



(12) 发明专利

(10) 授权公告号 CN 113678458 B

(45) 授权公告日 2024.03.22

(21) 申请号 202080025037.3

(22) 申请日 2020.09.10

(65) 同一申请的已公布的文献号
申请公布号 CN 113678458 A

(43) 申请公布日 2021.11.19

(30) 优先权数据
62/903,639 2019.09.20 US
62/905,319 2019.09.24 US
17/010,163 2020.09.02 US

(85) PCT国际申请进入国家阶段日
2021.09.27

(86) PCT国际申请的申请数据
PCT/US2020/050076 2020.09.10

(87) PCT国际申请的公布数据
W02021/055215 EN 2021.03.25

(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 崔秉斗 史蒂芬·文格尔 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 陈美娥 王琦

(51) Int.Cl.
H04N 19/61 (2006.01)
H04N 19/105 (2006.01)

(56) 对比文件
US 2014269912 A1, 2014.09.18
US 2015304666 A1, 2015.10.22
US 2017085917 A1, 2017.03.23
US 2016173889 A1, 2016.06.16
US 2019075323 A1, 2019.03.07
Benjamin Bross等.Versatile Video Coding (Draft 6).《Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11》.2019,正文第7.4节.
Byeongdo Choi等.AHG8/AHG17: On signaling reference picture resampling.《Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11》.2019,1-3页.

审查员 李若云

权利要求书2页 说明书20页 附图8页

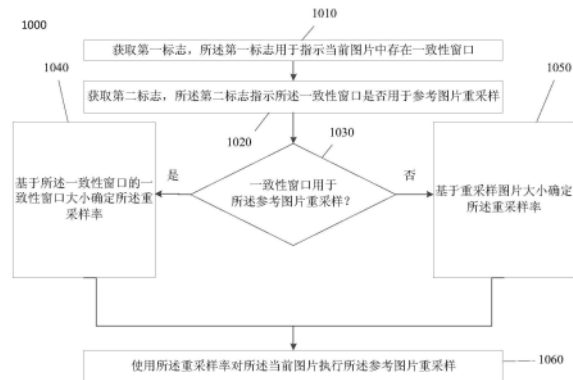
(54) 发明名称

对已编码视频比特流进行解码的方法

(57) 摘要

一种对已编码视频比特流进行解码的方法、设备和计算机可读介质,包括:获取第一标志,所述第一标志指示当前图片中存在一致性窗口;当所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;当所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率;当所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于重采样图片大小确定所述重采样

率;以及使用所述重采样率对所述当前图片执行所述参考图片重采样。



1. 一种对已编码视频比特流进行解码的方法,其特征在于,所述方法包括:
 - 获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口;
 - 当所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;
 - 当所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小,确定所述当前图片和参考图片之间的重采样率;
 - 当所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于距所述当前图片的参考区域的左上亮度样本的至少一个偏移距离,确定所述重采样图片大小,其中所述重采样图片大小在所述已编码视频比特流中指示为所述重采样图片大小的宽度和所述重采样图片大小的高度中的至少一个,在图片参数集中发信号通知所述宽度和所述高度中的所述至少一个,所述宽度和所述高度中的所述至少一个表示为包括在所述宽度和所述高度中的所述至少一个中的亮度样本的数量;
 - 基于所述重采样图片大小确定所述重采样率;以及
 - 使用所述重采样率对所述当前图片执行所述参考图片重采样;
 - 其中,所述第一标志为一致性窗口标志 `conformance_window_flag`,当所述 `conformance_window_flag` 等于0时,推断 `conf_win_left_offset`、`conf_win_right_offset`、`conf_win_top_offset` 和 `conf_win_bottom_offset` 的值等于0。
2. 根据权利要求1所述的方法,其特征在于,在所述图片参数集中发信号通知所述第一标志和所述第二标志。
3. 根据权利要求1所述的方法,其特征在于,在所述图片参数集中发信号通知所述至少一个偏移距离。
4. 一种视频比特流的编码方法,其特征在于,所述方法包括:
 - 编码视频比特流;
 - 发信号通知第一标志,所述第一标志指示当前图片中存在一致性窗口;
 - 当所述第一标志指示存在所述一致性窗口时,发信号通知第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;
 - 当所述第二标志指示所述一致性窗口用于所述参考图片重采样时,所述一致性窗口的一致性窗口大小被用于确定所述当前图片和参考图片之间的重采样率;
 - 当所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于距所述当前图片的参考区域的左上亮度样本的至少一个偏移距离,确定所述重采样图片大小,其中所述重采样图片大小在所述已编码视频比特流中指示为所述重采样图片大小的宽度和所述重采样图片大小的高度中的至少一个,在图片参数集中发信号通知所述宽度和所述高度中的所述至少一个,所述宽度和所述高度中的所述至少一个表示为包括在所述宽度和所述高度中的所述至少一个中的亮度样本的数量;所述视频比特流中包含的重采样图片大小被用于确定所述重采样率;
 - 其中,所述第一标志包括一致性窗口标志 `conformance_window_flag`,当所述 `conformance_window_flag` 等于0时,推断 `conf_win_left_offset`、`conf_win_right_offset`、`conf_win_top_offset` 和 `conf_win_bottom_offset` 的值等于0。
5. 一种产生视频比特流的编码装置,其特征在于,所述编码装置包括本地解码器,所述

本地解码器执行权利要求1-3任一项所述的解码方法。

6. 一种对已编码视频比特流进行解码的设备,其特征在于,所述设备包括:

至少一个存储器,被配置为存储程序代码;以及

至少一个处理器,被配置为读取所述程序代码,并按照所述程序代码的指令进行操作,以执行权利要求1-3任一项所述的方法。

7. 一种对已编码视频比特流进行解码的装置,其特征在于,所述装置包括:

第一获取模块,被配置为使所述至少一个处理器获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口;

第二获取模块,被配置为使所述至少一个处理器在所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;

第一确定模块,被配置为使所述至少一个处理器在所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率;

第二确定模块,被配置为使所述至少一个处理器在所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于距所述当前图片的参考区域的左上亮度样本的至少一个偏移距离,确定所述重采样图片大小,其中所述重采样图片大小在所述已编码视频比特流中指示为所述重采样图片大小的宽度和所述重采样图片大小的高度中的至少一个,在图片参数集中发信号通知所述宽度和所述高度中的所述至少一个,所述宽度和所述高度中的所述至少一个表示为包括在所述宽度和所述高度中的所述至少一个中的亮度样本的数量;基于重采样图片大小确定所述重采样率;以及

执行模块,被配置为使所述至少一个处理器使用所述重采样率对所述当前图片执行所述参考图片重采样;

其中,所述第一标志包括一致性窗口标志conformance_window_flag,当所述conformance_window_flag等于0时,推断conf_win_left_offset、conf_win_right_offset、conf_win_top_offset和conf_win_bottom_offset的值等于0。

8. 一种非易失性计算机可读介质,其特征在于,用于存储指令,所述指令包括一个或多个指令,当所述一个或多个指令由用于解码已编码视频比特流的设备的一个或多个处理器执行时,使所述一个或多个处理器执行权利要求1-3任一项所述的方法。

对已编码视频比特流进行解码的方法

[0001] 相关申请的交叉引用

[0002] 本申请要求于2019年9月20日提交的、申请号为62/903,639的美国临时申请、于2019年9月24日提交的、申请号为62/905,319的美国临时申请、以及于2020年9月2日提交的、申请号为17/010,163的美国申请的优先权,其全部内容并入本文。

技术领域

[0003] 所公开的主题涉及视频编码和解码,更具体地,涉及具有重采样图片大小指示的参考图片重采样信令。

背景技术

[0004] 使用具有运动补偿的图片间预测来进行视频编码和解码已经众所周知。未压缩的数字视频可以包括一系列图片,每个图片具有例如 1920×1080 亮度样本及相关色度样本的空间维度。所述系列图片可以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。未压缩的视频具有很高的比特率要求。例如,每个样本8比特的1080p60 4:2:0视频(60Hz帧率下 1920×1080 亮度样本分辨率)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。

[0005] 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。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。无损压缩和有损压缩、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。

[0006] 视频编码器和解码器可以利用几大类技术,例如包括运动补偿、变换、量化和熵编码,其中的一些技术将会在下文中介绍。

[0007] 历史上,视频编码器和解码器倾向于在给定图片大小上进行操作,在大多数情况下,该给定图片大小是针对已编码视频序列(CVS)、图片群组(GOP)或类似的多图片时间帧定义的,并保持不变。例如,在MPEG-2中,已知系统设计根据诸如场景的活动等因素改变水平分辨率(从而改变图片大小),但是仅在I图片上改变水平分辨率,因此通常用于GOP。例如,根据ITU-T Rec.H.263附录P,在CVS内使用不同分辨率对参考图片进行重采样是已知的。然而,这里的图片大小没有改变,只有参考图片被重采样,导致可能只有部分图片画布被使用(在下采样的情况下),或者只有部分场景被捕获(在上采样的情况下)。此外,H.263附录Q允许以因子2(在每个维度中)对单个宏块向上或向下重采样。同样,图片大小保持不变。宏块的大小在H.263中是固定的,因此不需要发信号通知。

[0008] 在现代视频编解码中,已预测图片中的图片大小的变化越来越成为主流。例如,VP9允许参考图片重采样并改变整个图片的分辨率。类似地,针对VVC提出了某些建议(包

括,例如Hendry等人的“关于VVC的自适应分辨率改变(On adaptive resolution change (ARC) for VVC)”,联合视频组文档JVET-M0135-v1,2019年1月9日至19日,其整体并入本文),允许将整个参考图片重采样为不同的更高或更低的分辨率。在该文档中,建议在序列参数集中对不同的候选分辨率进行编码,并由图片参数集中的每图片语法元素引用。

发明内容

[0009] 在实施例中,提供了一种使用至少一个处理器对已编码视频比特流进行解码的方法,包括:获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口;当所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;当所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率;当所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于重采样图片大小确定所述重采样率;以及使用所述重采样率对所述当前图片执行所述参考图片重采样。

[0010] 在实施例中,提供了一种对已编码视频比特流进行解码的设备,包括:至少一个存储器,被配置为存储程序代码;以及至少一个处理器,被配置为读取所述程序代码,并按照所述程序代码的指令进行操作,所述程序代码包括:第一获取代码,被配置为使所述至少一个处理器获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口;第二获取代码,被配置为使所述至少一个处理器在所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;第一确定代码,被配置为使所述至少一个处理器在所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率;第二确定代码,被配置为使所述至少一个处理器在所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于重采样图片大小确定所述重采样率;以及执行代码,被配置为使所述至少一个处理器使用所述重采样率对所述当前图片执行所述参考图片重采样。

[0011] 在实施例中,提供了一种非易失性计算机可读介质,用于存储指令,所述指令包括一个或多个指令,当所述一个或多个指令由用于解码已编码视频比特流的设备的一个或多个处理器执行时,使所述一个或多个处理器:获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口;当所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样;当所述第二标志指示所述一致性窗口用于所述参考图片重采样时,基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率;当所述第二标志指示所述一致性窗口不用于所述参考图片重采样时,基于重采样图片大小确定所述重采样率;以及使用所述重采样率对所述当前图片执行所述参考图片重采样。

附图说明

[0012] 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

[0013] 图1是根据实施例的通信系统的简化框图的示意图。

- [0014] 图2是根据实施例的通信系统的简化框图的示意图。
- [0015] 图3是根据实施例的解码器的简化框图的示意图。
- [0016] 图4是根据实施例的编码器的简化框图的示意图。
- [0017] 图5是根据实施例的用于发信号通知ARC参数的选项的示意图。
- [0018] 图6A-6B是根据实施例的语法表的示例的示意图。
- [0019] 图7是根据实施例的在SPS中发信号通知图片大小和一致性窗口的示意图。
- [0020] 图8A-8B是根据实施例的在PPS中发信号通知图片大小和一致性窗口的示意图。
- [0021] 图9是根据实施例的在PPS中发信号通知重采样图片大小的示意图。
- [0022] 图10是根据实施例的对已编码视频比特流进行解码的示例过程的流程图。
- [0023] 图11是根据实施例的计算机系统的示意图。

具体实施方式

[0024] 图1示出了根据本公开实施例的通信系统(100)的简化框图。通信系统(100)可以包括通过网络(150)互联的至少两个终端装置(110,120)。对于单向数据传输,第一终端装置(110)可在本地位置对视频数据进行编码,以通过网络(150)传输到另一终端装置(120)。第二终端装置(120)可从网络(150)接收另一终端装置的已编码视频数据,对已编码视频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等应用中是较常见的。

[0025] 图1示出了支持已编码视频的双向传输的第二对终端装置(130,140),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,每个终端装置(130,140)可对在本地位置采集的视频数据进行编码,以通过网络(150)传输到另一终端装置。每个终端装置(130,140)还可接收由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并在本地显示设备上显示恢复的视频数据。

[0026] 在图1中,终端装置(110-140)可为服务器、个人计算机和智能手机,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(150)表示在终端装置(110-140)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。代表性网络包括电信网络、局域网、广域网和/或互联网。出于本讨论的目的,除非在下文中有所解释,否则网络(150)的架构和拓扑对于本公开的操作来说可能是无关紧要的。

[0027] 作为所公开主题的应用实施例,图2示出了视频解码器和编码器在流式传输环境中的放置方式。所公开的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。

[0028] 流式传输系统可包括采集子系统(213),所述采集子系统可包括诸如数码相机等视频源(201),所述视频源创建例如未压缩的视频样本流(202)。较于已编码的视频比特流,样本流(202)被描绘为粗线,以强调其为高数据量的视频样本流,样本流(202)可由耦接到相机(201)的编码器(203)处理。编码器(203)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。相较于样本流(202),已编码的视频比特流(204)被描绘为细线,以强调较低数据量的已编码的视频比特流,其可存储在流式传输服务器(205)上以供将来使用。一个或多个流式传输客户端(206,208)可访问流式传输服务器

(205) 以检索已编码的视频比特流 (204) 的副本 (207, 209)。客户端 (206) 可包括视频解码器 (210)。视频解码器 (210) 对已编码的视频比特流的传入副本 (207) 进行解码, 且产生可在显示器 (212) 或另一呈现装置 (未示出) 上呈现的输出视频样本流 (211)。在一些流式传输系统中, 可以根据某些视频编解码/压缩标准对视频比特流 (204, 207, 209) 进行编码。这些标准的示例包括 ITU-T H.265 建议书。正在开发的视频编解码标准被非正式地称为多功能视频编解码 (VVC)。所公开的主题可以在 VVC 的上下文中使用。

[0029] 图3是根据本公开实施例的视频解码器 (210) 的功能框图。

[0030] 接收器 (310) 可接收将由视频解码器 (210) 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 (312) 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 (310) 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 (310) 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 (315) 可耦接在接收器 (310) 与熵解码器/解析器 (320) (此后称为“解析器”) 之间。当接收器 (310) 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 (315), 或可以将所述缓冲存储器做得较小。当然, 为了在互联网等分组网络上使用, 也可能需要缓冲存储器 (315), 所述缓冲存储器可相对较大且可有利地具有自适应性大小。

[0031] 视频解码器 (210) 可包括解析器 (320) 以根据熵编码视频序列重建符号 (321)。这些符号的类别包括用于管理视频解码器 (210) 的操作的信息, 以及用以控制诸如显示器 212 的显示装置的潜在信息, 所述显示装置不是解码器的组成部分, 但可耦接到解码器, 如图3中所示。用于显示装置的控制信息可以是辅助增强信息 (Supplementary 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) 的形式。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解码可根据视频编解码技术或标准进行, 且可遵循本领域技术人员公知的原理, 包括可变长度编解码、霍夫曼编解码 (Huffman coding)、具有或不具有上下文灵敏度的算术编解码等等。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、子图片、图块 (tile)、切片 (slice)、砖块 (brick)、宏块、编码树单元 (Coding Tree Unit, CTU)、编码单元 (Coding Unit, CU)、块、变换单元 (Transform Unit, TU)、预测单元 (Prediction Unit, PU) 等等。图块可以指示图片中的特定图块列和行内的 CU/CTU 的矩形区域。砖块可以指示特定图块内的 CU/CTU 行的矩形区域。切片可以指示图片的一个或多个砖块, 其包含在 NAL 单元中。子图片可以指示图片中的一个或多个切片的矩形区域。熵解码器/解析器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。

[0032] 解析器 (320) 可对从缓冲存储器 (315) 接收的视频序列执行熵解码/解析操作, 从而创建符号 (321)。

[0033] 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 (321) 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(320)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(320)与下文的多个单元之间的此类子群控制信息流。

[0034] 除已经提及的功能块以外,视频解码器(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中,这些单元中的许多单元彼此紧密交互并且可以至少部分地彼此集成。然而,出于描述所公开的主题的目的,概念上细分成下文的功能单元是适当的。

[0035] 第一单元是缩放器/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。

[0036] 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(部分重建的)当前图片(358)提取的周围已重建的信息生成大小和形状与正在重建的块相同的块。在一些情况下,聚合器(355)基于每个样本,将帧内预测单元(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。

[0037] 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。运动补偿预测单元从参考图片存储器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(321)的形式而供运动补偿预测单元使用,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,从参考图片存储器提取的样本值的内插、运动矢量预测机制等等。

[0038] 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比特流中的参数,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,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。

[0039] 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显示装置(212)以及存储在参考图片存储器,以用于后续的帧间图片预测。

[0040] 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,则当前图片(358)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。

[0041] 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。在已编码视频序列遵循在视频压缩技术文献或标准、特别是配置文件中所规定的视频压缩技术或标准的语法的意义上,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。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)规范和已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。

[0042] 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 noise ratio,SNR)增强层、冗余切片、冗余图片、前向纠错码等形式。

[0043] 图4是根据本公开实施例的视频编码器(203)的功能框图。

[0044] 视频编码器(203)可从视频源(201)(并非该编码器的一部分)接收视频样本,所述视频源可采集将由视频编码器(203)编码的视频图像。

[0045] 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 CrCb 4:2:0、Y CrCb4:4:4)。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)可以为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。本领域技术人员可以容易地理解像素和样本之间的关系。下文侧重于描述样本。

[0046] 根据实施例,视频编码器(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。由控制器设置的参数可包括速率控制相关参数(例如,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)的其它功能,因为它们可能属于针对特定系统设计而优化的视频编码器(203)。

[0047] 一些视频编码器以本领域技术人员容易识别为“编码环路”方式进行操作。作为简单的描述,编码环路可包括编码器(430)(下文称为“源编码器(source coder)”,其负责基于待编码的输入图片和参考图片创建符号)的编码部分、和嵌入于视频编码器(203)中的(本地)解码器(433)。“本地”解码器(433)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(因为在所公开的主题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)是本领域技术人员公知的。

[0048] “本地”解码器(433)的操作可与已在上文结合图3详细描述“远程”解码器(210)相同。然而,另外简要参考图3,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,视频解码器(210)的熵解码部分(包括信道(312)、接收器(310)、缓冲存储器(315)和解析器(320))可能无法完全在本地解码器(433)中实施。

[0049] 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,所公开的主题专注于解码器操作。对编码器技术的描述可以简写,因为它们与全面描述的了解码器技术互逆。仅在某些区域中需要更详细的描述,并且在下文提供。

[0050] 作为操作的一部分,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。

[0051] 本地视频解码器(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)的操作可以有利地为有损过程。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。本地视频解码器(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得的重建参考帧具有共同内容(不存在传输误差)。

[0052] 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)获得的搜索结果,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。

[0053] 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。

[0054] 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。熵编码器可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公知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。

[0055] 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)。

[0056] 控制器(450)可管理视频编码器(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以下任一种图片类型。

[0057] 帧内图片(I图片),其可以是不将序列中的任何其它帧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。本领域技术人员知道I图片的那些变体以及它们各自的应用和特征。

[0058] 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。

[0059] 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。

[0060] 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。

[0061] 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。

[0062] 在实施例中,传输器(440)可传输附加数据和已编码的视频。源编码器(430)可以包括诸如已编码视频序列的一部分的此类数据。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。

[0063] 最近,将多个语义独立的图片部分进行压缩域聚合或提取到单个视频图片中已经引起了一些关注。具体地,在例如360编码或某些监视应用的上下文中,多个语义独立的源图片(例如立方体投影的360场景的六个立方体表面、或在多摄像头监视设置情况下的各个摄像头输入)可能需要单独的自适应分辨率设置,以应对给定时间点的不同场景活动。换句话说,编码器在给定时间点可以对构成整个360场景或监视场景的不同语义独立的图片选择使用不同的重采样因子。当组合成单个图片时,这又要求对已编码图片的部分执行参考图片重采样,并且自适应分辨率编码信令可用。

[0064] 下面将介绍一些术语,这些术语将在本说明书的其余部分中提及。

[0065] 在某些情况下,子图片可以指样本、块、宏块、编码单元或类似实体的矩形排列,这些实体按语义分组,并且可以以变化的分辨率独立编码。一个或多个子图片可以形成一个图片。一个或多个已编码子图片可以形成一个已编码图片。可以将一个或多个子图片组合成图片,并且可以从图片中提取一个或多个子图片。在某些环境中,一个或多个已编码子图片可以在压缩域中组合,而不需要转码到样本级别成为已编码图片,并且在相同或其它情况下,可以从压缩域中的已编码图片中提取一个或多个已编码子图片。

[0066] 参考图片重采样(RPR)或自适应分辨率改变(ARC)可以指允许通过例如参考图片重采样来改变已编码视频序列内的图片或子图片的分辨率的机制。此后,RPR/ARC参数是指执行自适应分辨率改变所需的控制信息,该控制信息可以包括,例如滤波器参数、缩放因子、输出和/或参考图片的分辨率、各种控制标志等。

[0067] 在实施例中,可对单个语义独立的已编码视频图片执行编码和解码。在描述具有独立RPR/ARC参数的多个子图片的编码/解码的含义及其隐含的额外复杂度之前,将描述用于发信号通知RPR/ARC参数的选项。

[0068] 参考图5,示出了发信号通知RPR/ARC参数的几个实施例。如每个实施例所述,从编解码效率、复杂度和架构的观点来看,它们可能具有某些优点和缺点。视频编解码标准或技术可以选择这些实施例中的一个或多个,或相关技术中已知的选项,以发信号通知RPR/ARC参数。这些实施例可以不相互排斥,并且可以基于应用需要、所涉及的标准技术或编码器的选择进行互换。

[0069] RPR/ARC参数的类别可以包括:

[0070] -在X维度和Y维度上分开或组合的上/下采样因子

[0071] -加上时间维度的上/下采样因子,表示对给定数量的图片进行恒速放大/缩小

[0072] -上述两种类别中的任何一种都可能涉及一个或多个可能较短的语法元素的编码,该语法元素可以指向包含一个或多个因子的表。

[0073] -以样本、块、宏块、编码单元(CU)或任何其它合适的粒度为单位,输入图片、输出图片、参考图片、已编码图片在X或Y维度上的组合或单独的分辨率。如果存在多于一个的分辨率(例如,一个用于输入图片,一个用于参考图片),则在某些情况下,一组值可以从另一组值推断出来。这可以例如使用标志来选通。更详细的示例请参见下文。

[0074] -“翘曲”坐标类似于H.263附录P中使用的那些坐标,其同样以上述合适的粒度为单位。H.263附录P定义了一种对这种翘曲坐标(warping coordinate)进行编码的有效方式,但也可以设想其它可能更有效的方式。例如,附录P的翘曲坐标的可变长度可逆的“霍夫曼(Huffman)”式编码可以由适当长度的二进制编码代替,其中二进制码字的长度例如可以从最大图片大小导出,可能乘以某个因子并偏移某个值,以允许在最大图片大小的边界之外“翘曲”。

[0075] -上采样或下采样滤波器参数。在实施例中,可能只有单个滤波器用于上采样和/或下采样。然而,在实施例中,可能希望在滤波器设计中允许有更大的灵活性,并且这可能需要滤波器参数的信令。这些参数可以通过可能的滤波器设计列表中的索引来选择,滤波器可以(例如,通过滤波器系数的列表、使用适当的熵编码技术)完全指定,滤波器可以通过上采样/下采样率隐式地选择,该上采样/下采样率又根据上述机制中的任何一种发信号通知,等等。

[0076] 此后,本说明书假设对一组有限的上采样/下采样因子进行编码(在X维度和Y维度上使用相同的因子),通过码字表示。可以例如使用对诸如H.264和H.265的视频编解码规范中的某些语法元素通用的指数哥伦布(Ext-Golomb)代码,对该码字进行可变长度编码。例如,一种合适的从值到上采样/下采样因子的映射可以根据表1:

[0077] 表1

码字	Ext-Golomb 代码	原始/目标分辨率
0	1	1/1
1	010	1/1.5 (放大 50%)
2	011	1.5/1 (缩小 50%)
3	00100	1/2 (放大 100%)
4	00101	2/1 (缩小 100%)

[0080] 可以根据应用的需要以及视频压缩技术或标准中可用的放大和缩小机制的能力

来设计许多类似的映射。该表可扩展到更多的值。值也可以由除Ext-Golomb代码之外的熵编码机制来表示,例如使用二进制编码来表示。当例如MANE对视频处理引擎(最重要的是编码器和解码器)本身之外的重采样因子感兴趣时,这可能具有某些优势。应当注意,对于不需要改变分辨率的情况,可以选择较短的Ext-Golomb代码;在上表中,只有一个比特。在最常见的情况下,这比使用二进制码具有编解码效率的优势。

[0081] 该表中的条目的数量以及它们的语义可以完全或部分可配置。例如,该表的基本轮廓(outline)可以在诸如序列或解码器参数集之类的“高”参数集中传送。在实施例中,一个或多个这样的表可以在视频编解码技术或标准中定义,并且可以通过例如解码器或序列参数集来选择。

[0082] 下面描述如何在视频编解码技术或标准语法中包括如上所述编码的上采样/下采样因子(ARC信息)。类似的考虑可以应用于一个或几个控制上采样/下采样滤波器的码字。当滤波器或其它数据结构需要相对大量的数据时,请参见下面的讨论。

[0083] 如图5所示,H.263附录P以四个翘曲坐标形式将ARC信息(502)包括在图片头(501)中,具体包括在H.263PLUSPTYPE(503)头扩展中。当a)有可用的图片头,并且b)预计ARC信息频繁改变时,这可能是一个明智的设计选择。然而,当使用H.263样式的信令时,开销可能相当高,并且因为图片头可能具有瞬态性质,缩放因子可能不适用于图片边界。

[0084] 在相同或另一实施例中,ARC参数的信令可以遵循如图6A-6B中概述的详细示例。图6A-6B使用大致遵循C格式编程的符号以一种表示形式描绘语法图,例如至少自1993年以来在视频编解码标准中使用的。加粗字体的行表示比特流中存在的语法元素,不加粗字体的行通常表示控制流或变量设置。

[0085] 如图6A所示,作为可应用于图片的一部分(可能是矩形的)的头的示例性语法结构,图块组头(601)可以有条件地包含可变长度的、Exp-Golomb编码的语法元素dec_pic_size_idx(602)(以粗体字表示)。图块组头中该语法元素的存在可以基于自适应分辨率(603)的使用来选通—在这里,标志的值未以加粗字体表示,这意味着该标志在它在语法图中出现的点处存在于比特流中。是否对该图片或其部分使用自适应分辨率可以在比特流内部或外部的任何高级语法结构中发信号通知。在所示的示例中,它在序列参数集中发信号通知,如下所述。

[0086] 参考图6B,还示出了序列参数集(610)的摘录。所示的第一语法元素是adaptive_pic_resolution_change_flag(611)。当为真时,该标志可以指示使用自适应分辨率,这反过来可能需要某些控制信息。在该示例中,这种控制信息基于该标志的值有条件地出现,该标志的值基于参数集(612)和图块组头(601)中的if()语句。

[0087] 当使用自适应分辨率时,在该示例中,被编码的是以样本为单位的输出分辨率(613)。数字613是指output_pic_width_in_luma_samples和output_pic_height_in_luma_samples,它们可以共同定义输出图片的分辨率。在视频编解码技术或标准的其它地方,可以定义对任一个值的某些限制。例如,级别定义可以限制总输出样本的数量,该数量可以是这两个语法元素的值的乘积。而且,某些视频编解码技术或标准、或外部技术或标准(例如系统标准)可能会限制编号范围(例如,一个或两个维度必须能够被2的幂整除)或宽高比(例如,宽度和高度必须是例如4:3或16:9的关系)。可以引入此类限制以促进硬件实现,或出于其它原因引入此类限制,并且在本领域中是公知的。

[0088] 在某些应用中,可以建议编码器指示解码器使用某个参考图片大小,而不是隐式地假设该大小为输出图片大小。在该示例中,语法元素reference_pic_size_present_flag (614) 选通参考图片维度 (615) 的有条件存在(同样,该数字指的是宽度和高度)。

[0089] 最后,示出了可能的解码图片宽度和高度的表。可以例如通过表指示(num_dec_pic_size_in_luma_samples_minus1) (616) 来表示这样的表。“minus1”可以指对该语法元素的值的解释。例如,如果已编码值为零,则存在一个表条目。如果该值为5,则存在六个表条目。对于表中的每一“行”,已解码图片宽度和高度则包括在语法 (617) 中。

[0090] 可以使用图块组头中的语法元素dec_pic_size_idx (602) 对所呈现的表条目 (617) 进行索引,从而允许每个图块群组具有不同的已解码大小——实际上是缩放因子。

[0091] 某些视频编解码技术或标准(例如VP9)通过结合时间可缩放性实施某些形式的参考图片重采样(以与所公开的主题完全不同的方式发信号通知)来支持空间可缩放性,从而实现空间可缩放性。具体地,某些参考图片可以使用ARC样式的技术上采样到更高的分辨率,以形成空间增强层的基础。这些上采样的图片可以使用高分辨率的正常预测机制来修正,以增加细节。

[0092] 本文讨论的实施例可以在这样的环境中使用。在某些情况下,在相同或另一实施例中,NAL单元头中的值(例如时间ID字段)不仅可用于指示时间层,还可用于指示空间层。这样做对于某些系统设计可能有一定的好处。例如,基于NAL单元头时间ID值为时间层选择转发创建和优化的现有的选择转发单元(SFU, Selected Forwarding Unit)可以在不进行修改的情况下用于可缩放环境。为了实现这一点,可能需要已编码图片大小与时间层之间的映射由NAL单元头中的时间ID字段来指示。

[0093] 最近,将多个语义独立的图片部分进行压缩域聚合或提取到单个视频图片中已经引起了一些关注。具体地,在例如360编码或某些监视应用的上下文中,多个语义独立的源图片(例如立方体投影的360场景的六个立方体表面、或在多摄像头监视设置情况下的各个摄像头输入)可能需要单独的自适应分辨率设置,以应对给定时间点的不同场景活动。换句话说,编码器在给定时间点可以对构成整个360场景或监视场景的不同语义独立的图片选择使用不同的重采样因子。当组合成单个图片时,这又要求对已编码图片的部分执行参考图片重采样,并且自适应分辨率编码信令可用。

[0094] 在实施例中,一致性窗口可以是图片的矩形子部分,其中编码器已指示该子部分的样本由解码器输出,而窗口之外的样本可以从解码器的输出中省略。

[0095] 本文解决的一个问题是一致性窗口可用于两个目的。一个目的是定义如上所述的输出大小。输出大小可以是应用程序驱动的,并且可以与已编码图片大小完全不同,例如在立体或360立方体贴图(cube-map)应用中。第二个目的可以是一致性窗口也输入到上述参考图片重采样。

[0096] 当已编码和重采样的图片具有一致性窗口大小和图片大小的某些组合时,可能会参考填充区域之外的样本。虽然通过对一致性窗口大小进行比特流约束可能会解决这个问题,但是该解决方案会具有如下缺点:应用程序不能再自由选择一致性窗口。在一些应用中,分离这两个功能可能是有利的。

[0097] 在实施例中,一致性窗口大小可以在PPS中发信号通知。当参考图片的一致性窗口大小与当前图片的一致性窗口大小不同时,一致性窗口参数可用于计算重采样率。解码器

可能需要识别每个图片的一致性窗口大小,以确定是否需要重采样过程。

[0098] 在实施例中,参考图片重采样(RPR)的缩放因子可以基于输出宽度和输出高度来计算,该输出宽度和输出高度可以从当前图片和参考图片之间的一致性窗口参数导出。与使用已解码图片大小相比,这可以允许更准确地计算缩放因子。这可以适用于大多数视频序列,其输出图片大小与已解码图片大小几乎相同,具有较小的填充区域。

[0099] 然而,这也可能导致各种问题。例如,当一致性窗口大小由于较大的偏移值与已解码图片大小完全不同时,对于沉浸式媒体应用(immersive media application)(例如360立方体贴图、立体、点云),基于一致性窗口大小的缩放因子计算可能不能保证具有不同分辨率的帧间预测的质量。在极端情况下,参考图片中的当前CU的同位区域可能不存在。当RPR用于多层的可缩放性时,一致性窗口偏移可能不用于计算跨层的参考区域。注意,在HEVC可缩放性扩展(SHVC)中,可以在PPS扩展中显式地发信号通知每个直接相关层的参考区域。当从整个比特流中提取针对特定区域(子图片)的子比特流时,一致性窗口大小与图片大小根本不匹配。注意,一旦对比特流进行编码,只要参数用于缩放计算,一致性窗口参数就不能更新了。

[0100] 基于上述潜在的问题,基于一致性窗口大小的缩放因子计算可能存在困境(corner case),为此需要可替代的参数。作为备用,当一致性窗口参数不能用于计算缩放因子时,建议发信号通知参考区域参数,其可用于计算RPR的缩放和可缩放性。

[0101] 在实施例中,参考图7,conformance_window_flag可以在PPS中发信号通知。conformance_window_flag等于1可以指示在PPS中下一个是一致性裁剪(cropping)窗口偏移参数。conformance_window_flag等于0可以指示不存在一致性裁剪窗口偏移参数。

[0102] 在实施例中,仍然参考图7,conf_win_left_offset、conf_win_right_offset、conf_win_top_offset和conf_win_bottom_offset指定从解码过程输出的、参考PPS的图片的样本,其以图片坐标中指定的矩形区域的形式输出。当conformance_window_flag等于0时,可以推断conf_win_left_offset、conf_win_right_offset、conf_win_top_offset和conf_win_bottom_offset的值等于0。

[0103] 在实施例中,一致性裁剪窗口可以包含亮度样本,其具有从SubWidthC*conf_win_left_offset到pic_width_in_luma_samples-(SubWidthC*conf_win_right_offset+1)的水平图片坐标和从SubHeightC*conf_win_top_offset到pic_height_in_luma_samples-(SubHeightC*conf_win_bottom_offset+1)的垂直图片坐标,包括端点。

[0104] SubWidthC*(conf_win_left_offset+conf_win_right_offset)的值可以小于pic_width_in_luma_samples,而SubHeightC*(conf_win_top_offset+conf_win_bottom_offset)的值应小于pic_height_in_luma_samples。

[0105] 变量PicOutputWidthL和PicOutputHeightL可以如下面的等式1和等式2所示导出:

[0106]
$$\text{PicOutputWidthL} = \text{pic_width_in_luma_samples} - \text{SubWidthC} * (\text{conf_win_right_offset} + \text{conf_win_left_offset})$$
 (等式1)

[0107]
$$\text{PicOutputHeightL} = \text{pic_height_in_pic_size_units} - \text{SubHeightC} * (\text{conf_win_bottom_offset} + \text{conf_win_top_offset})$$
 (等式2)

[0108] 当ChromaArrayType不等于0时,两个色度阵列的对应指定样本可以为具有图片坐

标(x/SubWidthC,y/SubHeightC)的样本,其中(x,y)是指定亮度样本的图片坐标。

[0109] 在实施例中,标志可以存在于PPS或另一参数集中,并且可以指示是否在PPS或另一参数集中显式地发信号通知重采样图片大小(宽度和高度)。如果显式地发信号通知重采样图片大小参数,则可以基于重采样图片大小参数计算当前图片和参考图片之间的重采样率。

[0110] 在实施例中,参考图8A,use_conf_win_for_rpr_flag等于0可以指示resampled_pic_width_in_luma_samples和resampled_pic_height_in_luma_samples跟随在适当的位置,例如PPS中的下一个。

[0111] 在实施例中,use_conf_win_for_rpr_flag等于1可以指示不存在resampling_pic_width_in_luma_samples和resampling_pic_height_in_luma_samples。

[0112] 在实施例中,resampling_pic_width_in_luma_samples可以以亮度样本为单位指定参考PPS的每个参考图片的宽度,用于重采样.resampling_pic_width_in_luma_samples可以不等于0,可以为Max(8,MinCbSizeY)的整数倍,并且可以小于或等于pic_width_max_in_luma_samples。

[0113] 在实施例中,resampling_pic_height_in_luma_samples可以以亮度样本为单位指定参考PPS的每个参考图片的高度,用于重采样.resampling_pic_height_in_luma_samples可以不等于0,可以为Max(8,MinCbSizeY)的整数倍,并且可以小于或等于pic_height_max_in_luma_samples。

[0114] 当语法元素resampling_pic_width_in_luma_samples不存在时,可以推断resampling_pic_width_in_luma_samples的值等于PicOutputWidthL。

[0115] 当语法元素resampling_pic_height_in_luma_samples不存在时,可以推断resampling_pic_height_in_luma_samples的值等于PicOutputHeightL。

[0116] 在实施例中,具有参考图片重采样的分数插值过程的示例可以如下处理:

[0117] 该过程的输入可以为:指定当前编码子块的左上样本相对于当前图片的左上亮度样本的亮度位置(xSb,ySb)、指定当前编码子块的宽度的变量sbWidth、指定当前编码子块的高度的变量sbHeight、运动矢量偏移mvOffset、修正的运动矢量refMvLX、选定的参考图片样本阵列refPicLX、半样本插值滤波器索引hpelIfIdx、双向光流标志bdofFlag、以及指定当前块的颜色分量索引的变量cIdx。

[0118] 该过程的输出可以为:预测样本值的(sbWidth+brdExtSize) x (sbHeight+brdExtSize)阵列predSamplesLX。

[0119] 预测块边界扩展大小brdExtSize可以根据等式3导出,如下所示:

[0120]
$$\text{brdExtSize} = (\text{bdofFlag} \parallel (\text{inter_affine_flag}[\text{xSb}][\text{ySb}] \&\&\text{sps_affine_prof_enabled_flag})) ? 2 : 0$$
 (等式3)

[0121] 变量fRefWidth可以设置为等于亮度样本中的参考图片的resampling_pic_width_in_luma_samples。

[0122] 变量fRefHeight可以设置为等于亮度样本中的参考图片的resampling_pic_height_in_luma_samples。

[0123] 运动矢量mvLX可以设置为等于(refMvLX-mvOffset)。

[0124] 如果cIdx等于0,可以根据以下等式4和等式5定义缩放因子及其定点表示:

[0125] $\text{hori_scale_fp} = ((\text{fRefWidth} \ll 14) + (\text{resampling_pic_width_in_luma_samples} \gg 1)) / \text{resampling_pic_width_in_luma_samples}$ (等式4)

[0126] $\text{vert_scale_fp} = ((\text{fRefHeight} \ll 14) + (\text{resampling_pic_height_in_luma_samples} \gg 1)) / \text{resampling_pic_height_in_luma_samples}$ (等式5)

[0127] 设 $(x_{\text{IntL}}, y_{\text{IntL}})$ 是在全样本单元中给出的亮度位置, $(x_{\text{FracL}}, y_{\text{FracL}})$ 是在1/16样本单元中给出的偏移。这些变量可用于指定参考样本阵列 refPicLX 内的分数样本位置。

[0128] 用于参考样本填充的边界块的左上角坐标 $(x_{\text{SbIntL}}, y_{\text{SbIntL}})$ 可以设置为等于 $(x_{\text{Sb}} + (\text{mvLX}[0] \gg 4), y_{\text{Sb}} + (\text{mvLX}[1] \gg 4))$ 。

[0129] 对于预测亮度样本阵列 predSamplesLX 内的每个亮度样本位置 $(x_{\text{L}} = 0 \dots \text{sbWidth} - 1 + \text{brdExtSize}, y_{\text{L}} = 0 \dots \text{sbHeight} - 1 + \text{brdExtSize})$, 对应的预测亮度样本值 $\text{predSamplesLX}[x_{\text{L}}][y_{\text{L}}]$ 可以导出如下:

[0130] 设 $(\text{refxSb}_L, \text{refySb}_L)$ 和 $(\text{refx}_L, \text{refy}_L)$ 是在1/16样本单元中给出的运动矢量 $(\text{refMvLX}, \text{refMvLX})$ 指向的亮度位置, 变量 $\text{refxSb}_L, \text{refx}_L, \text{refySb}_L$ 和 refy_L 可以如等式6到等式9所示导出, 如下所示:

[0131] $\text{refxSb}_L = ((x_{\text{Sb}} \ll 4) + \text{refMvLX}[0]) * \text{hori_scale_fp}$ (等式6)

[0132] $\text{refx}_L = ((\text{Sign}(\text{refxSb}) * ((\text{Abs}(\text{refxSb}) + 128) \gg 8) + x_L * ((\text{hori_scale_fp} + 8) \gg 4) + 32) \gg 6$ (等式7)

[0133] $\text{refySb}_L = ((y_{\text{Sb}} \ll 4) + \text{refMvLX}[1]) * \text{vert_scale_fp}$ (等式8)

[0134] $\text{refy}_L = ((\text{Sign}(\text{refySb}) * ((\text{Abs}(\text{refySb}) + 128) \gg 8) + y_L * ((\text{vert_scale_fp} + 8) \gg 4) + 32) \gg 6$ (等式9)

[0135] 变量 $x_{\text{IntL}}, y_{\text{IntL}}, x_{\text{FracL}}$ 和 y_{FracL} 可以如等式10到等式13所示导出, 如下所示:

[0136] $x_{\text{IntL}} = \text{refx}_L \gg 4$ (等式10)

[0137] $y_{\text{IntL}} = \text{refy}_L \gg 4$ (等式11)

[0138] $x_{\text{FracL}} = \text{refx}_L \& 15$ (等式12)

[0139] $y_{\text{FracL}} = \text{refy}_L \& 15$ (等式13)

[0140] 如果 bdofFlag 等于真 (TRUE) 或 $(\text{sps_affine_prof_enabled_flag}$ 等于真且 $\text{inter_affine_flag}[x_{\text{Sb}}][y_{\text{Sb}}]$ 等于真), 并且以下条件中的一个或多个为真, 则预测亮度样本值 $\text{predSamplesLX}[x_{\text{L}}][y_{\text{L}}]$ 可以通过调用亮度整数样本获取过程来导出, 其中 $(x_{\text{IntL}} + (x_{\text{FracL}} \gg 3) - 1), y_{\text{IntL}} + (y_{\text{FracL}} \gg 3) - 1$ 和 refPicLX 作为输入。

[0141] $-x_{\text{L}}$ 等于0。

[0142] $-x_{\text{L}}$ 等于 $\text{sbWidth} + 1$ 。

[0143] $-y_{\text{L}}$ 等于0。

[0144] $-y_{\text{L}}$ 等于 $\text{sbHeight} + 1$ 。

[0145] 否则, 预测亮度样本值 $\text{predSamplesLX}[x_{\text{L}}][y_{\text{L}}]$ 通过调用亮度样本8抽头插值滤波过程来导出, 其中 $(x_{\text{IntL}} - (\text{brdExtSize} > 0 ? 1 : 0), y_{\text{IntL}} - (\text{brdExtSize} > 0 ? 1 : 0)), (x_{\text{FracL}}, y_{\text{FracL}}), (x_{\text{SbIntL}}, y_{\text{SbIntL}}), \text{refPicLX}, \text{hpelIfIdx}, \text{sbWidth}, \text{sbHeight}$ 和 $(x_{\text{Sb}}, y_{\text{Sb}})$ 作为输入。

[0146] 否则 (c_{Idx} 不等于0), 以下适用:

[0147] 设 $(x_{\text{IntC}}, y_{\text{IntC}})$ 是在全样本单元中给出的色度位置, $(x_{\text{FracC}}, y_{\text{FracC}})$ 是在1/32

样本单元中给出的偏移。这些变量可用于指定参考样本阵列refPicLX内的通用分数样本位置。

[0148] 用于参考样本填充的边界块的左上角坐标(xSbIntC,ySbIntC)可以设置为等于((xSb/SubWidthC)+(mvLX[0]>>5),(ySb/SubHeightC)+(mvLX[1]>>5))。

[0149] 对于预测色度样本阵列predSamplesLX内的每个色度样本位置(xC=0..sbWidth-1,yC=0..sbHeight-1),对应的预测色度样本值predSamplesLX[xC][yC]可以导出如下:

[0150] 设(refxSb_c,refySb_c)和(refx_c,refy_c)是在1/32样本单元中给出的运动矢量(mvLX[0],mvLX[1])指向的色度位置,变量refxSb_c,refySb_c,refx_c和refy_c可以如等式14到等式17所示导出,如下所示:

[0151] $refxSb_c = ((xSb/SubWidthC \ll 5) + mvLX[0]) * hori_scale_fp$ (等式14)

[0152] $refx_c = ((Sign(refxSb_c) * ((Abs(refxSb_c) + 256) \gg 9) + xC * ((hori_scale_fp + 8) \gg 4) + 16) \gg 5$ (等式15)

[0153] $refySb_c = ((ySb/SubHeightC \ll 5) + mvLX[1]) * vert_scale_fp$ (等式16)

[0154] $refy_c = ((Sign(refySb_c) * ((Abs(refySb_c) + 256) \gg 9) + yC * ((vert_scale_fp + 8) \gg 4) + 16) \gg 5$ (等式17)

[0155] 变量xInt_c,yInt_c,xFrac_c和yFrac_c可以如等式18到等式21所示导出,如下所示:

[0156] $xInt_c = refx_c \gg 5$ (等式18)

[0157] $yInt_c = refy_c \gg 5$ (等式19)

[0158] $xFrac_c = refx_c \& 31$ (等式20)

[0159] $yFrac_c = refy_c \& 31$ (等式21)

[0160] 预测样本值predSamplesLX[xC][yC]可以通过调用以(xIntC,yIntC)、(xFracC,yFracC)、(xSbIntC,ySbIntC)、sbWidth、sbHeight和refPicLX作为输入的过程来导出。

[0161] 在实施例中,参考图8B,use_conf_win_for_rpr_flag等于0可以指定在PPS中紧随其后的是resampled_pic_width_in_luma_samples和resampled_pic_height_in_luma_samples.use_conf_wid_for_rpr_flag等于1指定resampling_pic_width_in_luma_samples和resampling_pic_height_in_luma_samples不存在。

[0162] 在实施例中,ref_region_left_offset可以指定已解码图片中的参考区域的左上亮度样本之间的水平偏移.ref_region_left_offset的值应在-2¹⁴到2¹⁴-1的范围内,包括端值。当不存在时,可以推断ref_region_left_offset的值等于conf_win_left_offset。

[0163] 在实施例中,ref_region_top_offset可以指定已解码图片中的参考区域的左上亮度样本之间的垂直偏移.ref_region_top_offset的值应在-2¹⁴到2¹⁴-1的范围内,包括端值。当不存在时,可以推断ref_region_top_offset的值等于conf_win_right_offset。

[0164] 在实施例中,ref_region_right_offset可以指定已解码图片中的参考区域的右下亮度样本之间的水平偏移.ref_layer_right_offset的值应在-2¹⁴到2¹⁴-1的范围内,包括端值。当不存在时,可以推断ref_region_right_offset的值等于conf_win_top_offset。

[0165] 在实施例中,ref_region_bottom_offset可以指定已解码图片中的参考区域的右下亮度样本之间的垂直偏移.ref_layer_bottom_offset的值应在-2¹⁴到2¹⁴-1的范围内,包括端值。当不存在时,可以推断ref_region_bottom_offset[ref_loc_offset_layer_id[i]]的值等于conf_win_bottom_offset。

[0166] 变量PicRefWidthL和PicRefHeightL可以如等式22和等式23所示导出,如下所示:

[0167] $\text{PicRefWidthL} = \text{pic_width_in_luma_samples} - \text{SubWidthC} * (\text{ref_region_right_offset} + \text{ref_region_left_offset})$ (等式22)

[0168] $\text{PicRefHeightL} = \text{pic_height_in_pic_size_units} - \text{SubHeightC} * (\text{ref_region_bottom_offset} + \text{ref_region_top_offset})$ (等式23)

[0169] 变量fRefWidth可以设置为等于亮度样本中的参考图片的PicRefWidthL。

[0170] 变量fRefHeight可以设置为等于亮度样本中的参考图片的PicRefHeightL。

[0171] 运动矢量mvLX可以设置为等于 $(\text{refMvLX} - \text{mvOffset})$ 。

[0172] 如果cIdx等于0,则缩放因子及其固定点表示可以如等式24和等式25所示定义,如下所示“”

[0173] $\text{hori_scale_fp} = ((\text{fRefWidth} \ll 14) + (\text{PicRefWidthL} \gg 1)) / \text{PicRefWidthL}$ (等式24)

[0174] $\text{vert_scale_fp} = ((\text{fRefHeight} \ll 14) + (\text{PicRefHeightL} \gg 1)) / \text{PicRefHeight}$ (等式25)

[0175] 用于参考样本填充的边界块的左上角坐标 $(x_{\text{SbInt}_L}, y_{\text{SbInt}_L})$ 可以设置为等于 $(x_{\text{Sb}} + (\text{mvLX}[0] \gg 4), y_{\text{Sb}} + (\text{mvLX}[1] \gg 4))$ 。

[0176] 对于预测亮度样本阵列predSamplesLX内的每个亮度样本位置 $(x_L = 0 \dots \text{sbWidth} - 1 + \text{brdExtSize}, y_L = 0 \dots \text{sbHeight} - 1 + \text{brdExtSize})$,对应的预测亮度样本值predSamplesLX $[x_L][y_L]$ 可以导出如下:

[0177] 设 $(\text{refxSb}_L, \text{refySb}_L)$ 和 $(\text{refx}_L, \text{refy}_L)$ 是在1/16样本单元中给出的运动矢量 $(\text{refMvLX}, \text{refMvLX})$ 指向的亮度位置。变量 $\text{refxSb}_L, \text{refx}_L, \text{refySb}_L$ 和 refy_L 可以如等式26到等式29所示导出,如下所示:

[0178] $\text{refxSb}_L = (((x_{\text{Sb}} + \text{ref_region_left_offset}) \ll 4) + \text{refMvLX}[0]) * \text{hori_scale_fp}$ (等式26)

[0179] $\text{refx}_L = ((\text{Sign}(\text{refxSb}) * ((\text{Abs}(\text{refxSb}) + 128) \gg 8) + x_L * ((\text{hori_scale_fp} + 8) \gg 4)) + 32) \gg 6$ (等式27)

[0180] $\text{refySb}_L = (((y_{\text{Sb}} + \text{ref_region_top_offset}) \ll 4) + \text{refMvLX}[1]) * \text{vert_scale_fp}$ (等式28)

[0181] $\text{refy}_L = ((\text{Sign}(\text{refySb}) * ((\text{Abs}(\text{refySb}) + 128) \gg 8) + y_L * ((\text{vert_scale_fp} + 8) \gg 4)) + 32) \gg 6$ (等式29)

[0182] 在实施例中,参考图9,当基于一致性窗口大小计算参考图片重采样的重采样率时,执行具有图片间运动补偿预测的当前块解码过程的解码器可以访问参考图片中的已解码像素中的一个或多个像素。在一些示例中,解码过程可能会使用DPB中的不可用的已解码图片,从而可能导致解码器崩溃或不希望的解码行为。

[0183] 例如,如图9所示,基于具有已解码图片大小903和一致性窗口大小904的已解码图片、以及具有已解码图片大小901和一致性窗口大小902的参考图片,当前块906的解码过程可能会使用不可用的参考块905。

[0184] 为了解决这样的问题,在一些情况下,可以使用重采样图片大小的显式信令来计算参考图片重采样的重采样率。

[0185] 在实施例中,当一致性窗口大小与实际重采样率不一致时,一致性窗口大小可能不适合用于计算参考图片重采样的重采样率。例如,当当前图片由立体子图片(左和右)或360图片的多个面(例如,立方体贴图投影的六个面)组成时,一致性窗口大小可以不同于期望的重采样率。在一些情况下,一致性窗口大小可以覆盖已解码图片的子区域。在这些情况下,重采样图片大小可选地用于计算重采样率。

[0186] 图10是用于对已编码视频比特流进行解码的示例过程1000的流程图。在一些实施方式中,图10的一个或多个过程框可以由解码器210执行。在一些实施方式中,图10的一个或多个过程框可以由与解码器210分离或包括解码器210的另一设备或一组设备执行,例如编码器203。

[0187] 如图10所示,过程1000可以包括获取第一标志,所述第一标志用于指示当前图片中存在一致性窗口(框1010)。

[0188] 如图10进一步所示,过程1000可以包括当所述第一标志指示存在所述一致性窗口时,获取第二标志,所述第二标志指示所述一致性窗口是否用于参考图片重采样(框1020)。

[0189] 如图10进一步所示,过程1000可以包括确定所述第二标志指示所述一致性窗口用于所述参考图片重采样(框1030)。如果确定所述第二标志指示所述一致性窗口用于所述参考图片重采样(在框1030处为是),过程1000可以进行到框1040,然后进行到框1060。在框1040,过程1000可以包括基于所述一致性窗口的一致性窗口大小确定所述当前图片和参考图片之间的重采样率。

[0190] 如果确定所述第二标志指示所述一致性窗口不用于所述参考图片重采样(在框1030处为否),过程1000可以进行到框1050,然后进行到框1060。在框1050,过程1000可以包括基于重采样图片大小确定所述重采样率。

[0191] 如图10进一步所示,过程1000可以包括使用所述重采样率对所述当前图片执行所述参考图片重采样(框1060)。

[0192] 在实施例中,可以在图片参数集中发信号通知所述第一标志和所述第二标志。

[0193] 在实施例中,可以基于距所述当前图片的边界的至少一个偏移距离确定所述一致性窗口大小。

[0194] 在实施例中,可以在图片参数集中发信号通知所述至少一个偏移距离。

[0195] 在实施例中,所述重采样图片大小可以在所述已编码视频比特流中指示为所述重采样图片大小的宽度和所述重采样图片大小的高度中的至少一个。

[0196] 在实施例中,可以在图片参数集中发信号通知所述宽度和所述高度中的所述至少一个。

[0197] 在实施例中,所述宽度和所述高度中的所述至少一个可以表示为包括在所述宽度和所述高度中的所述至少一个中的亮度样本的数量。

[0198] 在实施例中,可以基于距所述当前图片的参考区域的左上亮度样本的至少一个偏移距离,确定所述重采样图片大小。

[0199] 在实施例中,可以在图片参数集中发信号通知所述至少一个偏移距离。

[0200] 虽然图10示出了过程1000的示例框,但是在一些实施方式中,过程1000可以包括比图10中所描绘的更多的框、更少的框、不同的框或不同排列的框。另外或可替代地,过程1000的两个或多于两个的框可以并行执行。

[0201] 此外,所提出的方法可以由处理电路(例如,一个或多个处理器或一个或多个集成电路)来实现。在一示例中,该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,以执行所提出的方法中的一个或多个。

[0202] 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11示出了计算机系统1100,其适于实现所公开主题的一些实施例。

[0203] 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。

[0204] 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。

[0205] 图11所示的用于计算机系统1100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1100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 or 要求。

[0206] 计算机系统1100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)。

[0207] 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1101、鼠标1102、触控板1103、触摸屏1110和相关图形适配器1150、数据手套、操纵杆1105、麦克风1106、扫描仪1107、照相机1108。

[0208] 计算机系统1100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1110、数据手套或操纵杆1105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1109、耳机(未示出))、视觉输出设备(例如,包括阴极射线管(CRT)屏幕、液晶显示(LCD)屏幕、等离子屏幕、有机发光二极管(OLED)屏幕的屏幕1110,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)。

[0209] 计算机系统1100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)1120或类似介质1121的光学介质、拇指驱动器1122、可移动硬盘驱动器或固体状态驱动器1123,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。

[0210] 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、载波或其它瞬时信号。

[0211] 计算机系统1100还可以包括通往一个或多个通信网络(1155)的接口。例如,网络

可以是无线的、有线的、光学的。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。网络还包括诸如以太网、无线局域网、蜂窝网络(包括全球移动通信系统(GSM)、第三代(3G)、第四代(4G)、第五代(5G)、长期演进(LTE)等)等的局域网、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(1154),用于连接到某些通用数据端口或外围总线(1149)(例如,计算机系统1100的通用串行总线(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1100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)。作为示例,网络1155可以使用网络接口1154连接到外围总线1149。通过使用这些网络中的任何一个,计算机系统1100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(1154)可使用某些协议和协议栈。

[0212] 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1100的核心1140。

[0213] 核心1140可包括一个或多个中央处理单元(CPU)1141、图形处理单元(GPU)1142、以现场可编程门阵列(FPGA)1143形式的专用可编程处理单元、用于特定任务的硬件加速器1144等。这些设备以及只读存储器(ROM)1145、随机存取存储器(RAM)1146、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(SSD)等)1147等可通过系统总线1148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1148,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1148,或通过外围总线1149进行连接。外围总线的体系结构包括外围组件互联(PCI)、通用串行总线USB等。

[0214] CPU 1141、GPU 1142、FPGA 1143和加速器1144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 1145或RAM 1146中。过渡数据也可以存储在RAM 1146中,而永久数据可以存储在例如内部大容量存储器1147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 1141、GPU 1142、大容量存储器1147、ROM 1145、RAM 1146等紧密关联。

[0215] 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,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。

[0216] 作为实施例而非限制,具有体系结构1100的计算机系统,特别是核心1140,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1140的特定存储器,例如核心内部大容量存储器1147或ROM 1145。实现本申请的各种实施例的软件可以存储在这种设备中并且由核心1140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。该软件可以使核心1140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 1146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1144)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。

[0217] 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。

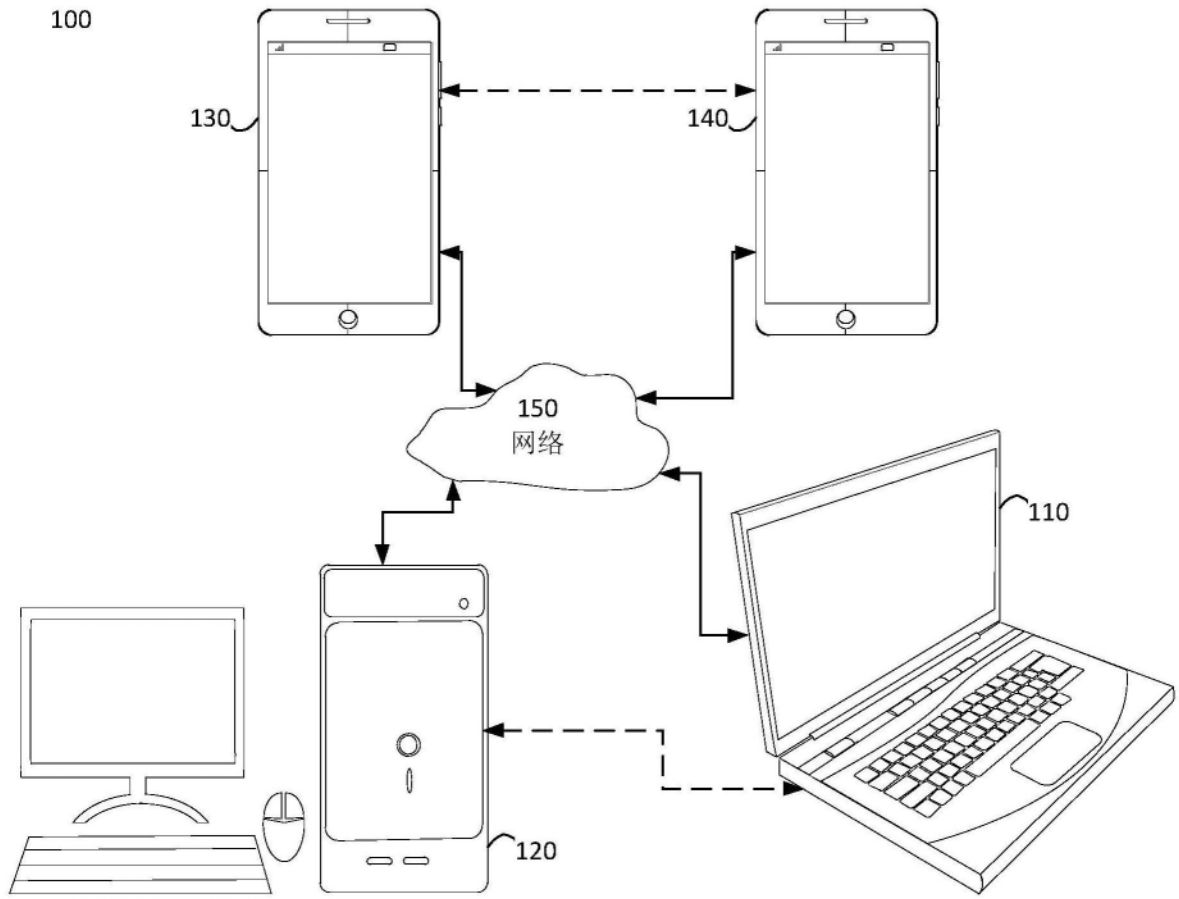


图1

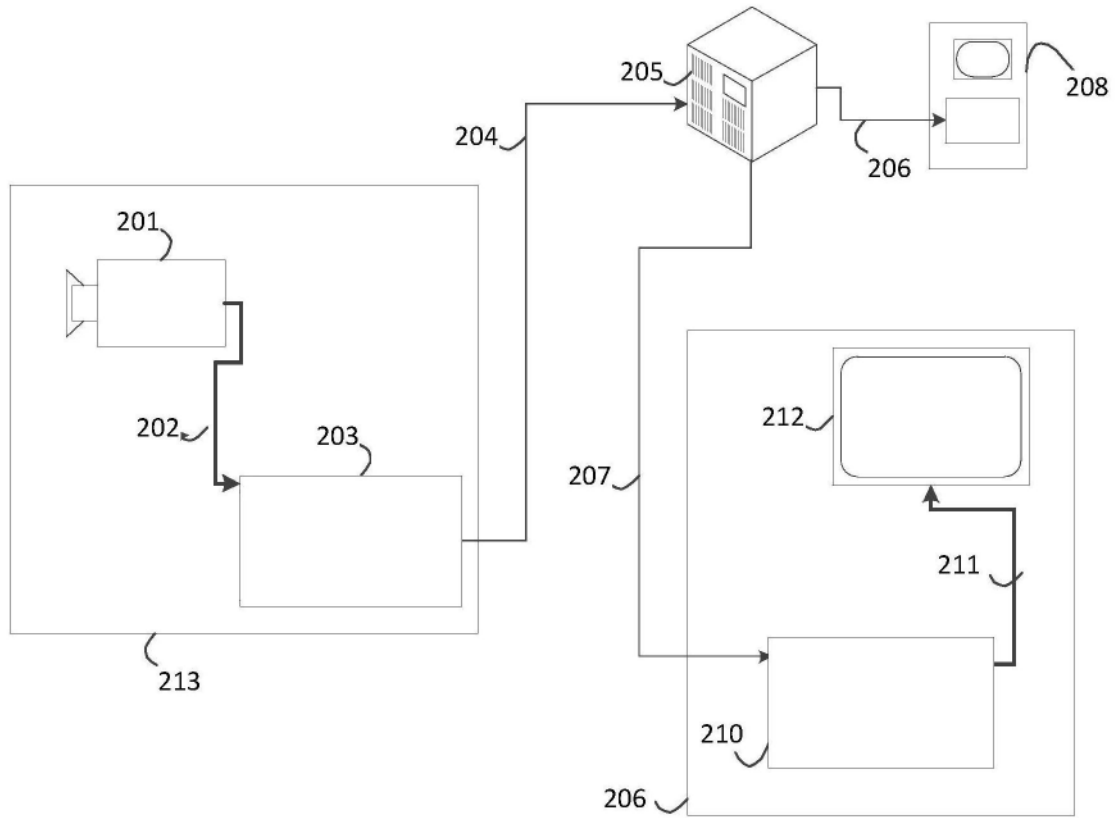


图2

解码器 210

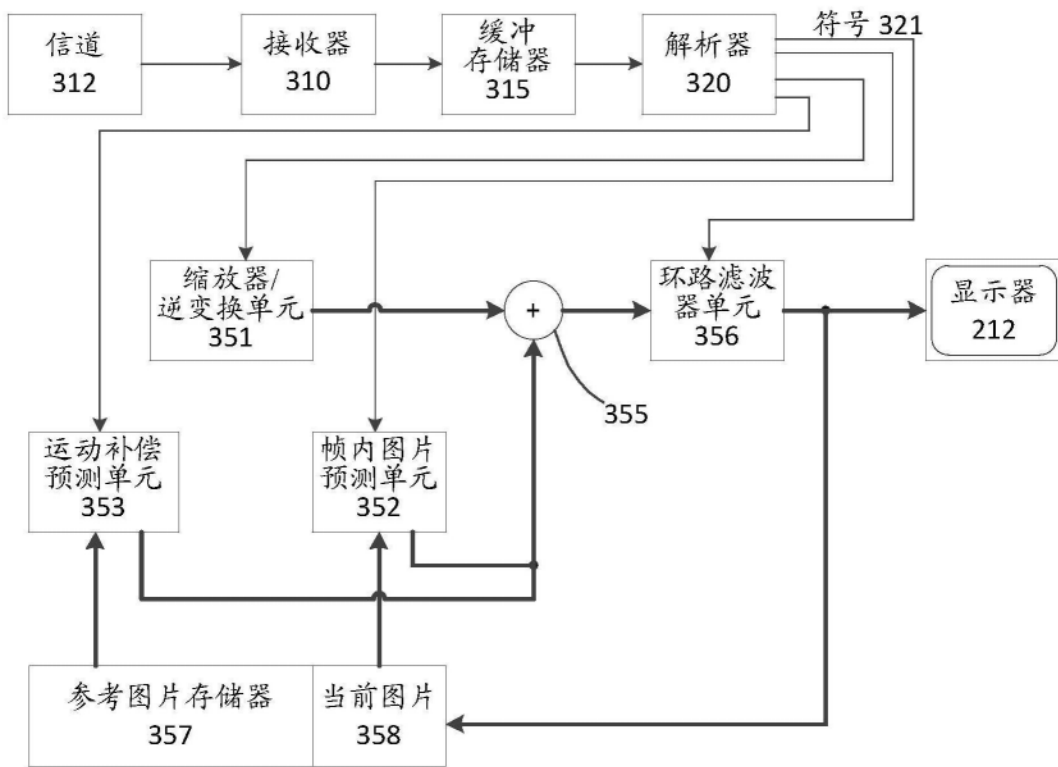


图3

编码器 203

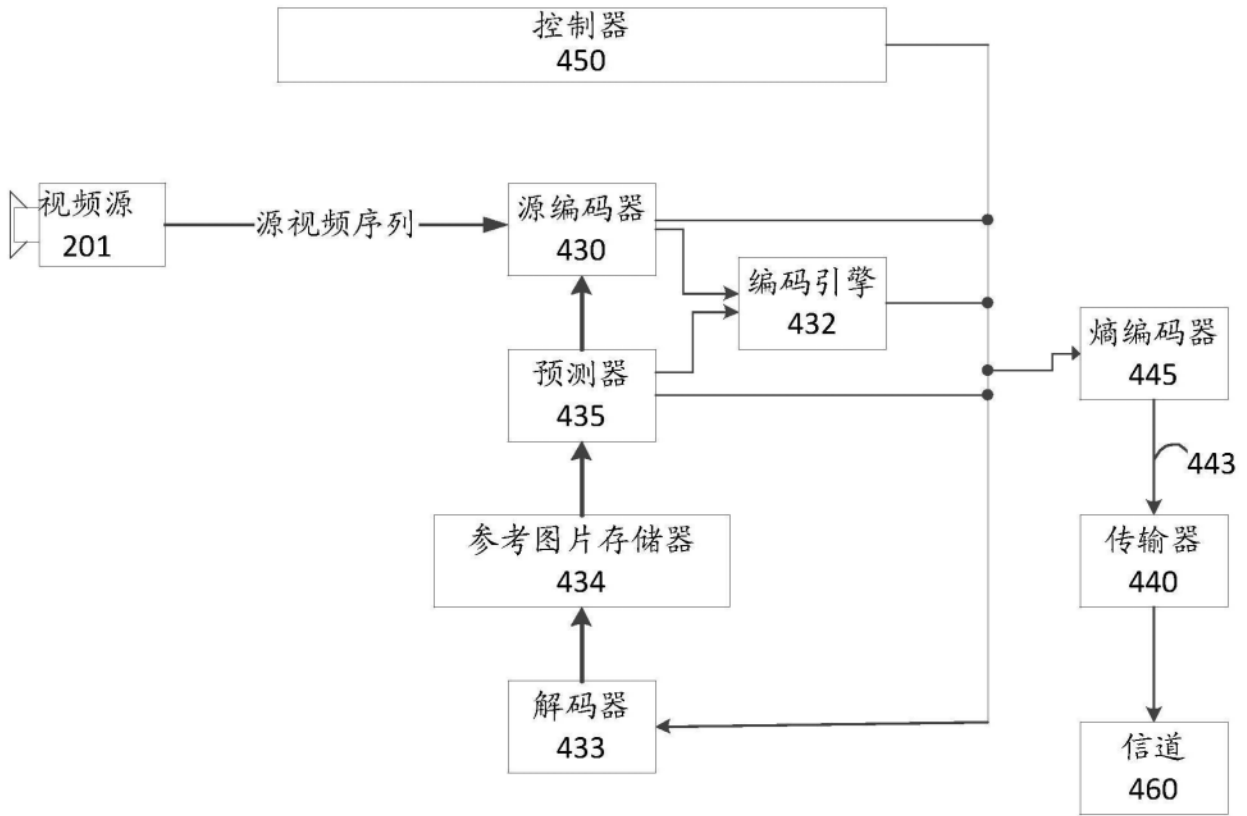


图4

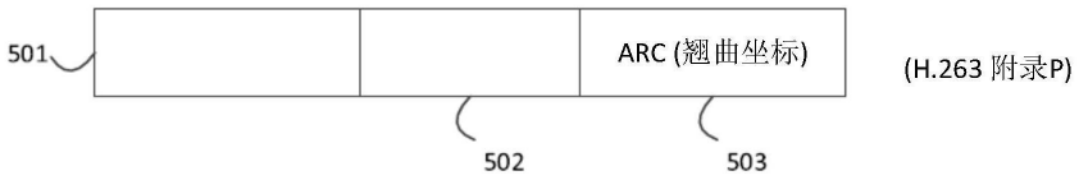


图5

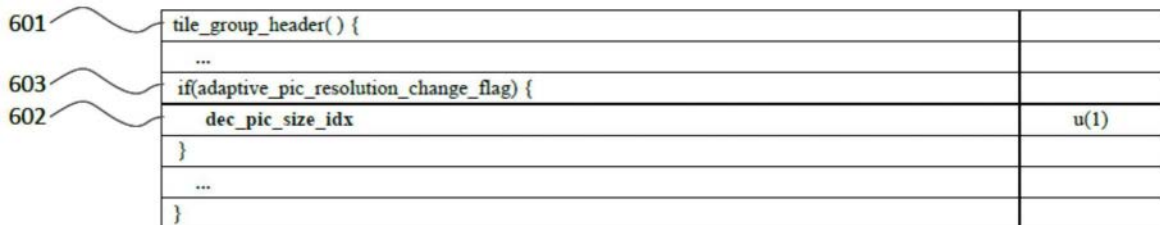


图6A

610	seq_parameter_set_rbsp() {	描述符
	...	
611	adaptive_pic_resolution_change_flag	u(1)
612	if(adaptive_pic_resolution_change_flag) {	
613	output_pic_width_in_luma_samples	ue(v)
	output_pic_height_in_luma_samples	ue(v)
614	reference_pic_size_present_flag	u(1)
	if(reference_pic_size_present_flag)	
	{	
615	reference_pic_width_in_luma_samples	ue(v)
	reference_pic_height_in_luma_samples	ue(v)
	}	
616	num_dec_pic_size_in_luma_samples_minus1	ue(v)
	for(i = 0; i <= num_dec_pic_size_in_luma_samples_minus1; i++) {	
617	dec_pic_width_in_luma_samples[i]	ue(v)
	dec_pic_height_in_luma_samples[i]	ue(v)
	}	
	}	
	...	
	}	

图6B

pic parameter set rbsp() {	描述符
...	
pic width in luma samples	ue(v)
pic height in luma samples	ue(v)
conformance window flag	u(1)
if(conformance window flag) {	
conf win left offset	ue(v)
conf win right offset	ue(v)
conf win top offset	ue(v)
conf win bottom offset	ue(v)
}	
...	
}	

图7

<code>pic_parameter_set_rbsp() {</code>	描述符
<code>...</code>	
<code>pic_width_in_luma_samples</code>	<code>ue(v)</code>
<code>pic_height_in_luma_samples</code>	<code>ue(v)</code>
<code>conformance_window_flag</code>	<code>u(1)</code>
<code>if(conformance_window_flag) {</code>	
<code> conf_win_left_offset</code>	<code>ue(v)</code>
<code> conf_win_right_offset</code>	<code>ue(v)</code>
<code> conf_win_top_offset</code>	<code>ue(v)</code>
<code> conf_win_bottom_offset</code>	<code>ue(v)</code>
<code>}</code>	
<code>use_conf_win_for_rpr_flag</code>	<code>u(1)</code>
<code>if(!use_conf_win_for_rpr_flag) {</code>	
<code> resampling_pic_width_in_luma_samples</code>	<code>ue(v)</code>
<code> resampling_pic_height_in_luma_samples</code>	<code>ue(v)</code>
<code>}</code>	
<code>...</code>	
<code>}</code>	

图8A

<code>pic_parameter_set_rbsp() {</code>	描述符
<code>...</code>	
<code>pic_width_in_luma_samples</code>	<code>ue(v)</code>
<code>pic_height_in_luma_samples</code>	<code>ue(v)</code>
<code>conformance_window_flag</code>	<code>u(1)</code>
<code>if(conformance_window_flag) {</code>	
<code> conf_win_left_offset</code>	<code>ue(v)</code>
<code> conf_win_right_offset</code>	<code>ue(v)</code>
<code> conf_win_top_offset</code>	<code>ue(v)</code>
<code> conf_win_bottom_offset</code>	<code>ue(v)</code>
<code>}</code>	
<code>use_conf_win_for_rpr_flag</code>	<code>u(1)</code>
<code>if(!use_conf_win_for_rpr_flag) {</code>	
<code> ref_region_left_offset</code>	<code>ue(v)</code>
<code> ref_region_right_offset</code>	<code>ue(v)</code>
<code> ref_region_top_offset</code>	<code>ue(v)</code>
<code> ref_region_bottom_offset</code>	<code>ue(v)</code>
<code>}</code>	
<code>...</code>	
<code>}</code>	

图8B

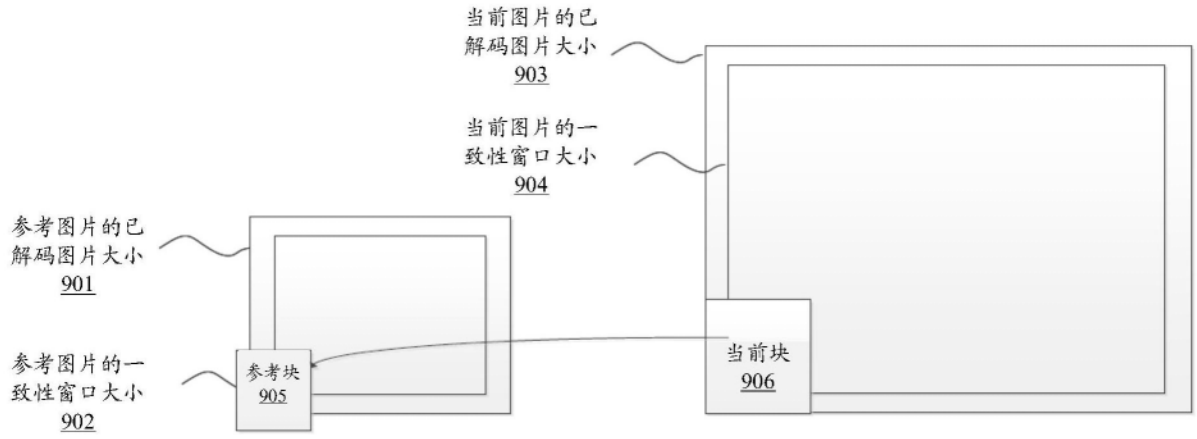


图9

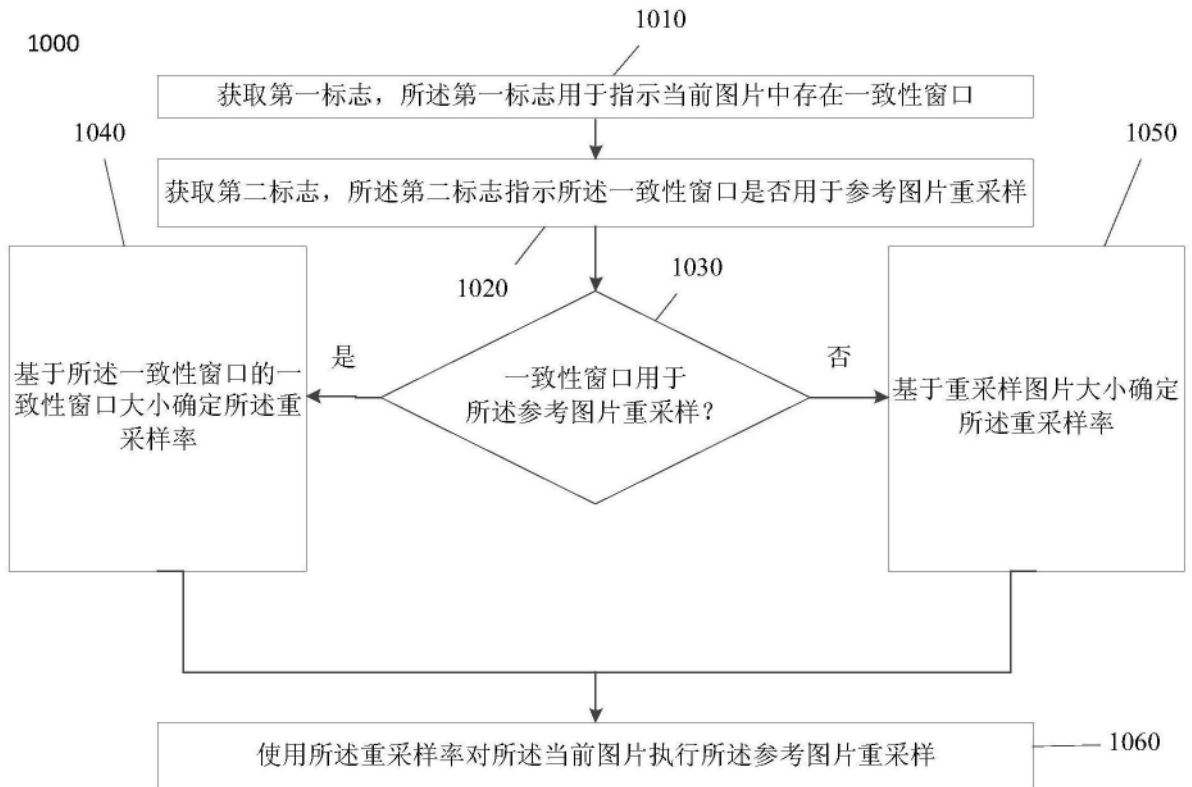


图10

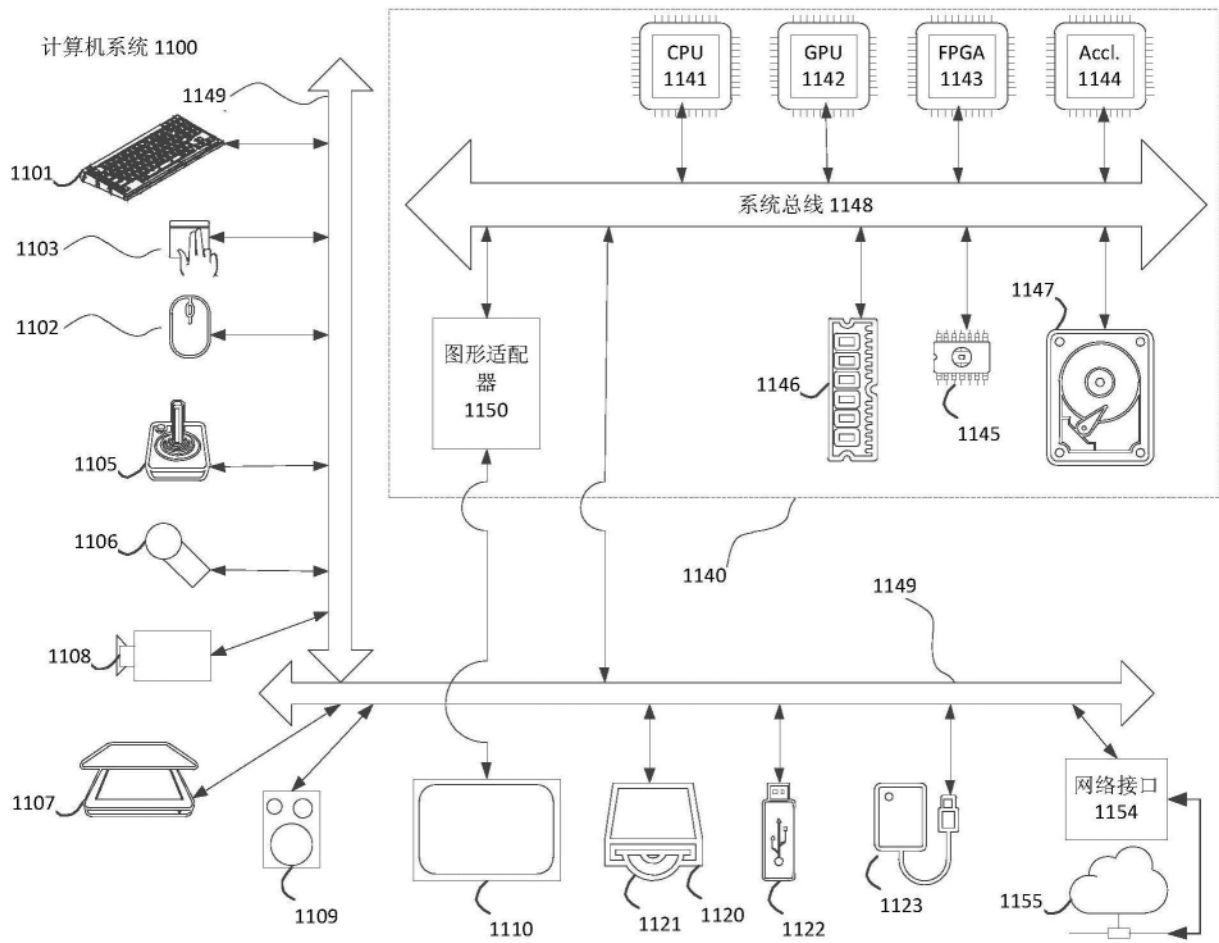


图11