



(12) 发明专利申请

(10) 申请公布号 CN 115668926 A

(43) 申请公布日 2023. 01. 31

(21) 申请号 202180038101.6

(74) 专利代理机构 北京市柳沈律师事务所
11105

(22) 申请日 2021.05.26

专利代理师 张亮

(30) 优先权数据

63/030,726 2020.05.27 US

(51) Int.Cl.

H04N 19/174 (2006.01)

(85) PCT国际申请进入国家阶段日

2022.11.25

(86) PCT国际申请的申请数据

PCT/US2021/034259 2021.05.26

(87) PCT国际申请的公布数据

W02021/242862 EN 2021.12.02

(71) 申请人 字节跳动有限公司

地址 美国加利福尼亚州

(72) 发明人 王业奎 张莉 许继征 张凯

权利要求书4页 说明书62页 附图15页

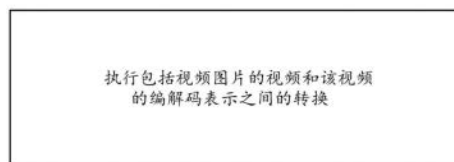
(54) 发明名称

在视频编解码中使用子图片和片

(57) 摘要

描述了用于视频编码和视频解码的几种技术。一种示例方法包括根据规则执行视频和视频的比特流之间的转换,该规则指定序列参数集包括在转换中使用的量化参数表,并且该规则指定是否或如何根据序列参数集中的第二语法元素来控制序列参数集包括指示量化参数表中的点的数量的第一语法元素。

500



1. 一种处理视频数据的方法,包括:
根据规则执行视频和所述视频的比特流之间的转换,
其中,所述规则指定序列参数集包括在所述转换中使用的量化参数表,并且
所述规则指定是否或者如何根据所述序列参数集中的第二语法元素来控制所述序列参数集包括指示所述量化参数表中的点的数量的第一语法元素。
2. 根据权利要求1所述的方法,其中,所述量化参数表是色度量化参数映射表。
3. 根据权利要求1所述的方法,其中,所述第一语法元素的值加1等于所述量化参数表中的所述点的数量。
4. 根据权利要求1所述的方法,其中,所述第二语法元素的值指示用于描述所述量化参数表的起始亮度和色度量化参数。
5. 根据权利要求1所述的方法,其中,所述量化参数表中的所述点的数量的最大值基于所述第二语法元素的值。
6. 根据权利要求1所述的方法,其中,所述规则指定所述量化参数表中的所述点的数量的最大值减1是最大量化参数值与用于描述第*i*个色度量化参数映射表的起始亮度和色度量化参数之间的差。
7. 根据权利要求6所述的方法,其中,所述最大值设置为 $(63 - (\text{qp_table_start_minus26}[i] + 26))$ 。
8. 根据权利要求6所述的方法,其中,所述最大值被设置为 $(64 - (\text{qp_table_start_minus26}[i] + 26))$ 。
9. 一种处理视频数据的方法,包括:
根据规则执行视频和所述视频的比特流之间的转换,
其中,所述规则指定当序列参数集中指示的子图片的数量减1大于零时,并且当所述序列参数集中的语法元素指示编解码层视频序列中的所述视频的第一视频图片的第*i*个子图片在排除环路滤波操作的解码过程中被视为所述视频的第二视频图片时,以下两个条件之一为真:
 - (1) 所述第一视频图片的第一参考图片列表或第二参考图片列表中的每个活动条目所参考的所述第二视频图片和所述第一视频图片对于以下各项中的每一项具有相同的值:
以亮度样点为单位的图片宽度,
以亮度样点为单位的图片高度,
所述子图片的数量减1,以及
对于0到所述子图片的数量减1,包括端值,的范围内的每个*i*值,所述第*i*个子图片的左顶编解码树单元的水平位置、所述第*i*子图片的左顶编解码树单元CTU的垂直位置、所述第*i*个子图片的宽度减1、所述第*i*个子图片的高度减1、所述第*i*个子图片的语法元素;和
 - (2) 所述第一参考图片列表或所述第二参考图片列表中的每个活动条目所参考的所述第二视频图片是层间参考图片,对于所述层间参考图片,所述子图片的数量减1的值等于零。
10. 一种处理视频数据的方法,包括:
根据规则执行包括视频块的视频和所述视频的比特流之间的转换,
其中所述规则指定当仿射编解码工具被启用并且基于子块的时域运动矢量预测编解

码工具被禁用时,子块merge候选的最大数量不允许为数量N。

11. 根据权利要求10所述的方法,其中 $N=0$ 。

12. 根据权利要求10所述的方法,其中 N 大于0。

13. 根据权利要求10所述的方法,其中,在序列参数集中支持的基于子块的merge运动矢量预测候选的最大数量减五的范围是从 $[0,4]$ 开始,而与所述序列参数集中指示SbTMVP编解码工具是被启用还是被禁用的语法元素的值无关。

14. 根据权利要求10所述的方法,其中,所述比特流是一致性比特流,当所述仿射编解码工具被启用时,所述一致性比特流包括1到5,包括端值,的范围内的基于子块的merge运动矢量预测候选的最大数量的值。

15. 根据权利要求10所述的方法,其中,所述比特流是一致性比特流,当所述仿射编解码工具被禁用时,所述一致性比特流包括0到1,包括端值,的范围内的基于子块的merge运动矢量预测候选的最大数量的值。

16. 根据权利要求10所述的方法,其中,所述比特流是否包括指示merge子块操作是否被启用的语法元素是基于所述仿射编解码工具是否被启用而不是基于所述子块merge候选的最大数量。

17. 一种处理视频数据的方法,包括:

执行包括一个或多个视频图片的视频和所述视频的比特流之间的转换,

其中,所述比特流符合格式规则,并且

所述格式规则指定指示是否启用仿射merge模式的语法元素是否在所述比特流中。

18. 根据权利要求17所述的方法,其中,所述格式规则指定基于所述仿射merge模式是否在序列参数集中启用而在所述比特流中选择性地指示所述语法元素。

19. 根据权利要求17所述的方法,其中,所述格式规则指定当所述语法元素指示所述仿射merge模式被启用时,在所述比特流中指示子块merge候选的最大数量。

20. 一种处理视频数据的方法,包括:

根据规则执行包括一个或多个视频图片的视频和所述视频的比特流之间的转换,所述一个或多个视频图片包括一个或多个子图片,

其中,所述规则指定从序列参数集中指示或省略以下子图片相关语法元素中的一个或多个:

指示参考相同序列参数集的所有子图片是否具有相同维度的第一语法元素,

指示具有所述相同维度的所有子图片的共同宽度和共同高度的第二语法元素,或

当所有子图片不具有所述相同维度时,省略根据第二规则的一个或多个语法元素。

21. 根据权利要求20所述的方法,其中,所述规则指定:

表示为numSubpicCols的子图片列的数量被推导为等于 $(\text{sps_pic_width_max_in_luma_samples} / ((\text{sps_uniform_subpic_width_minus1} + 1) * \text{CtbSizeY}))$,

表示为numSubpicRows的子图片行的数量被推导为等于 $(\text{sps_pic_height_max_in_luma_samples} / ((\text{sps_uniform_subpic_height_minus1} + 1) * \text{CtbSizeY}))$,并且

表示为sumSubpics的子图片的数量被推导为等于 $\text{numSubpicCols} * \text{numSubpicRows}$ 。

22. 根据权利要求20所述的方法,其中,所述规则指定从所述序列参数集中排除 $\text{sps_num_subpics_minus1}$,并且 $\text{sps_num_subpics_minus1}$ 的值被推断为等于 $\text{numSubpics} - 1$ 。

23. 根据权利要求20所述的方法, 其中, 所述规则指定对于0到sps_num_subpics_minus1, 包括端值, 的范围内的每个i值, sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]的值被推断为分别等于 $(\text{sps_uniform_subpic_width_minus1}+1) * (i \% \text{numSubpicCols})$ 、 $(\text{sps_uniform_subpic_height_minus1}+1) * (i / \text{numSubpicCols})$ 、sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1。

24. 根据权利要求20所述的方法, 其中, 所述规则指定对于0到sps_num_subpics_minus1, 包括端值, 的范围内的每个i值, 当sps_independent_subpics_flag等于0时, sps_subpic_treated_as_pic_flag[i]和sps_loop_filter_across_subpic_enabled_flag[i]在所述序列参数集中指示。

25. 根据权利要求20所述的方法, 其中, 所述第二规则指定使用当前子图片层信令通知。

26. 根据权利要求20所述的方法, 其中, 所述规则指定当sps_pic_width_max_in_luma_samples小于或等于CtbSizeY时, sps_uniform_subpic_width_minus1从所述序列参数集中被排除并且被推断为等于0。

27. 根据权利要求20所述的方法, 其中, 所述规则指定当sps_pic_height_max_in_luma_samples小于或等于CtbSizeY时, 将sps_uniform_subpic_height_minus1从所述序列参数集被排除并且被推断为等于0。

28. 根据权利要求20所述的方法, 其中, 所述规则指定sps_uniform_subpic_width_minus1、sps_uniform_subpic_height_minus1、sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]是ue(v)编解码的。

29. 根据权利要求20所述的方法, 其中, 所述规则指定:sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1分别以与sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]相同的长度进行u(v)编解码。

30. 根据权利要求20所述的方法, 其中, 所述序列参数集包括:

指示所述视频的所有子图片是否分别具有相同的宽度和高度的第一语法元素,

指示所有子图片的共同宽度的第二语法元素, 以及

指示所有子图片的共同高度的第三语法元素。

31. 根据权利要求20所述的方法,

其中, 当序列参数集中指示的子图片的数量减1大于零并且当序列参数集中的语法元素指示编解码层视频序列中的所述视频的第一视频图片的第i个子图片在排除环路滤波操作的解码过程中被视为所述视频的第二视频图片时, 以下两个条件之一为真:

(1) 所述第一视频图片的第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片和第一视频图片对于以下各项中的每一项具有相同的值:

以亮度样点为单位的图片宽度,

以亮度样点为单位的图片高度,

所述子图片的数量减1, 以及

对于0到所述子图片的数量减1, 包括端值, 的范围内的每个i值, 所述第i个子图片的左

顶编解码树单元的水平位置、所述第*i*子图片的左顶编解码树单元CTU的垂直位置、所述第*i*个子图片的宽度减1、所述第*i*个子图片的高度减1、所述第*i*个子图片的语法元素、和所述第*i*个子图片的标识符值；和

(2) 所述第一参考图片列表或所述第二参考图片列表中的每个活动条目所参考的所述第二视频图片是层间参考图片,对于所述层间参考图片,所述子图片的数量减1的值等于零。

32. 一种处理视频数据的方法,包括:

对于包括一个或多个视频图片的视频和所述视频的比特流之间的转换,响应于参考以亮度样点为单位的所述图片参数集的每个视频图片的宽度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片列的宽度减1等于以编解码树块为单位的视频图片的宽度减1的第一语法元素;和

基于所述确定执行所述转换。

33. 一种处理视频数据的方法,包括:

对于包括一个或多个视频图片的视频和所述视频的比特流之间的转换,响应于参考以亮度样点为单位的所述图片参数集的每个视频图片的高度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片行的高度减1等于以编解码树块为单位的视频图片的高度减1的第一语法元素;和

基于所述确定执行所述转换。

34. 根据权利要求1至33中任一项所述的方法,其中,执行所述转换包括将所述视频编码到所述比特流中。

35. 根据权利要求1至33中任一项所述的方法,其中,执行所述转换包括从所述视频生成所述比特流,并且所述方法还包括将所述比特流存储在非暂时性计算机可读记录介质中。

36. 根据权利要求1至33中任一项所述的方法,其中,执行所述转换包括从所述比特流解码所述视频。

37. 一种视频解码装置,包括被配置为实现权利要求1至36中的一项或多项所述的方法的处理器。

38. 一种视频编码装置,包括被配置为实现权利要求1至36中的一项或多项所述的方法的处理器。

39. 一种其上存储有计算机指令的计算机程序产品,所述指令在由处理器执行时使所述处理器实现根据权利要求1至36中任一项所述的方法。

40. 一种非暂时性计算机可读存储介质,存储根据权利要求1至36中任一项所述的方法生成的比特流。

41. 一种存储指令的非暂时性计算机可读存储介质,所述指令使处理器实现根据权利要求1至36中任一项所述的方法。

42. 一种比特流生成方法,包括:

根据权利要求1至36中任一项所述的方法生成视频的比特流,以及
将所述比特流存储在计算机可读程序介质上。

43. 一种方法、装置、根据本文档中描述的公开方法或系统生成的比特流。

在视频编解码中使用子图片和片

[0001] 相关申请的交叉引用

[0002] 根据巴黎公约适用的专利法和/或规则,提出本申请是为了及时要求2020年5月27日提交的美国临时专利申请No.63/030,726的优先权和权益。出于法律指定的所有目的,上述申请的全部公开内容通过引用并入作为本申请的公开内容的部分。

技术领域

[0003] 该专利文档涉及图像和视频的编解码和解码。

背景技术

[0004] 在互联网和其他数字通信网络中,数字视频占用了最大的带宽。随着能够接收和显示视频的连接用户设备数量的增加,预计数字视频使用的带宽需求将继续增长。

发明内容

[0005] 本文公开了视频编码器和解码器可以使用的技术,用于使用对编解码表示的解码有用的控制信息来处理视频的编解码表示。

[0006] 在一个示例方面,公开了一种视频处理方法。该方法包括根据规则执行视频和视频的比特流之间的转换,其中,规则指定序列参数集包括在转换中使用的量化参数表,并且其中,规则指定是否或者如何根据序列参数集中的第二语法元素来控制所述序列参数集包括指示量化参数表中的点的数量的第一语法元素。

[0007] 在另一个示例方面,公开了一种视频处理方法。该方法包括根据规则执行视频和视频的比特流之间的转换,其中,规则指定当序列参数集中指示的子图片的数量减1大于零时,并且当序列参数集中的语法元素指示编解码层视频序列中的视频的第一视频图片的第*i*个子图片在排除环路滤波操作的解码过程中被视为视频的第二视频图片时,以下两个条件之一为真:(1) 第一视频图片的第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片和第一视频图片对于以下各项中的每一项具有相同的值:以亮度样点为单位的图片宽度,以亮度样点为单位的图片高度,子图片的数量减1,以及对于0到子图片的数量减1,包括端值,的范围内的每个*i*值,第*i*个子图片的左顶编解码树单元的水平位置、第*i*子图片的左顶CTU的垂直位置、第*i*个子图片的宽度减1、第*i*个子图片的高度减1、第*i*个子图片的语法元素;和(2) 第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片是层间参考图片,对于层间参考图片,子图片的数量减1的值等于零。

[0008] 在另一个示例方面,公开了一种视频处理方法。该方法包括根据规则执行包括视频块的视频和视频的比特流之间的转换,其中规则指定当仿射编解码工具被启用并且基于子块的时域运动矢量预测编解码工具被禁用时,子块merge候选的最大数量不允许为数量*N*。

[0009] 在另一个示例方面,公开了一种视频处理方法。该方法包括执行包括一个或多个

视频图片的视频和视频的比特流之间的转换,其中比特流符合格式规则,并且格式规则指定指示是否启用仿射merge模式的语法元素是否在比特流中。

[0010] 在另一个示例方面,公开了一种视频处理方法。该方法包括根据规则执行包括一个或多个视频图片的视频和视频的比特流之间的转换,一个或多个视频图片包括一个或多个子图片,其中,规则指定从序列参数集中指示或省略以下子图片相关语法元素中的一个或多个:指示参考相同序列参数集的所有子图片是否具有相同维度的第一语法元素,指示具有相同维度的所有子图片的共同宽度和共同高度的第二语法元素,或当所有子图片不具有相同维度时,省略根据第二规则的一个或多个语法元素。

[0011] 在另一个示例方面,公开了一种视频处理方法。该方法包括为包括一个或多个视频图片的视频和视频的比特流之间的转换,响应于参考以亮度样点为单位的图片参数集的每个视频图片的宽度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片列的宽度减1等于以编解码树块为单位的视频图片的宽度减1的第一语法元素;和基于确定执行转换。

[0012] 在另一个示例方面,公开了一种视频处理方法。该方法包括为包括一个或多个视频图片的视频和视频的比特流之间的转换,响应于参考以亮度样点为单位的图片参数集的每个视频图片的高度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片行的高度减1等于以编解码树块为单位的视频图片的高度减1的第一语法元素;和基于确定执行转换。

[0013] 在另一个示例方面,公开了一种视频处理方法。该方法包括执行包括视频图片的视频和该视频的编解码表示之间的转换,其中该编解码表示符合格式规则,其中该格式规则指定指示在所述转换期间使用的一个或多个初始量化参数(QP)值的一个或多个语法元素是否以及在哪里被包括在该编解码表示中。在又一示例方面,公开了一种视频编码器装置。该视频编码器包括被配置为实现上述方法的处理器。

[0014] 在另一个示例方面,公开了另一种视频处理方法。该方法包括为视频的视频块和视频的编解码表示之间的转换,确定使用初始量化参数(QP)值;以及基于该确定来执行转换,其中编解码表示中的语法元素使用偏移关系来指示初始QP值,该偏移关系包括偏移值*K*,该偏移值*K*是:(a) 1和25之间或者(b) 大于26,或者(c) 用于转换的最大允许QP值的函数,或者(d) 用于转换的比特深度的函数。

[0015] 在另一个示例方面,公开了另一种视频处理方法。该方法包括为视频的色度块和视频的编解码表示之间的转换,根据规则,基于包含色度块的图片或条带的类型,确定一个或多个量化参数(QP)表;以及基于该确定来执行转换。

[0016] 在另一个示例方面,公开了另一种视频处理方法。该方法包括为视频的色度块和视频的编解码表示之间的转换确定一个或多个量化参数(QP)表;并且基于该确定来执行转换;其中该转换符合指定对应于一个或多个QP表的一个或多个语法元素的第一规则或者指定一个或多个QP表和色度块的编解码特性之间的关系的第二规则。

[0017] 在另一个示例方面,公开了另一种视频处理方法。该方法包括执行视频的色度块和视频的编解码表示之间的转换;其中编解码表示符合格式规则,其中该格式规则指定与用于该转换的一个或多个色度量化的参数(QP)表相关联的语法元素的特性。

[0018] 在另一个示例方面,公开了另一种视频处理方法。该方法包括基于与编解码表示

中的一个或多个语法元素相关的格式规则,确定用于在视频的编解码表示中信令通知量化参数表的信令通知方案,并且基于该确定,执行视频和编解码表示之间的转换。

[0019] 在另一个示例方面,公开了另一种视频处理方法。该方法包括根据规则执行视频的视频块和视频的编解码表示之间的转换,其中该规则指定数字N,在仿射编解码的使用被启用并且基于子块的时域运动矢量预测的使用被禁用的情况下,该数字N被用于限制子块merge候选的最大数量。

[0020] 在另一个示例方面,公开了另一种视频处理方法。该方法包括执行包括一个或多个视频图片的视频和视频的编解码表示之间的转换,该一个或多个视频图片包括一个或多个视频子图片,其中编解码表示符合格式规则,其中格式规则指定包括或省略一个或多个以下与子图片相关的语法元素:指示参考相同序列参数集的所有子图片是否具有相同维度的第一语法元素,信令通知具有相同维度的所有子图片的公共宽度和公共高度的第二语法元素,或者根据第二格式规则的第三语法元素。

[0021] 在又一示例方面,公开了一种视频解码器装置。视频解码器包括被配置为实现上述方法的处理器。

[0022] 在又一个示例方面,公开了一种其上存储有代码的计算机可读介质。该代码以处理器可执行代码的形式体现了本文描述的方法之一。

[0023] 这些和其他特征在整个本文档中都进行了描述。

附图说明

[0024] 图1是在依赖量化中使用的两个标量量化器的图示;

[0025] 图2示出了用于依赖量化的状态转变和量化器选择的示例;

[0026] 图3是一个示例视频处理系统的框图;

[0027] 图4是一个视频处理装置的框图;

[0028] 图5是视频处理的一个示例方法的流程图;

[0029] 图6是示出根据本公开的一些实施例的一个视频编解码系统的框图;

[0030] 图7是示出根据本公开的一些实施例的一个编码器的框图;

[0031] 图8是示出根据本公开的一些实施例的一个解码器的框图;以及

[0032] 图9至图15是视频处理的示例方法的流程图。

具体实施方式

[0033] 本文档中使用章节标题是为了易于理解并且不将各章节中公开的技术和实施例的适用性仅限制于该章节。此外,在一些描述中使用H.266术语仅仅是为了易于理解,而不是为了限制所公开技术的范围。因此,本文描述的技术也适用于其他视频编解码器协议和设计。

[0034] 1. 引言

[0035] 本发明涉及视频编解码技术。具体地,本发明涉及视频编解码中的初始量化参数(QP)和色度QP表。本发明可以被应用于现有的视频编解码标准(如HEVC)、或者待完成的标准(多功能视频编解码)。本发明也可以适用于未来的视频编解码标准或视频编解码器。

[0036] 2. 缩写

[0037]	APS	自适应参数集
[0038]	AU	接入单元
[0039]	AUD	接入单元分隔符
[0040]	AVC	高级视频编解码
[0041]	CLVS	编解码层视频序列
[0042]	CPB	编解码图片缓冲区
[0043]	CRA	完全随机接入
[0044]	CTU	编解码树单元
[0045]	CVS	编解码视频序列
[0046]	DPB	解码图片缓冲区
[0047]	DPS	解码参数集
[0048]	EOB	比特流结尾
[0049]	EOS	序列结尾
[0050]	GDR	逐步解码刷新
[0051]	HEVC	高效视频编解码
[0052]	HRD	假设参考解码器
[0053]	IDR	即时解码刷新
[0054]	JEM	联合探索模型
[0055]	MCTS	运动约束片集
[0056]	NAL	网络抽象层
[0057]	OLS	输出层集
[0058]	PH	图片标头
[0059]	PPS	图片参数集
[0060]	PTL	档次、层和级别
[0061]	PU	图片单元
[0062]	QP	量化参数
[0063]	RBSP	原始字节序列有效负载
[0064]	SEI	辅助增强信息
[0065]	SPS	序列参数集
[0066]	SVC	可缩放视频编解码
[0067]	VCL	视频编解码层
[0068]	VPS	视频参数集
[0069]	VTM	VVC测试模型
[0070]	VUI	视频可用性信息
[0071]	VVC	多功能视频编解码

[0072] 3. 视频编解码介绍

[0073] 视频编解码标准主要是通过众所周知的ITU-T和ISO/IEC标准的发展而得以演进。ITU-T制作了H.261和H.263标准,ISO/IEC制作了MPEG-1和MPEG-4Visual标准,并且两个组织联合制作了H.262/MPEG-2视频标准和H.264/MPEG-4高级视频编解码(AVC)标准和H.265/

HEVC标准。从H.262开始,视频编解码标准基于混合视频编解码结构,其中利用时域预测加变换编解码。为了探索HEVC之外的未来视频编解码技术,由VCEG和MPEG于2015年联合成立联合视频探索团队(JVET)。从那时起,JVET采用了许多新方法并将其纳入名为联合探索模型(JEM)的参考软件。2018年4月,VCEG(Q6/16)和ISO/IEC JTC1SC29/WG11(MPEG)之间的联合视频专家组(JVET)成立以致力于VVC标准,其目标是与HEVC相比降低50%的比特率。

[0074] VVC草案的最新版本,即多功能视频编解码(草案8)可以在以下地址找到:

[0075] http://phenix.int-evry.fr/jvet/doc_end_user/documents/17_Brussels/wg11/JVET-Q2001-v13.zip

[0076] 并且最新的测试模型软件可以在以下地址找到:https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/archive/VTM-8.0/VVCSoftware_VTM-VTM-8.0.zip

[0077] 3.1. 量化参数控制

[0078] 在VVC中,最大QP从51扩展到63,并且初始QP的信令通知也相应改变。当slice_qp_delta的非零值被编解码时,SliceQpY的初始值在条带段层被修改。具体地,init_qp_minus26的值被修改为在 $(-26+QpBdOffsetY)$ 到+37的范围内。当变换块的尺寸不是4的幂时,变换系数与对QP或QP级别缩放表的修改一起被处理,而不是乘以181/256(或181/128),以补偿变换过程的隐式缩放。对于变换跳过块,最小允许量化参数(QP)被定义为4,因为当QP等于4时,量化步长变为1。

[0079] 在HEVC中(也在H.264中),固定的查找表用于将亮度量化参数QP_Y转换为色度量化参数QP_C。在VVC中,使用更灵活的亮度到色度QP映射。不使用固定的表,而是使用灵活的分段线性模型在SPS中信令通知亮度到色度QP映射关系,对线性模型的唯一约束是每个片段的斜率不能为负(即,随着亮度QP增加,色度QP必须保持平坦或增加,但不能减少)。分段线性模型通过以下来定义:1)模型中的片段数量;2)该片段的输入(亮度)和输出(色度)增量QP。分段线性模型的输入范围为 $[-QpBdOffsetY, 63]$,并且分段线性模型的输出范围为 $[-QpBdOffsetC, 63]$ 。对于Cb、Cr和联合Cb/Cr编解码,可以单独信令通知QP映射关系,或者对于所有三种类型的残差编解码,可以联合信令通知QP映射关系。

[0080] 与HEVC相同,在VVC中允许CU级别QP自适应。亮度和色度分量的增量QP值可以被单独信令通知。对于色度分量,允许的色度QP偏移值以与HEVC中类似的方式在PPS中以偏移列表的形式被信令通知。对于Cb、Cr和联合Cb/Cr编解码,该列表被单独定义。针对Cb、Cr和联合Cb/Cr列表中的每一个,允许多达6个偏移值。在CU级别,信令通知索引以指示偏移列表中的哪一个偏移值用于调整该CU的色度QP。CU色度QP偏移信令通知也与VPDU CU QP增量可用性一致,并且对于大于 64×64 的CU,不管其是否具有非零CBF,都用第一变换单元发送色度QP偏移。

[0081] 3.2. 依赖量化

[0082] 此外,相同的HEVC标量量化与称为依赖标量量化的新概念一起使用。依赖标量量化是指一种方法,其中变换系数的容许重构值组依赖于按照重构顺序在当前变换系数级别之前的变换系数级别的值。该方法的主要效果是,与如HEVC所使用的传统独立标量量化相比,容许重构矢量在N维矢量空间中被压缩得更密集(N表示变换块中的变换系数的数量)。这意味着,对于每N维单位体积的给定平均数量的容许重构矢量,输入矢量和最接近的重构

矢量之间的平均失真减小。依赖标量量化的方法是通过以下方式实现的：(a) 定义具有不同重构级别的两个标量量化器，以及 (b) 定义在两个标量量化器之间切换的过程。

[0083] 在图1中示出了所使用的两个标量量化器，由Q0和Q1表示。可用重构级别的位置由量化步长 Δ 唯一指定。不在比特流中显式地信令通知所使用的标量量化器 (Q0或Q1)。相反，用于当前变换系数的量化器通过按照编解码/重构顺序在当前变换系数之前的变换系数级别的奇偶性来确定。

[0084] 如图2所示，两个标量量化器 (Q0和Q1) 之间的切换经由具有四个状态的状态机来实现。该状态可以取四个不同的值：0, 1, 2, 3。它通过按照编解码/重构顺序在当前变换系数之前的变换系数级别的奇偶性来唯一确定。在变换块的逆量化的开始时，状态被设置为等于0。变换系数按照扫描顺序 (即，按照与它们被熵解码的相同的顺序) 被重构。在重构当前变换系数之后，状态被更新，如图18所示，其中k表示变换系数级别的值。

[0085] 3.3. 缩放矩阵

[0086] VVC支持使用默认的缩放矩阵或信令通知用户定义的缩放矩阵。DEFAULT (默认) 模式缩放矩阵都是平坦的，其中对于所有TB尺寸，元素等于16。IBC和帧内编解码模式当前共享相同的缩放矩阵。因此，对于USER_DEFINED矩阵的情况，MatrixType和MatrixType_DC的数量被更新如下：

[0087] -MatrixType:30=2 (对于帧内&IBC/帧间为2) \times 3 (Y/Cb/Cr分量) \times 5 (正方形TB尺寸:对于亮度从 4×4 到 64×64 ,对于色度从 4×4 到 32×32)

[0088] -MatrixType_DC:14=2 (对于帧内&IBC/帧间为2 \times 对于Y分量为1) \times 3 (TB

[0089] 尺寸: $16 \times 16, 32 \times 32, 64 \times 64$) +4 (对于帧内&IBC/帧间为2 \times 对于Cb/Cr分量为2) \times 2 (TB尺寸: $16 \times 16, 32 \times 32$)

[0090] 对于以下缩放矩阵单独编解码的DC值: $16 \times 16, 32 \times 32$ 和 64×64 。对于尺寸小于 8×8 的TB，一个缩放矩阵中的所有元素都被信令通知。如果TB具有大于或等于 8×8 的尺寸，则仅一个 8×8 缩放矩阵中的64个元素作为基本缩放矩阵被信令通知。为了获得尺寸大于 8×8 的正方形矩阵， 8×8 基本缩放矩阵被上采样 (通过复制元素) 到对应的正方形尺寸 (即， $16 \times 16, 32 \times 32, 64 \times 64$)。当应用64点变换的高频系数的置零时，缩放矩阵的对应高频也被置零。也就是说，如果TB的宽度或高度大于或等于32，则仅保留系数的左半部分或上半部分，并且剩余系数被赋值为零。此外，针对 64×64 缩放矩阵信令通知的元素的数量也从 8×8 减少到三个 4×4 子矩阵，因为右下方的 4×4 元素从未被使用。在VVC中，不存在 $2 \times 2, 2 \times 4$ 和 4×2 色度帧内编解码块 (CB)，并且最小的帧内块尺寸等于 2×8 和 8×2 、以及最小的色度帧内块复制 (IBC) 块尺寸。此外，对 4×4 亮度CB禁用帧间预测。因此，小的 2×2 色度块只能通过应用子块变换 (SBT) 来创建。考虑到这些本质， 2×2 帧内色度量化的矩阵 (QM) 从默认QM列表中被移除，并且不对该尺寸的用户定义的帧内QM进行编解码。

[0091] 为了提高用户定义的量化矩阵 (QM) 的编解码效率，考虑以下方法。

[0092] -允许参考与当前QM具有相同基本尺寸的先前编解码的QM。

[0093] -允许对当前QM和参考QM之间的元素到元素差值进行编解码。

[0094] -保持当前QM内元素的原始DPCM编解码。

[0095] -使用组合matrixId和sizeId的单个矩阵标识符scalingListId。

[0096] 3.4. 色度残差的联合编解码

[0097] VVC支持色度残差被联合编解码的模式;这被称为JCCR(色度残差的联合编解码)。联合色度编解码模式的使用(激活)由TU级别标志tu_joint_cbr_residual_flag指示,并且所选择的模式由色度CBF隐式指示。如果TU的一个或两个色度CBF等于1,则存在标志tu_joint_cbr_residual_flag。在PPS和条带标头中,针对联合色度残差编解码模式信令通知色度QP偏移值,以区别于针对常规色度残差编解码模式信令通知的通常色度QP偏移值。这些色度QP偏移值用于推导使用联合色度残差编解码模式进行编解码的这些块的色度QP值。当对应的联合色度编解码模式(下表中的模式2)在TU中被激活时,在该TU的量化和解码期间,该色度QP偏移被添加到应用的亮度推导色度QP。对于其他模式(下表中的模式1和3),色度QP以与传统Cb或Cr块相同的方式被推导。在表1中描绘了从传输的变换块中重构色度残差(resCb和resCr)的过程。当激活该模式时,一个单个联合色度残差块(下表中的resJointC[x][y])被信令通知,并且Cb的残差块(resCb)和Cr的残差块(resCr)是在考虑诸如tu_cbf_cb、tu_cbf_cr和CSign的信息(其是条带标头中指定的符号值)的情况下被推导的。

[0098] 在编码器侧,联合色度分量的推导如下所述。根据(上表中列出的)模式,编码器生成resJointC{1,2},如下所示:

[0099] -如果模式等于2(重构为Cb=C、Cr=CSign*C的单个残差),则联合残差根据下式确定

$$[0100] \quad \text{resJointC}[x][y] = (\text{resCb}[x][y] + \text{CSign} * \text{resCr}[x][y]) / 2$$

[0101] -否则,如果模式等于1(重构为Cb=C、Cr=(CSign*C)/2的单个残差),则联合残差根据下式确定

$$[0102] \quad \text{resJointC}[x][y] = (4 * \text{resCb}[x][y] + 2 * \text{CSign} * \text{resCr}[x][y]) / 5$$

[0103] -否则(模式等于3,即单个残差,重构Cr=C、Cb=(CSign*C)/2),联合残差根据下式确定

[0104] $\text{resJointC}[x][y] = (4 * \text{resCr}[x][y] + 2 * \text{CSign} * \text{resCb}[x][y]) / 5$ 表1.色度残差的重构。值CSign是符号值(+1或-1),其在条带标头中指定,resJointC[][]是传输的残差。

tu_cbf_c b	tu_cbf_cr	Cb 和 Cr 残差的重构	模式
1	0	$\text{resCb}[x][y] = \text{resJointC}[x][y]$ $\text{resCr}[x][y] = (\text{CSign} * \text{resJointC}[x][y]) \gg 1$	1
1	1	$\text{resCb}[x][y] = \text{resJointC}[x][y]$ $\text{resCr}[x][y] = \text{CSign} * \text{resJointC}[x][y]$	2
0	1	$\text{resCb}[x][y] = (\text{CSign} * \text{resJointC}[x][y]) \gg 1$ $\text{resCr}[x][y] = \text{resJointC}[x][y]$	3

[0106] 上述三个联合色度编解码模式仅在I条带中支持。在P条带和B条带中,仅支持模式2。因此,在P条带和B条带中,语法元素tu_joint_cbr_residual_flag仅在两个色度cbf都为1时存在。注意,在tu_cbf_luma和tu_cbf_cb的上下文建模中移除了变换深度。

[0107] 3.5.SPS中的色度QP表

[0108] 在JVET-Q2001-vC的条款7.3.2.3中,SPS包括名为色度QP表的结构,如下所示:

	描述符
seq_parameter_set_rbsp() {	
.....	
if(ChromaArrayType != 0) {	
sps_joint_cbr_enabled_flag	u(1)
same_qp_table_for_chroma	u(1)
numQpTables = same_qp_table_for_chroma ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
for(i = 0; i < numQpTables; i++) {	
[0109] qp_table_start_minus26[i]	se(v)
num_points_in_qp_table_minus1[i]	ue(v)
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {	
delta_qp_in_val_minus1[i][j]	ue(v)
delta_qp_diff_val[i][j]	ue(v)
}	
}	
}	
}	
.....	

[0110] 它们具有以下语义和QP表推导:

[0111] sps_joint_cbr_enabled_flag等于0指定色度残差的联合编解码被禁用。sps_joint_cbr_enabled_flag等于1指定色度残差的联合编解码被启用。当不存在时,sps_joint_cbr_enabled_flag的值被推断为等于0。

[0112] same_qp_table_for_chroma等于1指定仅信令通知一个色度QP映射表,并且当sps_joint_cbr_enabled_flag等于1时,该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差。same_qp_table_for_chroma等于0指定在SPS中信令通知色度QP映射表,当sps_joint_cbr_enabled_flag等于1时,对于Cb和Cr信令通知两个色度QP映射表,并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当same_qp_table_for_chroma不存在于比特流中时,same_qp_table_for_chroma的值被推断为等于1。

[0113] qp_table_start_minus26[i]加26指定用于描述第i个色度QP映射表的起始亮度和色度QP。qp_table_start_minus26[i]的值应当在-26-QpBdOffset到36的范围(包括端值)内。当qp_table_start_minus26[i]不存在于比特流中时,qp_table_start_minus26[i]的值被推断为等于0。

[0114] num_points_in_qp_table_minus1[i]加1指定用于描述第i个色度QP映射表的点

的数量。num_points_in_qp_table_minus1[i]的值应当在0到63+QpBdOffset的范围(包括端值)内。当num_points_in_qp_table_minus1[0]不存在于比特流中时,num_points_in_qp_table_minus1[0]的值被推断为等于0。

[0115] delta_qp_in_val_minus1[i][j]指定用于推导第i个色度QP映射表的第j个枢轴点的输入坐标的增量值。当delta_qp_in_val_minus1[0][j]不存在于比特流中时,delta_qp_in_val_minus1[0][j]的值被推断为等于0。

[0116] delta_qp_diff_val[i][j]指定用于推导第i个色度QP映射表的第j个枢轴点的输出坐标的增量值。

[0117] 对于 $i=0\cdots\text{numQpTables}-1$,第i个色度QP映射表ChromaQpTable[i]被推导如下:

$$\text{qpInVal}[i][0] = \text{qp_table_start_minus26}[i] + 26$$

$$\text{qpOutVal}[i][0] = \text{qpInVal}[i][0]$$

```
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {
```

$$\text{qpInVal}[i][j+1] = \text{qpInVal}[i][j] + \text{delta_qp_in_val_minus1}[i][j] + 1$$

$$\text{qpOutVal}[i][j+1] = \text{qpOutVal}[i][j] +$$

$$(\text{delta_qp_in_val_minus1}[i][j] \wedge \text{delta_qp_diff_val}[i][j])$$

```
}
```

[0118] ChromaQpTable[i][qpInVal[i][0]] = qpOutVal[i][0]

```
for(k = qpInVal[i][0] - 1; k >= -QpBdOffset; k--)
```

$$\text{ChromaQpTable}[i][k] = \text{Clip3}(-\text{QpBdOffset}, 63, \text{ChromaQpTable}[i][k+1] - 1)$$

```
for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {
```

$$\text{sh} = (\text{delta_qp_in_val_minus1}[i][j] + 1) \gg 1$$

```
for(k = qpInVal[i][j] + 1, m = 1; k <= qpInVal[i][j+1]; k++, m++)
```

$$\text{ChromaQpTable}[i][k] = \text{ChromaQpTable}[i][\text{qpInVal}[i][j]] +$$

$$((\text{qpOutVal}[i][j+1] - \text{qpOutVal}[i][j]) * m + \text{sh}) /$$

$$(\text{delta_qp_in_val_minus1}[i][j] + 1)$$

```
}
```

```
for(k = qpInVal[i][num_points_in_qp_table_minus1[i] + 1] + 1; k <= 63; k++)
```

$$\text{ChromaQpTable}[i][k] = \text{Clip3}(-\text{QpBdOffset}, 63, \text{ChromaQpTable}[i][k-1] + 1)$$

[0120] 当same_qp_table_for_chroma等于1时,ChromaQpTable[1][k]和ChromaQpTable[2][k]被设置为等于ChromaQpTable[0][k],k在-QpBdOffset到63的范围(包括端值)内。

[0121] 比特流一致性的要求是,qpInVal[i][j]和qpOutVal[i][j]的值应当在-QpBdOffset到63的范围(包括端值)内,i在0到numQpTables-1的范围(包括端值)内,并且j在0到num_points_in_qp_table_minus1[i]+1的范围(包括端值)内。

[0122] 在以上描述中,QpBdOffset被推导为:

[0123] bit_depth_minus8指定亮度和色度数组的样点的比特深度BitDepth、以及亮度和

色度量量化参数范围偏移 $QpBdOffset$ 的值,如下所示:

[0124] $BitDepth = 8 + bit_depth_minus8$

[0125] $QpBdOffset = 6 * bit_depth_minus8$

[0126] bit_depth_minus8 应当在0到8的范围(包括端值)内。

[0127] 3.6.PPS中的初始QP

[0128] PPS中有命名为 $init_qp_minus26$ 的语法元素。语义如下所示: $init_qp_minus26$ 加26指定参考PPS的每个条带的 $SliceQp_Y$ 的初始值。当 ph_qp_delta 的非零值被解码时, $SliceQp_Y$ 的初始值在图片级别被修改,或者当 $slice_qp_delta$ 的非零值被解码时, $SliceQp_Y$ 的初始值在条带级别被修改。 $init_qp_minus26$ 的值应当在 $-(26+QpBdOffset)$ 到+37的范围(包括端值)内。

[0129] 当 $qp_delta_info_in_ph_flag$ 等于1时,图片的所有条带的 Qp_Y 量化参数的初始值 $SliceQp_Y$ 被推导如下:

[0130] $SliceQp_Y = 26 + init_qp_minus26 + ph_qp_delta$

[0131] 当 $qp_delta_info_in_ph_flag$ 等于0时,条带的 Qp_Y 量化参数的初始值 $SliceQp_Y$ 被推导如下:

[0132] $SliceQp_Y = 26 + init_qp_minus26 + slice_qp_delta$

[0133] $SliceQp_Y$ 的值应当在 $-QpBdOffset$ 到+63的范围(包括端值)内。

[0134] 4. 公开的技术方案及实施例所解决的技术问题

[0135] 1. 初始QP值(即, $init_qp_minus26$)的信令通知可能不合理。

[0136] 2. I和B/P图片或条带可能需要不同的色度QP表,然而,其在当前的VVC草案文本中共享相同的色度QP表。

[0137] 3. 变换跳过可能需要另一个色度QP表。

[0138] 4. 色度QP表起始点应该覆盖整个QP范围,在当前设计中并非如此。

[0139] 5. 色度QP表可能具有偏置的起始点。

[0140] 6. 色度QP表中的点的数量应该能够为0,但是这在当前设计中是不允许的,因为 $num_points_in_qp_table_minus1$ 被编解码,并且语法元素的最小允许值等于0,因此点的最小数量为1。

[0141] 7. 在色度QP的推导中指定QP裁剪可能比指定限制经推导的色度QP值的比特流一致性约束更好,因为前者将防止违规的发生。

[0142] 8. 在最新的VVC草案文本中,SH语法元素 $slice_ts_residual_coding_disabled_flag$ 用于指定基于变换跳过的残差编解码(TSRC)或常规残差编解码(RRC)是否用于变换块。然而,对于当前块是否使用TSRC或RRC,可能有更高级别(SPS/PPS)和/或更低级别(CU/TU)级别的控制标志。此外,不同级别控制标志之间以及控制标志和变换跳过标志之间的相互作用将被进一步指定。

[0143] 9. 在当前VVC中,当在SPS/PH/SH禁用亮度ALF时,色度ALF和CC-ALF被隐式禁用。然而,不在CTU级别应用这样的限制。当对CTU禁用亮度ALF时,色度ALF和CC-ALF仍然可以被应用于CTU。这样的设计与在更高级别基于亮度ALF来控制色度ALF/CC-ALF的意图冲突。

[0144] 10. 色度QP表可能有一组默认值,取决于颜色格式。

[0145] 11. 在JVET-Q2001-vE中最新VVC草案的条款C.6中指定的子比特流提取过程中,一

个步骤被指定如下：

[0146] -从outBitstream中移除以下所有条件为真的所有NAL单元：

[0147] -nal_unit_type不等于IDR_W_RADL、IDR_N_LP或CRA_NUT。

[0148] -nuh_layer_id等于LayerIdIn0ls[target0lsIdx][j]，j的值在0到NumLayersIn0ls[target0lsIdx]-1的范围（包括端值）内。

[0149] -TemporalId大于或等于NumSubLayersInLayerIn0LS[target0lsIdx][j]。

[0150] 然而，这意味着满足所有三个条件的参数集等非VCL NAL单元也将被移除。这是不希望，并且参数集（例如PPS和APS）可以由具有不同TemporalId值的图片共享，并且可以放置在比特流的开头。

[0151] 12. 当前规范分别允许8个ALF_APS、8个SCALING_APS和3个LMCS_APS。然而，图片所允许的APS NAL单元的数量可能是不够的，例如，典型的使用情况是图片包含96个子图片，并且每个子图片可能需要不同的ALF_APS。在这种情况下，当前的8个ALF_APS NAL单元不足以实现良好的编解码效率。因此，需要支持更多数量的APS。

[0152] 13. JVET-S0071建议在JVET-R2001-vA/v10中的VVC设计中添加用于信令通知子图片布局的快捷方式，如下所示：

[0153] a. 添加sps_subpic_same_res_flag，sps_subpic_same_res_flag等于1指定所有子图片分别具有相同的宽度和高度。

[0154] b. 当sps_subpic_same_res_flag等于1时，sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]仅在i等于0时信令通知，并且sps_subpic_ctu_top_left_x[i]和sps_subpic_ctu_top_left_y[i]对于所有i值都被跳过。

[0155] 然而，语法设计有些混乱，并且当所有子图片分别具有相同的宽度和高度时，可以推导子图片的数量，因此不应该被信令通知。

[0156] 5. 实施例和解决方案的示例列表

[0157] 为了解决以上问题和其他问题，公开了总结如下的方法。这些项目应该被认为是解释一般概念的示例，而不应该以狭隘的方式被解释。此外，这些项目可以被单独应用或者以任何方式组合。

[0158] 在下文中， $\text{floor}(x)$ 表示返回小于或等于x的最大整数的函数。

[0159] 1. PPS中用于指定初始QP值的初始QP值语法元素（例如，当前在JVET-Q2001-vC中表示为init_qp_minus26）可以被移动到PH。

[0160] a. 可替代地，初始QP值语法元素可以在PH重复。

[0161] b. 可替代地，可以在PPS和PH两者中信令通知初始QP值语法元素，并且在PH中信令通知的初始QP值语法元素可以覆盖或更新（通过添加信令通知的增量）在PPS中信令通知的初始QP值语法元素。

[0162] c. 可替代地，初始QP值语法元素可以在SPS中被信令通知，也可能在PPS、PH和SH中的一个或多个中被信令通知，并且当存在时，更低级别的值覆盖或更新（通过添加信令通知的增量）在更高级别信令通知的值。在更新的情况下，在最高级别信令通知的初始QP值是被 $\text{ue}(v)$ 编解码的，并且在更低级别信令通知的增量值是被 $\text{se}(v)$ 编解码的。

[0163] 2. 可以根据特定类型在SPS/PPS/PH/SH中信令通知用于初始QP值的指示的多个语法元素。

- [0164] a. 在一个示例中,它们中的每一个可以对应于特定类型。
- [0165] b. 在一个示例中,特定类型可以包括图片/条带类型(例如,I/P/B;帧内/帧间)。
- [0166] c. 在一个示例中,特定类型可以包括视频内容类型(例如,屏幕内容或相机捕捉的内容)。
- [0167] d. 在一个示例中,特定类型可以包括子图片的索引或者其他子图片标识信息,即,不同的子图片可以与初始QP推导的不同语法元素相关联。
- [0168] e. 在一个示例中,特定类型可以包括条带的索引或者其他条带标识信息,即,不同的条带可以与初始QP推导的不同语法元素相关联。
- [0169] f. 在一个示例中,特定类型可以包括片的索引或者其他片标识信息,即,不同的片可以与初始QP推导的不同语法元素相关联。
- [0170] g. 在一个示例中,特定类型可以包括变换类型。(例如,变换跳过模式或不是变换跳过模式)。
- [0171] 3. 初始QP值语法元素(例如,在SH/PH/PPS/SPS中)可以偏移不等于26的数K。
- [0172] a. 在一个示例中,K小于26。
- [0173] b. 在一个示例中,语法元素可以被 init_qp_minusK 替换,和/或语法元素的值应当在 $-(K+\text{QpBdOffset})$ 到 $(63-K)$ 的范围(包括端值)内,其中K小于26。
- [0174] i. 在一个示例中,K等于20。此外,可替代地,语法元素可以被 init_qp_minus20 替换,和/或语法元素的值应当在 $-(20+\text{QpBdOffset})$ 到43(即 $63-20$)的范围(包括端值)内。
- [0175] c. 在一个示例中,K大于26。
- [0176] i. 在一个示例中,语法元素可以被 init_qp_minusK 替换,并且语法元素的值应当在 $-(K+\text{QpBdOffset})$ 到 $(63-K)$ 的范围(包括端值)内,其中K是大于26的常数。
- [0177] ii. 在一个示例中,K等于32。此外,可替代地,语法元素可以用 init_qp_minus32 替换,并且语法元素的值应当在 $-(32+\text{QpBdOffset})$ 到31(即 $63-32$)的范围(包括端值)内。
- [0178] d. 可替代地,K可以被设置为最大允许QP和/或最小允许QP的函数。例如,K可以被设置为等于 $(\text{最大允许QP}-\text{最小允许QP})/2$ 或 $(\text{最大允许QP}+1-\text{最小允许QP})/2$ 。
- [0179] e. 在以上示例中,K可以依赖于图片/条带/块类型;和/或预测模式和/或比特深度。
- [0180] i. 在一个示例中,I条带/图片的K的值不大于P/B条带/图片的K的值。
- [0181] 4. 初始QP值语法元素(例如,在SH/PH/PPS中)可以偏移依赖于内部比特深度的数量。
- [0182] a. 在一个示例中,初始QP值语法元素可以偏移
- [0183] $\text{floor}((-QpBdOffset+M)/2)$ 。
- [0184] i. 此外,可替代地,语法元素的值加 $\text{floor}((-QpBdOffset+M)/2)$ 应当在 $-QpBdOffset$ 到63的范围(包括端值)内。
- [0185] b. 在一个示例中,初始QP值语法元素可以偏移 $(\text{floor}((-QpBdOffset+M)/2)+K)$,其中K是常数。
- [0186] i. 此外,可替代地,语法元素的值加 $(\text{floor}((-QpBdOffset+M)/2)+K)$ 应当在 $-QpBdOffset$ 到N的范围(包括端值)内。
- [0187] c. 在以上示例中,N可以被设置为最大允许QP值(例如,63)。

- [0188] d. 在以上示例中, M可以被设置为最大允许QP值(例如, 63)或最大允许QP值加/减1。
- [0189] 5. 色度QP表可以根据图片/条带类型来确定。
- [0190] a. 在一个示例中, 不同的图片/条带类型可以具有不同的色度QP表组。
- [0191] b. 在一个示例中, I图片/条带可以具有它们自己的色度QP表。
- [0192] c. 在一个示例中, I、B和P图片/条带可以各自具有它们自己的色度QP表。
- [0193] d. 在一个示例中, B和P图片/条带可以共享相同的色度QP表组。
- [0194] e. 在一个示例中, 图片/条带类型可以被分类为M种情况(对于仅为I、BP共享, $M=2$; 对于I/B/P分离, $M=3$)。要被信令通知的色度QP表的数量可以依赖于M。
- [0195] i. 在一个示例中, 要被信令通知的色度QP表的数量可以被设置为 $M * \text{numQpTables}$, 其中, ($\text{numQpTables} = \text{same_qp_table_for_chroma?1:}(\text{sps_joint_cbcr_enabled_flag?3:}2)$), 其中M大于1。
- [0196] 1) 此外, 可替代地, `same_qp_table_for_chroma`的语义可以被进一步修改为“`same_qp_table_for_chroma`等于1指定针对图片/条带类型仅信令通知一个色度QP映射表”。
- [0197] ii. 此外, 在一个示例中, 要被信令通知的色度QP表的数量可以被设置为($\text{same_qp_table_for_chroma?1:}(M * (\text{sps_joint_cbcr_enabled_flag?3:}2))$), 其中M大于1。
- [0198] iii. 在一个示例中, 对于每个类别, 首先信令通知所有块是否共享相同的色度QP表的指示, 并且QP表的数量可以被设置为($\text{same_qp_table_for_chroma?1:}(\text{sps_joint_cbcr_enabled_flag?3:}2)$), 随后是色度QP表的详细信息。
- [0199] 1) 此外, 可替代地, 对于所有类别, 可以进一步信令通知用于描述色度QP映射表的起始亮度和色度QP的指示, 例如使用实际值减K(例如, $K=26$)。
- [0200] 2) 此外, 可替代地, 可以进一步信令通知用于描述色度QP映射表的起始亮度和色度QP的指示, 例如使用实际值减K(例如, 对于帧内条带/图片, $K=1$, 或者对于帧间条带/图片, $K=32$), 并且K依赖于类别索引。
- [0201] 6. 可以将指示(例如, `sps_non_intra_present_flag`)添加到SPS中, 以指示当前序列是否可以包含B/P条带。
- [0202] a. 在一个示例中, 当标志为1时, 其指示当前序列中的所有条带都是帧内条带。
- [0203] b. 在一个示例中, 当标志为0时, 其指示在当前序列中可以有B/P条带。
- [0204] 7. 可以在SPS中信令通知一个或两个色度QP表组。
- [0205] a. 将标志添加到SPS, 例如, 名为`sps_one_set_of_chroma_qp_tables_flag`。
- [0206] b. `sps_one_set_of_chroma_qp_tables_flag`等于1指定SPS中一个色度QP表组的存在(即, 如在当前的VVC草案文本中), 并且该色度QP表组仅适用于帧内编解码实体(图片、条带、CTU、CU或编解码块)。
- [0207] i. 在一个示例中, 唯一的色度QP表组仅适用于参考SPS的CLVS中的帧内图片(对于该帧内图片, 所有条带都是帧内条带, 即I条带)。
- [0208] ii. 在一个示例中, 唯一的色度QP表组仅适用于参考SPS的CLVS中的帧内条带。
- [0209] iii. 在一个示例中, 唯一的色度QP表组仅适用于参考SPS的CLVS中的帧内CTU。
- [0210] iv. 在一个示例中, 唯一的色度QP表组仅适用于参考SPS的CLVS中的帧内CU。

- [0211] v. 在一个示例中,唯一的色度QP表组仅适用于参考SPS的CLVS中的帧内编解码块。
- [0212] c. `sps_one_set_of_chroma_qp_tables_flag`等于1指定SPS中一个色度QP表组的存在(即,如在当前的VVC草案文本中),并且该色度QP表组适用于帧内和帧间编解码实体(图片、条带、CTU、CU或编解码块)两者。
- [0213] d. `sps_one_set_of_chroma_qp_tables_flag`等于0指定SPS中两个色度QP表组的存在(即,添加一个或多个色度QP表组),第0色度QP表组仅适用于帧内编解码实体(图片、条带、CTU、CU或编解码块),并且第1色度QP表组仅适用于帧间编解码实体(图片、条带、CTU、CU或编解码块)
- [0214] i. 在一个示例中,第0色度QP表组仅适用于参考SPS的CLVS中的帧内图片(对于该帧内图片,所有条带都是帧内条带),并且第1色度QP表组仅适用于参考SPS的CLVS中的帧间图片(对于该帧间图片,所有条带都是帧间条带,即B条带或P条带)。
- [0215] ii. 在一个示例中,第0色度QP表组仅适用于参考SPS的CLVS中的帧内条带,并且第1色度QP表组仅适用于参考SPS的CLVS中的帧间条带。
- [0216] iii. 在一个示例中,第0色度QP表组仅适用于参考SPS的CLVS中的帧内CTU,并且第1色度QP表组仅适用于参考SPS的CLVS中的帧间CTU。
- [0217] iv. 在一个示例中,第0色度QP表组仅适用于参考SPS的CLVS中的帧内CU,并且第1色度QP表组仅适用于参考SPS的CLVS中的帧间CU。
- [0218] v. 在一个示例中,第0色度QP表组仅适用于参考SPS的CLVS中的帧内编解码块,并且第1色度QP表组仅适用于参考SPS的CLVS中的帧间编解码块。
- [0219] e. 仅当`ChromaArrayType`不等于0时,才可以发送该标志。
- [0220] f. 在实施例4中示出了示例性实施方式。
- [0221] g. 在实施例5中示出了另一示例性实施方式。
- [0222] 8. 色度QP表可以根据预测模式来确定。
- [0223] a. 在一个示例中,帧内CU和其他CU可以具有不同的色度QP表组。
- [0224] i. 在一个示例中,它可以仅适用于双树和/或本地双树。
- [0225] b. 可替代地,帧内/调色板CU和其他CU可以具有不同的色度QP表组。
- [0226] c. 可替代地,帧内/IBC/调色板CU和其他CU可以具有不同的色度QP表组。
- [0227] d. 可替代地,帧内/IBC CU和其他CU可以具有不同的色度QP表组。
- [0228] e. 在以上示例中,要被信令通知的色度QP表的数量可以依赖于分类的预测模式组的数量。
- [0229] f. 在以上示例中,预测模式可以意味着亮度CB的预测模式。
- [0230] 9. 变换跳过块可以具有不同的色度QP表组。
- [0231] a. 在一个示例中,它可以仅适用于在变换跳过模式下编解码的亮度块。
- [0232] 10. 色度QP表起始点语法元素的最大允许值(即,当前在JVET-Q2001-vC中表示为`qp_table_start_minus26`)可以为37。
- [0233] 11. 色度QP表起始点语法元素(即,当前在JVET-Q2001-vC中表示为`qp_table_start_minus26`)可以偏移小于26的数K。
- [0234] a. 在一个示例中,语法元素可以用`qp_table_start_minusK`替换,和/或语法元素的值加K应当在 $-(K+QpBdOffset)$ 到 $(M-K)$ 的范围(包括端值)内,其中K小于26。

- [0235] b. 在一个示例中,语法元素可以用`qp_table_start_minusK`替换,和/或语法元素的值应当在 $-(K+QpBdOffset)$ 到 $(M-1-K)$ 的范围(包括端值)内,其中 K 小于26。
- [0236] i. 在一个示例中, K 等于20。此外,可替代地,语法元素可以被`init_qp_minus20`替换和/或语法元素的值加20应当在 $-(20+QpBdOffset)$ 到 $(M-20)$ 的范围(包括端值)内。
- [0237] ii. 在一个示例中, K 等于20。此外,可替代地,语法元素可以被`init_qp_minus20`替换和/或语法元素的值加20应当在 $-(20+QpBdOffset)$ 到 $(M-1-20)$ 的范围(包括端值)内。
- [0238] c. 可替代地,初始QP值语法元素可以偏移大于26的数量 K 。
- [0239] i. 在一个示例中,语法元素可以被`init_qp_minusK`替换,并且语法元素的值加 K 应当在 $-(K+QpBdOffset)$ 到 $(M-K)$ 的范围
- [0240] (包括端值)内,其中 K 是大于26的常数。
- [0241] ii. 在一个示例中,语法元素可以被`init_qp_minusK`替换,并且语法元素的值加 K 应当在 $-(K+QpBdOffset)$ 到 $(M-1-K)$ 的范围(包括端值)内,其中 K 是大于26的常数。
- [0242] 1) 在一个示例中, K 等于32。此外,可替代地,语法元素可以被`init_qp_minus32`替换,并且语法元素的值加32应当在 $-(32+QpBdOffset)$ 到 $(M-32)$ 的范围(包括端值)内。
- [0243] 2) 在一个示例中, K 等于32。此外,可替代地,语法元素可以被`init_qp_minus32`替换,并且语法元素的值加32应当在 $-(32+QpBdOffset)$ 到 $(M-1-32)$ 的范围(包括端值)内。
- [0244] d. 在以上示例中, K 可以依赖于图片/条带类型;和/或预测模式和/或比特深度。
- [0245] i. 在一个示例中,对于帧内图片/条带, K 被设置为1。
- [0246] ii. 在一个示例中,对于P/B图片/条带, K 被设置为32。
- [0247] e. 在以上示例中, M 可以是最大允许QP值,例如63。
- [0248] f. 在一个示例中, K 为0。
- [0249] i. 此外,可替代地,语法元素可以用`uv(e)`而不是`sv(e)`进行二进位化。
- [0250] 12. 色度QP表起始点语法元素(即,当前在JVET-Q2001-vC中表示为`qp_table_start_minus26`)可以偏移依赖于当前图片是否是仅帧内图片的值。
- [0251] a. 可替代地,偏移可以依赖于`intra_only_constraint_flag`。
- [0252] 13. 与色度QP表中的枢轴点的数量有关的语法元素(即,当前表示为`num_points_in_qp_table_minus1`)应该能够呈现0个数量的点。
- [0253] a. 在一个示例中,语法元素`num_points_in_qp_table_minus1`可以被用于指定色度QP表中的点的数量的`num_points_in_qp_table`替换,并且该值是非负整数。
- [0254] i. 在一个示例中,表示枢轴点的数量的语法元素的值应当在0到 $(63+QpBdOffset)$ 的范围内。
- [0255] b. 此外,可替代地,当色度QP表中的枢轴点的数量为零时,色度QP表的第 i 个条目被设置为等于亮度QP表的第 i 个条目。
- [0256] c. 此外,可替代地,当色度QP表中的枢轴点的数量为零时,色度QP表的第 i 个条目被设置为等于亮度QP表的第 i 个条目加偏移。
- [0257] i. 在一个示例中,偏移可以依赖于编解码方法(例如,JCCR开启或关闭)
- [0258] 14. 色度QP表起始点语法元素(即,当前在JVET-Q2001-vC中表示为`qp_table_start_minus26`)的解析可以以枢轴点的数量是否为0为条件。
- [0259] a. 在一个示例中,当枢轴点的数量为0时,可以跳过语法元素的解析。

- [0260] 15. 在色度QP表的推导过程中,应该在(delta_qp_in_val_minus1[i][j]+1)和delta_qp_diff_val[i][j]之间执行XOR(异或)运算。
- [0261] a. 在实施例3中示出了示例。
- [0262] 16. QP裁剪可以总是被应用于色度QP表索引。
- [0263] a. 在一个示例中,裁剪范围从-QpBdOffset到63(包括端值)。
- [0264] 17. QP裁剪可以总是被应用于色度QP表中的映射的色度QP。
- [0265] a. 在一个示例中,裁剪范围从-QpBdOffset到63(包括端值)。
- [0266] 18. 色度QP表可以在SPS和PPS两者中被信令通知。
- [0267] a. 在一个示例中,PPS中的色度QP表可以覆写SPS中的对应表。
- [0268] 19. 色度QP表可以在PH或SH中被信令通知。
- [0269] a. 在一个示例中,PH或SH中的色度QP表可以覆写SPS或PPS中的对应表。
- [0270] 20. 默认的色度QP表可能取决于ChromaArrayType。
- [0271] a. 在一个示例中,当ChromaArrayType等于3时(即4:4:4颜色格式并且separate_colour_plane_flag等于0),可以将标识QP色度QP表(即亮度QP=色度QP)设置为默认。
- [0272] b. 在一个示例中,当ChromaArrayType等于2(即4:2:2颜色格式)时,可以将标识QP色度QP表(即亮度QP=色度QP)设置为默认。
- [0273] 21. 关于用于解决第八个问题的RRC和TSRC的控制,公开了以下方法中的一个或多个,例如,如在第七实施例组中:
- [0274] a. 在一个示例中,是否允许对视频单元(CLVS/图片组/图片/条带/片/CTU行/CTU/CU/PU/TU)的TSRC可以依赖于在SPS/PPS/PH/SH/块/CU/TU级别中信令通知的语法元素(例如,一个或多个标志)。
- [0275] i. 例如,TSRC启用/禁用标志可以在PH或SH中被信令通知,但不在两者中被信令通知,
- [0276] 1) 可附加地,是否在PH或SH中信令通知TSRC启用/禁用标志可以依赖于在PPS/SPS中信令通知的语法标志。
- [0277] 2) 可附加地,当在PH中信令通知TSRC启用/禁用标志时,SH中的TSRC启用/禁用标志不被信令通知。
- [0278] 3) 可附加地,当不存在SH中的TSRC启用/禁用标志时,其被推断为等于PH中的TSRC启用/禁用标志
- [0279] ii. 可替代地,TSRC启用/禁用标志可以在PH和SH两者中被信令通知,
- [0280] 1) 此外,可附加地,当PH中的TSRC启用/禁用标志指定对参考PH的所有条带禁用TSRC时,SH中的TSRC启用/禁用标志可以不被信令通知
- [0281] iii. 例如,用上下文自适应算术熵编解码的语法元素(例如ae(v)编解码)编解码块/CU/TU级别TSRC启用/禁用标志。
- [0282] iv. 例如,用使用n比特的无符号整数(例如,u(1)编解码)编解码SPS/PPS/PH/SH级别TSRC启用/禁用标志,例如n=1。
- [0283] b. 可附加地,视频单元的TSRC启用/禁用标志的信令通知可以依赖于是否对视频单元启用/使用变换跳过。
- [0284] i. 例如,如果在SPS级别禁用变换跳过(例如,sps_transform_skip_enabled_flag

等于0),则要求PPS级别中的TSRC启用/禁用标志等于指定对参考当前PPS的图片禁用TSRC的特定值。

[0285] ii.可附加地,如果在更高级别禁用变换跳过(例如,sps_transform_skip_enabled_flag等于0),则更低级别和/或相同层(例如,SPS/PH/SH/块/CU/TU级别)的TSRC启用/禁用标志不被信令通知。

[0286] iii.此外,可替代地,如果在更高级别(例如,SPS)启用变换跳过,但不用于视频单元(例如,TU级别transform_skip_flag等于0),则当前视频单元(例如,TU)的TSRC启用/禁用标志不被信令通知。

[0287] iv.可附加地,当不在视频单元级别信令通知TSRC启用/禁用标志时,TSRC启用/禁用标志的值被推断为指定对视频单元禁用TSRC的特定值。

[0288] c.此外,可替换地,低级别的TSRC启用/禁用标志的信令通知可以依赖于更高级别的TSRC启用/禁用标志。

[0289] i.例如,图片/条带级别的TSRC启用/禁用标志的信令通知可以依赖于是否在SPS/PPS级别启用TSRC。

[0290] ii.可附加地,块/CU/TU级别的TSRC启用/禁用标志的信令通知可以依赖于是否在SPS/PPS/图片/条带级别启用TSRC。

[0291] 1)例如,当在更高级别(例如,SPS/PPS/图片/条带级别)禁用TSRC时,块/CU/TU级别的TSRC启用/禁用标志不被信令通知。

[0292] 2)此外,可附加地,当不存在TSRC启用/禁用标志时,其被推断为特定值(诸如指定对当前视频单元禁用TSRC的值)。

[0293] iii.可附加地,PPS级别的TSRC启用/禁用标志的值可以依赖于是否在SPS级别启用TSRC。

[0294] 1)例如,当SPS级别TSRC启用/禁用标志指定对CLVS禁用TSRC时,要求PPS级别TSRC启用/禁用标志的值等于指定对参考当前PPS的图片禁用TSRC的特定值。

[0295] 关于ALF和CC-ALF

[0296] 22.建议当对当前CTU禁用亮度ALF时,对当前CTU使用色度ALF/CC-ALF的指示不被信令通知。

[0297] a.此外,可替代地,当对CTU禁用亮度ALF时,该使用被推断为假。换句话说,当在CTU中禁用亮度ALF时,色度ALF/CC-ALF被禁用。

[0298] 关于子比特流提取

[0299] 23.建议在数组NumSubLayersInLayerInOLS[][]中携带的信息可以在VVC的子比特流提取过程中仅用于移除VCL NAL单元。

[0300] a.在一个示例中,在VVC的条款C.6中指定的子比特流提取过程中的以下步骤:

[0301] -从outBitstream中移除以下所有条件为真的所有NAL单元:

[0302] -nal_unit_type不等于IDR_W_RADL、IDR_N_LP或CRA_NUT。

[0303] -nuh_layer_id等于LayerIdInOls[targetOlsIdx][j],j的值在0到NumLayersInOls[targetOlsIdx]-1的范围内。

[0304] -TemporalId大于或等于NumSubLayersInLayerInOLS[targetOlsIdx][j]。

[0305] 改为如下:

- [0306] -从outBitstream中移除以下所有条件为真的所有VCL NAL单元:
- [0307] -nal_unit_type不等于IDR_W_RADL、IDR_N_LP或CRA_NUT。
- [0308] -nuh_layer_id等于LayerIdInOls[targetOlsIdx][j], j的值在0到NumLayersInOls[targetOlsIdx]-1的范围内。
- [0309] -TemporalId大于或等于NumSubLayersInLayerInOLS[targetOlsIdx][j]。
- [0310] 关于APS的数量
- [0311] 在下面的讨论中,“滤波器的最大允许数量”可以指“对于一个序列/图片/条带,可以由一个或多个APS信令通知的滤波器的最大允许数量”。
- [0312] “APS NAL单元的最大允许数量”可以指“可以为序列/图片/条带信令通知的APS NAL单元的最大允许数量”。
- [0313] “滤波器系数的最大允许数量”可以指“对于一个序列/图片/条带,可以由一个或多个APS信令通知的滤波器系数的最大允许数量”。
- [0314] 24.关于解决第十二个问题的APS NAL单元数量:
- [0315] a.下面的APS NAL单元的种类可以是ALF APS、LMCS APS、SCALING APS中的一种或多种。
- [0316] b.在一个示例中,包含与不同颜色分量相关的信息的APS可以被单独信令通知。
- [0317] i.例如,APS NAL单元可能仅包含与亮度相关的语法元素(例如,亮度滤波器)。
- [0318] ii.此外,APS NAL单元可以仅包含与色度(例如Cb和/或Cr)相关的语法元素(例如,色度滤波器)。
- [0319] 1)例如,一个APS NAL单元可能仅包含色度ALF语法元素。
- [0320] 2)例如,一个APS NAL单元可能仅包含CCALF语法元素。
- [0321] 3)或者,一个APS NAL单元可能仅包含CCALF Cb语法元素。
- [0322] 4)或者,一个APS NAL单元可能仅包含CCALF Cr语法元素。
- [0323] iii.例如,ALF(和/或LMCS和/或缩放列表)的亮度和色度滤波器可以具有不同的APS类型(例如,aps_params_type)。
- [0324] iv.例如,在APS中信令通知一个或多个SE,以指示APS中包含(哪些)颜色分量。
- [0325] c.在一个示例中,可以针对序列/图片/条带信令通知的APS NAL单元的最大允许数量可以取决于预定义的值(例如预定义的存储器尺寸或预定义的滤波器数量)。
- [0326] d.在一个示例中,要求在APS中为序列/图片/条带信令通知的滤波器数量不得大于在APS中信令通知的滤波器的最大允许数量。
- [0327] e.在一个示例中,要求在APS中为序列/图片/条带信令通知的滤波器系数的数量不得大于在APS中信令通知的滤波器系数的最大允许数量。
- [0328] i.在APS中信令通知的滤波器系数的最大允许数量可以是预定义的数量。
- [0329] ii.对于不同的APS类型,在APS中信令通知的滤波器系数的最大允许数量可能不同。
- [0330] iii.在APS中信令通知的滤波器系数的最大允许数量可以从编解码器信令通知到解码器,例如在VPS/SPS/PPS/PH/SH/序列标头中。
- [0331] f.在一个示例中,要求在APS中为序列/图片/条带信令通知的滤波器数量不得大于在APS中信令通知的滤波器的最大允许数量。

- [0332] g. 在一个示例中,针对不同颜色分量在APS中信令通知的滤波器的最大允许数量可能不同。
- [0333] i. 在一个示例中,在APS中信令通知的亮度滤波器的最大允许数量和APS中信令通知的色度滤波器的最大数量可能不同。
- [0334] ii. 例如,在特定APS类型(例如,ALF/LMCS/SCALING)的APS中信令通知的亮度滤波器的最大允许数量等于第一特定值。
- [0335] iii. 可附加地,特定APS类型(例如,ALF/LMCS/SCALING)的APS中信令通知的色度(例如Cb和/或Cr)滤波器的最大允许数量等于第二特定值。
- [0336] iv. 例如,在所有APS类型(例如,ALF/LMCS/SCALING)的APS中信令通知的亮度滤波器的最大允许数量等于第一特定值。
- [0337] v. 可附加地,所有APS类型(例如,ALF/LMCS/SCALING)的APS中信令通知的色度(例如Cb和/或Cr)滤波器的最大允许数量等于第二特定值。
- [0338] h. 在一个示例中,ALF亮度滤波器的最大允许数量、ALF色度滤波器的最大允许数量和CCALF滤波器的最大允许数量可以不同。
- [0339] i. 例如,ALF亮度滤波器的最大允许数量等于 $X1$ (例如, $X1 = 25 \times 8 = 200$)。
- [0340] ii. 例如,对于ALF APS单元,ALF色度滤波器的最大允许数量等于 $X2$ (例如, $X2 = 8 \times 8 = 64$)。
- [0341] iii. 例如,对于ALF APS单元,CCALF滤波器的最大允许数量等于 $X3$ (例如, $X3 = 4 \times 8 \times 2 = 64$)。
- [0342] 1) 或者,CCALF Cb滤波器的最大允许数量为 $Y1$ (例如, $Y1 = 4 \times 8 = 32$)。
- [0343] 2) 此外,CCALF Cr滤波器的最大允许数量为 $Y2$ (例如, $Y2 = 4 \times 8 = 32$)。
- [0344] iv. 在一个示例中, $X1$ 、 $X2$ 、 $X3$ 、 $Y1$ 、 $Y2$ 是预定义值。
- [0345] i. 在一个示例中,APS NAL单元内的每个滤波器的最大允许数量等于 K_i ,其中 K 表示某个值,并且 i 表示滤波器类型(例如,亮度滤波器、色度滤波器、ccalf滤波器等)。
- [0346] i. 例如,ALF亮度滤波器的 K_i 等于 $25 \times 8 = 200$ 。
- [0347] ii. 例如,ALF色度滤波器的 K_i 等于 $8 \times 8 = 64$ 。
- [0348] iii. 例如,CCALF滤波器的 K_i 等于 $4 \times 8 \times 2 = 64$ 。
- [0349] 1) 或者,CCALF Cb滤波器的 K_i 等于 $4 \times 8 = 32$ 。
- [0350] 2) 或者,CCALF Cr滤波器的 K_i 等于 $4 \times 8 = 32$ 。
- [0351] j. 在一个示例中,最大允许数量可以从编解码器信令通知到解码器,例如在VPS/SPS/PPS/PH/SH/序列标头中。
- [0352] i. 例如,本权利要求中的“最大允许数量”可以指APS NAL单元的最大允许数量,或者滤波器的最大允许数量,或者滤波器系数的最大允许数量。
- [0353] ii. 例如,可以信令通知特定类型的APS NAL单元(例如,aps_params_type等于ALF APS)的最大允许数量。
- [0354] 1) 例如,最大允许数量可以在SPS级别被信令通知。
- [0355] 2) 例如,最大允许数量可以在PH/SH级别被信令通知。
- [0356] iii. 例如,最大允许数量的信令通知可能取决于编解码工具(ALF/CCALF/LMCS/缩放列表)是否启用。

- [0357] 1) 在一个示例中,当禁用ALF/CCALF时,可能不会信令通知ALF/CCALF APS的最大允许数量。
- [0358] a. 另外,该数字在不存在时被推断为等于某个值(例如0)。
- [0359] 2) 在一个示例中,当CCALF被禁用时,对于具有等于CCALF APS ID(例如,ph_cc_alf_cb_aps_id、ph_cc_alf_cr_aps_id、slice_cc_alf_cb_aps_id、slice_cc_alf_cr_aps_id)的APS ID(例如adaptation_parameter_set_id)的ALF APS的最大允许数量可能不会被信令通知。
- [0360] a. 另外,该数字在不存在时被推断为等于某个值(例如0)。
- [0361] 3) 如何信令通知较小视频单元(例如,图片、条带)的最大允许数量可以取决于更高级别(例如,SPS)的最大允许数量。
- [0362] a. 例如,ALF亮度APS的最大允许数量取决于ALF APS的最大允许数量。
- [0363] iv. 例如,可以信令通知从最大允许数(表示为N)(例如N-M或M-N,其中M是小于N的某个值)推导的变量。
- [0364] k. 在一个示例中,最大允许数量取决于色度格式。
- [0365] i. 例如,亮度/色度的最大允许数可能取决于ChromaArrayType是否等于0(例如,4:0:0和4:0:0具有单独的颜色平面编解码)。
- [0366] ii. 例如,ALF/SCALING/LMCS APS的最大允许数量取决于ChromaArrayType是否等于0。
- [0367] 1. 在一个示例中,上述的最大允许数量可能取决于是否启用了编解码工具。
- [0368] i. 例如,以上的最大允许数量取决于是否启用ALF,和/或是否在SPS/PPS/PH/SH级别启用CCALF。
- [0369] ii. 例如,ALF APS的最大允许数量取决于是否启用ALF,和/或是否在SPS/PPS/PH/SH级别启用CCALF。
- [0370] iii. 例如,LMCS APS的最大允许数量取决于是否启用了LMCS,和/或是否在SPS/PPS/PH/SH级别启用了色度残差缩放(CRS)。
- [0371] iv. 例如,SCALING APS的最大允许数量取决于是否在SPS/PPS/PH/SH级别启用显式缩放列表。
- [0372] m. 在一个示例中,如何对APS ID进行编解码可能取决于受限号码。
- [0373] i. 在一个示例中,APS ID可以被ue(v)编解码,在0到K(例如,127或255)的范围(包括端值)内。
- [0374] ii. 或者,APS ID可以是u(X)编解码的,例如X=7或8。
- [0375] 1) X可以在APS ID被编解码之前信令通知。
- [0376] 2) X可以在APS ID被编解码之前推导。
- [0377] 3) X可以基于最大允许数量推导。
- [0378] iii. 或者,信令通知APS ID的最大数量。
- [0379] 1) 例如,APS ID的最大数量可以是u(X)编解码的,例如X=7或8。
- [0380] 2) 此外,APS ID可以是u(v)编解码的。
- [0381] iv. 例如,上面提到的APS ID可能是以下一项或多项:
- [0382] 1) APS语法结构中的adaptation_parameter_set_id。

[0383] 2) PH语法结构中的`ph_alf_aps_id_luma[i]`、`ph_alf_aps_id_chroma`、`ph_cc_alf_cb_aps_id`、`ph_cc_alf_cr_aps_id`。

[0384] 3) SH语法结构中的`slice_alf_aps_id_luma[i]`、`slice_alf_aps_id_chroma`、`slice_cc_alf_cb_aps_id`、`slice_cc_alf_cr_aps_id`。

[0385] 4) PH语法结构中的`ph_lmcs_aps_id`。

[0386] 5) PH语法结构中的`ph_scaling_list_aps_id`。

[0387] n. 在一个示例中,上述约束可以由比特流一致性、比特流约束或语法表中的显式信令通知来表示。

[0388] o. 在一个示例中,上述限度/限制/约束 (limitations/restrictions/constraints) 可能取决于档次/级别/一般约束标志等。

[0389] 25. 建议指定QP表中的点的数量的语法元素的信令通知和/或范围和/或推断取决于其他语法元素。

[0390] a. 建议将`num_points_in_qp_table_minus1[i]`的最大值设置为(QP最大值-用于描述第*i*个色度QP映射表的起始亮度和色度QP)。

[0391] i. 在一个示例中,最大值设置为 $(63 - (\text{qp_table_start_minus26}[i] + 26))$ 。

[0392] ii. 在一个示例中,最大值设置为 $(64 - (\text{qp_table_start_minus26}[i] + 26))$ 。

[0393] 26. 当仿射被启用并且SbTMVP被禁用时,建议不允许子块merge候选的最大数量为0 (MaxNumSubblockMergeCand)。

[0394] a. 在一个示例中,当仿射被启用并且SbTMVP被禁用时,子块merge候选的最大数量被限制为大于0。

[0395] b. 在一个示例中,`sps_five_minus_max_num_subblock_merge_cand`的范围从[0, 4]开始,而与SPS SbTMVP启用标志(例如,`sps_sbtmvp_enabled_flag`)的值无关。

[0396] c. 当启用仿射时,一致性比特流应满足MaxNumSubblockMergeCand的值应在1到5的范围(包括端值)内。

[0397] i. 或者,此外,当仿射被禁用时,MaxNumSubblockMergeCand的值应在0到1的范围(包括端值)内。

[0398] d. 或者,此外,是否信令通知`merge_subblock_flag`取决于是否启用仿射,而不是检查子块merge候选的最大数量。

[0399] 27. 建议添加第一语法元素以指示是否允许仿射merge模式。

[0400] a. 或者,此外,可以有条件地信令通知第一语法元素,例如,根据SPS仿射标志被启用。

[0401] b. 或者,此外,当第一语法元素指示仿射merge模式被启用时,可以信令通知对子块merge候选的最大数量的指示。

[0402] 28. 为了解决问题13,子图片布局被信令通知,如下所示:

[0403] a. 添加一个标志,例如,名为`sps_subpic_same_res_flag`,以指定是否所有参考SPS的子图片分别具有相同的宽度和高度。

[0404] b. 当`sps_subpic_same_res_flag`等于1时,所有子图片的共同宽度和高度例如分别通过两个新语法元素`sps_uniform_subpic_width_minus1`和`sps_uniform_subpic_height_minus1`以CTU为单位信令通知。

[0405] i. 此外, 在一个示例中, 表示为numSubpicCols的子图片列的数量被推导为等于 $(\text{sps_pic_width_max_in_luma_samples} / ((\text{sps_uniform_subpic_width_minus1} + 1) * \text{CtbSizeY}))$, 表示为numSubpicRows的子图片行的数量被推导为等于 $(\text{sps_pic_height_max_in_luma_samples} / ((\text{sps_uniform_subpic_height_minus1} + 1) * \text{CtbSizeY}))$, 并且表示为sumSubpics的子图片的数量被推导出等于numSubpicCols*numSubpicRows。

[0406] ii. 此外, 在一个示例中, sps_num_subpics_minus1被跳过, 并且该值被推断为等于numSubpics-1。

[0407] iii. 在一个示例中, 对于0到sps_num_subpics_minus1范围(包括端值)内的每个i值, sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]的值被分别推断为等于 $(\text{sps_uniform_subpic_width} + 1) * (i \% \text{numSubpicCols})$ 、 $(\text{sps_uniform_subpic_height_minus1} + 1) * (i / \text{numSubpicCols})$ 、sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1。

[0408] iv. 在一个示例中, 对于在0到sps_num_subpics_minus1范围(包括端值)内的每个i值, 当sps_independent_subpics_flag等于0时, sps_subpic_treated_as_pic_flag[i]和sps_loop_filter_across_subpic_enabled_flag[i]被信令通知。

[0409] c. 当sps_subpic_same_res_flag等于0时, 适用JVET-R2001-vA/v10中的当前子图片布局信令通知。

[0410] d. 在一个示例中, 当sps_pic_width_max_in_luma_samples小于或等于CtbSizeY时, 跳过sps_uniform_subpic_width_minus1并推断为等于0。

[0411] e. 在一个示例中, 当sps_pic_height_max_in_luma_samples小于或等于CtbSizeY时, 跳过sps_uniform_subpic_height_minus1并推断为等于0。

[0412] f. 在一个示例中, sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1是ue(v)编解码的, 并且sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]也是ue(v)编解码的。

[0413] g. 在一个示例中, sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1是u(v)编解码的, 其长度分别与JVET-R2001-vA/v10中的sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]相同。

[0414] h. 在一个示例中, SPS语法更改如下, 其中粗斜体带下划线的文本表示新添加的文本, 而左右双括号(例如[[]])表示双括号之间的已删除文本:

[0415]

seq_parameter_set_rbsp() {	描述符
...	
<u>sps_subpic_same_res_flag</u>	<u>ue(v)</u>
<u>if(sps_subpic_same_res_flag) {</u>	
<u>if(sps_pic_width_max_in_luma_samples > CtbSizeY)</u>	
<u>sps_uniform_subpic_width_minus1</u>	<u>u(v)</u>
<u>if(sps_pic_height_max_in_luma_samples > CtbSizeY)</u>	
<u>sps_uniform_subpic_height_minus1</u>	<u>u(v)</u>
<u>} else</u>	
sps_num_subpics_minus1	uc(v)
if(sps_num_subpics_minus1 > 0)	
sps_independent_subpics_flag	u(1)
for(i = 0; <u>sps_subpic_same_res_flag &&</u> sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++) {	
if(i > 0 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
sps_subpic_ctu_top_left_x[i]	u(v)
if(i > 0 && sps_pic_height_max_in_luma_samples > CtbSizeY) {	
sps_subpic_ctu_top_left_y[i]	u(v)

	if(i < sps_num_subpics_minus1 && sps_pic_width_max_in_luma_samples > CtbSizeY)	
	sps_subpic_width_minus1[i]	u(v)
	if(i < sps_num_subpics_minus1 && sps_pic_height_max_in_luma_samples > CtbSizeY)	
	sps_subpic_height_minus1[i]	u(v)
	}	
[0416]	<u>for(i = 0; sps_num_subpics_minus1 > 0 && i <=</u> <u>sps_num_subpics_minus1; i++)</u>	
	if(!sps_independent_subpics_flag) {	
	sps_subpic_treated_as_pic_flag[i]	u(1)
	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	[[]]	
	...	

[0417] i. 在一个示例中,SPS语法更改如下:

	seq_parameter_set_rbsp() {	描述符
	...	
	<u>sps_subpic_same_res_flag</u>	<u>ue(v)</u>
	<u>if(sps_subpic_same_res_flag) {</u>	
	<u>sps_uniform_subpic_width_minus1</u>	<u>ue(v)</u>
[0418]	<u>sps_uniform_subpic_height_minus1</u>	<u>ue(v)</u>
	<u>} else</u>	
	sps_num_subpics_minus1	ue(v)
	if(sps_num_subpics_minus1 > 0)	
	sps_independent_subpics_flag	u(1)

	for(i = 0; <u>sps_subpic_same_res_flag</u> && sps_num_subpics_minus1 > 0 &&	
	i <= sps_num_subpics_minus1; i++) {	
	if(i > 0 [[&& sps_pic_width_max_in_luma_samples > CtbSizeY]])	
	sps_subpic_ctu_top_left_x[i]	ue(v)
	if(i > 0 [[&& sps_pic_height_max_in_luma_samples > CtbSizeY]]) {	
	sps_subpic_ctu_top_left_y[i]	ue(v)
	if(i < sps_num_subpics_minus1 [[&& sps_pic_width_max_in_luma_samples > CtbSizeY]])	
	sps_subpic_width_minus1[i]	ue(v)
[0419]	if(i < sps_num_subpics_minus1 [[&& sps_pic_height_max_in_luma_samples > CtbSizeY]])	
	sps_subpic_height_minus1[i]	ue(v)
	}	
	<u>for(i = 0; sps_num_subpics_minus1 > 0 && i <= sps_num_subpics_minus1; i++)</u>	
	if(!sps_independent_subpics_flag) {	
	sps_subpic_treated_as_pic_flag[i]	u(1)
	sps_loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	[[]]	
	...	

[0420] 29. 建议对JVET-R2001-vA/v10中的最新VVC规范文本进行以下更改：

[0421] SPS语义：

[0422] ...

[0423] sps_subpic_treated_as_pic_flag[i]等于1指定CLVS中每个编解码图片的第i个子图片在解码过程中被视为排除环路滤波操作的图片。sps_subpic_treated_as_pic_flag[i]等于0指定CLVS中每个编解码图片的第i个子图片在解码过程中不被视为排除环路滤波操作的图片。当不存在时，sps_subpic_treated_as_pic_flag[i]的值被推断为等于1。

[0424] [[当sps_num_subpics_minus1大于0且sps_subpic_treated_as_pic_flag[i]等于1时，对于参考SPS的当前层的每个CLVS，假设targetAuSet是从包含解码顺序中CLVS的第一个图片的AU开始到包含解码顺序中CLVS的最后一个图片的AU的所有AU，包括端值，比特流一致性的要求是，对于由当前层和将当前层作为参考层的所有层组成的targetLayerSet，以下所有条件都为真：]]

[0425] [[-对于targetAuSet中的每个AU, targetLayerSet中层的所有图片都应具有相同的pps_pic_width_in_luma_samples值和相同的pps_pic_height_in_luma_samples值。]]

[0426] [[-targetLayerSet中层参考的所有SPS应具有相同的sps_num_subpics_minus1值, 并且对于在0到sps_num_subpics_minus1的范围(包括端值)内的每个j值, 应分别具有相同的sps_subpic_ctu_top_left_x[j]、sps_subpic_ctu_top_left_y[j]、sps_subpic_width_minus1[j]、sps_subpic_height_minus1[j]和j]、sps_subpic_treated_as_pic_flag[j]]]

[0427] [[-对于targetAuSet中的每个AU, 对于每个j值在0到sps_num_subpics_minus1的范围(包括端值)内, targetLayerSet中层的所有图片都应具有相同的SubpicIdVal[j]值]]

[0428] ...

[0429] 参考图片列表构建的解码过程

[0430] ...

[0431] -[[当vps_independent_layer_flag[GeneralLayerIdx[nuh_layer_id]]等于0且sps_num_subpics_minus1大于0时, 以下两个条件中的任何一个(但不是两者)应为真:]]

当sps num subpics minus1大于0并且具有子图片索引

subPicIdx 的当前子图片具有等于1的

sps subpic treated as pic flag[subPicIdx]时, 以下两个条件中的任一

个(但不是两个)应该为真:

[0432] o[[RefPicList[0]或RefPicList[1]中每个活动条目所参考的图片与当前图片具有相同的子图片布局, 即该图片和当前图片所参考的SPS具有相同的sps_num_subpics_minus1的值以及对于在0到sps_num_subpics_minus1的范围(包括端值)内的每个j值, 分别具有相同的sps_subpic_ctu_top_left_x[j]、sps_subpic_ctu_top_left_y[j]、sps_subpic_width_minus1[j]和sps_subpic_height_minus1[j]的值。]]

o **RefPicList[0]或RefPicList[1]中的每个活动条目所参考的图片和当前图片对于以下中的每一项具有相同的值:**

- **pps pic width in luma samples**
- **pps pic height in luma samples**

[0433]

- **sps num subpics minus1**
- **sps subpic ctu top left x[i]、sps subpic ctu top left y[i]、sps subpic width minus1[i]、sps subpic height minus1[i]、sps subpic treated as pic flag[i] , 分别对于在0到sps num subpics minus1的范围(包括端值)内的每个i值**

[0434] oRefPicList[0]或RefPicList[1]中的每个活动条目所参考的图片是一个ILRP, 其sps_num_subpics_minus1的值等于0。

[0435] 30. 建议对上一项目进行以下更改:

[0436] PPS语义:

[0437] ...

[0438] pps_subpic_id[i]指定第i个子图片的子图片ID。pps_subpic_id[i]语法元素的长度是pps_subpic_id_len_minus1+1比特。

[0439] 对于0到sps_num_subpics_minus1范围内的每个i值,变量SubpicIdVal[i]的推导如下:

```

for(i=0; i <= sps_num_subpics_minus1; i++)
    if( sps_subpic_id_mapping_explicitly_signalled_flag )
        SubpicIdVal[ i ] = pps_subpic_id_mapping_present_flag ? pps_subpic_id[ i ] :
[0440] sps_subpic_id[ i ]
    else
        SubpicIdVal[ i ] = i

```

[0441] 以下两个约束都适用是比特流一致性的要求:

[0442] -对于0到sps_num_subpics_minus1范围(包括端值)内的任何两个不同的i和j值,SubpicIdVal[i]不应等于SubpicIdVal[j]。

[0443] -[[对于0到sps_num_subpics_minus1(包括端值)范围内的每个i值,当nuh_layer_id等于特定值layerId的当前图片的SubpicIdVal[i]的值不等于、nuh_layer_id等于layerId的参考图片的SubpicIdVal[i]的值时,当前图片的第i个子图片中的编解码条带的RPL的活动条目将不包括该参考图片。]]

[0444] ...

[0445] 参考图片列表构建的解码过程

[0446] ...

[0447] -当sps_num_subpics_minus1大于0且具有子图片索引subPicIdx的当前子图的sps_subpic_treated_as_pic_flag[subPicIdx]等于1时,以下两个条件之一(但不能同时)应为真:

[0448] oRefPicList[0]或RefPicList[1]中的每个活动条目所参考的图片和当前图片对于以下中的每一项具有相同的值:

[0449] • pps_pic_width_in_luma_samples

[0450] • pps_pic_height_in_luma_samples

[0451] • sps_num_subpics_minus1

[0452] • sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、

[0453] sps_subpic_width_minus1[i]、sps_subpic_height_minus1[i]、sps_subpic_treated_as_pic_flag[i]、**SubpicIdVal[i]**, 分别对于在0到sps_num_subpics_minus1的范围(包括端值)内的每个i值

[0454] oRefPicList[0]或RefPicList[1]中的每个活动条目所参考的图片是一个ILRP,其sps_num_subpics_minus1的值等于0。

[0455] 31.当pps_pic_width_in_luma_samples小于等于CtbSizeY时,要求pps_tile_

column_width_minus1[i]等于PicWidthInCtbsY-1。

[0456] 32.当pps_pic_height_in_luma_samples小于等于CtbSizeY时,要求pps_tile_row_height_minus1[i]等于PicHeight-1。

[0457] 6. 实施例

[0458] 6.1. 实施例1: 根据条带类型的色度QP表

[0459] 以粗体斜体下划线文本标记的改变基于JVET-Q2001-vC。被删除的文本以指示双括号之间的被删除的文本的左右双括号(例如,[[]])标记。

[0460] 7.3.2.3 序列参数集RBSP语法

	seq_parameter_set_rbsp() {	描述符
	
	<i>sps_non_intra_present_flag</i>	<i>u(1)</i>
	if(ChromaArrayType != 0) {	
	<i>sps_joint_cbr_enabled_flag</i>	<i>u(1)</i>
	<i>intra_same_qp_table_for_chroma</i>	<i>u(1)</i>
	numQpTables <i>Intra</i> = <i>intra</i> _same_qp_table_for_chroma ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
	for(i = 0; i < numQpTables <i>Intra</i> ; i++) {	
	<i>intra_qp_table_start_minus26</i> [i]	<i>se(v)</i>
	<i>intra_num_points_in_qp_table_minus1</i> [i]	<i>ue(v)</i>
	for(j = 0; j <= <i>intra_num_points_in_qp_table_minus1</i> [i]; j++) {	
	<i>intra_delta_qp_in_val_minus1</i> [i][j]	<i>ue(v)</i>
	<i>intra_delta_qp_diff_val</i> [i][j]	<i>ue(v)</i>
[0461]	}	
	}	
	<i>if(sps_non_intra_present_flag) {</i>	
	<i>inter_same_qp_table_for_chroma</i>	<i>u(1)</i>
	numQpTables <i>Inter</i> = <i>same_qp_table_for_chroma_inter</i> ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
	for(i = 0; i < numQpTables <i>Inter</i> ; i++) {	
	<i>inter_qp_table_start_minus26</i> [i]	<i>se(v)</i>
	<i>inter_num_points_in_qp_table_minus1</i> [i]	<i>ue(v)</i>
	for(j = 0; j <= <i>inter_num_points_in_qp_table_minus1</i> [i]; j++) {	
	<i>inter_delta_qp_in_val_minus1</i> [i][j]	<i>ue(v)</i>
	<i>inter_delta_qp_diff_val</i> [i][j]	<i>ue(v)</i>
	}	
	}	
[0462]	}	

[0463] 7.3.2.3序列参数集RBSP语法

[0464]

[0465] [[same_qp_table_for_chroma等于1指定仅信令通知一个色度QP映射表,并且当sps_joint_cbr_enabled_flag等于1时,该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差.same_qp_table_for_chroma等于0指定在SPS中信令通知色度QP映射表,当sps_joint_cbr_enabled_flag等于1时,对于Cb和Cr信令通知两个色度QP映射表,并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当same_qp_table_for_chroma不存在于比特流中时,same_qp_table_for_chroma的值被推断为等于1。]]

[0466] [[qp_table_start_minus26[i]加26指定用于描述第i个色度QP映射表的起始亮度和色度QP.qp_table_start_minus26[i]的值应当在-26-QpBdOffset到36的范围(包括端值)内。当qp_table_start_minus26[i]不存在于比特流中时,qp_table_start_minus26[i]的值被推断为等于0。]]

[0467] [[num_points_in_qp_table_minus1[i]加1指定用于描述第i个色度QP映射表的点的数量.num_points_in_qp_table_minus1[i]的值应当在0到63+QpBdOffset的范围(包括端值)内。当num_points_in_qp_table_minus1[0]不存在于比特流中时,num_points_in_qp_table_minus1[0]的值被推断为等于0。]]

[0468] [[delta_qp_in_val_minus1[i][j]指定用于推导第i个色度QP映射表的第j个枢轴点的输入坐标的增量值。当delta_qp_in_val_minus1[0][j]不存在于比特流中时,delta_qp_in_val_minus1[0][j]的值被推断为等于0。]]

[0469] [[delta_qp_diff_val[i][j]指定用于推导第i个色度QP映射表的第j个枢轴点的输出坐标的增量值。]]

[0470] [[第i个色度QP映射表ChromaQpTable[i](其中i=0...numQpTables-1)被推导如下:]]

[0471] sps_non_intra_present_flag等于0指定所有条带都是序列中的帧内条带。sps_non_intra_present_flag等于1指定序列中可能存在非帧内条带。

[0472] intra_same_qp_table_for_chroma等于1对于帧内条带指定仅信令通知一个色度QP映射表,并且当sps_joint_cbr_enabled_flag等于1时,该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差.intra_same_qp_table_for_chroma等于0对于帧内条带指定在SPS中信令通知色度QP映射表,当sps_joint_cbr_enabled_flag等于1时,对于Cb和Cr信令通知两个色度QP映射表,并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当same_qp_table_for_chroma不存在于比特流中时,intra_same_qp_table_for_chroma的值被推断为等于1。

[0473] intra_qp_table_start_minus26[i]加26对于帧内条带指定用于描述第i个色度QP映射表的起始亮度和色度QP.intra_qp_table_start_minus26[i]的值应当在-26-QpBdOffset到36的范围(包括端值)内。当intra_qp_table_start_minus26[i]不存在于比特流中时,intra_qp_table_start_minus26[i]的值被推断为等于0。

[0474] intra_num_points_in_qp_table_minus1[i]加1对于帧内条带指定用于描述第i个色度QP映射表的点的数量.intra_num_points_in_qp_table_minus1[i]的值应当在0到63+QpBdOffset的范围(包括端值)内。当intra_num_points_in_qp_table_minus1[0]不存

在于比特流中时, $\text{intra_num_points_in_qp_table_minus1}[0]$ 的值被推断为等于0。

[0475] $\text{intra_delta_qp_in_val_minus1}[i][j]$ 对于帧内条带指定用于推导第 i 个色度QP映射表的第 j 个枢轴点的输入坐标的增量值。当 $\text{intra_delta_qp_in_val_minus1}[0][j]$ 不存在于比特流中时, $\text{intra_delta_qp_in_val_minus1}[0][j]$ 的值被推断为等于0。

[0476] $\text{intra_delta_qp_diff_val}[i][j]$ 对于帧内条带指定用于推导第 i 个色度QP映射表的第 j 个枢轴点的输出坐标的增量值。

[0477] $\text{inter_same_qp_table_for_chroma}$ 等于1对于B或P条带指定仅信令通知一个色度QP映射表, 并且当 $\text{sps_joint_cbr_enabled_flag}$ 等于1时, 该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差。 $\text{inter_same_qp_table_for_chroma}$ 等于0对于B或P条带指定在SPS中信令通知色度QP映射表, 当 $\text{sps_joint_cbr_enabled_flag}$ 等于1时, 对于Cb和Cr信令通知两个色度QP映射表, 并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当 $\text{same_qp_table_for_chroma}$ 不存在于比特流中时, $\text{inter_same_qp_table_for_chroma}$ 的值被推断为等于1。

[0478] $\text{inter_qp_table_start_minus26}[i]$ 加26对于B或P条带指定用于描述第 i 个色度QP映射表的起始亮度和色度QP。 $\text{intra_qp_table_start_minus26}[i]$ 的值应当在 $-26 - \text{QpBdOffset}$ 到36的范围(包括端值)内。当 $\text{inter_qp_table_start_minus26}[i]$ 不存在于比特流中时, $\text{inter_qp_table_start_minus26}[i]$ 的值被推断为等于0。

[0479] $\text{inter_num_points_in_qp_table_minus1}[i]$ 加1对于B或P条带指定用于描述第 i 个色度QP映射表的点的数量。 $\text{inter_num_points_in_qp_table_minus1}[i]$ 的值应当在0到 $63 + \text{QpBdOffset}$ 的范围(包括端值)内。当 $\text{inter_num_points_in_qp_table_minus1}[0]$ 不存在于比特流中时, $\text{inter_num_points_in_qp_table_minus1}[0]$ 的值被推断为等于0。

[0480] $\text{inter_delta_qp_in_val_minus1}[i][j]$ 对于B或P条带指定用于推导第 i 个色度QP映射表的第 j 个枢轴点的输入坐标的增量值。当 $\text{inter_delta_qp_in_val_minus1}[0][j]$ 不存在于比特流中时, $\text{inter_delta_qp_in_val_minus1}[0][j]$ 的值被推断为等于0。

[0481] $\text{inter_delta_qp_diff_val}[i][j]$ 对于B条带或P条带指定用于推导第 i 个色度QP映射表的第 j 个枢轴点的输出坐标的增量值。

[0482] 第 i 个色度QP映射表 $\text{IntraChromaQpTable}[i]$ (其中 $i = 0 \dots \text{numQpTablesIntra} - 1$) 和第 i 个色度QP映射表 $\text{InterChromaQpTable}[i]$ (其中 $i = 0 \dots \text{numQpTablesInter} - 1$) 被推导如下:

[0483] 当 $\text{intra_same_qp_table_for_chroma}$ 等于1时, $\text{IntraChromaQpTable}[1][k]$ 和 $\text{IntraChromaQpTable}[2][k]$ 被设置为等于 $\text{IntraChromaQpTable}[0][k]$ (其中 k 在 $-\text{QpBdOffset}$ 到63的范围(包括端值)内)。

[0484] 当 $\text{inter_same_qp_table_for_chroma}$ 等于1时, $\text{InterChromaQpTable}[1][k]$ 和 $\text{InterChromaQpTable}[2][k]$ 被设置为等于 $\text{InterChromaQpTable}[0][k]$ (其中 k 在 $-\text{QpBdOffset}$ 到63的范围(包括端值)内)。

[0485] 假设 $\text{ChromaQpTable}[i]$ 对于I条带表示 $\text{IntraChromaQpTable}[i]$ (其中 $i = 0 \dots \text{numQpTablesIntra} - 1$), 并且对于B条带或P条带表示 $\text{InterChromaQpTable}[i]$ (其中 $i = 0 \dots \text{numQpTablesInter} - 1$), 以下适用:

```

qpInVal[ i ][ 0 ] = qp_table_start_minus26[ i ] + 26
qpOutVal[ i ][ 0 ] = qpInVal[ i ][ 0 ]
for( j = 0; j <= num_points_in_qp_table_minus1[ i ]; j++ ) {
[0486]   qpInVal[ i ][ j + 1 ] = qpInVal[ i ][ j ] + delta_qp_in_val_minus1[ i ][ j ] + 1
   qpOutVal[ i ][ j + 1 ] = qpOutVal[ i ][ j ] +
( delta_qp_in_val_minus1[ i ][ j ] ^ delta_qp_diff_val[ i ][ j ] )
}
ChromaQpTable[ i ][ qpInVal[ i ][ 0 ] ] = qpOutVal[ i ][ 0 ]
for( k = qpInVal[ i ][ 0 ] - 1; k >= -QpBdOffset; k -- )
   ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k + 1 ] - 1 ) (62)
for( j = 0; j <= num_points_in_qp_table_minus1[ i ]; j++ ) {
   sh = ( delta_qp_in_val_minus1[ i ][ j ] + 1 ) >> 1
[0487]   for( k = qpInVal[ i ][ j ] + 1, m = 1; k <= qpInVal[ i ][ j + 1 ]; k++, m++ )
       ChromaQpTable[ i ][ k ] = ChromaQpTable[ i ][ qpInVal[ i ][ j ] ] +
( ( qpOutVal[ i ][ j + 1 ] - qpOutVal[ i ][ j ] ) * m + sh ) /
( delta_qp_in_val_minus1[ i ][ j ] + 1 )
}
for( k = qpInVal[ i ][ num_points_in_qp_table_minus1[ i ] + 1 ] + 1; k <= 63; k++ )
   ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k - 1 ] + 1 )

```

[0488] 当same_qp_table_for_chroma等于1时,ChromaQpTable[1][k]和ChromaQpTable[2][k]被设置为等于ChromaQpTable[0][k],其中k在-QpBdOffset到63的范围(包括端值)内。

[0489] 比特流一致性的要求是,qpInVal[i][j]和qpOutVal[i][j]的值应当在-QpBdOffset到63的范围(包括端值)内,其中i在0到numQpTables-1的范围(包括端值)内,并且j在0到num_points_in_qp_table_minus1[i]+1的范围(包括端值)内。

[0490]

[0491] 8.7.1量化参数的推导过程

[0492]

[0493] 在当前条带是I条带时,ChromaQpTable[i]被设置为等于IntraChromaQpTable[i](其中i=0...2)。在当前条带是B或P条带时,ChromaQpTable[i]被设置为等于InterChromaQpTable[i](其中i=0...2)。

[0494] 当ChromaArrayType不等于0并且treeType等于SINGLE_TREE或DUAL_TREE_CHROMA时,以下适用:

[0495] -当treeType等于DUAL_TREE_CHROMA时,变量 Q_{p_y} 被设置为等于覆盖亮度位置(xCb+cbWidth/2,yCb+cbHeight/2)的亮度编解码单元的亮度量化参数 Q_{p_y} 。

[0496] -变量 qP_{Cb} 、 qP_{Cr} 和 qP_{CbCr} 被推导如下:

- [0497] $qP_{\text{Chroma}} = \text{Clip3}(-QpBdOffset, 63, Qp_Y)$ (1143)
- [0498] $qP_{\text{Cb}} = \text{ChromaQpTable}[0][qP_{\text{Chroma}}]$ (1144)
- [0499] $qP_{\text{Cr}} = \text{ChromaQpTable}[1][qP_{\text{Chroma}}]$ (1145)
- [0500] $qP_{\text{CbCr}} = \text{ChromaQpTable}[2][qP_{\text{Chroma}}]$ (1146)
- [0501]
- [0502] 6.2. 实施例2: 枢轴点的数量和映射的色度QP的裁剪
- [0503] 以粗体斜体标记的改变基于JVET-Q2001-vC。
- [0504] 7.3.2.3 序列参数集Rbsp语法

		描述符
	seq_parameter_set_rbsp() {	
	
	if(ChromaArrayType != 0) {	
	sps_joint_cbr_enabled_flag	u(1)
	same_qp_table_for_chroma	u(1)
	numQpTables = same_qp_table_for_chroma ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
	for(i = 0; i < numQpTables; i++) {	
	[[qp_table_start_minus26 [i]]]	[[se(v)]]
[0505]	num_points_in_qp_table [[minus1]][[i]]	ue(v)
	if(num_points_in_qp_table)	
	<u>qp_table_start_minus26[i]</u>	<u>se(v)</u>
	for(j = 0; j < [[=]] num_points_in_qp_table[[minus1]][[i]; j++) {	
	delta_qp_in_val_minus1 [i][j]	ue(v)
	delta_qp_diff_val [i][j]	ue(v)
	}	
	}	
	}	
	

[0506] **num_points_in_qp_table**[[**minus1**]][[i]] [[加1]] 指定用于描述第i个色度QP映射表的点的数量。**num_points_in_qp_table**[[**minus1**]][[i]] 的值应当在0到63+QpBdOffset的范围(包括端值)内。[[当**num_points_in_qp_table_minus1**[0]不存在于比特流中时,**num_points_in_qp_table_minus1**[0]的值被推断为等于0。]]

[0507] **delta_qp_in_val_minus1**[i][j] 指定用于推导第i个色度QP映射表的第j个枢轴点的输入坐标的增量值。当**delta_qp_in_val_minus1**[0][j]不存在于比特流中时，

delta_qp_in_val_minus1[0][j]的值被推断为等于0。

[0508] **delta_qp_diff_val**[i][j]指定用于推导第i个色度QP映射表的第j个枢轴点的输出坐标的增量值。

[0509] 第i个色度QP映射表ChromaQpTable[i] (其中i=0...numQpTables-1)被推导如下:

[0510] 当same_qp_table_for_chroma等于1时,ChromaQpTable[1][k]和ChromaQpTable[2][k]被设置为等于ChromaQpTable[0][k],其中k在-QpBdOffset到63的范围(包括端值)内。

```

[[qpInVal[ i ][ 0 ] = qp_table_start_minus26[ i ] + 26
qpOutVal[ i ][ 0 ] = qpInVal[ i ][ 0 ]
for( j = 0; j <= num_points_in_qp_table_minus1[ i ]; j++ ) {
    qpInVal[ i ][ j + 1 ] = qpInVal[ i ][ j ] + delta_qp_in_val_minus1[ i ][ j ] + 1
    qpOutVal[ i ][ j + 1 ] = qpOutVal[ i ][ j ] +
( delta_qp_in_val_minus1[ i ][ j ] ^ delta_qp_diff_val[ i ][ j ] )
}
ChromaQpTable[ i ][ qpInVal[ i ][ 0 ] ] = qpOutVal[ i ][ 0 ]
[0511] for( k = qpInVal[ i ][ 0 ] - 1 63; k >= -QpBdOffset; k -- )
    ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k + 1 ] - 1 )
for( j = 0; j <= num_points_in_qp_table_minus1[ i ]; j++ ) {
    sh = ( delta_qp_in_val_minus1[ i ][ j ] + 1 ) >> 1
    for( k = qpInVal[ i ][ j ] + 1, m = 1; k <= qpInVal[ i ][ j + 1 ]; k++, m++ )
        ChromaQpTable[ i ][ k ] = ChromaQpTable[ i ][ qpInVal[ i ][ j ] ] +
            ( ( qpOutVal[ i ][ j + 1 ] - qpOutVal[ i ][ j ] ) * m + sh ) /
( delta_qp_in_val_minus1[ i ][ j ] + 1 )
}

```

```

for( k = qpInVal[ i ][ num_points_in_qp_table_minus1[ i ] + 1 ] + 1; k <= 63; k++)
    ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k - 1 ] + 1 )]
for( k = 63; k >= -QpBdOffset; k -- )
    ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k + 1 ] - 1 )
if( num_points_in_qp_table[ i ] > 0 ) {
    qpInVal[ i ][ 0 ] = qp_table_start_minus26[ i ] + 26
    qpOutVal[ i ][ 0 ] = qpInVal[ i ][ 0 ]
}
for( j = 0; j < num_points_in_qp_table[ i ]; j++ ) {
    qpInVal[ i ][ j + 1 ] = qpInVal[ i ][ j ] + delta_qp_in_val_minus1[ i ][ j ] + 1
    qpOutVal[ i ][ j + 1 ] = qpOutVal[ i ][ j ] +
[0512] ( delta_qp_in_val_minus1[ i ][ j ] ^ delta_qp_diff_val[ i ][ j ] ) sh =
    ( delta_qp_in_val_minus1[ i ][ j ] + 1 ) >> 1
    for( k = qpInVal[ i ][ j ] + 1, m = 1; k <= qpInVal[ i ][ j + 1 ]; k++, m++ )
        ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63,
        ChromaQpTable[ i ][ qpInVal[ i ][ j ] ] +
        ( ( qpOutVal[ i ][ j + 1 ] - qpOutVal[ i ][ j ] ) * m + sh ) /
        ( delta_qp_in_val_minus1[ i ][ j ] + 1 ) )
}
if( num_points_in_qp_table[ i ] > 0 )
    for( k = qpInVal[ i ][ num_points_in_qp_table[ i ] ] + 1; k <= 63; k++ )
        ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k - 1 ]
        + 1 )

```

[0513] 比特流一致性的要求是, qpInVal[i][j][[和qpOutVal[i][j]]] 的值应当在 -QpBdOffset到63的范围(包括端值)内, 其中i在0到numQpTables-1的范围(包括端值)内, 并且j在0到num_points_in_qp_table_minus1[i]+1的范围(包括端值)内。

[0514] 6.3. 实施例3

[0515] 以粗体斜体下划线文本标记的改变基于JVET-Q2001-vE。

[0516] 第i个色度QP映射表ChromaQpTable[i] (其中i=0...numQpTables-1)被推导如下:

```

qpInVal[ i ][ 0 ] = qp_table_start_minus26[ i ] + 26
[0517] qpOutVal[ i ][ 0 ] = qpInVal[ i ][ 0 ]

```

```

for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {
    qpInVal[i][j + 1] = qpInVal[i][j] + delta_qp_in_val_minus1[i][j] + 1
    qpOutVal[i][j + 1] = qpOutVal[i][j] + ((delta_qp_in_val_minus1[i][j] + 1)
        ^ delta_qp_diff_val[i][j])
}
ChromaQpTable[i][qpInVal[i][0]] = qpOutVal[i][0]
for(k = qpInVal[i][0] - 1; k >= -QpBdOffset; k--)
    ChromaQpTable[i][k] = Clip3(-QpBdOffset, 63, ChromaQpTable[i][k + 1] - 1)
[0518] for(j = 0; j <= num_points_in_qp_table_minus1[i]; j++) {
    sh = (delta_qp_in_val_minus1[i][j] + 1) >> 1
    for(k = qpInVal[i][j] + 1, m = 1; k <= qpInVal[i][j + 1]; k++, m++)
        ChromaQpTable[i][k] = ChromaQpTable[i][qpInVal[i][j]] +
            ((qpOutVal[i][j + 1] - qpOutVal[i][j]) * m + sh) /
            (delta_qp_in_val_minus1[i][j] + 1)
}
for(k = qpInVal[i][num_points_in_qp_table_minus1[i] + 1] + 1; k <= 63; k++)
    ChromaQpTable[i][k] = Clip3(-QpBdOffset, 63, ChromaQpTable[i][k - 1] + 1)

```

[0519] 当same_qp_table_for_chroma等于1时,ChromaQpTable[1][k]和Chroma QpTable [2][k]被设置为等于ChromaQpTable[0][k],其中k在-QpBdOffset到63的范围(包括端值)内。

[0520] 比特流一致性的要求是,qpInVal[i][j]和qpOutVal[i][j]的值应当在-QpBdOffset到63的范围(包括端值)内,其中i在0到numQpTables-1的范围(包括端值)内,并且j在0到num_points_in_qp_table_minus1[i]+1的范围(包括端值)内。

[0521] 6.4. 实施例4

[0522] 新添加的文本以粗体斜体下划线文本标记,并且被删除的文本以指示双括号之间的被删除的文本的左右双括号(例如,[[]])标记。其基于JVET-Q2001-vE。

[0523] 7.3.2.3序列参数集RBSP语法

[0524]	seq_parameter_set_rbsp() {	描述符
	...	

[0525]

if(ChromaArrayType != 0) {	
<u>sps one set of chroma qp tables flag</u>	<u>u(1)</u>
sps_joint_cbr_enabled_flag	u(1)
same_qp_table_for_chroma_set0	u(1)
numQpTablesSet0 = same_qp_table_for_chroma_set0 ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
for(i = 0; i < numQpTables; i++) {	
[[qp_table_start_minus26[i]]	[[se(v)]]
num_points_in_qp_table[[_minus1]]_set0[i]	ue(v)
<u>if(num points in qp table set0[i])</u>	
<u>qp table start set0[i]</u>	<u>ue(v)</u>
for(j = 0; j < [[=]] num_points_in_qp_table[[_minus1]]_set0[i]; j++) {	
delta_qp_in_val_minus1_set0[i][j]	ue(v)
delta_qp_diff_val_set0[i][j]	ue(v)
}	
}	
<u>if(!sps one set of chroma qp tables flag) {</u>	
<u>same qp table for chroma set1</u>	<u>u(1)</u>
<u>numQpTablesSet1 = same qp table for chroma set1 ? 1 :</u> <u>(sps joint cbr enabled flag ? 3 : 2)</u>	
<u>for(i = 0; i < numQpTablesSet1; i++) {</u>	
<u>num points in qp table set1[i]</u>	<u>ue(v)</u>
<u>if(num points in qp table set1[i])</u>	
<u>qp table start set1[i]</u>	<u>ue(v)</u>
<u>for(j = 0; j < num points in qp table set1[i]; j++) {</u>	
<u>delta qp in val minus1 set1[i][j]</u>	<u>ue(v)</u>
<u>delta qp diff val set1[i][j]</u>	<u>ue(v)</u>
<u>}</u>	
<u>}</u>	
<u>}</u>	

	}	
[0526]	...	

[0527] 7.4.3.3序列参数集RBSP语义

[0528] ...

[0529] *sps one set of chroma qp tables flag* 等于 0 指定有两个色度QP映射表组。*sps one set of chroma qp tables flag* 等于 1 指定只有一个色度QP映射表组。

[0530] **same_qp_table_for_chroma_set0** 等于 1 对于第1色度QP映射表组 指定仅信令通知一个色度QP映射表,并且当sps_joint_cbr_enabled_flag等于1时,该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差。**same_qp_table_for_chroma_set0** 等于 0 对于第1色度QP映射表组 指定在SPS中信令通知色度QP映射表,当sps_joint_cbr_enabled_flag等于1时,对于Cb和Cr信令通知两个色度QP映射表,并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当**same_qp_table_for_chroma_set0** 不存在于比特流中时,**same_qp_table_for_chroma_set0** 的值被推断为等于1。

[0531] **num_points_in_qp_table_minus1_set0**[i] 加 1 对于第1色度QP映射表组 指定用于描述第i个色度QP映射表的点的数量。**num_points_in_qp_table_minus1_set0**[i] 的值应当在0到63+QpBdOffset的范围(包括端值)内。当**num_points_in_qp_table_minus1_set0**[0] 不存在于比特流中时,**num_points_in_qp_table_minus1_set0**[0] 的值被推断为等于0。

[0532] **qp_table_start[[_minus26]]_set0**[i][[_加26]] 指定 对于第1色度QP映射表组 指定用于描述第i个色度QP映射表的起始亮度和色度QP。**qp_table_start[[_minus26]]_set0**[i] 的值应当在[[-26]]-QpBdOffset到[[36]]62的范围(包括端值)内。当**qp_table_start[[_minus26]]_set0**[i] 不存在于比特流中时,**qp_table_start[[_minus26]]_set0**[i] 的值被推断为等于0。

[0533] **delta_qp_in_val_minus1_set0**[i][j] 对于第1色度QP映射表组 指定用于推导第i个色度QP映射表的第j个枢轴点的输入坐标的增量值。当**delta_qp_in_val_minus1_set0**[0][j] 不存在于比特流中时,**delta_qp_in_val_minus1_set0**[0][j] 的值被推断为等于0。

[0534] **delta_qp_diff_val_set0**[i][j] 对于第1色度QP映射表组 指定用于推导第i个色度QP映射表的第j个枢轴点的输出坐标的增量值。

same qp table for chroma set1 等于1对于第2色度QP映射表组指定仅信令通知一个色度QP映射表, 并且当sps joint cbr enabled flag 等于1时, 该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差。
same qp table for chroma set1 等于0对于第2色度QP映射表组指定在SPS中信令通知色度QP映射表, 当sps joint cbr enabled flag 等于1时, 对于Cb和Cr信令通知两个色度QP映射表, 并且对于联合Cb-Cr信令通知一个附加色度QP映射表。 当 same qp table for chroma set1 不存在于比特流中时, same qp table for chroma set1 的值被推断为等于1。

num points in qp table minus1 set1[i] 加1对于第2色度QP映射表组指定用于描述第*i*个色度QP映射表的点的数量。
num points in qp table minus1 set1[i] 的值应当在0到63+OpBdOffset的范围(包括端值)内。当num points in qp table minus1 set1[0]不存在于比特流中时, num points in qp table minus1 set1[0] 的值被推断为等于0。

[0535]

qp table start set1[i]对于第2色度QP映射表组指定用于描述第*i*个色度QP映射表的起始亮度和色度QP。qp table start set1[i] 的值应当在-OpBdOffset到62的范围(包括端值)内。当qp table start set1[i]不存在于比特流中时, qp table start set1[i] 的值被推断为等于0。

delta qp in val minus1 set1[i][j]对于第2色度QP映射表组指定用于推导第*i*个色度QP映射表的第*j*个枢轴点的输入坐标的增量值。当delta qp in val minus1 set1[0][j]不存在于比特流中时, delta qp in val minus1 set1[0][j] 的值被推断为等于0。

delta qp diff val set1[i][j]对于第2色度QP映射表组指定用于推导用于第*i*个色度QP映射表的第*j*个枢轴点的输出坐标的增量值。

第*i*个色度QP映射表ChromaOpTableSet0[i] (其中*i*=0...numOpTablesSet0-1) 和第*i*个色度QP映射表ChromaOpTableSet1[i] (*i*=0...numOpTablesSet1-1) 被推导如下:

假设ChromaOpTable[i]表示ChromaOpTableSetX[i], qp table start[i]表示 qp table start setX[i] , delta qp in val minus1[i] 表示 delta qp in val minus1 setX[i] , delta qp diff val[i] 表示 delta qp diff val setX[i] (其中X分别为0和1), 以下适用:

```

qpInVal[ i ][ 0 ] = qp_table_start[[_minus26]][ i ][ [+ 26]]
qpOutVal[ i ][ 0 ] = qpInVal[ i ][ 0 ]
for( j = 0; j < [=] num_points_in_qp_table[[_minus1]][ i ]; j++ ) {
    qpInVal[ i ][ j + 1 ] = qpInVal[ i ][ j ] + delta_qp_in_val_minus1[ i ][ j ] + 1
    qpOutVal[ i ][ j + 1 ] = qpOutVal[ i ][ j ] +
    ( delta_qp_in_val_minus1[ i ][ j ] ^ delta_qp_diff_val[ i ][ j ] )
}
ChromaQpTable[ i ][ qpInVal[ i ][ 0 ] ] = qpOutVal[ i ][ 0 ]
for( k = qpInVal[ i ][ 0 ] - 1; k >= -QpBdOffset; k -- )
    ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k + 1 ] - 1 )
(62)
for( j = 0; j < [=] num_points_in_qp_table[[_minus1]][ i ]; j++ ) {
    sh = ( delta_qp_in_val_minus1[ i ][ j ] + 1 ) >> 1
[0536] for( k = qpInVal[ i ][ j ] + 1, m = 1; k <= qpInVal[ i ][ j + 1 ]; k++, m++ )
        ChromaQpTable[ i ][ k ] = ChromaQpTable[ i ][ qpInVal[ i ][ j ] ] +
        ( ( qpOutVal[ i ][ j + 1 ] - qpOutVal[ i ][ j ] ) * m + sh ) /
    ( delta_qp_in_val_minus1[ i ][ j ] + 1 )
}
if( num_points_in_qp_table[ i ] == 0 )
for( k = qpInVal[ i ][ 0 ] + 1; k <= 63; k ++ )
ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63,
ChromaQpTable[ i ][ k - 1 ] + 1 )
else
    for( k = qpInVal[ i ][ num_points_in_qp_table[[_minus1]][ i ][ [+ 1]] ] + 1; k <= 63;
    k++)
        ChromaQpTable[ i ][ k ] = Clip3( -QpBdOffset, 63, ChromaQpTable[ i ][ k - 1 ] +
    1 )

```

[0537] 当 same_qp_table_for_chroma X 等于1时, ChromaQpTable X [1][k] 和 ChromaQpTable X [2][k] 被设置为等于 ChromaQpTable X [0][k], 其中 X 为0和1, 并且 k 在在 $-QpBdOffset$ 到 63 的范围 (包括端值) 内。

[0538] 比特流一致性的要求是, $qpInVal[i][j]$ 和 $qpOutVal[i][j]$ 的值应当在 $-QpBdOffset$ 到 63 的范围 (包括端值) 内, 其中 i 在 0 到 $numQpTables-1$ 的范围 (包括端值) 内, 并且 j 在 0 到 $num_points_in_qp_table_minus1[i]+1$ 的范围 (包括端值) 内。

[0539] ...

[0540] 8.7.1 量化参数的推导过程

[0541] ...

[0542] 在当前条带是I条带时，ChromaOpTable[i]设置为等于ChromaOpTableSet0[i]（其中i=0...2）。在当前条带为B或P条带时，当sps one set of chroma qp tables flag等于0时，ChromaOpTable[i]被设置为等于ChromaOpTableSet1[i]（其中i=0...2），否则（当sps one set of chroma qp tables flag等于1时），ChromaOpTable[i]设置为等于ChromaOpTableSet0[i]（其中i=0...2）。

[0543] 当ChromaArrayType不等于0并且treeType等于SINGLE_TREE或DUAL_TREE_CHROMA时,以下适用:

[0544] -当treeType等于DUAL_TREE_CHROMA时,变量 Q_{p_Y} 被设置为等于覆盖亮度位置($x_{Cb} + cbWidth/2, y_{Cb} + cbHeight/2$)的亮度编解码单元的亮度量化参数 Q_{p_Y} 。

[0545] -变量 qP_{Cb} 、 qP_{Cr} 和 qP_{CbCr} 被推导如下:

[0546] $qP_{Chroma} = Clip3(-QpBdOffset, 63, Q_{p_Y})$ (1143)

[0547] $qP_{Cb} = ChromaOpTable[0][qP_{Chroma}]$ (1144)

[0548] $qP_{Cr} = ChromaOpTable[1][qP_{Chroma}]$ (1145)

[0549] $qP_{CbCr} = ChromaOpTable[2][qP_{Chroma}]$ (1146)

[0550] ...

[0551] 6.5. 实施例5

[0552] 新添加的文本以粗体斜体下划线标记,并且被删除的文本以指示双括号之间的被删除的文本的左右双括号(例如,[[]])标记。其基于JVET-Q2001-vE。

[0553] 7.3.2.3 序列参数集RBSP语法

[0554] seq_parameter_set_rbsp() {	描述符
...	

	if(ChromaArrayType != 0) {	
	<u>sps one set of chroma qp tables flag</u>	<u>u(1)</u>
	sps_joint_cbr_enabled_flag	u(1)
	<u>numTables = sps one set of chroma qp tables flag ? 1 : 2</u>	
	<u>for(l = 0; l <= numTables; l++) {</u>	
	same_qp_table_for_chroma[l]	u(1)
	numQpTables[l] = same_qp_table_for_chroma[l] ? 1 : (sps_joint_cbr_enabled_flag ? 3 : 2)	
	for(i = 0; i < numQpTables[l]; i++) {	
[0555]	qp_table_start_minus26[l][i]	se(v)
	num_points_in_qp_table_minus1[l][i]	ue(v)
	for(j = 0; j <= num_points_in_qp_table_minus1[l][i]; j++) {	
	delta_qp_in_val_minus1[l][i][j]	ue(v)
	delta_qp_diff_val[l][i][j]	ue(v)
	}	
	}	
	}	
	}	
	...	

[0556] 7.4.3.3序列参数集RBSP语义

[0557] ...

[0558] sps one set of chroma qp tables flag 等于 0 指定有两个色度QP映射表组。sps one set of chroma qp tables flag 等于 1 指定只有一个色度QP映射表组。当不存在时，sps one set of chroma qp tables flag 的值被推断为等于 1。

[0559] same_qp_table_for_chroma[l] 等于 1 对于第l色度QP映射表组 指定仅信令通知一个色度QP映射表，并且当sps_joint_cbr_enabled_flag等于1时，该表适用于Cb和Cr残差并且附加地适用于联合Cb-Cr残差。same_qp_table_for_chroma[l] 等于 0 对于第l色度QP映射表组 指定在SPS中信令通知色度QP映射表，当sps_joint_cbr_enabled_flag等于1时，对于Cb和Cr信令通知两个色度QP映射表，并且对于联合Cb-Cr信令通知一个附加色度QP映射表。当same_qp_table_for_chroma[l] 不存在于比特流中时，same_qp_table_for_chroma[l] 的值被推断为等于1。

[0560] $qp_table_start_minus26[l][i]$ 加26对于第*l*色度QP映射表组指定用于描述第*i*个色度QP映射表的起始亮度和色度QP。 $qp_table_start_minus26[l][i]$ 的值应当在-26-QpBdOffset到36的范围(包括端值)内。当 $qp_table_start_minus26[l][i]$ 不存在于比特流中时, $qp_table_start_minus26[l][i]$ 的值被推断为等于0。

[0561] $num_points_in_qp_table_minus1[l][i]$ 加1对于第*l*色度QP映射表组指定用于描述第*i*色度QP映射表的点的数量。 $num_points_in_qp_table_minus1[l][i]$ 的值应当在0到63+QpBdOffset的范围(包括端值)内。当 $num_points_in_qp_table_minus1[l][i]$ 不存在于比特流中时, $num_points_in_qp_table_minus1[l][i]$ 的值被推断为等于0。

[0562] $delta_qp_in_val_minus1[l][i][j]$ 对于第*l*色度QP映射表组指定用于推导第*i*个色度QP映射表的第*j*个枢轴点的输入坐标的增量值。当 $delta_qp_in_val_minus1[l][0][j]$ 不存在于比特流中时, $delta_qp_in_val_minus1[l][0][j]$ 的值被推断为等于0。

[0563] $delta_qp_diff_val[l][i][j]$ 对于第*l*色度QP映射表组指定用于推导第*i*个色度QP映射表的第*j*个枢轴点的输出坐标的增量值。

第*l*个色度QP映射表 $ChromaOpTable[l][i]$ (其中 $l=0\dots$

[0564] **sps one set of chroma qp tables flag? 1:2, 并且 $i=0\dots numOpTables[l]-1$) 被推导如下:**

```

qpInVal[l][i][0] = qp_table_start_minus26[l][i] + 26
qpOutVal[l][i][0] = qpInVal[l][i][0]
for(j = 0; j <= num_points_in_qp_table_minus1[l][i]; j++) {
    qpInVal[l][i][j+1] = qpInVal[l][i][j] + delta_qp_in_val_minus1[l][i][j] + 1
    qpOutVal[l][i][j+1] = qpOutVal[l][i][j] +
        ( delta_qp_in_val_minus1[l][i][j] ^ delta_qp_diff_val[l][i][j] )
}
ChromaQpTable[l][i][ qpInVal[l][i][0] ] = qpOutVal[l][i][0]
for(k = qpInVal[l][i][0] - 1; k >= -QpBdOffset; k --)

```

```

    ChromaQpTable[L][i][k] = Clip3(-QpBdOffset, 63,
    ChromaQpTable[L][i][k+1]-1)
    for(j=0; j <= num_points_in_qp_table_minus1[L][i]; j++) {
        sh = (delta_qp_in_val_minus1[L][i][j]+1) >> 1
        for(k = qpInVal[L][i][j]+1, m = 1; k <= qpInVal[L][i][j+1]; k++, m++)
            ChromaQpTable[L][i][k] = ChromaQpTable[L][i][qpInVal[L][i][j]] +
[0566]         ((qpOutVal[L][i][j+1]-qpOutVal[L][i][j])*m+sh)/
            (delta_qp_in_val_minus1[L][i][j]+1)
    }
    for(k = qpInVal[L][i][num_points_in_qp_table_minus1[L][i]+1]+1; k <= 63;
    k++)
        ChromaQpTable[L][i][k] = Clip3(-QpBdOffset, 63, ChromaQpTable[L][i][k-1]
    +1)

```

[0567] 当 same_qp_table_for_chroma[L] 等于 1 时, ChromaQpTable[L][1][k] 和 ChromaQpTable[L][2][k] 被设置为等于 ChromaQpTable[L][0][k], 其中 k 在 -QpBdOffset 到 63 的范围 (包括端值) 内。

[0568] 比特流一致性的要求是, qpInVal[L][i][j] 和 qpOutVal[L][i][j] 的值应当在 -QpBdOffset 到 63 的范围 (包括端值) 内, 其中 i 在 0 到 numQpTables[L]-1 的范围 (包括端值) 内, 并且 j 在 0 到 num_points_in_qp_table_minus1[L][i]+1 的范围 (包括端值) 内。

[0569] ...

[0570] 8.7.1 量化参数的推导过程

[0571] ...

[0572] 在当前条带为 I 条带时, chromaQpTable[i] 被设置为等于 ChromaQpTable[0][i] (其中 i=0...2,)。在当前条带是 B 或 P 条带时, 当 sps one set of chroma qp tables flag 等于 0 时, chromaQpTable[i] 被设置为等于 ChromaQpTable[1][i] (其中 i=0...2), 否则 (当 sps one set of chroma qp tables flag 等于 1 时), chromaQpTable[i] 被设置为等于 ChromaQpTable[0][i] (其中 i=0...2)。

[0573] 当 ChromaArrayType 不等于 0 且 treeType 等于 SINGLE_TREE 或 DUAL_TREE_CHROMA 时, 以下适用:

[0574] -当 treeType 等于 DUAL_TREE_CHROMA 时, 变量 Q_{p_Y} 被设置为等于覆盖亮度位置 (xCb+cbWidth/2, yCb+cbHeight/2) 的亮度编解码单元的亮度量化参数 Q_{p_Y} 。

[0575] -变量 qP_{Cb} 、 qP_{Cr} 和 qP_{CbCr} 被推导如下:

[0576] $qP_{Chroma} = Clip3(-QpBdOffset, 63, Q_{p_Y})$ (1143)

[0577] $qP_{Cb} = [[C]]chromaQpTable[0][qP_{Chroma}]$ (1144)

[0578] $qP_{Cr} = [[C]]\underline{chromaQpTable}[1][qP_{Chroma}]$ (1145)

[0579] $qP_{CbCr} = [[C]]\underline{chromaQpTable}[2][qP_{Chroma}]$ (1146)

[0580] 6.6. 实施例6

[0581] 如下描述了在JVET-Q2001-vE的基础上的建议的规范改变。被删除的文本以指示双括号之间的被删除的文本的左右双括号(例如, [[]]) 标记, 并且新添加的文本以粗体斜体下划线文本突出显示。

[0582] 7.3.10.2编解码树单元语法

	coding_tree_unit() {	描述符
	xCtb = CtbAddrX << CtbLog2SizeY	
	yCtb = CtbAddrY << CtbLog2SizeY	
	if(slice_sao_luma_flag slice_sao_chroma_flag)	
	sao(CtbAddrX, CtbAddrY)	
	if(slice_alf_enabled_flag) {	
	alf_ctb_flag[0][CtbAddrX][CtbAddrY]	ae(v)
	if(alf_ctb_flag[0][CtbAddrX][CtbAddrY]) {	
	if(slice_num_alf_aps_ids_luma > 0)	
	alf_use_aps_flag	ae(v)
	if(alf_use_aps_flag) {	
	if(slice_num_alf_aps_ids_luma > 1)	
	alf_luma_prev_filter_idx	ae(v)
	} else	
[0583]	alf_luma_fixed_filter_idx	ae(v)
	} }	
	if(slice_alf_chroma_idc == 1 slice_alf_chroma_idc == 3) {	
	alf_ctb_flag[1][CtbAddrX][CtbAddrY]	ae(v)
	if(alf_ctb_flag[1][CtbAddrX][CtbAddrY]	
	&& alf_chroma_num_alt_filters_minus1 > 0)	
	alf_ctb_filter_alt_idx[0][CtbAddrX][CtbAddrY]	ae(v)
	}	
	if(slice_alf_chroma_idc == 2 slice_alf_chroma_idc == 3) {	
	alf_ctb_flag[2][CtbAddrX][CtbAddrY]	ae(v)
	if(alf_ctb_flag[2][CtbAddrX][CtbAddrY]	
	&& alf_chroma_num_alt_filters_minus1 > 0)	
	alf_ctb_filter_alt_idx[1][CtbAddrX][CtbAddrY]	ae(v)
	}	
	} }	
	if(slice_cc_alf_cb_enabled_flag)	
	alf_ctb_cc_cb_idc[CtbAddrX][CtbAddrY]	ae(v)
	if(slice_cc_alf_cr_enabled_flag)	
	alf_ctb_cc_cr_idc[CtbAddrX][CtbAddrY]	ae(v)
	} }	
[0584]	if(slice_type == 1 && qtbtt_dual_tree_intra_flag)	
	dual_tree_implicit_qt_split(xCtb, yCtb, CtbSizeY, 0)	
	else	
	coding_tree(xCtb, yCtb, CtbSizeY, CtbSizeY, 1, 1, 0, 0, 0, 0,	
	SINGLE_TREE, MODE_TYPE_ALL)	
	}	

[0585] 6.7. 第七组实施例

[0586] 这是上面章节5中总结的项目20的实施例组。

[0587] 改变后的文本是基于JVET-Q2001-vE中的最新VVC文本。已经被添加或修改的最相关部分以粗体斜体下划线文本突出显示,并且一些被删除的文本以指示双括号之间的被删除的文本的左右双括号(例如,[[]])标记。

[0588] 6.7.1. 示例实施例

[0589] 在一个示例中,SPS语法结构可以被改变如下:

	seq_parameter_set_rbsp() {	描述符
	...	u(4)
	sps_transform_skip_enabled_flag	u(1)
	if(sps_transform_skip_enabled_flag) {	
[0590]	<u>sps ts residual coding enabled flag</u>	<u>u(1)</u>
	log2_transform_skip_max_size_minus2	ue(v)
	sps_bdpcm_enabled_flag	u(1)
	}	
	...	

sps ts residual coding enabled flag 等于 1 指定 slice ts residual coding disabled flag 可以存在于 CLVS 中。

[0591] **sps ts residual coding enabled flag 等于 0 指定 slice ts residual coding disabled flag 不存在于 CLVS 中。当不存在时, sps ts residual coding enabled flag 的值被推断为等于 0。**

[0592] 在一个示例中,PPS语法结构可以被改变如下:

	pic_parameter_set_rbsp() {	描述符
	...	
	cabac_init_present_flag	u(1)
[0593]	<u>pps ts residual coding enabled flag</u>	<u>u(1)</u>
	if(pps ts residual coding enabled flag)	
	<u>ts residual coding flag in ph</u>	<u>u(1)</u>

pps ts residual coding enabled flag 等于 1 指定 ph/slice ts residual coding enabled flag 可以存在于参考 PPS 的图片/条带中。

[0594] **pps ts residual coding enabled flag 等于 0 指定 ph/slice ts residual coding enabled flag 不存在于参考 PPS 的图片/条带中。**

要求当 sps ts residual coding enabled flag 等于 0 时, pps ts residual coding enabled flag 的值应当等于 0。

要求当 *sps transform skip enabled flag* 等于 0 时，*pps ts residual coding enabled flag* 的值应当等于 0。

[0595] *ts residual coding flag in ph* 等于 1 指定 *ph ts residual coding disabled flag* 可以存在于 PH 中。*pps ts residual coding enabled flag* 等于 0 指定 *slice ts residual coding disabled flag* 可以存在于 SH 中。

[0596] 在一个示例中,PH语法结构可以改变如下:

	picture_header_structure() {	描述符
	...	
	if(output_flag_present_flag)	
[0597]	pic_output_flag	u(1)
	<u>if(<i>pps ts residual coding enabled flag</i> && <i>ts residual coding flag in ph</i>)</u>	
	<u><i>ph ts residual coding enabled flag</i></u>	<u>u(1)</u>
	...	

[0598] *ph ts residual coding enabled flag* 等于 0 指定 *residual coding()* 语法结构用于解析当前图片的所有条带的变换跳过块的残差样点。*ph ts residual coding enabled flag* 等于 1 指定 *residual ts coding()* 语法结构可用于解析当前图片的所有条带的变换跳过块的残差样点。当 *ph ts residual coding enabled flag* 不存在时，其被推断为等于 0。

[0599] 在一个示例中,PH语法结构可以改变如下:

	slice_header() {	描述符
	...	
[0600]	<u>if(<i>pps ts residual coding enabled flag</i> && <i>!ts residual coding flag in ph</i>)</u>	
	slice_ts_residual_coding_[[disabled]]<u>enabled</u> flag	u(1)

[0601] **slice_ts_residual_coding_[[disabled]]enabled flag** 等于 [[1]]**0** 指定使用 *residual_coding()* 语法结构用于解析当前条带的变换跳过块的残差样点。**slice_ts_residual_coding_[[disabled]]enabled flag** 等于 [[0]]**1** 指定 *residual_ts_coding()* 语法结构用于解析当前条带的变换跳过块的残差样点。当 **slice_ts_residual_coding_[[disabled]]enabled flag** 不存在时，其被推断为等于 [[0]]*ph ts residual coding enabled flag*。

[0602] 在一个示例中, transform_unit() 语法结构可以改变如下:

transform_unit(x0, y0, tbWidth, tbHeight, treeType, subTuIndex, chType) {	描述符
...	
if(tu_y_coded_flag[x0][y0] && treeType != DUAL_TREE_CHROMA) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][0] && tbWidth <= MaxTsSize && tbHeight <= MaxTsSize && (IntraSubPartitionsSplitType == ISP_NO_SPLIT) && !cu_sbt_flag)	
transform_skip_flag[x0][y0][0]	ae(v)
<u>if(slice_ts_residual_coding_enabled_flag && transform_skip_flag[x0][y0][0])</u>	
<u>tu ts residual coding flag[x0][y0][0]</u>	<u>ae(v)</u>
if(!transform_skip_flag[x0][y0][0] <u>!tu ts residual coding flag[x0][y0][0]</u> [[slice_ts_residual_coding_ disabled_flag]])	
residual_coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
else	
residual_ts_coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
}	
if(tu_cb_coded_flag[xC][yC] && treeType != DUAL_TREE_LUMA) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][1] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	

[0603]

[0604]

transform_skip_flag[xC][yC][1]	ae(v)
<u><i>if(slice ts residual coding enabled flag && transform skip flag[xC][yC][1])</i></u>	
<u><i>tu ts residual coding flag[xC][yC][1]</i></u>	<u><i>ae(v)</i></u>
if(!transform_skip_flag[xC][yC][1] <u><i>!tu ts residual coding flag[xC][yC][1]</i></u> [[slice_ts_residual_coding_disabled_flag]])	
residual_coding(xC, yC, Log2(wC), Log2(hC), 1)	
else	
residual_ts_coding(xC, yC, Log2(wC), Log2(hC), 1)	
}	
if(tu_cr_coded_flag[xC][yC] && treeType != DUAL_TREE_LUMA && !(tu_cb_coded_flag[xC][yC] && tu_joint_cbr_residual_flag[xC][yC])) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][2] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	
transform_skip_flag[xC][yC][2]	ae(v)
<u><i>if(slice ts residual coding enabled flag && transform skip flag[xC][yC][2])</i></u>	
<u><i>tu ts residual coding flag[xC][yC][2]</i></u>	<u><i>ae(v)</i></u>
if(!transform_skip_flag[xC][yC][2] <u><i>!tu ts residual coding flag[xC][yC][2]</i></u> [[slice_ts_residual_coding_disabled_flag]])	
residual_coding(xC, yC, Log2(wC), Log2(hC), 2)	
else	
residual_ts_coding(xC, yC, Log2(wC), Log2(hC), 2)	
}	
}	

[0605]

tu ts residual coding flag[x0][y0][cIdx] 等于 1 指定, 对于当前编解码单元, 当 slice ts residual coding enabled flag 和 transform skip flag[x0][y0][c

[0606] Idx 等于 1 时, 使用变换跳过残差编解码。 tu ts residual coding flag[x0][y0][cIdx] 等于 0 指定, 对于当前编解码单元, 当 slice ts residual coding enabled flag 和 transform skip flag[x0][y0][cIdx] 等于 1 时, 使用变换残差编解码。当 tu ts residual coding flag[x0][y0][cIdx] 没有被信令通知时, 其被推断为等于 0。

[0607] 6.7.2. 另一个示例实施例可替代地, PPS、PH、SH语法可以更改如下:

[0608]	pic_parameter_set_rbsp() {	描述符
	...	
	cabac_init_present_flag	u(1)
	<u>pps ts residual coding enabled flag</u>	<u>u(1)</u>

[0609] pps ts residual coding enabled flag 等于 1 指定 ph/slice ts residual coding enabled flag 可能存在于参考 PPS 的图片/条带中。 pps ts residual coding enabled flag 等于 0 指定 ph/slice ts residual coding enabled flag 不存在于参考 PPS 的图片/条带中。

要求当 sps ts residual coding enabled flag 等于 0 时, pps ts residual coding enabled flag 的值应当等于 0。

要求当 sps transform skip enabled flag 等于 0 时, pps ts residual coding enabled flag 的值应当等于 0。

[0610]	picture_header_structure() {	描述符
	...	
	if(output_flag_present_flag)	
	pic_output_flag	u(1)
	<u>if(pps ts residual coding enabled flag)</u>	
	<u>ph ts residual coding enabled flag</u>	<u>u(1)</u>
	...	

[0611] ph ts residual coding enabled flag 等于 0 指定 residual coding() 语法结构用于解析当前图片的所有条带的变换跳过块的残差样点。 ph ts residual coding enabled flag 等于 1 指定 residual ts coding() 语法结构可用于解析当前图片的所有条带的变换跳过块的残差样点。当 ph ts residual coding enabled flag 不存在时, 其被推断为等于 0。

	slice_header() {	描述符
	...	
[0612]	<u>if(ph ts residual coding enabled flag)</u>	
	slice_ts_residual_coding_:: disabled :: <u>enabled flag</u>	u(1)

[0613] **slice_ts_residual_coding_::**disabled**::enabled flag** 等于 **[[1]]0** 指定 residual_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。**slice_ts_residual_coding_::**disabled**::enabled flag** 等于 **[[0]]1** 指定 residual_ts_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。当 slice_ts_residual_coding_::**disabled**::enabled flag 不存在时,其被推断为等于0。

[0614] 6.7.3. 另一个示例实施例

[0615] 可替代地,PPS、SH语法可以更改如下:

	pic_parameter_set_rbsp() {	描述符
	...	
[0616]	cabac_init_present_flag	u(1)
	<u>pps ts residual coding enabled flag</u>	<u>u(1)</u>

[0617] pps ts residual coding enabled flag 等于1 指定 ph/slice ts residual coding enabled flag 可以存在于参考PPS的图片/条带中。pps ts residual coding enabled flag 等于0 指定 ph/slice ts residual coding enabled flag 不存在于参考PPS的图片/条带中。

要求当 sps ts residual coding enabled flag 等于0 时, pps ts residual coding enabled flag 的值应当等于0。

要求当 sps transform skip enabled flag 等于0 时, pps ts residual coding enabled flag 的值应当等于0。

	slice_header() {	描述符
	...	
[0618]	<u>if(pps ts residual coding enabled flag)</u>	
	slice_ts_residual_coding_:: disabled :: <u>enabled flag</u>	u(1)

[0619] **slice_ts_residual_coding_::**disabled**::enabled flag** 等于 **[[1]]0** 指定 residual_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。**slice_ts_residual_coding_::**disabled**::enabled flag** 等于 **[[0]]1** 指定 residual_ts_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。当 slice_ts_residual_coding_::**disabled**::enabled flag 不存在时,其被推断为等于0。

[0620] 6.7.4. 另一个示例实施例

[0621] 可替代地,SH语法可以被改变如下:

	slice_header() {	描述符
	...	
[0622]	<i>if(sps_transform_skip_enabled_flag)</i>	
	slice_ts_residual_coding_[[disabled]] <i>enabled_flag</i>	u(1)

[0623] **slice_ts_residual_coding_[[disabled]]*enabled_flag*** 等于 **[[1]]0** 指定使用 residual_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。**slice_ts_residual_coding_[[disabled]]*enabled_flag*** 等于 **[[0]]1** 指定 residual_ts_coding() 语法结构用于解析当前条带的变换跳过块的残差样点。当 slice_ts_residual_coding_[[disabled]]*enabled_flag* 不存在时,其被推断为等于0。

[0624] 图3是实现本文中所公开的各种技术的示例视频处理系统1900的框图。各种实现方式可以包括系统1900中的一些或全部组件。系统1900可以包括用于接收视频内容的输入1902。视频内容可以以原始或未压缩的格式(例如8或10比特多分量像素值)接收,或者可以以压缩或编码的格式接收。输入1902可以代表网络接口、外围总线接口或存储接口。网络接口的示例包括有线接口(诸如以太网、无源光网络(PON)等)和无线接口(诸如Wi-Fi或蜂窝接口)。

[0625] 系统1900可以包括可以实现本档中描述的各种编解码或编码方法的编解码组件1904。编解码组件1904可以减少从输入1902到编解码组件1904的输出的视频的平均比特率,以产生视频的编解码表示。因此,编解码技术有时称为视频压缩或视频转码技术。编解码组件1904的输出可以被存储或经由所连接的通信来发送,如组件1906所表示的。在输入1902处接收的视频的存储或通信的比特流(或编解码)表示可以由组件1908使用,以生成被发送到显示接口1910的像素值或可显示视频。从比特流表示中生成用户可见的视频的过程有时称为视频解压缩。此外,尽管某些视频处理操作被称为“编解码”操作或工具,但是应当理解,在编码器处使用编解码工具或操作,并且将由解码器进行反演编解码的结果的对应解码工具或操作。

[0626] 外围总线接口或显示接口的示例可以包括通用串行总线(USB)或高清晰度多媒体接口(HDMI)或Displayport等。存储接口的示例包括SATA(串行高级技术附件)、PCI、IDE接口等。本档中描述的技术可以实施在各种电子设备中,诸如移动电话、膝上型计算机、智能电话或其它能够进行数字数据处理和/或视频显示的装设备。

[0627] 图4是一个视频处理装置3600的框图。装置3600可以用于实现本文中所述的方法中的一个或多个。装置3600可以实施在智能电话、平板电脑、计算机、物联网(IoT)接收器等中。装置3600可以包括一个或多个处理器3602、一个或多个存储器3604和视频处理硬件3606。(多个)处理器3602可以配置为实现本档中所述的一个或多个方法。(多个)存储器3604可以用于存储数据和代码,该代码用于实现本文所描述的方法和技术。视频处理硬件3606可以用于在硬件电路中实现本档中所描述的一些技术。

[0628] 图6是示出可利用本公开的技术的示例视频编解码系统100的框图。

[0629] 如图6所示,视频编解码系统100可以包括源设备110和目的地设备120。源设备110生成编码的视频数据,其可以被称为视频编码设备。目的地设备120可以解码由源设备110生成的编码的视频数据,该目的地设备120可以被称为视频解码设备。

[0630] 源设备110可以包括视频源112、视频编码器114和输入/输出(I/O)接口116。

[0631] 视频源112可以包括例如视频捕获设备的源、从视频内容提供者接收视频数据的接口、和/或生成视频数据的计算机图形系统,或这些源的组合。视频数据可以包括一个或多个图片。视频编码器114对来自视频源112的视频数据进行编码以生成比特流。比特流可以包括形成视频数据的编解码表示的比特序列。比特流可以包括编解码图片和相关联的数据。编解码图片是图片的编解码表示。相关联的数据可以包括序列参数集、图片参数集和其他语法元素。I/O接口116包括调制器/解调器(调制解调器)和/或发送器。可以将编码的视频数据经由I/O接口116通过网络130a直接发送到目的地设备120。还可以将编码的视频数据存储到存储介质/服务器130b上,用于由目的地设备120存取。

[0632] 目的地设备120可以包括I/O接口126、视频解码器124和显示设备122。

[0633] I/O接口126可以包括接收器和/或调制解调器。I/O接口126可以从源设备110或存储介质/服务器130b获取编码的视频数据。视频解码器124可以对编码的视频数据进行解码。显示设备122可以向用户显示解码的视频数据。显示设备122可以与目的地设备120集成,或可以在配置为与外置显示设备相接的目的地设备120外部。

[0634] 视频编码器114和视频解码器124可以根据视频压缩标准(例如,高效视频编解码(HEVC)标准、多功能视频编解码(VVC)标准和其他当前和/或其他标准)进行操作。

[0635] 图7是示出视频编码器200的一个示例的框图,该视频编码器200可以是图6中示出的系统100中的视频编码器114。

[0636] 视频编码器200可以被配置为执行本公开的任何或全部技术。在图7的示例中,视频编码器200包括多个功能组件。本公开所描述的技术可以在视频编码器200的各种组件之间共享。在一些示例中,处理器可以配置为进行本公开中描述的任何或全部技术。

[0637] 视频编码器200的功能组件可以包括分割单元201、预测单元202(其可以包括模式选择单元203、运动估计单元204、运动补偿单元205和帧内预测单元206)、残差生成单元207、变换单元208、量化单元209、逆量化单元210、逆变换单元211、重建单元212、缓冲器213和熵编码单元214。

[0638] 在其他示例中,视频编码器200可以包括更多、更少或不同的功能组件。在一个示例中,预测单元202可以包括帧内块复制(IBC)单元。IBC单元可以以IBC模式进行预测,其中至少一个参考图片是当前视频块所位于的图片。

[0639] 此外,例如运动估计单元204和运动补偿单元205的一些组件可以被高度集成,但是出于解释的目的在图7的示例中分开表示。

[0640] 分割单元201可以将图片分割成一个或多个视频块。视频编码器200和视频解码器300可以支持各种视频块尺寸。

[0641] 模式选择单元203可以例如基于错误结果选择帧内或帧间的编解码模式中的一个,并且将得到的帧内或帧间编解码块提供到残差生成单元207来生成残差块数据而且提供到重建单元212来重建编解码块以用作参考图片。在一些示例中,模式选择单元203可以选择帧内和帧间预测的组合(CIIP)模式,其中预测是基于帧间预测信号和帧内预测信号。

模式选择单元203还可以为帧间预测情况下的块选择运动矢量的分辨率(例如子像素或整像素精度)。

[0642] 为了对当前视频块进行帧间预测,运动估计单元204可以通过将来自缓冲器213的一个或多个参考帧与当前视频块进行比较,生成当前视频块的运动信息。运动补偿单元205可以基于来自缓冲器213的图片(而不是与当前视频块相关联的图片)的运动信息和解码样点来为当前视频块确定预测的视频块。

[0643] 运动估计单元204和运动补偿单元205可以为当前视频块进行不同操作,例如执行不同操作取决于当前视频块是在I条带、P条带还是B条带中。

[0644] 在一些示例中,运动估计单元204可以进行当前视频块的单向预测,并且运动估计单元204可以在列表0或列表1的参考图片中搜索当前视频块的参考视频块。运动估计单元204然后可以生成指示列表0或列表1的参考图片中含有参考视频块的参考索引以及指示在当前视频块与参考视频块之间的空域位移的运动矢量。运动估计单元204可以输出参考索引、预测方向指示符、和运动矢量作为当前视频块的运动信息。运动补偿单元205可以基于由当前视频块的运动信息指示的参考视频块来生成当前块的预测视频块。

[0645] 在其他示例中,运动估计单元204可以进行当前视频块的双向预测,运动估计单元204可以在列表0的参考图片中搜索当前视频块的参考视频块并且还可以在列表1的参考图片中搜索当前视频块的另一个参考视频块。运动估计单元204然后可以生成指示列表0或列表1的参考图片中含有参考视频块的参考索引以及指示在参考视频块与当前视频块之间的空域位移的运动矢量。运动估计单元204可以输出参考索引和当前视频块的运动矢量作为当前视频块的运动信息。运动补偿单元205可以基于由当前视频块的运动信息指示的参考视频块来生成当前视频块的预测视频块。

[0646] 在一些示例中,运动估计单元204可以输出运动信息的全部集合,用于解码器的解码处理。

[0647] 在一些示例中,运动估计单元204可以不输出当前视频的运动信息的全部集合。而是,运动估计单元204可以参考另一个视频块的运动信息来信令通知当前视频块的运动信息。例如,运动估计单元204可以确定当前视频块的运动信息与相邻视频块的运动信息足够相似。

[0648] 在一个示例中,运动估计单元204可以在与当前视频块相关联的语法结构中指示:向视频解码器300指示当前视频块具有与另一个视频块相同的运动信息的值。

[0649] 在另一个示例中,运动估计单元204可以在与当前视频块相关联的语法结构中标识另一个视频块和运动矢量差(MVD)。运动矢量差指示当前视频块的运动矢量与指示视频块的运动矢量之间的差。视频解码器300可以使用指示视频块的运动矢量和运动矢量差来确定当前视频块的运动矢量。

[0650] 如上所讨论的,视频编码器200可以预测性地信令通知运动矢量。可以由视频编码器200实现的预测性的信令通知技术的两个示例包括高级运动矢量预测(AMVP)和merge模式信令通知。

[0651] 帧内预测单元206可以对当前视频块进行帧内预测。当帧内预测单元206对当前视频块进行帧内预测时,帧内预测单元206可以基于相同图片中其他视频块的解码样点来生成当前视频块的预测数据。当前视频块的预测数据可以包括预测视频块和各种语法元素。

[0652] 残差生成单元207可以通过从当前视频块中减去(例如,由减号表示)当前视频块的(多个)预测视频块来生成当前视频块的残差数据。当前视频块的残差数据可以包括对应于当前视频块中样点的不同样点分量的残差视频块。

[0653] 在其他示例中,例如在跳过模式下,对于当前视频块可能不存在当前视频块的残差数据,并且残差生成单元207可以不进行减去操作。

[0654] 变换处理单元208可以通过将一个或多个变换应用于与当前视频块相关联的残差视频块来生成当前视频块的一个或多个变换系数视频块。

[0655] 在变换处理单元208生成与当前视频块相关联的变换系数视频块之后,量化单元209可以基于与当前视频块相关联的一个或多个量化参数(QP)值来量化与当前视频块相关联的变换系数视频块。

[0656] 逆量化单元210和逆变换单元211可以将逆量化和逆变换分别应用于变换系数视频块,来从变换系数视频块重建残差视频块。重建单元212可以将重建的残差视频块添加到来自自由预测单元202生成的一个或多个预测视频块的对应样点,以产生与当前块相关联的重建视频块用于存储在缓冲器213中。

[0657] 在重建单元212重建视频块之后,可以进行环路滤波操作以降低视频块中视频块化伪影。

[0658] 熵编码单元214可以从视频编码器200的其他功能组件接收数据。当熵编码单元214接收数据时,熵编码单元214可以进行一个或多个熵编码操作以生成熵编码数据并且输出包括熵编码数据的比特流。

[0659] 所公开的技术的一些实施例包括作出决定或确定以启用视频处理工具或模式。在一个示例中,当启用视频处理工具或模式时,编码器将在视频的块的处理中使用或实现该工具或模式,但不一定基于该工具或模式的使用来修改得到的比特流。也就是说,当基于该决定或确定来启用视频处理工作或模式时,从视频的块到视频的比特流(或比特流表示)的转换将使用该视频处理工具或模式。在另一示例中,当启用视频处理工具或模式时,解码器将在知道比特流已基于视频处理工具或模式被修改的情况下处理比特流。即,将使用基于该决定或确定而启用的视频处理工具或模式来执行从视频的比特流到视频的块的转换。

[0660] 图8是示出视频解码器300的示例的框图,该视频解码器300可以是图6中示出的系统100中的视频解码器114。

[0661] 视频解码器300可以被配置为进行本公开的任何或全部技术。在图8的示例中,视频解码器300包括多个功能组件。本公开所描述的技术可以在视频解码器300的各种组件之间共享。在一些示例中,处理器可以配置为进行本公开中描述的任何或全部技术。

[0662] 在图8的示例中,视频解码器300包括熵解码单元301、运动补偿单元302、帧内预测单元303、逆量化单元304、逆变换单元305以及重建单元306和缓冲器307。在一些示例中,视频解码器300可以进行与关于视频编码器200(图7)所描述的编码过程总体反演的解码过程。

[0663] 熵解码单元301可以检索编码比特流。编码比特流可以包括熵编解码视频数据(例如,视频数据的编解码块)。熵解码单元301可以对熵编解码视频进行解码,并且根据熵解码视频数据,运动补偿单元302可以确定包括运动矢量、运动矢量精度、参考图片列表索引和其他运动信息的运动信息。运动补偿单元302例如可以通过进行AMVP和merge模式确定此类

信息。

[0664] 运动补偿单元302可以产生运动补偿块,可能地基于插值滤波器进行插值。要以子像素精度使用的插值滤波器的标识符可以包括在语法元素中。

[0665] 运动补偿单元302可以使用由视频编码器200在编码视频块的期间所使用的插值滤波器,来计算出参考块的子整数个像素的插值的值。运动补偿单元302可以根据接收的语法信息确定由视频编码器200所使用的插值滤波器并且使用插值滤波器来产生预测块。

[0666] 运动补偿单元302可以使用一些语法信息来确定:用于对编码视频序列的(多个)帧和/或(多个)条带进行编码的块的尺寸,描述编码视频序列的每个宏块如何被分割的分割信息,指示如何编码每个分割的模式,每个帧间编码块的一个或多个参考帧(和参考帧列表),以及对编码视频序列进行解码的其他信息。

[0667] 帧内预测单元303可以使用例如在比特流中接收的帧内预测模式来从空域相邻块形成预测块。逆量化单元303逆量化(即,去量化)在比特流中提供的且由熵解码单元301解码的量化的视频块系数。逆变换单元303应用逆变换。

[0668] 重建单元306可以用由运动补偿单元202或帧内预测单元303生成的对应预测块求和残差块,以形成解码块。如所期望的,去块滤波器还可以应用于滤波解码块以便移除块效应伪影。解码视频块然后存储在缓冲器307中,该缓冲器307提供用于随后的运动补偿/帧内预测的参考块,并且还产生用于在显示设备上呈现的解码视频。

[0669] 接下来提供了一些实施例优选的解决方案的列表。

[0670] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目1)。

[0671] 1.一种视频处理方法(例如,图5中的方法500),包括:执行(502)包括视频图片的视频和该视频的编解码表示之间的转换,其中该编解码表示符合格式规则,其中该格式规则指定指示在所述转换期间使用的一个或多个初始量化参数(QP)值的一个或多个语法元素是否以及在哪儿被包括在该编解码表示中。

[0672] 2.如解决方案1所述的方法,其中,格式规则指定将指示一个或多个QP值的语法元素被包括在图片标头中,而不将语法元素包括在图片参数集中。

[0673] 3.如解决方案1所述的方法,其中,格式规则指定指示所述一个或多个QP值的语法元素被包括在图片参数集中并在一个或多个图片标头中重复。

[0674] 4.如解决方案1所述的方法,其中,格式规则指定一个或多个语法元素根据规则在多个级别被包括在编解码表示中,该多个级别包括序列参数集、图片参数集、图片标头和序列标头中的两个或多个。

[0675] 5.如解决方案4所述的方法,其中,该规则基于视频图片的视频图片类型。

[0676] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目2-3)。

[0677] 6.一种视频处理方法,包括:为视频的视频块和视频的编解码表示之间的转换,确定使用初始量化参数(QP)值;以及基于该确定来执行转换,其中编解码表示中的语法元素使用偏移关系来指示初始QP值,该偏移关系包括偏移值K,该偏移值K是(a) 1和25之间或者(b) 大于26,或者(c) 用于转换的最大允许QP值的函数,或者(d) 用于转换的比特深度的函数。

[0678] 7.如解决方案6所述的方法,其中,K是视频块的类型或包含视频块的视频图片或视频条带的类型的函数。

- [0679] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目4-5)。
- [0680] 8.一种视频处理方法,包括:为视频的色度块和视频的编解码表示之间的转换,根据规则,基于包含色度块的图片或条带的类型,确定一个或多个量化参数(QP)表;以及基于该确定来执行转换。
- [0681] 9.如解决方案8所述的方法,其中,规则指定一个或多个QP表不同于I、B和P视频图片。
- [0682] 10.如解决方案8所述的方法,其中,规则指定来自I编解码视频图片的一个或多个QP表不同于用于B和P编解码视频图片的一个或多个QP表。
- [0683] 11.如解决方案8-10中任一项所述的方法,其中,规则基于色度块所属的图片类型和条带类型的分类,其中数量M表示分类的索引,其中M是大于1的整数。
- [0684] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目6-8)。
- [0685] 12.一种视频处理方法,包括:为视频的色度块和视频的编解码表示之间的转换确定一个或多个量化参数(QP)表;并且基于该确定来执行转换;其中该转换符合指定对应于一个或多个QP表的一个或多个语法元素的第一规则或者指定一个或多个QP表和色度块的编解码特性之间的关系第二规则。
- [0686] 13.如解决方案12所述的方法,其中,第一规则指定一个或多个语法元素被包括在与编解码表示相关联的序列参数集中。
- [0687] 14.如解决方案12-13中任一项所述的方法,其中,第二规则指定所述一个或多个QP表基于用于所述转换的预测模式。
- [0688] 15.如解决方案13所述的方法,其中,第二规则指定用于帧内编解码色度块的所述一个或多个QP表不同于使用其他预测模式的色度块。
- [0689] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目9-14)。
- [0690] 16.一种视频处理方法,包括:执行视频的色度块和视频的编解码表示之间的转换;其中编解码表示符合格式规则,其中该格式规则指定与用于所述转换的一个或多个色度量化的参数(QP)表相关联的语法元素的特性。
- [0691] 17.如解决方案16所述的方法,其中,格式规则指定用于一个或多个色度QP表的起始点语法元素的最大值。
- [0692] 18.如解决方案16所述的方法,其中,格式规则指定起始点语法元素具有偏移小于26的数量K的值。
- [0693] 19.如解决方案16所述的方法,其中,格式规则指定起始点语法元素偏移一个值,所述值取决于色度块的当前图片是否是仅帧内图片。
- [0694] 20.如解决方案16-19中任一项所述的方法,其中,格式规则指定一个或多个色度QP表中的枢轴点的数量为零。
- [0695] 21.如解决方案16-20中任一项所述的方法,由于枢轴点的数量为零,一个或多个色度QP表中的第i个条目被设置为等于对应的亮度QP表中的第i个条目。
- [0696] 22.如解决方案16-21中任一项所述的方法,其中,语法元素被包括在编解码表示中的序列参数集和图片参数集两者中。
- [0697] 23.如解决方案16-22中任一项所述的方法,其中,语法元素被包括在图片标头或序列标头中。

[0698] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目25)。

[0699] 24.一种视频处理方法,包括:基于与编解码表示中的一个或多个语法元素相关的格式规则,确定用于在视频的编解码表示中信令通知量化参数表的信令通知方案,并且基于该确定,执行视频和编解码表示之间的转换。

[0700] 25.如解决方案24所述的方法,其中,信令通知方案包括量化参数表中的点的数量的指示。

[0701] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目26)。

[0702] 26.一种视频处理方法,包括:根据规则执行视频的视频块和视频的编解码表示之间的转换,其中所述规则指定数字N,在仿射编解码的使用被启用并且基于子块的时域运动矢量预测的使用被禁用的情况下,所述数字N被用于限制子块merge候选的最大数量。

[0703] 27.如解决方案26所述的方法,其中,N=0。

[0704] 以下解决方案示出了上一章节中讨论的技术的示例实施例(例如,项目27-29)。

[0705] 28.一种视频处理方法,包括:执行包括一个或多个视频图片的视频和所述视频的编解码表示之间的转换,该一个或多个视频图片包括一个或多个视频子图片,其中所述编解码表示符合格式规则,其中所述格式规则指定包括或省略一个或多个以下与子图片相关的语法元素:指示参考相同序列参数集的所有子图片是否具有相同维度的第一语法元素,信令通知具有相同维度的所有子图片的公共宽度和公共高度的第二语法元素,或者根据第二格式规则的第三语法元素。

[0706] 29.如解决方案28所述的方法,其中,所述第二格式规则指定在某些条件下省略指示样点宽度的语法元素。

[0707] 30.如解决方案1-29中任一项所述的方法,其中,所述转换包括将所述视频编码成所述编解码表示。

[0708] 31.如解决方案1-29中任一项所述的方法,其中,所述转换包括对所述编解码表示进行解码以生成所述视频的像素值。

[0709] 32.一种视频解码装置,包括被配置为实现解决方案1至31中的一个或多个所述的方法的处理器。

[0710] 33.一种视频编码装置,包括被配置为实现解决方案1至31中的一个或多个所述的方法的处理器。

[0711] 34.一种其上存储有计算机代码的计算机程序产品,该代码在由处理器执行时使用所述处理器实现解决方案1至31中任一项所述的方法。

[0712] 图9是视频处理的一个示例方法900的流程图。操作902包括根据规则执行视频和视频的比特流之间的转换,其中,规则指定序列参数集包括在转换中使用的量化参数表,并且规则指定是否或者如何根据序列参数集中的第二语法元素来控制序列参数集包括指示量化参数表中的点的数量的第一语法元素。

[0713] 在方法900的一些实施例中,量化参数表是色度量化参数映射表。在方法900的一些实施例中,第一语法元素的值加1等于量化参数表中的点的数量。在方法900的一些实施例中,第二语法元素的值指示用于描述量化参数表的起始亮度和色度量化参数。在方法900的一些实施例中,量化参数表中的点的数量的最大值基于第二语法元素的值。在方法900的一些实施例中,规则指定量化参数表中的点的数量的最大值减1是最大量化参数值与用于

描述第*i*个色度量化参数映射表的起始亮度和色度量化参数之间的差。

[0714] 在方法900的一些实施例中,最大值设置为(63-(qp_table_start_minus26[i]+26))。在方法900的一些实施例中,最大值被设置为(64-(qp_table_start_minus26[i]+26))。

[0715] 图10是视频处理的一个示例方法1000的流程图。操作1002包括根据规则执行视频和视频的比特流之间的转换,其中,规则指定当序列参数集中指示的子图片的数量减1大于零时,并且当序列参数集中的语法元素指示编解码层视频序列中的视频的第一视频图片的第*i*个子图片在排除环路滤波操作的解码过程中被视为视频的第二视频图片时,以下两个条件之一为真:(1)第一视频图片的第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片和第一视频图片对于以下各项中的每一项具有相同的值:以亮度样点为单位的图片宽度,以亮度样点为单位的图片高度,子图片的数量减1,以及对于0到子图片的数量减1,包括端值,的范围内的每个*i*值,第*i*个子图片的左顶编解码树单元的水平位置、第*i*子图片的左顶编解码树单元CTU的垂直位置、第*i*个子图片的宽度减1、第*i*个子图片的高度减1、第*i*个子图片的语法元素;和(2)第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片是层间参考图片,对于层间参考图片,子图片的数量减1的值等于零。

[0716] 图11是视频处理的一个示例方法1100的流程图。操作1102包括根据规则执行包括视频块的视频和视频的比特流之间的转换,其中规则指定当仿射编解码工具被启用并且基于子块的时域运动矢量预测编解码工具被禁用时,子块merge候选的最大数量不允许为数量*N*。

[0717] 在方法1100的一些实施例中, $N=0$ 。在方法1100的一些实施例中, N 大于0。在方法1100的一些实施例中,在序列参数集中支持的基于子块的merge运动矢量预测候选的最大数量减五的范围是从[0,4]开始,而与序列参数集中指示SbTMVP编解码工具是被启用还是被禁用的语法元素的值无关。在方法1100的一些实施例中,比特流是一致性比特流,当仿射编解码工具被启用时,一致性比特流包括1到5,包括端值,的范围内的基于子块的merge运动矢量预测候选的最大数量的值。在方法1100的一些实施例中,比特流是一致性比特流,当仿射编解码工具被禁用时,一致性比特流包括0到1,包括端值,的范围内的基于子块的merge运动矢量预测候选的最大数量的值。在方法1100的一些实施例中,比特流是否包括指示merge子块操作是否被启用的语法元素是基于仿射编解码工具是否被启用而不是基于子块merge候选的最大数量。

[0718] 图12是视频处理的一个示例方法1200的流程图。操作1202包括执行包括一个或多个视频图片的视频和视频的比特流之间的转换,其中比特流符合格式规则,并且格式规则指定指示是否启用仿射merge模式的语法元素是否在比特流中。

[0719] 在方法1200的一些实施例中,格式规则指定基于仿射merge模式是否在序列参数集中启用而在比特流中选择性地指示语法元素。在方法1200的一些实施例中,格式规则指定当语法元素指示仿射merge模式被启用时,在比特流中指示子块merge候选的最大数量。

[0720] 图13是视频处理的一个示例方法1300的流程图。操作1302包括根据规则执行包括一个或多个视频图片的视频和视频的比特流之间的转换,一个或多个视频图片包括一个或多个子图片,其中规则指定从序列参数集中指示或省略以下子图片相关语法元素中的一个

或多个:指示参考相同序列参数集的所有子图片是否具有相同维度的第一语法元素,指示具有相同维度的所有子图片的共同宽度和共同高度的第二语法元素,或当所有子图片不具有相同维度时,省略根据第二规则的一个或多个语法元素。

[0721] 在方法1300的一些实施例中,规则指定:表示为numSubpicCols的子图片列的数量被推导为等于 $(\text{sps_pic_width_max_in_luma_samples}/((\text{sps_uniform_subpic_width_minus1}+1)*\text{CtbSizeY}))$,表示为numSubpicRows的子图片行的数量被推导为等于 $(\text{sps_pic_height_max_in_luma_samples}/((\text{sps_uniform_subpic_height_minus1}+1)*\text{CtbSizeY}))$,并且表示为sumSubpics的子图片的数量被推导为等于 $\text{numSubpicCols}*\text{numSubpicRows}$ 。在方法1300的一些实施例中,规则指定从序列参数集中排除sps_num_subpics_minus1,并且sps_num_subpics_minus1的值被推断为等于 $\text{numSubpics}-1$ 。在方法1300的一些实施例中,规则指定对于0到sps_num_subpics_minus1,包括端值,的范围内的每个i值,sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]的值被推断为分别等于 $(\text{sps_uniform_subpic_width_minus1}+1)*(i\% \text{numSubpicCols})$ 、 $(\text{sps_uniform_subpic_height_minus1}+1)*(i/\text{numSubpicCols})$ 、sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1。

[0722] 在方法1300的一些实施例中,规则指定对于0到sps_num_subpics_minus1,包括端值,的范围内的每个i值,当sps_independent_subpics_flag等于0时,sps_subpic_treated_as_pic_flag[i]和sps_loop_filter_across_subpic_enabled_flag[i]在序列参数集中指示。在方法1300的一些实施例中,第二规则指定使用当前子图片层信令通知。在方法1300的一些实施例中,规则指定当sps_pic_width_max_in_luma_samples小于或等于CtbSizeY时,sps_uniform_subpic_width_minus1从序列参数集中被排除并且被推断为等于0。在方法1300的一些实施例中,规则指定当sps_pic_height_max_in_luma_samples小于或等于CtbSizeY时,将sps_uniform_subpic_height_minus1从序列参数集被排除并且被推断为等于0。

[0723] 在方法1300的一些实施例中,规则指定sps_uniform_subpic_width_minus1、sps_uniform_subpic_height_minus1、sps_subpic_ctu_top_left_x[i]、sps_subpic_ctu_top_left_y[i]、sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]是ue(v)编解码的。在方法1300的一些实施例中,规则指定:sps_uniform_subpic_width_minus1和sps_uniform_subpic_height_minus1分别以与sps_subpic_width_minus1[i]和sps_subpic_height_minus1[i]相同的长度进行u(v)编解码。在方法1300的一些实施例中,序列参数集包括:指示视频的所有子图片是否分别具有相同的宽度和高度的第一语法元素,指示所有子图片的共同宽度的第二语法元素,以及指示所有子图片的共同高度的第三语法元素。在方法1300的一些实施例中,当序列参数集中指示的子图片的数量减1大于零并且当序列参数集中的语法元素指示编解码层视频序列中的视频的第一视频图片的第i个子图片在排除环路滤波操作的解码过程中被视为视频的第二视频图片时,以下两个条件之一为真:(1)第一视频图片的第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片和第一视频图片对于以下各项中的每一项具有相同的值:以亮度样点为单位的图片宽度,以亮度样点为单位的图片高度,子图片的数量减1,以及对于0到子图片的数量减1,包

括端值,的范围内的每个*i*值,第*i*个子图片的左顶编解码树单元的水平位置、第*i*子图片的左顶编解码树单元CTU的垂直位置、第*i*个子图片的宽度减1、第*i*个子图片的高度减1、第*i*个子图片的语法元素、和第*i*个子图片的标识符值;和(2)第一参考图片列表或第二参考图片列表中的每个活动条目所参考的第二视频图片是层间参考图片,对于层间参考图片,子图片的数量减1的值等于零。

[0724] 图14是视频处理的一个示例方法1400的流程图。操作1402包括对于包括一个或多个视频图片的视频和视频的比特流之间的转换,响应于参考以亮度样点为单位的图片参数集的每个视频图片的宽度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片列的宽度减1等于以编解码树块为单位的视频图片的宽度减1的第一语法元素。操作1404包括基于确定执行转换。

[0725] 图15是视频处理的一个示例方法1500的流程图。操作1502包括对于包括一个或多个视频图片的视频和视频的比特流之间的转换,响应于参考以亮度样点为单位的图片参数集的每个视频图片的高度小于或等于亮度编解码树块尺寸,确定图片参数集中指示以编解码树块为单位的第*i*个片行的高度减1等于以编解码树块为单位的视频图片的高度减1的第一语法元素。操作1504包括基于确定执行转换。

[0726] 在(多个)方法900-1500的一些实施例中,执行转换包括将视频编码到比特流中。在(多个)方法900-1500的一些实施例中,执行转换包括从视频生成比特流,并且该方法还包括将比特流存储在非暂时性计算机可读记录介质中。在(多个)方法900-1500的一些实施例中,执行转换包括从比特流解码视频。

[0727] 在一些实施例中,一种视频解码装置,包括被配置为实现(多个)方法900-1500的一个或多个实施例所述的方法的处理器。在一些实施例中,一种视频编码装置,包括被配置为实现(多个)方法900-1500的一个或多个实施例所述的方法的处理器。在一些实施例中,一种其上存储有计算机指令的计算机程序产品,该指令在由处理器执行时使处理器实现(多个)方法900-1500的一个或多个实施例所述的方法。在一些实施例中,一种存储根据(多个)方法900-1500的一个或多个实施例所述的方法生成的比特流的非暂时性计算机可读存储介质。在一些实施例中,一种非暂时性计算机可读存储介质,存储使处理器实现(多个)方法900-1500的一个或多个实施例所述的方法的指令。在一些实施例中,一种比特流生成的方法,包括:根据(多个)方法900-1500的一个或多个实施例所述的方法生成视频的比特流,以及将比特流存储在计算机可读程序介质上。在一些实施例中,一种方法、装置、根据本文档中描述的公开方法或系统生成的比特流。

[0728] 在本文档中,术语“视频处理”可指视频编码、视频解码、视频压缩或视频解压缩。例如,在从视频的像素表示到对应的比特流表示的转换期间,可以应用视频压缩算法,反之亦然。如语法所定义,当前视频块的比特流表示可(例如)对应于共位或散布在比特流内不同位置的比特。例如,可以根据变换和编解码的误差残差值并且还使用头中的比特和比特流中的其他字段,对宏块进行编码。此外,在转换期间,解码器可以基于该确定,在知道一些字段可能存在或不存在的条件下解析比特流,如以上解决方案中所述。类似地,编码器可确定包括或不包括某些语法字段,并通过从编解码表示中包括或排除语法字段来相应地生成编解码表示。

[0729] 本文档中所述的公开和其他方案、示例、实施例、模块和功能操作可以被实现在数

字电子电路中或者在计算机软件、固件或硬件中,含有本文档中所公开的结构以及其结构的等同物,或者它们中的一个或多个的组合。所公开的和和其他实施例可以被实现为计算机可读介质上所编码的一个或多个计算机程序产品,例如一个或多个计算机程序指令模块,用于由数据处理装置执行或者控制数据处理装置的操作。该计算机可读介质可以是机器可读存储设备、机器可读存储基板、存储器设备、影响机器可读可传播信号的复合物,或其一个或多个的组合。术语“数据处理装置”涵盖用于处理数据的所有装置、设备和机器,包括例如可编程处理器、计算机或多个处理器或计算机。除了硬件之外,该装置还可以包括为所讨论的计算机程序创建执行环境的代码,例如,构成处理器固件、协议栈、数据库管理系统、操作系统、或者它们中的一个或多个的组的代码。传播信号是人工生成的信号,例如,机器生成的电、光或电磁信号,其被生成来编码信息以传输到合适的接收器装置。

[0730] 计算机程序(也称为程序、软件、软件应用、脚本或代码)可以用任何形式的编程语言编写,包括编译或解释语言,并且可以以任何形式来部署计算机程序,包括独立程序或适合在计算环境中使用的模块、组件、子例程或其它单元。计算机程序不必对应于文件系统中的文件。程序可以存储在保存其他程序或数据的文件的部分中(例如,在标记语言文档中存储的一个或多个脚本)、在专用于所讨论的程序的单个文件中、或在多个协同文件中(例如存储一个或多个模块、子程序或代码部分的文件)。计算机程序可以部署为在一个计算机上或者在多个计算机上执行,该多个计算机位于一个站点处或者分布跨多个站点并由通信网络互连。

[0731] 可以由执行一个或多个计算机程序的一个或多个可编程处理器来进行在本文档中所描述的过程和逻辑流,以通过在输入数据上操作并且生成输出来进行功能。也可以由专用逻辑电路(例如,现场可编程门阵列(FPGA)或专用集成电路(ASIC))进行过程和逻辑流,并且装置可以实现为专用逻辑电路(例如FPGA或ASIC)。

[0732] 适合于计算机程序的执行的处理器包括例如通用和专用微处理器两者,以及任何类型的数字计算机的任何一个或多个处理器。通常,处理器将从只读存储器或随机存取存储器或者该两者接收指令和数据。计算机的基本元件是用于进行指令的处理器和用于存储指令和数据的一个或多个存储器设备。通常,计算机还将包括用于存储数据的一个或多个海量存储设备(例如磁、磁光盘或光盘),或者可操作地耦合以从海量存储设备(例如磁、磁光盘或光盘)接收数据或者将数据传输到海量存储设备(例如磁、磁光盘或光盘),或者以上两者。但是,计算机不必具有此类设备。适用于存储计算机程序指令和数据的计算机可读介质含有所有形式的非易失性存储器、介质和存储器设备,含有例如半导体存储器设备(例如EPROM、EEPROM和闪速存储器设备);磁盘(例如内部硬盘或可移动磁盘);磁光盘;和CD ROM和DVD-ROM磁盘。处理器和存储器可以由专用逻辑电路补充,或者合并并在专用逻辑电路中。

[0733] 虽然本专利文档含有许多细节,但这些细节不应被解释为对任何主题或可要求保护的范围的限制,而是作为规定于特定技术的特定实施例的特征的描述。在本专利文档中,在单独的实施例的上下文中描述的某些特征也可以在单个实施例中组合实现。相反地,在单个实施例的上下文中所描述的各种特征还可以分别在多个实施例中来实现或者以各种合适的子组合来实现。此外,尽管特征可以如上文描述为以某些组合起作用并且甚至最初同样地要求,但是在某些情况下来自所要求保护的组合的一个或多个特征可以从组合中去除,并且所要求保护的组合可以针对子组合或子组合的变化。

[0734] 类似地,尽管在附图中以特定顺序描绘了操作,但这不应当理解为要求按所示的特定次序或顺序次序进行此类操作或者进行所有示出的操作,以实现期望的结果。此外,在本专利文档中描述的实施例中的各种系统组件的分开不应被理解为在所有实施例中都要求这种分开。

[0735] 仅描述了几个实现方式和示例,并且可以基于本专利文档中描述和示出的内容来作出其它实现方式、增强和变型。

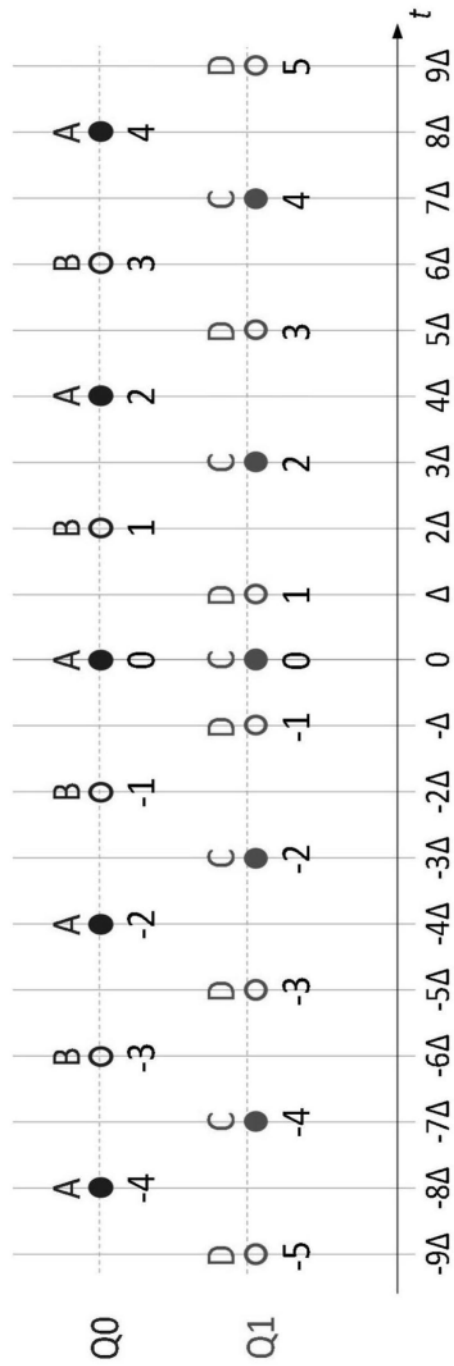
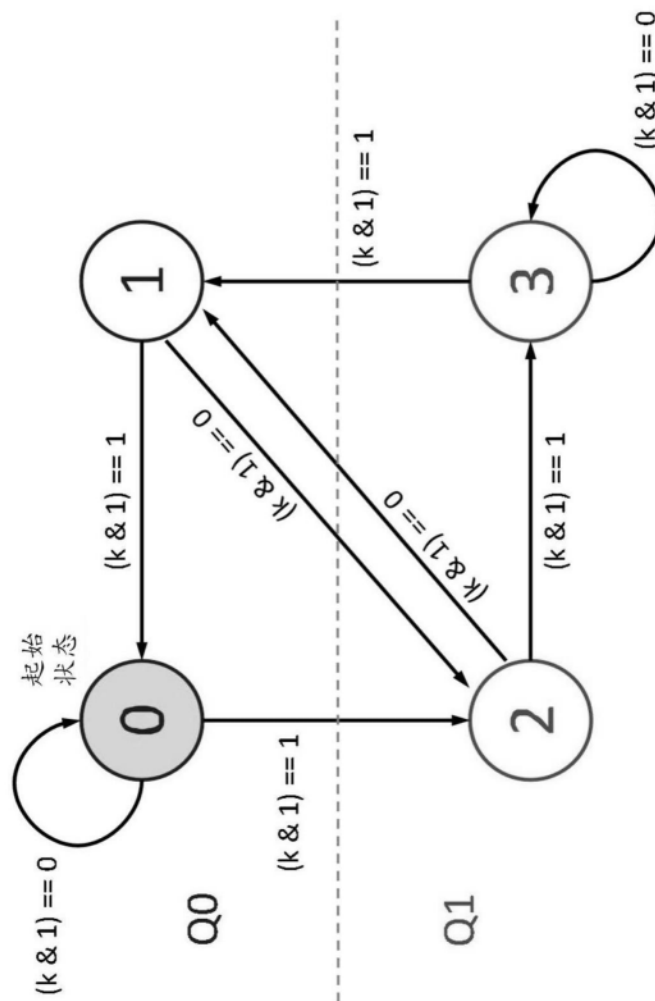


图1



当前状态	...的下一状态	
	$(k \& 1) == 0$	$(k \& 1) == 1$
0	0	2
1	2	0
2	1	3
3	3	1

图2

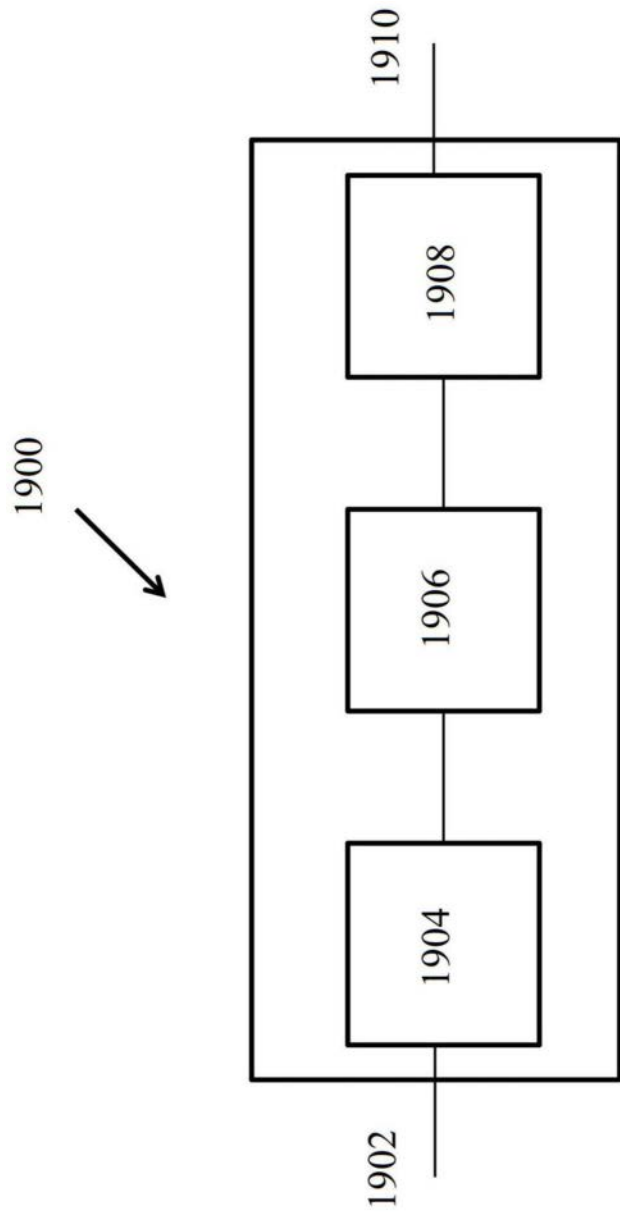


图3

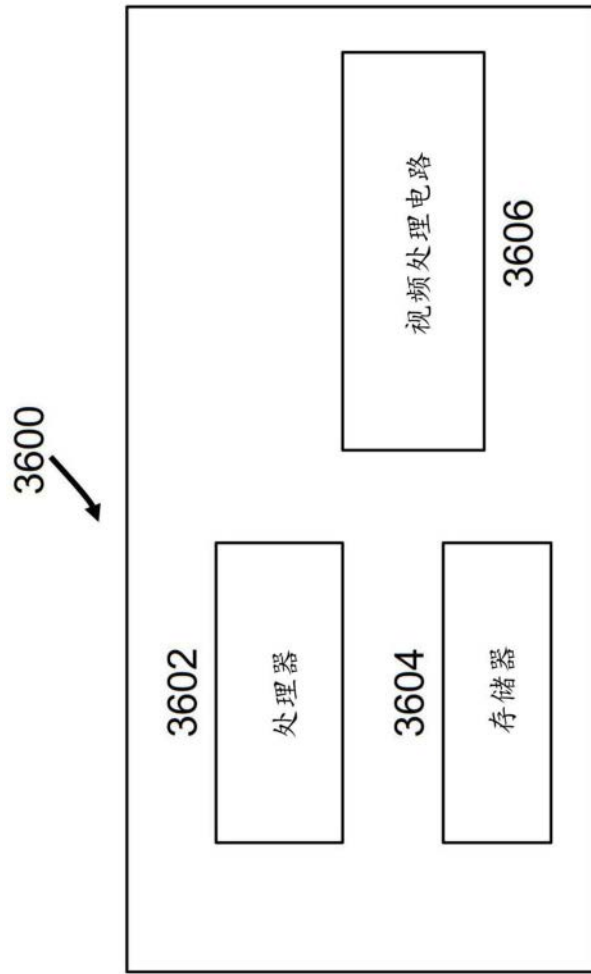


图4

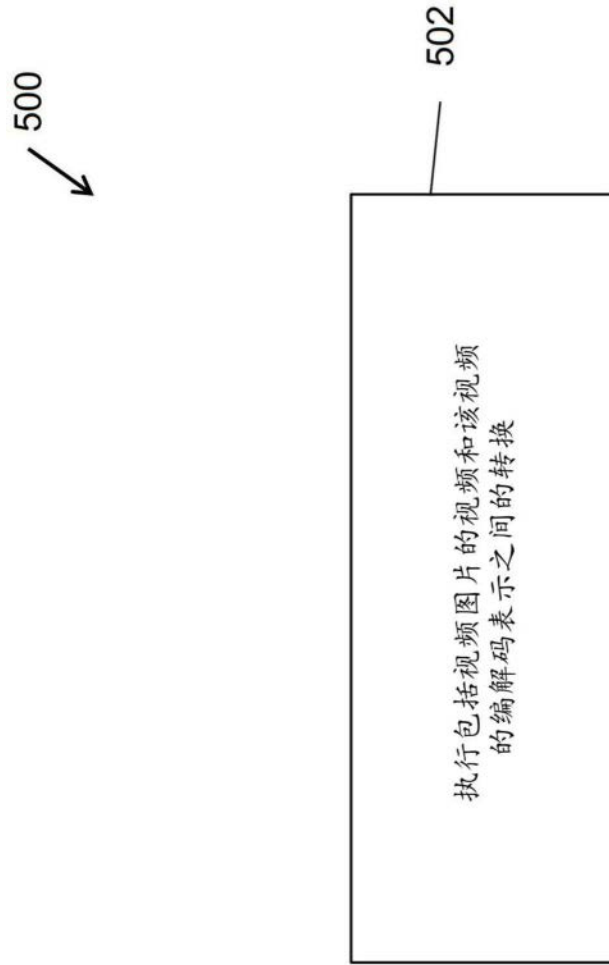


图5

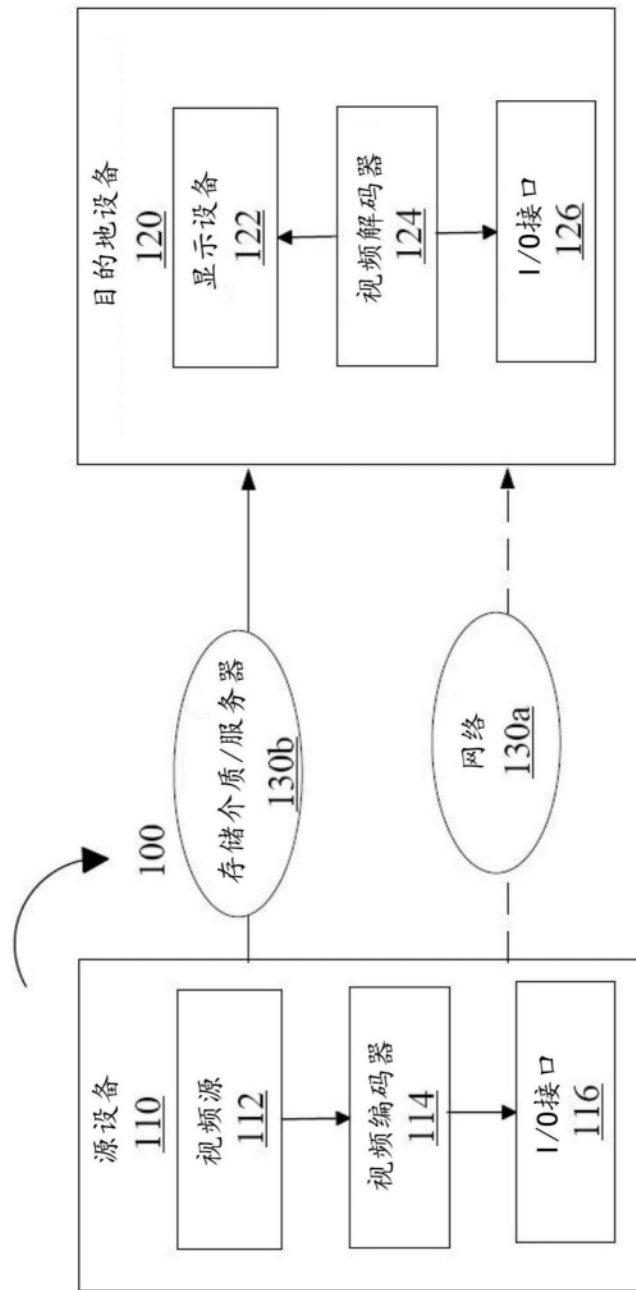


图6

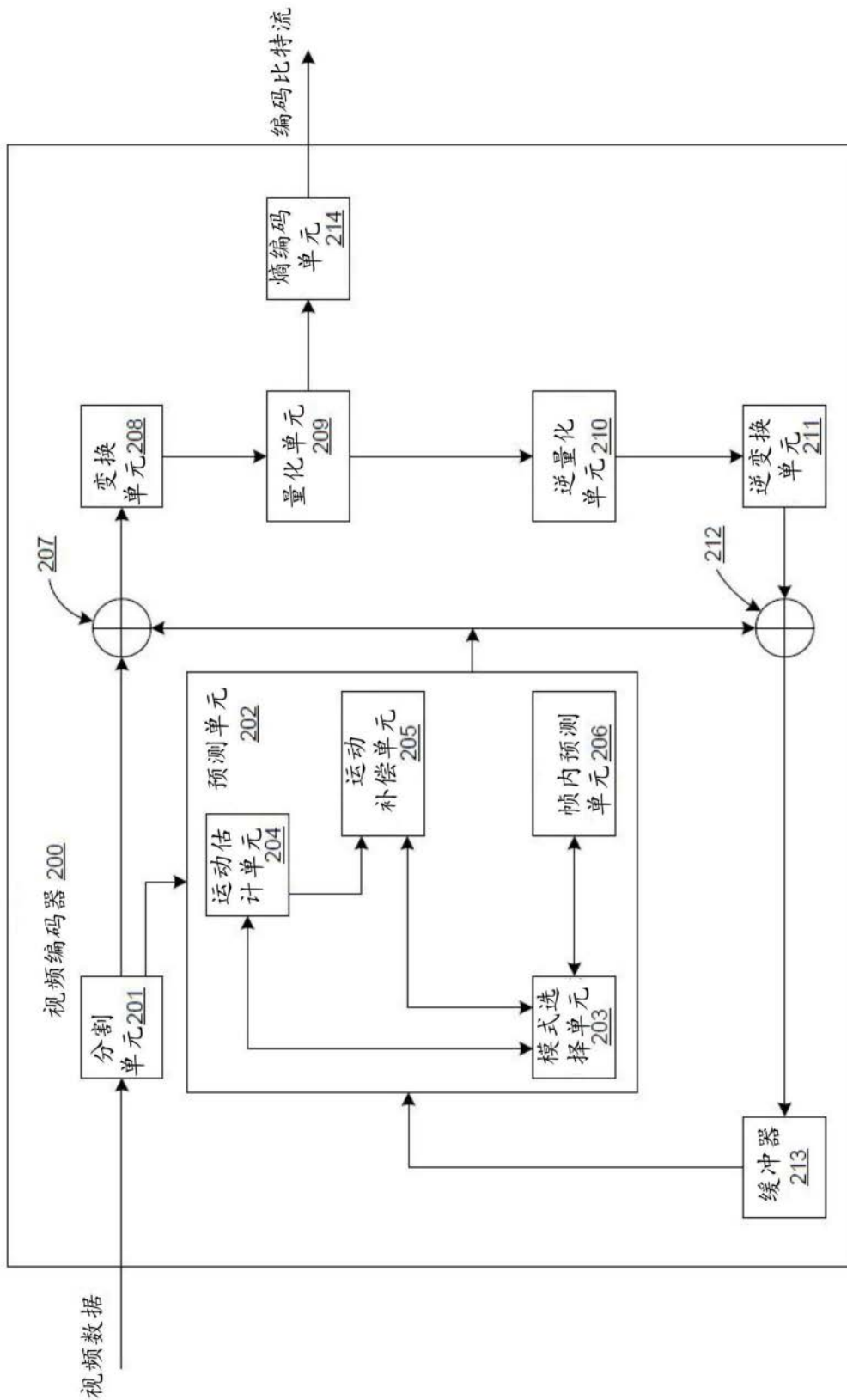


图7

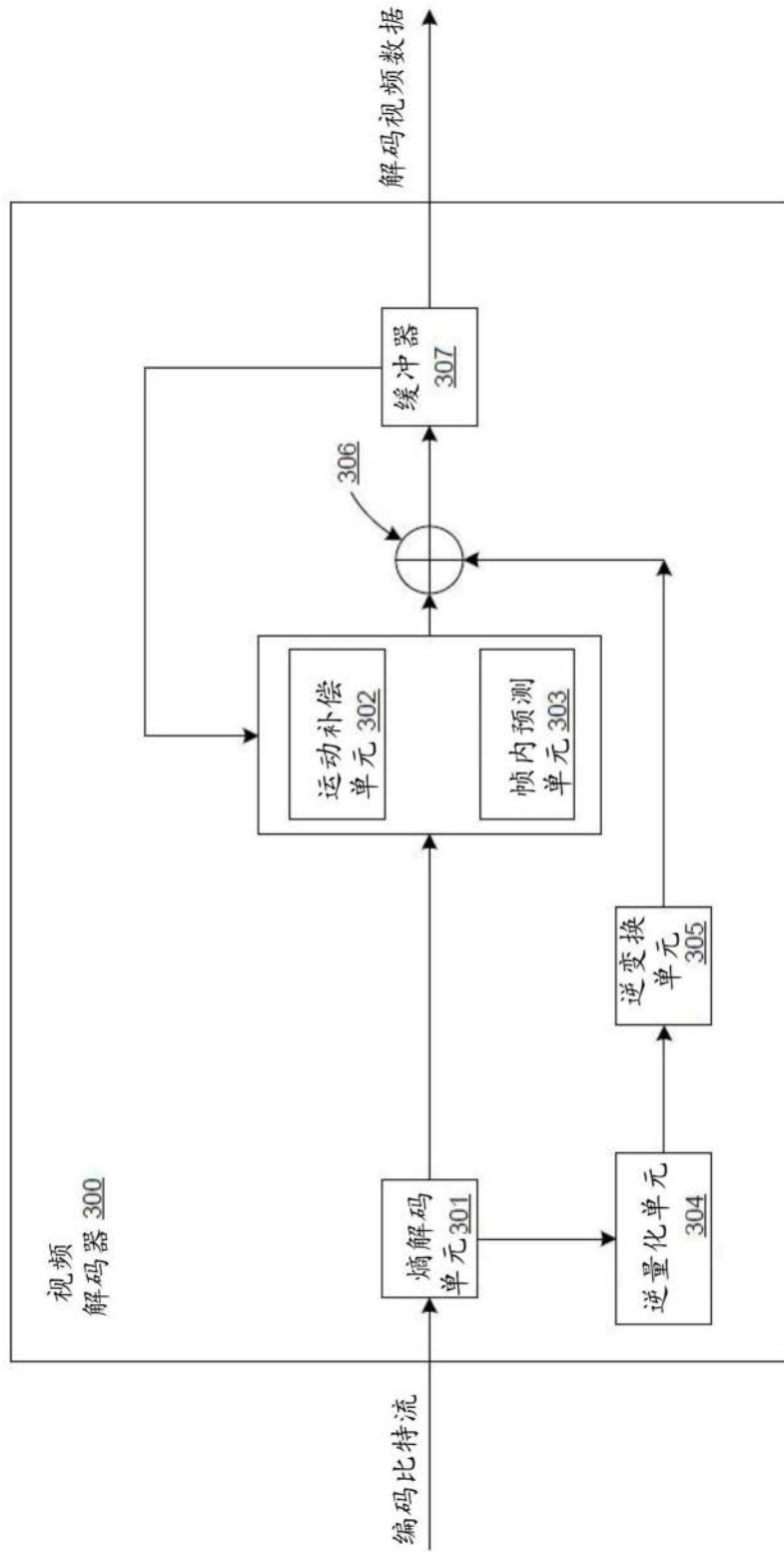


图8

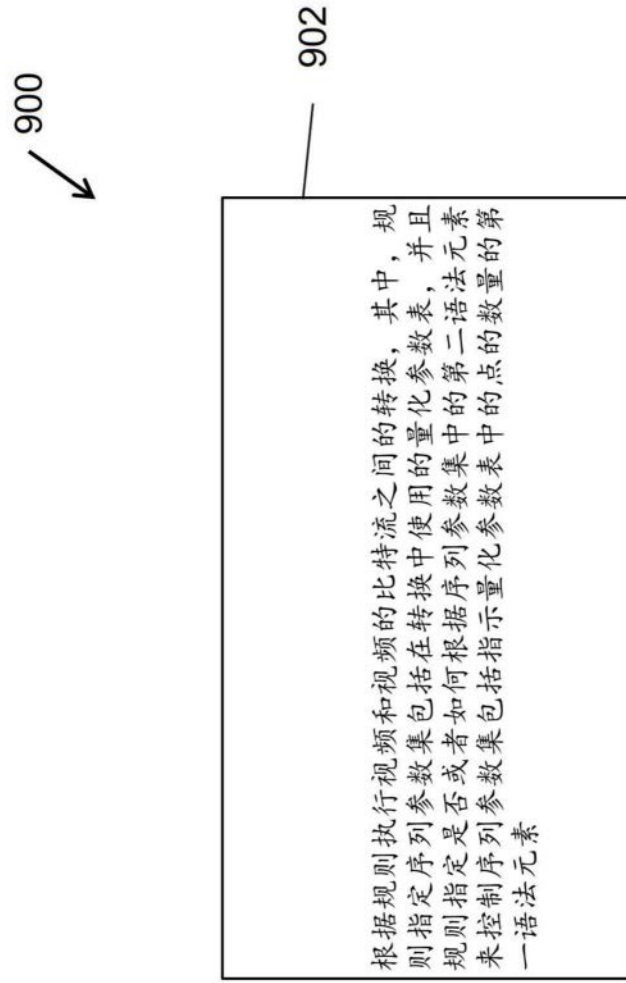


图9

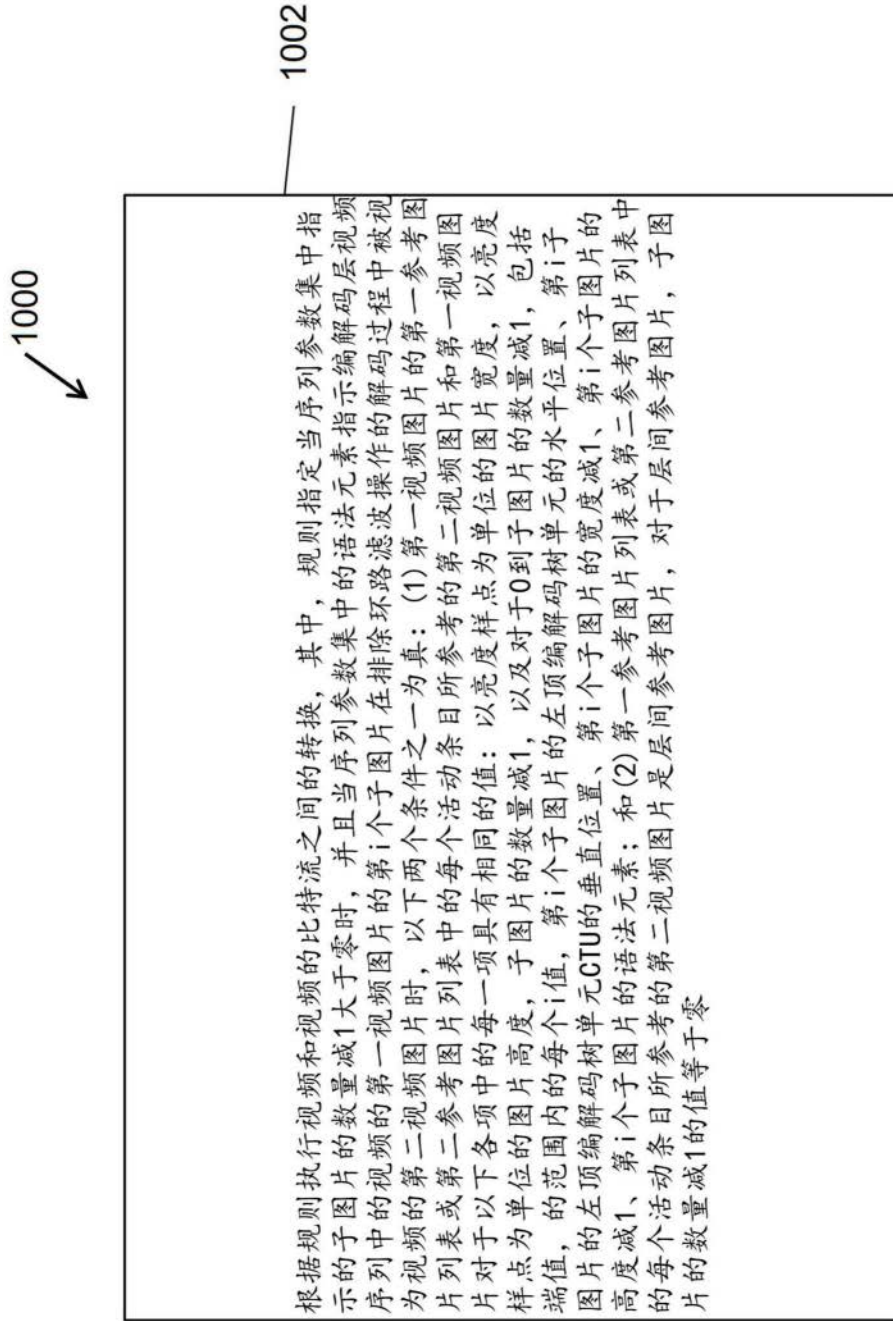


图10

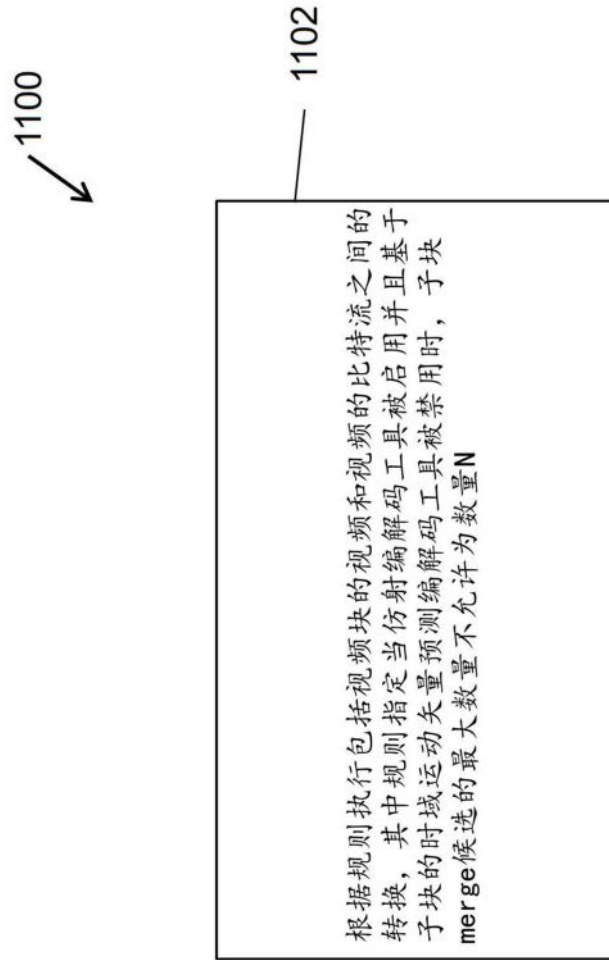


图11

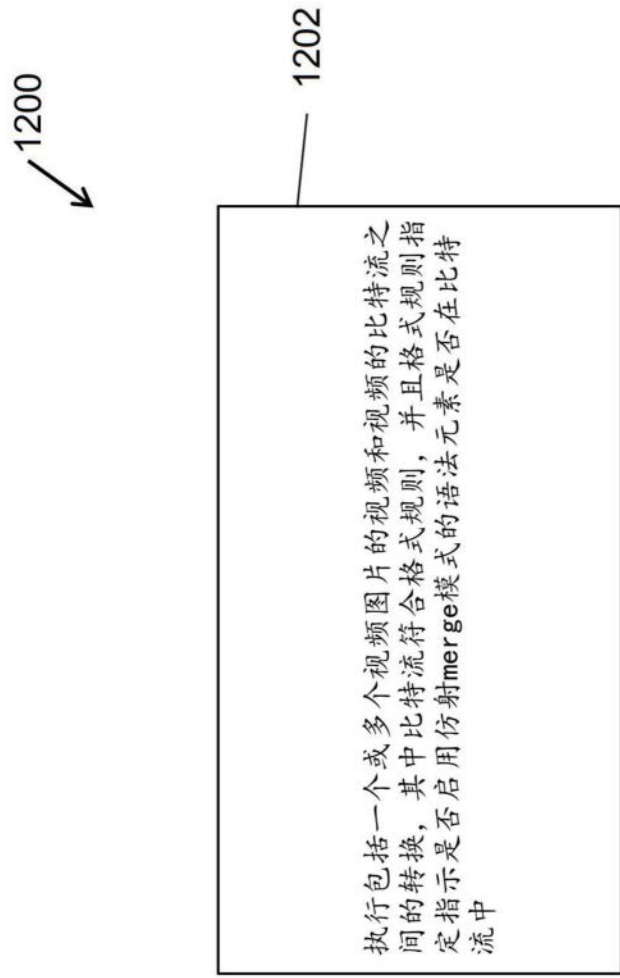


图12

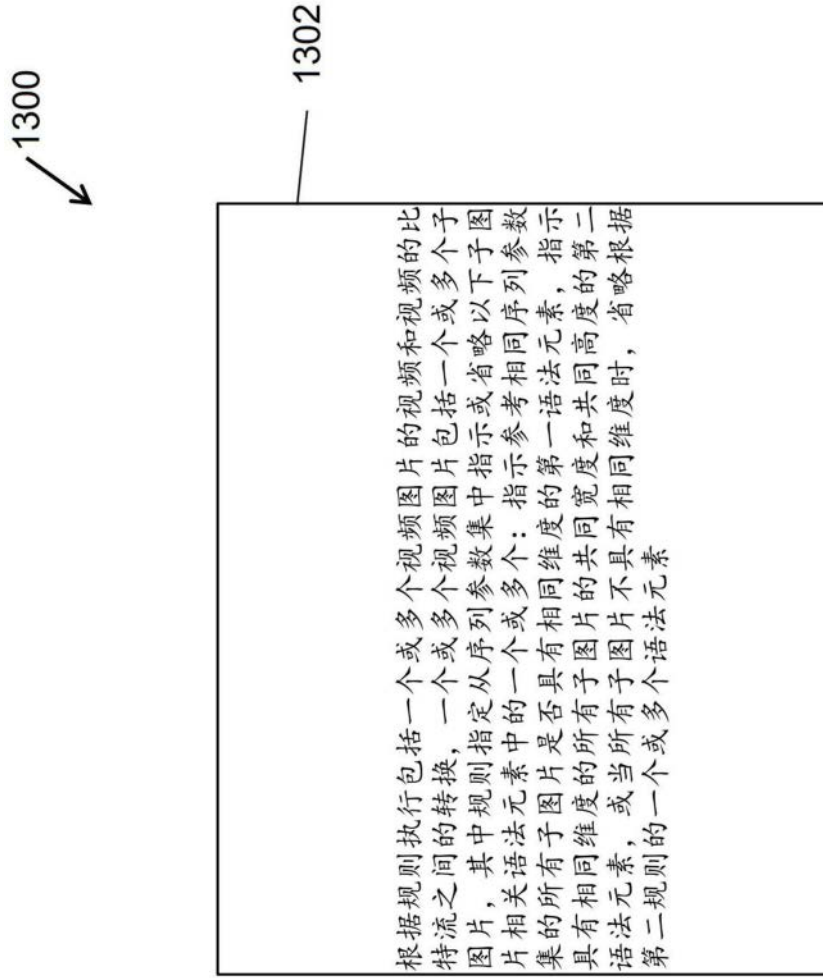


图13

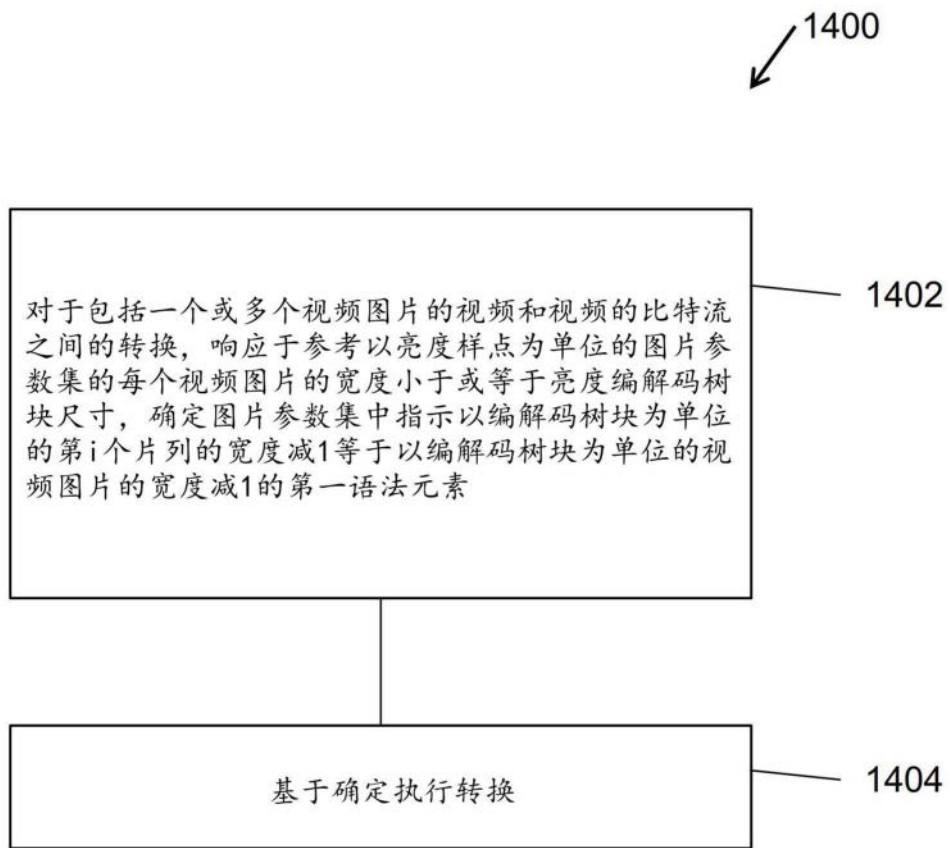


图14

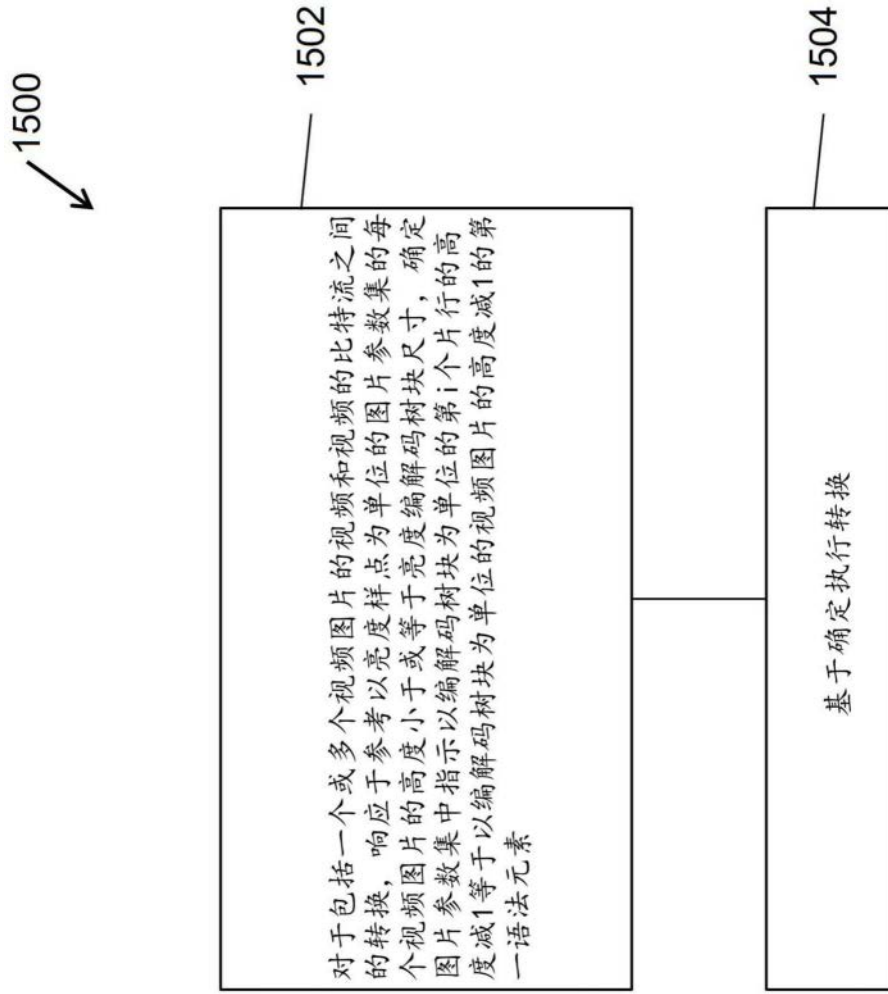


图15