM/RW)(2120)或类似介质(2121)的光学介质、拇指驱动器(2122)、可移动硬盘驱动器或固体状态驱动器(2123),诸如磁带和软盘(未示出)的传统磁介质,诸如安全软件保护器(未示出)等的基于ROM/ASIC/PLD的专用设备,等等。

[0191] 本领域技术人员还应当理解,结合所公开的主题使用的术语“计算机可读介质”不包括传输介质、载波或其它瞬时信号。

[0192] 计算机系统(2100)还可以包括通往一个或多个通信网络的接口(1654)。该网络例如可以是无线的、有线的、光学的。该网络还可为局域网、广域网、城域网、车载网络和工业网络、实时网络、延迟容忍网络等等。该网络的示例还包括以太网、无线局域网、蜂窝网络(GSM、3G、4G、5G、LTE等)等局域网、电视有线或无线广域数字网络(包括有线电视、卫星电视、和地面广播电视)、车载和工业网络(包括CANBus)等等。某些网络通常需要外部网络接口适配器,用于连接到某些通用数据端口或外围总线(2149)(例如,计算机系统(2100)的USB端口);其它系统通常通过连接到如下所述的系统总线集成到计算机系统(2100)的核心(例如,以太网接口集成到PC计算机系统或蜂窝网络接口集成到智能电话计算机系统)。通过使用这些网络中的任何一个,计算机系统(2100)可以与其它实体进行通信。所述通信可以是单向的,仅用于接收(例如,无线电视),单向的仅用于发送(例如CAN总线到某些CAN总线设备),或双向的,例如通过局域或广域数字网络到其它计算机系统。上述的每个网络和网络接口可使用某些协议和协议栈。

[0193] 上述的人机界面设备、人可访问的存储设备以及网络接口可以连接到计算机系统(2100)的核心(2140)。

[0194] 核心(2140)可包括一个或多个中央处理单元(CPU)(2141)、图形处理单元(GPU)(2142)、以现场可编程门阵列(FPGA)(2143)形式的专用可编程处理单元、用于特定任务的硬件加速器(2144)等。这些设备以及只读存储器(ROM)(2145)、随机存取存储器(2146)、内部大容量存储器(例如内部非用户可存取硬盘驱动器、固态硬盘等)(2147)等可通过系统总线(2148)进行连接。在某些计算机系统中,可以以一个或多个物理插头的形式访问系统总线(2148),以便可通过额外的中央处理单元、图形处理单元等进行扩展。外围装置可直接附接到核心的系统总线(2148),或通过外围总线(2149)进行连接。外围总线的体系结构包括外部控制器接口PCI、通用串行总线USB等。

[0195] CPU(2141)、GPU(2142)、FPGA(2143)和加速器(2144)可以执行某些指令,这些指令组合起来可以构成上述计算机代码。该计算机代码可以存储在ROM(2145)或RAM(2146)中。过渡数据也可以存储在RAM(2146)中,而永久数据可以存储在例如内部大容量存储器(2147)中。通过使用高速缓冲存储器可实现对任何存储器设备的快速存储和检索,高速缓



冲存储器可与一个或多个CPU (2141)、GPU (2142)、大容量存储器 (2147)、ROM (2145)、RAM (2146)等紧密关联。

[0196] 所述计算机可读介质上可具有计算机代码,用于执行各种计算机实现的操作。介质和计算机代码可以是为本公开的目的而特别设计和构造的,也可以是计算机软件领域的技术人员所熟知和可用的介质和代码。

[0197] 作为实施例而非限制,具有体系结构 (2100)的计算机系统,特别是核心 (2140),可以作为处理器 (包括CPU、GPU、FPGA、加速器等)提供执行包含在一个或多个有形的计算机可读介质中的软件的功能。这种计算机可读介质可以是与上述的用户可访问的大容量存储器相关联的介质,以及具有非易失性的核心 (2140)的特定存储器,例如核心内部大容量存储器 (2147)或ROM (2145)。实现本公开的各种实施例的软件可以存储在这种设备中并且由核心 (2140)执行。根据特定需要,计算机可读介质可包括一个或一个以上存储设备或芯片。该软件可以使得核心 (2140)特别是其中的处理器 (包括CPU、GPU、FPGA等)执行本文所述的特定过程或特定过程的特定部分,包括定义存储在RAM (2146)中的数据结构以及根据软件定义的过程来修改这种数据结构。另外或作为替代,计算机系统可以提供逻辑硬连线或以其它方式包含在电路 (例如,加速器 (2144))中的功能,该电路可以代替软件或与软件一起运行以执行本文所述的特定过程或特定过程的特定部分。在适当的情况下,对软件的引用可以包括逻辑,反之亦然。在适当的情况下,对计算机可读介质的引用可包括存储执行软件的电路 (如集成电路 (IC)),包含执行逻辑的电路,或两者兼备。本公开包括任何合适的硬件和软件组合。

[0198] 附录A:缩略词

[0199] JEM:联合开发模型

[0200] VVC:多功能视频编解码

[0201] BMS:基准集

[0202] MV:运动矢量

[0203] HEVC:高效视频编码

[0204] SEI:补充增强信息

[0205] VUI:视频可用性信息

[0206] GOPs:图片组

[0207] TUs:变换单元

[0208] PUs:预测单元

[0209] CTUs:编码树单元

[0210] CTBs:编码树块

[0211] PBs:预测块

[0212] HRD:假定参考解码器

[0213] SNR:信号噪声比

[0214] CPUs:中央处理单元

[0215] GPUs:图片处理单元

[0216] CRT:阴极射线管

[0217] LCD:液晶显示器

- [0218] OLED:有机发光二极管
- [0219] CD:光盘
- [0220] DVD:数码视盘
- [0221] ROM:只读存储器
- [0222] RAM:随机存取存储器
- [0223] ASIC:专用集成电路
- [0224] PLD:可编程逻辑器件
- [0225] LAN:局域网
- [0226] GSM:全球移动通信系统
- [0227] LTE:长期演进
- [0228] CANBus:控制器局域网总线
- [0229] USB:通用串行总线
- [0230] PCI:外围组件互联
- [0231] FPGA:现场可编程门区域
- [0232] SSD:固态驱动器
- [0233] IC:集成电路
- [0234] CU:编码单元

[0235] 虽然本公开已对多个示例性实施例进行了描述,但实施例的各种变更、排列和各种等同替换均属于本公开的范围之内。因此应理解,本领域技术人员能够设计多种系统和方法,所述系统和方法虽然未在本文中明确示出或描述,但其体现了本公开的原则,因此属于本公开的精神和范围之内。

[0236] (1)、一种在视频解码器中执行的视频解码方法,所述方法包括:接收包括当前图片的已编码视频比特流;对所述当前图片中的当前块执行逆量化;在执行所述逆量化之后,对所述当前块执行逆变换;在执行所述逆变换之后,对所述当前块执行预测过程;对所述当前块执行预测过程之后,确定是否满足预定条件;以及,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

[0237] (2) 根据特征(1)所述的方法,其中,所述逆颜色变换是逆自适应颜色变换(Adaptive Color Transform,ACT),并且执行所述逆颜色变换将重建的当前块从颜色空间变换转换为RGB格式。

[0238] (3) 根据特征(1)或(2)所述的方法,其中,执行所述逆颜色变换包括将常数减去所述逆颜色变换的一个或多个颜色分量。

[0239] (4) 根据特征(3)所述的方法,其中,所述常数是对输入样本的比特深度减1,再执行左移操作导出的。

[0240] (5) 根据特征(1)-(4)中任意一个特征所述的方法,其中,当确定发信号通知当前块进行颜色变换时,满足所述预定条件。

[0241] (6) 根据特征(5)所述的方法,其中,为具有最大编码单元(Coding Unit,CU)的每个编码树单元(Coding Tree Unit,CTU),发信号通知所述颜色变换。

[0242] (7) 根据特征(1)-(6)中任意一个特征所述的方法,其中,当确定使用相同的变换单元分割树,对所述逆颜色变换的不同颜色分量进行编码,满足所述预定条件。

[0243] (8) 根据特征 (7) 所述的方法,其中,当确定将双树(DualTree)应用于帧内切片时,将所述逆颜色变换应用于帧间切片。

[0244] (9) 根据特征 (1) - (8) 中任意一个特征所述的方法,进一步包括:当确定满足所述预定条件时,不将跨分量线性模式(Cross-component Linear Mode,CCLM)应用于所述当前块的色度单元。

[0245] (10) 根据特征 (1) - (9) 中任意一个特征所述的方法,其中,所述预测过程是帧间预测或帧内预测之一。

[0246] (11) 一种用于视频解码的视频解码器,包括处理电路,其中,所述处理电路用于:接收包括当前图片的已编码视频比特流;对所述当前图片中的当前块执行逆量化;在执行所述逆量化之后,对所述当前块执行逆变换;在执行所述逆变换之后,对所述当前块执行预测过程;对所述当前块执行预测过程之后,确定是否满足预定条件;以及,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

[0247] (12) 根据特征 (11) 所述的视频解码器,其中,所述逆颜色变换是逆自适应颜色变换(Adaptive Color Transform,ACT),并且执行所述逆颜色变换将重建的当前块从颜色空间变换转换为RGB格式。

[0248] (13) 根据特征 (11) 或 (12) 所述的视频解码器,其中,执行所述逆颜色变换包括所述处理电路将常数减去所述逆颜色变换的一个或多个颜色分量。

[0249] (14) 根据特征 (13) 所述的视频解码器,其中,所述常数是对输入样本的比特深度减1,再执行左移操作导出的。

[0250] (15) 根据特征 (11) - (14) 中任意一个特征所述的视频解码器,其中,当确定发信号通知当前块进行颜色变换时,满足所述预定条件。

[0251] (16) 根据特征 (15) 所述的视频解码器,其中,为具有最大编码单元(Coding Unit, CU)的每个编码树单元(Coding Tree Unit,CTU),发信号通知所述颜色变换。

[0252] (17) 根据特征 (11) - (16) 中任意一个特征所述的视频解码器,其中,当确定使用相同的变换单元分割树,对所述逆颜色变换的不同颜色分量进行编码,满足所述预定条件。

[0253] (18) 根据特征 (17) 所述的视频解码器,其中,将双树(DualTree)应用于帧内切片时,将所述逆颜色变换应用于帧间切片。

[0254] (19) 根据特征 (11) 所述的视频解码器,其中,当确定满足所述预定条件时,不将跨分量线性模式(Cross-component Linear Mode,CCLM)应用于所述当前块的色度单元。

[0255] (20) 一种非易失性计算机可读介质,所述非易失性计算机可读介质存储指令,当所述指令被视频解码器中的处理器执行时,使得所述视频解码器执行一个方法,其中,该方法包括:在视频解码器中执行的视频解码方法,其特征在于,所述方法包括:接收包括当前图片的已编码视频比特流;对所述当前图片中的当前块执行逆量化;在执行所述逆量化之后,对所述当前块执行逆变换;在执行所述逆变换之后,对所述当前块执行预测过程;对所述当前块执行预测过程之后,确定是否满足预定条件;以及,当确定满足所述预定条件时,对所述当前块执行逆颜色变换。

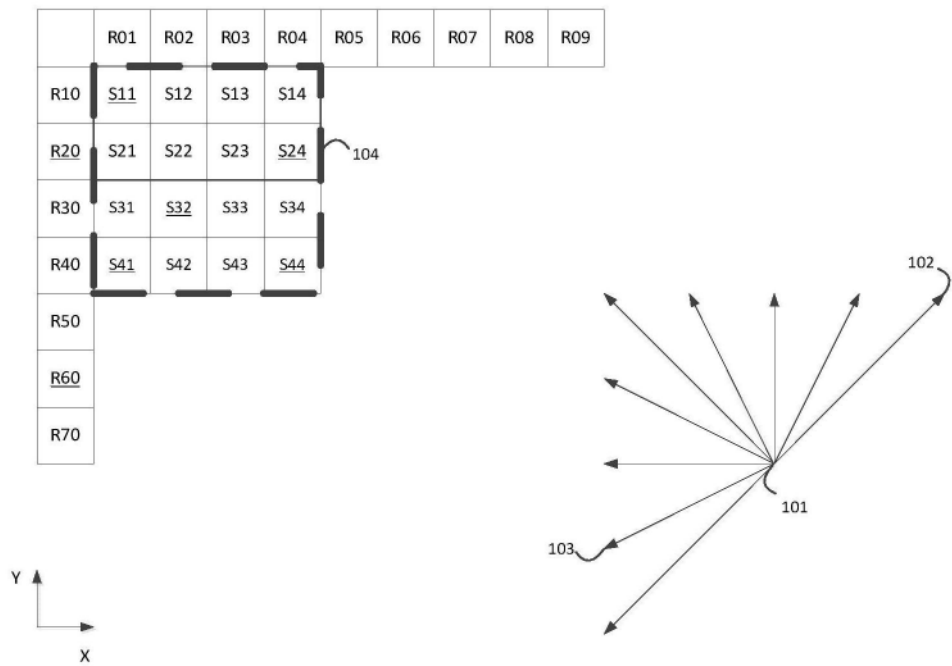


图1A

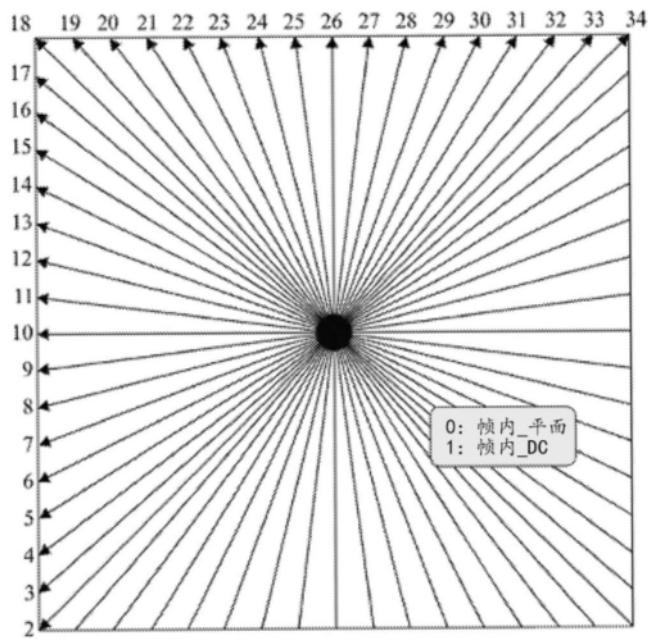


图1B

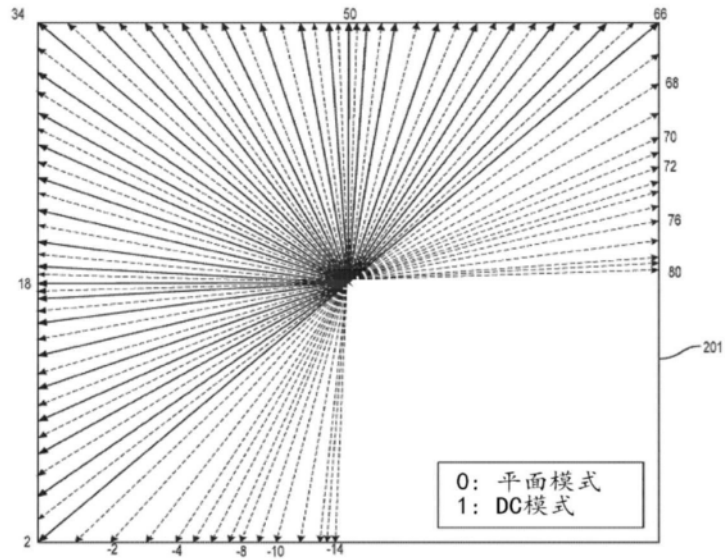


图1C

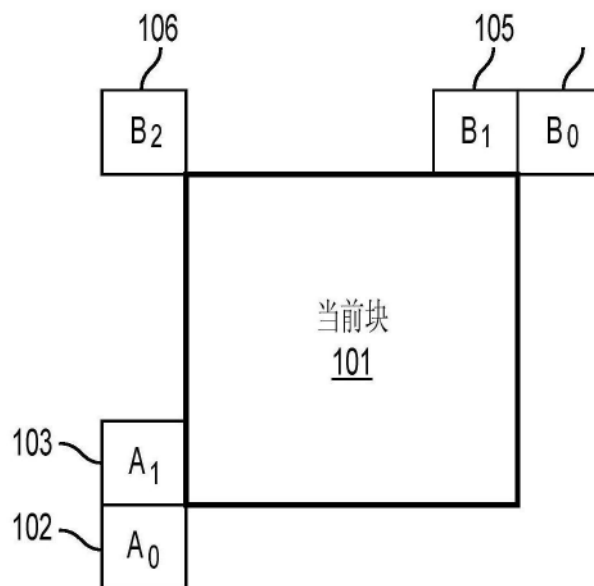


图1D

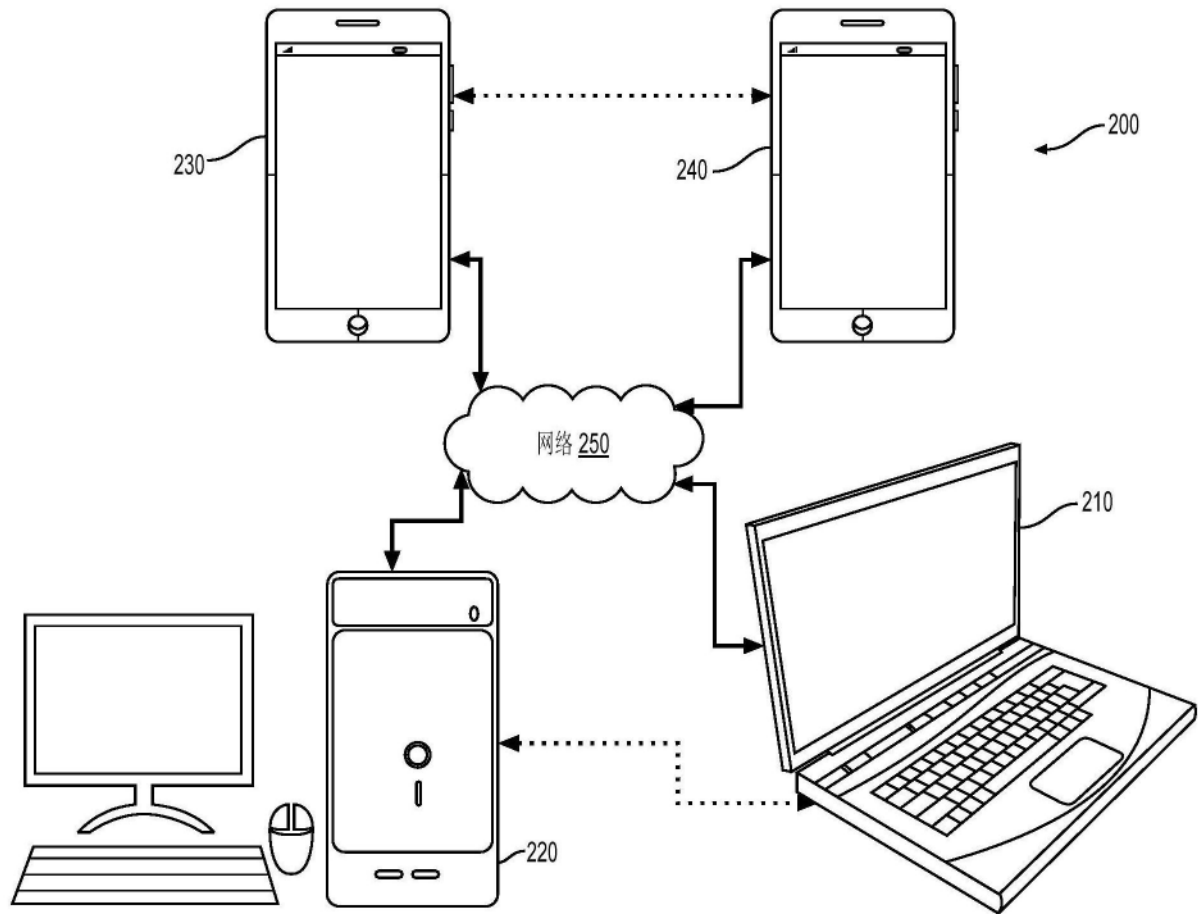


图2

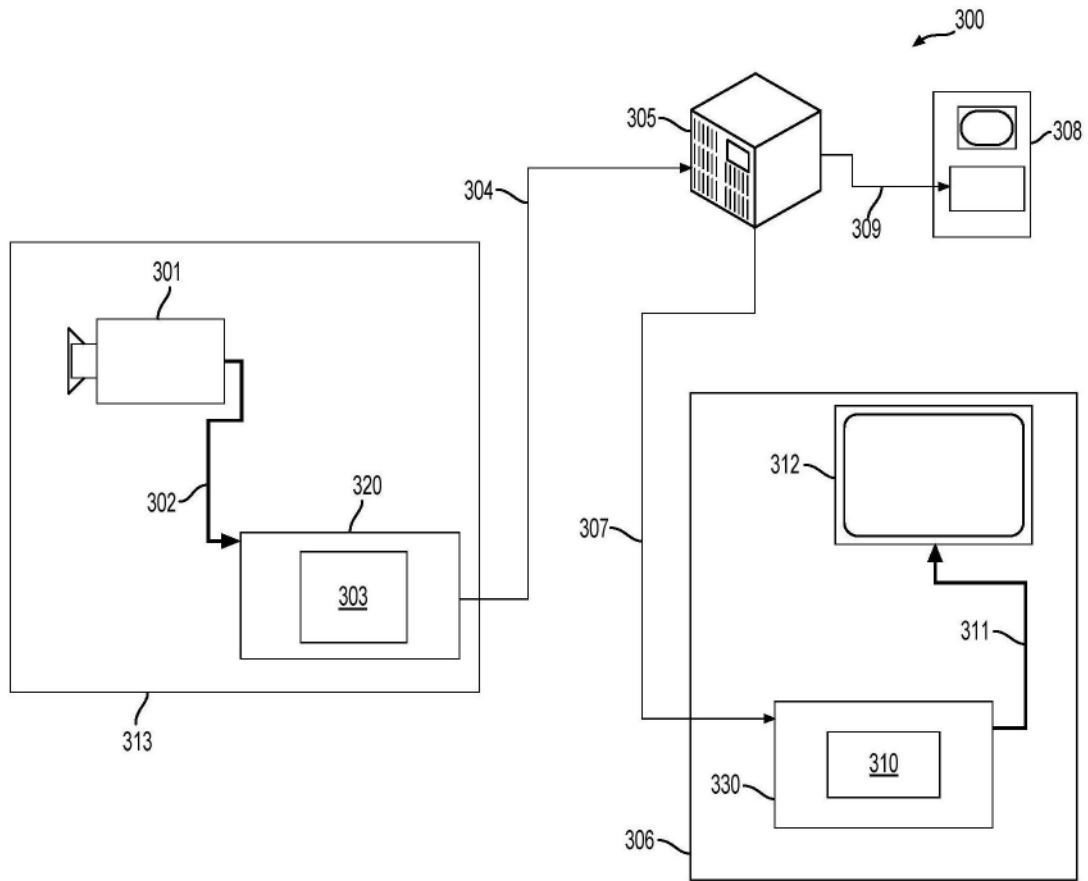


图3

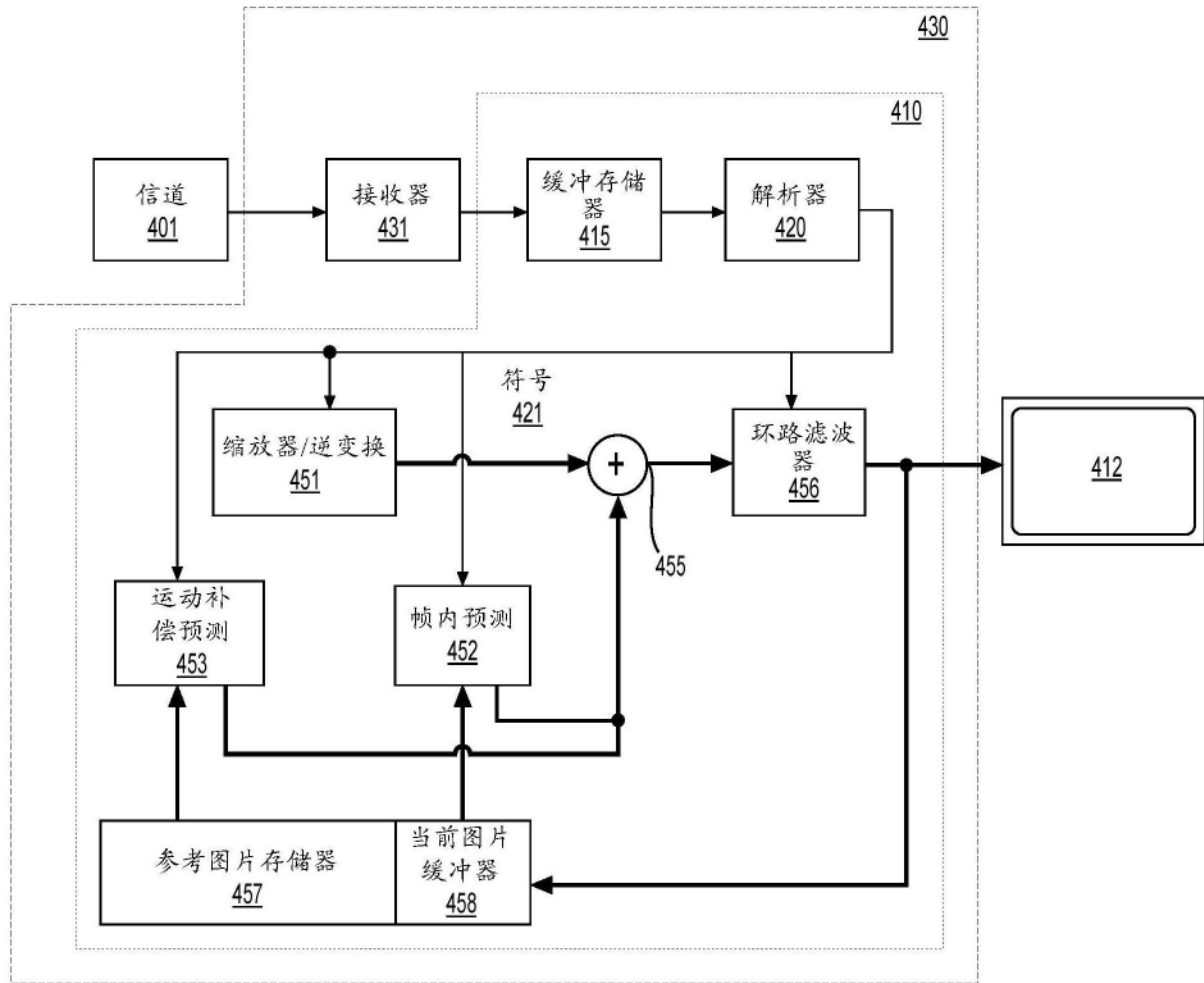


图4



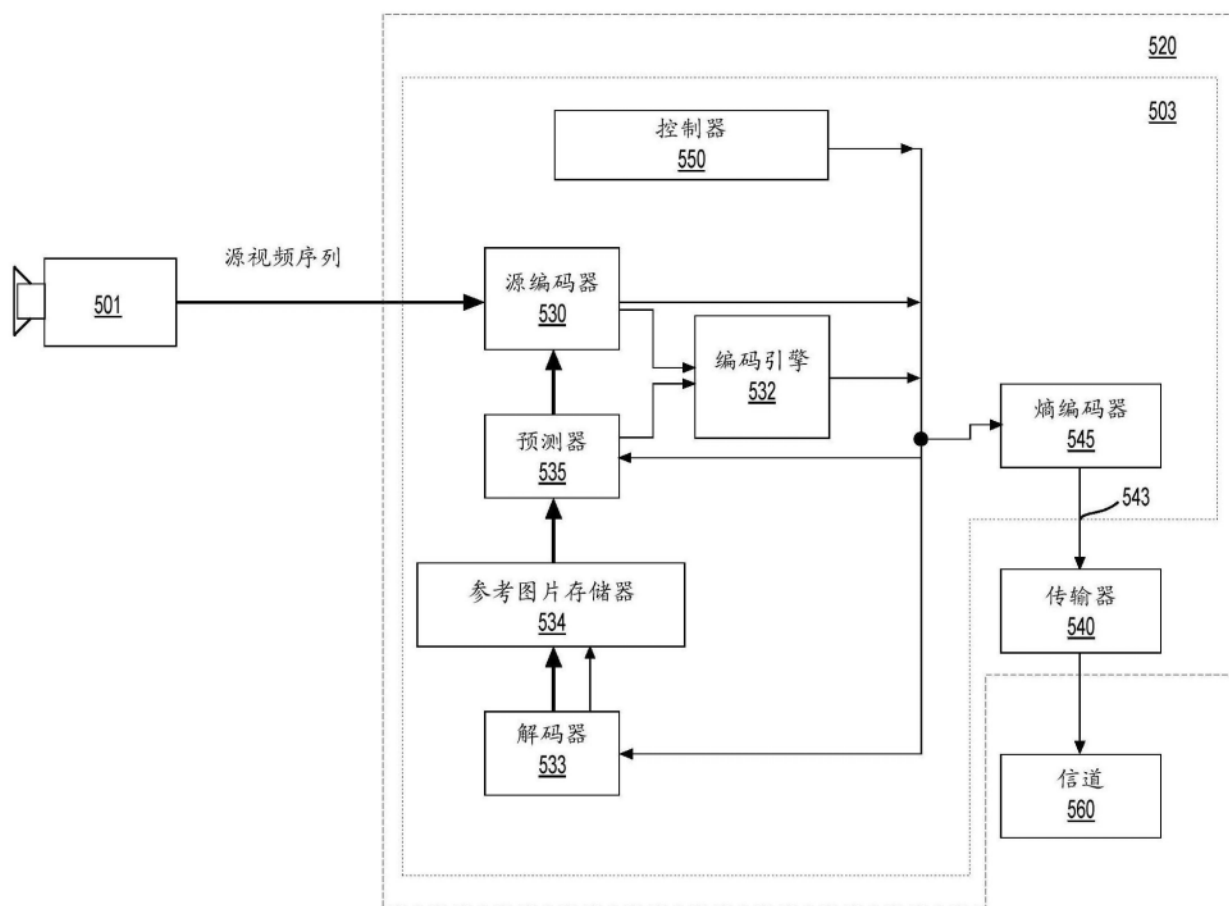


图5

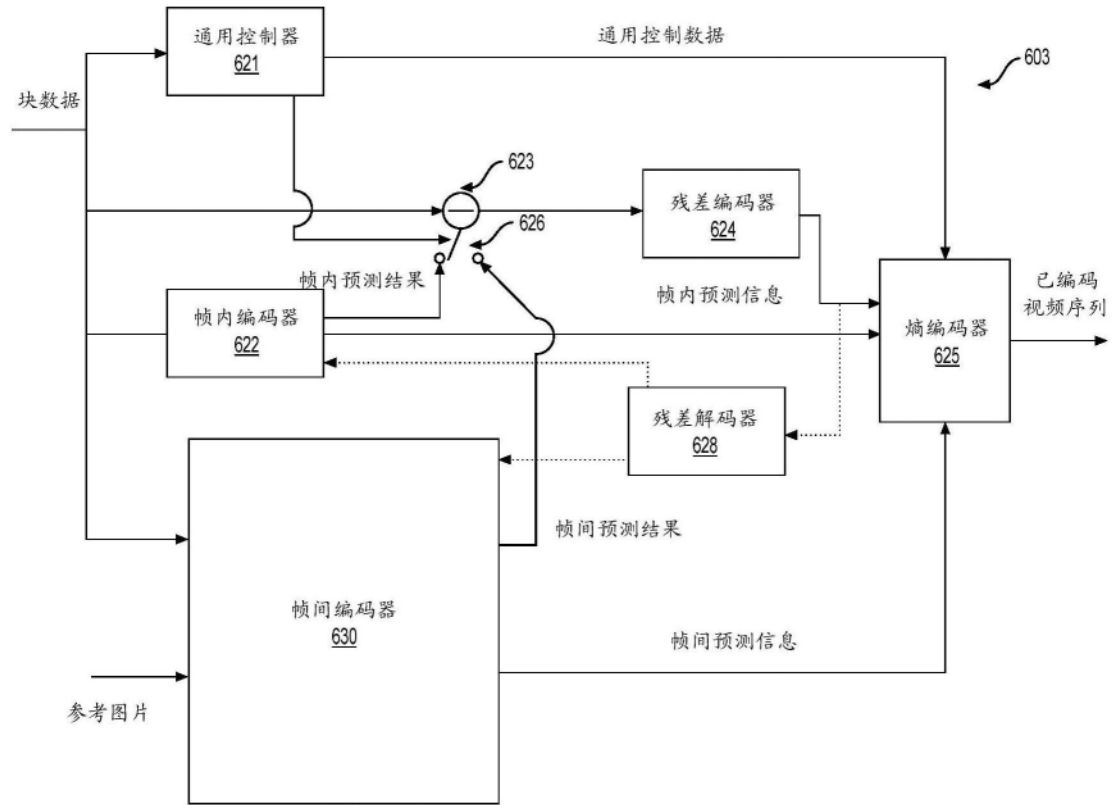


图6

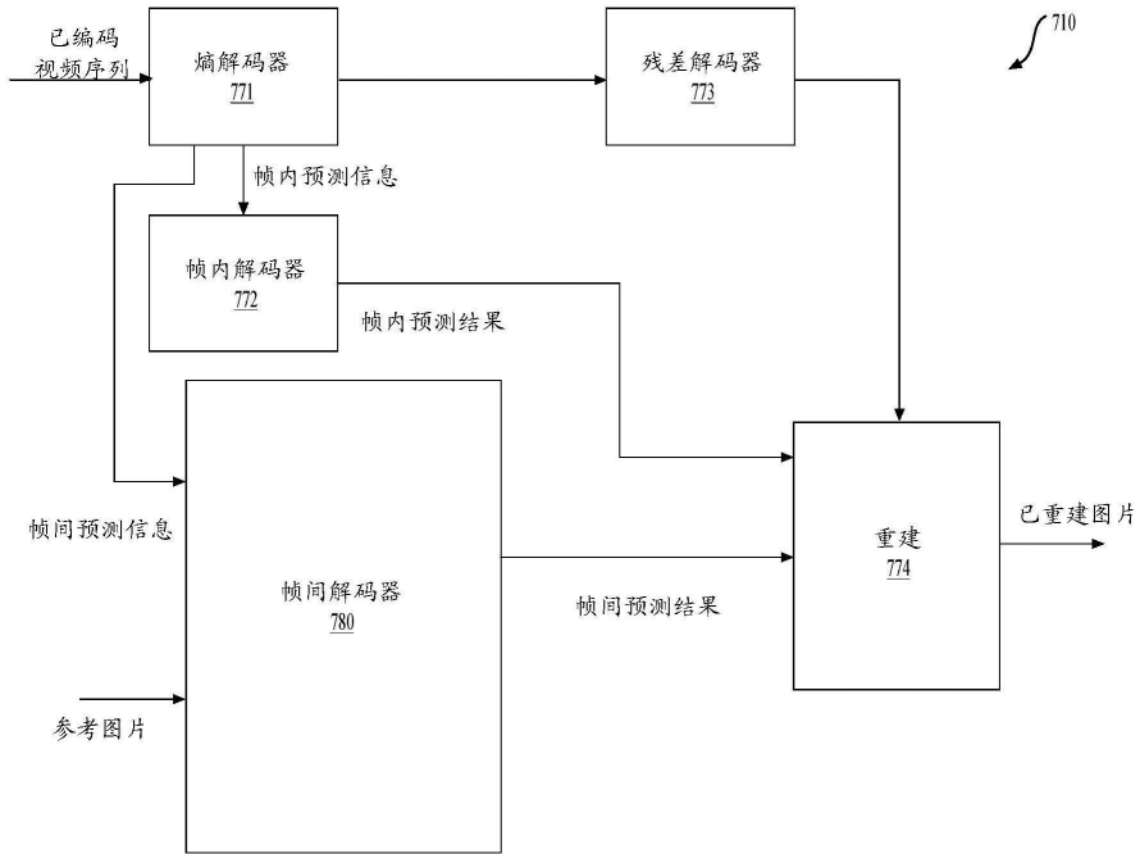


图7

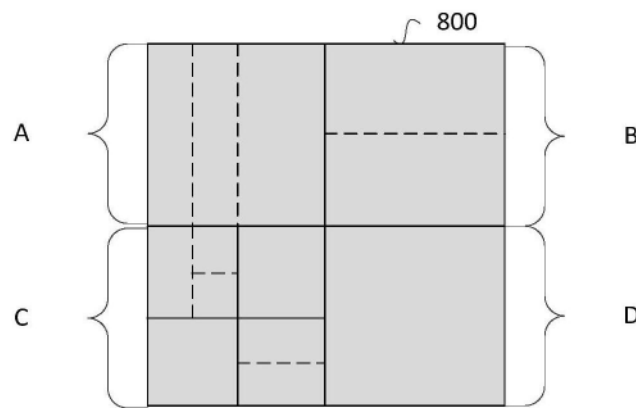


图8A

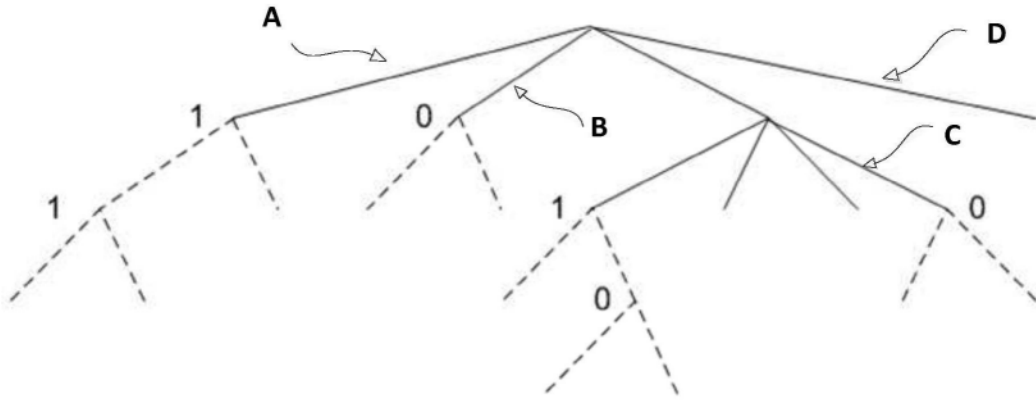


图8B

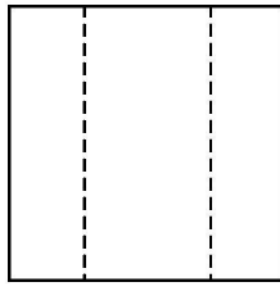


图9A

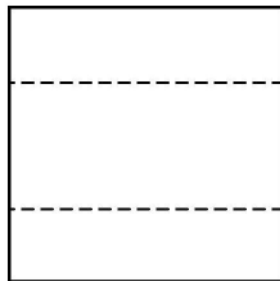


图9B

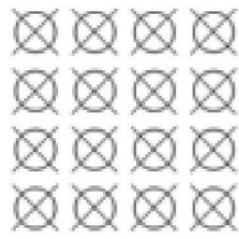


图10A

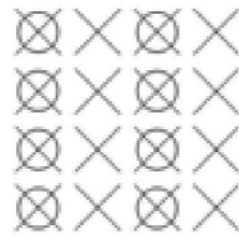


图10B

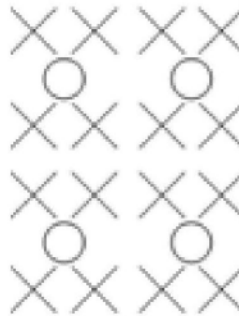
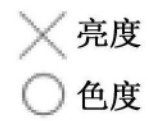


图10C



图10D

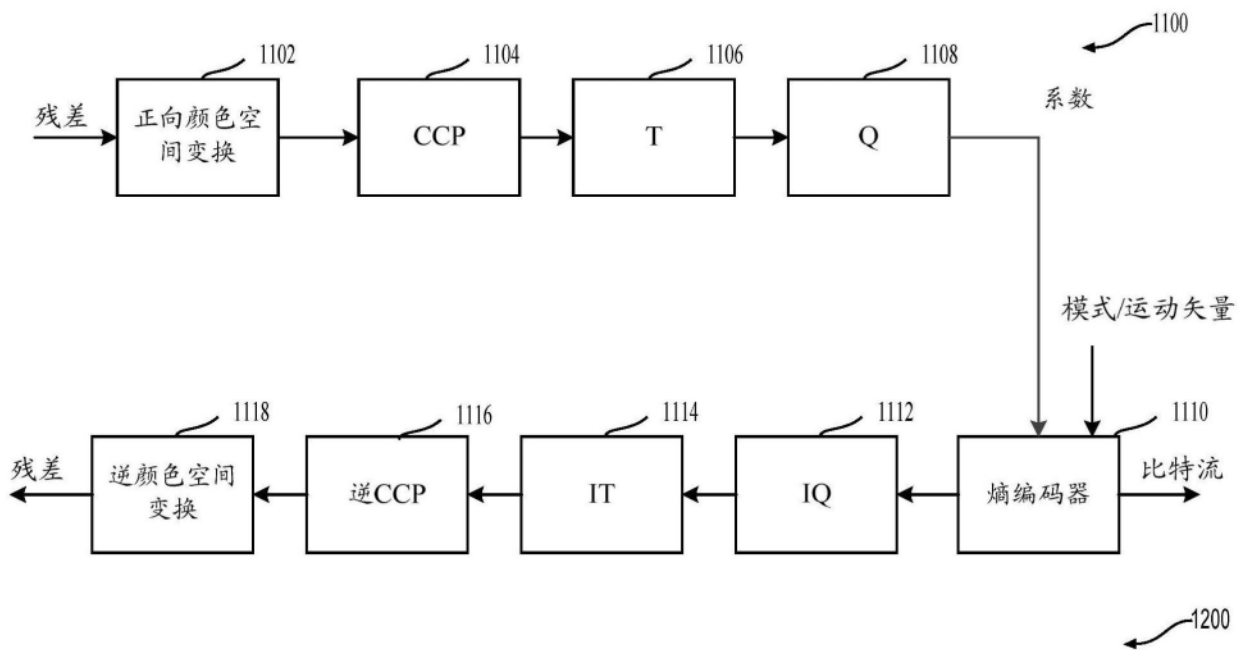


图11

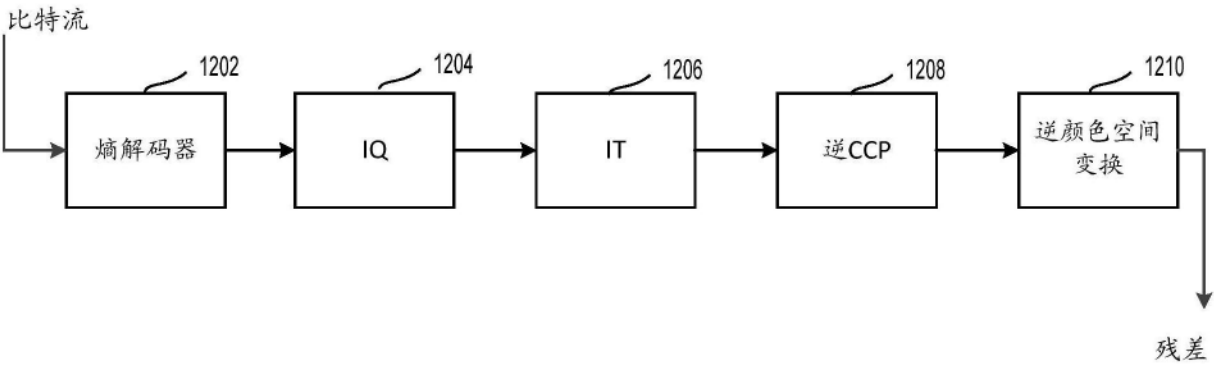


图12

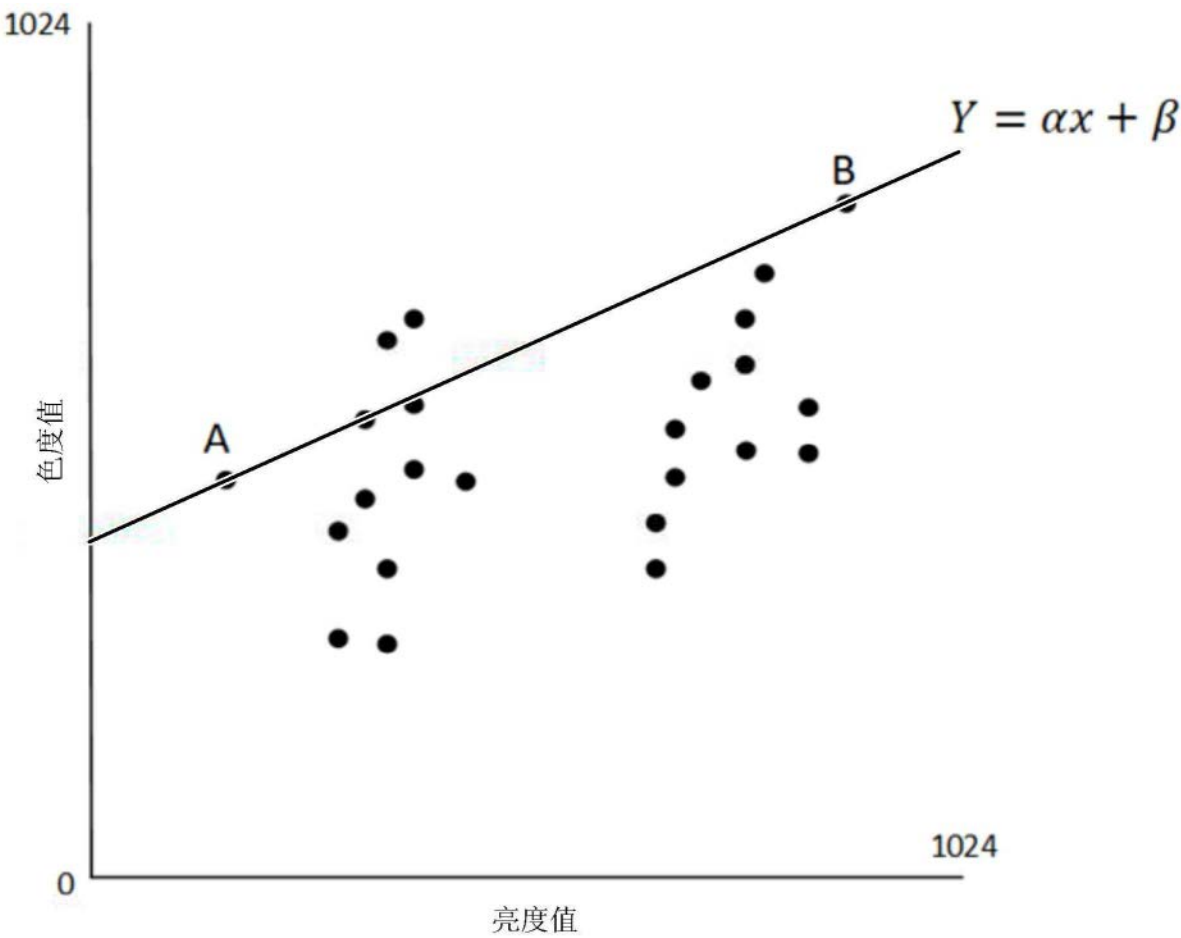


图13

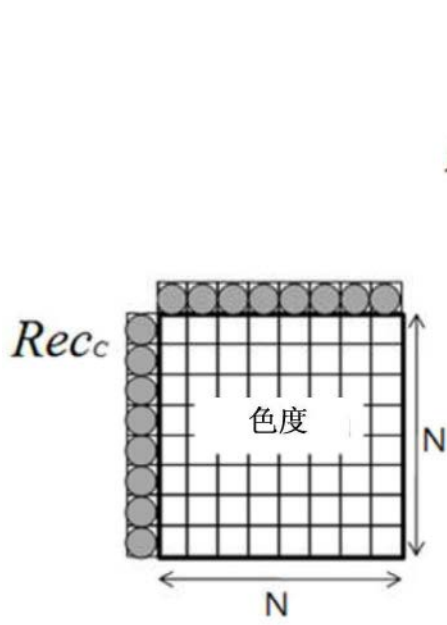


图 14A

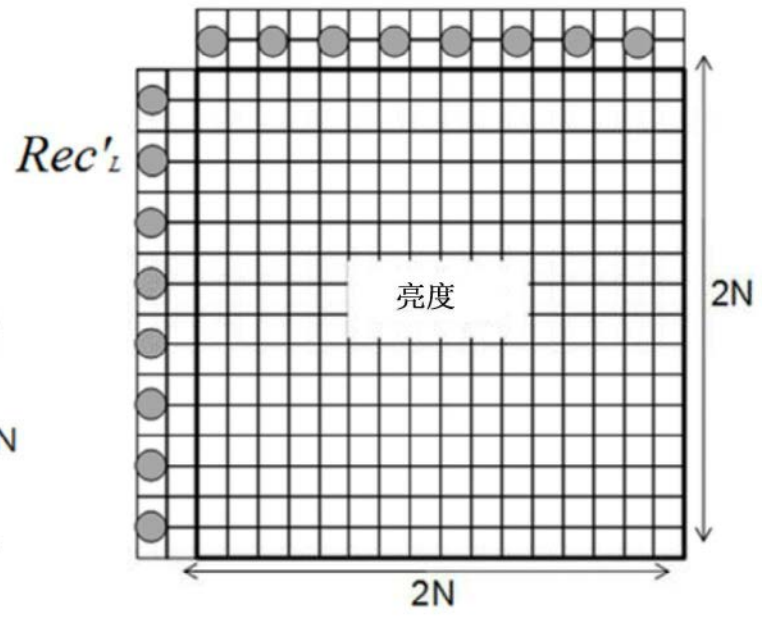


图 14B

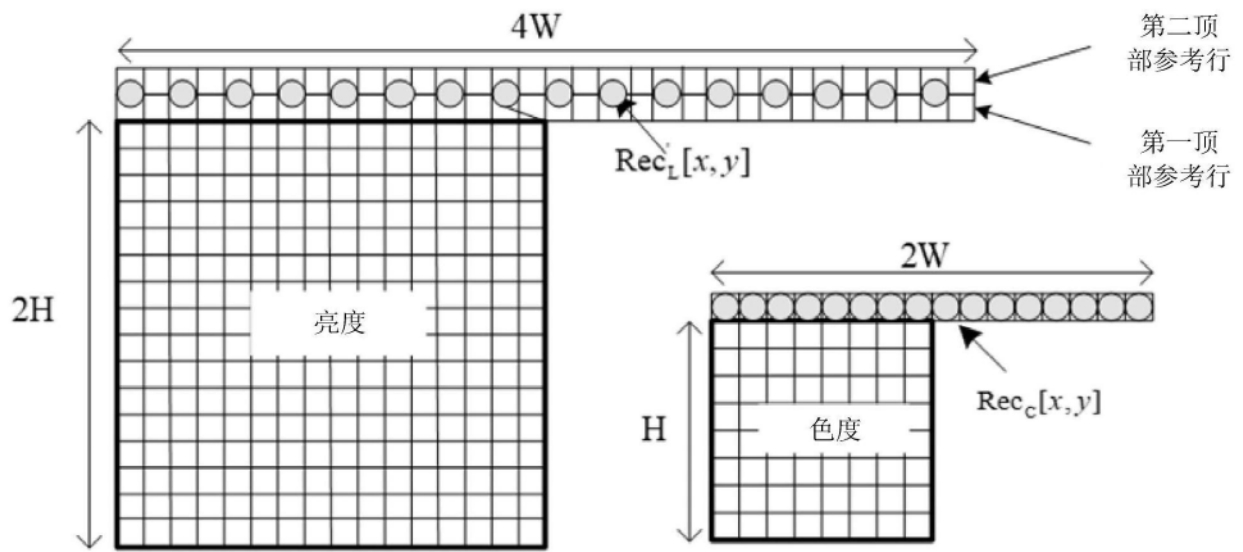


图 15A

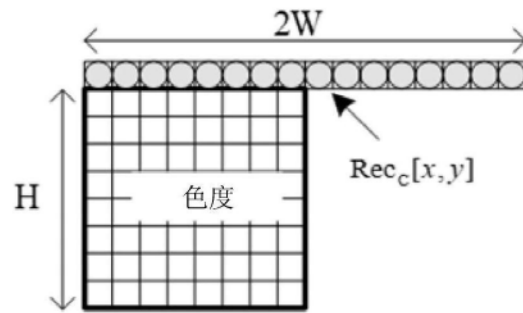


图 15B

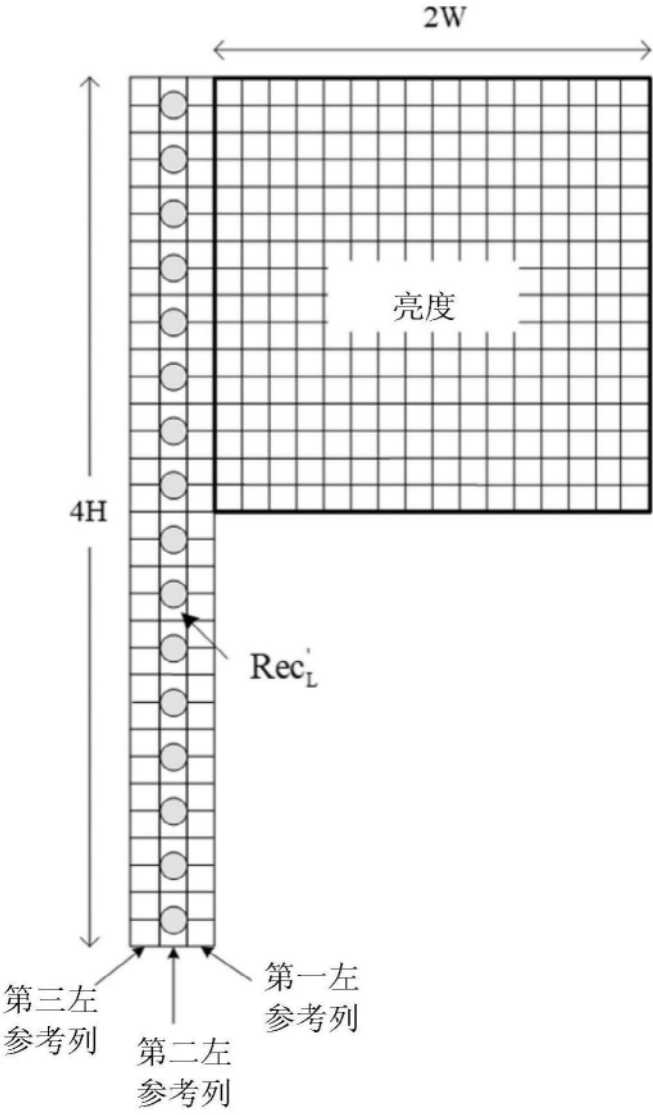


图16A



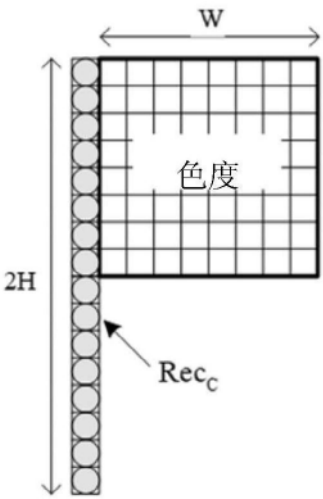


图16B

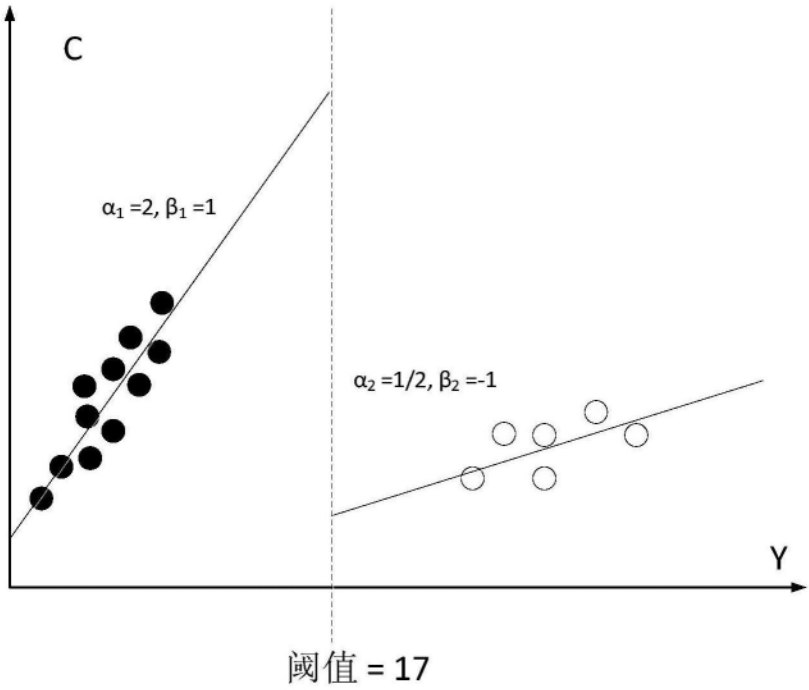


图17

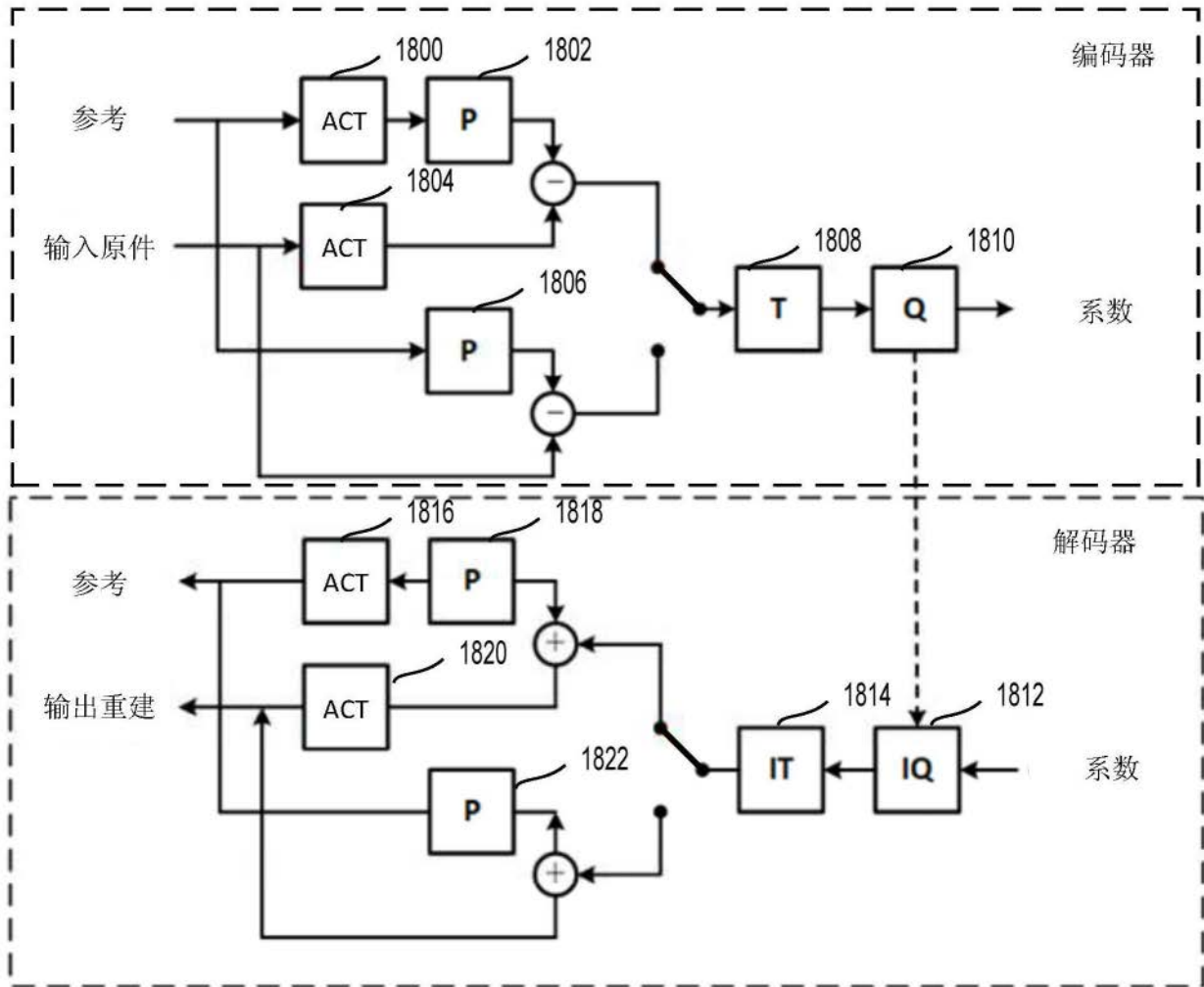


图18

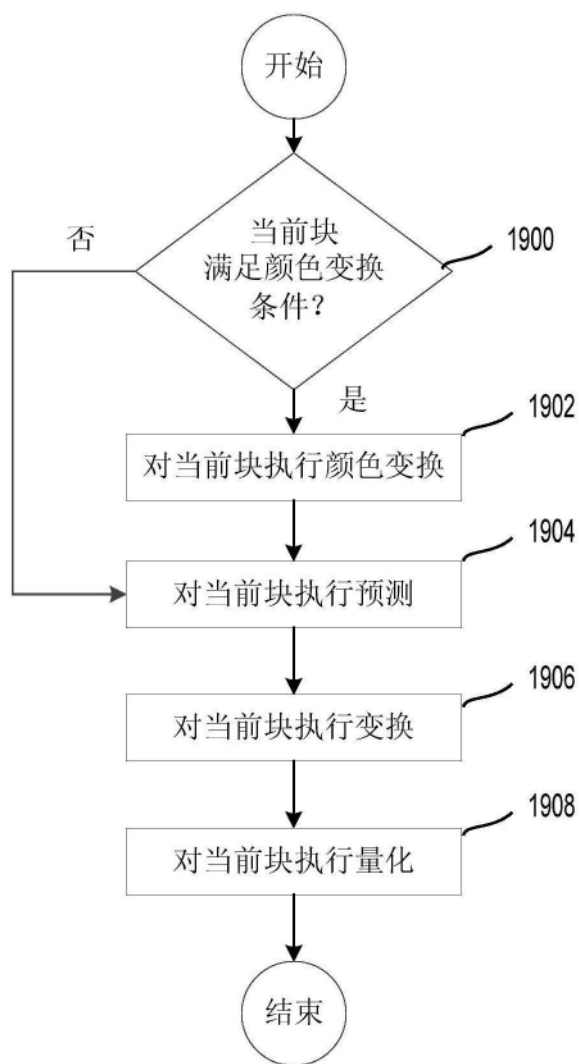


图19

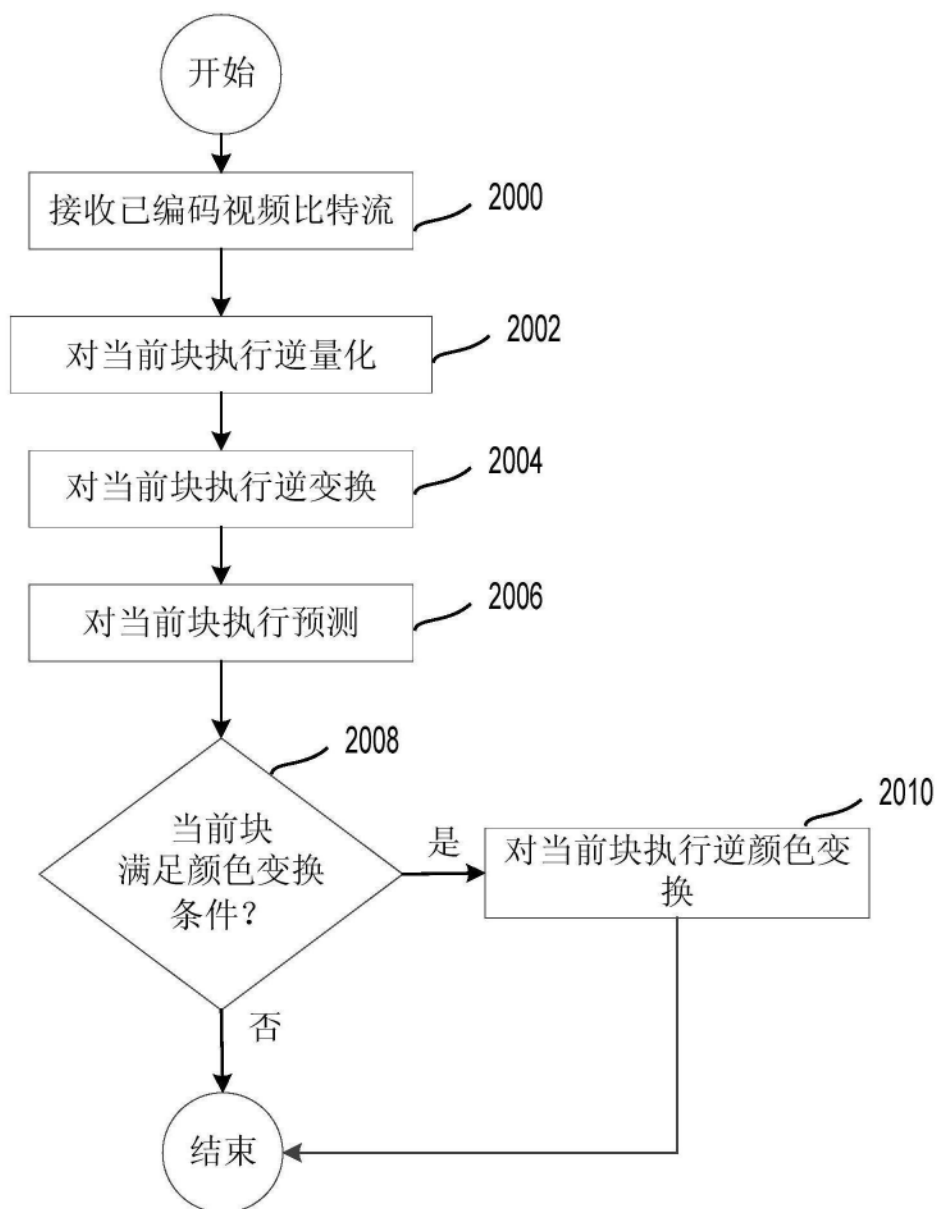


图20

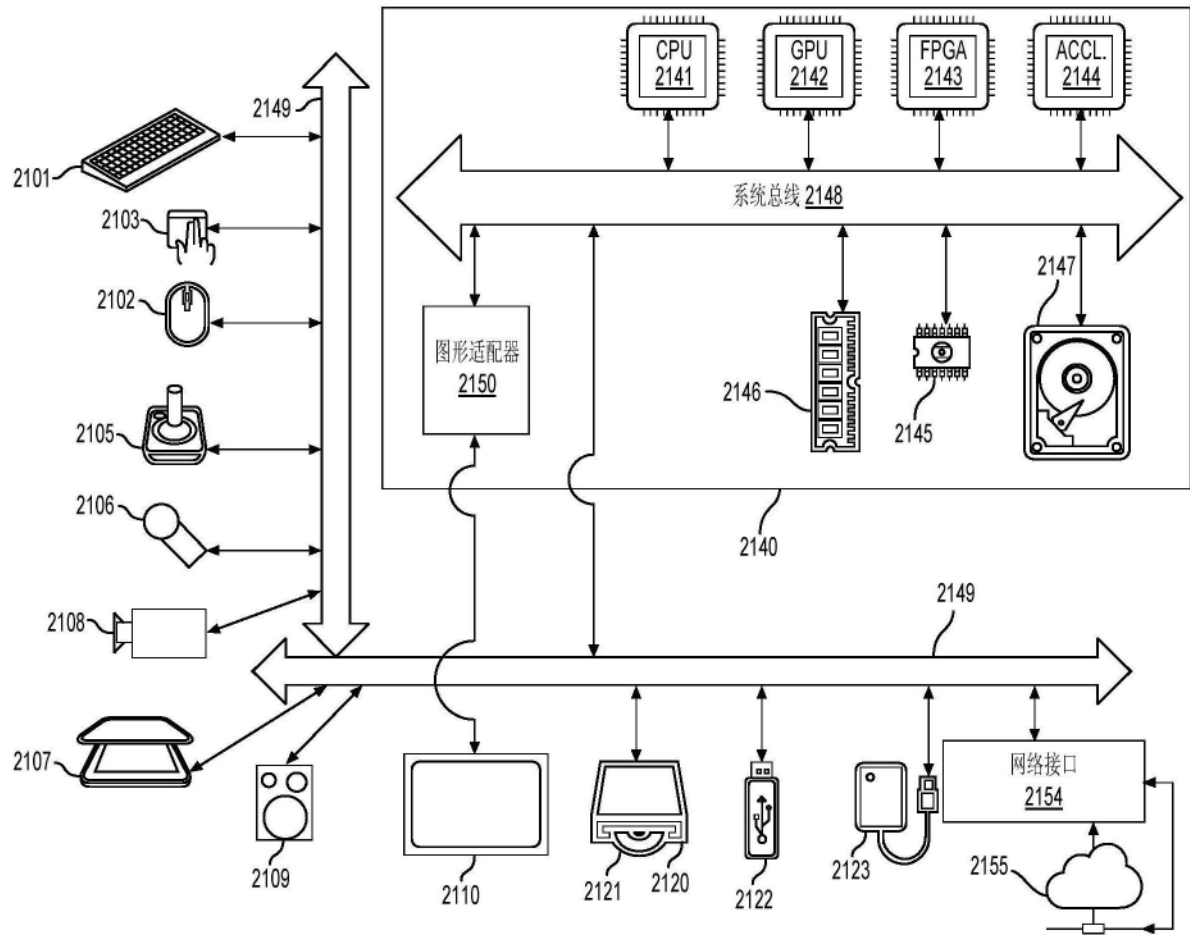


图21