



(12) 发明专利

(10) 授权公告号 CN 114666597 B

(45) 授权公告日 2024. 11. 26

(21) 申请号 202210472084.3

(22) 申请日 2020.03.11

(65) 同一申请的已公布的文献号
申请公布号 CN 114666597 A

(43) 申请公布日 2022.06.24

(30) 优先权数据
62/816,521 2019.03.11 US
62/850,985 2019.05.21 US
62/883,195 2019.08.06 US
62/904,744 2019.09.24 US

(62) 分案原申请数据
202080002086.5 2020.03.11

(73) 专利权人 杜比实验室特许公司
地址 美国加利福尼亚州

(72) 发明人 R·阿特金斯 尹鹏 吕陶然

浦方君 S·T·麦卡锡
W·J·胡萨克 陈涛 苏冠铭

(74) 专利代理机构 北京市汉坤律师事务所
11602

专利代理师 魏小微 吴丽丽

(51) Int.Cl.
H04N 19/186 (2014.01)
H04N 19/184 (2014.01)
H04N 19/70 (2014.01)
H04N 19/30 (2014.01)
H04N 19/146 (2014.01)

(56) 对比文件
CN 108141619 A, 2018.06.08
US 2015016547 A1, 2015.01.15

审查员 章子衡

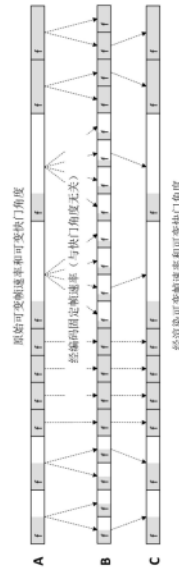
权利要求书2页 说明书33页 附图4页

(54) 发明名称

帧速率可伸缩视频编码

(57) 摘要

描述了用于帧速率可伸缩性的方法和系统。对于具有跨场景的可变帧速率和可变快门角度的输入和输出视频序列,或者对于具有固定的输入帧速率和输入快门角度、但允许解码器以不同于对应输入值的输出帧速率和快门角度生成视频输出的输入视频序列,提供支持。还呈现了允许解码器在计算上更高效地对所允许的那些帧速率和快门角度中的特定向后兼容目标帧速率和快门角度进行解码的技术。



1. 一种非暂态处理器可读介质,其上存储有指令,其中,所述指令在被处理器执行时,使所述处理器:

接收一个或多个视频图片;以及

基于所述一个或多个视频图片生成编码视频流,所述编码视频流包括:

编码图片部分,所述编码图片部分包括所述一个或多个视频图片的编码;以及

元数据,其中,所述元数据包括:

快门间隔时间标度参数的编码,所述快门间隔时间标度参数指示一秒中经过的时间单位的数量;

快门间隔时钟周期参数的编码,所述快门间隔时钟周期参数指示以所述快门间隔时间标度参数的频率操作的时钟的时间单位的数量,

快门间隔持续时间标志的编码,所述快门间隔持续时间标志指示曝光持续时间信息是否对于所述编码图片部分中的所有时间子层都是固定的;并且

如果所述快门间隔持续时间标志指示所述曝光持续时间信息是固定的,则通过基于所述快门间隔时间标度参数和所述快门间隔时钟周期参数计算所述曝光持续时间值来针对所述编码图片部分中的所有所述时间子层的所述一个或多个视频图片的解码版本进行解码,否则

所述元数据包括一个或多个子层参数阵列,其中,与所述快门间隔时间标度参数组合的所述一个或多个子层参数阵列中的值被用来针对每个子层计算用于显示所述一个或多个视频图片的时间子层的解码版本的对应的子层曝光持续时间值,

其中,所述快门间隔时钟周期参数除以所述快门间隔时间标度参数指示曝光持续时间值,

其中,所述一个或多个子层参数阵列包括子层快门间隔时钟周期值阵列,每一个子层快门间隔时钟周期值指示以所述快门间隔时间标度参数的频率操作的所述时钟的时间单位的数量。

2. 如权利要求1所述的非暂态处理器可读介质,其中,所述元数据进一步包括帧重复值的编码,所述帧重复值指示应连续显示所述一个或多个视频图片中的解码图片的次数。

3. 如权利要求1所述的非暂态处理器可读介质,其中,所述元数据包括补充增强信息SEI消息传递部分或视频用户信息VUI消息传递部分。

4. 一种用于将视频比特流解码的方法,所述方法包括:

接收编码视频比特流,所述编码视频比特流包括图片数据部分和元数据部分,所述图片数据部分包括一个或多个视频图片的编码,并且所述元数据部分包括快门间隔参数,其中,所述快门间隔参数包括:

快门间隔时间标度参数,所述快门间隔时间标度参数指示一秒中经过的时间单位的数量;

快门间隔时钟周期参数,所述快门间隔时钟周期参数指示以所述快门间隔时间标度参数的频率操作的时钟的时间单位的数量;

快门间隔持续时间标志,所述快门间隔持续时间标志指示曝光持续时间信息是否对于所述图片数据部分中的所有时间子层都是固定的;并且

如果所述快门间隔持续时间标志指示所述曝光持续时间信息是固定的,则通过基于所

述快门间隔时间标度参数和所述快门间隔时钟周期参数计算曝光持续时间值来针对对所述图片数据部分中的所有所述时间子层的所述一个或多个视频图片的解码版本进行解码, 否则

所述元数据部分包括一个或多个子层参数阵列, 其中, 与所述快门间隔时间标度参数组合的所述一个或多个子层参数阵列中的值被用来针对每个子层计算用于显示所述一个或多个视频图片的时间子层的解码版本的对应的子层曝光持续时间值; 以及

基于所述快门间隔参数将所述一个或多个视频图片解码,

其中, 所述一个或多个子层参数阵列包括子层快门间隔时钟周期值阵列, 每一个子层快门间隔时钟周期值指示以所述快门间隔时间标度参数的频率操作的所述时钟的时间单位的数量,

其中, 所述曝光持续时间值以所述快门间隔时钟周期参数除以所述快门间隔时间标度参数的商来计算。

5. 如权利要求4所述的方法, 其中, 所述元数据部分包括补充增强信息SEI消息传递部分或视频用户信息VUI消息传递部分。

6. 一种用于将视频比特流编码的方法, 所述方法包括:

接收一个或多个视频图片和相关联的快门间隔信息; 以及

生成编码视频比特流, 所述编码视频比特流包括图片数据部分和元数据部分, 所述图片数据部分包括所述一个或多个视频图片的编码, 并且所述元数据部分包括用于所述一个或多个视频图片的快门间隔参数, 其中, 所述快门间隔参数包括:

快门间隔时间标度参数, 所述快门间隔时间标度参数指示一秒中经过的时间单位的数量;

快门间隔时钟周期参数, 所述快门间隔时钟周期参数指示以所述快门间隔时间标度参数的频率操作的时钟的时间单位的数量;

快门间隔持续时间标志, 所述快门间隔持续时间标志指示曝光持续时间信息是否对于所述图片数据部分中的所有时间子层都是固定的; 并且

如果所述快门间隔持续时间标志指示所述曝光持续时间信息是固定的, 则通过基于所述快门间隔时间标度参数和所述快门间隔时钟周期参数计算曝光持续时间值来针对对所述图片数据部分中的所有所述时间子层的所述一个或多个视频图片的解码版本进行解码, 否则

所述元数据部分包括一个或多个子层参数阵列, 其中, 与所述快门间隔时间标度参数组合的所述一个或多个子层参数阵列中的值被用来针对每个子层计算用于显示所述一个或多个视频图片的时间子层的解码版本的对应的子层曝光持续时间值,

其中, 所述一个或多个子层参数阵列包括子层快门间隔时钟周期值阵列, 每一个子层快门间隔时钟周期值指示以所述快门间隔时间标度参数的频率操作的所述时钟的时间单位的数量,

其中, 所述曝光持续时间值以所述快门间隔时钟周期参数除以所述快门间隔时间标度参数的商来计算。

7. 如权利要求6所述的方法, 其中, 所述元数据部分包括补充增强信息SEI消息传递部分或视频用户信息VUI消息传递部分。

帧速率可伸缩视频编码

[0001] 本申请是申请日为2020年3月11日、于2020年9月24日进入中国国家阶段的申请号为202080002086.5、发明名称为“帧速率可伸缩视频编码”的中国发明专利申请的分案申请。

[0002] 相关申请的交叉引用

[0003] 本申请要求于2019年3月11日提交的美国临时申请第62/816,521号、于2019年5月21日提交的美国临时申请第62/850,985号、于2019年8月6日提交的美国临时申请第62/883,195号和于2019年9月24日提交的美国临时申请第62/904,744号的优先权的权益,所述美国临时申请中的每一个以引用方式以其全文并入。

技术领域

[0004] 本文件总体上涉及图像。更特别地,本发明的实施例涉及帧速率可伸缩(frame-rate scalable)视频编码。

背景技术

[0005] 如本文所使用的,术语“动态范围(DR)”可以涉及人类视觉系统(HVS)感知图像中的强度(例如,光亮度(luminance)、亮度(luma))的范围的能力,所述强度的范围例如是从最暗的灰色(黑色)到最亮的白色(高光)。从这个意义上说,DR与“参考场景的(scene-referred)”强度有关。DR还可以涉及显示设备充分或近似渲染特定宽度(breadth)的强度范围的能力。从这个意义上说,DR与“参考显示的(display-referred)”强度有关。在本文的描述中除非在任何一点明确指定特定的意义具有特定的意思,否则应推断为所述术语可以在任一意义上例如可互换地被使用。

[0006] 如本文所使用的,术语“高动态范围(HDR)”涉及跨越人类视觉系统(HVS)的14至15个数量级的DR宽度。在实践中,关于HDR,人类可以同时感知强度范围广泛宽度的DR可能会被稍微截短。

[0007] 在实践中,图像包括一个或多个颜色分量(例如,亮度Y以及色度Cb和Cr),其中,每个颜色分量由每像素n比特的精度表示(例如, $n=8$)。使用线性光亮度编码,其中 $n \leq 8$ 的图像(例如,彩色24比特JPEG图像)被视为标准动态范围(SDR)的图像,而其中 $n > 8$ 的图像可以被视为增强动态范围的图像。HDR图像还可以使用高精度(例如,16比特)浮点格式被存储和分发,诸如由工业光魔公司(Industrial Light and Magic)开发的OpenEXR文档格式。

[0008] 目前,视频高动态范围内容(诸如杜比实验室的杜比视界或蓝光中的HDR10)的分发被许多回放设备的性能限制在4K分辨率(例如, 4096×2160 或 3840×2160 等)和60帧/秒(fps)。在未来版本中,预计高达8K分辨率(例如, 7680×4320)和120fps的内容可用于分发和回放。期望未来的内容类型将与现有的回放设备兼容,以简化HDR回放内容生态系统,诸如杜比视界。理想地,内容生产者应能够采用和分发未来的HDR技术,而不必还获得和分发与现有HDR设备(诸如HDR10或杜比视界)兼容的内容的特殊版本。如发明人在此所理解的,期望用于视频内容(尤其是HDR内容)的可伸缩分发的改进技术。

[0009] 在本节中描述的方法是可以追寻的方法,但不一定是之前已经设想到或追寻的方法。因此,除非另有指示,否则不应假设本节所描述的任何方法仅凭其纳入本节就可称为现有技术。类似地,除非另有指示,否则关于一种或多种方法所认定的问题不应基于本节而被认为在任何现有技术中已被认识到。

附图说明

[0010] 在附图中以举例而非限制的方式来图示本发明的实施例,并且其中相似的附图标记指代相似的元素,并且在附图中:

[0011] 图1描绘了视频传输(video delivery)流水线的示例过程;

[0012] 图2描绘了根据本发明实施例的组合连续原始帧以便以目标快门角度渲染目标帧速率的示例过程;

[0013] 图3描绘了根据本发明实施例的在具有固定帧速率的容器中具有可变输入帧速率和可变快门角度的输入序列的示例表示;以及

[0014] 图4描绘了根据本发明实施例的具有向后兼容性的各种帧速率和快门角度下的时间可伸缩性的示例表示。

具体实施方式

[0015] 本文描述了涉及用于视频编码的帧速率可伸缩性(scalability)的示例实施例。在以下说明中,出于解释的目的,阐述了许多特定细节以便提供对本发明的各种实施例的透彻理解。然而,清楚的是,可以在没有这些具体细节的情况下实践本发明的各种实施例。在其他情况下,为了避免不必要地遮蔽、模糊或混淆本发明的实施例,没有详尽地描述众所周知的结构和设备。

[0016] 概述

[0017] 本文描述的示例实施例涉及视频编码中的帧速率可伸缩性。在实施例中,一种具有处理器的系统接收编码比特流(coded bitstream),所述编码比特流包括编码视频帧(coded video frames),其中,一个或多个编码帧以第一帧速率和第一快门角度(shutter angle)被编码。所述处理器接收第一标志,所述第一标志指示存在要以第二帧速率和第二快门角度被解码的一组编码帧,所述处理器从所述编码比特流访问用于所述一组编码帧的所述第二帧速率和所述第二快门角度的值,并且基于所述一组编码帧、所述第一帧速率、所述第一快门角度、所述第二帧速率和所述第二快门角度,生成所述第二帧速率和所述第二快门角度的解码帧(decoded frames)。

[0018] 在第二实施例中,一种具有处理器的解码器:

[0019] 接收编码比特流,所述编码比特流包括多组编码视频帧,其中,所述编码比特流中的所有编码视频帧以第一帧速率被编码;

[0020] 接收组合帧的数量N;

[0021] 接收基线帧速率的值;

[0022] 访问一组N个连续编码帧,其中,所述一组N个连续编码帧中的第i编码帧表示在编码器中基于第一快门角度和所述第一帧速率以所述基线帧速率和第i快门角度编码的最多i个输入视频帧的平均值,其中, $i=1,2,\dots,N$;

[0023] 从所述编码比特流或从用户输入访问第二帧速率和第二快门角度的值,以用于以所述第二帧速率和所述第二快门角度对所述一组N个连续编码帧进行解码;以及

[0024] 基于所述一组N个连续编码帧、所述第一帧速率、所述第一快门角度、所述第二帧速率和所述第二快门角度,生成所述第二帧速率和所述第二快门角度的解码帧。

[0025] 在第三实施例中,一种编码视频流结构包括:

[0026] 编码图片部分,所述编码图片部分包括对视频图片序列的编码;以及

[0027] 信令部分,所述信令部分包括对以下的编码:

[0028] 快门间隔时间标度参数,所述快门间隔时间标度参数指示一秒中经过的时间单位的数量;

[0029] 快门间隔时钟周期参数,所述快门间隔时钟周期参数指示以所述快门间隔时间标度参数的频率操作的时钟的时间单位的数量,

[0030] 其中,所述快门间隔时钟周期参数除以所述快门间隔时间标度参数指示曝光持续时间值;

[0031] 快门间隔持续时间标志,所述快门间隔持续时间标志指示曝光持续时间信息是否对于所述编码图片部分中的所有时间子层都是固定的;以及

[0032] 如果所述快门间隔持续时间标志指示所述曝光持续时间信息是固定的,则通过基于所述快门间隔时间标度参数和所述快门间隔时钟周期参数计算所述曝光持续时间值来针对所述编码图片部分中的所有所述时间子层对所述视频图片序列的解码版本进行解码,否则

[0033] 所述信令部分包括一个或多个子层参数阵列,其中,与所述快门间隔时间标度参数组合的所述一个或多个子层参数阵列中的值被用来针对每个子层计算用于显示所述视频图片序列的时间子层的解码版本的对应的子层曝光持续时间值。

[0034] 示例视频传输处理流水线

[0035] 图1描绘了传统视频传输流水线(100)的示例过程,所述传统视频传输流水线示出了从视频捕获到视频内容显示的各个阶段。使用图像生成块(105)来捕获或生成视频帧(102)序列。视频帧(102)可以被(例如,由数码相机)数字地捕获或者由计算机(例如,使用计算机动画)生成以提供视频数据(107)。替代性地,视频帧(102)可以由胶片相机捕获在胶片上。胶片被转换为数字格式以提供视频数据(107)。在制作阶段(110),对视频数据(107)进行编辑以提供视频制作流(112)。

[0036] 制作流(112)的视频数据然后在块(115)处被提供给处理器以进行后期制作编辑。块(115)后期制作编辑可以包括调节或修改图像的特定区中的颜色或明亮度(brightness),以根据视频创作者的创作意图来增强图像质量或实现图像的特定外观。这有时被称为“颜色调整(color timing)”或“颜色分级(color grading)”。可以在块(115)处执行其他编辑(例如,场景选择和排序、图像裁剪、添加计算机生成的视觉特效、抖动或模糊控制、帧速率控制等)以产生用于分发的作品的最终版本(117)。在后期制作编辑(115)期间,在参考显示器(125)上观看视频图像。在后期制作(115)之后,可以将最终作品(117)的视频数据传输到编码块(120),以便向下游传输到解码和回放设备,诸如电视机、机顶盒、电影院等。在一些实施例中,编码块(120)可以包括音频编码器和视频编码器,诸如由ATSC、DVB、DVD、蓝光和其他传输格式定义的那些,以生成编码比特流(122)。在接收器中,编码比

特流(122)由解码单元(130)解码,以生成表示信号(117)的相同或接近近似版本的解码信号(decoded signal)(132)。接收器可以附接到目标显示器(140),所述目标显示器可以具有与参考显示器(125)完全不同的特性。在这种情况下,显示管理块(135)可以用于通过生成显示器映射信号(display-mapped signal)(137)来将解码信号(132)的动态范围映射到目标显示器(140)的特性。

[0037] 可伸缩编码

[0038] 可伸缩编码已经是许多视频编码标准(诸如MPEG-2、AVC和HEVC)的一部分。在本发明的实施例中,可伸缩编码被扩展用于提高性能和灵活性,尤其是在可伸缩编码涉及非常高分辨率的HDR内容时。

[0039] 如本文所使用的,术语“快门角度”表示可调节的快门设置,所述快门设置控制胶片在每个帧间隔期间曝光的时间比例。例如,在实施例中

$$[0040] \quad \frac{\text{快门角度}}{360} = \frac{\text{曝光时间}}{\text{帧间隔}} \quad (1)$$

[0041] 所述术语来自传统的机械旋转快门;然而,现代的数码相机也可以通过电子方式调节其快门。电影摄影师可以使用快门角度来控制每一帧中记录的运动模糊或抖动的量。应注意,还可以使用像“曝光持续时间”、“快门间隔”和“快门速度”等替代性术语,以代替“曝光时间”。类似地,可以使用术语“帧持续时间”,以代替“帧间隔”。替代性地,可以用“1/帧速率”来替换“帧间隔”。曝光时间的值通常小于或等于帧的持续时间。例如,180度的快门角度指示曝光是帧持续时间的一半。在某些情况下,曝光时间可以大于编码视频的帧持续时间,例如,当编码帧速率为120fps并且编码和显示之前的相关视频内容的帧速率为60fps时。

[0042] 非限制性地,考虑在原始帧速率(例如,120fps)下以360度的快门角度拍摄(或生成)原始内容的实施例。然后,在接收设备中,可以通过原始帧的公平组合(例如,通过平均或本领域已知的其他操作)以等于或低于原始帧速率的各种帧速率渲染视频输出。

[0043] 组合过程可以用非线性编码信号(例如,使用 γ (gamma)、PQ或HLG)来执行,但是最佳图像质量是通过以下各项组合线性光域中的帧来获得的:首先将非线性编码信号转换成线性光表示,然后组合经转换的帧,并且最后用非线性传递函数对输出进行重新编码。此过程提供了比在非线性域中组合更准确的对物理相机曝光的模拟。

[0044] 一般而言,组合帧的过程可以用原始帧速率、目标帧速率、目标快门角度和要组合的帧的数量来表达,如:

$$[0045] \quad n_frames = (\text{target_shutter_angle}/360) * (\text{original_frame_rate}/\text{target_frame_rate}), \quad (2)$$

[0046] 其等价于

$$[0047] \quad \text{target_shutter_angle} = 360 * n_frames * (\text{target_frame_rate}/\text{original_frame_rate}), \quad (3)$$

[0048] 其中, n_frames 是组合帧的数量, $original_frame_rate$ 是原始内容的帧速率, $target_frame_rate$ 是要渲染的帧速率(其中, $target_frame_rate \leq original_frame_rate$),并且 $target_shutter_angle$ 指示期望的运动模糊的量。在此示例中, $target_shutter_angle$ 的最大值为360度并且对应于最大运动模糊。 $target_shutter_angle$ 的最小

值可以表示为 $360 * (\text{target_frame_rate} / \text{original_frame_rate})$ 并且对应于最小运动模糊。 n_frames 的最大值可以表示为 $(\text{original_frame_rate} / \text{target_frame_rate})$ 。应选择 target_frame_rate 和 $\text{target_shutter_angle}$ 的值,使得 n_frames 的值为非零整数。

[0049] 在原始帧速率为120fps的特殊情况下,等式(2)可以被重写为

$$[0050] \quad n_frames = \text{target_shutter_angle} / (3 * \text{target_frame_rate}), \quad (4)$$

[0051] 其等价于

$$[0052] \quad \text{target_shutter_angle} = 3 * n_frames * \text{target_frame_rate}. \quad (5)$$

[0053] 对于 $\text{original_frame_rate} = 120\text{fps}$ 的情况,表1中示出了 target_frame_rate 、 n_frames 和 $\text{target_shutter_angle}$ 的值之间的关系。在表1中,“NA”指示不允许目标帧速率和组合帧的数量的对应组合。

[0054] 表1:对于120fps的原始帧速率,目标帧速率、组合帧的数量和目标快门角度之间的关系。

目标帧速率 (fps)	组合帧数				
	5	4	3	2	1
	目标快门角度 (度)				
[0055] 24	360	288	216	144	72
30	NA	360	270	180	90
40	NA	NA	360	240	120
60	NA	NA	NA	360	180

[0056] 图2描绘了根据实施例的组合连续原始帧以便以目标快门角度渲染目标帧速率的示例过程。考虑120fps和360度快门角度下的输入序列(205),所述过程通过组合五个连续帧的集合中的三个输入帧(例如,前三个连续帧)并丢弃其他两个帧来生成24fps和216度快门角度下的输出视频序列(210)。应注意,在一些实施例中,(210)的输出帧-01可以通过组合替代性输入帧(205)(诸如帧1、3和5或者帧2、4和5等)而被生成;然而,预计组合连续帧将产生质量更好的视频输出。

[0057] 期望支持具有可变帧速率的原始内容,例如,以用于管理艺术和风格效果。还期望将原始内容的可变输入帧速率封装在具有固定帧速率的“容器”中,以简化内容制作、交换和分发。作为示例,呈现了关于如何在固定帧速率容器中表示可变帧速率视频数据的三个实施例。出于清楚的目的,但非限制性地,以下描述使用固定的120fps容器,但是所述方法可以轻松地被扩展到替代性帧速率容器。

[0058] 第一实施例(可变帧速率)

[0059] 第一实施例是对封装在具有恒定帧速率的容器中的具有可变(非恒定)帧速率的原始内容的明确描述。例如,对于不同的场景具有不同帧速率(例如,24、30、40、60或120fps)的原始内容可以封装在具有120fps的恒定帧速率的容器中。对于此示例,每个输入帧可以被复制5x、4x、3x、2x或1x次,以将其封装到共同的120fps容器中。

[0060] 图3描绘了在具有固定帧速率的编码比特流B中表示的具有可变帧速率和可变快门角度的输入视频序列A的示例。然后,在解码器中,解码器以期望的帧速率和快门角度重建输出视频序列C,所述期望的帧速率和快门角度可以因场景而异。例如,如图3所描绘的,为了构建序列B,输入帧中的一些被复制,一些被原样编码(没有复制),并且一些被拷贝四

次。然后,为了构建序列C,从每组复制帧中选择任何一帧来生成输出帧,从而与原始帧速率和快门角度相匹配。

[0061] 在此实施例中,元数据被插入比特流中以指示原始(基本)帧速率和快门角度。元数据可以使用诸如序列参数集(SPS)、图片参数集(PPS)、切片或图块组标题等高级语法用信号表示。元数据的存在使编码器和解码器能够执行有益的功能,如:

[0062] a) 编码器可以忽略复制的帧,由此提高编码速度并简化处理。例如,可以使用SKIP模式和参考帧的LIST 0中的参考索引0对复制的帧中的所有编码树单元(CTU)进行编码,所述参考帧是指从中拷贝复制的帧的解码帧。

[0063] b) 解码器可以绕过对复制帧的解码,由此简化处理。例如,比特流中的元数据可以指示帧是解码器可以通过拷贝并且无需对新的帧进行解码来再现的先前经解码帧的副本。

[0064] c) 回放设备可以通过指示基本帧速率来优化下游处理(例如,通过调节帧速率转换或降噪算法)。

[0065] 此实施例使终端用户能够以内容创作者预期的帧速率观看渲染的内容。此实施例不提供与不支持容器的帧速率(例如,120fps)的设备的向后兼容性。

[0066] 表2和表3描绘了用于序列参数集和图块组标题的原始字节序列有效载荷(RBSB)的示例语法,其中,提出的新语法元素以高亮字体描绘。其余语法遵循多功能视频编解码器(VVC)的建议规范中的语法(参考文献[2])。

[0067] 作为示例,在SPS(参见表2)中,可以添加标志来启用可变帧速率。

[0068] `sps_vfr_enabled_flag`等于1指定编码视频序列(CVS)可以包含可变帧速率内容。`sps_vfr_enabled_flag`等于0指定CVS包含固定帧速率内容。

[0069] 在`tile_group_header()`中(参见表3),

[0070] `tile_group_vrf_info_present_flag`等于1指定语法元素`tile_group_true_fr`和`tile_group_shutterangle`存在于语法中。`tile_group_vrf_info_present_flag`等于0指定语法元素`tile_group_true_fr`和`tile_group_shutterangle`在语法中不存在。当`tile_group_vrf_info_present_flag`不存在时,推断其为0。

[0071] `tile_group_true_fr`指示此比特流中携带的视频数据的真实帧速率。

[0072] `tile_group_shutterangle`指示对应于此比特流中携带的视频数据的真实帧速率的快门角度。

[0073] `tile_group_skip_flag`等于1指定当前图块组是从另一图块组拷贝的。`tile_group_skip_flag`等于0指定当前图块组不是从另一图块组拷贝的。

[0074] `tile_group_copy_pic_order_cnt_lsb`指定当`tile_group_skip_flag`设置为1时当前图片拷贝自的先前解码图片的图片顺序计数模`MaxPicOrderCntLsb`。

[0075] 表2:具有可变帧速率的内容的示例参数集RBSP语法

	seq_parameter_set_rbsp() {	描述符
	sps_max_sub_layers_minus1	u(3)
	sps_reserved_zero_5bits	u(5)
	profile_tier_level(sps_max_sub_layers_minus1)	
	sps_seq_parameter_set_id	ue(v)
	...	
[0076]	sps_vfr_enabled_flag	u(1)
	...	
	sps_extension_flag	u(1)
	if(sps_extension_flag)	
	while(more_rbsp_data())	
	sps_extension_data_flag	u(1)
	rbsp_trailing_bits()	
	}	

[0077] 表3:支持具有可变帧速率的内容的图块组标题语法的示例

tile_group_header() {	描述符
tile_group_pic_parameter_set_id	ue(v)
if(NumTilesInPic > 1) {	
tile_group_address	u(v)
num_tiles_in_tile_group_minus1	ue(v)
}	
tile_group_type	ue(v)
tile_group_pic_order_cnt_lsb	u(v)
if(sps_vfr_enabled_flag) {	
tile_group_vfr_info_present_flag	u(1)
if(tile_group_vfr_info_present_flag) {	

[0078]

tile_group_true_fr	u(9)
tile_group_shutterangle	u(9)
}	
tile_group_skip_flag	u(1)
}	
if(tile_group_skip_flag)	
tile_group_copy_pic_order_cnt_lsb	u(v)
else{	
ALL OTHER TILE_GROUP_SYNTAX	
}	
if(num_tiles_in_tile_group_minus1 > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < num_tiles_in_tile_group_minus1; i++)	
entry_point_offset_minus1[i]	u(v)
}	
byte_alignment()	
}	

[0079] 第二实施例——固定帧速率容器

[0080] 第二实施例实现了这样的使用情况：其中，具有固定帧速率和快门角度的原始内容可以由解码器以替代性帧速率和可变模拟快门角度来渲染，诸如图2所图示的。例如，在原始内容的帧速率为120fps并且快门角度为360度的情况下（意味着快门打开1/120秒），解码器可以渲染出小于或等于120fps的多个帧速率。例如，如表1所描述的，为了用216度模拟

快门角度对24fps进行解码,解码器可以组合三个解码帧并以24fps显示。表4扩展了表1并图示了如何组合不同数量的编码帧以便以输出目标帧速率和期望目标快门角度渲染。可以通过简单的像素平均、通过加权像素平均(其中,来自某帧的像素可以比其他帧的像素被加权更多,并且所有权重之和共计为一)或者通过本领域已知的其他滤波插值方案来执行对帧的组合。在表4中,函数 $Ce(a,b)$ 表示编码帧a至b的组合,其中,所述组合可以通过平均、加权平均、滤波等来执行。

[0081] 表4:以120fps组合输入帧以生成目标fps和快门角度值下的输出帧的示例

输入	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
编码	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
解码 120fps										
@360	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10
解码 60fps										
@360	Ce(1,2)		Ce(3,4)		Ce(5,6)		Ce(7,8)		Ce(9,10)	
@180	e1		e3		e5		e7		e9	
解码 40fps										
@360	Ce(1,3)			Ce(4,6)			Ce(7,9)			Ce(10,12)
@240	Ce(1,2)			Ce(4,5)			Ce(7,8)			Ce(10,11)
@120	e1			e4			e7			e10
解码 30fps										
@360	Ce(1,4)				Ce(5,8)				Ce(9,12)	
@270	C(1,3)				Ce(5,7)				Ce(9,11)	
@180	Ce(1,2)				Ce(5,6)				Ce(9,10)	
@90	e1				e5				e9	
解码 24fps										

[0082]

@360	Ce(1,5)				Ce(6,10)					
@288	Ce(1,4)				Ce(6,9)					
@216	Ce(1,3)				Ce(6,8)					
@144	Ce(1,2)				Ce(6,7)					
@72	e1				e6					

[0083] 当目标快门角度的值小于360度时,解码器可以组合不同的解码帧集合。例如,根据表1,考虑120fps@360度的原始流,为了生成40fps和240度快门角度下的流,解码器需要组合三个可能的帧中的两个帧。因此,解码器可以组合第一帧和第二帧或者第二帧和第三帧。对要组合哪些帧的选择可以用表示为以下等式的“解码相位(decoding phase)”来描述:

[0084] $\text{decode_phase} = \text{decode_phase_idx} * (360 / n_frames)$, (6)

[0085] 其中, decode_phase_idx 表示索引值在 $[0, n_frames_max - 1]$ 中的序列帧的集合内的偏移索引,其中, n_frames 由等式(2)给出,并且

[0086] $n_frames_max = \text{orig_frame_rate} / \text{target_frame_rate}$. (7)

[0087] decode_phase_idx 的范围通常为 $[0, n_frames_max - n_frames]$ 。例如,对于在120fps和360度快门角度下的原始序列,对于在240度快门角度下40fps的目标帧速率, $n_frames_max = 120 / 40 = 3$ 。根据等式(2), $n_frames = 2$,因此 decode_phase_idx 的范围为 $[0, 1]$ 。因此, $\text{decode_phase_idx} = 0$ 指示选择索引为0和1的帧,并且 $\text{decode_phase_idx} = 1$ 指示选择索引为1和2的帧。

[0088] 在此实施例中,内容创作者预期的渲染的可变帧速率可以用信号表示为元数据,诸如补充增强信息(SEI)消息或视频可用性信息(VUI)。可选地,渲染的帧速率可以由接收器或用户控制。表5中示出了指定内容创作者的优选帧速率和快门角度的帧速率转换SEI消息传递的示例。SEI消息还可以指示组合帧是在编码信号域(例如, γ (gamma)、PQ等)中还是在线性光域中执行。应注意,除了解码器图片缓冲器(DPB)之外,后处理还需要帧缓冲器。SEI消息可以指示需要多少额外的帧缓冲器或者指示用于组合帧的某种替代性方法。例如,为了降低复杂性,可以以降低的空间分辨率来重新组合帧。

[0089] 如表4所描绘的,在帧速率和快门角度的某些组合下(例如,在30fps和360度下或在24fps和288度或360度下),解码器可能需要组合多于三个解码帧,这增加了解码器所需的缓冲器空间的数量。为了减轻解码器中额外缓冲器空间的负担,在一些实施例中,帧速率和快门角度的某些组合可能不能使用允许的解码参数的集合(例如,通过设置适当的编码配置文件和级别)。

[0090] 再次考虑以24fps回放的情况作为示例,解码器可以决定显示要以120fps输出帧速率显示的同帧五次。这与以24fps输出帧速率示出该帧一次完全相同。保持恒定的输出帧速率的优点在于显示器可以以恒定的时钟速度运行,这使得所有硬件更加简单。如果显示器可以动态地改变时钟速度,则只示出该帧一次(持续1/24秒)可能会更合理,而不是重复同一帧五次(每次1/120秒)。前一种方法可以产生略高的图片质量、更好的光学效率或更好的功率效率。类似的考虑也适用于其他帧速率。

[0091] 表5描绘了根据实施例的帧速率转换SEI消息传递语法的示例。

[0092] 表5:允许帧速率转换的SEI消息语法的示例

	framerate_conversion (payloadSize) {	描述符
	framerate_conversion_cancel_flag	u(1)
	if(!frame_conversion_cancel_flag) {	
	base_frame_rate	u(9)
	base_shutter_angle	u(9)
	decode_phase_idx_present_flag	u(1)
[0093]	if(decode_phase_idx_present_flag) {	
	decode_phase_idx	u(3)
	}	
	conversion_domain_idc	u(1)
	num_frame_buffer	u(3)
	framerate_conversion_persistence_flag	u(1)
	}	
	}	

[0094] framerate_conversion_cancel_flag等于1指示SEI消息按输出顺序取消任何先前帧速率转换SEI消息的留存。framerate_conversion_cancel_flag等于0指示随后是帧速率转换信息。

[0095] base_frame_rate指定期望的帧速率。

[0096] base_shutter_angle指定期望的快门角度。

[0097] decode_phase_idx_present_flag等于1指定解码相位信息存在。decode_phase_idx_present_flag等于0指定解码相位信息不存在。

[0098] decode_phase_idx指示索引值为0..(n_frames_max-1)的序列帧的集合内的偏移索引,其中,n_frames_max=120/base_frame_rate。decode_phase_idx的值应在0..(n_frames_max-n_frames)的范围中,其中,n_frames=base_shutter_angle/(3*base_frame_rate)。当decode_phase_idx不存在时,推断其为0。

[0099] conversion_domain_idc等于0指定在线性域中执行帧组合。conversion_domain_idc等于1指定在非线性域中执行帧组合。

[0100] num_frame_buffers指定帧缓冲器的附加数量(DPB不算在内)。

[0101] framerate_conversion_persistence_flag指定当前层的帧速率转换SEI消息的留存。framerate_conversion_persistence_flag等于0指定帧速率转换SEI消息仅适用于当前解码图片。设picA为当前图片。framerate_conversion_persistence_flag等于1指定帧速率转换SEI消息按输出顺序为当前层留存,直到以下条件中的一个或多个条件为真:

[0102] -当前层的新编码分层视频序列(CLVS)开始。

[0103] -比特流结束。

[0104] -输出访问单元中的当前层中的图片picB,所述图片picB包含适用于当前层的帧速率转换SEI消息,对于所述图片picB,PicOrderCnt(picB)大于PicOrderCnt(picA),其中,PicOrderCnt(picB)和PicOrderCnt(picA)分别是紧接在调用用于picB的图片顺序计数的解码过程之后的picB和picA的PicOrderCntVal值。

[0105] 第三实施例——以多个快门角度编码的输入

[0106] 第三实施例是允许从比特流中提取子帧速率从而支持向后兼容性的编码方案。在HEVC中,这通过时间可伸缩性实现。通过为解码帧的temporal_id语法元素分配不同的值,

实现了时间层可伸缩性。由此可以简单地在temporal_id值的基础上提取比特流。然而,时间可伸缩性的HEVC风格的方法并未实现以不同的快门角度渲染输出帧速率。例如,从120fps原始帧速率中提取的60fps基本帧速率将始终具有180度的快门角度。

[0107] 在ATSC 3.0中,描述了替代性方法,其中,具有360度快门角度的60fps下的帧被模拟为两个120fps帧的加权平均值。模拟的60fps帧被分配有temporal_id值0,并与分配有temporal_id值1的交替的原始120fps帧组合。当需要60fps时,解码器只需要对temporal_id为0的帧进行解码。当需要120fps时,解码器可以从每个对应的temporal_id=0的帧(即,模拟的60fps帧)的缩放版本中减去每个temporal_id=1的帧(即,120fps帧),以恢复没有被明确传送的对应的原始120fps帧,由此重新构成所有的原始120fps帧。

[0108] 在本发明的实施例中,描述了以向后兼容(BC)的方式支持多个目标帧速率和目标快门角度的新算法。提出在若干快门角度下以基本帧速率对原始120fps内容进行预处理。然后,在解码器处,可以简单地得到各种其他快门角度下的其他帧速率。ATSC 3.0方法可以被认为是所提出方案的特殊情况,其中,temporal_id=0的帧以60fps@360快门角度携带帧,并且temporal_id=1的帧以60fps@180快门角度携带帧。

[0109] 作为第一示例,如图4所描绘的,考虑120fps和360快门角度下的输入序列,所述输入序列用于以40fps的基本帧速率以及120度、240度和360度的快门角度对序列进行编码。在此方案中,编码器通过组合原始输入帧中的多达三个帧来计算新的帧。例如,通过组合输入帧1和2来生成表示40fps和240度下的输入的编码帧2(En-2),并且通过将帧En-2组合到输入帧3来生成表示40fps和360度下的输入的编码帧3(En-3)。在解码器中,为了重建输入序列,通过从帧En-2中减去帧En-1来生成解码帧2(Dec-2),并且通过从帧En-3中减去帧En-2来生成解码帧3(Dec-3)。这三个解码帧表示120fps的基本帧速率和360度快门角度下的输出。如表6所描绘的,可以使用解码帧来外推附加帧速率和快门角度。在表6中,函数Cs(a,b)表示输入帧a至b的组合,其中,所述组合可以通过平均、加权平均、滤波等来执行。

[0110] 表6:基线为40fps的帧组合的示例

输入帧 120 fps@360	s1	s2	s3	s4	s5	s6	s7	s8	s9
编码 帧 120 fps	e1 = s1	e2 = Cs(1,2)	e3 = Cs(1,3)	e4 = s4	e5 = Cs(4,5)	e6 = Cs(4,6)	e7 = s7	e8 = Cs(7,8)	e9 = Cs(7,9)
解码 120 fps									
@360	e1 = s1	e2 - e1 = s2	e3 - e2 = s3	e4 = s4	e5 - e4 = s5	e6 - e4 = s6	e7 = s7	e8 - e7 = s8	e9 - e8 = s9
解码 60 fps									
@360	e2		e3 - e2 + e4 = Cs(3,4)		e6 - e4 = Cs(5,6)		e8 = Cs(7,8)		e9 - e8 + e10
@180	e1		e3 - e2 = s3		e5 - e4 = s5		e7		e9 - e8
解码 40 fps									
@360	e3 = Cs(1,3)			e6			e9		
@240	e2 = Cs(1,2)			e5			e8		
@120	e1 = s1			e4			e7		
解码 30 fps									
@360	e3 + e4 = Cs(1,4)			e6 - e4 + e8 = Cs(5,8)			e9 - e8 + e12		
@270	e3 = Cs(1,3)			e6 - e5 + e7 = Cs(5,7)			e9 - e8 + e11		
@180	e2 = Cs(1,2)			e6 - e4 = Cs(5,6)			e9 - e8 + e10		
@90	e1			e5 - e4 = s5			e9 - e8		
解码 24 fps									
@360	e3 + e5 = Cs(1,5)			e6 - e5 + e9 + e10 = Cs(6,10)					
@288	e3 + e4 = Cs(1,4)			e6 - e5 + e9 = Cs(6,9)					
@216	e3 = Cs(1,3)			e6 - e5 + e8 = Cs(6,8)					
@144	e2 = Cs(1,2)			e6 - e5 + e7 = Cs(6,7)					
@72	e1 = s1			e6 - e5 = s6					

[0112] 此方法的优点在于如表6所描绘的,可以对所有40fps版本进行解码,而不需要任何进一步的处理。另一个优点在于可以在各种快门角度下得到其他帧速率。例如,考虑了以30fps和360的快门角度解码的解码器。根据表4,输出对应于由 $Ce(1,4) = Cs(1,4)$ 、 $Cs(5,8)$ 、 $Cs(9,12)$ 等生成的帧的序列,所述序列还与表6中描绘的解码序列相匹配;然而,在表6中, $Cs(5,8) = e6 - e4 + e8$ 。在实施例中,可以使用查找表(LUT)来定义需要如何组合解码帧以生成指定的输出帧速率和仿真快门角度下的输出序列。

[0113] 在另一个示例中,提出在编码器中组合多达五个帧,以便简化在72度、144度、216度、288度和360度的快门角度下对24fps基本层的提取,如下所示。对于在传统电视上以24fps最佳呈现的电影内容而言,这是期望的。

[0114] 表7:基线为24fps的帧组合的示例

输入 帧 120 fps @360	s1	s2	s3	s4	s5	s6	s7	s8	s9
编码 帧	e1 = s1	e2 = Cs(1,2)	e3 = Cs(1,3)	e4 = Cs(1,4)	e5 = Cs(1,5)	e6 = s6	e7 = Cs(6,7)	e8 = Cs(6,8)	e9 = Cs(6,9)
解码 120 fps									
@360	e1	e2 - e1	e3 - e2	e4 - e3	e5 - e4	e6	e7 - e6	e8 - e7	e9 - e8
解码 60 fps									
@360	e2		e4 - e2		e5 - e4 + e6		e8 - e6		e10 - e8
@180	e1		e3 - e2		e5 - e4		e7 - e6		e9 - e8
解码 40 fps									
@360		e3			e5 - e3 + e6			e9 - e6	
@240		e2			e5 - e3			e8 - e6	
@120		e1			e4 - e3			e7 - e6	
解码 30 fps									
@360			e4			e5 - e4 + e8			e10 - e8 + e12
@270			e3			e5 - e4 + e7			e10 - e8 + e11
@180			e2			e5 - e4 + e6			e10 - e8
@90			e1			e5 - e4			e9 - e8
解码 24 fps									
@360				e5				e10	
@288				e4				e9	
@216				e3				e8	
@144				e2				e7	
@72				e1				e6	

[0115] 如表7所描绘的,如果解码帧速率与基线帧速率(24fps)相匹配,则在每组五个帧(例如,e1至e5)中,解码器可以仅选择期望的快门角度下的一个帧(例如,对于144度的快门角度为e2)。为了以不同的帧速率和特定快门角度进行解码,解码器将需要确定如何适当地组合(例如,通过加法或减法)解码帧。例如,为了以30fps和180度的快门角度进行解码,可以遵循以下步骤:

[0117] a) 解码器可以考虑假设的编码器以120fps和360度传送而不对向后兼容性做任何考虑,然后,根据表1,解码器需要组合4个帧中的2个帧以便以期望的帧速率和快门角度生成输出序列。例如,如表4所描绘的,所述序列包括 $Ce(1,2) = Avg(s1,s2)$ 、 $Ce(5,6) = Avg(s5,s6)$ 等,其中, $Avg(s1,s2)$ 可以表示帧s1和s2的平均。

[0118] b) 考虑根据定义,编码帧可以表示为 $e1 = s1$ 、 $e2 = Avg(s1,s2)$ 、 $e3 = Avg(s1,s3)$ 等,可以容易地得到步骤a)中的帧的序列还可以表示为:

[0119] $Ce(1,2) = Avg(s1,s2) = e2$

[0120] • $Ce(5,6) = Avg(s5,s6) = Avg(s1,s5) - Avg(s1,s4) + s6 = e5 - e4 + e6$

[0121] 等等。

[0122] 如前,可以预先计算解码帧的适当组合,并且可以将其用作LUT。

[0123] 所提出方法的优点在于其为内容创作者和用户两者都提供了选项;即,其实现了导演/编辑选择和用户选择。例如,编码器中的预处理内容允许用各种快门角度创建基本帧速率。每个快门角度可以分配有在范围 $[0, (n_frames - 1)]$ 中的temporal_id值,其中,n_frames的值等于120除以基本帧速率。(例如,对于24fps的基本帧速率,temporal_id在范围 $[0, 4]$ 中。)可以作出选择,以优化压缩效率,或者是出于美观的原因。在一些使用情况下,例如,对于过顶流式传输(over the top streaming),可以对具有不同基本层的多个比特流进行编码和存储,并将其提供给用户以进行选择。

[0124] 在所公开方法的第二示例中,可以支持多个向后兼容的帧速率。理想地,可能会希望能够以24帧/秒进行解码以获得24fps基本层,以30帧/秒进行解码以获得30fps序列,以60帧/秒进行解码以获得60fps序列等。如果未指定目标快门角度,则建议在源帧速率和目标帧速率允许的那些快门角度中的默认目标快门角度尽可能接近180度。例如,对于表7中描绘的值,120、60、40、30和24的fps的优选目标快门角度是360度、180度、120度、180度和216度。

[0125] 从以上示例可以观察到,对如何对内容进行编码的选择会影响对特定基本层帧速率进行解码的复杂性。本发明的一个实施例是基于期望的基本层帧速率自适应地选择编码方案。例如,对于电影内容,可以是24fps,而对于体育内容,可以是60fps。

[0126] 下文以及表8和表9中示出了本发明的BC实施例的示例语法。

[0127] 在SPS(表8)中,添加了两个语法元素:sps_hfr_BC_enabled_flag和sps_base_framerate(如果sps_hfr_BC_enabled_flag被设置为等于1)。

[0128] sps_hfr_BC_enabled_flag等于1指定在编码视频序列(CVS)中启用具有向后兼容性的高帧速率。sps_hfr_BC_enabled_flag等于0指定在CVS中不启用具有向后兼容性的高帧速率。

[0129] sps_base_framerate指定当前CVS的基本帧速率。

[0130] 在图块组标题中,如果sps_hfr_BC_enabled_flag设置为1,则语法number_avg_frames在比特流中被发送。

[0131] number_avg_frames指定被组合以便以基本帧速率生成当前图片的在最高帧速率(例如,120fps)下的帧的数量。

[0132] 表8:各种快门角度下的输入的示例RBSP语法

seq_parameter_set_rbsp() {	描述符
sps_max_sub_layers_minus1	u(3)
sps_reserved_zero_5bits	u(5)
profile_tier_level(sps_max_sub_layers_minus1)	
sps_seq_parameter_set_id	ue(v)
...	
sps_hfr_BC_enabled_flag	u(1)
if(sps_hfr_BC_enabled_flag) {	u(1)

[0133]

sps_base_frame_rate	u(9)
}	
...	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data())	
sps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

[0134] 表9: 各种快门角度下的输入的示例图片参数集RBSB语法

pic_parameter_set_rbsp() {	描述符
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	ue(v)
...	
if(sps_hfr_BC_enabled_flag)	
number_avg_frames	
...	se(v)
rbsp_trailing_bits()	
}	

[0136] 第二实施例的变化(固定帧速率)

[0137] HEVC(H.265) 编码标准(参考文献[1])和正在开发的通用视频编码标准(通常称为VVC,参见参考文献[2])定义语法元素pic_struct,所述语法元素指示图片应显示为帧还是一个或多个字段以及解码图片是否应被重复。为了便于参考,附录中提供了来自HEVC的表D.2“pic_struct解释”的拷贝。

[0138] 重要的是应注意,如发明人所理解的,当使用固定帧速率编码容器时,现有的pic_struct语法元素可以仅支持内容帧速率的特定子集。例如,当使用60fps的固定帧速率容器时,在fixed_pic_rate_within_cvs_flag等于1时,现有的pic_struct语法可以通过使用帧

双倍增 (frame doubling) 来支持30fps,并且以每隔一帧交替组合的方式使用帧双倍增和帧三倍增 (frame tripling) 来支持24fps。然而,当使用120fps的固定帧速率容器时,当前pic_struct语法不能支持24fps或30fps的帧速率。为了缓解这个问题,提出了两种新方法:一种方法是HEVC版本的扩展,而另一种不是。

[0139] 方法1:没有向后兼容性的pic_struct

[0140] VVC仍在开发中,因此可以以最大自由度设计语法。在实施例中,在pic_struct中,提出删除帧双倍增和帧三倍增的选项、使用pic_struct的特定值来指示任意帧重复,并且添加指定要重复的帧数量的新语法元素num_frame_repetition_minus2。下表中描述了所提出语法的示例,其中,表10表示相对于HEVC中的表D.2.3的变化,并且表11表示附录所示出的表D.2的变化。

[0141] 表10:示例图片定时SEI消息语法,方法1

	pic_timing (payloadSize) {	描述符
	if (frame field info present flag) {	
	pic_struct	u(4)
	if(pic_struct == 7)	u(4)
[0142]	num_frame_repetition_minus2	u(4)
	source_scan_type	u(2)
	duplicate_flag	u(1)
	}	
 (如正本)	

[0143] num_frame_repetition_minus2加2指示当fixed_pic_rate_within_cvs_flag等于1时,帧应在显示器上连续显示num_frame_repetition_minus2加2次,其中帧刷新间隔等于DpbOutputElementalInterval[n],如等式E-73所给出的。

[0144] 表11:根据方法1的经修订pic_struct的示例

值	指示的图片显示	限制
0	(逐行) 帧	field_seq_flag 应等于 0
1	顶部字段	field_seq_flag 应等于 1
2	底部字段	field_seq_flag 应等于 1
3	顶部字段、底部字段, 按此顺序	field_seq_flag 应等于 0
4	底部字段、顶部字段, 按此顺序	field_seq_flag 应等于 0
5	顶部字段、底部字段、重复的顶部字段, 按此顺序	field_seq_flag 应等于 0
6	底部字段、顶部字段、重复的底部字段, 按此顺序	field_seq_flag 应等于 0
[0145] 7	帧重复	field_seq_flag 应等于 0 fixed_pic_rate_within_cvs_flag 应等于 1
8	按输出顺序与前一个底部字段配对的顶部字段	field_seq_flag 应等于 1
9	按输出顺序与前一个顶部字段配对的底部字段	field_seq_flag 应等于 1
10	按输出顺序与下一个底部字段配对的顶部字段	field_seq_flag 应等于 1
11	按输出顺序与下一个顶部字段配对的底部字段	field_seq_flag 应等于 1

[0146] 方法2:pic_struct的HEVC版本的扩展版本

[0147] 已经部署了AVC和HEVC解码器,因此可能会期望简单扩展现有的pic_struct语法,而不删除旧的选项。在实施例,添加了新的“帧重复扩展”值pic_struct=13和新的语法元素num_frame_repetition_minus4。表12和表13中描述了所提出语法的示例。对于pic_struct值0至12,所提出语法与表D.2中的语法相同(如附录所示),因此为了简洁起见,省略了这些值。

[0148] 表12:示例图片定时SEI消息语法,方法2

pic_timing (payloadSize) {	描述符
if (frame_field_info_present flag) {	
pic_struct	u(4)
if (pic_struct == 13)	u(4)
num_frame_repetition_minus4	u(4)
source_scan_type	u(2)
duplicate_flag	u(1)
}	
... (如正本)	

[0149] [0150] num_frame_repetition_minus4加4指示当fixed_pic_rate_within_cvs_flag等于1时,帧应在显示器上连续显示num_frame_repetition_minus4加4次,其中帧刷新间隔等于DpbOutputElementalInterval[n],如等式E-73所给出的。

[0151] 表13:经修订pic_struct的示例,方法2

	值	指示的图片显示	限制
[0152]	0 至 12	如表 D.2 中	如表 D.2 中
	13	帧重复扩展	field_seq_flag 应等于 0 fixed_pic_rate_within_cvs_flag 应等于 1

[0153] 在HEVC中,参数frame_field_info_present_flag存在于视频可用性信息(VUI)中,但语法元素pic_struct、source_scan_type和duplicate_flag在pic_timing() SEI消息中。在实施例,提出将所有相关语法元素连同frame_field_info_present_flag一起移动到VUI。表14中描绘了所提出语法的示例。

[0154] 表14:支持经修订pic_struct语法元素的示例VUI参数语法

	vui_parameters() {	描述符
	...	u(1)
	field_seq_flag	u(1)
	frame_field_info_present_flag	u(1)
[0155]	if (frame field info present flag) {	
	pic_struct	u(4)
	source_scan_type	u(2)
	duplicate_flag	u(1)
	}	
	...	
	}	

[0156] 快门角度信息的替代性信令

[0157] 当处理可变帧速率时,期望确定期望的帧速率和期望的快门角度两者。在现有的视频编码标准中,“视频可用性信息”(VUI)提供了适当显示视频内容所必不可少的信息,诸如纵横比、基色(colour primaries)、色度子采样等。如果固定pic速率设置为1,则VUI还可以提供帧速率信息;然而,不支持快门角度信息。实施例允许不同的快门角度用于不同的时间层,并且解码器可以使用快门角度信息来改善显示器上的最终外观。

[0158] 例如,HEVC支持本质上使用丢帧技术从较高的帧速率变为较低的帧速率的时间子层。其主要问题在于有效的快门角度随着每次帧丢弃而减小。作为示例,通过每隔一帧丢弃一帧,可以从120fps视频得到60fps;通过丢弃4个帧中的3个帧,可以得到30fps;并且通过丢弃5个帧中的4个帧,可以得到24fps。假设120Hz的完全的360度快门,通过简单的丢帧,60fps、30fps和24fps的快门角度分别为180度、90度和72度[3]。经验表明,低于180度的快门角度通常是不可接受的,尤其是在帧速率低于50Hz的情况下。通过提供快门角度信息,例如,如果期望显示器从针对每个时间层具有减小的快门角度的120Hz视频产生电影效果,则可以应用智能技术来改善最终外观。

[0159] 在另一个示例中,可能会希望在相同快门角度下支持不同时间层(例如,120fps比特流内的60fps子比特流)。那么,主要的问题在于当120fps视频以120Hz显示时,偶数/奇数帧具有不同的有效快门角度。如果显示器具有相关信息,则可以应用智能技术来改善最终外观。表15中示出了所提出语法的示例,其中,HEVC的E.2.1VUI参数语法表(参考文献[1])被修改为支持所提到的快门角度信息。应注意,在另一个实施例中,shutter_angle语法可以替代性地被表示为帧速率与快门速度之比(参见等式(1)),而不是以绝对度数表示。

[0160] 表15:支持快门角度的示例VUI参数语法

	描述符
vui_parameters () {	
...	
vui_timing_info_present_flag	u(1)
if (vui_timing_info_present_flag) {	
vui_num_units_in_tick	u(32)
vui_time_scale	u(32)
vui_poc_proportional_to_timing_flag	u(1)
if (vui_poc_proportional_to_timing_flag)	
vui_num_ticks_poc_diff_one_minus1	ue(v)
vui_hrd_parameters_present_flag	u(1)
if (vui_hrd_parameters_present_flag)	
hrd_parameters (1, sps_max_sub_layers_minus1)	
}	
vui_shutter_angle_info_present_flag	u(1)
if (vui_shutter_angles_info_present_flag) {	
fixed_shutter_angle_within_cvs_flag	u(1)
if (fixed_shutter_angle_within_cvs_flag)	
fixed_shutter_angle	u(9)
else {	
for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
sub_layer_shutter_angle[i]	u(9)
}	
}	
...	
}	

[0162] vui_shutter_angle_info_present_flag等于1指定快门角度信息存在于vui_parameters()语法结构中。vui_shutter_angle_info_present_flag等于0指定快门角度信息在vui_parameters()语法结构中不存在。

[0163] fixed_shutter_angle_within_cvs_flag等于1指定快门角度信息对于CVS中的所有时间子层是相同的。fixed_shutter_angle_within_cvs_flag等于0指定快门角度信息对于CVS中的所有时间子层可以不相同。

[0164] fixed_shutter_angle指定CVS内的快门角度度数。fixed_shutter_angle的值应在0至360的范围中。

[0165] sub_layer_shutter_angle[i]指定当HighestTid等于i时的快门角度度数。sub_layer_shutter_angle[i]的值应在0至360的范围中。

[0166] 编码视频序列 (CVS) 内的逐渐帧速率更新

[0167] 实验已经表明,对于HDR显示器上显示的HDR内容,为了感知与100尼特显示器中的标准动态范围 (SDR) 回放相同的运动抖动,需要基于内容的明亮度来提高帧速率。在大多数标准 (AVC、HEVC、VVC等) 中,可以使用vui_time_scale、vui_num_units_in_tick和elemental_duration_in_tc_minus1[temporal_id_max]语法元素在VUI (被包含在SPS中) 中指示视频帧速率,例如,如下表16所示 (参见参考文献[1]中的E.2.1节)。

[0168] 表16: 用于指示HEVC中的帧速率的VUI语法元素

	<code>vui_parameters () {</code>	描述符
	<code>...</code>	
	<code>vui_timing_info_present_flag</code>	u(1)
	<code>if(vui_timing_info_present_flag) {</code>	
	<code> vui_num_units_in_tick</code>	u(32)
	<code> vui_time_scale</code>	u(32)
[0169]	<code>vui_poc_proportional_to_timing_flag</code>	u(1)
	<code>if(vui_poc_proportional_to_timing_flag)</code>	
	<code> vui_num_ticks_poc_diff_one_minus1</code>	ue(v)
	<code>vui_hrd_parameters_present_flag</code>	u(1)
	<code>if(vui_hrd_parameters_present_flag)</code>	
	<code> hrd_parameters(1, sps_max_sub_layers_minus1)</code>	
	<code>}</code>	
	<code>....</code>	

[0170] 如参考文献[1]中所讨论的,

[0171] 变量ClockTick是如下得到的并且被称为时钟周期:

[0172] $ClockTick = vui_num_units_in_tick \div vui_time_scale$

[0173] $picture_duration = ClockTick * (elemental_duration_in_tc_minus1[i] + 1)$

[0174] $frame_rate = 1 / pic_duration.$

[0175] 然而,帧速率只能在特定时刻被改变,例如,在HEVC中,只能在内部随机访问点(IRAP)帧处或新的CVS开始时改变。对于HDR回放,在存在淡入或淡出的情况下,由于图片的明亮度逐帧变化,因此可能需要改变每个图片的帧速率或图片持续时间。为了允许帧速率或图片持续时间在任何时刻(甚至在逐帧的基础上)刷新,在实施例中,提出了用于“逐渐刷新率”的新SEI消息,如表17所示。

[0176] 表17:用于支持SEI消息传递中的逐渐刷新帧速率的示例语法

	<code>gradual_refresh_rate(payloadSize) {</code>	描述符
[0177]	<code> num_units_in_tick</code>	u(32)
	<code> time_scale</code>	u(32)
	<code>}</code>	

[0178] 新语法num_units_in_tick的定义与vui_num_units_in_tick相同,并且time_scale的定义与vui_time_scale的定义相同。

[0179] num_units_in_tick是以对应于时钟周期计数器的一个增量(称为时钟周期)的频率time_scale Hz操作的时钟的时间单位的数量。num_units_in_tick应大于0。以秒为单位的时钟周期等于num_units_in_tick除以time_scale的商。例如,当视频信号的图片速率为25Hz时,time_scale可以等于27 000 000,并且num_units_in_tick可以等于1 080 000,并且因此时钟周期可以等于0.04秒。

[0180] time_scale是一秒中经过的时间单位的数量。例如,使用27MHz时钟测量时间的坐标系的时间单位为27 000 000。time_scale的值应大于0。

[0181] 使用gradual_refresh_rate SEI消息的图片的图片持续时间定义为:

[0182] $picture_duration = num_units_in_tick \div time_scale.$

[0183] 经由SEI消息传递的快门角度信息的信令

[0184] 如之前所讨论的,表15提供了支持快门角度的VUI参数语法的示例。作为示例,但非限制性地,表18列出了相同的语法元素,但现在作为快门角度信息的SEI消息的一部分。

应注意,SEI消息传递仅用作示例,并且类似的消息传递可以在高级语法的其他层构建,诸如序列参数集(SPS)、图片参数集(PPS)、切片或图块组标题等。

[0185] 表18:快门角度信息的示例SEI消息语法

	描述符
shutter_angle_information (payloadSize) {	
fixed_shutter_angle_within_cvs_flag	u(1)
if (fixed_shutter_angle_within_cvs_flag)	
fixed_shutter_angle	u(9)
else {	
for (i = 0; i <= sps_max_sub_layers_minus1; i++)	
sub_layer_shutter_angle[i]	u(9)
}	
}	

[0187] 快门角度通常以0至360度的度数表示。例如,180度的快门角度指示曝光持续时间是1/2的帧持续时间。快门角度可以表示为: $\text{shutter_angle} = \text{frame_rate} * 360 * \text{shutter_speed}$,其中,shutter_speed是曝光持续时间,并且frame_rate是帧持续时间的倒数。给定时间子层Tid的frame_rate可以由num_units_in_tick、time_scale、elemental_duration_in_tc_minus1[Tid]来指示。例如,当fixed_pic_rate_within_cvs_flag[Tid]等于1时:

[0188] $\text{frame_rate} = \text{time_scale} / (\text{num_units_in_tick} * (\text{elemental_duration_in_tc_minus1}[\text{Tid}] + 1))$

[0189] 在一些实施例中,快门角度(例如,fixed_shutter_angle)的值可以不是整数,例如,其可以是135.75度。为了使得能够更精确,在表21中,可以用u(16)或某个其他合适的位深(例如,12比特、14比特或多于16比特)来替换u(9)(无符号的9比特)。

[0190] 在一些实施例中,用“时钟周期”表达快门角度信息可以是有益的。在VVC中,变量ClockTick是如下得到的:

[0191] $\text{ClockTick} = \text{num_units_in_tick} \div \text{time_scale}$. (8)

[0192] 然后,可以将帧持续时间和曝光持续时间两者表示为时钟周期的倍数或分数:

[0193] $\text{exposure_duration} = fN * \text{ClockTick}$, (9)

[0194] $\text{frame_duration} = fM * \text{ClockTick}$, (10)

[0195] 其中,fN和fM是浮点值,并且 $fN \leq fM$ 。

[0196] 然后

$$\begin{aligned} \text{shutter_angle} &= \text{frame_rate} * 360 * \text{shutter_speed} = \\ &= (1/\text{frame_duration}) * 360 * \text{exposure_duration} = \end{aligned} \quad (11)$$

$$\begin{aligned} &= (\text{exposure_duration} * 360) / \text{frame_duration} = \\ &= (fN * \text{ClockTick} * 360) / (fM * \text{ClockTick}) = \\ &= (fN / fM) * 360 = (\text{Numerator} / \text{Denominator}) * 360, \end{aligned}$$

[0198] 其中,分子和分母是近似fN/fM比率的整数。

[0199] 表19示出了由等式(11)指示的SEI消息传递的示例。在此示例中,对于真实世界中的相机,快门角度必须大于0。

[0200] 表19:用于基于时钟周期的快门角度信息的示例SEI消息传递

	shutter_angle_information (payloadSize) {	描述符
	fixed_shutter_angle_within_cvs_flag	u(1)
	if (fixed_shutter_angle_within_cvs_flag) {	
	fixed_shutter_angle_numer_minus1	u(16)
	fixed_shutter_angle_denom_minus1	u(16)
[0201]	}	
	else {	
	for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
	sub_layer_shutter_angle_numer_minus1[i]	u(16)
	sub_layer_shutter_angle_denom_minus1[i]	u(16)
	}	
	}	

[0202] 如之前所讨论的,描绘了为了快门角度的精度对u(16) (无符号的16比特)的使用作为示例并且其对应于以下精度: $360/2^{16}=0.0055$ 。精度可以基于实际应用进行调节。例如,使用u(8),精度为 $360/2^8=1.4063$ 。

[0203] 注释——快门角度以大于0但小于或等于360度的度数表示。例如,180度的快门角度指示曝光持续时间是1/2的帧持续时间。

[0204] `fixed_shutter_angle_within_cvs_flag`等于1指定快门角度值对于CVS中的所有时间子层是相同的。`fixed_shutter_angle_within_cvs_flag`等于0指定快门角度值对于CVS中的所有时间子层可以不相同。

[0205] `fixed_shutter_angle_numer_minus1`加1指定用于得到快门角度值的分子。`fixed_shutter_angle_numer_minus1`的值应在0至65535 (包括0和65535) 的范围内。

[0206] `fixed_shutter_angle_denom_minus1`加1指定用于得到快门角度值的分母。`fixed_shutter_angle_denom_minus1`的值应在0至65535 (包括0和65535) 的范围内。

[0207] `fixed_shutter_angle_numer_minus1`的值应小于或等于`fixed_shutter_angle_denom_minus1`的值。

[0208] 变量`shutterAngle`的度数是如下得到的:

[0209]
$$\text{shutterAngle} = 360 * (\text{fixed_shutter_angle_numer_minus1} + 1) \div (\text{fixed_shutter_angle_denom_minus1} + 1)$$

[0210] `sub_layer_shutter_angle_numer_minus1[i]`加1指定当`HighestTid`等于*i*时用于得到快门角度值的分子。`sub_layer_shutter_angle_numer_minus1[i]`的值应在0至65535 (包括0和65535) 的范围内。

[0211] `sub_layer_shutter_angle_denom_minus1[i]`加1指定当`HighestTid`等于*i*时用于得到快门角度值的分母。`sub_layer_shutter_angle_denom_minus1[i]`的值应在0至65535 (包括0和65535) 的范围内。

[0212] `sub_layer_shutter_angle_numer_minus1[i]`的值应小于或等于`sub_layer_shutter_angle_denom_minus1[i]`的值。

[0213] 变量`subLayerShutterAngle[i]`的度数是如下得到的:

[0214]
$$\text{subLayerShutterAngle}[i] = 360 * (\text{sub_layer_shutter_angle_numer_minus1}[i] + 1) \div (\text{sub_layer_shutter_angle_denom_minus1}[i] + 1)$$

[0215] 在另一个实施例中,帧持续时间 (例如,`frame_duration`) 可以通过某种其他手段来指定。例如,在DVB/ATSC中,当`fixed_pic_rate_within_cvs_flag[Tid]`等于1时:

[0216] $frame_rate = time_scale / ((num_units_in_tick * (elemental_duration_in_tc_minus1[Tid+1])), frame_duration = 1 / frame_rate.$

[0217] 表19和后续表格中的一些表格中的语法假设快门角度将总是大于零;然而,快门角度=0可以用来用信号表示一种创作意图,其中,内容应在没有任何运动模糊的情况下被显示。这可以是移动图形、动画、CGI纹理和无光泽屏幕等的情况。因此,例如,信令快门角度=0可以可用于转码器中的模式决策(例如,以选择保留边缘的转码模式)以及通过CTA接口或3GPP接口接收快门角度元数据的显示器中的模式决策。例如,快门角度=0可以用于指示不应执行任何运动处理(诸如去噪、帧插值等)的显示器。在这种实施例中,语法元素fixed_shutter_angle_numer_minus1和sub_layer_shutter_angle_numer_minus1[i]可以被语法元素fixed_shutter_angle_numer和sub_layer_shutter_angle_numer[i]替换,其中,

[0218] fixed_shutter_angle_numer指定用于得到快门角度值的分子。fixed_shutter_angle_numer的值应在0至65535(包括0和65535)的范围中。

[0219] sub_layer_shutter_angle_numer[i]指定当HighestTid等于i时用于得到快门角度值的分子。sub_layer_shutter_angle_numer[i]的值应在0至65535(包括0和65535)的范围中。

[0220] 在另一个实施例中,fixed_shutter_angle_denom_minus1和sub_layer_shutter_angle_denom_minus1[i]还可以被语法元素fixed_shutter_angle_denom以及sub_layer_shutter_angle_denom[i]替换。

[0221] 在实施例中,如表20所描绘的,通过在VVC中将general_hrd_parameters_present_flag设置为等于1,可以重复使用在SPS中定义的num_units_in_tick和time_scale语法。在这种情况下,SEI消息可以被重新命名为曝光持续时间SEI消息。

[0222] 表20:用于用信号表示曝光持续时间的示例SEI消息传递

	exposure_duration_information (payloadSize) {	描述符
	fixed_exposure_duration_within_cvs_flag	u(1)
	if (fixed_shutter_angle_within_cvs_flag) {	
	fixed_exposure_duration_numer_minus1	u(16)
	fixed_exposure_duration_denom_minus1	u(16)
[0223]	}	
	else {	
	for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
	sub_layer_exposure_duration_numer_minus1[i]	u(16)
	sub_layer_exposure_duration_denom_minus1[i]	u(16)
	}	
	}	

[0224] fixed_exposure_duration_within_cvs_flag等于1指定有效曝光持续时间值对于CVS中的所有时间子层是相同的。fixed_exposure_duration_within_cvs_flag等于0指定有效曝光持续时间值对于CVS中的所有时间子层可以不相同。

[0225] fixed_exposure_duration_numer_minus1加1指定用于得到曝光持续时间值的分子。fixed_exposure_duration_numer_minus1的值应在0至65535(包括0和65535)的范围中。

[0226] fixed_exposure_duration_denom_minus1加1指定用于得到曝光持续时间值的分母。fixed_exposure_duration_denom_minus1的值应在0至65535(包括0和65535)的范围

中。

[0227] `fixed_exposure_during_numer_minus1`的值应小于或等于`fixed_exposure_duration_demom_minus1`的值。

[0228] 变量`fixedExposureDuration`是如下得到的：

[0229]
$$\text{fixedExposureDuration} = (\text{fixed_exposure_duration_numer_minus1} + 1) \div (\text{fixed_exposure_duration_demom_minus1} + 1) * \text{ClockTicks}$$

[0230] `sub_layer_exposure_duration_numer_minus1[i]`加1指定当`HighestTid`等于`i`时用于得到曝光持续时间值的分子。`sub_layer_exposure_duration_numer_minus1[i]`的值应在0至65535(包括0和65535)的范围中。

[0231] `sub_layer_exposure_duration_demom_minus1[i]`加1指定当`HighestTid`等于`i`时用于得到曝光持续时间值的分母。`sub_layer_exposure_duration_demom_minus1[i]`的值应在0至65535(包括0和65535)的范围中。

[0232] `sub_layer_exposure_duration_numer_minus1[i]`的值应小于或等于`sub_layer_exposure_duration_demom_minus1[i]`的值。

[0233] `HighestTid`的变量`subLayerExposureDuration[i]`等于`i`是如下得到的：

[0234]
$$\text{subLayerExposureDuration}[i] = (\text{sub_layer_exposure_duration_numer_minus1}[i] + 1) \div (\text{sub_layer_exposure_duration_demom_minus1}[i] + 1) * \text{ClockTicks}.$$

[0235] 在另一个实施例中,如表21所示,可以通过语法元素`expo_num_units_in_tick`和`expo_time_scale`来明确地定义`clockTick`。此处的优点在于它不依赖于在VVC中`general_hrd_parameters_present_flag`是否如先前的实施例那样设置为等于1,于是

[0236]
$$\text{clockTick} = \text{expo_num_units_in_tick} \div \text{expo_time_scale}. \quad (12)$$

[0237] 表21:用于曝光时间信令的示例SEI消息传递

	exposure_duration_information (payloadSize) {	描述符
	expo_num_units_in_tick	u(32)
	expo_time_scale	u(32)
	fixed_exposure_duration_within_cvs_flag	u(1)
[0238]	if (!fixed_exposure_duration_within_cvs_flag)	
	for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
	sub_layer_exposure_duration_numer_minus1[i]	u(16)
	sub_layer_exposure_duration_denom_minus1[i]	u(16)
	}	
	}	

[0239] `expo_num_units_in_tick`是以对应于时钟周期计数器的一个增量(被称为时钟周期)的频率`time_scale` Hz操作的时钟的时间单位的数量。`expo_num_units_in_tick`应大于0。由变量`clockTick`定义的以秒为单位的时钟周期等于`expo_num_units_in_tick`除以`expo_time_scale`的商。

[0240] `expo_time_scale`是一秒中经过的时间单位的数量。

[0241]
$$\text{clockTick} = \text{expo_num_units_in_tick} \div \text{expo_time_scale}.$$

[0242] 注释:以下两个语法元素:`expo_num_units_in_tick`和`expo_time_scale`被定义为测量曝光持续时间。

[0243] 比特流一致性要求当`num_units_in_tick`和`time_scale`存在时,`clockTick`应小于

或等于ClockTick。

[0244] `fixed_exposure_duration_within_cvs_flag`等于1指定有效曝光持续时间值对于CVS中的所有时间子层是相同的。`fixed_exposure_duration_within_cvs_flag`等于0指定有效曝光持续时间值对于CVS中的所有时间子层可以不相同。当`fixed_exposure_duration_within_cvs_flag`等于1时,变量`fixedExposureDuration`设置为等于`clockTick`。

[0245] `sub_layer_exposure_duration_numer_minus1[i]`加1指定当`HighestTid`等于`i`时用于得到曝光持续时间值的分子。`sub_layer_exposure_duration_numer_minus1[i]`的值应在0至65535(包括0和65535)的范围中。

[0246] `sub_layer_exposure_duration_denom_minus1[i]`加1指定当`HighestTid`等于`i`时用于得到曝光持续时间值的分母。`sub_layer_exposure_duration_denom_minus1[i]`的值应在0至65535(包括0和65535)的范围中。

[0247] `sub_layer_exposure_duration_numer_minus1[i]`的值应小于或等于`sub_layer_exposure_duration_denom_minus1[i]`的值。

[0248] `HighestTid`的变量`subLayerExposureDuration[i]`等于`i`是如下得到的:

[0249]
$$\text{subLayerExposureDuration}[i] = (\text{sub_layer_exposure_duration_numer_minus1}[i] + 1) \div (\text{sub_layer_exposure_duration_denom_minus1}[i] + 1) * \text{clockTick}.$$

[0250] 如之前所讨论的,语法参数`sub_layer_exposure_duration_numer_minus1[i]`和`sub_layer_exposure_duration_denom_minus1[i]`还可以被`sub_layer_exposure_duration_numer[i]`和`sub_layer_exposure_duration_denom[i]`替换。

[0251] 在另一个实施例中,如表22所示,可以通过语法元素`sii_num_units_in_shutter_interval`和`sii_time_scale`定义参数`ShutterInterval`(即,曝光持续时间),其中,

[0252]
$$\text{ShutterInterval} = \text{sii_num_units_in_shutter_interval} \div \text{sii_time_scale}.$$

(13)

表22:用于曝光持续时间(快门间隔信息)信令的示例SEI消息传递

	shutter_interval_information (payloadSize) {	描述符
	sii_num_units_in_shutter_interval	u(32)
	sii_time_scale	u(32)
	fixed_shutter_interval_within_cvs_flag	u(1)
[0253]	if (!fixed_shutter_interval_within_cvs_flag)	
	for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
	sub_layer_shutter_interval_numer[i]	u(16)
	sub_layer_shutter_interval_denom[i]	u(16)
	}	

[0254] 快门间隔信息SEI消息语义

[0255] 快门间隔信息SEI消息指示在编码和显示之前的相关视频内容的快门间隔——例如,对于相机捕获的内容而言,指示使图像传感器曝光以产生图片的时间量。

[0256] `sii_num_units_in_shutter_interval`指定以对应于快门时钟周期计数器的一个增量的频率`sii_time_scale` Hz操作的时钟的时间单位的数量。由变量`ShutterInterval`定义的以秒为单位的快门间隔等于`sii_num_units_in_shutter_interval`除以`sii_time_scale`的商。例如,当`ShutterInterval`等于0.04秒时,`sii_time_scale`可以等于27000000,并且`sii_num_units_in_shutter_interval`可以等于1080000。

[0257] `sii_time_scale`指定一秒中经过的时间单位的数量。例如,使用27MHz时钟测量时间的坐标系中的`sii_time_scale`为27000000。

[0258] 当`sii_time_scale`的值大于0时,`ShutterInterval`的值由以下等式指定:

[0259]
$$\text{ShutterInterval} = \text{sii_num_units_in_shutter_interval} \div \text{sii_time_scale}$$

[0260] 否则(`sii_time_scale`的值等于0),`ShutterInterval`应被解释为未知或未指定。

[0261] 注释1——`ShutterInterval`的值等于0可以指示相关视频内容包含屏幕捕获内容、计算机生成的内容或其他非相机捕获内容。

[0262] 注释2——`ShutterInterval`的值大于编码图片速率的倒数(即,编码图片间隔)的值可以指示编码图片速率大于创建相关视频内容的图片速率——例如,当编码图片速率为120Hz并且在编码和显示之前的相关视频内容的图片速率为60Hz时。给定时间子层Tid的编码间隔可以由`ClockTick`和`elemental_duration_in_tc_minus1[Tid]`来指示。例如,当`fixed_pic_rate_within_cvs_flag[Tid]`等于1时,由变量`PictureInterval[Tid]`定义的给定时间子层Tid的图片间隔可以由以下等式指定:

[0263]
$$\text{PictureInterval}[\text{Tid}] = \text{ClockTick} * (\text{elemental_duration_in_tc_minus1}[\text{Tid}] + 1) .$$

[0264] `fixed_shutter_interval_within_cvs_flag`等于1指定`ShutterInterval`的值对于CVS中的所有时间子层是相同的。`fixed_shutter_interval_within_cvs_flag`等于0指定`ShutterInterval`的值对于CVS中的所有时间子层可以不相同。

[0265] `sub_layer_shutter_interval_numer[i]`指定当`HighestTid`等于*i*时用于得到由变量`subLayerShutterInterval[i]`定义的以秒为单位的子层快门间隔的分子。

[0266] `sub_layer_shutter_interval_denom[i]`指定当`HighestTid`等于*i*时用于得到由变量`subLayerShutterInterval[i]`定义的以秒为单位的子层快门间隔的分母。

[0267] `HighestTid`的`subLayerShutterInterval[i]`的值等于*i*是如下得到的。当`fixed_shutter_interval_within_cvs_flag`的值等于0并且`sub_layer_shutter_interval_denom[i]`的值大于0时:

[0268]
$$\text{subLayerShutterInterval}[\text{i}] = \text{ShutterInterval} * \text{sub_layer_shutter_interval_numer}[\text{i}] \div \text{sub_layer_shutter_interval_denom}[\text{i}]$$

[0269] 否则(`sub_layer_shutter_interval_denom[i]`的值等于0),`subLayerShutterInterval[i]`应被解释为未知或未指定。当`fixed_shutter_interval_within_cvs_flag`的值不等于0时,

[0270]
$$\text{subLayerShutterInterval}[\text{i}] = \text{ShutterInterval} .$$

[0271] 在替代性实施例,使用单个值,而不是使用分子和分母用于用信号表示子层快门间隔。表23中示出了这种语法的示例。

[0272] 表23:用于快门间隔信令的示例SEI消息传递

	shutter_interval_information (payloadSize) {	描述符
	sii_num_units_in_shutter_interval	u(32)
	sii_time_scale	u(32)
	fixed_shutter_interval_within_cvs_flag	u(1)
[0273]	if (!fixed_shutter_interval_within_cvs_flag)	
	for (i = 0; i <= sps_max_sub_layers_minus1; i++) {	
	sub_layer_num_units_in_shutter_interval[i]	u(32)
	}	
	}	

[0274] 快门间隔信息SEI消息语义

[0275] 快门间隔信息SEI消息指示在编码和显示之前的相关视频内容的快门间隔——例如,对于相机捕获的内容而言,指示使图像传感器曝光以产生图片的时间量。

[0276] `sii_num_units_in_shutter`指定以对应于快门时钟周期计数器的一个增量的频率`sii_time_scale` Hz操作的时钟的时间单位的数量。由变量`ShutterInterval`定义的以秒为单位的快门间隔等于`sii_num_units_in_shutter_interval`除以`sii_time_scale`的商。例如,当`ShutterInterval`等于0.04秒时,`sii_time_scale`可以等于27 000 000,并且`sii_num_units_in_shutter_interval`可以等于1 080 000。

[0277] `sii_time_scale`指定一秒中经过的时间单位的数量。例如,使用27MHz时钟测量时间的坐标系的时间坐标系的`sii_time_scale`为27 000 000。

[0278] 当`sii_time_scale`的值大于0时,`ShutterInterval`的值由以下等式指定:

[0279] $ShutterInterval = sii_num_units_in_shutter_interval \div sii_time_scale$

[0280] 否则(`sii_time_scale`的值等于0),`ShutterInterval`应被解释为未知或未指定。

[0281] 注释1——`ShutterInterval`的值等于0可以指示相关视频内容包含屏幕捕获内容、计算机生成的内容或其他非相机捕获内容。

[0282] 注释2——`ShutterInterval`的值大于编码图片速率的倒数(即,编码图片间隔)的值可以指示编码图片速率大于创建相关视频内容的图片速率——例如,当编码图片速率为120Hz并且在编码和显示之前的相关视频内容的图片速率为60Hz时。给定时间子层`Tid`的编码图片间隔可以由`ClockTick`和`elemental_duration_in_tc_minus1[Tid]`来指示。例如,当`fixed_pic_rate_within_cvs_flag[Tid]`等于1时,由变量`PictureInterval[Tid]`定义的给定时间子层`Tid`的图片间隔可以由以下等式指定:

[0283] $PictureInterval[Tid] = ClockTick * (elemental_duration_in_tc_minus1[Tid] + 1)$ 。

[0284] `fixed_shutter_interval_within_cvs_flag`等于1指定`ShutterInterval`的值对于CVS中的所有时间子层是相同的。`fixed_shutter_interval_within_cvs_flag`等于0指定`ShutterInterval`的值对于CVS中的所有时间子层可以不相同。

[0285] `sub_layer_num_units_in_shutter_interval[i]`指定以对应于快门时钟周期计数器的一个增量的频率`sii_time_scale` Hz操作的时钟的时间单位的数量。当`HighestTid`等于`i`时,由变量`subLayerShutterInterval[i]`定义的以秒为单位的子层快门间隔等于`sub_layer_num_units_in_shutter_interval[i]`除以`sii_time_scale`的商。

[0286] 当`fixed_shutter_interval_within_cvs_flag`的值等于0并且`sii_time_scale`的值大于0时,`subLayerShutterInterval[i]`的值由以下等式指定:

[0287] $\text{subLayerShutterInterval}[i] = \text{sub_layer_num_unitd_in_shutter_interval}[i] \div \text{sii_time_scale}$

[0288] 否则(sii_time_scale 的值等于0), $\text{subLayerShutterInterval}[i]$ 应被解释为未知或未指定。当 $\text{fixed_shutter_interval_within_cvs_flag}$ 的值不等于0时,

[0289] $\text{subLayerShutterInterval}[i] = \text{ShutterInterval}$.

[0290] 表24提供了表18至表23中讨论的用于提供与快门角度或曝光持续时间相关的SEI信息传递的六种方法的概述。

[0291] 表24: 用于用信号表示信号快门角度信息的SEI消息传递方法的概述

表号	关键信令元素和依赖性
18	明确地用信号表示快门角度 (0 至 360)
19	快门角度表示为要用360缩放的分子值与分母值之比 (暗示时钟周期值)
20	曝光持续时间用信号表示为分子值与分母值之比 (暗示时钟周期值)
[0292] 21	曝光持续时间用信号表示为分子值与分母值之比; 时钟周期值明确地用信号表示为两个值之比
22	快门间隔信息用信号表示为以下两个值之比: 曝光的时钟周期单位数和曝光时间标度; 子层相关曝光时间用信号表示为两个值之比
23	快门间隔信息或曝光持续时间用信号表示为以下两个值之比: 曝光的时钟周期单位数和曝光时间标度; 子层相关曝光时间用信号表示为每个子层中曝光的时钟周期数

[0293] 可变帧速率信令

[0294] 如2019年8月6日提交的美国临时申请62/883,195中所讨论的,在许多应用中,期望解码器支持以可变帧速率进行回放。帧速率适配通常是假设的参考解码器(HRD)中的操作的一部分,如例如参考文献[2]的附录C中所描述的。在实施例,提出通过SEI消息传递或其他手段用信号表示将图片呈现时间(PPT)定义为90kHz时钟的函数的语法元素。这是对如HRD中指定的标称解码器图片缓冲器(DPB)输出时间的一种重复,但现在使用如MPEG-2系统中指定的90kHz ClockTicks精度。此SEI消息的益处在于:a)如果未启用HRD,则仍然可以使用PPTSEI消息来指示每个帧的定时;b)可以使比特流定时和系统定时的转化变得容易。

[0295] 表25描述了所提出PPT定时消息的语法的示例,所述语法与MPEG-2传输(H.222)(参考文献[4])中使用的呈现时间戳(PTS)变量的语法相匹配。

[0296] 表25: 用于图片呈现时间消息传递的示例语法

[0297] <code>picture_presentation_time(payloadSize) {</code>	描述符
<code>PPT</code>	<code>u(33)</code>
<code>}</code>	

[0298] PPT(图片呈现时间)

[0299] ——呈现时间应与解码时间如下相关:PPT是在三个分开的字段中被编码的33比特数字。PPT指示基本流 n 的呈现单位 k 的系统目标解码器中的呈现时间 $tp_n(k)$ 。PPT的值以系统时钟频率周期除以300(从而产生90kHz)为单位被指定。根据下文的等式从PPT得到图片呈现时间。

[0300]
$$\text{PPT}(k) = ((\text{system_clock_frequency} \times tp_n(k)) / 300) \% 2^{33}$$

[0301] 其中, $tp_n(k)$ 是呈现单位 $P_n(k)$ 的呈现时间。

[0302] 参考文献

[0303] 本文列出的参考文献中的每一篇都通过引用以其全文并入。

[0304] [1] High efficiency video coding [高效视频编码], H.265, H系列, Coding of moving video [移动视频编码], ITU, (02/2018)。

[0305] [2] B. Bross, J. Chen 和 S. Liu, “Versatile Video Coding (Draft 5) [多功能视频编码(第5稿)]”, JVET output document [JVET输出文件], JVET-N1001, v5, 2019年5月14日上传。

[0306] [3] C. Carbonara, J. DeFilippis, M. Korpi, “High Frame Rate Capture and Production” [高帧速率捕获和制作], SMPTE 2015 Annual Technical Conference and Exhibition [SMPTE 2015 年度技术会议和展览会], 2015年10月26至29日。

[0307] [4] Infrastructure of audiovisual services—Transmission multiplexing and synchronization [视听服务基础设施——传送复用和同步], H.222.0, H系列, Generic coding of moving pictures and associated audio information: Systems [移动图片和相关音频信息的通用编码: 系统], ITU, 08/2018。

[0308] 示例计算机系统实施方式

[0309] 本发明的实施例可以利用计算机系统、以电子电路和部件来配置的系统、集成电路(IC)设备(诸如微控制器、现场可编程门阵列(FPGA)或另一可配置或可编程逻辑设备(PLD)、离散时间或数字信号处理器(DSP)、专用IC(ASIC))和/或包括一个或多个这种系统、设备或部件的装置来实施。计算机和/或IC可以执行、控制或实行与帧速率可伸缩性相关的指令,诸如本文所描述的那些指令。计算机和/或IC可以计算与本文描述的帧速率可伸缩性相关的各种参数或值中的任何参数或值。图像和视频实施例可以以硬件、软件、固件及其各种组合来实施。

[0310] 本发明的某些实施方式包括执行软件指令的计算机处理器,所述软件指令使处理器执行本发明的方法。例如,显示器、编码器、机顶盒、转码器等中的一个或多个处理器可以通过执行处理器可访问的程序存储器中的软件指令来实施如上所述的与帧速率可伸缩性相关的方法。还可以以程序产品的形式提供本发明的实施例。程序产品可以包括携带一组计算机可读信号的任何非暂态且有形介质,所述一组计算机可读信号包括指令,所述指令当由数据处理器执行时使数据处理器执行本发明的方法。根据本发明的程序产品可以采用各种非暂态且有形形式中的任何一种。程序产品可以包括例如物理介质,诸如包括软盘、硬盘驱动器的磁性数据存储介质、包括CD ROM、DVD的光学数据存储介质、包括ROM、闪存存储器RAM的电子数据存储介质等。程序产品上的计算机可读信号可以任选地被压缩或加密。

[0311] 在上面提到部件(例如,软件模块、处理器、组件、设备、电路等)的情况下,除非另有指示,否则对所述部件的引用(包括对“装置”的引用)都应被解释为包括执行所描述部件的功能的任何部件作为所述部件的等同物(例如,功能上等同的),包括在结构上不等于于执行在本发明的所图示示例实施例中的功能的所公开结构的部件。

[0312] 等同物、扩展、替代方案和杂项

[0313] 因此,描述了与帧速率可伸缩性相关的示例实施例。在前述说明书中,已经参考可以根据实施方式而变化的许多具体细节描述了本发明的实施例。因此,指明本发明是什么以及申请人意图发明是什么的唯一且排他性指示是根据本申请以权利要求授权时的特定

形式授权的一组权利要求,包括任何后续更正。本文中针对这种权利要求中包含的术语明确阐述的任何定义应支配如在权利要求中使用的这种术语的含义。因此,权利要求中未明确记载的限制、要素、特性、特征、优点或属性不应以任何方式限制这种权利要求的范围。因此,说明书和附图应当被认为是说明性而非限制性意义的。

[0314] 附录

[0315] 本附录提供了表D.2的拷贝和来自H.265规范的相关联的pic_struct相关信息(参考文献[1])。

[0316] 表D.2——pic_struct的解释

值	指示的图片显示	限制
0	(逐行) 帧	field_seq_flag 应等于 0
1	顶部字段	field_seq_flag 应等于 1
2	底部字段	field_seq_flag 应等于 1
3	顶部字段、底部字段, 按此顺序	field_seq_flag 应等于 0
4	底部字段、顶部字段, 按此顺序	field_seq_flag 应等于 0
5	顶部字段、底部字段、重复的顶部字段, 按此顺序	field_seq_flag 应等于 0
6	底部字段、顶部字段、重复的底部字段, 按此顺序	field_seq_flag 应等于 0
[0317] 7	帧双倍增	field_seq_flag 应等于 0 fixed_pic_rate_within_cvs_flag 应等于 1
8	帧三倍增	field_seq_flag 应等于 0 fixed_pic_rate_within_cvs_flag 应等于 1
9	按输出顺序与前一个底部字段配对的顶部字段	field_seq_flag 应等于 1
10	按输出顺序与前一个顶部字段配对的底部字段	field_seq_flag 应等于 1
11	按输出顺序与下一个底部字段配对的顶部字段	field_seq_flag 应等于 1
12	按输出顺序与下一个顶部字段配对的底部字段	field_seq_flag 应等于 1

[0318] pic_struct语法元素的语义

[0319] pic_struct指示图片应显示为帧还是显示为一个或多个字段,并且对于帧的显示,当fixed_pic_rate_within_cvs_flag等于1时,可以指示使用如等式E-73所给出的等于DpbOutputElementalInterval[n]的固定帧刷新闻隔的显示器的帧双倍增或三倍增重复周期。表D.2中指定了pic_struct的解释。表D.2中未列出的pic_struct值保留下来以供将来由ITU-T|ISO/IEC使用并且不应存在于符合本规范的此版本的比特流中。解码器应忽略pic_struct的保留值。

[0320] 当存在时,比特流一致性要求pic_struct的值应被约束成使得以下条件中的恰好一个条件为真:

[0321] -对于CVS中的所有图片,pic_struct的值等于0、7或8。

[0322] -对于CVS中的所有图片,pic_struct的值等于1、2、9、10、11或12。

[0323] -对于CVS中的所有图片,pic_struct的值等于3、4、5或6。

[0324] 当fixed_pic_rate_within_cvs_flag等于1时,帧双倍增由pic_struct等于7指示,这指示帧应在显示器上连续显示两次,其中帧刷新闻隔等于如等式E-73所给出的DpbOutputElementalInterval[n],并且帧三倍增由pic_struct等于8指示,这指示帧应在显示器上连续显示三次,其中帧刷新闻隔等于如等式E-73所给出的DpbOutputElementalInterval[n]。

[0325] 注释3——帧双倍增可以用于促进例如25Hz逐行扫描视频在50Hz逐行扫描显示器上的显示或30Hz逐行扫描视频在60Hz逐行扫描显示器上的显示。以每隔一帧交替组合的方式使用帧双倍增和帧三倍增可以用于促进24Hz逐行扫描视频在60Hz逐行扫描显示器上的显示。

[0326] 字段的关联指示符(pic_struct等于9到12)提供了将互补奇偶校验的字段作为帧关联在一起的提示。字段的奇偶校验可以是顶部或底部,并且当一个字段的奇偶校验是顶部,而另一个字段的奇偶校验是底部时,两个字段的奇偶校验被认为是互补的。

[0327] 当frame_field_info_present_flag等于1时,比特流一致性要求表D.2的第三列中指定的约束应适用。

[0328] 注释4——当frame_field_info_present_flag等于0时,则在许多情况下,默认值可以通过其他手段被推断或指示。在缺少图片的预期显示类型的其他指示的情况下,当frame_field_info_present_flag等于0时,解码器应推断pic_struct的值等于0。

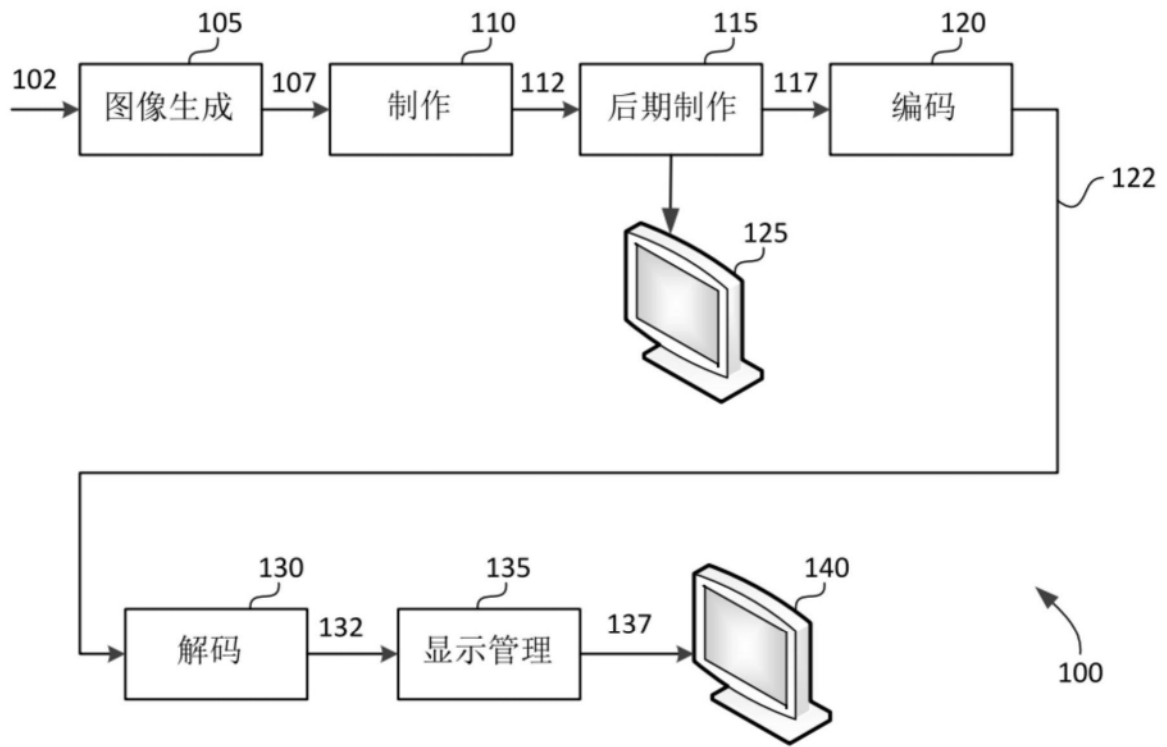


图1

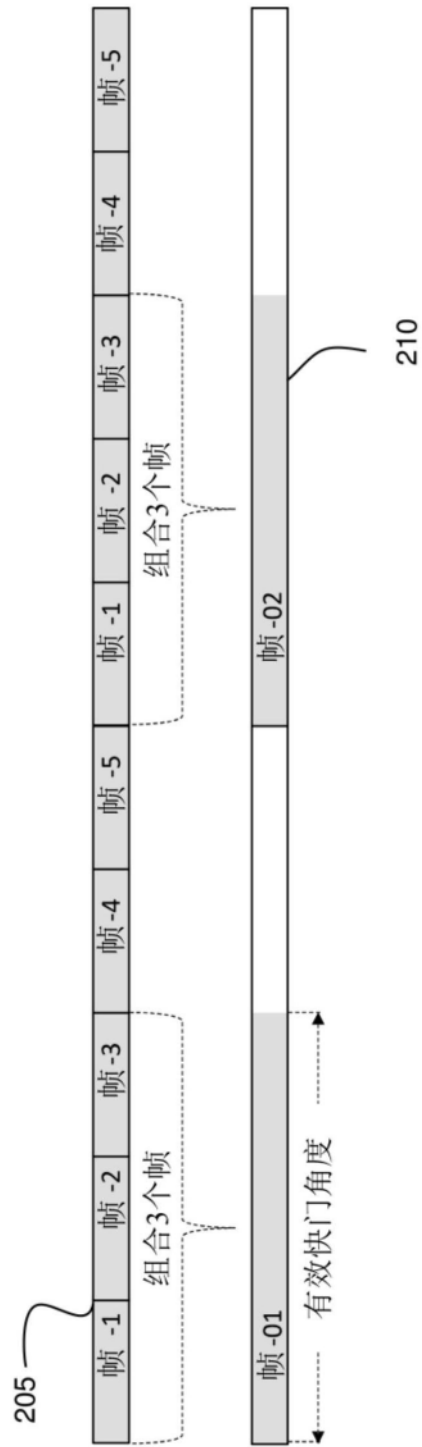


图2

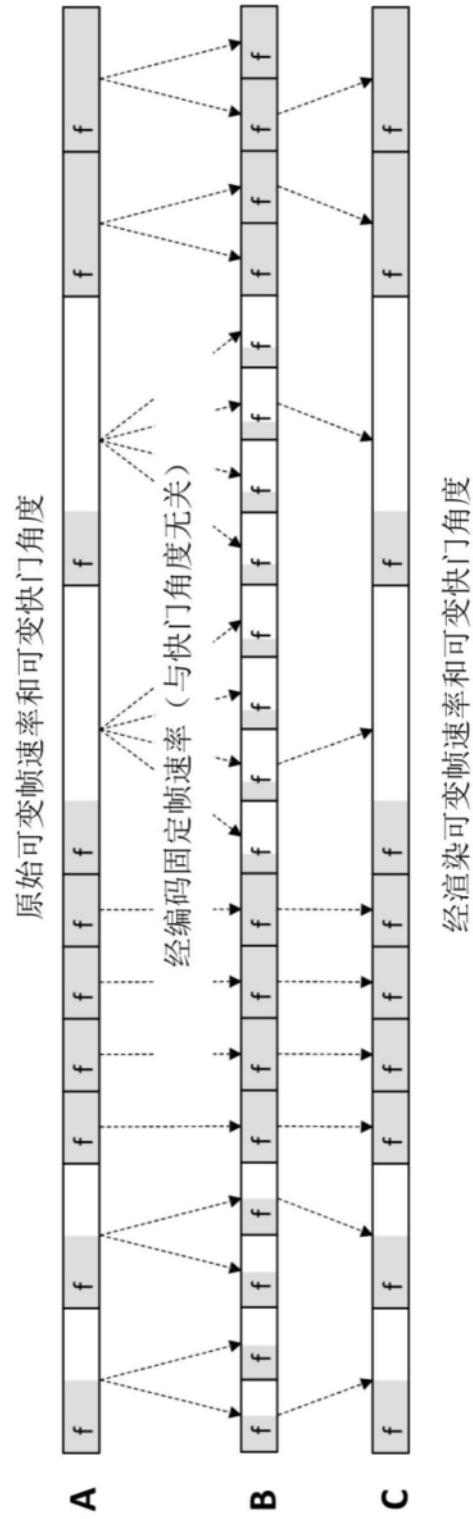


图3

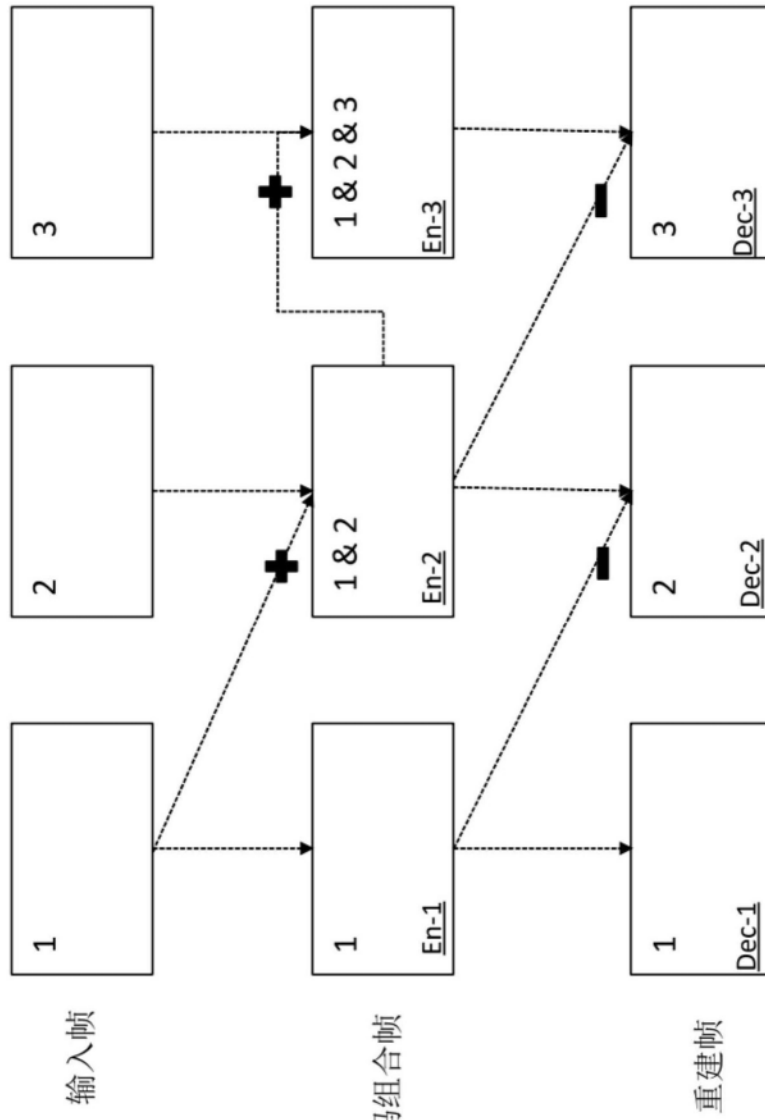


图4