



(21) 申请号 202280086189.3

赵欣 夜静 刘杉

(22) 申请日 2022.11.14

(74) 专利代理机构 北京德琦知识产权代理有限公司 11018

(30) 优先权数据

63/358,309 2022.07.05 US

17/983,876 2022.11.09 US

专利代理师 程杰 王琦

(85) PCT国际申请进入国家阶段日

2024.06.26

(51) Int.Cl.

H04N 19/44 (2006.01)

H04N 19/61 (2006.01)

(86) PCT国际申请的申请数据

PCT/US2022/049781 2022.11.14

(87) PCT国际申请的公布数据

W02024/010607 EN 2024.01.11

(71) 申请人 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 马杜·柏林加色·克里什南 赵亮

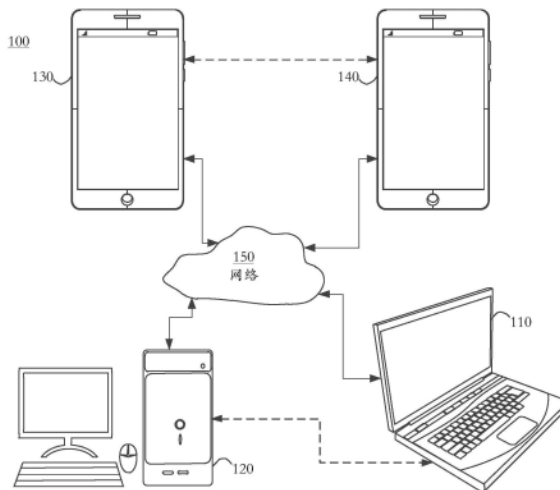
权利要求书2页 说明书19页 附图8页

(54) 发明名称

包括用于多符号算术编码的语法设计的系统和方法

(57) 摘要

提供了使用用于多符号算术编码的语法设计进行编码和解码的系统和方法。一种方法包括接收包括多个语法元素的已编码视频码流；由算术编码引擎确定用于算术编码的第一最大字母表大小，所述第一最大字母表大小是基于硬件约束确定的；确定小于所述第一最大字母表大小的第二最大字母表大小；以及基于所确定的第二最大字母表大小，解码出包括在所述已编码视频码流中的所述多个语法元素，其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。



1. 一种由解码器执行的方法,其特征在于,所述方法包括:
接收包括多个语法元素的已编码视频码流;
由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;
确定小于所述第一最大字母表大小的第二最大字母表大小;以及
基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。
2. 根据权利要求1所述的方法,其特征在于,所述第二最大字母表大小是2和15之间的整数,包括2和15。
3. 根据权利要求1所述的方法,其特征在于,所述多个语法元素包括第一语法元素和第二语法元素,所述第一语法元素是集合索引,指示选择的一个符号在哪个集合中,所述第二语法元素是指示选择的集合中哪个符号用于编码或解码的索引。
4. 根据权利要求3所述的方法,其特征在于,所述符号是运动矢量差的级别,并且所述第一语法元素的字母表大小是2,并且各集合中的运动矢量差的级别的总数分别是3和8。
5. 根据权利要求3所述的方法,其特征在于,所述符号是运动矢量差的级别,并且所述第一语法元素的字母表大小是3,并且各集合中的运动矢量差的级别的总数分别是3、4和4。
6. 根据权利要求3所述的方法,其特征在于,所述符号各自指示从亮度到色度(CfL)模式的缩放参数的绝对值。
7. 根据权利要求3所述的方法,其特征在于,所述符号各自指示亮度帧内模式编码的模式。
8. 根据权利要求3所述的方法,其特征在于,所述符号各自指示色度帧内模式编码的模式。
9. 根据权利要求3所述的方法,其特征在于,所述符号各自指示楔形波图案。
10. 根据权利要求3所述的方法,其特征在于,所述符号各自是块结束(eob)值。
11. 一种系统,包括:
至少一个存储器,被配置为存储计算机程序代码;以及
至少一个处理器,被配置为接收包括多个语法元素的已编码视频码流,访问所述计算机程序代码,并按照所述计算机编程代码的指示进行操作,所述计算机程序代码包括:
第一确定代码,被配置为使所述至少一个处理器由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;
第二确定代码,被配置为使所述至少一个处理器确定小于所述第一最大字母表大小的第二最大字母表大小;以及
解码代码,被配置为使得所述至少一个处理器基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。
12. 根据权利要求11所述的系统,其特征在于,所述第二最大字母表大小是2和15之间

的整数值,包括2和15。

13.根据权利要求11所述的系统,其特征在于,所述多个语法元素包括第一语法元素和第二语法元素,所述第一语法元素是集合索引,指示选择的一个符号在哪个集合中,所述第二语法元素是指示选择的集合中哪个符号用于编码或解码的索引。

14.根据权利要求13所述的系统,其特征在于,所述符号是运动矢量差的级别,并且所述第一语法元素的字母表大小是2,并且各集合中的运动矢量差的级别的总数分别是3和8。

15.根据权利要求13所述的系统,其特征在于,所述符号是运动矢量差的级别,并且所述第一语法元素的字母表大小是3,并且各集合中的运动矢量差的级别的总数分别是3、4和4。

16.根据权利要求13所述的系统,其特征在于,所述符号各自指示从亮度到色度(CfL)模式的缩放参数的绝对值。

17.根据权利要求13所述的系统,其特征在于,所述符号各自指示亮度帧内模式编码的模式。

18.根据权利要求13所述的系统,其特征在于,所述符号各自指示色度帧内模式编码的模式。

19.根据权利要求13所述的系统,其特征在于,所述符号各自指示楔形波图案。

20.一种非易失性计算机可读介质,存储有计算机代码,所述计算机代码被配置为当由至少一个处理器执行时,使所述至少一个处理器实现解码器,所述解码器:

接收包括多个语法元素的已编码视频码流;

由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;

确定小于所述第一最大字母表大小的第二最大字母表大小;以及

基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。

包括用于多符号算术编码的语法设计的系统和方法

相关申请的交叉引用

[0001] 本申请要求2022年7月5日提交的美国临时申请第63/358,309号和2022年11月9日提交的美国申请第17/983,876号的优先权,其公开内容通过引用整体并入本文。

技术领域

[0002] 本公开的实施例涉及高级视频编码技术,更具体地,涉及用于多符号算术编码的语法设计。

背景技术

[0003] AOMedia视频1 (AV1) 是为因特网上的视频传输而设计的开放视频编码格式。它是由开放媒体联盟 (AOMedia) 为接替VP9而开发的,该联盟是创立于2015年的联合企业,包括半导体公司、视频点播提供商、视频内容制作商、软件开发公司和网页浏览器厂商。AV1项目的许多组成部分来源于联盟成员先前的研究工作。个体贡献者在几年前开始实验技术平台:Xiph/Mozilla的Daala于2010年发布了代码,谷歌 (Google) 的实验VP9进化项目VP10于2014年9月12日公布,并且思科 (Cisco) 的Thor于2015年8月11日发布。建立在VP9的代码基础上,AV1结合了附加的技术,其中的几种是以这些实验形式开发的。AV1参考编解码器的第一个版本0.1.0于2016年4月7日发布。该联盟于2018年3月28日公布了AV1码流规范的发行以及参考的基于软件的编码器和解码器。2018年6月25日,发行了规范的确认版本1.0.0。2019年1月8日,发布了一个经过验证的版本1.0.0,其中包含勘误表1。AV1比特流规范包括参考视频编解码器。

发明内容

[0004] 根据实施例,提供一种由解码器执行的方法。所述方法包括:接收包括多个语法元素的已编码视频码流;由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;确定小于所述第一最大字母表大小的第二最大字母表大小;以及基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。

[0005] 根据一个或多个实施例,所述第二最大字母表大小是2和15之间的整数值,包括2和15。

[0006] 根据一个或多个实施例,所述多个语法元素包括第一语法元素和第二语法元素,所述第一语法元素是集合索引,指示选择的一个符号在哪个集合中,所述第二语法元素是指示选择的集合中哪个符号用于编码或解码的索引。

[0007] 根据一个或多个实施例,所述符号是运动矢量差的级别,所述第一语法元素的字母表大小是2,并且各集合中的运动矢量差的级别的总数分别是3和8。

[0008] 根据一个或多个实施例,所述符号是运动矢量差的级别,并且所述第一语法元素

的字母表大小是3,并且各集合中的运动矢量差的级别的总数分别是3、4和4。

[0009] 根据一个或多个实施例,所述符号各自指示从亮度到色度(CfL)模式的缩放参数的绝对值。

[0010] 根据一个或多个实施例,所述符号各自指示亮度帧内模式编码的模式。

[0011] 根据一个或多个实施例,所述符号各自指示色度帧内模式编码的模式。

[0012] 根据一个或多个实施例,所述符号各自指示楔形波图案。

[0013] 根据一个或多个实施例,所述符号各自是块结束(eob)值。

[0014] 根据实施例,提供一种系统。该系统包括:至少一个存储器,被配置为存储计算机程序代码;和至少一个处理器,被配置为接收包括多个语法元素的已编码视频码流,访问所述计算机程序代码,并按照所述计算机编程代码的指示进行操作,所述计算机程序代码包括:第一确定代码,被配置为使所述至少一个处理器由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;第二确定代码,被配置为使所述至少一个处理器确定小于所述第一最大字母表大小的第二最大字母表大小;以及解码代码,被配置为使得所述至少一个处理器基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。

[0015] 根据一个或多个实施例,所述第二最大字母表大小是2和15之间的整数值,包括2和15。

[0016] 根据一个或多个实施例,所述多个语法元素包括第一语法元素和第二语法元素,所述第一语法元素是集合索引,指示选择的一个符号在哪个集合中,所述第二语法元素是指示选择的集合中哪个符号用于编码或解码的索引。

[0017] 根据一个或多个实施例,所述符号是运动矢量差的级别,所述第一语法元素的字母表大小是2,并且各集合中的运动矢量差的级别的总数分别是3和8。

[0018] 根据一个或多个实施例,所述符号是运动矢量差的级别,并且所述第一语法元素的字母表大小是3,并且各集合中的运动矢量差的级别的总数分别是3、4和4。

[0019] 根据一个或多个实施例,所述符号各自指示从亮度到色度(CfL)模式的缩放参数的绝对值。

[0020] 根据一个或多个实施例,所述符号各自指示亮度帧内模式编码的模式。

[0021] 根据一个或多个实施例,所述符号各自指示色度帧内模式编码的模式。

[0022] 根据一个或多个实施例,所述符号各自指示楔形波图案。

[0023] 根据实施例,提供一种非易失性计算机可读介质,存储有计算机代码。所述计算机代码被配置为当由至少一个处理器执行时,使所述至少一个处理器实现解码器,所述解码器:接收包括多个语法元素的已编码视频码流;由算术编码引擎确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;确定小于所述第一最大字母表大小的第二最大字母表大小;以及基于所确定的第二最大字母表大小,解码出包括在所述已编码视频码流中的所述多个语法元素,其中所述多个语法元素中的每一个语法元素被以小于或等于所确定的第二最大字母表大小的字母表大小熵编码。

附图说明

- [0024] 从以下详细描述和附图中,所公开的主题的其它特征、性质以及各种优点将更加明显。
- [0025] 图1是根据实施例的通信系统的简化框图的示意图。
- [0026] 图2是根据实施例的通信系统的简化框图的示意图。
- [0027] 图3是根据实施例的解码器的简化框图的示意图。
- [0028] 图4是根据实施例的编码器的简化框图的示意图。
- [0029] 图5是根据实施例的用于解码二进制数的过程的示例流程图。
- [0030] 图6示出了根据一个实施例的在亮度帧内模式编码中使用的当前块的相邻块。
- [0031] 图7是从亮度到色度(CfL)预测过程的示意图。
- [0032] 图8是根据一个实施例的计算机代码的框图。
- [0033] 图9是适用于实现本公开的实施例的计算机系统的示意图。

具体实施方式

[0034] 以下示例实施例的详细描述参考附图。不同附图中的相同附图标记可以标识相同或相似的元件。

[0035] 上述公开提供了说明和描述,但并非旨在穷举或将实施方式限制为所公开的精确形式。根据本公开,修改和变化是可能的,或者可以从实施方式的实践中获得。此外,一个实施例的一个或多个特征或部件可以并入另一实施例(或另一实施方案的一个或者多个特征)中或者与另一实施方式组合。此外,在下面提供的流程图和操作描述中,可以理解的是,可以省略一个或多个操作,可以添加一个或多个操作,一个或多个操作可以同时(至少部分地)执行,并且可以切换一个或多个操作的顺序。

[0036] 很明显,本文所述的系统和/或方法可以以不同形式的硬件、软件或硬件和软件的组合来实现。用于实现这些系统和/或方法的实际专用控制硬件或软件代码并不限制这些实现。应当理解,软件和硬件可以被设计为基于本文的描述来实现系统和/或方法。

[0037] 以下描述的特征可以单独使用,也可以按任何顺序组合使用。此外,实施例可以通过处理电路(例如,一个或多个处理器或一个或多个集成电路)来实现。在一个示例中,一个或多个处理器执行存储在非瞬态计算机可读介质中的程序。

[0038] 除非明确说明,否则本文中使用的任何元素、行为或指令都不应被解释为关键或必要的。此外,如本文所用,术语“a”和“an”旨在包括一个或多个项目,并可与“一个或多个”互换使用。如果仅用于一个项目,则使用术语“一个”或类似语言。此外,如本文中所使用的,术语“具有(has)”、“有(have)”、“拥有(having)”、“包括(include)”、“包含(including)”等旨在是开放式术语。此外,除非另有明确说明,否则“基于”一词的意思是“至少部分基于”。此外,诸如“A和B中的至少一个”或“A或B中的最少一个”的表达应理解为仅包括A、仅包括B或同时包括A和B。

[0039] 图1示出了根据本公开的实施例的通信系统100的简化框图。通信系统100可以包括经由网络150互连的至少两个终端110和120。对于数据的单向传输,第一终端110可以在本地位置对视频数据进行编码,以便经由网络150传输到另一终端120。第二终端120可以从网络150接收另一终端的编码视频数据,对编码数据进行解码并显示恢复的视频数据。单向

数据传输在媒体服务应用等中是较常见的。

[0040] 图1示出了被提供来支持编码视频的双向传输的第二对终端130和140,该编码视频可以发生在例如视频会议期间。对于数据的双向传输,终端130和140中每个终端可以对在本地位置采集的视频数据进行编码,以便经由网络150传输到另一个终端。终端130和140中每个终端还可以接收由另一个终端发送的编码视频数据,可以解码编码数据,并且可以在本地显示设备上显示恢复的视频数据。

[0041] 在图1的实施例中,终端110、终端120、终端130和终端140可为服务器、个人计算机和智能电话,和/或其他类型的终端。例如,终端110-140可以是膝上型计算机、平板电脑、媒体播放器和/或专用视频会议设备。网络150表示在终端110、终端120、终端130和终端140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150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。出于本论述的目的,除非在下文中有所解释,否则网络150的架构和拓扑对于本申请公开的操作来说可能是无关紧要的。

[0042] 作为实施例,图2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。本申请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。

[0043] 如图2所示,流传输系统200可以包括采集子系统213,采集子系统可以包括视频源201和编码器203。视频源201例如可以是数码相机,可以被配置为创建例如未压缩的视频采样流202。相较于编码视频码流,未压缩的视频采样流202可以提供高数据量,并且可以由耦合到视频源201的编码器203来处理。编码器203可以包括硬件、软件或其组合,以实现或实现如以下更详细描述的主题的各方面。已编码的视频码流204可以被存储在流服务器205上以供将来使用,相较于采样流,已编码的视频码流204可以包括较低的数据量。一个或多个流客户端206可以访问流服务器205以检索视频码流209,视频码流209例如为已编码视频码流204的副本。

[0044] 在实施例中,流式传输服务器205也可以用作媒体感知网元(Media-Aware Network Element, MANE)。例如,流式传输服务器205可以被配置为修剪编码的视频比特流204,以针对流式传输客户端206中的一个或多个定制潜在的不同比特流。在实施例中,MANE可以与流传输系统200中的流传输服务器205分开提供。

[0045] 流式传输客户端206可以包括视频解码器210和显示器212。视频解码器210可以例如解码视频码流209,其是已编码视频码流的传入副本204,并且创建可在显示器212或另一呈现设备(未示出)上呈现的传出视频采样流211。在一些流传输系统中,可以根据某些视频编码/压缩标准对视频码流204和209进行编码。这样的标准的示例包括但不限于ITU-T建议H.265。正在开发一种视频编码标准,非正式地称为通用视频编码(VVC)。本公开的实施例可以在VVC的上下文中使用。

[0046] 图3示出了根据本公开的实施例的附接到显示器212的视频解码器210的示例功能框图。

[0047] 视频解码器210可包括信道312、接收器310、缓冲存储器315、熵解码器/解析器320、缩放器/逆变换单元351、帧内图片预测单元352、运动补偿预测单元353、聚合器355、环路滤波器单元356、参考图片存储器357和当前图片存储器。在至少一个实施例中,视频解码

器210可以包括集成电路、一系列集成电路和/或其他电子电路。视频解码器210也可以部分或全部体现在具有相关存储器的一个或多个CPU上运行的软件中。

[0048] 在本实施例和其他实施例中,接收器310可以接收一个或多个编码视频序列,以由解码器210解码,每次一个编码视频顺序,其中每个编码视频顺序的解码独立于其他编码视频顺序。可以从信道312接收编码的视频序列,该信道可以是到存储编码的视频数据的存储设备的硬件/软件链路。接收器310可以接收编码的视频数据和其他数据,例如,编码的音频数据和/或辅助数据流,这些数据可以被转发到它们各自的使用实体(未示出)。接收器310可以将编码的视频序列与其他数据分离。为了对抗网络抖动,缓冲存储器315可以耦合在接收器310和熵解码器/解析器320(以下称为“解析器”)之间。当接收器310正在从具有足够带宽和可控性的存储/转发设备接收数据时,或者从等同步网络接收数据,缓冲存储器315可以不被使用,或者可以很小。为了在诸如因特网的尽力而为分组网络上使用,缓冲存储器315可能是必需的,可以相对较大,并且可以具有自适应大小。

[0049] 视频解码器210可包括解析器320以根据已编码视频序列重建符号321。这些符号的类别例如包括用于管理视频解码器210的操作的信息,以及用以控制显示器212等显示装置的潜在信息,所述显示装置可耦接到如图2所示的解码器。用于显示装置的控制信息可以是例如辅助增强信息(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320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,且可遵循本领域技术人员熟知的原理,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320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320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。

[0050] 解析器320可以对从缓冲存储器315接收的视频序列执行熵解码/解析操作,从而创建符号321。

[0051] 符号321的重建可能涉及多个不同的单元,这取决于编码视频画面或其部分的类型(例如:画面间和画面内、块间和块内)以及其他因素。哪些单元被涉及,以及它们是如何被涉及的,可以由解析器从编码视频序列解析的子组控制信息来控制320。为了清楚起见,下面没有描述解析器320和多个单元之间的这种子组控制信息的流。

[0052] 除已经提及的功能块以外,解码器210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。

[0053] 一个单元是缩放器/逆变换单元351。缩放器/逆变换单元351从解析器320接收作为符号321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351可输出包括样本值的块,所述样本值可输入到聚合器355中。

[0054] 在一些情况下,缩放器/逆变换单元351的输出样本可属于帧内编码块;即:不使用

来自先前重建的图像的预测性信息,但可使用来自当前图像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352提供。在一些情况下,帧内图片预测单元352采用从当前图片存储器358的当前(部分已重建)图片提取的周围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。在一些情况下,聚合器355基于每个样本,将帧内图片预测单元352生成的预测信息添加到由缩放器/逆变换单元351提供的输出样本信息中。

[0055] 在其它情况下,缩放器/逆变换单元351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353可访问参考图片存储器357以提取用于预测的样本。在根据符号321对提取的样本进行运动补偿之后,这些样本可由聚合器355添加到缩放器/逆变换单元351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。运动补偿单元从参考图片存储器357内的地址获取预测样本可受到运动矢量控制。所述运动矢量以所述符号321的形式而供运动补偿预测单元353使用,所述符号321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357提取的样本值的内插、运动矢量预测机制等等。

[0056] 聚合器355的输出样本可在环路滤波器单元356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码流中的参数,且所述参数作为来自解析器320的符号321可用于环路滤波器单元356。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)部分期间获取的元信息,以及响应于先前重建且经过环路滤波的样本值。

[0057] 环路滤波器单元356的输出可以是样本流,所述样本流可输出到例如显示器的显示器212以及存储在参考图片存储器357,以用于后续的帧间图片预测。

[0058] 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。一旦已编码图片被完全重建,且已编码图片(通过例如解析器320)被识别为参考图片,则当前参考图片可变为参考图片存储器357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存储器。

[0059] 视频解码器210可根据例如ITU-T H.265标准中记录的预定视频压缩技术执行解码操作。已编码视频序列可以符合由所使用的视频压缩技术或标准所指定的语法,在这种意义上,它遵循视频压缩技术或者标准的语法,如在视频压缩技术文档或标准中指定的,并且具体地在其中的简档文档中指定的。为了符合某些视频压缩技术或标准,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。

[0060] 在实施例中,接收器310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210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 to noise ratio,SNR)增强层、冗余切片、冗余图片、前向纠错码等形式。

[0061] 图4示出了根据本公开的实施例的与视频源201相关联的视频编码器203的示例功能框图。

[0062] 视频编码器203可以包括,例如,作为源编码器430、编码引擎432、(本地)解码器433、参考图片存储器434、预测器435、发射机440、熵编码器445、控制器450和信道460的编码器。

[0063] 编码器203可以从视频201(不是编码器的一部分)接收视频样本,视频源201可以采集要由编码器203编码的视频图像。

[0064] 视频源201可提供将由视频编码器203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 CrCb4:2:0、Y CrCb 4:4:4)。在媒体服务系统中,视频源201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201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。

[0065] 根据实施例,编码器203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443。施行适当的编码速度是控制器450的一个功能。控制器450还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450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 λ 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。本领域技术人员可以容易地识别控制器450的其他功能,因为它们可能涉及针对特定系统设计优化的视频编码器203。

[0066] 一些视频编码器在本领域技术人员容易认识的编码环路中进行操作。作为简单的描述,编码环路可包括源编码器430的编码部分(负责基于待编码的输入图片和参考图片创建符号)和嵌入于编码器203中的(本地)解码器433。解码器433创建(远程)解码器也创建的样本数据(因为某些视频压缩技术中,符号与已编码视频码流之间的任何压缩是无损的)。将重建的样本流输入到参考图片存储器434。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内容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)是本领域技术人员熟知的。

[0067] “本地”解码器433的操作可与例如已在上文结合图3详细描述“远程”解码器210相同。然而,当符号可用且熵编码器445和解析器320能够无损地将符号编码/解码为已编码视频序列时,包括信道312、接收器310、缓冲存储器315和解析器320在内的解码器210的熵解码部分,可能无法完全在本地解码器433中实施。

[0068] 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。由于这个原因,所公开的主题集中在解码器操作上。可简化编码器技术的描述,因为编码器技术与全面地描述解码器技术互逆。仅在某些区域中需要更详细的描述,并且在下文提供。

[0069] 作为其部分操作,源编码器430可执行运动补偿预测编码。参考来自视频序列中被

指定为“参考帧”的一个或多个先前已编码帧,所述运动补偿预测编码对输入帧进行预测性编码。以此方式,编码引擎432对输入帧的像素块与参考帧的像素块之间的差异进行编码,所述参考帧可被选作所述输入帧的预测参考。

[0070] 本地解码器433可基于源编码器430创建的符号,对可指定为参考帧的帧的已编码视频数据进行解码。编码引擎432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4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。本地解码器433复制解码过程,所述解码过程可由视频解码器对参考帧执行,且可使重建的参考帧存储在参考图片存储器434中。以此方式,编码器203可在本地存储重建的参考帧的副本,所述副本与将由远端视频解码器获取的重建参考帧具有共同内容(不存在传输误差)。

[0071] 预测器435可针对编码引擎432执行预测搜索。即,对于将要编码的新帧,预测器435可在参考图片存储器434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435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435获取的搜索结果,可确定输入图片可具有从参考图片存储器434中存储的多个参考图片取得的预测参考。

[0072] 控制器450可管理源编码器430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。

[0073] 可在熵编码器445中对所有上述功能单元的输出进行熵编码。熵编码器根据例如霍夫曼编码、可变长度编码、算术编码等本领域技术人员知道的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。

[0074] 传输器440可缓冲由熵编码器445创建的已编码视频序列,从而为通过通信信道460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440可将来自源编码器430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)。

[0075] 控制器450可管理编码器203的操作。在编码期间,控制器450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帧内图片(I图片),预测性图片(P图片)或者双向预测性图片(B图片):

[0076] 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。

[0077] 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。

[0078] 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。

[0079] 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非预测编码。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非预测编码。

[0080] 视频编码器203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203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。

[0081] 在实施例中,传输器440可在传输已编码的视频时传输附加数据。源编码器430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、补充增强信息(Supplementary Enhancement Information, SEI)消息、视觉可用性信息(Visual Usability Information, VUI)参数集片段等。

[0082] [HEVC和VVC中的上下文自适应算术编码引擎]

[0083] HEVC和VVC中的上下文自适应算术编码(context-adaptive arithmetic coding, CABAC)引擎可以在64种不同代表性概率状态之间使用基于表的概率转移过程。在HEVC中,表示编码引擎状态的范围ivlCurrRange可以在计算新的区间范围之前被量化为一组四个值。HEVC状态转移可以被实施为使用包含所有 64×4 的8位预先计算的值的表来近似估计 $ivlCurrRange * pLPS(pStateIdx)$ 的值,其中pLPS是最不可能的符号(least probable symbol, LPS)的概率并且pStateIdx是当前状态的索引。此外,可以使用预先计算的查找表(LUT)来实施解码判定。首先,ivlLpsRange可以使用LUT获得,如下面等式1所示。然后,ivlLpsRange可以用于更新ivlCurrRange并且计算输出binVal。

$$ivlLpsRange = rangeTabLps[pStateIdx][qRangeIdx] \quad (\text{等式1})$$

[0084] 在VVC中,概率可以由概率索引pStateIdx线性表示。因此,在没有LUT操作的情况下可以用等式完成所有的计算。为了提高概率估计的准确性,可以应用多假设概率更新模型。在二进制算术编码器的区间细分中使用的pStateIdx可以是两个概率pStateIdx0和pStateIdx1的组合。这两个概率可以与每个上下文模型相关联,并且可以用不同的自适应速率独立地更新。每个上下文模型的pStateIdx0和pStateIdx1的自适应速率可以基于相关联的二进制数的统计被预训练。概率估计pStateIdx可以是来自两个假设的估计的平均值。

[0085] 图5图示了用于解码单个二进制判定的过程(500)的实施例。过程500可以在操作502处开始以确定变量ivlCurrRange的值。在操作504处,如果变量ivlCurrRange小于或等于变量ivl0ffset,则过程进行到操作506以更新变量binVal、ivl0ffset和ivlCurrRange的值。如果变量ivlCurrRange不小于变量ivl0ffset的值,则过程进行到操作508以更新变量binVal的值。该过程从操作506或操作508进行到操作510以更新变量pStateIdx0和pStateIdx1。过程进行到操作512以执行RenormD过程。

[0086] 如HEVC中所执行的,VVC CABAC还可以具有在每个切片的开始处调用的量化参数(QP)相关初始化过程。给定用于切片的亮度QP的初始值,可以导出上下文模型的初始概率状态(表示为preCtxState),如下在等式2至4中所示:

$$m = slopeIdx \times 5 - 45 \quad (\text{等式2})$$

$$n = (\text{offsetIdx} \ll 3) + 7 \text{ (等式3)}$$

$$\text{preCtxState} = \text{Clip3}(1, 127, ((m \times (QP - 32)) \gg 4) + n) \text{ (等式4)}$$

其中slopeIdx和offsetIdx可以被限制为3位,并且总初始化值可以由6位精度表示。概率状态preCtxState可以直接表示线性域中的概率。因此,参考下面的等式5至6,preCtxState在输入到算术编码引擎之前可能只需要适当的移位操作,并且保存了对数到线性域映射以及256字节的表。

$$\text{pStateIdx0} = \text{preCtxState} \ll 3 \text{ (等式5)}$$

$$\text{pStateIdx1} = \text{preCtxState} \ll 7 \text{ (等式6)}$$

[0087] [AV1中的多符号算术编码]

[0088] 在AV1中,M进制算术编码引擎可以用于对语法元素进行熵编码。每个语法元素可以与M个元素的字母表相关联,其中M可以是在2和16之间的任何整数值。编码的输入可以是M进制符号,并且编码上下文可以包括由累积分布函数(cumulative distribution function, CDF)表示的一组M个概率。可以在编码/解析每个语法元素之后更新概率。累积分布函数可以是如下等式7所示的M个15位整数的阵列:

$$C = [c_0, c_1, \dots, c_{(M-2)}, 2^{15}] \text{ (等式7)}$$

其中 $c_n/32768$ 是符号小于或等于n的概率。

[0089] 可以使用以下等式8执行概率更新:

$$\begin{cases} c_m = c_m \cdot (1 - \alpha) & m \in [0, \text{符号}) \\ c_m = c_m + \alpha \cdot (1 - c_m) & m \in [\text{符号}, M - 1) \end{cases}, \quad (\text{等式8})$$

其中 α 是基于符号已被解码的次数(例如,最多32)而适应的概率更新速率,并且m是CDF中的元素的索引。

[0090] α 的这种自适应允许在编码/解析语法元素开始时更快的概率更新。M进制算术编码过程可以遵循常规算术编码引擎设计。然而,仅15位概率值中的最高有效9位可以被输入到算术编码器/解码器。与符号相关联的概率更新速率 α 可以在解析码流时基于相关联的符号的出现次数来计算,并且 α 的值可以在帧或图块的开始处使用以下等式9来重置:

$$\alpha = \frac{1}{2^{3 + (\text{count} > 15) + (\text{count} > 32) + \min(\log_2(M), 2)}} \quad (\text{等式9})$$

[0091] 根据等式9,概率更新速率在开始时具有较大值,并且然后在发生32次出现之后饱和。

[0092] [多假设算术编码]

[0093] 可以提供用于编码M进制符号的多假设概率模型。例如,一种方法可以包括对与多假设估计有关的AV1算术编码引擎的修改。例如,AV1使用用于概率更新的数据自适应模型,其中语法元素的出现次数越少,更新速率越高,观察次数越多,更新速率就越低。然而,引擎仅使用单个概率模型。一种多假设估计,其中每个语法元素维持两个或更多个不同更新速率的概率表,可以带来附加的压缩效率。因此,多假设概率模型可以用以下等式10至11中所示的两个更新速率来实施:

$$\alpha_1 = \frac{1}{2^{\alpha_1 + (\text{count} > 15) + (\text{count} > 31) + \min(\log_2(M), 2)}} \quad (\text{等式 } 10)$$

$$\alpha_2 = \frac{1}{2^{\alpha_2 + (\text{count} > 7) + (\text{count} > 15) + \min(\log_2(M), 2)}} \quad (\text{等式 } 11)$$

其中 α_2 以较快的更新进行建模,而 α_1 以较慢的更新进行建模。最终概率模型可以被计算为假设的线性组合。根据实施例,可以使用两个假设的平均值。为了适应跨帧/图块的变化符号统计,可以在帧/图块级处发信号通知 α_1 、 α_2 。

[0094] [亮度帧内模式编码]

[0095] 对于亮度帧内已编码块,所有可用的帧内预测模式可以被重新排序并且在发信号通知之前插入到模式列表中,其中两个相邻块的帧内预测模式可以涉及该重新排序过程。两个相邻块的位置如图6所示。具体地,图6图示了当前块520,以及作为当前块520的相邻块的上方块522和左侧块524。

[0096] 特别地,该重新排序过程可以在以下三个步骤中定义:(1)总是首先将DC、SMOOTH、SMOOTH_V、SMOOTH_H、PAETH模式置于模式列表中;(2)对于相邻块中的每个方向帧内预测模式,通过将偏移[0, -1, +1, -2, +2, -3, +3, -4, +4]添加到方向模式来导出9个模式,并且如果还没有添加它们,则将它们添加到模式列表中;以及(3)如果模式列表未满,则将预定义的默认模式插入到模式列表中。

[0097] 在以上重新排序过程之后,可以基于模式索引将帧内预测模式分成五个模式集合。每个模式集合中的模式数量可以是5、8、16、16和16。下面的表1图示了重新排序的模式索引和相关联的模式集合索引之间的映射。

表1:模式索引和模式集合索引之间的映射

重新排序的模式索引	模式集合索引
0 ~ 4	0
5 ~ 12	1
13 ~ 28	2
29 ~ 44	3
45 ~ 60	4

[0098] 为了发信号通知帧内预测模式,可以发信号通知第一语法以指示帧内预测模式的模式集合索引,并且可以发信号通知第二语法以指示模式集合中的模式索引。

[0099] 根据实施例,只有当AV1中的块大小等于或大于 8×8 时才允许增量角度,而对于AI音乐创造力(AI Music Creativity, AIMC)中的所有块大小都可以允许增量角度。根据实施例,对于垂直和水平模式,可以总是针对零参考行启用基于偏移的帧内预测修正(offset-based intra prediction refinement, ORIP)并且针对非零参考行禁用基于偏移的帧内预测修正。

[0100] [色度帧内模式编码]

[0101] 由于色度分量中的纹理通常比亮度分量更平滑,因此对于每个色度块,可以自适应地选择14个帧内预测模式并且仅将其包括在一个帧内模式集合中。

[0102] 具体地,以下方面可以包括在实施例中:

[0103] (1)用于发信号通知色度模式编码的上下文可以取决于同位的亮度模式的模式集

合索引是否为零。

[0104] (2) 色度标称模式可以基于同位的亮度帧内预测模式重新排序。可以首先将从亮度到色度(CfL)模式添加到色度帧内模式列表中。如果模式列表是方向帧内预测模式,则可以将同位的亮度帧内预测模式的标称模式添加到模式列表中。根据以下顺序[DC、SMOOTH、SMOOTH_V、SMOOTH_H、PAETH、VERTICAL、HORIZONTAL、D45_PRED、D67_PRED、D113_PRED、D135_PRED、D157_PRED、D203_PRED],可以将剩余的12个标称模式添加到模式列表中。

[0105] (3) 对于色度分量,可以不发信号通知增量角度,而是根据同位的亮度模式导出增量角度。具体地,如果色度帧内预测模式是方向模式并且等于同位的亮度块的帧内预测模式,则色度块的增量角度可以被设置为等于同位的亮度块的增量角度。否则,色度块的增量角度可以设置为零。

[0106] [根据亮度预测的色度]

[0107] 对于色度分量,除了56个方向模式和5个非方向模式之外,还可以提供CfL。CfL是仅色度的帧内预测模式,其将色度像素建模为重合的已重建亮度像素的线性函数。CfL预测可以表示为如下等式12:

$$\text{CfL}(\alpha) = \alpha \times L_{AC} + DC \quad (\text{等式12})$$

其中, L_{AC} 表示亮度分量的AC贡献, α 表示线性模型的缩放参数,并且DC表示色度分量的DC贡献。

[0108] 图7是CfL预测过程530的示意图。参考图7和等式12,将已重建亮度像素子采样为色度分辨率,然后减去平均值以形成AC贡献。为了从AC贡献中近似估算出色度AC分量,不需要解码器如在一些背景技术中那样计算缩放参数,AV1CfL可以基于原始色度像素来确定缩放参数 α ,并且在码流中发信号通知缩放参数。这降低了解码器复杂度并且产生更精确的预测。色度分量的DC贡献可以使用帧内DC模式来计算,该帧内DC模式对于大多数色度内容是足够的并且具有成熟的快速实现方式。

[0109] 当选择CfL模式时,可以首先发信号通知U和V分量的缩放因子的联合标志。一个缩放因子的标志可以是负的、零或正的。另外,在CfL模式中可能不允许(零,零)的组合,因为它导致“DC”预测。因此,两个缩放因子的标志的组合总共可以具有 $8(3 \times 3 - 1 = 8)$ 种组合。因此,联合标志可能需要8值符号。可以仅采用一个上下文来发信号通知联合标志。

[0110] 关于用于缩放参数的幅度的信令,16值符号可以用于以1/8的步长表示范围从0至2的值。16值符号充分利用了多符号熵编码器的能力。发信号通知缩放参数的上下文取决于联合标志的值。

[0111] [AV1中的运动矢量差编码]

[0112] AV1允许1/8像素运动矢量精度(或准确度),并且以下语法元素可以用于发信号通知参考帧列表0或参考帧列表1中的运动矢量差。

[0113] 语法元素mv_joint可以指定运动矢量差的哪些分量是非零的。例如,值0可以指示沿水平或垂直方向没有非零MVD;值1可以指示仅沿水平方向存在非零MVD;值2可以指示仅沿着垂直方向存在非零MVD;值3可以指示沿水平和垂直方向均存在非零MVD。

[0114] 语法元素mv_sign可以指定运动矢量差是正的还是负的。

[0115] 语法元素mv_class可以指定运动矢量差的级别。如下表2所示,较高的级别可能意味着运动矢量差具有较大的幅度。

表2:运动矢量差的幅度级别

MV 级别	MVD 的幅度
MV_CLASS_0	(0, 2]
MV_CLASS_1	(2, 4]
MV_CLASS_2	(4, 8]
MV_CLASS_3	(8, 16]
MV_CLASS_4	(16, 32]
MV_CLASS_5	(32, 64]
MV_CLASS_6	(64, 128]
MV_CLASS_7	(128, 256]
MV_CLASS_8	(256, 512]
MV_CLASS_9	(512, 1024]
MV_CLASS_10	(1024, 2048]

[0116] 语法元素mv_bit可以指定每个MV级别的运动矢量差和起始幅度之间的偏移的整数部分。

[0117] 语法元素mv_fr可以指定运动矢量差的前2个分数位。

[0118] 语法元素mv_hp可以指定运动矢量差的第三分数位。

[0119] [AV1中的多符号算术编码的问题]

[0120] 关于AV1中的多符号算术编码,AV1中使用的M进制算术编码引擎可以支持具有M个元素的字母表大小的语法元素,其中M可以为2和16之间的任何整数值。在硬件中,算术编码引擎应能够处理最坏情况(字母表大小为16),导致芯片面积增加和吞吐量降低。

[0121] [示例实施例]

[0122] 本公开的实施例可以单独使用或以任何顺序组合使用。此外,本公开的实施例中的每一个(例如,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)来实施。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。

[0123] 本公开的实施例可以通过支持减小的字母表大小N(N进制算术编码引擎)来降低算术编码引擎的复杂度,其中 $N \leq M$ 。为此,可能要求所有熵编码的语法元素的字母表大小 $<$

=N。

[0124] 本公开的实施例可以将最大字母表大小设置为小于硬件强加的约束。

[0125] 本公开的实施例可以将熵编码的语法元素的字母表大小限制为小于或等于N。

[0126] 根据实施例,N可以是2和16之间的任何整数值,诸如2、4或8。

[0127] 根据实施例,在AV1中定义的语法元素mv_class可以具有字母表大小为11。根据实施例,mv_class可以指定运动矢量差的级别。如果N小于16(例如,4或8),则可以定义新的语法元素mv_class_set和mv_class_idx来替换mv_class,其中mv_class_set和mv_class_idx的字母表大小可以小于或等于N。这可以通过将mv_class的符号分组为多个集合,并且为一个集合中的各符号提供一个索引(例如,mv_class_set)来实施。根据实施例,可以提供/发信号通知另一索引(例如,mv_class_idx)以指示所选集合中的哪一候选正用于编码和解码。根据实施例,编码器可以执行上述功能(例如,基于N的值定义语法元素以替换mv_class、将mv_class的符号分组为多个集合,以及为集合中的每个符号提供索引),熵编码语法元素以获得已编码码流,并且将已编码码流发送到解码器。根据实施例,解码器可以接收由编码器发送的已编码码流,从已编码码流获得语法元素(及其符号)(例如,通过熵解码),并且基于语法元素解码已编码码流的视频数据。

[0128] 根据一个或多个实施例,mv_class_set的字母表大小被设置(例如,由编码器)为2,并且各个mv_class_set中分组的MV级别的数量分别为3和8。

[0129] 根据一个或多个实施例,mv_class_set的字母表大小被设置(例如,由编码器)为3,并且每个mv_class_set中分组的MV级别的数量分别为3、4和4。

[0130] 根据实施例,在AV1中定义的语法元素cfl_alpha_u和cfl_alpha_v具有字母表大小为16。语法元素cfl_alpha_u可以包含针对U分量的alpha减1的绝对值,并且语法元素cfl_alpha_v可以包含针对V分量的alpha减1的绝对值。根据实施例,“alpha”可以是用于CfL模式的缩放参数(例如,等式12中的 α)。如果N小于16,例如4或8,则可以定义新的语法元素cfl_alpha_u_set和cfl_alpha_u_idx来替换cfl_alpha_u,并且可以定义新的语法元素cfl_alpha_v_set和cfl_alpha_v_idx来替换cfl_alpha_v,其中新语法元素中的每一个的字母表大小小于或等于N。根据实施例,编码器可以执行上述功能(例如,定义语法元素以基于N的值来替换cfl_alpha_u和cfl_alpha_v),并且将已编码码流发送到解码器。根据实施例,解码器可以接收由编码器发送的已编码码流,从已编码码流获得语法元素(及其符号)(例如,通过熵解码),并且基于语法元素解码已编码码流的视频数据。

[0131] 根据一个或多个实施例,可以将候选CfL alpha值分组为不同集合(即,CfL alpha值集合),可以发信号通知集合索引(例如,cfl_alpha_u_set、cfl_alpha_v_set)以指示所选CfL alpha值属于哪一集合,并且可以发信号通知另一索引(例如,cfl_alpha_u_idx和cfl_alpha_v_idx)以指示所选集合中的哪一候选正用于编码和解码。根据实施例,编码器可以执行上述功能(例如,定义语法元素以基于N的值来替换cfl_alpha_u和cfl_alpha_v、将候选CfL alpha值分组为多个集合),对语法元素进行熵编码以获得已编码码流,并且将已编码码流发送到解码器。根据实施例,解码器可以接收由编码器发送的已编码码流,从已编码码流获得语法元素(及其符号)(例如,通过熵解码),并且基于语法元素解码已编码码流的视频数据。

[0132] 根据实施例,预定义CfL alpha值集合中的一个或多个中的候选CfL alpha值。

[0133] 根据实施例, (例如, 由编码器或解码器) 通过已编码信息 (包括但不限于亮度和色度的相邻已重建样本值、相邻块中使用的alpha值以及块大小/形状) 来确定一个或多个CfL alpha值集合中的候选CfL alpha值。在一个示例中, 一个或多个CfL alpha值集合定义了一组最可能的CfL alpha值。

[0134] 根据实施例, (例如, 由编码器或解码器) 基于相关联的alpha值来确定一个或多个CfL alpha值集合中的候选CfL alpha值。例如, 当存在两个集合时, 一个集合可以被设置为包括小于阈值的所有CfL alpha值, 并且其余的CfL alpha值可以被包括在另一集合中。根据实施例, 编码器和/或解码器可以执行上述功能。

[0135] 根据实施例, 对集合索引 (例如, cfl_alpha_u_set、cfl_alpha_v_set) 进行上下文编码。

[0136] 根据实施例, 指定所选集合中的哪个候选正被使用的索引 (例如, cfl_alpha_u_idx和cfl_alpha_v_idx) 在没有任何上下文的情况下被编码 (即, 旁路编码或以固定概率编码)。

[0137] 根据实施例, 可以基于已编码信息, 诸如先前的CfL模式集合索引, 或相邻样本的亮度已重建像素和色度已重建像素, 来决定集合索引。在这种情况下, 可以隐式地发信号通知集合索引。根据实施例, 编码器和/或解码器可以确定集合索引。

[0138] 根据一个或多个实施例, cfl_alpha_u_set (和cfl_alpha_v_set) 的字母表大小设置 (例如, 由编码器设置) 为2 (或4), 并且每个集合中的候选CfL alpha值的数量为8或4。

[0139] 根据实施例, 与亮度帧内模式编码相关联的语法元素mode_set_index和mode_idx具有 $\leq N$ 的字母表大小。

[0140] 根据一个或多个实施例, mode_set_index的值被设置 (例如, 由编码器设置) 为8。第一组中的模式数量被设置 (例如, 由编码器设置) 为5, 并且其它集合中的模式数量被设置 (例如, 由编码器设置) 为8。

[0141] 根据实施例, 与色度帧内模式编码相关联的语法元素uv_mode_set_index和uv_mode_idx具有 $\leq N$ 的字母表大小。

[0142] 根据一个或多个实施例, 将uv_mode_set_index的值设置 (例如, 由编码器设置) 为2, 并且将第一集合中的模式数量设置 (例如, 由编码器设置) 为6, 并且将第二集合中的模式设置 (例如, 由编码器设置) 为8。

[0143] 根据实施例, 当发信号通知楔形波图案索引 (即, wedge_idx) 时, 可以将所有可能的楔形波图案分成不同集合, 并且每个集合中的候选楔形波图案的数量可能小于或等于N。

[0144] 根据一个或多个实施例, 可以发信号通知第一语法元素wedgelet_set来指示哪个楔形波图案的集合 (即, 楔形波图案集合) 是来自所选楔形波图案。

[0145] 根据一个或多个实施例, 可以发信号通知第二语法wedgelet_idx, 以指示所选楔形波图案集合中的哪个候选被选择用于编码和解码。

[0146] 根据一个或多个实施例, 基于已编码信息 (包括但不限于相邻已重建样本值、相邻块中使用的楔形波图案, 以及块大小/形状) 来生成 (例如, 由编码器或解码器) 一个或多个楔形波图案集合。

[0147] 根据实施例, 一个或多个楔形波图案集合定义了一组最可能的楔形波图案。

[0148] 根据实施例,当残差块中的样本数量大于阈值T时,则使用多个语法元素来发信号通知eob值。语法元素eob可以是指示块的末端的索引的变量。根据实施例,编码器可以确定残差块中的样本数量是否大于阈值T,并且基于确定残差块中的样本数量大于阈值T,通过使用多个语法元素发信号通知已编码码流中的eob值。

[0149] 根据一个或多个实施例,T的值等于 $2^{(N-1)}$ 。

[0150] 根据一个或多个实施例,eob值范围首先被分类为不同的组,并且首先发信号通知组索引,即eob_group_idx,然后进一步(例如,在已编码码流中)发信号通知所选组的所选条目以指示eob值。根据实施例,编码器可以执行上述功能。根据实施例,解码器可以接收由编码器发送的已编码码流,从已编码码流获得语法元素(例如,通过熵解码),并且基于语法元素解码已编码码流的视频数据。

[0151] 根据实施例,可以提供至少一个处理器和存储计算机程序指令(即计算机代码)的存储器。计算机程序指令在由至少一个处理器执行时可以实施编码器或解码器并且可以执行本公开中所描述的任何数量的功能,包括由编码器或解码器执行的功能。

[0152] 例如,参考图8,至少一个处理器可以实施编码器702或解码器704。计算机程序指令可以包括例如编码(或解码)代码710,其被配置成使至少一个处理器对根据本公开的实施例接收的至少一个图片进行编码(或解码)。编码(或解码)代码710可以包括例如字母表确定代码720、分组代码730、定义代码740、算术编码(或解码)代码750、确定代码760和发送代码790。虽然本文描述了使某些功能被执行的示例代码,但是应当理解,本公开的实施例可以包括用于执行本公开中描述的任何其它功能的其它代码。

[0153] 根据本公开的实施例,字母表确定代码720可以被配置成使至少一个处理器确定用于算术编码(和/或解码)的最大字母表大小。例如,字母表确定代码720可以被配置成使至少一个处理器确定用于算术编码的第一最大字母表大小,所述第一最大字母表大小是基于硬件约束确定的;以及基于该第一最大字母表大小确定小于第一最大字母表大小的第二最大字母表大小。

[0154] 分组代码730可以被配置成使至少一个处理器将与视频编码相关的符号分组为集合。根据实施例,可以基于确定最大字母表大小小于或等于预定值来进行分组。例如,符号可以是运动矢量差的类别、用于CfL模式的缩放参数的绝对值的指示、亮度帧内模式编码的模式、色度帧内模式编码的模式、楔形波图案或eob值。根据实施例,分组可以对语法元素(例如,mv_class、cfl_alpha_u和cfl_alpha_v)的符号进行分组以替换语法元素。

[0155] 根据本公开的实施例,定义代码740可以被配置成使至少一个处理器定义与符号相关的语法元素。根据实施例,可以基于确定最大字母表大小小于或等于预定值来进行定义。根据实施例,被定义的新语法元素可以替换该语法元素。

[0156] 根据本公开的实施例,算术编码(或解码)代码720可以被配置成使至少一个处理器对语法元素进行算术编码以获得已编码视频流或对语法元素进行算术解码。例如,可以基于所确定的第二最大字母表大小对语法元素进行编码(或解码),其中以小于或等于所确定的第二最大字母表大小的字母表大小对多个语法元素中的每一个进行熵编码(或解码)。

[0157] 根据本公开的实施例,确定代码760可以被配置成使至少一个处理器基于信息(例如,已编码信息)来确定集合、一个或多个集合中的符号和/或索引。

[0158] 根据本公开的实施例,当提供编码器702时,计算机程序指令可以进一步包括例如

发送代码790,该发送代码790可以被配置成使至少一个处理器将包括已编码图片的码流发送到解码器(例如,解码器704)。根据本公开的实施例,编码器702可以在码流中提供信息(例如,语法元素及其符号),诸如本公开中所描述的信息。

[0159] 以上所描述的技术可以被实施为使用计算机可读指令并且物理地存储在一个或多个计算机可读介质中的计算机软件。例如,图9示出了适于实施所公开的主题的实施例的计算机系统900。

[0160] 计算机软件可以使用任何合适的机器代码或计算机语言进行编码,机器代码或计算机语言可以通过汇编、编译、链接或类似机制来创建包括可以由计算机中央处理单元(CPU)、图形处理单元(GPU)等直接或通过解释、微代码执行等执行的指令的代码。

[0161] 指令可以在各种类型的计算机或其部件上执行,包括例如个人计算机、平板计算机、服务器、智能电话、游戏设备、物联网设备等。

[0162] 图9所示的用于计算机系统900的部件本质上是示例性的,并且不旨在对实施本公开的实施例的计算机软件的使用范围或功能性提出任何限制。部件的配置也不应被解释为对计算机系统900的示例性实施例中所图示的任何一个部件或其组合具有任何依赖性要求。

[0163] 计算机系统900可以包括某些人机接口输入设备。此类人机接口输入设备可以响应于一个或多个人类用户通过例如触觉输入(诸如:击键、滑动、数据手套移动)、音频输入(诸如:语音、拍打)、视觉输入(诸如:手势)、嗅觉输入(未示出)的输入。人机接口设备还可以用于捕获不一定与人的有意识输入直接相关的某些介质,诸如音频(诸如:语音、音乐、环境声音)、图像(诸如:扫描图像、从静止图像照相机获取的摄影图像)、视频(诸如二维视频、包括立体视频的三维视频)。

[0164] 输入人机接口设备可以包括以下中的一个或多个(每种仅描绘了一个):键盘901、鼠标902、触控板903、触摸屏910、数据手套或操纵杆905、麦克风906、扫描仪907、照相机908。

[0165] 计算机系统900还可以包括某些人机接口输出设备。此类人机接口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觉。此类人机接口输出设备可以包括触觉输出设备(例如触摸屏910、数据手套或操纵杆905的触觉反馈,但也可以有不用作输入设备的触觉反馈设备)。例如,这些设备可以是音频输出设备(诸如:扬声器909、耳机(未描绘))、视觉输出设备(诸如屏幕910,包括CRT屏幕、LCD屏幕、等离子屏幕、OLED屏幕,其各自具有或不具有触摸屏输入能力,各自具有或不具有触觉反馈能力—其中一些能够通过诸如立体输出的方式输出二维视觉输出或多于三维输出;虚拟现实眼镜(未描绘)、全息显示器和烟雾罐(未描绘)),以及打印机(未描绘)。

[0166] 计算机系统900还可以包括人类可访问的存储设备及其相关联的介质,诸如包括具有CD/DVD等介质921的CD/DVD ROM/RW 920、拇指驱动器922、可移动硬盘驱动器或固态驱动器923、诸如磁带和软盘(未描绘)的传统磁性介质、诸如安全软件狗(未描绘)的基于专用ROM/ASIC/PLD的设备等。

[0167] 本领域技术人员还应当理解,结合当前公开的主题使用的术语“计算机可读介质”不包括传输介质、载波或其它易失性信号。

[0168] 计算机系统900还可以包括到一个或多个通信网络的接口。网络例如可以是无线

的、有线的、光学的。网络还可以是本地的、广域的、城域的、车辆的和工业的、实时的、延迟容许型的等等。网络的示例包括诸如以太网的局域网、无线LAN、包括GSM、3G、4G、5G、LTE等的蜂窝网络、包括有线TV、卫星TV和地面广播TV的TV有线或无线广域数字网络、包括CAN总线的车辆的和工业的网络等。某些网络通常需要附接到某些通用数据端口或外围总线949的外部网络接口适配器(诸如例如,计算机系统900的USB端口);其它网络通常通过附接到如下所描述的系统总线而集成到计算机系统900的核心中(例如以太网接口到PC计算机系统中或蜂窝网络接口到智能电话计算机系统中)。使用这些网络中的任何一个,计算机系统900可以与其它实体进行通信。此类通信可以是单向的仅接收(例如,广播TV)、单向的仅发送(例如,CANbus到某些CANbus设备),或双向的,例如到使用局域数字网络或广域数字网络的其它计算机系统。此类通信可以包括到云计算环境的通信955。可以在如上所描述的那些网络和网络接口中的每一个上使用某些协议和协议栈。

[0169] 上述人机接口设备、人类可访问的存储设备和网络接口954可以附接到计算机系统900的核心940。

[0170] 核心940可以包括一个或多个中央处理单元(CPU)941、图形处理单元(GPU)942、现场可编程门阵列(FPGA)943形式的专用可编程处理单元、用于某些任务的硬件加速器944等。这些设备连同只读存储器(ROM)945、随机存取存储器946、诸如内部非用户可访问硬盘驱动器、SSD等的内部大容量存储器947可以通过系统总线948连接。在一些计算机系统中,系统总线948可以以一个或多个物理插头的形式访问,以使得能够通过附加CPU、GPU等进行扩展。外围设备可以直接或通过外围总线949附接到核心的系统总线948。外围总线的体系结构包括PCI、USB等。图形适配器950可以包括在核心940。

[0171] CPU 941、GPU 942、FPGA 943和加速器944可以执行某些指令,这些指令的组合可以构成上述计算机代码。该计算机代码可以存储在ROM 945或RAM 946中。过渡数据也可以存储在RAM 946中,而永久数据可以存储在例如内部大容量存储器947中。可以通过使用高速缓存存储器来启用对任何存储器设备的快速存储和检索,高速缓存存储器可以与一个或多个CPU 941、GPU 942、大容量存储器947、ROM 945、RAM 946等紧密相关联。

[0172] 计算机可读介质可以在其上具有用于执行各种计算机实施的操作的计算机代码。介质和计算机代码可以是为了本公开的目的而专门设计和构建的那些,或者它们可以是计算机软件领域的技术人员公知和可用的类型。

[0173] 作为非限制性示例,对应计算机系统900的体系结构,特别是核心940可以提供作为一个或多个处理器(包括CPU、GPU、FPGA、加速器等)执行体现在一个或多个有形的计算机可读介质中的软件的结果的功能性。此类计算机可读介质可以是与如上所介绍的用户可访问的大容量存储器相关联的介质,以及具有非易失性性质的核心940的某些存储器,诸如核心内部大容量存储器947或ROM 945。实施本公开的各种实施例的软件可以存储在此类设备中并由核心940执行。根据特定需要,计算机可读介质可以包括一个或多个存储器设备或芯片。软件可以使核心940并且特别是其中的处理器(包括CPU、GPU、FPGA等)执行本文描述的特定过程或特定过程的特定部分,包括定义存储在RAM 946中的数据结构并且根据由软件定义的过程修改此类数据结构。另外或作为替代,计算机系统可以提供作为逻辑硬连线的结果或以其它方式体现在电路(例如:加速器944)中的功能性,其可以代替软件或与软件一起操作以执行本文描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以包括存储用于执行的软件的电路(诸如集成电路(IC))、体现用于执行的逻辑的电路,或两者。本公开包括硬件和软件的任何合适的组合。

[0174] 虽然本公开描述了几个非限定的示例性实施例,但存在变化、排列和各种替代等价物,这些都落在本公开的范围内。因此,本领域的技术人员将能够设计出许多系统和方法,这些系统和方法虽然在本文中并没有明确示出或描述,但是体现了本公开的原理,并且因此在本公开的精神和范围内。

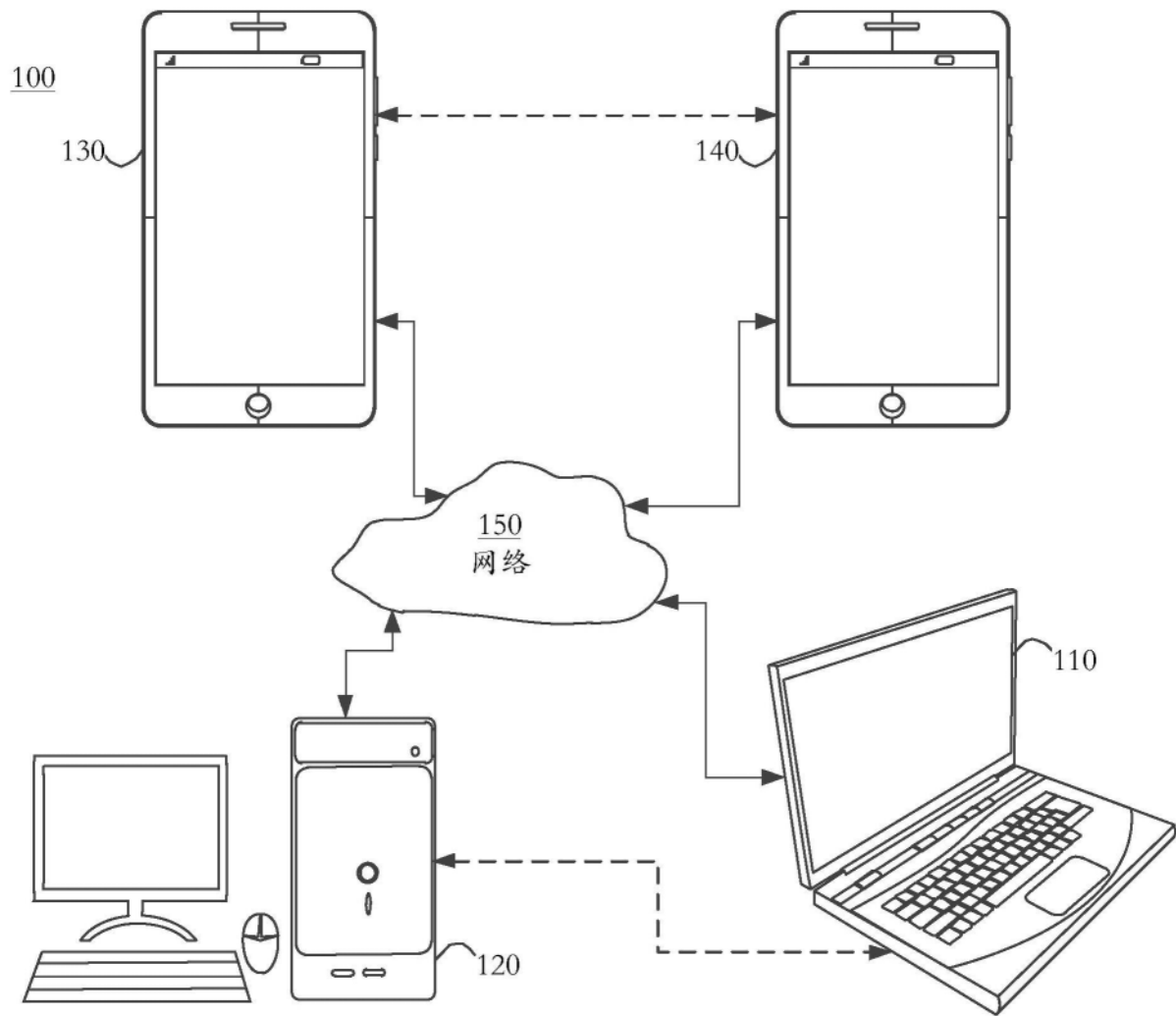


图1

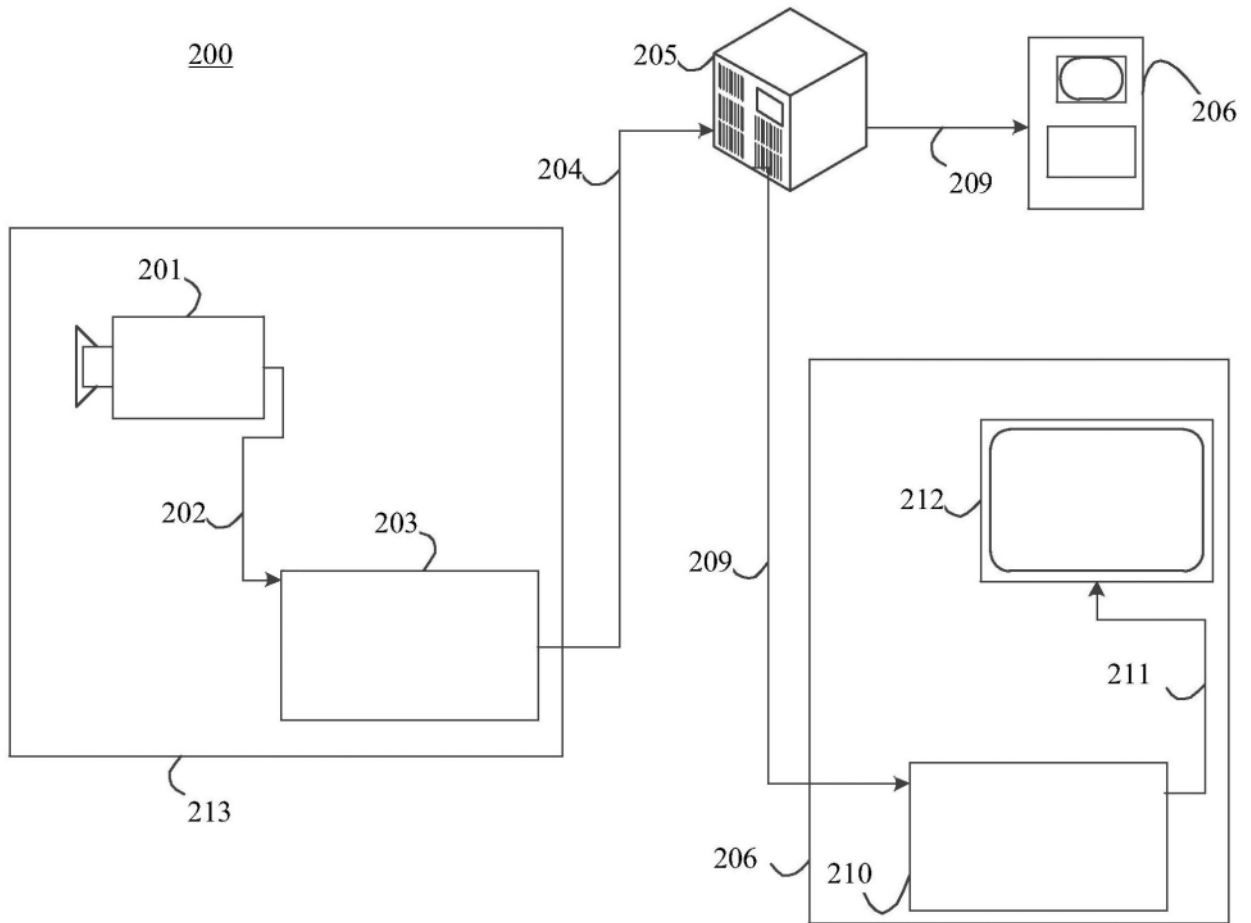


图2

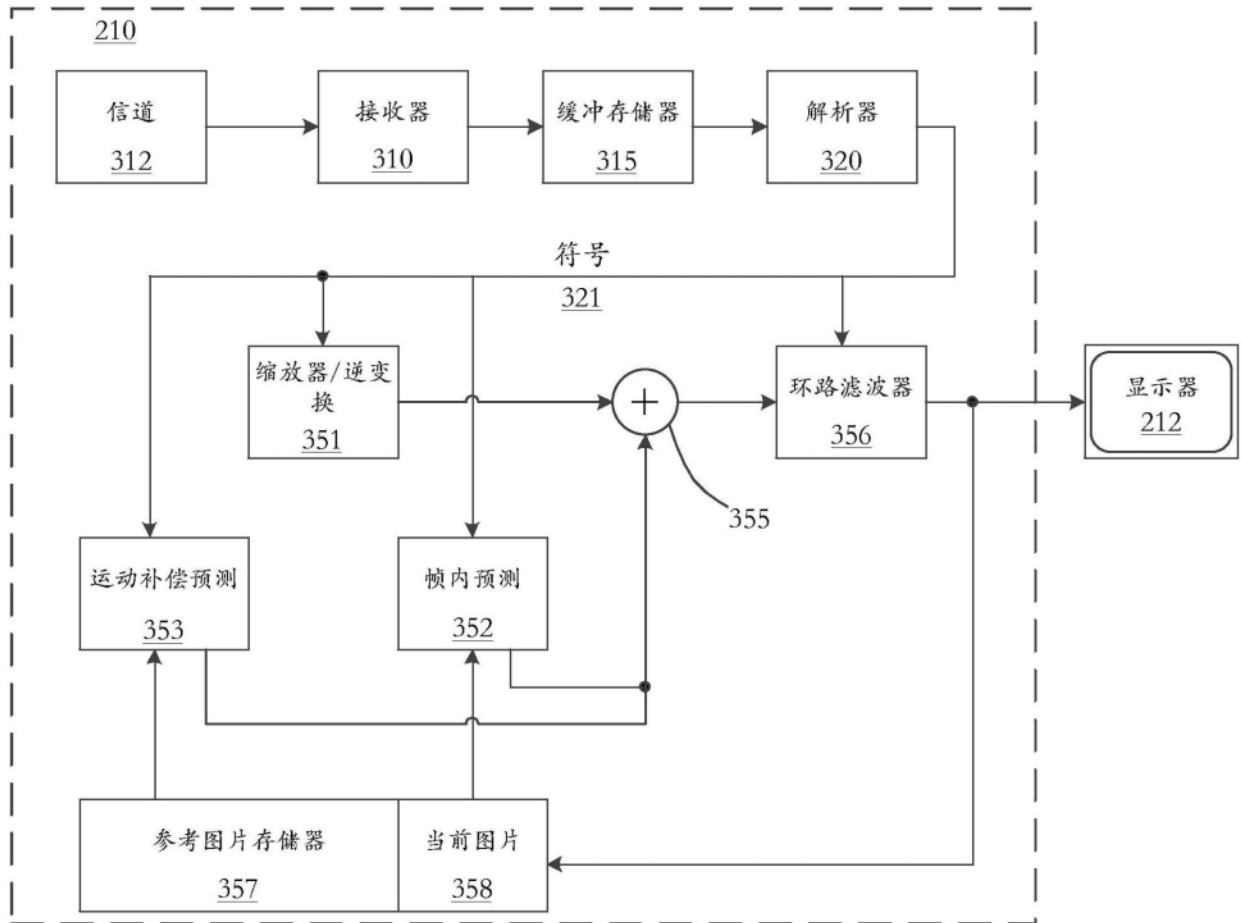


图3

203

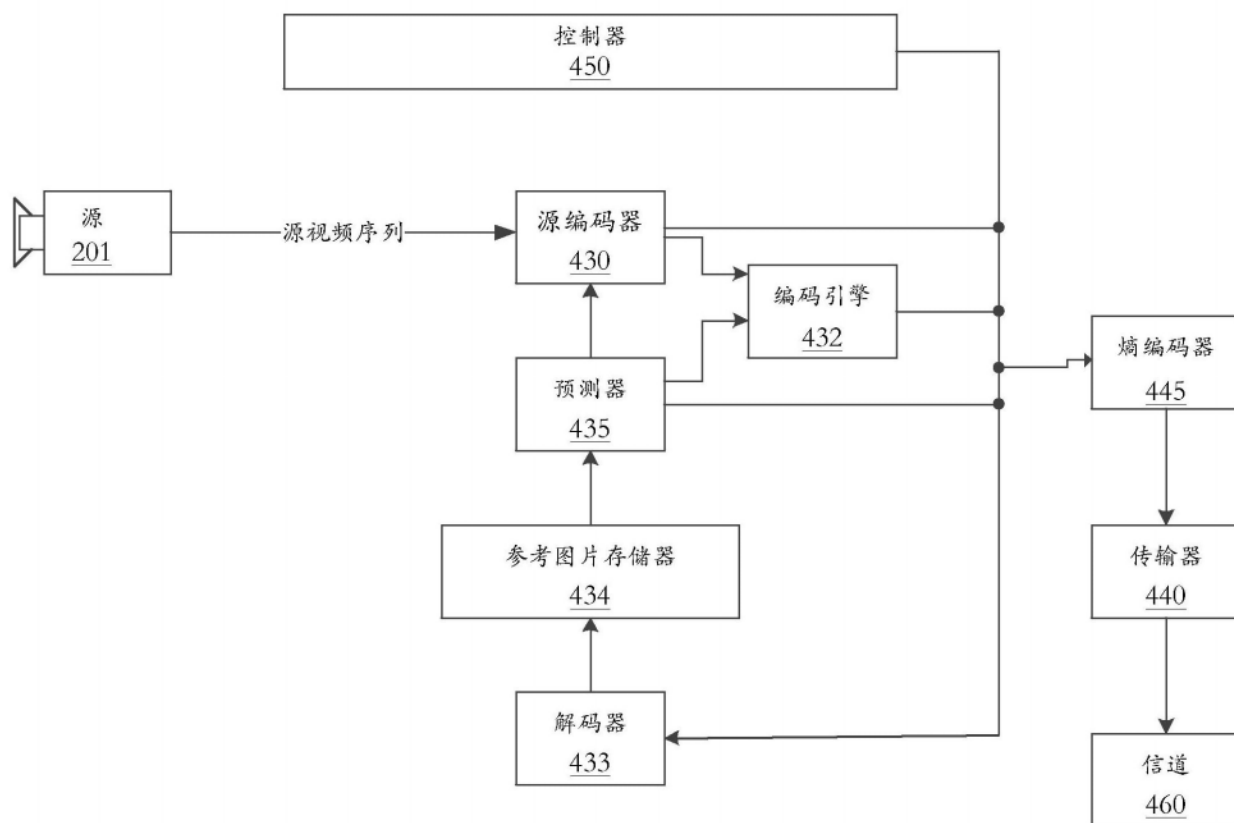


图4

500

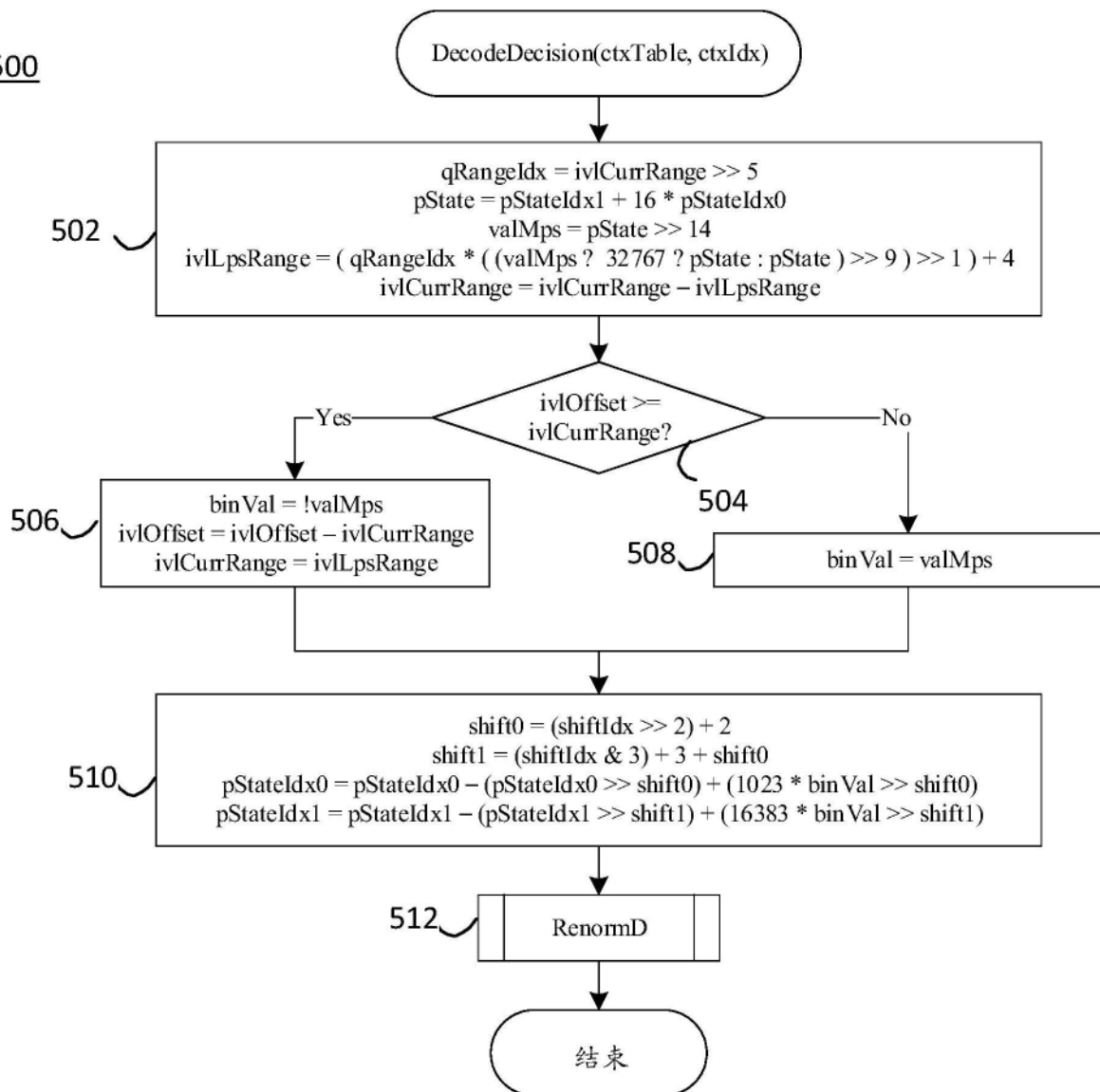


图5

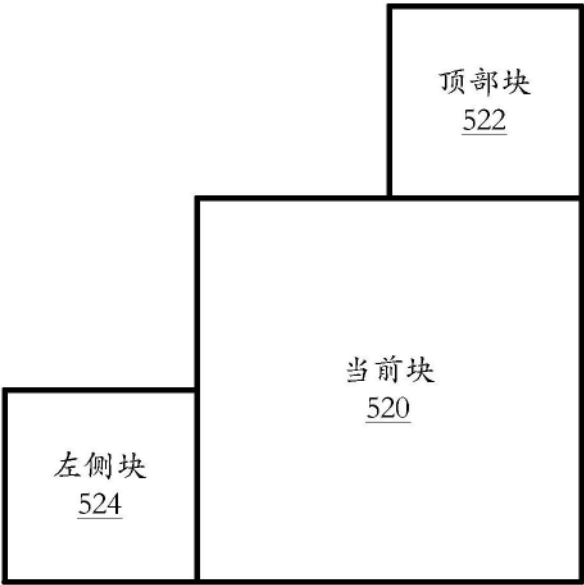


图6

530

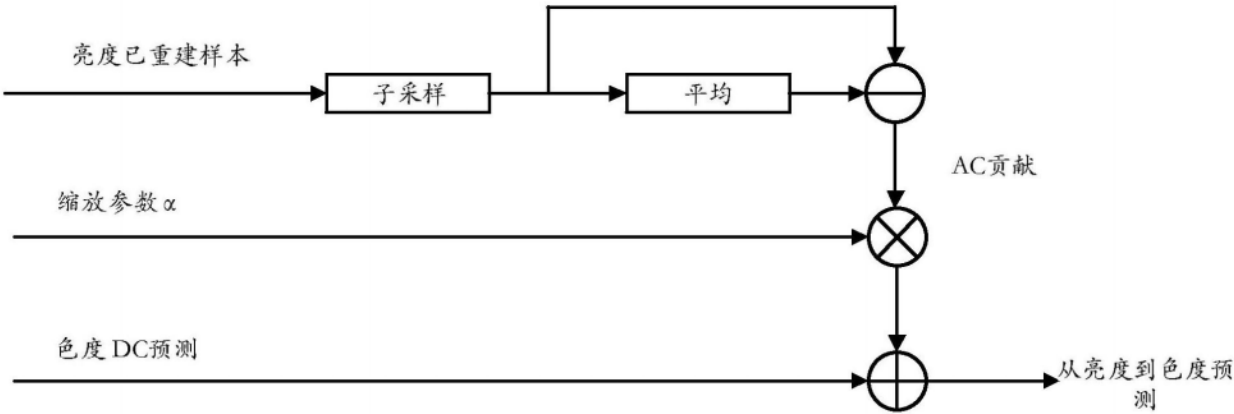


图7

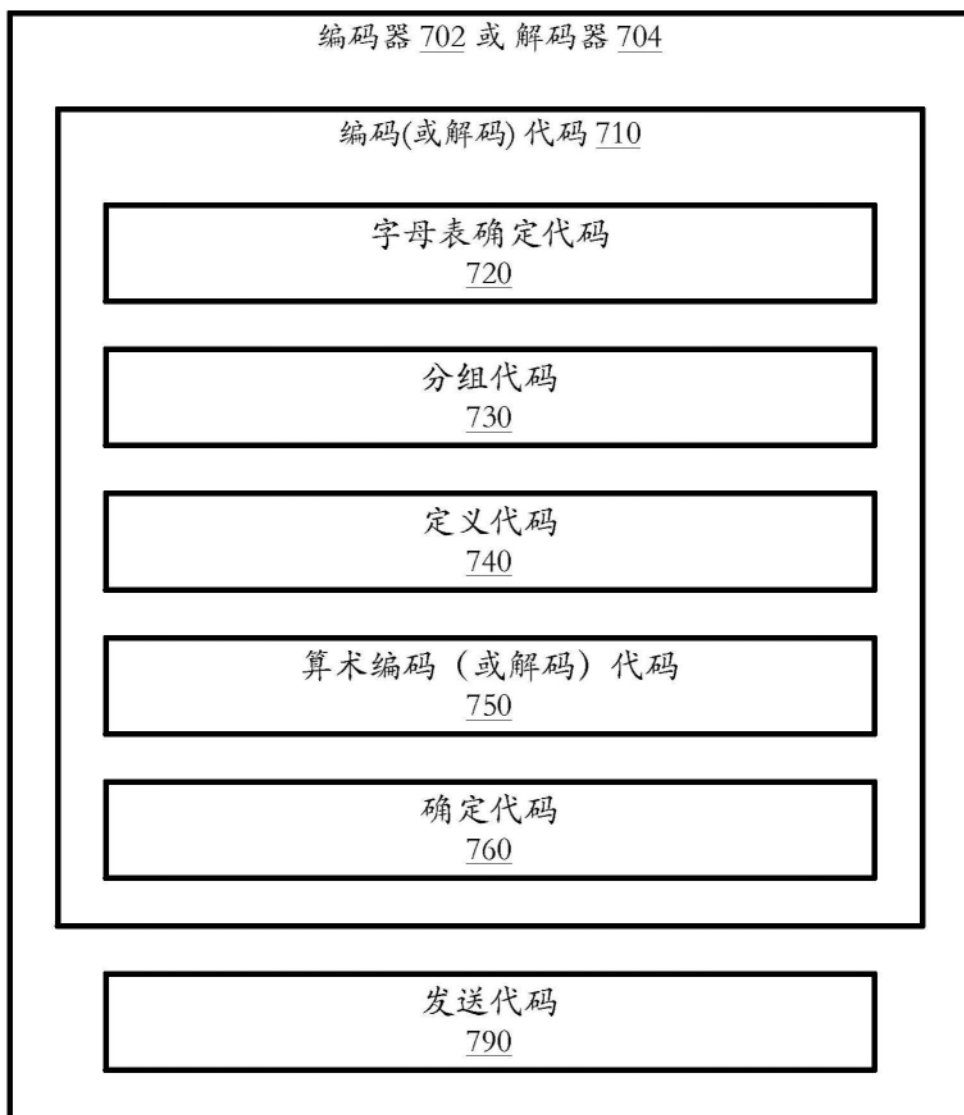


图8

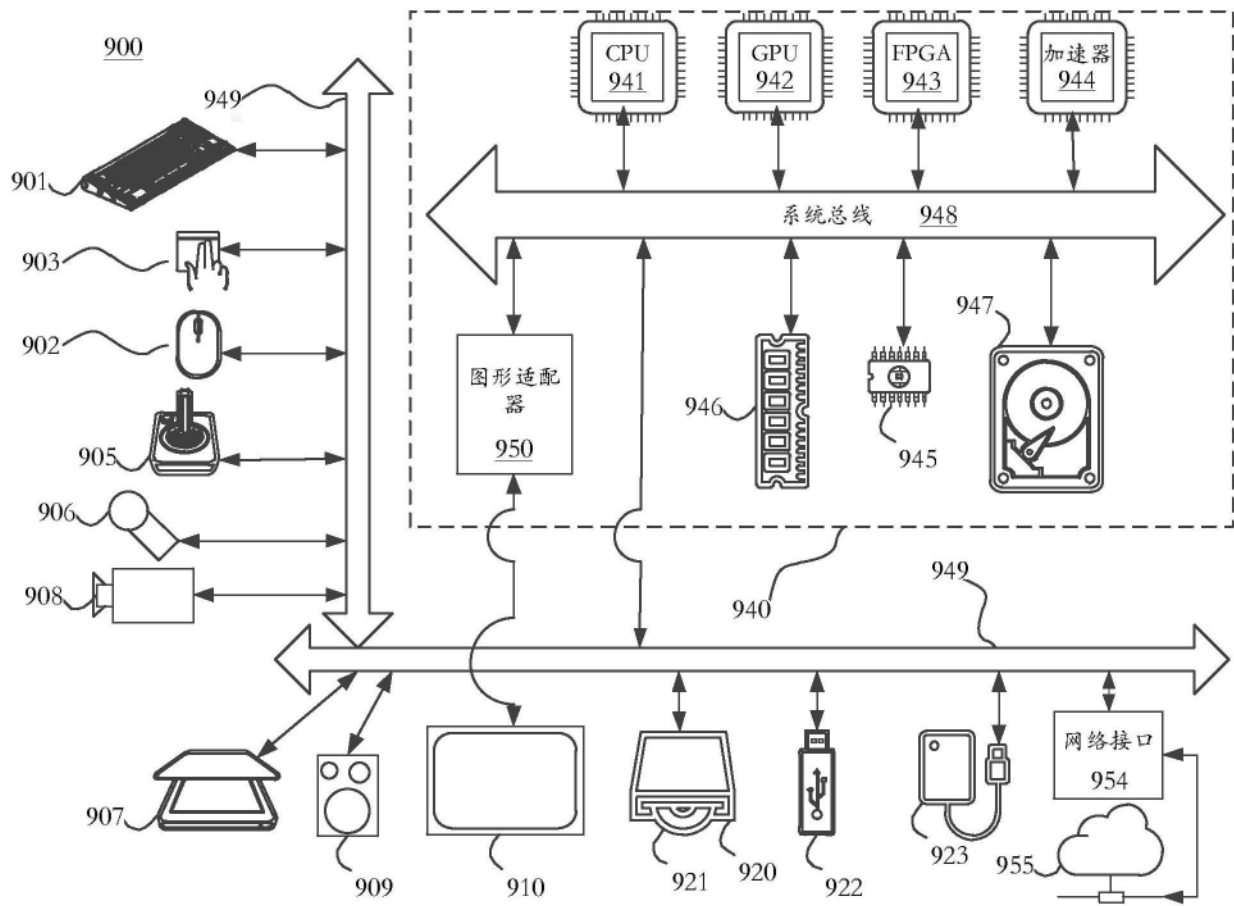


图9