(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101715652 B
(45)授权公告日 2013.02.13

(21)申请号 200880021842.8

(22)申请日 2008.06.25

(30)优先权数据
10-2007-0062513 2007.06.25 KR
10-2008-0014999 2008.02.19 KR

(85)PCT申请进入国家阶段日
2009.12.24

(86)PCT申请的申请数据
PCT/KR2008/003645 2008.06.25

(87)PCT申请的公布数据
WO2009/002092 EN 2008.12.31

(73)专利权人 三星电子株式会社
地址 韩国京畿道
专利权人 庆熙大学校产学协力团

(72)发明人 郑度泳 朴泰诚 吴润济 徐德荣 李英烈 朴光勋 金奎宪 朴慜祐 林成昶

(74)专利代理机构 北京铭硕知识产权代理有限公司 11286
代理人 韩明星 王艳娇

(51)Int.Cl.
H04N 7/32(2006.01)

(56)对比文件
WO 2006/033953 A1,2006.03.30, 全文.
US 2006/0023787 A1,2006.02.02, 全文.
WO 2007/047736 A2,2007.04.26, 全文.
Y-L LEE ET AL.CE11:Illumination compensation.《ISO/IEC JTC1/SC29/WG11 AND ITU-T SG16 Q6,NO.JVT-U052》.2006, 第1节、第2.2节,第3节以及图1、图2.

审查员 李婉怡

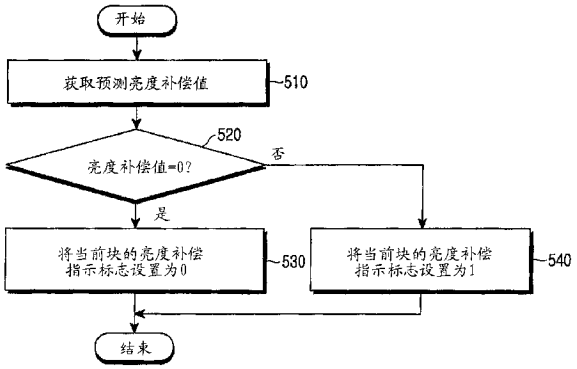
权利要求书 3 页 说明书 8 页 附图 11 页

(54)发明名称

用于多视点视频编码中的亮度补偿的方法和设备

(57)摘要

公开了一种在多视点视频编码中确定亮度补偿的方法。所述方法包括:确定执行亮度补偿的当前块何时处于B跳过模式,通过使用关于当前块的相邻块的信息来获取当前块的预测亮度补偿值;在获取的预测亮度补偿值还没有被确定时,将当前块设置为不执行亮度补偿;在获取的预测亮度补偿值已经被确定时,将当前块设置为执行亮度补偿。



CN 101715652 B

1. 一种在多视点视频编码方案中确定当前块的亮度补偿的方法，所述方法包括以下步骤：

通过使用关于处于 B 跳过模式中的当前块的相邻块的信息来获取当前块的预测亮度补偿值；

在当前块的获取的预测亮度补偿值无法被获取时，设置不执行亮度补偿的指示；

在当前块的获取的预测亮度补偿值已经被获取时，设置执行亮度补偿的指示。

2. 如权利要求 1 所述的方法，其中，关于当前块的相邻块的信息包括亮度补偿值、参考图像索引、块类型和指示相应的相邻块是否执行亮度补偿的标志值。

3. 如权利要求 1 所述的方法，其中，在当前块处于 B 跳过模式时，当前块的相邻块包括第一块、第二块、第三块和第四块，其中，第一块是当前块的上相邻块，第二块是当前块的左相邻块，第三块是第一块的右相邻块，第四块是第二块的上相邻块。

4. 如权利要求 3 所述的方法，其中，获取当前块的预测亮度补偿值的步骤包括以下步骤：

确定第一块是否使用亮度补偿，在第一块参照的图像与当前块参照的图像相同时，将第一块的亮度补偿值设置为当前块的预测亮度补偿值；

在第一块参照的图像与当前块参照的图像不相同时，确定第二块是否使用亮度补偿，在第二块参照的图像与当前块参照的图像相同时，将第二块的亮度补偿值设置为当前块的预测亮度补偿值；

在第二块参照的图像与当前块参照的图像不相同时，确定第三块是否使用亮度补偿，在第三块参照的图像与当前块参照的图像相同时，将第三块的亮度补偿值设置为当前块的预测亮度补偿值；

在第三块参照的图像与当前块参照的图像不相同时，确定第四块是否使用亮度补偿，在第四块参照的图像与当前块参照的图像相同时，将第四块的亮度补偿值设置为当前块的预测亮度补偿值；

在第四块参照的图像与当前块参照的图像不相同时，确定第一块、第二块和第三块是否全部使用亮度补偿，在第一块、第二块和第三块全部使用亮度补偿时，将第一块、第二块和第三块的亮度补偿值的中间值设置为当前块的预测亮度补偿值；以及

在第一块、第二块和第三块的任意一个不执行亮度补偿时，设置当前块的预测亮度补偿值以指示无亮度补偿。

5. 如权利要求 1 所述的方法，其中，所述将当前块设置为不执行亮度补偿的步骤对应于将指示当前块是否执行亮度补偿的标志值设置为第一值，所述将当前块设置为执行亮度补偿的步骤对应于将指示当前块是否执行亮度补偿的标志设置为第二值。

6. 一种在多视点视频编码中对在 B 跳过模式中执行亮度补偿的当前块进行亮度补偿的设备，所述设备包括：

预测亮度补偿值获取器，接收当前块的参考图像索引和关于当前块的相邻块的信息，获取和输出预测亮度补偿值；

标志设置单元，接收获取的预测亮度补偿值，基于获取的预测亮度补偿值设置和输出指示当前块是否执行亮度补偿的标志；

亮度补偿 / 运动补偿单元，通过使用运动矢量、参考图像、预测亮度补偿值和指示当前

块是否执行亮度补偿的标志来执行亮度补偿和运动补偿。

7. 如权利要求 6 所述的设备，其中，关于当前块的相邻块的信息包括亮度补偿值、参考图像索引、块类型和指示相应的相邻块是否执行亮度补偿的标志值。

8. 如权利要求 6 所述的设备，其中，当前块的相邻块包括第一块、第二块、第三块和第四块，其中，第一块是当前块的上相邻块，第二块是当前块的左相邻块，第三块是第一块的右相邻块，第四块是第二块的上相邻块。

9. 如权利要求 7 所述的设备，其中，预测亮度补偿值获取器执行以下步骤：

确定第一块是否使用亮度补偿，在第一块参照的图像与当前块参照的图像相同时，将第一块的亮度补偿值设置为当前块的预测亮度补偿值；

在第一块参照的图像与当前块参照的图像不相同时，确定第二块是否使用亮度补偿，在第二块参照的图像与当前块参照的图像相同时，将第二块的亮度补偿值设置为当前块的预测亮度补偿值；

在第二块参照的图像与当前块参照的图像不相同时，确定第三块是否使用亮度补偿，在第三块参照的图像与当前块参照的图像相同时，将第三块的亮度补偿值设置为当前块的预测亮度补偿值；

在第三块参照的图像与当前块参照的图像不相同时，确定第四块是否使用亮度补偿，在第四块参照的图像与当前块参照的图像相同时，将第四块的亮度补偿值设置为当前块的预测亮度补偿值；

在第四块参照的图像与当前块参照的图像不相同时，确定第一块、第二块和第三块是否全部使用亮度补偿，在第一块、第二块和第三块全部使用亮度补偿时，将第一块、第二块和第三块的亮度补偿值的中间值设置为当前块的预测亮度补偿值；以及

在第一块、第二块和第三块的任意一个不执行亮度补偿时，设置当前块的预测亮度补偿值以指示不执行亮度补偿。

10. 如权利要求 6 所述的设备，其中，当接收的预测亮度补偿值指示不执行时，标志设置单元将指示当前块是否执行亮度补偿的标志值设置为第一值，当接收的预测亮度补偿值指示执行亮度补偿时，将所述标志值设置为第二值。

11. 一种在多视点视频编码方案中确定当前块的亮度补偿的装置，所述装置包括：

通过使用关于处于 B 跳过模式中的当前块的相邻块的信息来获取当前块的预测亮度补偿值的模块；

在当前块的获取的预测亮度补偿值无法被获取时，设置不执行亮度补偿的指示的模块；

在当前块的获取的预测亮度补偿值已经被获取时，设置执行亮度补偿的指示的模块。

12. 如权利要求 11 所述的装置，其中，关于当前块的相邻块的信息包括亮度补偿值、参考图像索引、块类型和指示相应的相邻块是否执行亮度补偿的标志值。

13. 如权利要求 11 所述的装置，其中，在当前块处于 B 跳过模式时，当前块的相邻块包括第一块、第二块、第三块和第四块，第一块是当前块的上相邻块，第二块是当前块的左相邻块，第三块是第一块的右相邻块，第四块是第二块的上相邻块。

14. 如权利要求 13 所述的装置，其中，获取当前块的预测亮度补偿值的步骤包括以下步骤：

在第一块参照的图像与当前块参照的图像相同并且第一块使用亮度补偿时，将第一块的亮度补偿值设置为当前块的预测亮度补偿值；

在第二块参照的图像与当前块参照的图像相同并且第二块使用亮度补偿时，将第二块的亮度补偿值设置为当前块的预测亮度补偿值；

在第三块参照的图像与当前块参照的图像相同并且第三块使用亮度补偿时，将第三块的亮度补偿值设置为当前块的预测亮度补偿值；

在第四块参照的图像与当前块参照的图像相同并且第四块使用亮度补偿时，将第四块的亮度补偿值设置为当前块的预测亮度补偿值；

确定第一块、第二块和第三块是否全部使用亮度补偿，在第一块、第二块和第三块全部使用亮度补偿时，将第一块、第二块和第三块的亮度补偿值的中间值设置为当前块的预测亮度补偿值；以及

在第一块、第二块和第三块都不执行亮度补偿时，设置当前块的预测亮度补偿值以指示无亮度补偿。

15. 如权利要求 11 所述的装置，其中，设置当前块不执行亮度补偿与以下步骤对应：将指示当前块是否执行亮度补偿的标志值设置为第一值；设置当前块执行亮度补偿与以下步骤对应：将指示当前块是否执行亮度补偿的标志值设置为第二值。

用于多视点视频编码中的亮度补偿的方法和设备

技术领域

[0001] 本发明涉及多视点视频编码 (MVC)，更具体地说，涉及用于在多视点视频编码中的亮度补偿的处理期间获取预测的亮度补偿值和亮度补偿标志的方法和设备。

背景技术

[0002] 随着视频技术的发展，模拟图像处理方法正在被数字图像处理方法取代。结果，可以为用户提供比使用旧有技术拍摄的图像更清楚和具有更高图像质量的图像。图像质量的改进已经为用户提供了更强烈的存在感。最近，正在提供用于广播高质量图像的高清晰度 (HD) 电视 (TV) 业务，并且无论位置如何都发送逼真和清楚的图像的数字多媒体广播 (DMB) 业务的商业化也正在准备中。然而，仅改进图像本身的质量的高质量数字视频业务在改进用户得到的自由度和存在感方面受限。在图像的使用中降低存在感的代表性因素包括 :具有固定视点的相机和缺乏深度的图像。固定的相机视点降低了自由度，这是因为不允许用户在用户观看当前图像的同时观看除了当前图像之外的远景。此外，由于人们在观看物体时，他们的眼睛从不同的位置朝向相同的方向，因此，人们可通过两眼视差感受物体的深度。然而，由于不考虑双眼视差，通过固定相机获取的图像不能给予深度的感觉。为了解决此问题，正在开发通过位于多个视点的多个相机同时获取多视点图像的方法。

[0003] 如上所述的对多视点图像编码的方案被称为多视点视频编码 (MVC) 方案。MVC 的意思是用于处理通过多个相机获取的多视点的图像的序列的方案。根据 MVC，多个相机位于离相同的物体不同距离和 / 或它们朝向相同的物体。通过多个相机获取的多视点图像显示了来自相同物体的表面的光根据它们的方向而不同的反射度，而视点的图像高度相关。多视点图像的此特性导致相同物体的图像根据具体的视点而亮度和色彩不同。因此，为了改进多视点视频编码的编码效率，应当考虑多视点图像的上述特性。

[0004] 正在积极推动多视点视频编码的标准化，多视点视频编码正在寻求基于 H.264/MPEG-4 部分 10 先进视频编码 (H.264/AVC，是现有的用于视频编码的国际标准) 来考虑多视点图像的上述两个特性以改进编码效率的方法。例如，分层式 B- 图像编码是一种用于支持 H.264/AVC 中定义的联合可分级视频编码 (JSVC) 的时间可分级性的方法，分层式 B- 图像编码还应用于多视点视频编码内的视点内预测编码。此外，在多视点视频编码执行的同时，视点间预测编码也被并行执行，以提高编码效率。

[0005] 图 1 示出预测结构的示例，该预测结构显示了多视点视频编码中的视点内预测和视点间预测关系。在图 1 示出的示例中，存在八个视点，时间方向的图像组 (GOP) 的大小为 8。在图 1 中，S0、S1、S2、S3...S7 的每一个指示一个视点，T0、T1、T2、T3...T100 指示时间方向。

[0006] 参照图 1，注意到在每个视点中通过使用 H.264/AVC 中定义的分层式 B 图像编码执行时间方向上的预测编码。此外，在每个视点中，第一时间带 T0 的图像和离第一时间带 T0 的图像每隔 8 帧 (其为 GOP 的大小) 的图像 (位于 T8、T16、T24...) 被称为“锚图像”。对于锚图像，仅执行从相邻视点的预测 (即，视点间预测)。例如，在时间带 T0、T8、T16、

T24....，从相同时间带的 S0 视点对 S2 视点执行预测，从相同时间带的 S0 视点和 S2 视点对 S1 视点执行预测，从相同时间带的 S2 视点对 S4 视点执行预测，从相同时间带的 S2 视点和 S4 视点对 S3 视点执行预测，从相同时间带的 S4 视点对 S6 视点执行预测，从相同时间带的 S4 视点和 S6 视点对 S5 视点执行预测，从相同时间带的 S6 视点对 S7 视点执行预测。在 S7 视点的情况下，由于是最后的视点，因此其预测从 S6 视点执行。

[0007] 对于非锚图像，基本执行时间方向的预测，每隔两个视点（即，S1、S3、S5 和 S7）额外执行从相邻视点的预测。也就是说，不仅执行时间方向的视点内预测，还从 S0 视点和 S2 视点对 S1 视点执行视点间预测，从 S2 视点和 S4 视点对 S3 执行视点间预测，从 S4 视点和 S6 视点对 S5 视点执行视点间预测。

[0008] 多视点视频编码的国际标准目前正在进行中，多视点视频编码采用亮度补偿方法以便通过补偿存在于时间方向的亮度信号之间的差或者视点之间的差来提高编码效率。

[0009] 亮度补偿相关的数据包括：标志信息，指示当前块是否使用亮度补偿；亮度补偿值，包括 IC 偏移量或亮度改变差值 (DVIC，当前块和参考块之间的亮度信号平均差）；以及关于是否将执行亮度补偿和关于从相邻块得到的预测亮度补偿的信息。

发明内容

[0010] 因此，本发明提供了一种用于在多视点视频编码中有效地预测指示是否执行亮度补偿的标志和预测亮度补偿值的方法和设备。

[0011] 根据本发明的一方面，提供了一种在多视点视频编码中进行亮度补偿的方法。所述方法包括以下步骤：确定执行亮度补偿的当前块何时处于 B 跳过模式，通过使用关于当前块的相邻块的信息来获取当前块的预测亮度补偿值；在获取的预测亮度补偿值是 0 时，将当前块设置为不执行亮度补偿；在获取的预测亮度补偿值不是 0 时，将当前块设置为执行亮度补偿。

[0012] 在当前块处于 B 跳过模式时，当前块的相邻块包括第一块、第二块、第三块和第四块，第一块是当前块的上相邻块，第二块是当前块的左相邻块，第三块是第一块的右相邻块，第四块是第二块的上相邻块。

[0013] 获取当前块的预测亮度补偿值的步骤包括以下步骤：确定作为当前块的上相邻块的第一块是否使用亮度补偿以及第一块参照的图像与当前块参照的图像是否相同，在第一块参照的图像与当前块参照的图像相同时，将第一块的亮度补偿值设置为当前块的预测亮度补偿值；在第一块参照的图像与当前块参照的图像不相同时，确定作为当前块的左相邻块的第二块是否使用亮度补偿以及第二块参照的图像与当前块参照的图像是否相同，在第二块参照的图像与当前块参照的图像相同时，将第二块的亮度补偿值设置为当前块的预测亮度补偿值；在第二块参照的图像与当前块参照的图像不相同时，确定作为第一块的右相邻块的第三块是否使用亮度补偿以及第三块参照的图像与当前块参照的图像是否相同，在第三块参照的图像与当前块参照的图像相同时，将第三块的亮度补偿值设置为当前块的预测亮度补偿值；在第三块参照的图像与当前块参照的图像不相同时，确定作为第二块的上相邻块的第四块是否使用亮度补偿以及第四块参照的图像与当前块参照的图像是否相同，在第四块参照的图像与当前块参照的图像相同时，将第四块的亮度补偿值设置为当前块的预测亮度补偿值；在第四块参照的图像与当前块参照的图像不相同时，确定第一块、第二块

和第三块是否全部使用亮度补偿，在第一块、第二块和第三块全部使用亮度补偿时，将第一块、第二块和第三块的亮度补偿值的中间值设置为当前块的预测亮度补偿值；在第一块、第二块和第三块的任意一个不执行亮度补偿时，将当前块的预测亮度补偿值设置为 0。

[0014] 根据本发明的另一方面，提供了一种在多视点视频编码中当执行亮度补偿的当前块处于 B 跳过模式中时用于亮度补偿的设备，所述设备包括：预测亮度补偿值获取器，接收当前块的参考图像索引和关于当前块的相邻块的信息，获取和输出预测亮度补偿值；标志设置单元，接收获取的预测亮度补偿值，基于获取的预测亮度补偿值设置和输出指示当前块是否执行亮度补偿的标志；亮度补偿 / 运动补偿单元，通过使用运动矢量、参考图像、预测亮度补偿值和指示当前块是否执行亮度补偿的标志来执行亮度补偿和运动补偿。

[0015] 当接收的预测亮度补偿值不是 0 时，标志设置单元将指示当前块是否执行亮度补偿的标志值设置为 0，当接收的预测亮度补偿值是 0 时，将所述标志值设置为 1。

附图说明

[0016] 通过下面结合附图的详细描述，本发明的上述和其他方面、特征和优点将会变得更清楚，其中：

[0017] 图 1 示出了显示在多视点视频编码中传统的视点内预测和视点间预测关系的预测结构的示例；

[0018] 图 2 示出在多视点视频编码中为了获取预测亮度补偿值而使用的相邻块；

[0019] 图 3 是在发送亮度补偿指示标志和亮度补偿值差的情况下从相邻块获取预测亮度补偿值的处理的流程图；

[0020] 图 4 是从 P 跳过模式块获取预测亮度补偿值和亮度补偿指示标志的处理的流程图；

[0021] 图 5 是根据本发明实施例的在发送亮度补偿指示标志和亮度补偿值差的情况下预测亮度补偿指示标志的处理的流程图；

[0022] 图 6A、图 6B 和图 6C 示出实际实施从处于 B 跳过模式的块预测亮度补偿指示标志的现有处理的程序的示例；

[0023] 图 7 示出根据本发明实施例的实际实施从处于 B 跳过模式的块预测亮度补偿指示标志的处理的程序的示例；

[0024] 图 8 是示出根据本发明实施例的用于从跳过模式块获取预测亮度补偿值和亮度补偿指示标志的装置的框图；

[0025] 图 9 是显示现有亮度补偿方法和根据本发明实施例的亮度补偿方法用于 Racel VGA 30Hz 图像序列的比较的曲线图；

[0026] 图 10 是显示现有亮度补偿方法和根据本发明实施例的亮度补偿方法用于 Ballroom VGA 25Hz 图像序列的比较的曲线图。

具体实施方式

[0027] 将参照附图描述本发明的示例性实施例。在以下描述中，提供在以下描述中找到的各种特定定义仅用以帮助对于本发明的总体理解，对于本领域的技术人员很明显，可不使用这样的定义来实施本发明。此外，在本发明的以下描述中，当可能使得本发明的主题不

清楚时，将省略在此引入的已知功能和配置的详细描述。

[0028] 图 2 示出在多视点视频编码中为了获取预测亮度补偿值而使用的相邻块。在确定是否执行亮度补偿和从相邻块获取预测亮度补偿值的情况下，如图 2 所示的当前块周围的相邻块 A、B、C 和 D 用作相邻块。在发送亮度补偿值的情况下，从相邻块获取预测亮度补偿值，并随后发送亮度值差（即，亮度补偿值减去预测亮度补偿值）。

[0029] 获取关于是否执行亮度补偿的信息和预测亮度补偿值的方法根据块的类型而不同，块的类型包括三种。第一块类型对应于这样的情况：发送指示是否执行亮度补偿的标志和亮度补偿值差（帧间 16×16 模式，B 直接 16×16 模式），第二块类型对应于 P 跳过模式，第三块类型对应于 B 跳过模式。

[0030] 首先，将参照图 3 描述在发送标志和亮度补偿值差（帧间 16×16 模式，B 直接 16×16 模式）的情况下，从与当前块邻接的相邻块（图 2 的块 A、B、C 和 D）获取预测亮度补偿的方法。

[0031] 图 3 是在发送亮度补偿指示标志和亮度补偿值差的情况下从相邻块获取预测亮度补偿的处理的流程图。

[0032] 参照图 3，首先在步骤 305，确定作为当前块的上相邻块的块 A 是否使用亮度补偿，以及当前块所参照的图像是否等同于块 A 所参照的图像。作为在步骤 305 的确定的结果，当块 A 使用亮度补偿并且当前块和块 A 参照相同的图像时，块 A 的亮度补偿值用作当前块的预测亮度补偿值（步骤 330）。随后，终止处理。

[0033] 作为在步骤 305 确定的结果，当答案是“否”时，确定作为当前块的左相邻块的块 B 是否使用亮度补偿，以及当前块和块 B 是否参照相同的图像（步骤 310）。作为在步骤 310 确定的结果，当块 B 使用亮度补偿并且当前块和块 B 参照相同的图像时，块 B 的亮度补偿值用作当前块的预测亮度补偿值（步骤 335）。随后，终止处理。

[0034] 作为在步骤 310 确定的结果，当答案是“否”时，确定作为当前块的右相邻块的块 C 是否使用亮度补偿，以及当前块和块 C 是否参照相同的图像（步骤 315）。作为在步骤 315 确定的结果，当块 C 使用亮度补偿并且当前块和块 C 参照相同的图像时，块 C 的亮度补偿值用作当前块的预测亮度补偿值（步骤 340）。随后，终止处理。

[0035] 作为在步骤 315 确定的结果，当答案是“否”时，确定作为块 B 的上相邻块的块 D 是否使用亮度补偿，以及当前块和块 D 是否参照相同的图像（步骤 320）。作为在步骤 320 确定的结果，当块 D 使用亮度补偿并且当前块和块 D 参照相同的图像时，块 D 的亮度补偿值用作当前块的预测亮度补偿值（步骤 345）。随后，终止处理。

[0036] 作为在步骤 320 确定的结果，当答案是“否”时，确定块 A、B 和 C 的每一个是否使用亮度补偿（步骤 325）。作为在步骤 325 确定的结果，当块 A、B 和 C 的每一个都使用亮度补偿时，将块 A、B 和 C 的亮度补偿值的中间值用作当前块的预测亮度补偿值（步骤 350）。随后，终止处理。

[0037] 作为在步骤 325 确定的结果，当答案是“否”时，当前块的预测亮度补偿值被设置为“0”（步骤 355）。随后，终止处理。

[0038] 在步骤 350 设置的中间值对应于这样的值：从一组数字中选择该值，使得该组的一半的数字大于该中间值而另一半小于该中间值。如果该组包括偶数个数字，则将两个中值的平均确定为中间值。

[0039] 在对应于第二块类型的 P 跳过模式块的情况下，从位于当前块周围的块 A 和 B 获取的预测亮度补偿值用作亮度补偿值。以下将参照图 4 描述在 P 跳过模式中获取预测亮度补偿值的方法。

[0040] 图 4 是从 P 跳过模式块获取预测亮度补偿值和亮度补偿指示标志的处理的流程图。参照图 4，在步骤 410，确定作为当前块的上相邻块的块 A 和作为当前块的左相邻块的块 B 是否都使用亮度补偿。作为在步骤 410 确定的结果，当块 A 和块 B 都使用亮度补偿时，确定块 A 和块 B 的亮度补偿值的平均值并将其用作当前块的预测亮度补偿值，并将指示当前块使用亮度补偿的标志设置为“1”（步骤 440）。随后，终止处理。

[0041] 作为在步骤 410 确定的结果，当块 A 或块 B 不使用亮度补偿时，确定是否仅块 A 使用亮度补偿而块 B 不使用亮度补偿（步骤 420）。作为在步骤 420 确定的结果，当仅块 A 使用亮度补偿时，将块 A 的亮度补偿值用作当前块的预测亮度补偿值，并将指示当前块使用亮度补偿的标志设置为“1”（步骤 450）。随后，处理终止。

[0042] 作为在步骤 420 确定的结果，当块 A 不使用亮度补偿时，确定是否仅块 B 使用亮度补偿而块 A 不使用亮度补偿（步骤 430）。作为在步骤 430 确定的结果，当仅块 B 使用亮度补偿时，将块 B 的亮度补偿值用作当前块的预测亮度补偿值，并将指示当前块使用亮度补偿的标志设置为“1”（步骤 460）。随后，处理终止。

[0043] 作为在步骤 430 确定的结果，当块 B 也不使用亮度补偿时，将当前块的亮度补偿值设置为“0”并将指示当前块使用亮度补偿的标志设置为“0”（步骤 470）。随后，处理终止。

[0044] 在 B 跳过模式块的情况下，从当前块的相邻块 A、B 和 C 获取关于当前块是否使用亮度补偿的标志信息。

[0045] 当获取的亮度补偿指示标志为“1”时，按原样使用获取预测亮度值的方法（如图 3 所示），该方法用于与发送亮度补偿指示标志和亮度补偿值差（帧间 16×16 模式，B 直接 16×16 模式）的情况相应的块类型中。此外，当获取的亮度补偿指示标志是“0”时，亮度补偿值被设置为“0”。

[0046] 基本上，在获取亮度补偿指示标志中使用块 A、B 和 C。当块 C 不可用时，使用块 D 代替块 C。以下将描述 B 跳过模式的获取亮度补偿指示标志的方法。

[0047] 首先，当当前块周围的块 A、B 和 C 中的一个块的参考图像等同于当前块的参考图像时，使用具有与当前块相同的参考图像的块的亮度补偿指示标志作为当前块的亮度补偿指示标志。然而，当当前块周围的块 A、B 和 C 中的一个块的参考图像不等于当前块的参考图像时，使用块 A、B 和 C 的亮度补偿指示标志的中间值作为当前块的亮度补偿指示标志。

[0048] 在包括在当前国际多视点视频编码标准中的获取预测亮度补偿值的方法中的 B 跳过模式的情况中，所述操作被划分为两个部分。一个部分是预测指示是否执行亮度补偿的标志值的操作，另一个部分是获取预测亮度补偿值的操作。然而，在现有方法中，这两个部分不能执行一致的预测。

[0049] 在 B 跳过模式中，块 A、B 和 C（仅当块 C 不可用时使用块 D）主要用于预测亮度补偿指示值，全部块 A、B、C 和 D 用于获取预测亮度补偿值。

[0050] 此外，在预测亮度补偿指示值时，当全部块 A、B、C 和 D 参照与当前块参照的图像不同的图像并且块 A、B 和 C 中的两个块使用亮度补偿时，“1”被预测为亮度补偿指示值。然而，在获取预测亮度补偿值的方法中，0 被获取作为预测亮度补偿值。

[0051] 如上所述的获取亮度补偿指示标志和预测亮度补偿值的现有方法可产生这样一种情况：当亮度补偿指示标志是 1 时，预测亮度补偿值是 0。这种情况可对预测具有持续影响，这会降低整个编码能力。

[0052] 本发明提出了一种用于预测的方法，该方法以简单和恒定的方式在多视点视频编码的亮度补偿处理的跳过模式中获取预测亮度补偿值和亮度补偿指示标志，提出了一种在发送亮度补偿指示标志和亮度补偿值差的块模式中用于有效发送亮度补偿指示标志和预测亮度补偿值的信息的方法。

[0053] 用于 B 跳过模式的亮度补偿的方法

[0054] 现将描述根据本发明实施例的在 B 跳过模式中预测亮度补偿指示标志和亮度补偿值的方法。

[0055] 首先，通过使用在 B 跳过模式中预测亮度补偿指示标志和亮度补偿值的现有方法来获取预测亮度补偿值。随后，基于获取的预测亮度补偿值来确定亮度补偿指示标志。具体地，当获取的预测亮度补偿值是 0 时，亮度补偿指示标志被确定为 0，当获取的预测亮度补偿值不是 0 时，亮度补偿指示标志被确定为 1。

[0056] 图 5 是根据本发明实施例的从处于 B 跳过模式的块预测亮度补偿指示标志的处理的流程图。参照图 5，在根据本发明实施例的在 B 跳过模式中获取预测亮度补偿值和预测亮度补偿指示标志的处理中，在步骤 510 首先获取预测亮度补偿值。在这个步骤中，如上参照图 3 所述的在 B 跳过模式中获取预测亮度补偿值的方法（在发送标志和亮度补偿值差的块类型的情况下获取预测亮度补偿值的方法）用于获取预测亮度补偿值。随后，在步骤 520，通过使用在步骤 510 中获取的预测亮度补偿值，确定计算的亮度补偿值是否为“0”。作为步骤 520 中的确定结果，当计算的亮度补偿值是“0”时，当前块的亮度补偿指示标志被设置为“0”（步骤 530）。相比较的是，步骤 520 中的确定显示亮度补偿值不是“0”，则亮度补偿指示标志被设置为“1”（步骤 540）。

[0057] 图 6A、图 6B 和图 6C 示出实际实施从处于 B 跳过模式的块预测亮度补偿指示标志的现有处理的程序的示例。

[0058] 图 6A、图 6B 和图 6C 示出一个程序的示例，该程序由于每张纸的有限容量而被分为三页独立的纸上的三个部分，但是所述三个部分仅构成一个程序。参照图 6A、图 6B 和图 6C 中的加粗字体的强调部分，通过使用“getlcpFlagPredictor()”函数（图 6C 的第 29 到 38 行）首先从作为当前块的相邻块的块 A、块 B 和块 C 获取标志信息，随后基于获取的标志信息执行所述操作。

[0059] 图 7 示出根据本发明实施例的实施从处于 B 跳过模式的块预测亮度补偿指示标志的处理的程序的示例。参照图 7 中的加粗字体的强调部分，没有（通过使用图 6C 的“getlcpFlagPredictor()”函数）执行从当前块的相邻块（图 2 的块 A、块 B 和块 C）获取标志信息的步骤。取代的是，首先确定获取的亮度补偿值是否是“0”（“if(clcp.getPredOffset() == 0)”），基于根据本发明实施例的所述处理中确定的结果来设置标志值。在图 6A、图 6B 和图 6C 中示出的示例与图 7 中示出的示例之间的比较中，根据本发明实施例的标志预测方法不单独执行获取标志信息的步骤（“getlcpFlagPredictor()”函数）。因此，与现有方法相比，本发明的方法可通过更简单的操作实现标志信息预测部分。

[0060] B 跳过块的亮度补偿设备

[0061] 图 8 是示出根据本发明实施例的用于从跳过模式块获取预测亮度补偿值和亮度补偿指示标志的装置的框图。以下，将参照图 8 详细描述根据本发明实施例的用于从处于跳过模式的块获取预测亮度补偿值和亮度补偿指示标志的装置。

[0062] 根据本发明实施例的从处于跳过模式的块获取预测亮度补偿值和亮度补偿指示标志的设备包括预测亮度补偿值获取器 1210、标志设置单元 1220 和亮度补偿 / 运动补偿单元 1230。

[0063] 预测亮度补偿值获取器 1210 接收当前块的参考图像索引和当前块的相邻块（图 2 的块 A、B、C 和 D）的信息（亮度补偿指示标志、亮度补偿值、参考图像索引、块类型等），获取预测亮度补偿值，并输出预测亮度补偿值。

[0064] 正在进行国际标准化的多视点视频编码在 B 跳过模式中使用块 A、B、C 和 D，P 跳过模式中使用块 A 和块 B。具体地，B 跳过模式使用获取用于块模式的预测亮度补偿值的现有方法，所述块模式将亮度补偿指示标志和亮度补偿值差编码到比特流之后发送所述比特流，P 跳过模式使用预测亮度补偿值的现有方法。

[0065] 标志设置单元 1220 接收由预测亮度补偿值获取器 1210 获取的预测亮度补偿值，并根据本发明提出的方法设置和输出亮度补偿指示标志。

[0066] 在 B 跳过模式的情况下，根据获取用于块模式的预测亮度补偿值的现有方法来获取预测亮度补偿值，所述块模式将亮度补偿指示标志和亮度补偿值差编码到比特流之后发送所述比特流。随后，当获取的预测亮度补偿值是“0”时，亮度补偿指示标志被设置为“0”，当获取的预测亮度补偿值不是“0”时，亮度补偿指示标志被设置为“1”。

[0067] 亮度补偿 / 运动补偿单元 1230 通过执行亮度补偿和运动补偿来输出解码的图像，其中，通过使用亮度补偿指示标志、预测亮度补偿值、参考图像和运动矢量来执行所述亮度补偿和运动补偿。

[0068] 图 9 和图 10 是分别示出现有亮度补偿方法与根据本发明实施例的亮度补偿方法用于 Racel VGA 30Hz 图像序列和 Ballroom VGA 25Hz 图像序列的比较的曲线图。

[0069] 分别通过对于 Racel VGA 30Hz 图像序列和 Ballroom VGA 25Hz 图像序列改变 22、27、32 和 37 的量化参数 (QP) 来使用 JMVM 4.0 测试现有方法和本发明的方法。测试的结果在图 9 和图 10 中显示，表示现有方法和提出的方法实际上具有相似的性能。

[0070] 在根据本发明的多视点编码中用于亮度补偿的方法中，基于获取的预测亮度补偿值设置亮度补偿指示标志。因此，可以省略获取亮度补偿指示标志的操作，这可减少操作的数量。此外，本发明可防止每个块类型中的亮度补偿指示标志被同时获取为“1”的同时而预测亮度补偿被获取为“0”。因此，根据本发明，可以更有效地获取亮度补偿指示标志和预测亮度补偿值，这可提高编码效率。

[0071] 根据本发明的上述方法可以以硬件或软件或计算机代码实现，所述软件或计算机代码可以存储在诸如 CD-ROM、RAM、软盘、硬盘或磁光盘的记录介质中或可从网络下载，从而在此描述的方法可通过所述软件使用通用计算机或专用处理器或可编程或专用硬件（例如，ASIC 或 FPGA）执行。如本领域所知，计算机、处理器或可编程硬件包括可存储或接收软件或计算机代码的存储器部件，例如，RAM、ROM、闪存等，所述软件或计算机代码在被计算机处理器或硬件访问和执行时，实施在此描述的处理方法。

[0072] 虽然已经参照本发明的特定示例性实施例示出和描述了本发明，但是本领域的技

术人员将理解，在不脱离权利要求限定的本发明的精神和范围的情况下，可以在形式和细节上做出各种改变。例如，虽然参考被设置成值 1 和 0，应认识到这些值仅仅是作为示例参考，这些值可以互换或由其他值替换，例如，第一值和第二值。

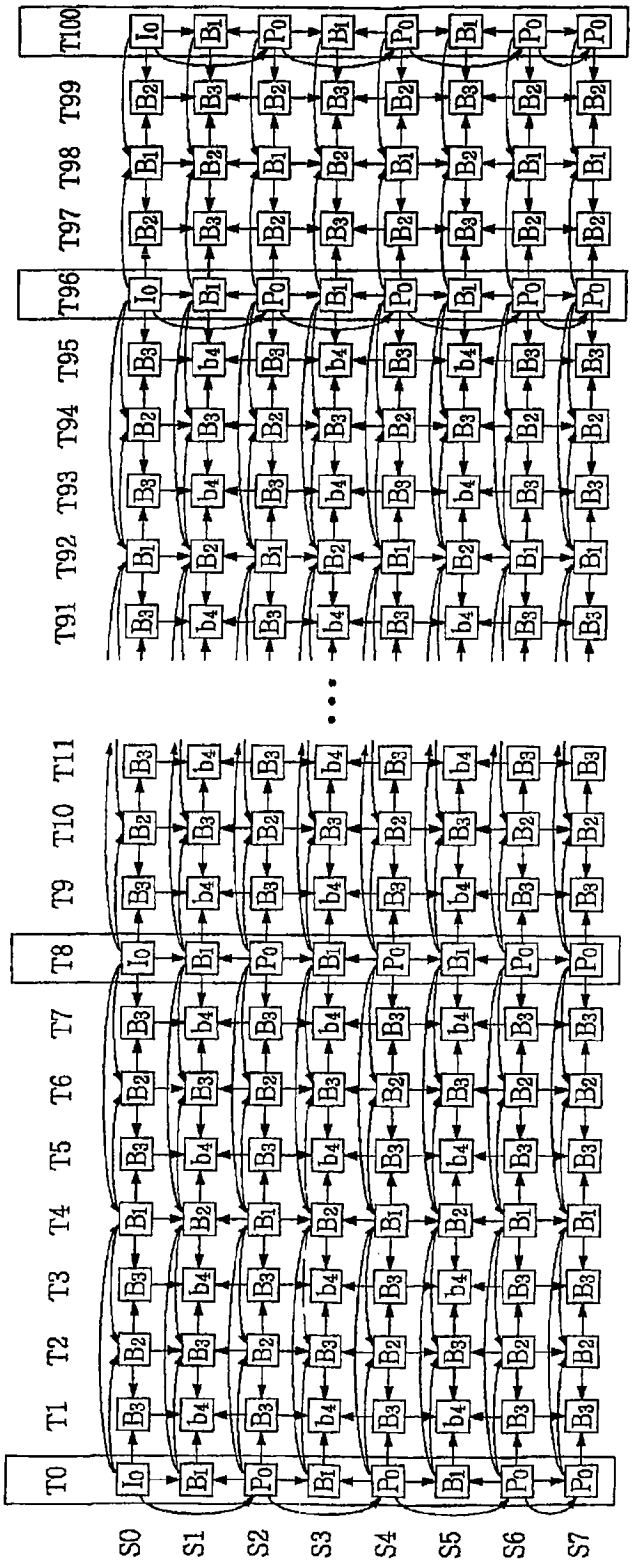

T100
T99
T98
T97
T96
T95
T94
T93
T92
T91
T11
T10
T9
T8
T7
T6
T5
T4
T3
T2
T1
T0
S0
S1
S2
S3
S4
S5
S6
S7

图 1

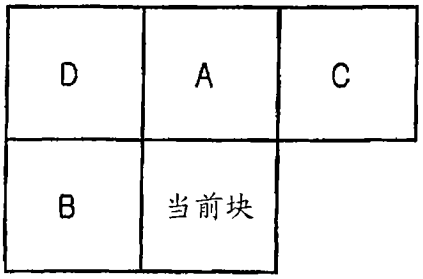

D
A
C
B
当前块

图 2

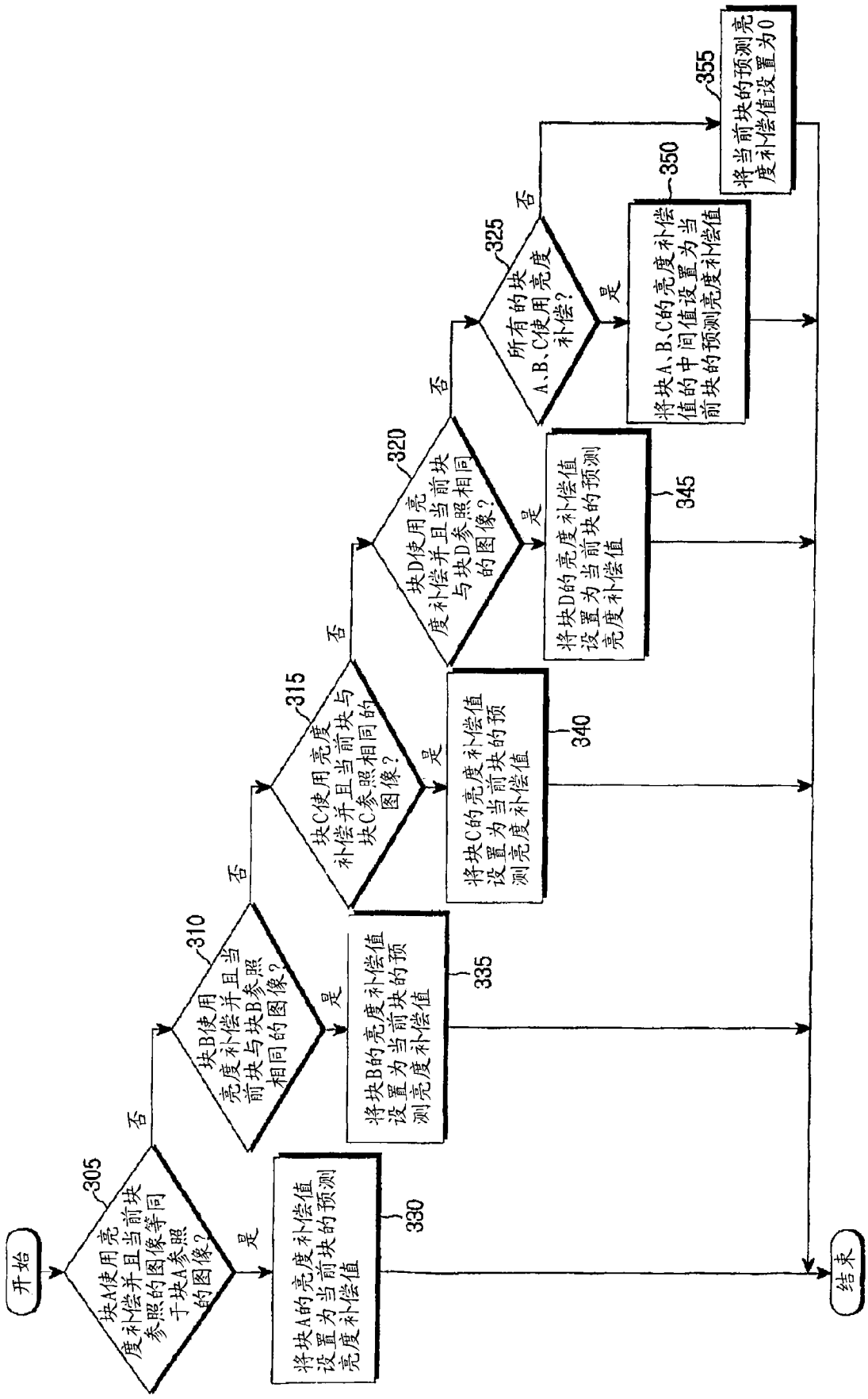

开始
块A使用亮度补偿并且当前块参照的图像等同于块A参照的图像?
305
是
否
将块A的亮度补偿值设置为当前块的预测亮度补偿值
330
块B使用亮度补偿并且当前块与块B参照相同的图像?
310
是
否
将块B的亮度补偿值设置为当前块的预测亮度补偿值
335
块C使用亮度补偿并且当前块C参照相同的图像?
315
是
否
将块C的亮度补偿值设置为当前块的预测亮度补偿值
340
块D使用亮度补偿并且当前块与块D参照相同的图像?
320
是
否
将块D的亮度补偿值设置为当前块的预测亮度补偿值
345
所有的块A、B、C使用亮度补偿?
325
是
否
将块A、B、C的亮度补偿值的中间值设置为当前块的预测亮度补偿值
350
将当前块的预测亮度补偿值设置为0
355
结束

图 3

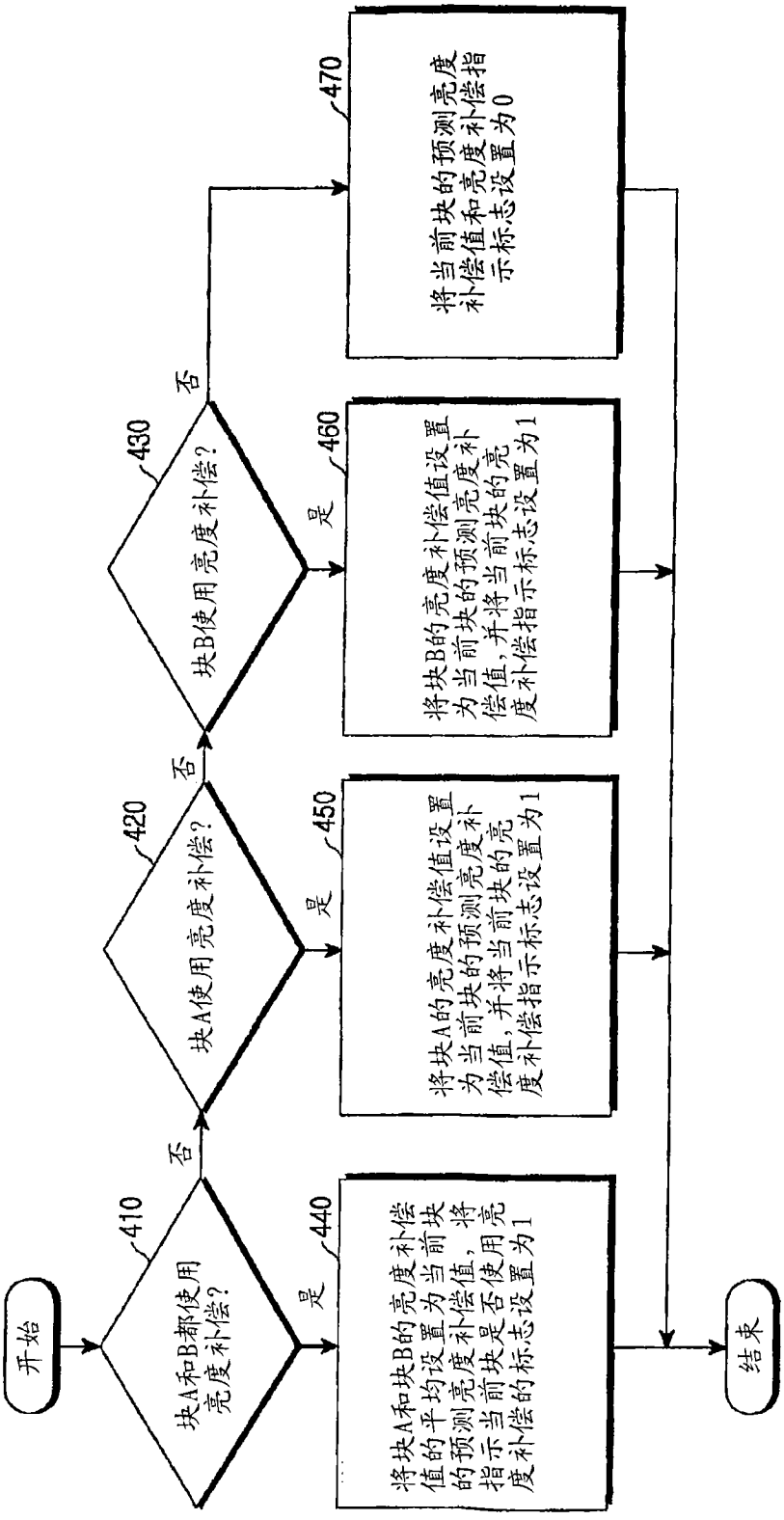

开始
410
块A和B都使用亮度补偿?
是
440
将块A和块B的亮度补偿值的平均设置为当前块的预测亮度补偿值，将指示当前块是否使用亮度补偿的标志设置为1
否
420
块A使用亮度补偿?
是
450
将块A的亮度补偿值设置为当前块的预测亮度补偿值，并将当前块的亮度补偿指示标志设置为1
否
430
块B使用亮度补偿?
是
460
将块B的亮度补偿值设置为当前块的预测亮度补偿值，并将当前块的亮度补偿指示标志设置为1
否
470
将当前块的预测亮度补偿值和亮度补偿指示标志设置为0
结束

图 4

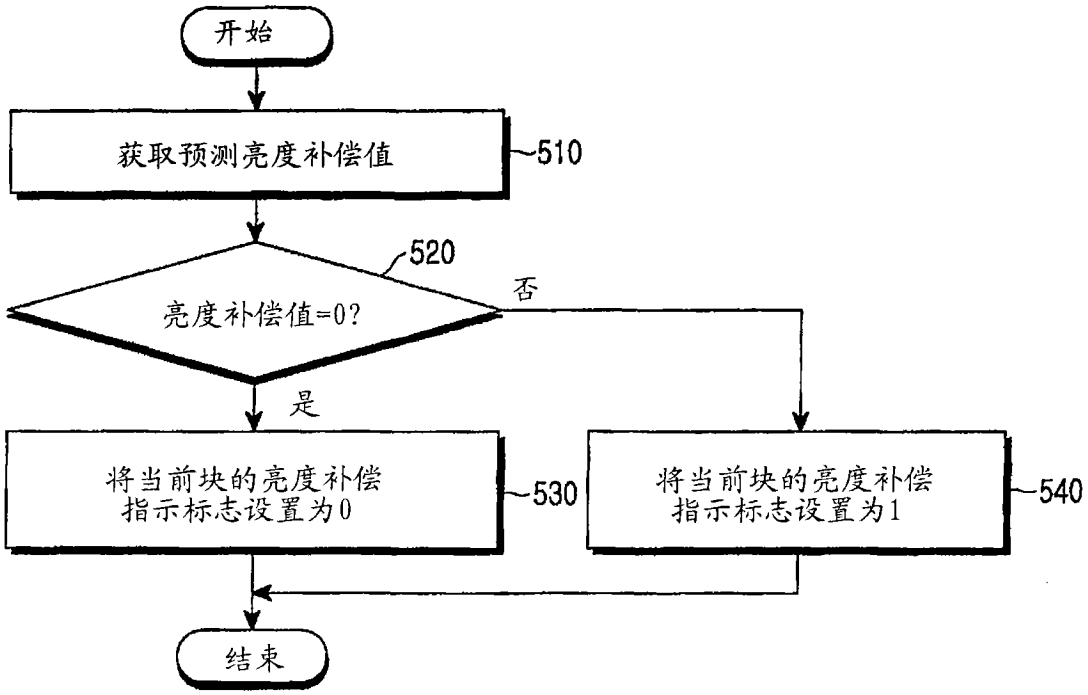

开始
获取预测亮度补偿值
510
520
亮度补偿值=0?
否
是
将当前块的亮度补偿
指示标志设置为0
530
将当前块的亮度补偿
指示标志设置为1
540
结束

图 5

```
Void MbDataAccess::xSpatialDirectMode( ParIdx8x8 eParIdx, Bool b8x8 )
{
  ...

#if TMS_SE_MVC_IC & TMS_SE_MVC_IC_DIRECT
  if(!getSH().getAVCFlag()) //JVT-W035
  {
  if( getMbData().getBlkMode( Par8x8( eParIdx < PART_8x8_2 ? (eParIdx>>1) :
((eParIdx - 4)>>1)) ) != BLK_SKIP ) {
    if( eParIdx == PART_8x8_0 ) {
      Icp& cIcp = getMbData().getMbIcp().getIcp();
      getIcpPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
      getIcpFlagPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]); //JVT-W031
    }
    else
    if( eParIdx == PART_8x8_3 ) {
      Icp cIcp, &cPrevIcp = getMbData().getMbIcp().getIcp();
      getIcpPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
      cPrevIcp.setPredOffset( (cPrevIcp.getPredOffset() + cIcp.getPredOffset())/4 );
      getIcpFlagPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]); //JVT-W031
      cPrevIcp.setPredIcAct( cPrevIcp.getPredIcAct() | cIcp.getPredIcAct() ); //JVT-W031
    }
    else {
      Icp cIcp, &cPrevIcp = getMbData().getMbIcp().getIcp();
      getIcpPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
      cPrevIcp.setPredOffset( cPrevIcp.getPredOffset() + cIcp.getPredOffset() );
      getIcpFlagPredictor(cIcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]); //JVT-W031
      cPrevIcp.setPredIcAct( cPrevIcp.getPredIcAct() | cIcp.getPredIcAct() ); //JVT-W031
    }
  }
  }
#endif
}

//JVT-W031
Void MbDataAccess::getIcpFlagPredictor( Icp& rcIcp, SChar scRefIdx0, SChar scRefIdx1 )

{ B4x4Idx    cIdx;
  MbMode         ePrevMbMode;
  MbIcp          cPrevMbIcp;
  MbMotionData cPrevMbMotionData0, cPrevMbMotionData1;

  B4x4Idx cIdxA = B4x4Idx(0).b4x4();
  B4x4Idx cIdxB = B4x4Idx(0).b4x4();
  B4x4Idx cIdxD = B4x4Idx(0).b4x4();
  B4x4Idx cIdxC = B4x4Idx(3).b4x4();
```

图 6A

```
const MbData& rcMbDataAbove          = xGetBlockAbove    ( cIdxA );
const MbData& rcMbDataLeft           = xGetBlockLeft     ( cIdxB );
const MbData& rcMbDataAboveRight = xGetBlockAboveRight( cIdxC );
const MbData& rcMbDataAboveLeft  = xGetBlockAboveLeft ( cIdxD );

Bool  bBlkValidA, bBlkValidB, bBlkValidC;
short sICflagA, sICflagB, sICflagC;

bBlkValidA = bBlkValidB = bBlkValidC = false;
sICflagA  = sICflagB  = sICflagC  = 0;

// above block
if( xIsAvailable( rcMbDataAbove ) )
{
   ePrevMbMode      = rcMbDataAbove.getMbMode();
   cPrevMbIcp       = rcMbDataAbove.getMbIcp();
   cPrevMbMotionData0 = rcMbDataAbove.getMbMotionData(LIST_0);
   cPrevMbMotionData1 = rcMbDataAbove.getMbMotionData(LIST_1);
        cIdx          = cIdxA;
        if( ePrevMbMode == MODE_SKIP || ePrevMbMode == MODE_16x16 )
          sICflagA  = cPrevMbIcp.getIcp().getIcAct();
    if(                  scRefIdx0==cPrevMbMotionData0.getRefIdx(cIdx)          &&
scRefIdx1==cPrevMbMotionData1.getRefIdx(cIdx) )
         bBlkValidA = true;
}
// left block
if( xIsAvailable( rcMbDataLeft ) )
{
   ePrevMbMode      = rcMbDataLeft.getMbMode();
   cPrevMbIcp       = rcMbDataLeft.getMbIcp();
   cPrevMbMotionData0 = rcMbDataLeft.getMbMotionData(LIST_0);
   cPrevMbMotionData1 = rcMbDataLeft.getMbMotionData(LIST_1);
        cIdx          = cIdxB;
        if( ePrevMbMode == MODE_SKIP || ePrevMbMode == MODE_16x16 )
          sICflagB  = cPrevMbIcp.getIcp().getIcAct();
    if(                  scRefIdx0==cPrevMbMotionData0.getRefIdx(cIdx)          &&
scRefIdx1==cPrevMbMotionData1.getRefIdx(cIdx) )
         bBlkValidB = true;
}
// aboveright block
if( xIsAvailable( rcMbDataAboveRight ) )
{
   ePrevMbMode      = rcMbDataAboveRight.getMbMode();
```

图 6B

```
        cPrevMbIcp       = rcMbDataAboveRight.getMbIcp();
        cPrevMbMotionData0 = rcMbDataAboveRight.getMbMotionData(LIST_0);
        cPrevMbMotionData1 = rcMbDataAboveRight.getMbMotionData(LIST_1);
              cIdx             = cIdxC;
              if( ePrevMbMode == MODE_SKIP || ePrevMbMode == MODE_16x16 )
                sICflagC  = cPrevMbIcp.getIcp().getIcAct();
        if(                    scRefIdx0==cPrevMbMotionData0.getRefIdx(cIdx)          &&
scRefIdx1==cPrevMbMotionData1.getRefIdx(cIdx) )
              bBlkValidC = true;
      }
      else
      // aboveleft block
      if( xIsAvailable( rcMbDataAboveLeft ) )
      {
        ePrevMbMode       = rcMbDataAboveLeft.getMbMode();
        cPrevMbIcp        = rcMbDataAboveLeft.getMbIcp();
        cPrevMbMotionData0 = rcMbDataAboveLeft.getMbMotionData(LIST_0);
        cPrevMbMotionData1 = rcMbDataAboveLeft.getMbMotionData(LIST_1);
              cIdx                = cIdxD;
              if( ePrevMbMode == MODE_SKIP || ePrevMbMode == MODE_16x16 )
                sICflagC  = cPrevMbIcp.getIcp().getIcAct();
        if(                    scRefIdx0==cPrevMbMotionData0.getRefIdx(cIdx)          &&
scRefIdx1==cPrevMbMotionData1.getRefIdx(cIdx) )
              bBlkValidC = true;
      }
      #define MEDIAN(a,b,c)  ((a)>(b)?(a)>(c)?(b)>(c)?(b):(c):(a):(b)>(c)?(a)>(c)?(a):(c):(b))
      // IC flag prediction
      short  sICflag = 0;
      {
            if( bBlkValidA && !bBlkValidB && !bBlkValidC )
                  sICflag = sICflagA;
            else if( !bBlkValidA && bBlkValidB && !bBlkValidC )
                  sICflag = sICflagB;
            else if( !bBlkValidA && !bBlkValidB && bBlkValidC )
                  sICflag = sICflagC;
            else
              sICflag = MEDIAN( sICflagA, sICflagB, sICflagC );
      }
      rcIcp.setPredIcAct( sICflag );
}
```

图 6C

```
Void MbDataAccess::xSpatialDirectMode( ParIdx8x8 eParIdx, Bool b8x8 )
{
  ...
#if TMS_SE_MVC_IC & TMS_SE_MVC_IC_DIRECT
  if(!getSH().getAVCFlag()) //JVT-W035
  {
  if( getMbData().getBlkMode( Par8x8( eParIdx < PART_8x8_2 ? (eParIdx>>1) :
  ((eParIdx - 4)>>1)) ) !=
BLK_SKIP )
  if( eParIdx == PART_8x8_0 ) {
     IcpA clcp = getMbData().getMbIcp().getIcp();
     getIcpPredictor(clcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
     if (clcp.getPredOffset() == 0)  clcp.setPredIcAct(0);
     else                            clcp.setPredIcAct(1);

  }
  else
  if( eParIdx == PART_8x8_3 ) {
     Icp clcp, &cPrevIcp = getMbData().getMbIcp().getIcp();
     getIcpPredictor(clcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
     cPrevIcp.setPredOffset( (cPrevIcp.getPredOffset() + clcp.getPredOffset())/4 );
     if (clcp.getPredOffset() == 0) {
        clcp.setPredIcAct(0);
        cPrevIcp.setPredIcAct(0);
     }
      else {
        clcp.setPredIcAct(1);
        cPrevIcp.setPredIcAct(1);
      }
   }
   else {
   {
      Icp clcp, &cPrevIcp = getMbData().getMbIcp().getIcp();
     getIcpPredictor(clcp, ascRefIdx[LIST_0], ascRefIdx[LIST_1]);
     cPrevIcp.setPredOffset( cPrevIcp.getPredOffset() + clcp.getPredOffset() );
     if (clcp.getPredOffset() == 0) {
        clcp.setPredIcAct(0);
        cPrevIcp.setPredIcAct(0);
     }
     else {
       clcp.setPredIcAct(1);
       cPrevIcp.setPredIcAct(1);
     }
   }
 }
 }
#endif
}
```

图 7

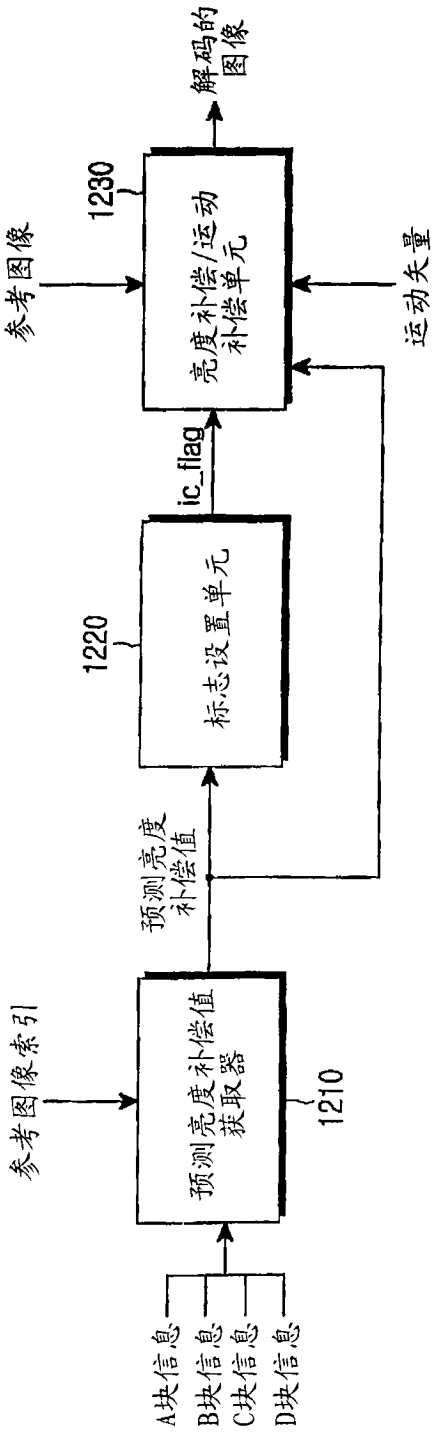

解码的图像
1230
参考图像
亮度补偿/运动补偿单元
运动矢量
ic_flag
1220
标志设置单元
预测亮度补偿值
参考图像索引
预测亮度补偿值获取器
1210
A块信息
B块信息
C块信息
D块信息

图 8

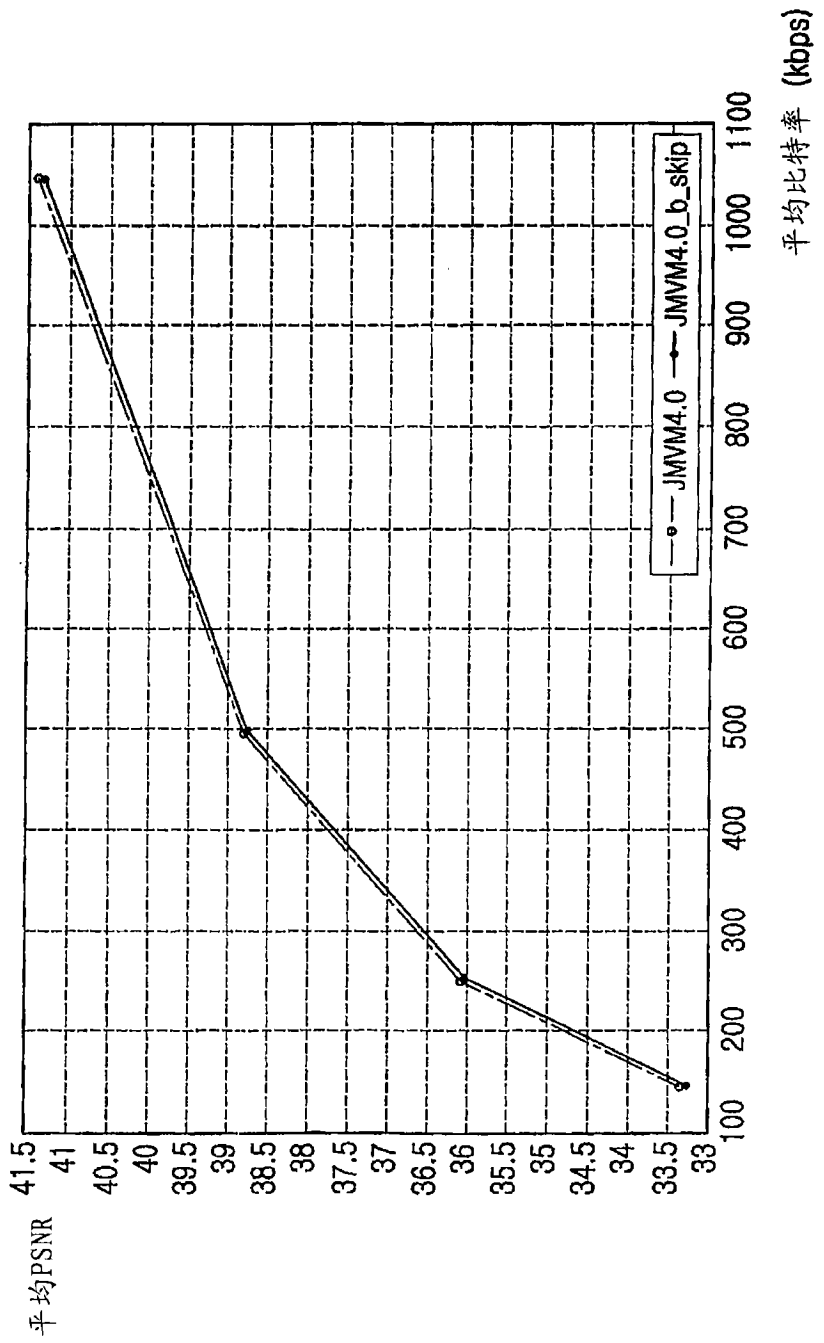

平均PSNR
41.5
41
40.5
40
39.5
39
38.5
38
37.5
37
36.5
36
35.5
35
34.5
34
33.5
33
100
200
300
400
500
600
700
800
900
1000
1100
平均比特率 (kbps)
JMVM4.0
JMVM4.0_b_skip

图 9

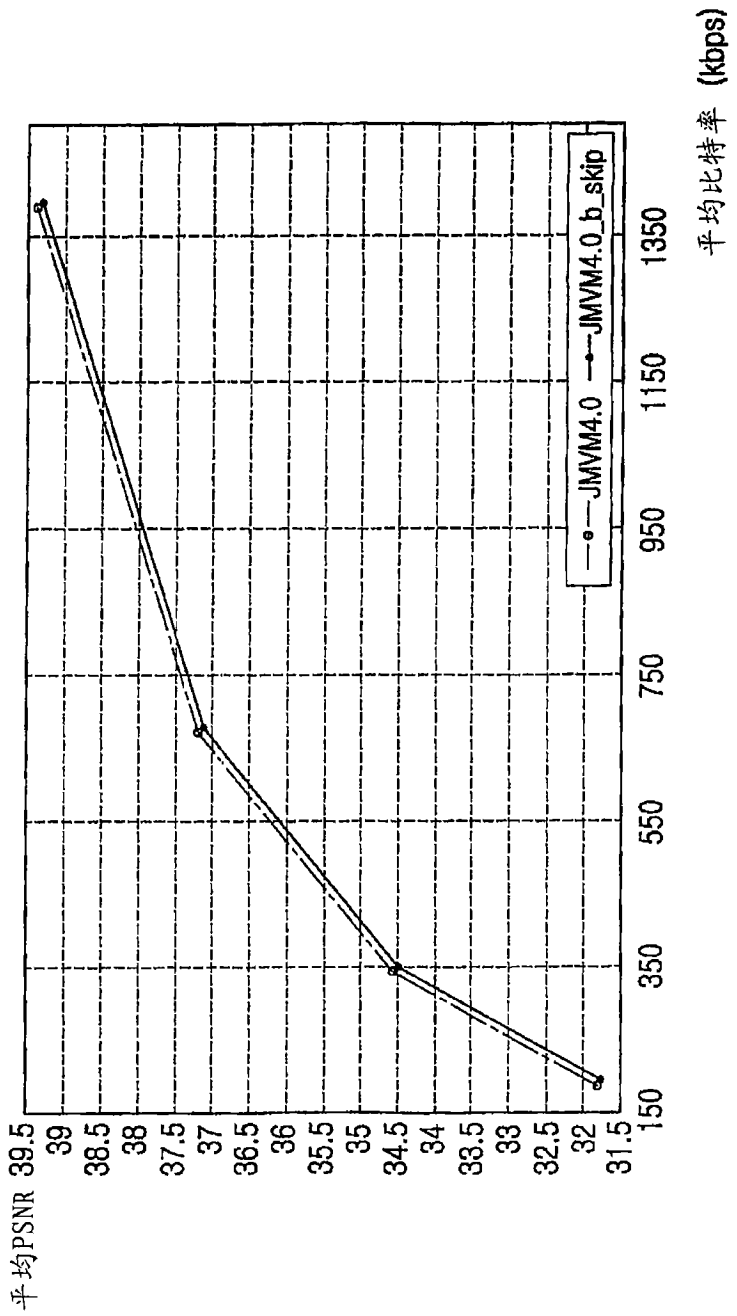

平均PSNR
39.5
39
38.5
38
37.5
37
36.5
36
35.5
35
34.5
34
33.5
33
32.5
32
31.5
150
350
550
750
950
1150
1350
平均比特率 (kbps)
JMVM4.0
JMVM4.0_b_skip

图 10