



(12)发明专利

(10)授权公告号 CN 104919804 B

(45)授权公告日 2019.06.04

(21)申请号 201380051471.9

Y.吴 S.卡努穆里 S.萨瓦尼

(22)申请日 2013.09.26

G.J.萨利文

(65)同一申请的已公布的文献号

申请公布号 CN 104919804 A

(74)专利代理机构 北京市金杜律师事务所

11256

(43)申请公布日 2015.09.16

代理人 王茂华

(30)优先权数据

61/708328 2012.10.01 US

14/027013 2013.09.13 US

(51)Int.Cl.

H04N 19/70(2014.01)

H04N 19/30(2014.01)

H04N 19/186(2014.01)

H04N 19/80(2014.01)

H04N 19/59(2014.01)

H04N 19/88(2014.01)

H04N 9/64(2006.01)

(85)PCT国际申请进入国家阶段日

2015.04.01

(86)PCT国际申请的申请数据

PCT/US2013/061784 2013.09.26

(87)PCT国际申请的公布数据

W02014/055310 EN 2014.04.10

(73)专利权人 微软技术许可有限责任公司

地址 美国华盛顿州

(56)对比文件

EP 2456204 A1, 2012.05.23,

EP 0788282 A2, 1997.08.06,

US 8139081 B1, 2012.03.20,

审查员 刘艳

(72)发明人 L.朱 S.桑库拉特里 A.B.库马

权利要求书3页 说明书27页 附图11页

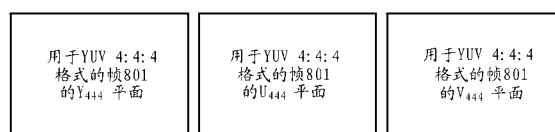
(54)发明名称

帧封装和解封较高分辨率色度采样格式

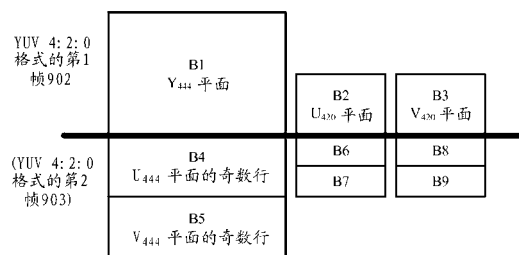
(57)摘要

将诸如YUV 4:4:4之类的较高分辨率色度采样格式的视频帧封装成诸如YUV 4:2:0之类的较低分辨率色度采样格式的视频帧,以用于视频编码的目的。例如,将用于YUV 4:4:4格式的帧的样本值封装成两个YUV 4:2:0格式的帧。解码后,可以对较低分辨率色度采样格式的视频帧进行解封,以便重构较高分辨率色度采样格式的视频帧。通过这种方式,能够在仍然保留较高分辨率色度信息的同时,使用在较低分辨率色度采样格式上操作的可用的编码器和解码器。在示例实施方式中,将YUV 4:4:4格式的帧封装成YUV 4:2:0格式的帧,使得YUV 4:2:0格式的帧的Y、U和V分量之间的几何对应得以保持。

900



封装为两个YUV 4:2:0格式的视频帧,每个第二行作为B4和B5



1. 一种方法,包括:

接收一个或多个较高分辨率色度采样格式的帧,所述一个或多个较高分辨率色度采样格式的帧包括色度分量的样本值和亮度分量的样本值;以及

将一个或多个较高分辨率色度采样格式的帧封装成一个或多个较低分辨率色度采样格式的帧,其中所述较低分辨率色度采样格式具有比所述较高分辨率色度采样格式更低的色度分辨率,并且其中所述封装包括将一个或多个较高分辨率色度采样格式的帧的色度分量的样本值的一些分配给所述一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量的样本值。

2. 根据权利要求1所述的方法,还包括:在所述封装后,对所述一个或多个较低分辨率色度采样格式的帧进行编码。

3. 根据权利要求2所述的方法,其中,所述封装根据(1)所述一个或多个较低分辨率色度采样格式的帧中的所述一个帧的亮度分量的样本值与(2)所述一个或多个较低分辨率色度采样格式的帧中的所述一个帧的色度分量的相应比例的位置处的样本值之间的关系,保持了所述一个或多个较高分辨率色度采样格式的帧的色度分量的相邻样本值之间的几何对应,所述一个或多个较低分辨率色度采样格式的帧中的所述一个帧的亮度分量的所述样本值已经被分配了所述一个或多个较高分辨率色度采样格式的帧的色度分量的样本值中的一些样本值,并且其中所述编码利用所述几何对应。

4. 根据权利要求3所述的方法,其中,所述编码(a)在编码操作中利用所述几何对应,该编码操作包括运动向量的导出和预测模式的导出中的一个或多个,并且/或者(b)利用所述几何对应来指导编码决策,该编码决策包括运动估计、量化参数选择和预测模式选择中的一个或多个。

5. 根据权利要求1所述的方法,其中,所述一个或多个较低分辨率色度采样格式的帧的第一部分代表所述一个或多个较高分辨率色度采样格式的帧的较低色度分辨率版本,并且其中,所述一个或多个较低分辨率色度采样格式的帧的第二部分代表来自所述一个或多个较高分辨率色度采样格式的帧的剩余色度信息,所述第二部分包括所述一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量。

6. 根据权利要求1所述的方法,其中所述封装包括:

对所述一个或多个较高分辨率色度采样格式的帧的色度分量的样本值进行抗混叠滤波;和

将经滤波的样本值分配给所述一个或多个较低分辨率色度采样格式的帧之中的主视图的色度分量的样本值。

7. 一种计算设备,包括:

一个或多个处理单元;

易失性存储器;以及

非易失性存储器和/或存储器,所述非易失性存储器和/或存储器具有存储于其中的计算机可执行指令,从而使得所述计算设备当被编程时执行重构较高分辨率色度采样格式的帧的操作,所述操作包括:

接收一个或多个较低分辨率色度采样格式的帧,其中所述较低分辨率色度采样格式具有比所述较高分辨率色度采样格式更低的色度分辨率;以及

将一个或多个较低分辨率色度采样格式的帧解封成一个或多个较高分辨率色度采样格式的帧,所述一个或多个较高分辨率色度采样格式的帧包括色度分量的样本值和亮度分量的样本值,其中所述解封包括将一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量的样本值分配给所述一个或多个较高分辨率色度采样格式的帧的色度分量的样本值的一些。

8. 根据权利要求7所述的计算设备,其中所述操作还包括:在所述接收和所述解封前:对所述一个或多个较低分辨率色度采样格式的帧进行解码。

9. 根据权利要求7所述的计算设备,其中,所述一个或多个较低分辨率色度采样格式的帧的第一部分代表所述一个或多个较高分辨率色度采样格式的帧的较低色度分辨率版本,并且其中所述一个或多个较低分辨率色度采样格式的帧的第二部分代表来自所述一个或多个较高分辨率色度采样格式的帧的剩余色度信息,所述第二部分包括所述一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量。

10. 根据权利要求7所述的计算设备,其中,所述操作还包括:

接收指示所述一个或多个较低分辨率色度采样格式的帧的第一部分代表所述一个或多个较高分辨率色度采样格式的帧的较低色度分辨率版本,并指示所述一个或多个较低分辨率色度采样格式的帧的第二部分代表来自所述一个或多个较高分辨率色度采样格式的帧的剩余色度信息的元数据,所述第二部分包括所述一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量。

11. 一种计算设备,包括处理器和存储器,其中所述计算设备实施:

适于对YUV4:2:0格式的帧进行编码的视频编码器和/或适于对YUV4:2:0格式的帧进行解码的视频解码器;和

用于处理补充增强信息信息的模块,所述补充增强信息消息指示(a)一个或多个所述YUV4:2:0格式的帧的第一部分代表一个或多个YUV4:4:4格式的帧的较低色度分辨率版本,(b)所述一个或多个YUV4:2:0格式的帧的第二部分代表来自所述一个或多个YUV4:4:4格式的帧的剩余色度信息,和(c)所述一个或多个YUV4:4:4格式的帧的色度分量的样本值是否已作为封装成所述一个或多个YUV4:4:0格式的帧的一部分而被滤波,所述一个或多个YUV4:4:4格式的帧的色度分量的样本值的一些被分配给所述一个或多个YUV4:2:0格式的帧的第二部分之中的亮度分量的样本值。

12. 一种或多种有形的计算机可读媒体,存储计算机可执行指令,从而使得一个或多个处理单元当被编程时执行重构较高分辨率色度采样格式的帧的操作,所述操作包括:

接收一个或多个较低分辨率色度采样格式的帧,其中所述较低分辨率色度采样格式具有比所述较高分辨率色度采样格式更低的色度分辨率;以及

将一个或多个较低分辨率色度采样格式的帧解封成一个或多个较高分辨率色度采样格式的帧,所述一个或多个较高分辨率色度采样格式的帧包括色度分量的样本值和亮度分量的样本值,其中所述解封包括将一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量的样本值分配给所述一个或多个较高分辨率色度采样格式的帧的色度分量的样本值的一些。

13. 根据权利要求12所述的一种或多种有形的计算机可读媒体,其中所述操作还包括:在所述接收和所述解封前:

对所述一个或多个较低分辨率色度采样格式的帧进行解码。

14. 根据权利要求12所述的一种或多种有形的计算机可读媒体,其中,所述一个或多个较低分辨率色度采样格式的帧中的一个帧的亮度分量是所述一个或多个较低分辨率色度采样格式的帧之中辅助视图的亮度分量,并且其中针对所述一个或多个较高分辨率色度采样格式的帧的色度分量,所述解封包括:

将对应于所述一个或多个较低分辨率色度采样格式的帧之中辅助视图的亮度分量的样本值分配给所述一个或多个较高分辨率色度采样格式的帧的色度分量的隔行或隔列的样本值;

将对应于所述一个或多个较低分辨率色度采样格式的帧之中主视图的色度分量的样本值分配给所述一个或多个较高分辨率色度采样格式的帧的色度分量的其它行或列中的相隔的位置上的样本值;以及

将对应于所述一个或多个较低分辨率色度采样格式的帧之中辅助视图的色度分量的样本值分配给所述一个或多个较高分辨率色度采样格式的帧的色度分量的其它行或列的其它位置上的样本值。

15. 根据权利要求12所述的一种或多种有形的计算机可读媒体,其中,所述一个或多个较高分辨率色度采样格式的帧具有第一比特深度,并且其中所述一个或多个较低分辨率色度采样格式的帧的样本值具有高于所述第一比特深度的第二比特深度。

帧封装和解封较高分辨率色度采样格式

背景技术

[0001] 工程师们使用压缩(也称为源编码)来降低数字视频的比特率。通过将信息转换成较低的比特率形式,压缩减少了存储和传输视频信息的成本。解压缩(也称为解码)是对来自压缩形式的原始信息的版本的重构。“编解码器”是一种编码器/解码器系统。

[0002] 在过去二十年,已通过了各种视频编解码器标准,包括ITU-T H.261、H.262(MPEG-2或ISO/IEC 13818-2)、H.263和H.264(MPEG-4AVC或ISO/IEC 14496-10)标准和MPEG-1(ISO/IEC 11172-2)、MPEG-4视讯(ISO/IEC 14496-2)和SMPTE 421M标准。最近,HEVC标准(ITU-T H.265或ISO/IEC 23008-2)已经在开发中。例如,参见HEVC标准的草案版本JCTVC-I1003,即,“高效视频编码(HEVC)文本规范草案7”(JCTVC-I1003_d5,2012年4月日内瓦第9次会议)。视频编解码器标准通常定义了用于编码视频比特流的语法选项,详细介绍了在编码和解码中使用特定特征时的比特流中的参数。在很多情况下,视频编解码器标准还提供有关解码器为在解码中获得一致结果而应执行的解码操作的详细内容。除了编解码器标准,各种专用的编解码器格式定义了用于编码视频比特流的语法的其他选项以及相应的解码操作。

[0003] 摄像机、动画输出、屏幕捕获模块等视频源通常提供被转换成诸如YUV 4:4:4色度采样格式之类的格式的视频。YUV格式包括具有代表近似亮度值的样本值的亮度(或Y)分量,以及具有代表色差值的样本值的多个色度(或U和V)分量。在YUV 4:4:4格式中,色度信息以和亮度信息相同的空间分辨率表示。

[0004] 许多市售视频编码器和解码器仅支持YUV 4:2:0色度采样格式。与YUV 4:4:4相比,YUV 4:2:0格式是一种对色度信息进行子采样的格式,从而使色度分辨率在水平方向和垂直方向上均为亮度分辨率的一半。作为设计原则,决定使用YUV 4:2:0格式进行编码/解码的前提是,理解对于自然的摄像机捕获的视频内容的编码/解码等大多数使用案例而言,观看者一般不会注意到以YUV 4:2:0格式进行编码/解码的视频和以YUV 4:4:4格式进行编码/解码的视频之间的多处视觉差异。对于每帧具有较少样本的YUV 4:2:0格式而言,压缩的优势因此变得引人注目。然而,有一些使用案例,对其而言,具有更丰富色彩信息和更高色彩保真度的视频可能是合理的。在这些使用案例中,YUV 4:4:4和YUV 4:2:0色度采样格式之间的差异更容易被观看者所察觉。例如,对计算机屏幕文本内容、具有人工硬边边界的动画视频内容、或者更普遍的视频内容(诸如滚动字幕和硬边图形、或信息集中在色度通道中的视频)的某些特征的编码/解码,与4:2:0格式相比,可以优选4:4:4格式。虽然有支持4:4:4格式的编码和解码的屏幕捕获编解码器可用,但是,对这些使用案例而言,缺乏对支持4:4:4格式的编解码器的广泛支持(尤其是关于硬件编解码器的实现)是一道阻碍。

发明内容

[0005] 总之,具体实施方式提出了将较高分辨率色度采样格式的视频帧封装成较低分辨率色度采样格式的视频帧的创新,以用于编码目的。例如,较高分辨率色度采样格式是YUV 4:4:4格式,较低分辨率色度采样格式是YUV 4:2:0格式。解码后,可对较低分辨率色度

采样格式的视频帧进行解封,以便重构较高分辨率色度采样格式的视频帧。通过这种方式,能够在仍然保持较高分辨率色度信息的同时,使用在较低分辨率色度采样格式上操作的可用的编码器和解码器。

[0006] 例如,计算设备将一个或多个较高分辨率色度采样格式的帧封装成一个或多个较低分辨率色度采样格式的帧。该计算设备然后可以对一个或多个较低分辨率色度采样格式的帧进行编码。

[0007] 作为另一示例,计算设备将一个或多个较低分辨率色度采样格式的帧解封成一个或多个较高分辨率色度采样格式的帧。在这种解封之前,该计算设备可以对一个或多个较低分辨率色度采样格式的帧进行解码。

[0008] 封装或解封可以作为方法的一部分、作为适于执行该方法的计算设备的一部分、或作为存储了使计算设备执行该方法的计算机可执行指令的有形计算机可读介质的一部分实现。

[0009] 根据参考附图所进行的以下具体实施方式,本发明的上述及其他目标、特征、和优点将变得更加显而易见。

附图说明

[0010] 图1是可以实现一些所描述的实施例的示例计算系统的图。

[0011] 图2a和2b是可以实现一些所描述的实施例的示例网络环境的图。

[0012] 图3是可以实现一些所描述的实施例的通用帧封装/解封系统的图。

[0013] 图4是与其结合可以实现一些所描述的实施例的示例编码器系统的图。

[0014] 图5是与其结合可以实现一些所描述的实施例的示例解码器系统的图。

[0015] 图6是示出了与其结合可以实现一些所描述的实施例的示例视频编码器的图。

[0016] 图7是示出了与其结合可以实现一些所描述的实施例的示例视频解码器的图。

[0017] 图8是示出了示例帧封装途径的图,该帧封装使用帧的空间分割。

[0018] 图9是示出了示例帧封装途径的图,在该帧封装中,对每个第2行较高分辨率色度采样格式的帧的色度分量平面进行复制。

[0019] 图10是示出了根据图9的途径所封装的示例帧的图。

[0020] 图11是示出了示例帧封装途径的图,在该帧封装中,对每个第2列较高分辨率色度采样格式的帧的色度分量平面进行复制。

[0021] 图12是示出了用于对较高分辨率色度采样格式的帧进行帧封装的通用技术的流程图。

[0022] 图13是示出了用于对较高分辨率色度采样格式的帧进行帧解封的通用技术的流程图。

具体实施方式

[0023] 诸如摄像机、动画输出、屏幕捕获模块等之类的视频源通常提供被转换成诸如YUV4:4:4色度采样格式(更一般地,为4:4:4格式的示例)之类的格式的视频。YUV格式包括具有代表近似亮度值的样本值的亮度(或Y)分量,以及具有代表色差值的样本值的多个色度(或U和V)分量。色差值的精确定义(和向/从YUV色彩空间到诸如RGB之类的另一色彩空间

进行的转换操作)取决于实施方式。一般情况下,如本文中所使用的,术语YUV表示具有亮度分量和一个或多个色度分量的任何色彩空间,包括Y'UV、YIQ、Y'IQ和YDbDr以及诸如YCbCr和YCoCg之类的变体。所使用的分量信号量度可以通过应用非线性传递特点函数(通常称为“伽马预补偿”并且通常使用撇符号指出,尽管撇符号通常为了印刷方便而被省略)进行调整。或者,该分量信号量度可以位于与光振幅有线性关系的域中。亮度和色度分量信号可以与亮度和颜色的感知很好地匹配以用于人类视觉系统,或者该亮度和色度分量信号可以与这种量度稍微偏离(例如,正如在YCoCg变体中,其中应用的公式简化了颜色分量值的计算)。如本文所描述的YUV格式的示例包括那些在名为ITU-R BT.601、ITU-R BT.709和ITU-R BT.2020的国际标准中所描述的示例。色度样本类型的示例在H.264/AVC标准的图E-I中示出。4:4:4格式可以是YUV 4:4:4格式或者是用于另一色彩空间(诸如RGB或GBR)的格式。

[0024] 许多市售视频编码器和解码器仅支持YUV 4:2:0色度采样格式(更一般地,为4:2:0格式的示例)。与YUV 4:4:4相比,YUV 4:2:0是一种对色度信息进行子采样的格式,其保留了全分辨率色度信息(即,色度信息被以和亮度信息相同的分辨率表示)。作为设计原则,决定使用YUV 4:2:0格式进行编码/解码的前提是,理解对于自然的摄像机捕获的诸如视频内容的编码/解码之类的大多数使用案例而言,观看者通常不会注意到以YUV 4:2:0格式进行编码/解码的视频和以YUV 4:4:4格式进行编码/解码的视频之间的多处视觉差异。对于每帧具有较少样本的YUV 4:2:0格式而言,压缩的优势因此变得引人注目。

[0025] 然而,有一些使用案例,对其而言,这两种格式之间的差异更容易被观看者所察觉。例如,对计算机屏幕文本内容(尤其是使用ClearType技术进行渲染的文本)、具有人工硬边边界的动画视频内容、或者更普遍的视频内容(例如滚动字幕和硬边图形、或信息集中在色度通道中的视频)的某些特征进行的编码/解码,与4:2:0格式相比,可能优选4:4:4格式。对这些使用案例而言,缺乏对支持4:4:4格式的视频编解码器的广泛支持(尤其是指硬件编解码器的实现)是一道阻碍。

[0026] 具体实施方式提出了多种将较高分辨率色度采样格式的帧封装成较低分辨率色度采样格式的帧的途径。然后,可以使用专为较低分辨率色度采样格式而设计的编码器,对较低分辨率色度采样格式的帧进行编码。解码(使用专为较低分辨率色度采样格式而设计的解码器)后,可输出较低分辨率色度采样格式的帧,以用于进一步的处理和显示。或者,在这种解码之后,可以通过帧解封来重新获得较高分辨率色度采样格式的帧,以用于输出和显示。在很多情况下,通过保留来自较高分辨率色度采样格式的帧的色度信息,同时利用了适于较低分辨率色度采样格式的市售编解码器,这些途径缓解了现有途径的缺点。特别是,配有专业专用硬件的广泛可用的编解码器能够以更低的功耗,为将YUV 4:4:4视频帧封装成YUV4:2:0视频帧提供更快的编码/解码。

[0027] 在编码/解码使用另一种色度采样格式时,所描述的途径可以用来为一种色度采样格式的帧保留色度信息。本文所描述的一些示例涉及YUV 4:4:4格式的帧的帧封装/解封,以用于使用适于YUV 4:2:0格式的编解码器的编码/解码。本文所描述的其他示例涉及YUV4:2:2格式的帧的帧封装/解封,以用于使用适于YUV 4:2:0格式的编解码器的编码/解码。更一般地,所描述的途径能够用于其他色度采样格式。例如,除了采样率为诸如4:4:4、4:2:2、4:2:0、4:1:1、4:0:0等之类的诸如Y'UV、YIQ、Y'IQ、YdbDr、YCbCr、YCoCg等之类的YUV色彩空间的变体,所描述的途径可以用于以诸如4:4:4、4:2:2、4:2:0、4:1:1、4:0:0等之类

的采样率为色度采样格式的诸如RGB、GBR等之类的色彩空间。

[0028] 在示例实施方式中,本文所描述的创新的具体方面包括但不限于以下内容:

[0029] • 将4:4:4帧封装成两个4:2:0帧,并使用专为4:2:0格式而设计的视频编码器对两个4:2:0帧进行编码。

[0030] • 使用专为4:2:0格式而设计的视频解码器对经编码的帧进行解码,并对两个经解码的4:2:0帧进行解封形成经解码的4:4:4帧。

[0031] • 对YUV格式执行封装,使得用于两个4:2:0帧中的每一个的Y、U和V分量之间的几何对应得以保持。

[0032] • 对YUV格式执行封装,使得两个4:2:0帧(主要视图)中的其中一个代表正在由4:4:4帧代表的完整场景,尽管色度分量的分辨率较低,而其他4:2:0帧(辅视视图)将剩余色度信息封装。

[0033] • 使用一种补充增强信息(“SEI”)消息或其他元数据来用信号通知使用两个4:2:0帧的信号的指示,使得处理该SEI消息的解码器能够输出代表该场景的4:4:4帧或者4:2:0帧。

[0034] • 在只有一个4:2:0帧(来自两个4:2:0帧)用于最终显示时,进行可以改善用于YUV格式的最终显示的帧的质量的预处理和后处理操作。与这种预处理和后处理操作相结合,4:2:0帧能够具有更高的比特深度以用于编码/解码,从而避免了色度信息在预处理和后处理操作中的损失。

[0035] • 将4:2:2帧封装成(两个或更少)4:2:0帧,并使用专为4:2:0格式而设计的视频编码器对4:2:0帧进行编码。

[0036] • 使用专为4:2:0格式而设计的视频解码器对经编码的帧进行解码,并对经解码的4:2:0帧进行解封形成经解码的4:2:2帧。

[0037] 在使用帧封装排列SEI消息的具体示例实施方式中,扩展了帧封装排列SEI消息的定义,从而在标称4:2:0比特流中支持代表4:4:4内容。在一些示例中,一个组成帧(例如,在上下封装(top-bottom packing)或交替帧编码方案中)能够作为普通4:2:0图像以兼容的方式被解码,或者能够使用来自另一个组成帧的数据进行补充,从而形成一个完整的4:4:4图像表示。因为YUV 4:2:0是产品中得到最广泛支持的格式(尤其是对于硬件编解码器的实现),所以,具有一种通过这种解码器传输YUV 4:4:4内容的有效方式,可以提供使能够对YUV 4:4:4能力进行广泛的近期部署的实质性好处(尤其是对屏幕内容编码而言)。在示例实施方式中,4:4:4帧的样本被封装成两个4:2:0帧,并且这两个4:2:0帧作为帧封装排列的组成帧被编码。对于使用帧封装排列SEI消息的实施方式,Content_interpretation_type语法元素的语义被扩展以用信号通知该用法。Content_interpretation_type语法元素用信号通知如何解释使用封装排列进行封装的数据,并且用于所述封装排列的帧配置使用不同的语法元素用信号通知。本文所描述的一些途径对于涉及屏幕内容的应用具有很高的实用价值。此外,相对于原始4:4:4编码,本文所描述的一些途径能够提供与普通4:2:0解码过程相兼容的优点,该普通4:2:0解码过程预期在解码产品中将获得更加广泛的支持。

[0038] 同时,还对用于较高分辨率色度采样格式的帧封装和帧解封的其他创新方面进行了描述。所描述的技术可应用于除视频编码/解码以外的其他应用,诸如静止图像编码、医疗扫描内容编码、多光谱图像内容编码等。虽然本文所描述的操作在某些地方描述为正由

编码器(例如,视频编码器)或解码器(例如,视频解码器)所执行,但是,在许多情况下,操作可以替代性地由另一种类型的媒体处理工具执行。

[0039] 参照针对HEVC标准的语法元素和操作,对本文所描述一些创新进行示例。例如,参考了HEVC标准的草案版本JCTVC-I1003,即“高效视频编码(HEVC)文本规范草案7”(JCTVC-I1003_d5,2012年4月日内瓦第9次会议)。本文所描述的创新还可针对其他标准或格式来实现。例如,本文中所描述的创新可以针对H.264/AVC标准使用帧封装排列SEI消息来实现。

[0040] 更一般地,本文所描述示例可能有各种替代示例。例如,通过改变所描述的方法措施的顺序,通过拆分、重复或省略某些方法动作等,可以更改任何本文所描述的方法。所公开的技术的各个方面能够组合使用或单独使用。不同实施例使用一个或多个所描述的创新。本文所描述的一些创新解决了一个或多个背景技术中指出的问题。通常情况下,特定技术/工具不会解决所有这些问题。

[0041] I. 示例计算系统。

[0042] 图1示出了合适的计算系统(100)的一般性示例,其中可以实现多个所描述的创新。因为创新可以在不同的通用或专用计算系统中实现,所以该计算系统(100)并不旨在对使用范围或功能暗示任何限制。

[0043] 参照图1,计算系统(100)包括一个或多个处理单元(110,115)和内存(120,125)。在图1中,该最基本配置(130)包括在虚线内。处理单元(110,115)执行计算机可执行指令。处理单元可以是通用中央处理单元(“CPU”),特定于应用的集成电路中的处理器或其他任何类型的处理器。在多处理系统中,多个处理单元执行计算机可执行指令以提高处理能力。例如,图1示出了中央处理单元(110)以及图形处理单元或协同处理单元(115)。有形内存(120,125)可以是易失性内存(例如,寄存器、高速缓存、RAM)、非易失性内存(例如,ROM、EEPROM、闪存内存等)或二者的某种组合,可以由(多个)处理单元访问。内存(120,125)以适于(多个)处理单元执行的计算机可执行指令的形式,存储了实现一个或多个创新的软件(180),以用于较高分辨率色度采样格式的帧封装和/或解封。

[0044] 计算系统可具有额外特征。例如,计算系统(100)包括存储器(140)、一个或多个输入设备(150)、一个或多个输出设备(160)、和一个或多个通信连接(170)。诸如总线、控制器或网络之类的互连机制(未示出)将计算系统(100)的组件互相连接。通常,操作系统软件(未示出)为在计算系统(100)中执行的其他软件提供操作环境,并协调计算系统(100)的组件的活动。

[0045] 有形存储器(140)可以是可移动的或不可移动的,并包括磁盘、磁带或盒式磁带、CD-ROM、DVD、或者能够以非易失的方式存储信息并能够在计算系统(100)内被访问的任何其他介质。存储器(140)存储了用于实施的一个或多个创新的软件(180)的指令,以用于较高分辨率色度采样格式的封装和/或解封。

[0046] (多个)输入设备(150)可以是触摸输入设备,诸如键盘、鼠标、笔、或轨迹球、语音输入设备、扫描设备、或向计算系统(100)提供输入的其他设备。对于视频编码,(多个)输入设备(150)可以是摄像机、视频卡、电视调谐卡、或接受模拟或数字形式的视频输入的类似设备,或者是将视频样本读入计算系统(100)的CD-ROM或CD-RW。(多个)输出设备(160)可以是显示器、打印机、扬声器、CD刻录机、或提供来自计算系统(100)的输出的其他设备。

[0047] (多个)通信连接(170)能够通过通信介质实现到另一个计算实体的通信。通信介

质在经调制的数据信号中传递诸如计算机可执行指令、音频或视频输入或输出、或其它数据之类的信息。经调制的数据信号是以在信号中对信息进行编码的方式设置或改变其一个或多个特性的信号。举例来说,而非限制,通信介质能够使用电、光、RF或其它载体。

[0048] 能够在计算机可读介质的一般上下文中对创新进行描述。计算机可读介质是可以在计算环境内被访问的任何可用的有形介质。举例来说,而非限制,关于计算系统(100),计算机可读介质包括内存(120,125)、存储器(140)、以及上述的任何组合。

[0049] 能够在计算机可执行指令(诸如包括在程序模块中的,正在目标真实或虚拟处理器上的计算系统中执行的指令)的一般上下文中对创新进行描述。一般而言,程序模块包括例程、程序、库、对象、类、组件、数据结构等,其执行特定的任务或实现特定的抽象数据类型。程序模块的功能可以在各种实施例中根据需要在程序模块之间被组合或拆分。用于程序模块的计算机可执行指令可以在本地或分布式计算系统内执行。

[0050] 术语“系统”和“设备”在本文互换使用。除非上下文清楚地指出,否则,任一术语均不暗示对计算系统或计算设备的类型的任何限制。一般情况下,计算系统或计算设备可以是本地的或分布式的,并且可包括特殊用途硬件和/或通用硬件与实现本文所描述功能的软件的任何组合。

[0051] 所公开的方法也可以使用配置来执行任何所公开的方法的专业计算硬件来实现。例如,所公开的方法能够通过特别设计或配置用于实现任何所公开的方法的集成电路(例如,专用集成电路(“ASIC”)(诸如ASIC数字信号处理单元(“DSP”)、图形处理单元(“GPU”)、或可编程逻辑器件(“PLD”),诸如现场可编程门阵列(“FPGA”)))来实现。

[0052] 为了便于介绍,具体实施方式使用了像“确定”和“使用”的术语来描述在计算系统中的计算机操作。这些术语是对由计算机执行的操作的高级抽象,不应和由人类执行的动作相混淆。对应于这些术语的实际的计算机操作根据实施方式的不同而发生变化。

[0053] II. 示例网络环境。

[0054] 图2a和图2b示出了示例网络环境(201,202),该示例网络环境(201,202)包括视频编码器(220)和视频解码器(270)。编码器(220)和解码器(270)采用适当的通信协议通过网络(250)连接。网络(250)可以包括因特网或其他计算机网络。

[0055] 在图2a所示的网络环境(201)中,每个实时通信(“RTC”)工具(210)均包括编码器(220)和解码器(270),以用于双向通信。给定编码器(220)可以产生符合SMPTE421M标准、ISO/IEC 14496-10标准(也称为H.264/AVC)、H.265/HEVC标准、其他标准、或专有格式的输出,而相应的解码器(270)从编码器(220)中接收编码数据。双向通信可以是视频会议、视频电话、或其他两方通信场景的一部分。虽然图2a中的网络环境(201)包括两个实时通信工具(210),但是,网络环境(201)却可以包括三个或更多个参与多方通信的实时通信工具(210)。

[0056] 实时通信工具(210)通过编码器(220)对编码进行管理。图4示出了可以包括在实时通信工具(210)中的示例编码器系统(400)。可选地,该实时通信工具(210)使用另一种编码器系统。实时通讯工具(210)还通过解码器(270)对解码进行管理。图5示出了可以包括在实时通信工具(210)中的示例解码器系统(500)。可选地,该实时通信工具(210)使用另一种解码器系统。

[0057] 在图2b所示的网络环境(202)中,编码工具(212)包括编码器(220),该编码器

(220)可以对视频进行编码以传输到包括解码器(270)在内的多个播放工具(214)。可以为视频监视系统、网络摄像机监视系统、远程桌面会议演示、或视频被编码并从一个位置发送到一个或多个其他位置的其他场景提供单向通信。虽然图2b中的网络环境(201)包括两个播放工具(214),但是,网络环境(202)可以包括更多或更少的播放工具(214)。一般情况下,播放工具(214)与编码工具(212)进行通信,以确定播放工具(214)要接收的视频流。播放工具(214)接收流,缓冲所接收的编码数据达一段合适的时间,并开始解码和播放。

[0058] 图4示出了可以包括在解码工具(212)中的示例编码器系统(400)。可选地,该编码工具(212)使用另一种编码器系统。编码工具(212)还可以包括用于管理与一个或多个播放工具(214)的连接的服务器侧控制器逻辑。图5示出了可以包括在播放工具(214)中的示例解码器系统(500)。可选地,该播放工具(214)使用另一种解码器系统。播放工具(214)还可以包括用于管理和编码工具(212)的连接的客户端侧控制器逻辑。

[0059] III. 示例帧封装/解封系统。

[0060] 图3是与其结合可以实现一些所描述的实施例的通用帧封装/解封系统(300)的框图,。

[0061] 系统(300)包括是视频源(310),其产生诸如4:4:4格式之类的较高分辨率色度采样格式的源帧(311)。该视频源(310)可以是摄像机、调谐卡、存储介质、或其它数字视频源。

[0062] 帧封装器(315)对较高分辨率色度采样格式的帧(311)进行重新排列,以产生诸如4:2:0格式之类的较低分辨率色度采样格式的源帧(316)。帧封装的示例途径如下所描述。帧封装器(315)能够用信号通知指出是否以及如何执行帧封装的元数据(317),以便在解码后由帧解封器(385)使用。用信号通知帧封装排列元数据的示例途径如下所描述。

[0063] 编码器(340)对较低分辨率色度采样格式的帧(316)进行编码。以下参照图4和图6对示例编码器进行描述。编码器(340)通过通道(350)输出经编码的数据(341),该通道(350)代表存储器、通信连接或用于输出的另一个通道。

[0064] 解码器(360)接收经编码的数据(341)并对较低分辨率色度采样格式的帧(316)进行解码。以下参照图5和图7对示例解码器进行描述。解码器输出较低分辨率色度采样格式的重构帧(381)。

[0065] 帧解封器(385)对较低分辨率色度采样格式的重构帧(381)进行任选地重新排列,从而重构较高分辨率色度采样格式的帧(386)。以下对帧解封的示例途径进行描述。帧解封器(385)能够接收指出是否以及如何执行帧封装的元数据(317),并使用这样的元数据(317)指导解封操作。帧解封器(385)将较高分辨率色度采样格式的重构帧输出到输出目的地(390)。

[0066] IV. 示例编码器系统。

[0067] 图4是与其结合可以实现一些所描述的实施例的示例编码器系统(400)的框图。编码器系统(400)可以是能够在多种编码模式(诸如用于实时通信的低延迟编码模式、转码模式、以及用于来自文件或流的媒体播放的常规编码模式)的任一种中操作的通用编码工具,或者可以是适于一个这样的编码模式的特殊用途编码工具。编码器系统(400)可以作为操作系统模块、作为应用程序库的一部分、或作为独立的应用程序而被实现。总体来说,编码器系统(400)从视频源(410)接收一系列源视频帧(411)(较高分辨率色度采样格式,诸如4:4:4格式),执行帧封装获得诸如4:2:0格式之类的较低分辨率色度采样格式,对较低分辨率

色度采样格式的帧进行编码,并产生编码数据作为对通道(490)的输出。

[0068] 视频源(410)可以是摄像机、调谐卡、存储介质、或其它数字视频源。视频源(410)产生一系列的速率例如每秒30帧的视频帧。如本文所使用的,术语“帧”一般是指经编码或重构的源图像数据。对于逐行扫描视频,帧是一种逐行扫描视频帧。对于隔行视频,在示例实施例中,在编码之前对隔行视频帧进行去隔行。可选地,两个互补的隔行视频字段被编码为一个隔行视频帧或单独字段。除了指示逐行扫描视频帧,术语“帧”可指示单个非成对视频字段、视频字段的互补对、代表给定时间视频对象的视频对象平面、或较大图像中的感兴趣区域。视频对象平面或区域可以是较大图像的一部分,该图像包括场景的多个对象或区域。色彩空间从捕获格式(例如,RGB格式)转换后,源帧(411)为较高分辨率色度采样格式,诸如4:4:4格式。

[0069] 帧封装器(415)对较高分辨率色度采样格式的帧(411)进行重新排列,以产生诸如4:2:0格式之类的较低分辨率色度采样格式的源帧(416)。帧封装的示例途径如下所描述。帧封装器(415)能够用信号通知指出是否以及如何执行帧封装的元数据(未示出),以便在解码后由帧解封器使用。用信号通知帧封装排列元数据的示例途径如下所描述。帧封装器(415)能够例如如下文所描述地执行预处理操作。

[0070] 到达源帧(416)被存储在源帧临时内存存储区域(420)中,该区域包括多个帧缓冲器存储区域(421,422,...,42n)。帧缓冲器(421,422等)将一个源帧保持在源帧存储区域(420)中。在一个或多个源帧(416)已被存储在帧缓冲器(421,422等)中之后,帧选择器(430)周期性地选择来自源帧存储区域(420)的单个源帧。帧选择器(430)选择用于输入到编码器(440)中的帧的顺序可能和视频源(410)产生帧的顺序不同,例如,选择的帧可能在顺序前面以促进暂时反向预测。

[0071] 帧封装器(415)和帧存储区域(420)的顺序可以交换。在编码器(440)之前,编码器系统(400)可以包括另一个预处理器(未示出),该预处理器在编码之前对所选择的帧(431)执行预处理(例如,滤波)。

[0072] 编码器(440)对所选择的(较低分辨率色度采样格式的)帧(431)进行编码以产生编码帧(441),并同时产生内存管理控制操作(“MMCO”)信号(442)或参考图片集(“PS”)信息。如果当前帧并非已被编码的第1帧,那么对其执行编码过程时,编码器(440)可以使用一个或多个已被保存在解码帧临时内存存储区域(460)中的之前编码/解码的帧(469)。这种存储的解码帧(469)被用作参考帧,以用于对当前源帧(431)的内容进行的帧间预测。一般而言,编码器(440)包括诸如执行运动估计和补偿、频率变换、量化和熵编码之类的编码任务的多个编码模块。由编码器(440)执行的确切操作可能根据压缩格式而发生变化。输出编码数据的格式可以是Windows Media Video格式、VC-1格式、MPEG-x格式(例如,MPEG-1、MPEG-2、或MPEG-4)、H.26x格式(例如,H.261、H.262、H.263、H.264)、HEVC格式或其他格式。一般情况下,编码器(440)适于对较低分辨率色度采样格式的帧进行编码。

[0073] 例如,在编码器(440)内,帧间编码的预测帧根据依据参考帧的预测来表示。运动估计器相对于一个或多个参考帧(469)对源帧(441)的样本组的运动进行估计。一组样本可以是宏块、子宏块或子宏块分区(如在H.264标准中),或者可以是编码树单元或预测单元(如在HEVC标准中)。一般情况下,如本文所使用的,术语“块”表示一组样本,其可以是单个二维(“2D”)阵列或多个2D阵列(例如,一个用于亮度分量的阵列和两个用于色度分量的阵

列)。在使用多个参考帧时,该多个参考帧可以来自不同的时间方向或相同的时间方向。运动估计器输出经熵编码的诸如运动向量信息之类的运动信息。运动补偿器将运动向量应用到参考帧,以确定运动补偿的预测值。编码器确定块的运动补偿预测值和相应的原始值之间的差异(如果有的话)。使用频率变换、量化和熵编码对这些预测残余值(即残差,残余的值)进行进一步编码。同样,对于帧内预测,编码器(440)可以确定块的帧内预测值,确定预测残余值,并对该预测残余值进行编码。编码器(440)的熵编码器对量化的变换系数值、以及某些边信息(例如,运动向量信息、QP值、模式决策、参数选择)进行压缩。典型的熵编码技术包括EXP-Golomb编码、算术编码、差分编码、哈夫曼编码、行程长度编码、可变长度到可变长度(“V2V”)编码、可变长度到固定长度(“V2F”)编码、LZ编码、词典编码、概率间隔分区熵编码(“PIPE”)、以及上述的组合。熵编码器可以针对不同类型的信息使用不同的编码技术,并且可以在特定编码技术内从多个代码表中进行选择。

[0074] 编码帧(441)和MMCO/RPS信息(442)由解码过程模拟器(450)进行处理。解码过程模拟器(450)实现解码器的一些功能,例如,重构编码器(440)在运动估计和补偿中所使用的参考帧的解码任务。解码过程模拟器(450)使用MMCO/RPS信息(442)来确定给定的编码帧(441)是否需要被存储,以在待编码的后续帧的帧间预测中用作参照帧。如果MMCO/RPS信息(442)指出需要存储编码帧(441),那么,解码过程模拟器(450)对解码过程进行模拟,该解码过程将由接收编码帧(441)并产生相应的解码帧(451)的解码器进行。在这种情况下,当编码器(440)已使用已被存储在解码帧存储区域(460)中的解码帧(469)时,解码过程模拟器(450)也把来自存储区域(460)的(多个)解码帧(469)用作解码过程的一部分。

[0075] 解码帧临时内存存储区域(460)包括多个帧缓冲器存储区域(461,462,...,46n)。解码过程模拟器(450)使用MMCO/RPS信息(442)管理存储区域(460)的内容,从而识别具有编码器(440)不再需要其用作参考帧的帧的任何帧缓冲器(461,462等)。模拟解码过程之后,解码过程模拟器(450)在已经以这种方式识别的帧缓冲器(461,462等)中存储新解码的帧(451)。

[0076] 编码帧(441)和MMCO/RPS信息(442)还在临时编码数据区域(470)中被缓存。聚集在编码数据区域(470)的编码数据也可以包括与编码视频数据(例如,正如在一个或多个SEI消息(诸如帧封装排列SEI消息)或视频可用性信息(“VUI”)消息中的一个或多个参数)相关的媒体元数据。

[0077] 来自临时编码数据区域(470)的聚集数据(471)由通道编码器(480)进行处理。通道编码器(480)可以封装聚集数据以作为媒体流(例如,根据ISO IEC 14496-12等媒体容器格式)进行传输,在这种情况下,通道编码器(480)可以添加语法元素作为媒体传输流的语法的一部分。或者,通道编码器(480)可以对聚集数据进行组织以作为文件(例如,诸如根据ISO/IEC 14496-12之类的媒体容器格式)进行存储,在这种情况下,通道编码器(480)可以添加语法元素作为媒体存储文件的语法的一部分。或者,更一般地,通道编码器(480)可以实现一个或多个媒体系统复用协议或传输协议,在这种情况下,通道编码器(480)可以添加语法元素作为(多个)协议语法的一部分。这种用于媒体传输流、媒体存储流、复用协议或传送协议的语法元素可以包括帧封装排列元数据。通道编码器(480)向通道(490)提供输出,该通道代表存储器、通信连接或用于输出的另一个通道。

[0078] V. 示例解码器系统。

[0079] 图5是与其结合可以实现一些所描述的实施例的示例解码器系统(500)的框图。解码器系统(500)可以是能够以多种解码模式(诸如用于实时通信的低延迟解码模式和用于来自文件或流的媒体播放的常规解码模式)的任一种来操作的通用解码工具,或者可以是适于一种这样的解码模式的特殊用途解码工具。解码器系统(500)可以作为操作系统模块、作为应用程序库的一部分、或作为独立的应用程序而被实现。总体而言,解码器系统(500)接收来自通道(510)的编码数据,对诸如4:2:0格式之类的较低分辨率色度采样格式的帧进行解码,可选地执行从较低分辨率色度采样格式到诸如4:4:4格式之类的较高分辨率色度采样格式的帧解封,并产生作为输出的(较高分辨率色度采样格式的)重构帧用于输出目的地(590)。

[0080] 解码器系统(500)包括通道(510),其可以代表存储器、通信连接或用于作为输入的编码数据的另一个通道。通道(510)产生已进行通道编码的编码数据。通道解码器(520)可以对编码数据进行处理。例如,通道解码器(520)对已聚集的数据进行解封以作为媒体流(例如,根据诸如ISO IEC 14496-12之类的媒体容器格式)用于传输,在这种情况下,通道解码器(520)可以对作为媒体传输流的语法的一部分而添加的语法元素进行解析。或者,通道解码器(520)对已聚集作为文件(例如,根据诸如ISO IEC 14496-12之类的媒体容器格式)进行存储的编码视频数据进行分离以,在这种情况下,通道解码器(520)可以对作为媒体存储文件的语法的一部分而添加的语法元素进行解析。或者,更一般地,通道解码器(480)可以实现一个或多个媒体系统复用分解协议或传输协议,在这种情况下,通道解码器(520)可以对作为(多个)协议的语法的一部分而添加的语法元素进行解析。这种用于媒体传输流、媒体存储流、复用协议或传输协议的语法元素可以包括帧封装排列元数据。

[0081] 从通道解码器(520)输出的编码数据(521)被存储在临时编码数据区域(530),直到已经接收到足够量的这些数据为止。编码数据(521)包括编码帧(531)(较低分辨率色度采样格式)和MMCO/RPS信息(532)。编码数据区域(530)中的编码数据(521)还可以包括与编码的视频数据(例如,正如在一个或多个SEI消息(诸如帧封装排列SEI消息)或VUI消息中的一个或多个参数)相关的媒体元数据。一般情况下,编码数据区域(530)暂时存储编码数据(521),直至该编码数据(521)被解码器(550)使用为止。此时,从编码数据区域(530)向解码器(550)传送用于编码帧(531)的编码数据和MMCO/RPS信息(532)。随着解码的继续,新的编码数据被添加到编码数据区域(530),并且保留在编码数据区域(530)的最早的编码数据被传输到解码器(550)。

[0082] 解码器(550)周期性地对编码帧(531)进行解码,以产生较低分辨率的色度采样格式的相应的解码帧(551)。在适当的情况下,当执行其解码过程时,解码器(550)可以将一个或多个先前解码的帧(569)用作帧间预测的参考帧。解码器(550)从解码帧临时内存存储区域(560)读取该先前解码的帧(569)。一般情况下,解码器(550)包括执行诸如熵解码、反量化、反频率变换和运动补偿之类的解码任务的多个解码模块。由解码器(550)执行的确切操作可以根据压缩格式而发生变化。一般情况下,解码器(550)适于对较低分辨率色度采样格式的帧进行解码。

[0083] 例如,解码器(550)接收编码数据用于压缩帧或帧序列,并产生包括较低分辨率色度采样格式的解码帧(551)的输出。在解码器(550)中,缓冲器接收编码数据用于压缩帧,并将所接收的编码数据提供给熵解码器。熵解码器对熵编码的量化数据以及熵编码的边信息

进行熵解码,通常在编码器中执行反熵编码。运动补偿器将运动信息应用于一个或多个参考帧来形成被重构的帧的块(例如,宏块、子宏块、子宏块分区、编码树单元、预测单元、或其部分)的运动补偿预测。帧内预测模块可以根据邻近的、先前重构的样本值在空间上对当前块的样本值进行预测。解码器(550)还对预测残差进行重构。反量化器对熵解码的数据进行反量化。反频率变换器将经量化的频域数据转换成空间域的信息。对于预测的帧,解码器(550)将重构的预测残差与运动补偿预测相结合以形成重构帧。解码器(550)可以以类似方式将预测残差与来自帧内预测的空间预测相结合。视频解码器(550)中的运动补偿环路包括自适应去块滤波器,以平滑解码帧(551)中的跨越块边界行和/或列的不连续性。

[0084] 解码帧临时内存存储区域(560)包括多个帧缓冲器存储区域(561,562,...,56n)。解码帧存储区域(560)是DPB的示例。解码器(550)使用MMC0/RPS信息(532)来识别可存储较低分辨率色度采样格式的解码帧(551)的帧缓冲器(561,562等)。解码器(550)在该帧缓冲器中存储解码帧(551)。

[0085] 输出定序器(580)使用MMC0 RPS信息(532)来识别按输出顺序产生的下一帧在解码帧存储区域(560)中何时可用。当按输出顺序产生的较低分辨率色度采样格式的下一帧(581)在解码帧存储区域(560)中可用时,下一帧由输出定序器(580)读出,并被输出到(a)输出目的地(590)(例如,显示器)用于显示较低分辨率色度采样格式的帧,或(b)帧解封器(585)。一般情况下,帧由输出定序器(580)从解码帧存储区域(560)输出的顺序可能与帧被解码器(550)解码的顺序不同。

[0086] 帧解封器(585)对较低分辨率色度采样格式的帧(581)进行重新排列,以产生诸如4:2:0格式之类的较高分辨率色度采样格式的输出帧(586)。以下对帧解封的示例途径进行描述。帧解封器(585)能够使用指示是否以及如何执行帧封装的元数据(未示出),来指导帧解封操作。帧解封器(585)例如能够如下文所描述地执行后处理操作。

[0087] VI. 示例视频编码器。

[0088] 图6是与其结合可以实现一些所描述的实施例的通用视频编码器(600)的框图。编码器(600)接收包括当前帧(605)在内的一系列诸如4:2:0格式之类的较低分辨率色度采样格式的视频帧,并产生作为输出的编码数据(695)。

[0089] 编码器(600)基于块,并使用取决于实施方式的宏块格式。可以在不同阶段(例如,在频率变换阶段和熵编码阶段)对块进一步细分。例如,帧可以被分成 16×16 个宏块,其可以进而被分为 8×8 个块和更小的像素值子块,以用于编码和解码。

[0090] 编码器系统(600)对预测的帧和帧内编码的帧进行压缩。为了便于介绍,图6示出了通过编码器(600)用于帧内编码的“帧内路径”和用于帧间编码的“帧间路径”。编码器(600)的多个组件同时用于帧内编码和帧间编码。由上述组件执行的确切操作可能根据正在压缩的信息类型的不同而发生变化。

[0091] 如果当前帧(605)为预测帧,运动估计器(610)则相对于一个或多个参考帧对当前帧(605)的块(例如,宏块、子宏块、子宏块分区、编码树单元、预测单元、或其部分)的运动进行估计。帧存储器(620)缓存一个或多个重构的先前帧(625)以用作参考帧。在使用多个参考帧时,该多个参考帧可以来自不同的时间方向或相同的时间方向。运动估计器(610)输出作为边信息的运动信息(615),诸如差分运动向量信息。

[0092] 在形成运动补偿的当前帧(635)时,运动补偿器(630)将重构的运动向量应用到重

构的(多个)参考帧(625)。运动补偿的当前帧(635)的块和原始当前帧(605)的相应部分之间的差(如果有的话)为用于块的预测残差(645)。在当前帧的后期重构期间,重构的预测残差被添加到运动补偿的当前帧(635),以获得更接近原始当前帧(605)的重构帧。然而,在有损压缩中,一些信息仍从原始当前帧(605)中丢失。帧内路径可以包括帧内预测模块(未示出),其根据相邻的、先前重构的像素值在空间上对当前块的像素值进行预测。

[0093] 频率变换器(660)将空间域视频信息转换成频域(即,频谱,变换)数据。对于基于块的视频帧,频率变换器(660)将离散余弦变换、其整数近似、或另一种类型的前向块变换,应用到像素值数据或预测残差数据的块,从而产生频率变换系数块。然后,量化器(670)对变换系数进行量化。例如,量化器(670)将非均匀标量量化应用到频域数据,其中步长在逐帧的基础、逐宏块的基础或其它基础上发生变化。

[0094] 当需要将当前帧的重构版本用于后续运动估计/补偿时,反量化器(676)对量化的频率系数数据进行反量化。反频率变换器(666)执行反频率变换,从而产生重构预测残差或像素值的块。对于预测的帧,编码器(600)将重构的预测残差(645)与运动补偿预测(635)相结合以形成重构帧(605)。(虽然在图6中未示出,但是,在帧内路径中,编码器(600)可以将预测残差与来自帧内预测的空间预测相结合。)帧存储器(620)缓冲重构的当前帧以在随后的运动补偿预测中使用。

[0095] 编码器(600)中的运动补偿环路包括帧存储器(620)之前或之后的自适应环内去块滤波器(610)。解码器(600)将环内滤波应用到重构帧,以自适应地平滑帧中跨边界的不连续性。自适应环内去块滤波器(610)可针对某些类型的内容而被禁用。例如,在使用主要视图及辅助视图的帧封装途径中,自适应环内去块滤波器(610)可以在对辅助视图(包括不是主要视图的一部分的剩余色度信息)进行编码时被禁用,从而不会引入诸如模糊之类的伪影。

[0096] 熵编码器(680)对量化器(670)的输出以及运动信息(615)和某些边信息(例如,QP值)进行压缩。熵编码器(680)向缓冲器(690)提供编码数据(695),该缓冲器(690)将编码数据复用到输出比特流。

[0097] 控制器(未示出)接收来自编码器的各个模块的输入。控制器在编码期间对中间结果进行评估,例如,设置QP值并执行速率-失真分析。控制器与其他模块合作以便在编码期间设置并更改编码参数。特别是,控制器可以改变QP值和其他控制参数,以便在编码期间控制亮度分量和色度分量的量化。例如,与给定帧的色度内容相比,控制器可以改变QP值以将更多比特用于该帧(在帧封装途径中,其可能是主要视图或辅助视图)的亮度内容。或者,在采用主要视图及辅助视图的帧封装途径中,与辅助视图(包括剩余色度信息)相比,控制器可以改变QP值以将更多的比特用于主要视图(包括亮度和子采样的色度分量)。

[0098] 在一些帧封装途径中,即使在来自较高分辨率色度采样格式的帧的色度信息已被封装成较低分辨率色度采样格式的待编码帧之后,编码器仍可以通过多种方式利用色度分量的样本值之间的几何对应。术语几何对应表示(1)从较低分辨率色度采样格式构建的帧的(标称)亮度分量的位置处的色度信息和(2)较低分辨率色度采样格式的帧的色度分量的相应比例的(scaled)位置处的色度信息之间的关系。比例因数(scaling factor)应用在亮度和色度分量的位置之间。例如,对于4:2:0,比例因数在水平方向和垂直方向均为2,而对于4:2:2,比例因数在水平方向为2,在垂直方向为1。

[0099] 编码器可以使用几何对应,以便在对待编码帧的当前块进行编码时,通过首先对相邻块的最近结果进行评估来指导运动估计、QP选择、预测模式选择或其它从块到块的决策过程。或者,编码器可以使用几何对应,以便使用来自对封装成待编码的帧的“亮度”分量的高分辨率色度信息编码的结果,来指导用于封装成待编码帧的色度分量的高分辨率色度信息的这种决策过程。或者,更直接地,编码器可以使用几何对应来改善压缩性能,其中运动向量、预测模式、或用于封装成待编码的帧的“亮度”分量的高分辨率色度信息的其他决策,还用于封装成待编码的帧的色度分量的高分辨率色度信息。特别是,在本文所描述的一些途径(例如,以下途径2)中,当色度信息被封装到较低分辨率色度采样格式的辅助帧时,该辅助帧的标称亮度分量和该辅助帧的标称色度分量之间的空间对应和运动向量位移的关系被保留。辅助帧的Y、U和V分量中的相应的空间位置的样本值趋于一致,这对于诸如空间块大小划分和编码块模式信息或其他指示存在/不存在非零系数值的信息的联合编码之类的目的而言,非常有用。用于辅助帧的Y、U和V分量的相应部分的运动向量趋于一致(例如,在Y中的两个样本的垂直或水平位移对应于U和V中的一个样本的位移),这也有助于编码效率。

[0100] 根据实施方式和所需的压缩类型,可对编码器的模块进行添加、省略、分成多个模块、与其它模块组合、和/或使用相似模块替代。在替代实施例中,具有不同模块和/或其他模块配置的编码器执行一个或多个所描述的技术。编码器的具体实施例通常使用编码器(600)的变体或补充版本。编码器(600)内模块之间示出的关系表示编码器中的一般信息流;为简单起见,其它关系并未示出。

[0101] VII. 示例视频解码器。

[0102] 图7是与其结合可以实现若干所描述的实施例的通用解码器(700)的框图。解码器(700)接收编码数据(795)用于压缩帧或一系列帧,并产生包括诸如4:2:0格式之类的较低分辨率色度采样格式的重构帧(705)的输出。为了便于介绍,图7示出了通过解码器(700)用于帧内解码的“帧内路径”和用于帧间编码的“帧间路径”。解码器(700)的多个组件同时用于帧内解码和帧间解码。由那些组件执行的确切操作可以根据正在解压缩的信息类型的不同而发生变化。

[0103] 缓冲器(790)接收用于压缩帧的编码数据(795),并使解析器/熵解码器(780)可以得到所接收的编码数据。解析器/熵解码器(780)对熵编码的量化数据以及熵编码的边信息进行熵解码,通常在编码器中应用反熵编码。

[0104] 运动补偿器(730)将运动信息(715)应用于一个或多个参考帧(725)来形成被重构的帧(705)的块(例如,宏块、子宏块、子宏块分区、编码树单元、预测单元、或其部分)的运动补偿预测(735)。帧存储器(720)存储一个或多个先前重构的帧(625)以用作参考帧。

[0105] 帧内路径可以包括帧内预测模块(未示出),该帧内预测模块根据相邻的、先前重构的像素值在空间上对当前块的像素值进行预测。在帧间路径中,解码器(700)对预测残差进行重构。反量化器(770)对熵解码的数据进行反量化。反频率变换器(760)将经量化的频域数据转换成空间域信息。例如,反频率变换器(760)将反块变换应用到频率变换系数,从而产生像素值数据或预测残差数据。反频率变换可以是反离散余弦变换、其整数近似、或另一种类型的反频率变换。

[0106] 对于预测的帧,解码器(700)将重构的预测残差(745)与运动补偿预测(735)相结

合以形成重构帧(705)。(虽然图7中未示出,但是,在帧内路径中,解码器(700)可以将预测残差与来自帧内预测的空间预测向结合。)解码器(700)中的运动补偿环路包括帧存储器(720)之前或之后的自适应环内去块滤波器(710)。解码器(700)将环内滤波应用于重构帧,以便自适应地平滑帧中跨边界的不连续性。当自适应环内去块滤波器(710)在编码期间被禁用时,其可以针对一些类型的内容而被禁用。例如,在采用主要视图及辅助视图的帧封装途径中,自适应环内去块滤波器(710)可以在对辅助视图(包括并非主要视图的一部分的剩余色度信息)解码时被禁用。

[0107] 在图7中,解码器(700)还包括后处理去块滤波器(708)。该后处理去块滤波器(708)视情况可对重构帧中的不连续性进行平滑。其它滤波(诸如去环滤波)也可以作为后处理滤波的一部分而被应用。通常情况下,要被稍后帧解封的重构帧绕过后处理去块滤波器(708)。

[0108] 根据实施方式和所需的解压缩类型,可以对解码器的模块进行添加、省略、分成多个模块、与其它模块组合、和/或使用相似模块替代。在替代实施例中,具有不同模块和/或其他模块配置的解码器执行一个或多个所描述的技术。解码器的具体实施例通常使用解码器(700)的变体或补充版本。解码器(700)内模块之间示出的关系表示解码器中的一般信息流;为简单起见,其它关系并未示出。

[0109] VIII. 用于较高分辨率色度采样格式的帧封装/解封。

[0110] 本节描述了各种将较高分辨率色度采样格式的帧封装成较低分辨率色度采样格式的帧的途径。然后,可以使用专为较低分辨率色度采样格式而设计的编码器,对较低分辨率色度采样格式的帧进行编码。(使用专为较低分辨率色度采样格式而设计的解码器)解码后,可以输出较低分辨率色度采样格式的帧,以用于进一步的处理和显示。或者,这种解码之后,可以通过帧解封重新获得更高分辨率色度采样格式的帧,以用于输出和显示。

[0111] A. 用于YUV 4:4:4视频的帧封装/解封途径。

[0112] 作为一个具体示例,在编码/解码使用4:2:0格式时,本文所描述的各种途径可以用来保存4:4:4格式的帧的色度信息。在这些途径中,例如,YUV 4:4:4帧被封装成两个YUV4:2:0帧。典型的4:4:4帧每4个像素位置包含12个样本值,而4:2:0帧每4个像素位置仅包含6个样本值。所以,包含在4:4:4帧中的所有样本值均可被封装成两个4:2:0帧。

[0113] 1. 途径1。

[0114] 在途径1中,使用空间分割将YUV 4:4:4帧封装成两个YUV 4:2:0帧。图8示出了这种使用了YUV 4:4:4帧的空间分割的帧封装途径(800)。

[0115] Y_{444} 平面、 U_{444} 平面和 V_{444} 平面是YUV 4:4:4帧(801)的三个分量平面。每个平面的分辨率具有宽度W和高度H。为了便于描述本文所使用的示例,W和H均可被4除,而不暗示其对该途径构成限制。将YUV 4:4:4帧封装成两个YUV 4:2:0帧的途径(800)对YUV 4:4:4帧进行拆分,如图8所示。使用空间分割,将YUV 4:4:4帧(801)的 U_{444} 平面分割成下半部 $H2-U_{444}$ 和两个四分之一上部 $Q1-U_{444}$ 和 $Q2-U_{444}$ 。使用空间分割,将YUV4:4:4帧(801)的 V_{444} 平面分割成下半部 $H2-V_{444}$ 和两个四分之一上部 $Q1-V_{444}$ 和 $Q2-V_{444}$ 。

[0116] 然后,将YUV 4:4:4帧(801)所分割的平面重组为一个或多个YUV 4:2:0帧。用于YUV 4:4:4帧的 Y_{444} 平面变成YUV 4:2:0格式的第1帧(802)的亮度分量平面。 U_{444} 平面和 V_{444} 平面的下半部变成YUV 4:2:0格式的第2帧(803)的亮度分量平面。如图8所示, U_{444} 平面和

V_{444} 平面的四分之一顶部变成YUV 4:2:0格式的第1帧(802)和第2帧(803)的色度分量平面。

[0117] YUV 4:2:0格式的第1帧(802)和第2帧(803)可以被组织为单独帧(在图8中由黑线分离出来)。或者,YUV 4:2:0格式的第1帧(802)和第2帧(803)可以被组织为高度为 $2xH$ 的单个帧(忽略图8中的黑线)。或者,YUV 4:2:0格式的第1帧(802)和第2帧(803)可以被组织为宽度为 $2xW$ 的单个帧。或者,使用H.264/AVC标准或HEVC标准中为frame_packing_arrangement_type定义的任意方法,可以将YUV 4:2:0格式的第1帧(802)和第2帧(803)组织为单个帧或多个帧。

[0118] 虽然这种类型的帧封装可行,但是,它不会实现两个YUV 4:2:0帧的每一个内的Y、U和V分量之间的几何对应。特别是,对于YUV 4:2:0格式的第2帧(803),亮度分量和色度分量之间通常没有几何对应。本文所描述的其他封装途径通常获得更好的几何对应。

[0119] 可选地,途径1能够以诸如4:4:4、4:2:2、4:2:0之类的采样率作为色度采样格式用于诸如RGB、GBR之类的色彩空间。

[0120] 2. 途径2。

[0121] 在途径2中,YUV 4:4:4帧被封装成两个YUV 4:2:0帧,同时保持几何对应以用于YUV 4:4:4帧的色度信息。YUV 4:2:0帧在其Y、U和V分量之间具有良好的几何对应,可以被更好地压缩,这是因为YUV 4:2:0帧适于对YUV 4:2:0进行编码的典型编码器所期望的模型。

[0122] 尽管颜色分量的分辨率较低,但是,还可以完成封装使得两个YUV 4:2:0帧中的一个代表正由4:4:4帧代表的完整场景。这提供了解码选项。不能执行帧解封或选择不执行帧解封的解码器,可以只采用代表该场景的YUV 4:2:0帧的重构版本,并将其直接馈送到显示器。

[0123] 图9示出了帧封装的示例途径(900),该帧封装符合这些设计约束。在该途径(900)中,将YUV 4:4:4帧(801)封装成两个YUV 4:2:0帧(902,903)。第1帧(902)提供了YUV 4:2:0格式的“主要视图”——由YUV 4:4:4帧(801)代表的完整场景的较低色度分辨率版本。第2帧(903)提供了YUV 4:2:0格式的“辅助视图”并包含剩余色度信息。

[0124] 在图9中,区域 $B1 \cdots B9$ 为YUV 4:2:0格式的各个帧(902,903)内的不同区域。YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的奇数行的样本值被分配到区域 $B4$ 和 $B5$,而YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的偶数行的样本值则在区域 $B2$ 、 $B3$ 和 $B6 \cdots B9$ 之间分布。具体而言,YUV 4:4:4帧(801)的 Y_{444} 平面、 U_{444} 平面和 V_{444} 平面的样本值映射到区域 $B1 \cdots B9$,如下所示。

[0125] • 对于区域 $B1$, $Y_{420}^{main}(x, y) = Y_{444}(x, y)$,其中 (x, y) 的范围为 $[0, W-1] \times [0, H-1]$ 。

[0126] • 对于区域 $B2$, $U_{420}^{main}(x, y) = U_{444}(2x, 2y)$,其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/2-1]$ 。

[0127] • 对于区域 $B3$, $V_{420}^{main}(x, y) = V_{444}(2x, 2y)$,其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/2-1]$ 。

[0128] • 对于区域 $B4$, $Y_{420}^{aux}(x, y) = U_{444}(x, 2y + 1)$,其中 (x, y) 的范围为 $[0, W-1] \times [0, H/2-1]$ 。

[0129] • 对于区域B5, $Y_{420}^{aux}\left(x, \frac{H}{2} + y\right) = V_{444}(x, 2y + 1)$, 其中 (x, y) 的范围为 $[0, W-1] \times [0, H/2-1]$ 。

[0130] • 对于区域B6, $U_{420}^{aux}(x, y) = U_{444}(2x + 1, 4y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/4-1]$ 。

[0131] • 对于区域B7, $U_{420}^{aux}\left(x, \frac{H}{4} + y\right) = V_{444}(2x + 1, 4y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/4-1]$ 。

[0132] • 对于区域B8, $V_{420}^{aux}(x, y) = U_{444}(2x + 1, 4y + 2)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/4-1]$ 。

[0133] • 对于区域B9, $V_{420}^{aux}\left(x, \frac{H}{4} + y\right) = V_{444}(2x + 1, 4y + 2)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/4-1]$ 。

[0134] 可选地, YUV 4:4:4帧 (801) 的 Y_{444} 平面、 U_{444} 平面和 V_{444} 平面的样本值能够以不同的方式分配到区域B1...B9。例如, YUV 4:4:4帧 (801) 的 U_{444} 平面和 V_{444} 平面的偶数行的样本值被分配到区域B4和B5, 而YUV 4:4:4帧 (801) 的 U_{444} 平面和 V_{444} 平面的奇数行的样本值则在区域B2、B3和B6...B9之间分布。或者, 作为另一示例, 来自YUV 4:4:4帧的原始U平面的数据能够被布置在辅助YUV 4:2:0帧的U平面之中, 而来自YUV 4:4:4帧的原始V平面的数据能够被布置在辅助YUV 4:2:0帧的V平面之中。在该示例中, 与图9相比, 得自 $V_{444}(2x+1, 4y)$ 的、在上述等式被分配到区域B7的样本值, 相反能够被分配到区域B8, 而得自 $U_{444}(2x+1, 4y+2)$ 的、在上述等式中被分配到区域B8的样本值, 相反能够被分配到区域B7。或者, 来自 U_{444} 的相同样本值能够被复制到B6和B7的单个区域, 而无需将每个第2行分离, 并且来自 V_{444} 的相同样本值能够被复制到B8和B9的单个区域, 而无需将每个第2行分离。无论哪种方式, 辅助YUV 4:2:0帧的U平面 (或V平面) 是从YUV 4:4:4帧的U平面 (或V平面) 构建的, 没有混合来自不同原始U平面和V平面的内容。(与此相反, 在图9的示例中, 辅助YUV 4:2:0帧的U平面 (或V平面) 具有来自辅助YUV 4:4:4帧的U分量和V分量的数据混合。辅助YUV 4:2:0帧的U平面 (或V平面) 的上半部包含来自原始U平面的数据, 而下半部包含来自原始V平面的数据。) YUV 4:2:0格式的第1帧 (902) 和第2帧 (903) 能够被组织为单独帧 (在图9中由黑线分离出来)。或者, YUV 4:2:0格式的第1帧 (902) 和第2帧 (903) 能够被组织为高度为 $2xH$ 的单个帧 (忽略图9中的黑线)。或者, YUV 4:2:0格式的第1帧 (902) 和第2帧 (903) 能够被组织为宽度为 $2xW$ 的单个帧。或者, 使用H.264/AVC标准或HEVC标准中为frame_packing_arrangement_type定义的任何方法, 能够将YUV 4:2:0格式的第1帧 (902) 和第2帧 (903) 组织为单个帧。

[0135] 图10示出了根据图9的途径 (900) 封装的示例帧 (1000)。图10示出了包括 Y_{444} 平面、 U_{444} 平面和 V_{444} 平面的YUV 4:4:4帧 (1001)。

[0136] 帧封装之后, 主要视图 (1002) (第1个YUV 4:2:0帧) 是相当于原始YUV 4:4:4帧 (1001) 的YUV 4:2:0。如果既不支持YUV 4:4:4又认为YUV 4:4:4不必要, 解码系统能够简单地显示主要视图 (1002) 的重构版本。

[0137] 辅助视图 (1003) 包含YUV 4:4:4帧 (1001) 的色度信息。即使如此, 辅助视图 (1003) 适合YUV 4:2:0帧的内容模型, 并且非常适于采用典型YUV 4:2:0视频编码器的压缩。在帧

内,辅助视图(1003)展示出跨Y、U和V分量的几何对应。在帧间,辅助视图预期示出跨Y、U和V分量高度相关的运动。

[0138] 图11示出了符合这些设计约束的帧封装的另一示例途径(1100)。在该途径(1100)中,将YUV 4:4:4帧(801)封装成两个YUV 4:2:0帧(1102,1103)。与图9中的途径(900)很像,在图11中的途径(1100)中,第1帧(1102)提供了YUV 4:2:0格式的“主要视图”——由YUV 4:4:4帧(801)代表的完整场景的较低色度分辨率版本——而第2帧(1103)提供了YUV 4:2:0格式的“辅助视图”,并包含剩余色度信息。

[0139] 在图11中,区域A1…A9均为YUV 4:2:0格式的各个帧(1102,1103)内的不同区域。YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的奇数列的样本值被分配到区域A4和A5,而YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的偶数列的样本值则在区域A2、A3和A6…A9之间分布。具体而言,YUV 4:4:4帧(801)的 Y_{444} 平面、 U_{444} 平面和 V_{444} 平面的样本值映射到区域A1…A9,如下所示。

[0140] • 对于区域A1, $Y_{420}^{main}(x, y) = Y_{444}(x, y)$, 其中 (x, y) 的范围为 $[0, W-1] \times [0, H-1]$ 。

[0141] • 对于区域A2, $U_{420}^{main}(x, y) = U_{444}(2x, 2y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/2-1]$ 。

[0142] • 对于区域A3, $V_{420}^{main}(x, y) = V_{444}(2x, 2y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H/2-1]$ 。

[0143] • 对于区域A4, $Y_{420}^{aux}(x, y) = U_{444}(2x + 1, y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H-1]$ 。

[0144] • 对于区域A5, $Y_{420}^{aux}\left(\frac{W}{2} + x, y\right) = V_{444}(2x + 1, y)$, 其中 (x, y) 的范围为 $[0, W/2-1] \times [0, H-1]$ 。

[0145] • 对于区域A6, $U_{420}^{aux}(x, y) = U_{444}(4x, 2y + 1)$, 其中 (x, y) 的范围为 $[0, W/4-1] \times [0, H/2-1]$ 。

[0146] • 对于区域A7, $U_{420}^{aux}\left(\frac{W}{4} + x, y\right) = V_{444}(4x, 2y + 1)$, 其中 (x, y) 的范围为 $[0, W/4-1] \times [0, H/2-1]$ 。

[0147] • 对于区域A8, $V_{420}^{aux}(x, y) = U_{444}(4x + 2, 2y + 1)$, 其中 (x, y) 的范围为 $[0, W/4-1] \times [0, H/2-1]$ 。

[0148] • 对于区域A9, $V_{420}^{aux}\left(\frac{W}{4} + x, y\right) = V_{444}(4x + 2, 2y + 1)$, 其中 (x, y) 的范围为 $[0, W/4-1] \times [0, H/2-1]$ 。

[0149] 可选地,YUV 4:4:4帧(801)的 Y_{444} 平面、 U_{444} 平面和 V_{444} 平面的样本值能够以不同的方式分配到区域A1…A9。例如,YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的偶数列的样本值被分配到区域A4和A5,而YUV 4:4:4帧(801)的 U_{444} 平面和 V_{444} 平面的奇数列的样本值则在区域A2、A3和A6…A9之间分布。或者,作为另一示例,来自YUV 4:4:4帧的原始U平面的数据能够被布置在辅助YUV 4:2:0帧的U平面中,而来自YUV 4:4:4帧的原始V平面的数据能够被布置在辅助YUV 4:2:0帧的V平面中。在该示例中,与图11相比,得自 $V_{444}(4x, 2y+1)$ 的、在上述

等式中被分配到区域A7的样本值,相反被分配到区域A8,而得自 $U_{444}(4x+2, 2y+1)$ 的、在上述等式被分配到区域A8的样本值,相反被分配到区域A7。或者,来自 U_{444} 的同样样本值能够被复制到A6和A7的单个区域,而并不将每个第2列分离,并且来自 V_{444} 的同样样本值能够被复制到A8和A9的单个区域,而并不将每个第2列分离。无论哪种方式,辅助YUV 4:2:0帧的U平面(或V平面)从YUV 4:4:4帧的U平面(或V平面)构建,而不混合来自不同原始U平面和V平面的内容。

[0150] YUV 4:2:0格式的第1帧(1102)和第2帧(1103)能够被组织为单独帧(在图11中由黑线分离出来)。或者,YUV 4:2:0格式的第1帧(1102)和第2帧(1103)能够被组织为高度为 $2 \times H$ 的单个帧(忽略图11中的黑线)。或者,YUV 4:2:0格式的第1帧(1102)和第2帧(1103)能够被组织为宽度为 $2 \times W$ 的单个帧。或者,使用H.264/AVC标准或HEVC标准中为frame_packing_arrangement_type定义的任意方法,能够将YUV 4:2:0格式的第1帧(1102)和第2帧(1103)组织为单个帧。

[0151] 帧解封能够简单地反映(mirror)帧封装。分配到YUV 4:2:0格式的帧的区域的样本被分配回YUV 4:4:4格式的帧的色度分量中的原始位置。在一个实施方式中,例如,在帧解封期间,YUV 4:2:0格式的帧的区域B2...B9中的样本被分配到YUV 4:4:4格式的帧的重构色度分量 U'_{444} 和 V'_{444} ,如以下伪代码所示。

```

for( x = 0; x < (W >> 1); x++ ) {
    for( y = 0; y < (H >> 1); y++ ) {
         $U'_{444}(2x, 2y + 1) = Y''_{420}{}^{aux}(2x, y)$ 
         $V'_{444}(2x, 2y + 1) = Y''_{420}{}^{aux}(2x, (H \gg 1) + y)$ 
         $U'_{444}(2x + 1, 2y + 1) = Y''_{420}{}^{aux}(2x + 1, y)$ 
         $V'_{444}(2x + 1, 2y + 1) = Y''_{420}{}^{aux}(2x + 1, (H \gg 1) + y)$ 
        if( y % 2 == 0 ) {
             $U'_{444}(2x + 1, 2y) = U''_{420}{}^{aux}(x, y \gg 1)$ 
             $V'_{444}(2x + 1, 2y) = U''_{420}{}^{aux}(x, (H \gg 2) + (y \gg 1))$ 
        } else {
             $U'_{444}(2x + 1, 2y) = V''_{420}{}^{aux}(x, y \gg 1)$ 
             $V'_{444}(2x + 1, 2y) = V''_{420}{}^{aux}(x, (H \gg 2) + (y \gg 1))$ 
        }
         $U'_{444}(2x, 2y) = U''_{420}{}^{main}(x, y)$ 
         $V'_{444}(2x, 2y) = V''_{420}{}^{main}(x, y)$ 
    }
}

```

[0153] 其中标记"表示根据(可能有损的)编码的重构。

[0154] B. 用信号通知帧封装信息的值的语法和语义。

[0155] 在示例实施方式中, 帧封装排列SEI消息被用来用信号通知两个4:2:0帧包括一个封装的4:4:4帧。在H.264/AVC标准以及HEVC标准中, 均对帧封装排列SEI消息进行了定义, 虽然这种帧封装排列SEI消息之前已被用于不同的目的。

[0156] 帧封装排列SEI消息已被设计用来使用2D视频编解码器发送立体3D视频帧。在这种情况下, 两个4:2:0帧代表立体3D视频场景的左视图和右视图。对于本文所描述的途径, 能够扩展帧封装排列SEI消息的范围来相反地支持从单个4:4:4帧获得的两个4:2:0帧的编码/解码, 接着进行帧解封来重新获得4:4:4帧。两个4:2:0帧代表主要视图和辅助视图。主要视图(帧)和辅助视图(帧)的格式均相当于4:2:0格式。主要视图(帧)可以独立使用, 而辅助视图(帧)在与主要视图一起得到恰当解释时是有用的。因此, 这些途径可以使用帧封装排列SEI消息, 使用能够编码/解码4:2:0帧的视频编解码器, 来有效地支持编码/解码4:4:4帧。

[0157] 为此, 扩展SEI消息。例如, 对语法元素content_interpretation_type的语义进行如下扩展。在相关的帧封装途径中, 对于YUV 4:4:4帧, 有两个YUV 4:2:0组成帧——用于主要视图的第1帧和用于辅助视图的第2帧。Content_interpretation_type指示下表中指定的组成帧的预期解释。值0、1和2的解释与H.264/AVC标准和HEVC标准中相同。对用于content_interpretation_type的新值进行定义来指示组成帧应当被解释为包含来自YUV 4:4:4帧的数据:

值	解释
0	帧封装的组成帧之间未指定关系
1	指示两个组成帧形成立体视图场景的左视图和右视图, 其中第 0 帧与左视图相关联, 第 1 帧与右视图相关联。
2	指示两个组成帧形成立体视图场景的右视图和左视图, 其中第 0 帧与右视图相关联, 第 1 帧与左视图相关联。
3	指示两个组成帧形成代表 YUV 4:4:4 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧, 其中第 0 帧与主要视图相关联, 第 1 帧与辅助视图相关联。指示第 0 帧的色度样本应被解释为 4:4:4 帧的未滤波样本 (未经抗混叠 (anti-alias) 滤波)。
4	指示两个组成帧形成代表 YUV 4:4:4 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧, 其中第 0 帧与主要视图相关联, 第 1 帧与辅助视图相关联。指示第 0 帧的色度样本应当被解释为已在帧封装之前被抗混叠滤波。
5	指示两个组成帧形成代表 YUV 4:4:4 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧, 其中第 1 帧与主要视图相关联, 第 0 帧与辅助视图相关联。指示第 1 帧的色度样本应被解释为 4:4:4 帧的未滤波样本 (未经抗混叠滤波)。
6	指示两个组成帧形成代表 YUV 4:4:4 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧, 其中第 1 帧与主要视图相关联, 第 0 帧与辅助视图相关联。指示第 1 帧的色度样本应当被解释为已在帧封装之前被抗混叠滤波。

[0158] 可选地, 语法元素content_interpretation_type的不同值与上表所示的解释相关。或者, 用于content_interpretation_type的其它和/或另外解释, 可以用来支持较低分辨率色度采样格式的帧的编码/解码, 该较低分辨率色度采样格式的帧通过帧封装从一个或多个较高分辨率色度采样格式的帧获得。

[0160] 另外,为了简化的目的,也可以为帧封装排列SEI消息的其它语法元素施加以下一个或多个约束。当content_interpretation_type的值介于3和6之间时(即,对于涉及将YUV4:4:4帧封装成YUV 4:2:0帧的情况),语法元素quincunx_sampling_flag,spatial_flipping_flag,frame0_grid_position_x,frame0_grid_position_y,frame1_grid_position_x和frame1_grid_position_y的值应为0。此外,当content_interpretation_type等于3或5时(表示没有在预处理中进行滤波),chroma_loc_info_present_flag应为1,并且chroma_sample_loc_type_top_field和chroma_sample_loc_type_bottom_field应为2。

[0161] 在H.264/AVC标准中(以及在HEVC标准中),语法元素frame_packing_arrangement_type指示如何排列立体视图的两个组成帧。例如,frame_packing_arrangement_type==3指示两个组成帧并排封装,frame_packing_arrangement_type==4指示两个组成帧上下封装,并且frame_packing_arrangement_type==5指示两个组成帧时间交织。语法元素frame_packing_arrangement_type同样可以和content_interpretation_type的值结合使用,该content_interpretation_type的值指示对较高分辨率色度采样格式的帧的封装。例如,frame_packing_arrangement_type==3可以指示主要帧和辅助帧的并排封装,frame_packing_arrangement_type==4可以指示主要帧和辅助帧的上下封装,并且frame_packing_arrangement_type==5可以指示主要帧和辅助帧的时间交织。或者,以某种其他方式用信号通知帧封装排列元数据。可选地,可以扩展frame_packing_arrangement_type的语义来指示对较高分辨率色度采样格式的帧的封装,而不是扩展content_interpretation_type语法元素的语义来指示对较高分辨率色度采样格式的帧的封装。例如,除了指示如何排列主要视图和辅助视图,帧封装排列元数据(诸如frame_packing_arrangement_type的值大于5)可以指示是否使用或未使用帧封装/帧解封、是否已使用或未使用滤波或其他预处理操作(并因此是否应使用或不使用相应的后处理滤波或其他后处理操作)、要执行的后处理操作的类型、或关于帧封装/解封的其他信息。

[0162] 在这些示例中,帧封装排列SEI消息通知解码器经解码的图片包含作为帧封装排列的组成帧的4:4:4帧的主要视图和辅助视图。该信息可以被用来恰当地处理主要视图和辅助视图,以便用于显示或其它目的。例如,在解码端的系统需要4:4:4格式的视频并且能够从主要视图和辅助视图对4:4:4帧进行重构的情况下,该系统可以这样做,并且输出格式将为4:4:4。否则,只给出主要视图作为输出,而输出格式然后将为4:2:0。

[0163] C. 预处理操作和后处理操作。

[0164] 对较高分辨率色度采样格式的帧的色度样本值的简单子采样可能会在下采样(downsampled)的色度样本值中引入混叠伪影。为了减少混叠,帧封装可以包括预处理操作来对色度样本值进行滤波。这种滤波可以被称为抗混叠滤波。而然后相应的帧解封可以包括后处理操作,以对色度样本值的预处理滤波进行补偿。例如,参照上表,当content_interpretation_type为4或6时,预处理操作可以用来在帧封装期间对色度样本值进行滤波,而帧解封可以包括相应的后处理操作。

[0165] 有各种理由进行适于帧封装/解封的预处理和后处理。

[0166] 例如,当只有代表主要视图的YUV 4:2:0帧用于显示时,预处理可以帮助提高质

量。这可以允许解码器忽略代表辅助视图的YUV 4:2:0帧,而无需冒着由简单的色度信息子采样带来的混叠伪影的风险。在不进行预处理的情况下(当通过对来自YUV 4:4:4帧的色度信号的直接子采样,获得用于代表主要视图的YUV 4:2:0帧的色度信号时),当只有主要视图被用于生成输出时,可以在一些内容上看到混叠伪影,例如,ClearType文本内容。

[0167] 作为另一示例,预处理和后处理可以帮助保持/强制实现YUV 4:4:4域中的经压缩的色度信号的一致性和平滑度。当帧封装被用来将YUV 4:4:4帧封装成两个YUV 4:2:0帧时,色度信号被拆分到多个区域,并且根据每个区域的位置,可以对其进行不同的压缩(例如,进行不同程度的量化)。正因如此,当通过对来自不同区域的数据进行交织来再次组装色度信号时,可能会引入人工不连续性和高频噪声。后处理操作可以帮助平滑因压缩而在这些区域产生的差异。

[0168] 作为另一示例,预处理可以帮助强化对代表辅助视图的YUV 4:2:0帧的压缩,该辅助视图包含剩余色度信息。

[0169] 在一些示例实施方式中,预处理操作和后处理操作受到限制,使其只影响到代表主要视图的YUV 4:2:0帧的一部分的色度信号。即,经滤波的样本值是主要视图的色度分量的一部分。

[0170] 此外,对于和AVC编码/解码或HEVC编码/解码相结合的帧封装/解封,预处理操作和后处理操作可以基于色度样本位置类型(指示色度样本网格与亮度样本网格对齐)。色度样本位置类型是根据作为压缩比特流的一部分用信号通知的chroma_sample_loc_type_top_field和chroma_sample_loc_type_bottom_field语法元素来确定。(对于逐行扫描源内容,这两个元素通常会具有相等的值。)对于给定的色度样本位置类型,如果色度样本与亮度样本针对特定方向(水平或垂直)对齐,则奇数抽头对称滤波器(诸如 $[1 \ 21]/4$,或 $[0.25 \ 0.5 \ 0.25]$,连同舍入操作)被用来在该方向对色度进行滤波。另一方面,如果色度样本与亮度样本未针对特定方向(水平或垂直)对齐,并且色度样本网格位置在特定方向(水平垂直)的亮度样本位置之间居中,则偶数抽头对称滤波器(通常为 $[11]/2$,或 $[0.5 \ 0.5]$,连同舍入操作)被用来在该方向对色度进行滤波。用于后者情况的另一种可能的滤波器选择是 $[1 \ 3 \ 3 \ 1]/8$,或 $[0.125 \ 0.375 \ 0.375 \ 0.125]$,连同舍入操作。通常做出后处理操作的选择以使后处理操作对预处理操作进行补偿。在某些情况下,后处理直接反转预处理,而在另一些情况下,后处理仅仅对预处理进行近似反转,如下文所说明。

[0171] 在结合AVC编码/解码或HEVC编码/解码的帧封装/解封的实施方式中,如果用于chroma_sample_loc_type_top_field和chroma_sample_loc_type_bottom_field语法元素的色度样本位置类型为1,色度样本则不与水平方向或垂直方向的亮度样本对齐,并且因此,滤波器 $[0.5 \ 0.5]$ 被应用在水平方向和垂直方向以进行预处理操作。在这种情况下,对于参照图9所示的途径(900),导出样本值区域B2和B3的等式如下所示:

[0172] • 对于区域B2:

[0173]

$$U_{420}^{\text{main_filt}}(x,y) = [U_{444}(2x,2y) + U_{444}(2x+1,2y) + U_{444}(2x,2y+1) + U_{444}(2x+1,2y+1) + 2]/4$$

[0174] ,并且

[0175] • 对于区域B3:

[0176]

$$V_{420}^{\text{main_filt}}(x, y) = [V_{444}(2x, 2y) + V_{444}(2x + 1, 2y) + V_{444}(2x, 2y + 1) + V_{444}(2x + 1, 2y + 1) + 2]/4,$$

[0177] 其中两个区域的(x, y)的范围均为 $[0, w/2-1] \times [0, H/2-1]$ 。

[0178] 由于该滤波,所以,来自YUV 4:4:4帧的在位置 $U_{444}(2x, 2y)$ 和 $V_{444}(2x, 2y)$ 处的样本值未在主要视图(902)中直接表示;相反,经滤波的样本值($U_{420}^{\text{main_filt}}(x, y)$ 和 $V_{420}^{\text{main_filt}}(x, y)$)位于主要视图(902)中的位置。来自YUV 4:4:4帧的位于 $U_{444}(2x+1, 2y)$ 、 $U_{444}(2x, 2y+1)$ 、 $U_{444}(2x+1, 2y+1)$ 、 $V_{444}(2x+1, 2y)$ 、 $V_{444}(2x, 2y+1)$ 、 $V_{444}(2x+1, 2y+1)+2]$ 的样本值仍然在区域B4...B9之间在辅助视图(903)中直接表示。

[0179] 在作为后处理操作的一部分的相应滤波中,当要输出YUV 4:4:4格式的帧时,用于YUV 4:4:4帧的位置 $U_{444}(2x, 2y)$ 和 $V_{444}(2x, 2y)$ 处的样本值可以根据封装的帧中的值,作为 $U_{444}(2x, 2y)$ 和 $V'_{444}(2x, 2y)$ 进行如下计算:

[0180]

$$\bullet U'_{444}(2x, 2y) = (1 + 3\alpha) * U_{420}^{\text{main_filt}}(x, y) - \alpha * [U''_{444}(2x + 1, 2y) + U''_{444}(2x, 2y + 1) + U''_{444}(2x + 1, 2y + 1)],$$

[0181] 并且

[0182]

$$\bullet V'_{444}(2x, 2y) = (1 + 3\alpha) * V_{420}^{\text{main_filt}}(x, y) - \alpha * [V''_{444}(2x + 1, 2y) + V''_{444}(2x, 2y + 1) + V''_{444}(2x + 1, 2y + 1)],$$

[0183] 其中(x, y)的范围是 $[0, w/2-1] \times [0, H/2-1]$, α 是取决于实施方式的加权因子,而标记"指示根据(可能有损)编码的重构。在亮度样本网格位置在水平方向和垂直方向的亮度样本位置之间居中并且建议的抗混叠滤波器为 $[0.5 \ 0.5]$ 的条件下,值 $\alpha=1$ 时将在不存在量化误差和舍入误差的条件下对输入值进行完全重构,从而直接反转在预处理中执行的滤波。对于其他值的 α ,后处理期间的滤波只对预处理中执行的滤波进行近似反转。当把量化误差考虑在内时,可能建议使用具有稍小值的 α (例如, $\alpha=0.5$),以减少可察觉的伪影。一般情况下, α 的范围应介于0.0和1.0之间,并且当量化步长较大时, α 应该更小。使用较大值的 α 可能会加剧因有损压缩而引入的伪影。

[0184] 或者,可以将不同的加权分配给不同的样本位置。用于YUV 4:4:4帧的位置 $U_{444}(2x, 2y)$ 和 $V_{444}(2x, 2y)$ 处的样本值可以根据封装的帧中的值,作为 $U'_{444}(2x, 2y)$ 和 $V'_{444}(2x, 2y)$ 进行如下计算:

[0185]

$$\bullet U'_{444}(2x, 2y) = (1 + \alpha + \beta + \gamma) * U_{420}^{\text{main_filt}}(x, y) - \alpha * U''_{444}(2x + 1, 2y) - \beta * U''_{444}(2x, 2y + 1) - \gamma * U''_{444}(2x + 1, 2y + 1),$$

[0186]

$$\bullet V'_{444}(2x, 2y) = (1 + \alpha + \beta + \gamma) * V_{420}^{\text{main_filt}}(x, y) - \alpha * V''_{444}(2x + 1, 2y) - \beta * V''_{444}(2x, 2y + 1) - \gamma * V''_{444}(2x + 1, 2y + 1),$$

[0187] 其中(x, y)的范围是 $[0, w/2-1] \times [0, H/2-1]$, α 、 β 和 γ 是取决于实施方式的加权因子,而标记"表示根据(可能有损)编码的重构。在亮度样本网格位置在水平方向和垂直方向上在亮度样本位置之间居中并且建议的抗混叠滤波器为 $[0.5 \ 0.5]$ 的条件下,值 $\alpha=\beta=\gamma=1$ 时将在不存在量化误差和舍入误差的情况下对输入值进行完全重构,从而直接反转在预处理中执行的滤波。对于其他值的 α 、 β 和 γ ,后处理期间的滤波只对预处理中执行的滤波进行近似反转。当考虑到量化误差时,可能建议使用具有稍小值的 α 、 β 和 γ (例如, $\alpha=\beta=\gamma=0.5$),以减少可察觉的伪影。一般情况下, α 、 β 和 γ 的范围应介于0.0到1.0之间,并且当量化

步长较大时, α 、 β 和 γ 应该更小。使用大值的 α 、 β 和 γ 可能会加剧因有损压缩而引入的伪影。 α 、 β 和 γ 的值可以使用互相关分析, 针对条件最优性进行设计。

[0188] 当 $\alpha=\beta=\gamma=1$ 时, 用于YUV 4:4:4帧的位置 $U_{444}(2x, 2y)$ 和 $V_{444}(2x, 2y)$ 处的样本值可以根据封装的帧中的值, 作为 $U'_{444}(2x, 2y)$ 和 $V'_{444}(2x, 2y)$ 进行如下计算:

[0189]

$$\bullet U'_{444}(2x, 2y) = 4 * U''_{420}^{main_filt}(x, y) - U''_{444}(2x + 1, 2y) - U''_{444}(2x, 2y + 1) - U''_{444}(2x + 1, 2y + 1) ,$$

[0190] 并且

[0191]

$$\bullet V'_{444}(2x, 2y) = 4 * V''_{420}^{main_filt}(x, y) - V''_{444}(2x + 1, 2y) - V''_{444}(2x, 2y + 1) - V''_{444}(2x + 1, 2y + 1)$$

[0192] 其中 (x, y) 的范围是 $[0, w/2-1] \times [0, H/2-1]$ 。

[0193] 例如, 在预处理期间, 对位置 $(2x, 2y)$ 、 $(2x+1, 2y)$ 、 $(2x, 2y+1)$ 和 $(2x+1, 2y+1)$ 处的样本值29、15、7和18进行过滤以产生样本值17.25, 然后舍入得17。值经滤波的样本值17被用来代替原始样本值29。在后处理期间, 位置 $(2x, 2y)$ 处的样本值被重构为 $68-15-7-18=28$ 。在原始样本值(29)和重构样本值(28)之间的差示出因预处理操作的滤波而造成的精度损失。

[0194] 可选地, 即使在预处理期间执行滤波, 设备可以在后处理期间选择性地跳过滤波操作。例如, 设备可以在后处理期间跳过滤波, 以减少解码和播放的计算负荷。

[0195] 可选地, 预处理操作和后处理操作不限于4:4:4帧的色度信号, 该4:4:4帧是代表主要视图的4:2:0帧的一部分(例如, 图9中所表示的帧902的区域B2和B3)。相反, 还对4:4:4帧的色度信号执行预处理操作和后处理操作, 该4:4:4帧是代表辅助视图的4:2:0帧的一部分(例如, 图9中所表示的帧903的区域B4到和B9)。这种预处理操作和后处理操作(用于代表辅助视图的4:2:0帧的一部分的4:4:4帧的色度信号)可以使用不同的滤波操作, 而不是构成代表主要视图的4:2:0帧的一部分的4:4:4帧的色度信号的预处理和后处理。

[0196] 在预处理操作和后处理操作的前述示例中, 在预处理期间使用平均化滤波并在后处理期间使用相应的滤波。可选地, 预处理操作和后处理操作可以实现一个变换/反变换对。例如, 该变换/反变换对可以是小波变换、提升变换和其他变换的类的一种。也可以根据使用情况情境设计具体变换, 以便满足以上提及的用于在封装4:4:4帧的上下文中的预处理操作和后处理操作的不同设计理由。或者, 预处理和后处理可使用其它滤波器结构, 具有其他支持的滤波器区域, 或使用关于内容和/或保真度自适应(例如, 关于用于编码的量化步长自适应)的滤波。

[0197] 在一些示例实施方式中, 经帧封装的4:2:0内容的表示和/或压缩, 可以使用比4:4:4内容的原始样本比特深度更大的样本比特深度。例如, 4:4:4帧的样本比特深度为每个样本8比特, 而经帧封装的4:2:0帧的样本比特深度为每个样本10比特。这可以帮助减少预处理操作和后处理操作的应用期间的精度损失。或者, 在使用有损压缩对4:2:0帧进行编码时, 这可以帮助获得更高水平的保真度。例如, 如果4:4:4内容的样本比特深度为每个样本8比特, 并且经帧封装的4:2:0内容的样本比特深度为每个样本10比特, 那么, 可以在编码器和解码器的全部或大部分内部模块中保持每个样本10比特的比特深度。在接收端将内容解封为4:4:4格式之后, 如果需要的话, 可以将样本比特深度减少到每个样本8比特。更一般

地,较高分辨率色度采样格式的帧的样本值可以具有第1比特深度(诸如每个样本8、10、12或16比特),而较低分辨率色度采样格式的帧的样本值(帧封装之后)具有比第1比特深度更大的第2比特深度。

[0198] D.YUV 4:2:2视频的替代。

[0199] 在许多前述示例中,YUV 4:4:4帧被封装成YUV 4:2:0帧用于编码和解码。在其它示例中,YUV 4:2:2帧被封装成YUV 4:2:0帧用于编码和解码。典型的4:2:2帧每4个像素位置包含8个样本值,而4:2:0帧每4像素位置仅包含6个样本值。所以,包含在4:2:2帧中的样本值均可被封装成4/3 4:2:0帧。即,在进行有效封装时,三个4:2:2帧可被封装成四个4:2:0帧。

[0200] 在一种途径中,对4:2:2帧的帧封装以和在图8中示出的用于4:4:4到4:2:0帧封装的简单途径(800)相似的简单方式完成。

[0201] 在其他途径中,YUV 4:2:2帧被封装成YUV 4:2:0帧,同时保持用于YUV 4:2:2帧的色度信息的几何对应。所得到的YUV 4:2:0帧在其Y、U和V分量之间具有良好的几何对应,可被更好地压缩,这是因为其适合于对编码的YUV 4:2:0帧的典型编码器所期望的模型。同时,尽管颜色分量的分辨率较低,但是,还是可以完成封装来使YUV 4:2:0帧代表正由YUV 4:2:2帧代表的完整场景。

[0202] 在将YUV 4:2:2封装成两个YUV 4:2:0帧(主要视图和辅助视图)的同时,可以满足这些设计约束。辅助视图将具有“空”区域,但这些区域可以使用固定值填充,或者通过复制色度值来填充。或者,该空区域可用来指示其他信息,诸如场景深度。例如,对于参照图9所描述的封装途径(900),该途径(900)可按原样使用,除了区域B4和B5将没有数据。或者,对于参照图11所描述的封装途径(1100),该途径(1100)可按原样使用,除了区域A4和A5将没有数据。

[0203] 在示例实施方式中,对用于content_interpretation_type的新值进行定义来用信号通知将YUV 4:2:2帧封装成YUV 4:2:0组成帧,如下表所示。

[0204]

值	解释
0	帧封装的组成帧之间未指定关系
1	指示两个组成帧形成立体视图场景的左视图和右视图，其中第 0 帧与左视图相关联，第 1 帧与右视图相关联。
2	指示两个组成帧形成立体视图场景的右视图和左视图，其中第 0 帧与右视图相关联，第 1 帧与左视图相关联。
...	...
7	指示两个组成帧形成代表 YUV 4:2:2 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧，其中第 0 帧与主要视图相关联，第 1 帧与辅助视图相关联。指示第 0 帧的色度样本应被解释为 4:2:2 帧的未滤波样本（未经抗混叠滤波）。
8	指示两个组成帧形成代表 YUV 4:2:2 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧，其中第 0 帧与主要视图相关联，第 1 帧与辅助视图相关联。指示第 0 帧的色度样本应当被解释为已在帧封装之前被抗混叠滤波。
9	指示两个组成帧形成代表 YUV 4:2:2 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧，其中第 1 帧与主要视图相关联，第 0 帧与辅助视图相关联。指示第 1 帧的色度样本应被解释为 4:2:2 帧的未滤波样本（未经抗混叠滤波）。
10	指示两个组成帧形成代表 YUV 4:2:2 帧的主要 YUV 4:2:0 帧和辅助 YUV 4:2:0 帧，其中第 1 帧与主要视图相关联，第 0 帧与辅助视图相关联。指示第 1 帧的色度样本应当被解释为已在帧封装之前被抗混叠滤波。

[0205] 可选地，语法元素content_interpretation_type的不同值与上表所示的解释相关联。或者，用于content_interpretation_type的其它和/或另外解释，可以用来支持较低分辨率色度采样格式的帧的编码/解码，该较低分辨率色度采样格式的帧通过帧封装从一个或多个较高分辨率色度采样格式的帧获得。

[0206] E. 其他色度采样格式。

[0207] 本文所描述的许多示例涉及以诸如4:4:4、4:2:2、4:2:0等之类的采样率作为色度采样格式的诸如Y'UV、YIQ、Y'IQ、YdbDr、YCbCr、YCoCg等之类的YUV色彩空间的变形。可选地，所描述的途径能够用于以诸如4:4:4、4:2:2、4:2:0等之类的采样率作为色度采样格式的诸如RGB、GBR等之类的色彩空间。例如，设备可以将较高分辨率非YUV色度采样格式（诸如RGB 4:4:4或GBR 4:4:4）的帧封装成较低分辨率格式（诸如4:2:0格式）的帧，然后可对其进行编码。在编码中，标称亮度分量和标称色度分量代表非YUV分量的样本值（而不是近似的亮度值和色差值）。在相应的解封中，设备将较低分辨率格式（诸如4:2:0格式）的帧封装成较高分辨率非YUV色度采样格式（诸如RGB 4:4:4或GBR 4:4:4）的帧。

[0208] 此外，所描述的途径可用于将4:4:4格式、4:2:2格式或4:2:0格式的视频内容帧封装成通常用于灰度或单色视频内容的4:0:0格式。来自4:4:4格式、4:2:2格式或4:2:0格式的帧的色度信息可以被封装成4:0:0格式的一个或多个附加帧或辅助帧的主分量。

[0209] F. 帧封装/解封的一般性技术。

[0210] 图12示出了用于帧封装的一般性技术（1200）。例如参照图4所描述的、实施帧封装器的计算设备可以执行该技术（1200）。

[0211] 该设备将一个或多个较高分辨率色度采样格式的帧封装（1210）成一个或多个较低分辨率色度采样格式的帧。例如，该设备将4:4:4格式（例如，YUV 4:4:4格式）的（多个）帧

封装成4:2:0格式(例如,YUV 4:2:0格式)的(多个)帧。或者,该设备将4:2:2格式(例如,YUV 4:2:2格式)的(多个)帧封装成4:2:0格式(例如,YUV 4:2:0格式)的(多个)帧。或者,该设备将4:4:4格式(例如,YUV 4:4:4格式)的(多个)帧封装成4:2:2格式(例如,YUV 4:2:2格式)的(多个)帧。

[0212] 对于YUV格式,该设备可以执行帧封装(1210),以便在封装后在较高分辨率色度采样格式的(多个)帧的色度分量的相邻样本值之间保持几何对应。例如,这种样本值被保持为较低分辨率色度采样格式的(多个)帧的亮度分量和色度分量的相邻样本和/或并置(collocated)部分。稍后的编码可以利用这种几何对应。

[0213] 在一些帧封装途径中,该设备可以将较高分辨率色度采样格式的(多个)帧的较低色度分辨率版本,作为较低分辨率色度采样格式的(多个)帧的一部分嵌入。因此,较低分辨率色度采样格式的(多个)帧的一部分代表较高分辨率色度采样格式的(多个)帧的较低色度分辨率版本。较低分辨率色度采样格式的(多个)帧的剩下部分代表来自较高分辨率色度采样格式的(多个)帧的剩余色度信息。在其它帧封装途径中,根据较高分辨率色度采样格式的(多个)帧的空间分割,该设备将较高分辨率色度采样格式的(多个)帧的色度分量的样本值分配给较低分辨率色度采样格式的(多个)帧的亮度分量和色度分量。

[0214] 在帧封装期间,可对较高分辨率色度采样格式的(多个)帧的色度分量的样本值进行滤波,并将滤波后的样本值分配给较低分辨率色度采样格式的(多个)帧的色度分量的部分。在一些实施方式中,较高分辨率色度采样格式的(多个)帧的色度分量的样本值具有较低的比特深度(例如,每个样本8比特),而滤波后样本值具有较高的比特深度(例如,每个样本10比特)以用于以较高比特深度编码。

[0215] 然后,该设备可以对较低分辨率色度采样格式的(多个)帧进行编码(1220)。可选地,不同的设备执行编码(1220)。该(多个)设备可以在逐帧的基础上或其他基础上重复该技术(1200)。

[0216] 该设备可以用信号通知有关帧封装/解封的元数据。例如,该设备用信号通知指示是否使用或未使用帧封装/解封的元数据。或者,该设备用信号通知较高分辨率色度采样格式的帧的色度分量的样本值已在帧封装期间被滤波并且应该作为后处理的一部分而被滤波的指示。可以用信号通知关于帧封装/解封的元数据作为补充增强信息消息的一部分或作为某个其它类型的元数据。

[0217] 图13示出了用于帧解封的一般性技术(1300)。例如,参考图5所描述的帧解封器的计算设备可以执行该技术(1300)。

[0218] 在帧解封本身之前,该设备能够对较低分辨率色度采样格式的(多个)帧进行解码(1310)。可选地,不同的设备执行解码(1310)。

[0219] 该设备将一个或多个较低分辨率色度采样格式的帧解封(1320)成一个或多个较高分辨率色度采样格式的帧。例如,该设备将4:2:0格式(例如,YUV 4:2:0格式)的(多个)帧解封成4:4:4格式(例如,YUV 4:4:4格式)的(多个)帧。或者,该设备将4:2:0格式(例如,YUV 4:2:0格式)的(多个)帧解封成4:2:2格式(例如,YUV 4:2:2格式)的(多个)帧。或者,该设备将4:2:2格式(例如,YUV 4:2:2格式)的(多个)帧解封成4:4:4格式(例如,YUV 4:4:4格式)的(多个)帧。

[0220] 当较高分辨率色度采样格式的(多个)帧的较低色度分辨率版本作为较低分辨率

色度采样格式的(多个)帧的一部分被嵌入时,该设备可以选择进行显示。可以重构代表较高分辨率色度采样格式的(多个)帧的较低色度分辨率版本的较低分辨率色度采样格式的(多个)帧的部分,以用于输出和显示。较低分辨率色度采样格式的(多个)帧的剩下部分代表来自较高分辨率色度采样格式的(多个)帧的剩余色度信息,并且可以用作帧解封的一部分。在其它帧解封途径中,为了反转较高分辨率色度采样格式的(多个)帧的空间分割,该设备将较低分辨率色度采样格式的(多个)帧的亮度分量和色度分量的样本值,分配给较高分辨率色度采样格式的(多个)帧的色度分量。

[0221] 在帧解封期间,较高分辨率色度采样格式的(多个)帧的色度分量的样本值可以作为后处理的一部分被滤波。在一些实施方式中,较高分辨率色度采样格式的(多个)帧的色度分量的至少一些样本值在后处理滤波之前具有较高的比特深度(例如,每个样本10比特),并且这些样本值在后处理滤波之后具有较低的比特深度(例如,每个样本8比特)。

[0222] 该设备还可以接收有关帧封装/解封的元数据。例如,该设备接收指示是否使用或未使用帧封装/解封的元数据。或者,该设备接收较高分辨率色度采样格式的(多个)帧的色度分量的样本值已在帧封装期间被滤波并且应该作为后处理的一部分被滤波的指示。关于帧封装/解封的元数据可以作为补充增强信息消息的一部分或作为某个其它类型的元数据用信号通知。

[0223] 该(多个)设备可以在逐帧的基础上或其他基础上重复该技术(1300)。

[0224] 鉴于所公开的发明的原理可以应用到的多个可能的实施例,应当认识到,所示出的实施例仅是本发明的优选示例,而不应被视为对本发明范围的限制。相反,本发明的范围由以下权利要求限定。因此,我们要求保护作为我们的发明的所有落入这些权利要求的范围和精神的方案。

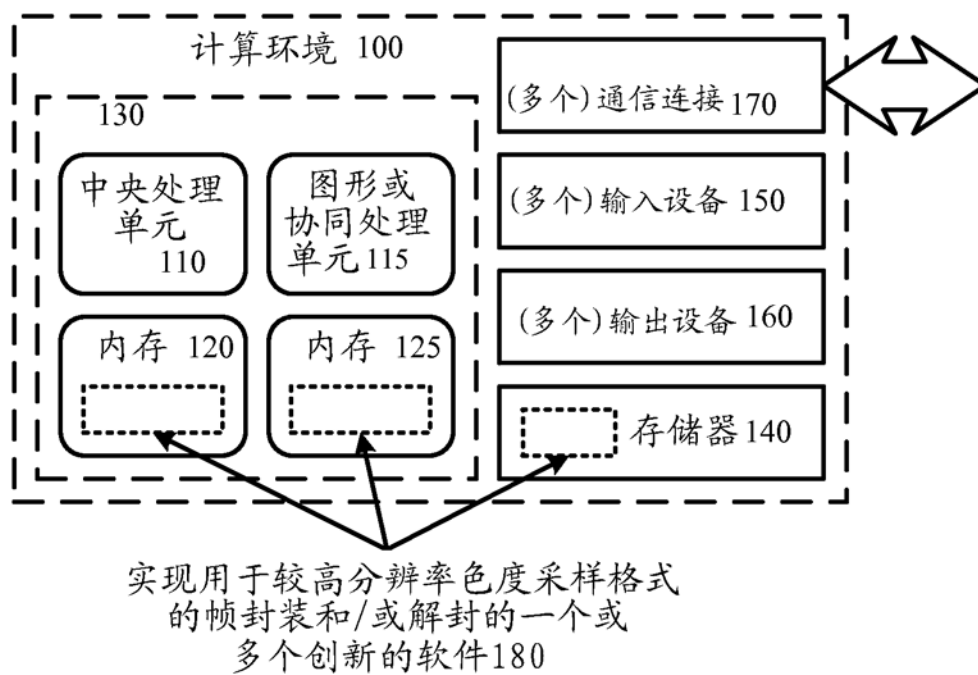


图 1

201

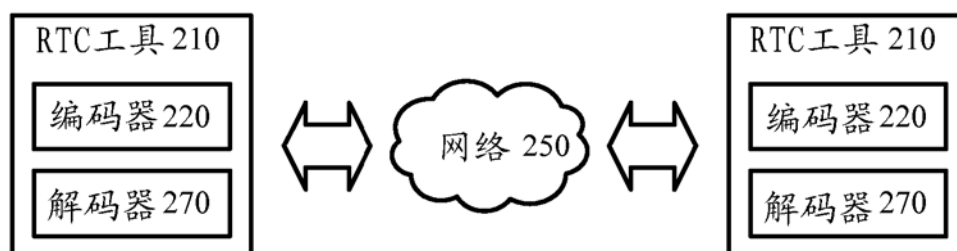


图 2a

202

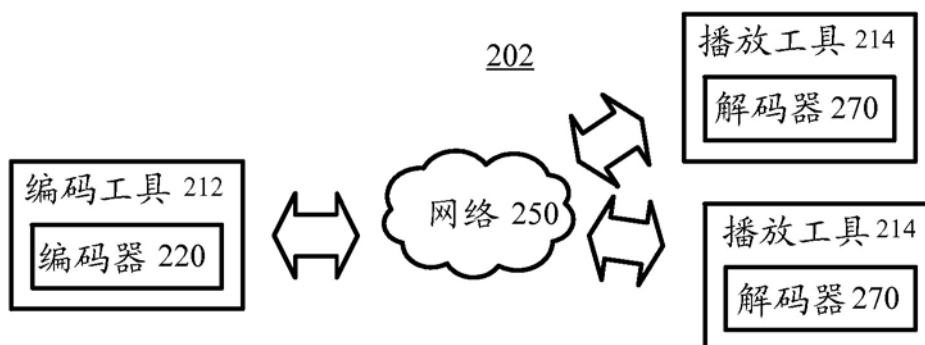


图 2b

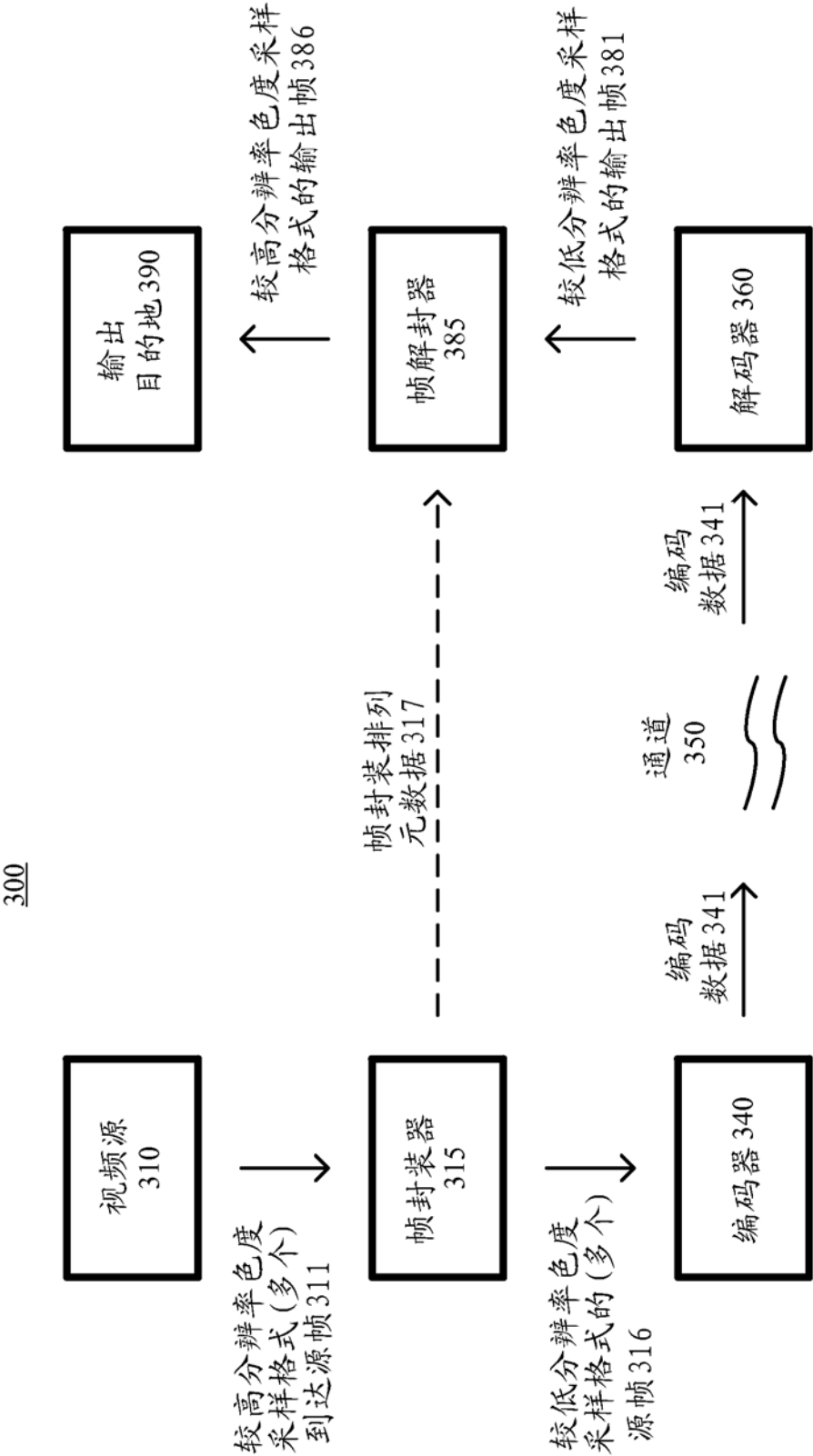


图 3

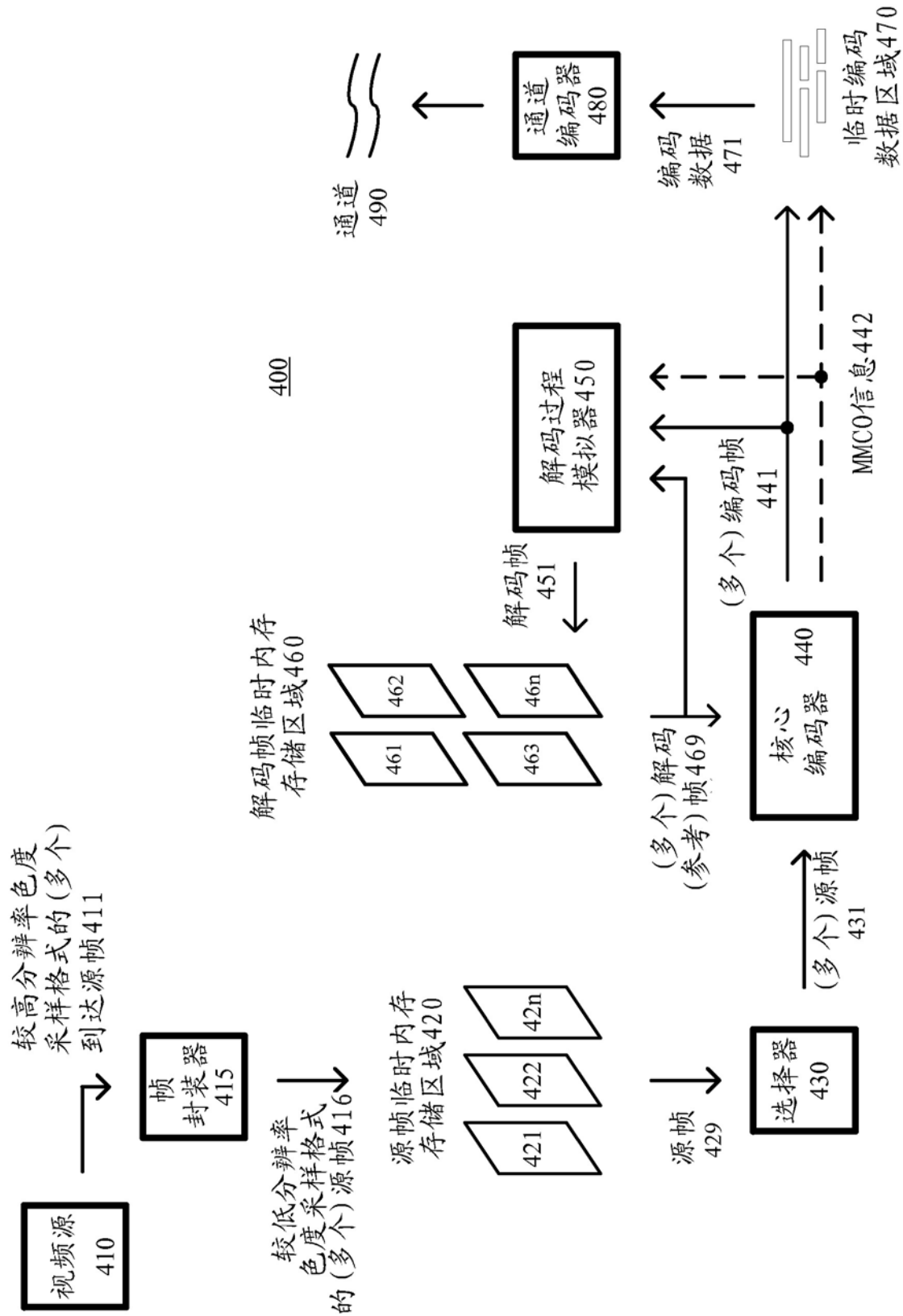


图 4

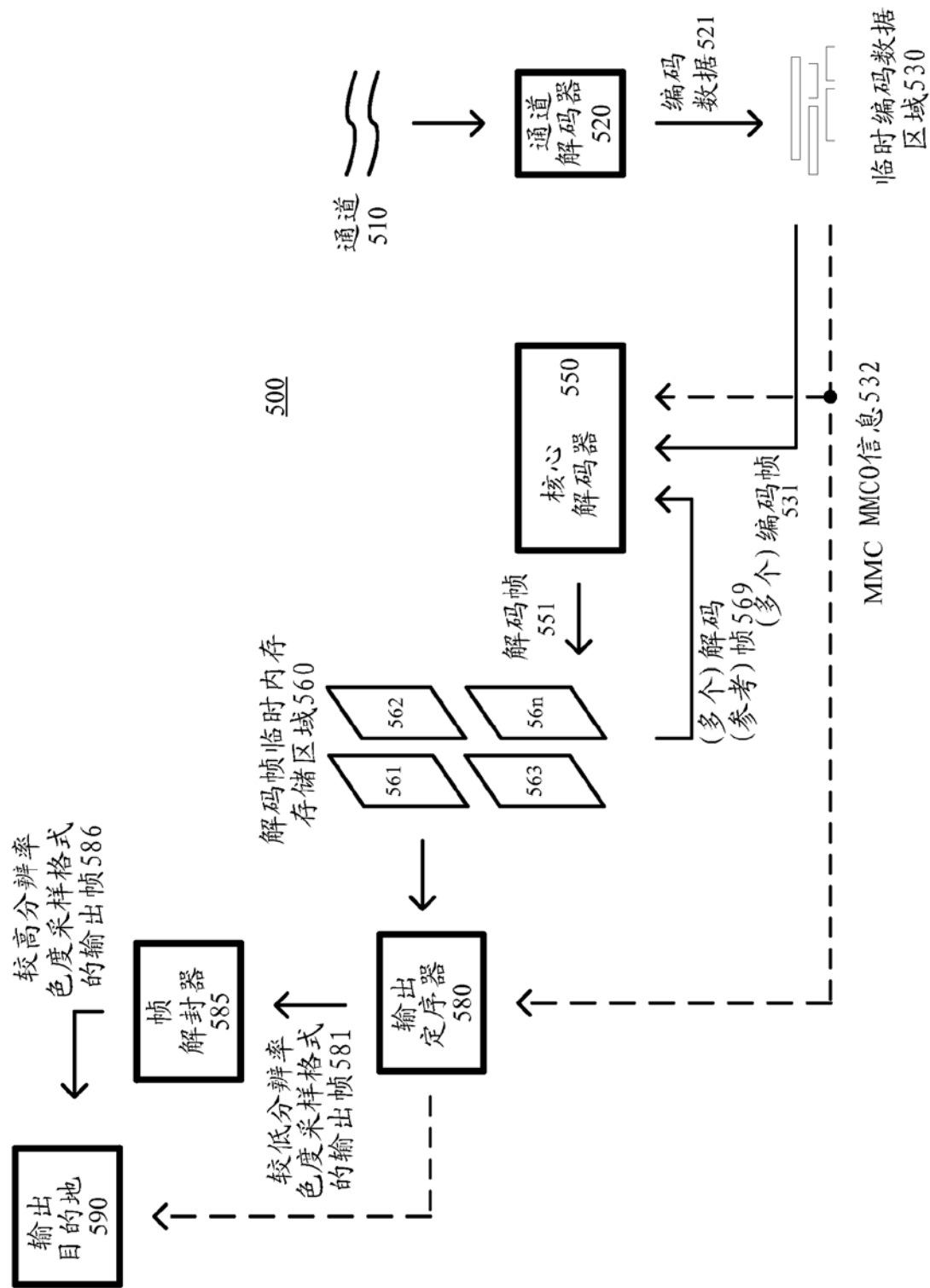


图 5

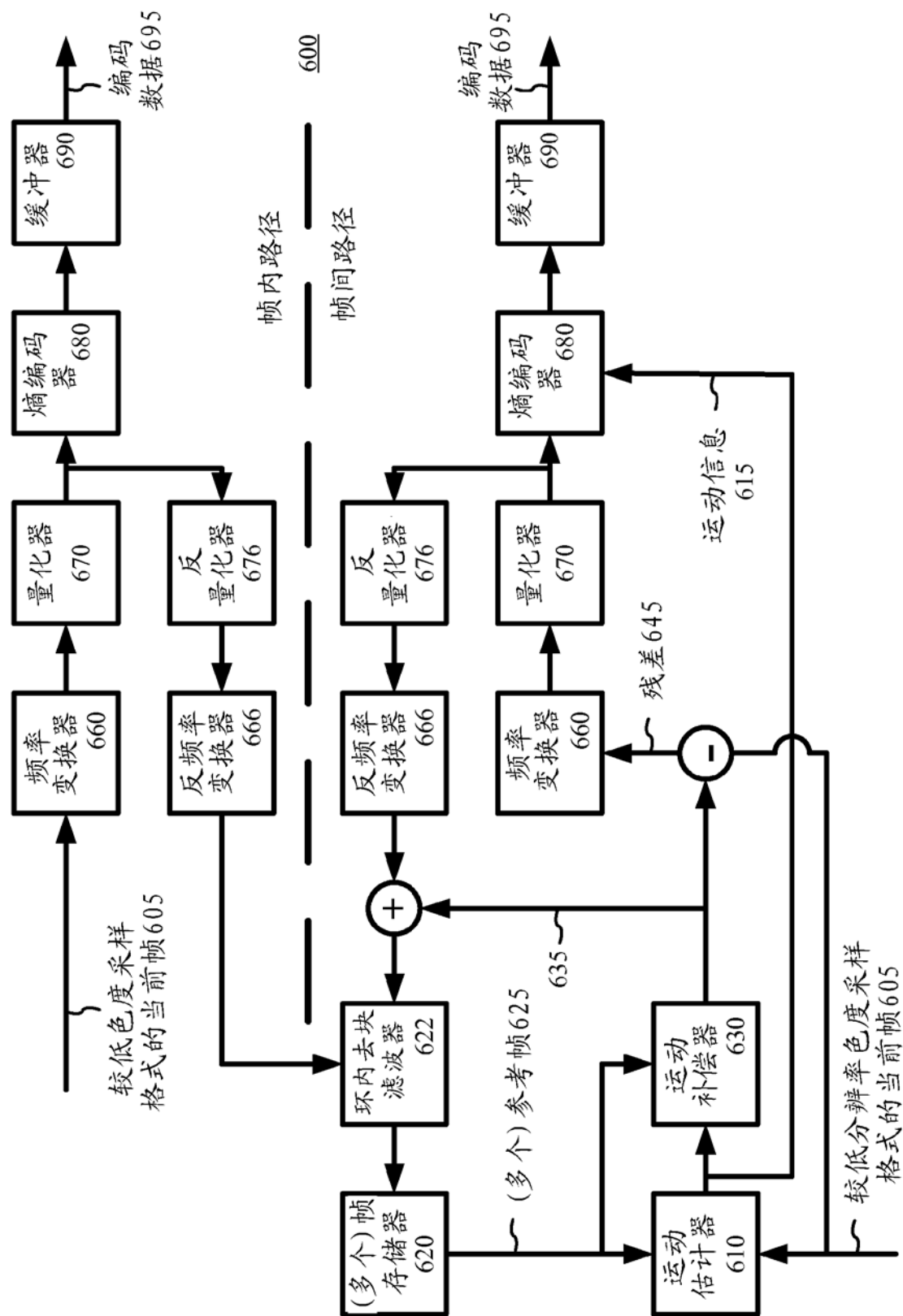


图 6

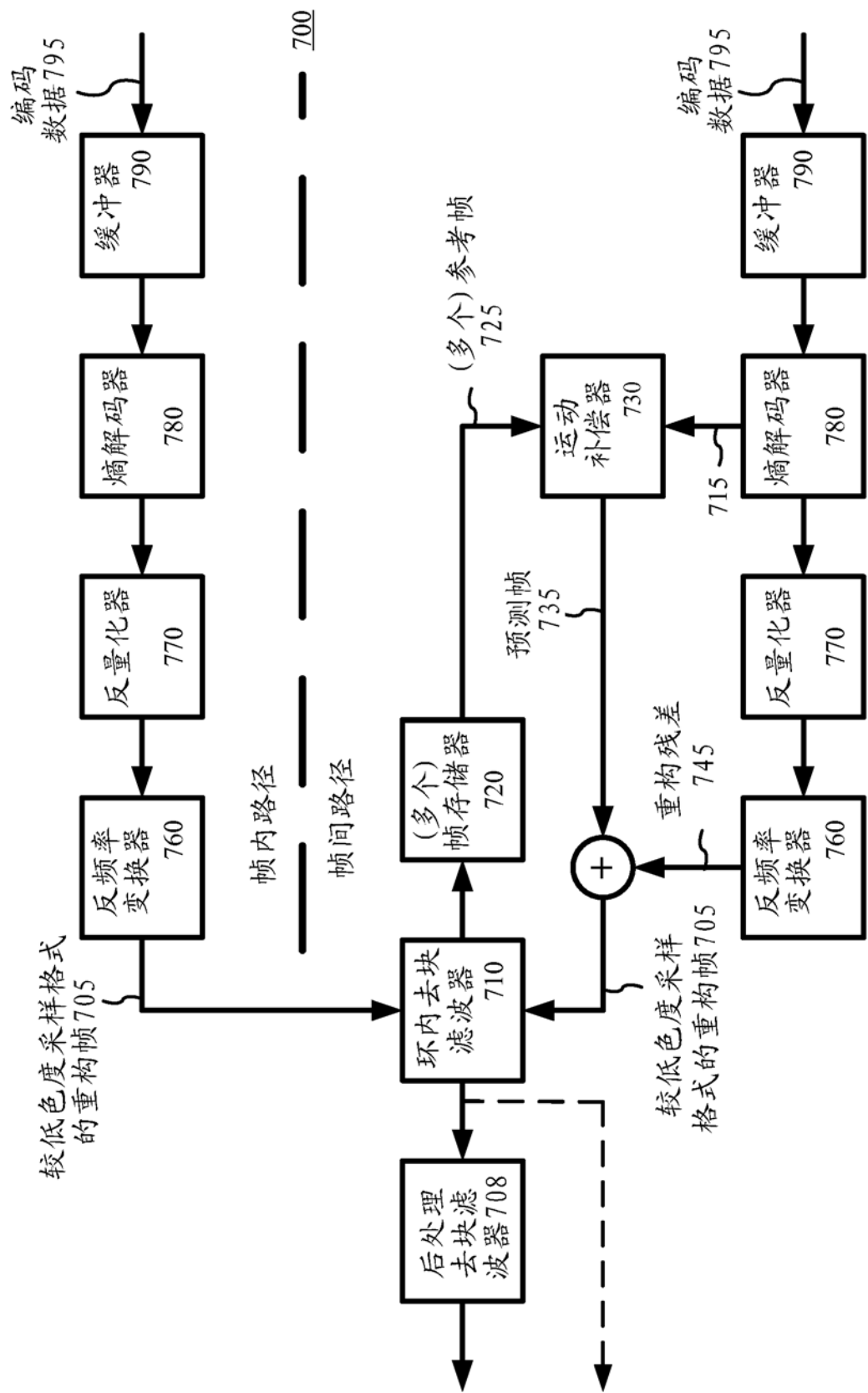


图 7

800

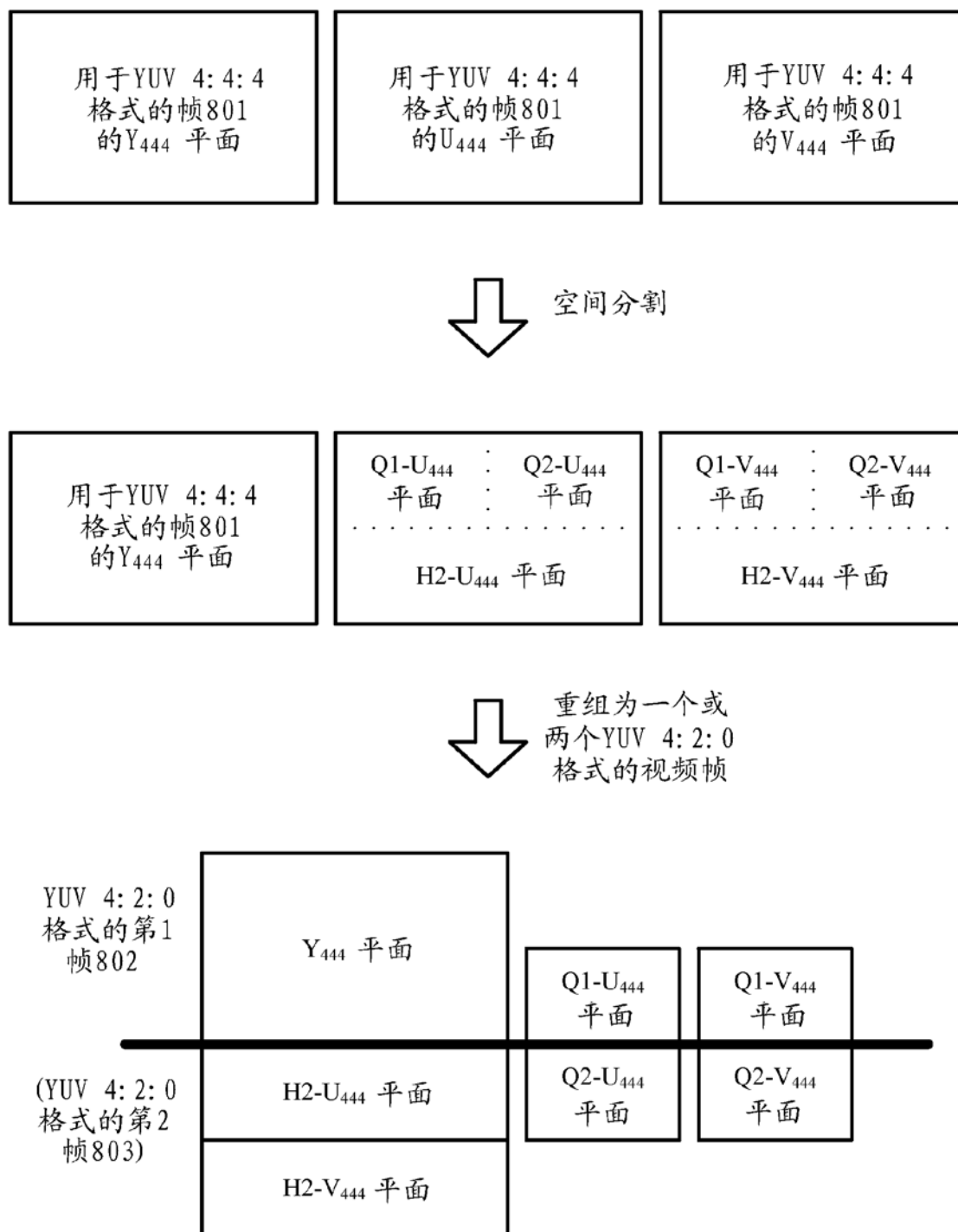


图 8

900

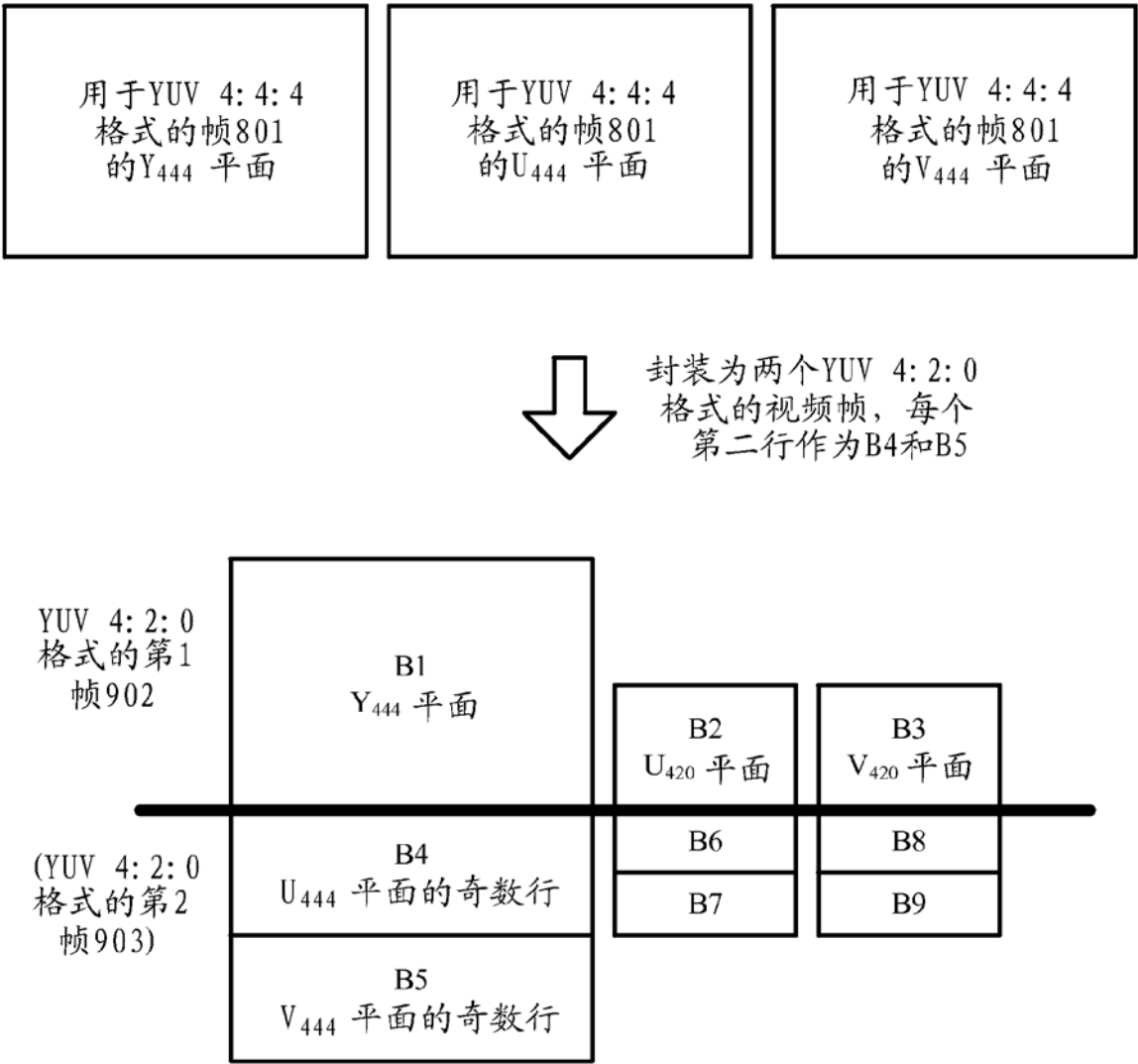


图 9

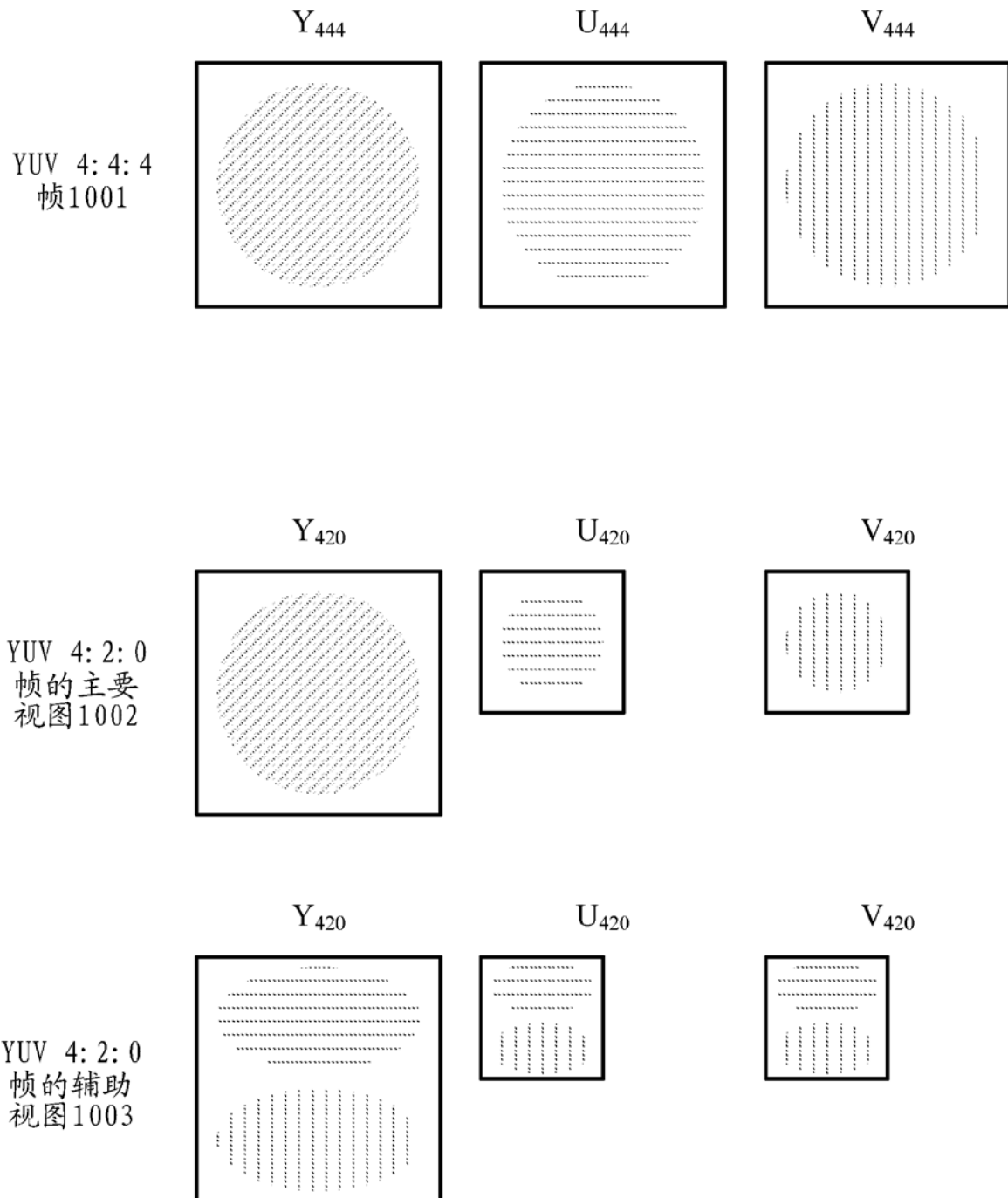
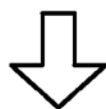
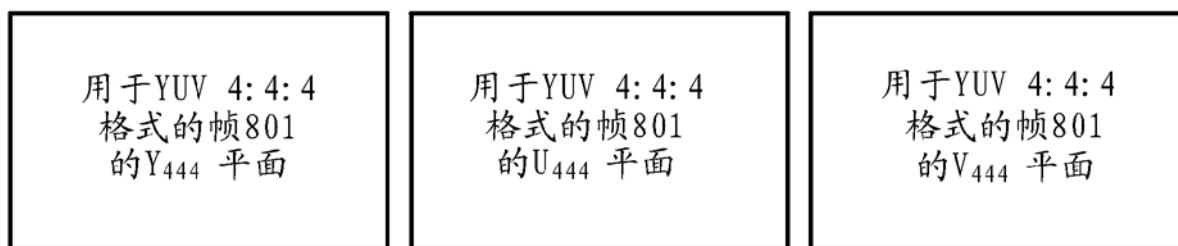
1000

图 10

1100

封装为两个YUV 4:2:0 格式的视频帧，每个第二列作为A4和A5

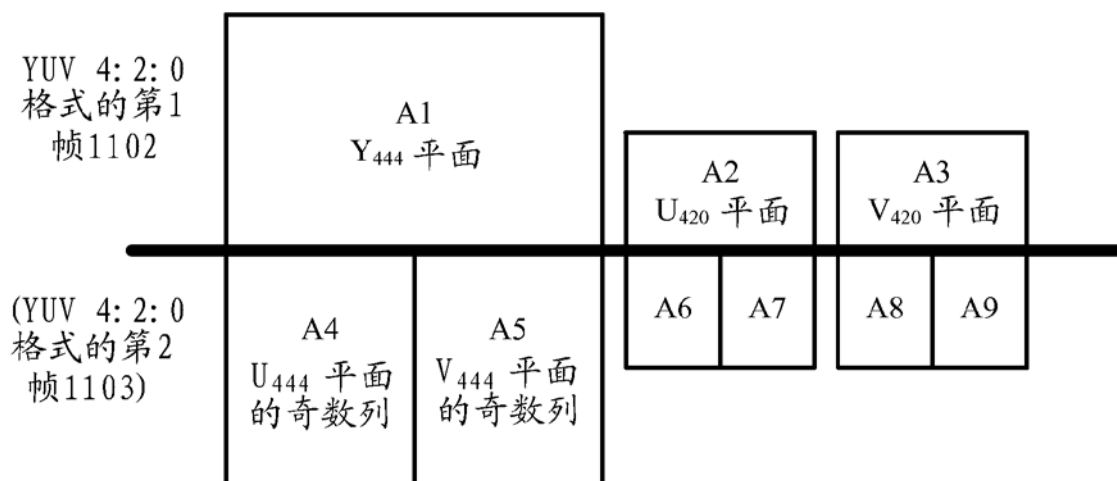


图 11

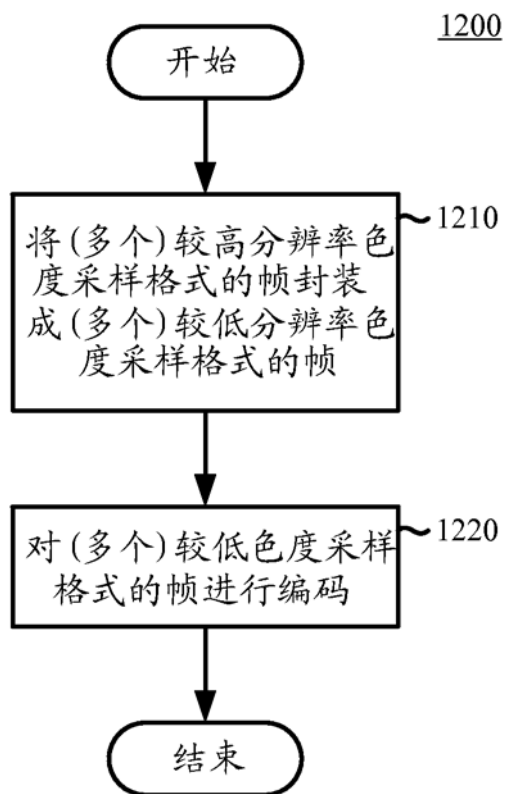


图 12

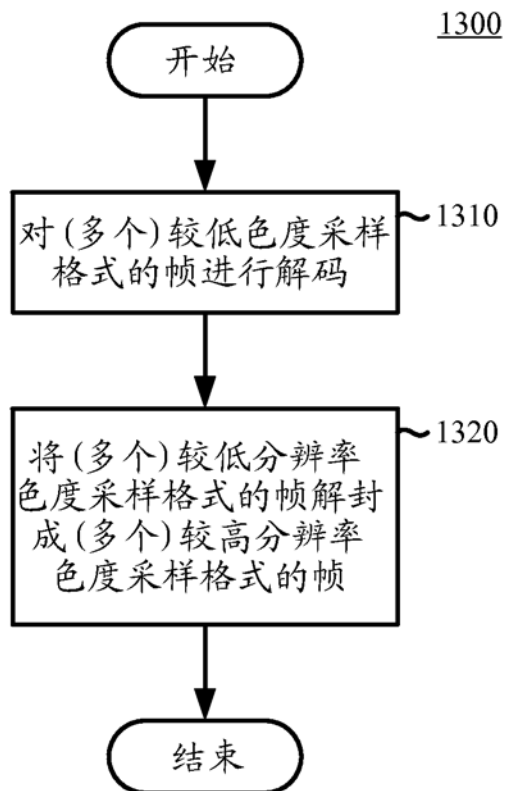


图 13