



(12)发明专利

(10)授权公告号 CN 103583044 B

(45)授权公告日 2018.01.26

(21)申请号 201280015709.8

(22)申请日 2012.01.31

(65)同一申请的已公布的文献号

申请公布号 CN 103583044 A

(43)申请公布日 2014.02.12

(30)优先权数据

10-2011-0009636 2011.01.31 KR

10-2011-0019166 2011.03.03 KR

10-2011-0050853 2011.05.27 KR

10-2011-0065707 2011.07.01 KR

10-2012-0010096 2012.01.31 KR

(85)PCT国际申请进入国家阶段日

2013.09.27

(86)PCT国际申请的申请数据

PCT/KR2012/000770 2012.01.31

(87)PCT国际申请的公布数据

W02012/105807 KO 2012.08.09

(73)专利权人 韩国电子通信研究院

地址 韩国大田市

(72)发明人 林成昶 金晖容 郑洗润 曹叔嬉

全东山 金钟昊 李河贤 李镇浩

崔振秀 金镇雄

(74)专利代理机构 北京市柳沈律师事务所

11105

代理人 李芳华

(51)Int.Cl.

H04N 19/137(2014.01)

(56)对比文件

CN 101841712 A,2010.09.22,

Gary Sullivan.Report of 6th JVT

Meeting.《Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG》.2002,第7节.

审查员 李春彦

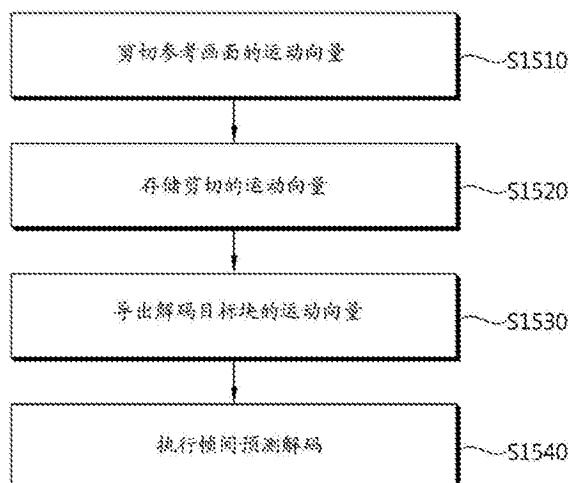
权利要求书1页 说明书22页 附图10页

(54)发明名称

用于使用运动向量来编码/解码图像的方法和
设备

(57)摘要

本发明涉及用于使用运动向量来编码/解码图像的方法和设备。根据本发明的一个实施例,提供了一种图像解码方法。该图像解码方法包括以下步骤:利用预定动态范围来剪切参考画面的运动向量,以便生成剪切的运动向量;在缓冲器中存储剪切的运动向量;使用缓冲器中存储的运动向量,来导出要解码的块的运动向量;和使用要解码的块的运动向量,来执行帧间预测解码。根据本发明,可降低存储运动向量所需的存储空间尺寸。



1. 一种图像解码设备,包括:
参考画面缓冲器,用于存储参考画面;和
运动补偿单元,用于使用该参考画面和该参考画面的运动向量,来生成预测块,
其中在对该参考画面的运动向量进行缩放之后,在预定范围中剪切该参考画面的运动向量,
其中基于该参考画面的画面顺序计数,来执行该参考画面的运动向量的缩放,
其中该参考画面的运动向量是该参考画面中的并置块的运动向量,以及
其中通过执行以下操作来确定并置块:
计算与当前画面中的当前块的右下位置对应的位置;和
确定该参考画面中覆盖所计算的位置的块。
2. 根据权利要求1的图像解码设备,其中该运动向量在预定固定值范围中剪切。
3. 根据权利要求1的图像解码设备,
其中该参考画面的运动向量被存储在预定块单元中,并且
其中该运动补偿单元使用该预定块单元中存储的参考画面的运动向量,来生成预测块。
4. 根据权利要求2的图像解码设备,其中该运动向量的X和Y分量在同一固定值范围中剪切。
5. 根据权利要求1的图像解码设备,其中该运动向量是按照帧间预测模式解码的块的运动向量。

用于使用运动向量来编码/解码图像的方法和设备

技术领域

[0001] 本发明涉及图像处理,并更具体地,涉及使用剪切的运动向量的视频编码/解码方法及其设备。

背景技术

[0002] 最近,根据支持高清晰度(HD)分辨率的广播系统在全国和全世界的扩展,许多用户已习惯于高分辨率和清晰度图像,使得许多组织已进行开发下一代视频装置的许多尝试。另外,因为对于HDTV和具有比HDTV的分辨率高四倍的分辨率的超高清晰度(UHD)的兴趣已增长,所以已需要用于更高分辨率和更高清晰度视频的压缩技术。

[0003] 关于图像压缩,可使用根据当前画面之前和/或之后的画面来预测当前画面中包括的像素值的帧间预测技术、使用画面中的像素信息来预测像素值的帧内预测技术、向具有高出现频率的码元分配短代码并向具有低出现频率的码元分配长代码的熵编码技术等。

发明内容

[0004] 【技术问题】

[0005] 本发明提供了使用剪切的运动向量的视频编码/解码方法及其设备。

[0006] 本发明还提供了用于剪切参考画面的运动向量的方法。

[0007] 本发明还提供了用于传送关于运动向量的信息的方法。

[0008] 【技术方案】

[0009] 1. 本发明的实施例涉及一种编码图像的方法。该方法包括:在预定动态范围中剪切参考画面的运动向量,以生成剪切的运动向量,在缓冲器中存储剪切的运动向量,并使用缓冲器中存储的运动向量,来对编码树块(treeblock)的运动向量进行编码。

[0010] 2. 在1中,该动态范围可通过视频编解码器的级别来定义。

[0011] 3. 在1中,该动态范围可通过预定比特深度来确定,并且该比特深度可通过视频编解码器的级别来定义。

[0012] 4. 在1中,可在不同动态范围中剪切该参考画面的运动向量的X分量和Y分量。

[0013] 5. 本发明的另一实施例涉及一种解码图像的方法。该方法包括:在预定动态范围中剪切参考画面的运动向量,以生成剪切的运动向量,在缓冲器中存储剪切的运动向量,使用缓冲器中存储的运动向量,来导出编码树块的运动向量,并使用编码树块的运动向量,来执行帧间预测解码。

[0014] 6. 在5中,该动态范围可通过视频编解码器的级别来定义。

[0015] 7. 在5中,该动态范围可通过预定比特深度来确定,并且该比特深度可通过视频编解码器的级别来定义。

[0016] 8. 在5中,该动态范围可通过预定比特深度来确定,并且该比特深度可通过从编码图像的设备传送的序列参数集来获得。

[0017] 9. 在8中,该序列参数集可包括:标记,指示是否剪切该参考画面的运动向量;和参

数,用于获得该比特深度。

[0018] 10.在9中,所述解码图像的方法可包括压缩该参考画面的运动向量,其中该序列参数集包括:标记,指示是否压缩该参考画面的运动向量;和参数,用于获得该参考画面的运动向量的压缩比。

[0019] 11.在5中,所述解码图像的方法可包括限制该参考画面的运动向量的表现分辨率。

[0020] 12.在5中,所剪切的运动向量可根据优先级来存储。

[0021] 13.在5中,所剪切的运动向量可以是按照帧间预测模式编码的块的运动向量。

[0022] 14.在5中,所述解码图像的方法可包括执行该参考画面的运动向量的缩放。

[0023] 15.在5中,可在不同动态范围中剪切该参考画面的运动向量的X分量和Y分量。

[0024] 16.在15中,该X分量的动态范围和该Y分量的动态范围可通过视频编解码器的级别来定义。

[0025] 17.本发明的另一实施例涉及一种解码图像的设备。该设备包括:参考画面缓冲器,用于存储参考画面;和运动补偿器,用于使用该参考画面和该参考画面的运动向量来生成预测块,其中在预定动态范围中剪切该参考画面的运动向量。

[0026] 18.在17中,该动态范围可通过视频编解码器的级别来定义。

[0027] 19.在17中,该动态范围可通过预定比特深度来确定,并且该比特深度可通过视频编解码器的级别来定义。

[0028] 20.在17中,该动态范围可通过预定比特深度来确定,并且该比特深度可通过从编码图像的设备传送的序列参数集来获得。

[0029] **【有利效果】**

[0030] 根据本发明的示范实施例,可使用剪切的运动向量来对视频进行编码。

[0031] 根据本发明的示范实施例,可减小用于存储运动向量所需的存储器的尺寸。

[0032] 根据本发明的示范实施例,可减小用于从存储器中取得数据所需的存储器存取带宽。

附图说明

[0033] 图1是示出了根据本发明示范实施例的编码器的结构的示例的框图。

[0034] 图2是示出了根据本发明示范实施例的解码器的结构的示例的框图。

[0035] 图3示出了编码/解码目标画面和参考画面的示例。

[0036] 图4示出了限制运动向量的动态范围的示例。

[0037] 图5到8是示出了用于存储参考画面的运动向量的方法的流程图。

[0038] 图9示出了对运动向量进行量化的示例。

[0039] 图10到13示出了从参考画面取得运动信息的示例。

[0040] 图14是示出了根据本发明示范实施例的用于编码图像的方法的流程图。

[0041] 图15是示出了根据本发明示范实施例的用于解码图像的方法的流程图。

具体实施方式

[0042] 其后,将参考附图来详细描述本发明的示范实施例。然而,在描述本发明的示范实

施例时,将不详细描述公知功能或构造,因为它们可能不必要地使得本发明的理解模糊。

[0043] 应理解的是,当将任一元件称为“连接到”或“耦接到”另一元件时,该元件可直接连接到或直接耦接到另一元件、或者在其间插入有其他元件的情况下连接到或耦接到另一元件。此外,在本说明书中,在描述“包括”特定组件的情况下,应理解的是,不排除除了对应组件之外的附加组件,而是可包括在本发明的示范实施例或技术范围内。

[0044] 该说明书中使用的术语“第一”、“第二”等可用来描述各种组件,但是这些组件不应被解释为限于这些术语。即,这些术语仅用来区分一个组件与另一组件。例如,“第一”组件可被称作“第二”组件,并且反之亦然,而不脱离本发明的范围。

[0045] 此外,本发明的示范实施例中描述的组件被独立示出,仅为了指示它们执行不同的特性功能。所以,独立示出的组件并不意味着可以不将每一组件实现为一个硬件或软件。即,为了便于解释可划分每一组件,并且多个组件可彼此组合以由此作为一个组件操作,或者可将一个组件划分为多个组件以由此作为多个组件操作,它们被包括在本发明的范围内,只要其脱离本发明的必要特性。

[0046] 另外,一些组件可以不是执行本发明的必要功能的必要组件,而仅是改进其性能的选择性组件。还可仅通过包括除了选择性组件之外的必要组件的结构,来实现本发明,并且仅包括必要组件的结构也被包括在本发明的范围内。

[0047] 图1是示出了根据本发明示范实施例的编码器的结构的示例的框图。

[0048] 参考图1,编码器100包括运动预测器111、运动补偿器112、帧内预测器120、开关115、减法器125、变换器130、量化器140、熵编码器150、反量化器160、逆变换器170、加法器175、滤波器单元180、和参考画面缓冲器190。

[0049] 编码器100按照帧内预测模式或帧间预测模式对编码器的输入图像进行编码,以输出比特流。帧内预测意味着画面内预测,而帧间预测意味着画面间预测。通过开关115的切换,编码器100在帧内预测模式和帧间预测模式之间切换。编码器100生成用于输入图像的输入块的预测块,并然后对输入块和预测块之间的残差进行编码。

[0050] 在帧内预测模式的情况下,帧内预测器120使用已经编码的相邻块的像素值来执行空间预测,以生成预测块。

[0051] 在帧间预测模式的情况下,运动预测器111在运动预测处理期间在参考画面缓冲器190中存储的参考画面中搜索与输入块最佳匹配的参考块,以获得运动向量。运动补偿器112通过使用该运动向量来执行运动补偿,以生成预测块。这里,运动向量可以是用于帧间预测的二维向量,并表示当前编码树块和参考块之间的偏移。

[0052] 减法器125基于输入块和预测块之间的残差而生成残差块,而变换器130对残差块进行变换以输出变换系数。此外,量化器140量化该变换系数,以输出量化后的系数。

[0053] 熵编码器150基于在编码/量化处理期间获得的信息,来执行熵编码,以输出比特流。熵编码将频繁生成的码元表示为少量比特,由此减少用于编码码元的比特流的尺寸。所以,可期望通过熵编码来改进视频的压缩性能。熵编码器150可使用诸如指数哥伦布、上下文自适应可变长度编码(CAVLC)、上下文自适应二进制算术编码(CABAC)等的编码方法用于熵编码。

[0054] 编码的画面需要被再次解码并存储,以便使用作用于执行帧间预测编码的参考画面。所以,反量化器160对量化后的系数进行反量化,而逆变换器170对反量化后的系数进行

逆变换,以输出重构的残差块。加法器175将重构的残差块添加到预测块,以生成重构块。

[0055] 滤波器单元180也被称为自适应环路内滤波器,并对重构块应用解块滤波、样本自适应偏移(SAO)补偿、和自适应环形滤波(ALF)中的至少一个。解块滤波意味着去除在块之间的边界处出现的块失真,而SAO补偿意味着向像素值添加适当偏移以便补偿编码误差。另外,ALF意味着基于重构图像和原始图像之间的比较值来执行滤波。

[0056] 其间,参考画面缓冲器190在其中存储经过该滤波器单元180的重构块。

[0057] 图2是示出了根据本发明示范实施例的解码器的结构的示例的框图。

[0058] 参考图2,解码器包括熵解码器210、反量化器220、逆变换器230、帧内预测器240、运动补偿器250、加法器255、滤波器单元260、和参考画面缓冲器270。

[0059] 解码器200按照帧内预测模式或帧间预测模式来解码比特流,以输出重构的图像。通过开关的切换而在帧内预测模式和帧间预测模式之间切换该解码器200。解码器200从比特流获得残差块,以生成预测块,并然后将残差块和预测块彼此相加,以生成重构块。

[0060] 熵解码器210基于概率分布来执行熵解码。熵解码处理是与上述熵编码处理相对的处理。即,熵解码器210根据其中将频繁生成的码元表示为少量比特的比特流来生成包括量化的系数的码元。

[0061] 反量化器220对量化后的系数进行反量化,而逆变换器230对反量化后的系数进行逆变换,以生成残差块。

[0062] 在帧内预测模式的情况下,帧内预测器240使用已编码的相邻块的像素值来执行空间预测,以生成预测块。

[0063] 在帧间预测模式的情况下,运动补偿器250通过使用运动向量和在参考画面缓冲器270中存储的参考画面,来执行运动补偿,以生成预测块。

[0064] 加法器255将预测块和残差块相加,并且滤波器单元260向经过加法器的块应用解块滤波、SAO补偿、和ALF中的至少一个,以输出重构图像。

[0065] 重构图像可被存储在参考画面缓冲器270中,以由此用于运动补偿。

[0066] 其后,块意味着编码/解码单元。在编码/解码处理中,图像按照预定尺寸划分并然后编码/解码。所以,块也可被称为编码单元(CU)、预测单元(PU)、变换单元(TU)等,并且单一块也可被划分为具有较小尺寸的子块。

[0067] 这里,预测单元意味着其中执行预测和/或运动补偿的基本单元。预测单元可被划分为多个分区,并且每一分区也可被称为预测单元分区。当将预测单元划分为多个分区时,每一预测单元分区可成为其中执行预测和/或运动补偿的基本单元。其后,在本发明的示范实施例中,预测单元也可意味着预测单元分区。

[0068] 其间,在高效率视频编码(HEVC)中,使用基于先进运动向量预测(AMVP)的运动向量预测方法。

[0069] 在基于先进运动向量预测的运动向量预测方法中,可使用参考画面中的在与编码树块的位置相同或对应的位置处存在的块的运动向量(MV)、以及位于该编码树块周围的重构块的运动向量。这里,参考画面中的在与编码树块的位置相同或空间对应的位置处存在的块被称为并置块,而并置块的运动向量被称为并置运动向量或时间运动向量。然而,并置块可以是参考画面中的在与编码树块的位置类似的(即,对应的)位置处存在的块、以及在与编码树块的位置相同的位置处存在的块。

[0070] 在运动信息合并方法中,根据并置块以及位于该编码树块周围的重构块来估计运动信息,以由此用作编码树块的运动信息。这里,运动信息包括在帧间预测时需要的指示参考画面索引、运动向量、单向、双向等的帧间预测模式信息、参考画面列表、以及关于按照帧内预测模式还是按照帧间预测模式来执行编码的预测模式信息中的至少一个。

[0071] 编码树块中的预测运动向量可以是作为与编码树块时间相邻的块的并置块的运动向量、以及与编码树块空间相邻的相邻块的运动向量。

[0072] 图3示出了编码/解码目标画面和参考画面的示例。

[0073] 在图3中,块X指示编码/解码目标画面310中的编码树块,而块A、块B、块C、块D和块E指示位于编码树块周围的重构块。另外,参考画面320中的块T指示在与编码树块的位置对应的位置处存在的并置块。

[0074] 将编码树块中的哪个运动向量用作预测运动向量可通过运动向量预测器索引来识别。

[0075] [表1]

[0076]

prediction_unit(x0,y0,log2PUWidth,log2PUHeight) {	描述符
if (skip_flag[x0][y0]) {	
if (NumMVPCand(L0)>1)	
mvp_idx_10[x0][y0]	ue(v) ae(v)
if (NumMVPCand(L1)>1)	
mvp_idx_11[x0][y0]	ue(v) ae(v)
} else if (PredMode==MODE_INTRA) {	
...	
} else { /*MODE_MERGE,MODE_INTER*/	
if (merge_flag[x0][y0] && NumMergeCandidates>1) {	
...	
} else {	
if (inter_pred_idc[x0][y0]!=Pred_L1) {	
if (NumMVPCand(L0)>1)	
mvp_idx_10	ue(v) ae(v)
}	
if (inter_pred_idc[x0][y0]!=Pred_L0) {	
if (NumMVPCand(L1)>1)	
mvp_idx_11	ue(v) ae(v)
}	
}	
}	
}	

[0077] 如表1中示出的,每一参考画面列表的运动向量预测器索引mvp_idx_10和mvp_idx_11被传送到解码器,并且解码器使用与编码器所预测的运动向量相同的运动向量作为

预测运动向量。

[0078] 在其中使用与编码树块空间相邻的相邻块的运动向量来编码/解码该编码树块的情况下,可仅利用具有相对小尺寸的存储器来存储运动向量。然而,在其中使用时间运动向量的情况下,由于参考画面的所有运动向量需要被存储在存储器中,所以需要具有相对大尺寸的存储器,并且从存储器取得数据所需的存储器存取带宽的尺寸也增加。所以,存在在在其中便携式终端等的存储空间不充足或者功耗最小化的应用环境下、更有效地存储时间运动向量的需求。

[0079] 其间,作为在存储器中存储运动向量的技术,存在降低运动向量的空间分辨率的方法。在该方法中,运动向量按照任意比率压缩并然后存储在存储器中。例如,4×4块单元中存储的运动向量被存储在4×4或更大的块单元中,以减少存储的运动向量的数目。这里,为了调整存储的运动向量的块尺寸,传送关于压缩比率的信息。通过表2中示出的序列参数集 (SPS) 来传送该信息。

[0080] [表2]

[0081]

seq_parameter_set_rbsp() {	C	描述符
.....		
motion_vector_buffer_comp_flag	0	u (1)
if (motion_vector_buffer_comp_flag)		
motion_vector_buffer_comp_ratio_log2	0	u (8)
rbps_trailing_bits()	0	
}		

[0082] 参考表2,在其中motion_vector_buffer_comp_flag为1的情况下,执行运动向量缓冲器压缩处理。

[0083] motion_vector_buffer_comp_ratio_log2指示运动向量缓冲器压缩处理的压缩比率。在其中不存在motion_vector_buffer_comp_ratio_log2的情况下,估计motion_vector_buffer_comp_ratio_log2为0,并且运动向量缓冲器压缩比率由等式1表示。

[0084] [等式1]

[0085] $MVBufferCompRatio.1 < \text{motion_vector_buffer_comp_ratio_log2}$

[0086] 例如,在其中1920×1080个画面的所有4×4块具有不同运动向量、并使用其每一个使用两个参考画面的两个参考画面列表的情况下,如下面描述的那样需要3.21M字节的总存储空间来存储时间运动向量。

[0087] 1. 每一个运动向量26比特的比特深度

[0088] (1) 运动向量的X分量的动态范围:-252到+7676 (比特深度:13比特)

[0089] (2) 运动向量的Y分量的动态范围:-252到+4316 (比特深度:13比特)

[0090] (3) (基于对应画面中的第一预测单元,来计算运动向量的每一分量的动态范围。)

[0091] 2. 在其中所有4×4块单元具有不同运动向量的情况下:480×270=129600块

[0092] 3. 每一块两个运动向量的使用

[0093] 4. 参考画面列表的数目:2

[0094] 5. 每一参考画面列表的两个参考画面的使用

[0095] $= > 26 \text{ 比特} \times 129600 \text{ 块} \times \text{两个运动向量} \times \text{两个参考画面列表} \times \text{两个参考画面} = 26956800 \text{ 比特} = 3.21 \text{ M 字节}$

[0096] 根据如上所述用于降低运动向量的空间分辨率的方法,可能使用运动向量的空间相关性,来降低需要的存储空间和存储器存取带宽的尺寸。然而,用于降低运动向量的空间分辨率的方法不限制运动向量的动态范围。

[0097] 当将存储空间的尺寸降低为1/4时,上述示例中需要的存储空间的尺寸降低为大约0.8M字节。这里,当通过附加限制运动向量的动态范围来对于运动向量的每一分量使用用于存储该运动向量所需的比特深度的仅六个比特时,需要的存储空间的尺寸可进一步降低为0.37M字节。

[0098] 所以,在本发明的示范实施例中,限制运动向量的动态范围,以便降低用于存储该运动向量所需的存储空间和用于从存储器取得数据所需的存储器存取带宽的尺寸。可使用限制其动态范围的参考画面的运动向量作为编码树块中的时间运动向量。

[0099] 其后,动态范围意味着基于0的运动向量的负分量或正分量的最小值和最大值之间的范围,而指示用于存储该运动向量所需的空间的尺寸的比特深度意味着比特宽度。另外,除非特别描述,否则运动向量意味着参考画面的运动向量,即时间运动向量。

[0100] 在其中运动向量的每一分量超出该动态范围的情况下,其由对应动态范围的最小值或最大值表示。例如,在其中运动向量的X分量为312并且该运动向量的每一分量的动态范围的最大值为256的情况下,运动向量的X分量被限制为256。

[0101] 同样,在其中运动向量的每一分量的比特深度为16比特并且运动向量为(-36,24)的情况下,当运动向量的每一分量的比特深度被限制为6比特时,运动向量的每一分量具有动态范围-32到+31,使得运动向量由(-32,24)表示,这在其动态范围中。

[0102] 此外,在其中运动向量的每一分量的比特深度为16比特并且运动向量为(-49,142)的情况下,当运动向量的每一分量的比特深度被限制为9比特时,运动向量的每一分量具有动态范围-256到+255,使得运动向量由(-49,142)表示,而没有改变。

[0103] 图4示出了限制运动向量的动态范围的示例。

[0104] 参考图4,当具有动态范围-4096到+4095的动态范围的运动向量的动态范围被限制为-128到+127时,比特深度可从13比特降低到8比特。

[0105] 时间运动向量的每一分量如同等式2和3表示的那样被剪切,以便被存储在N比特的比特深度中,其中N指示正整数。

[0106] [等式2]

[0107] $\text{clippedMV_X} = \min(1 \ll (N-1) - 1, \max(-1 \ll (N-1), \text{MV_X}))$

[0108] [等式3]

[0109] $\text{clippedMV_Y} = \min(1 \ll (N-1) - 1, \max(-1 \ll (N-1), \text{MV_Y}))$

[0110] 其中MV_X指示运动向量的X分量,MV_Y指示运动向量的Y分量,min(a,b)意味着输出a和b中的较小值的操作,而max(a,b)意味着输出a和b中的较大值的操作。clippedMV_X和clippedMV_Y中的每一个指示剪切的时间运动向量的X和Y分量,并被存储在存储器中,以由此用作编码树块的时间运动向量。

[0111] 例如,如表3中所示,在其中存储空间的尺寸是48字节并且运动向量的每一分量使用16比特的比特深度的情况下,可存储总共十二个运动向量。

[0112] [表3]

[0113]

MV1-X	MV1-Y	MV2-X	MV2-Y	MV3-X	MV3-Y	MV4-X	MV4-Y
MV5-X	MV5-Y	MV6-X	MV6-Y	MV7-X	MV7-Y	MV8-X	MV8-Y
MV9-X	MV9-Y	MV10-X	MV10-Y	MV11-X	MV11-Y	MV12-X	MV12-Y

[0114] 然而,当运动向量的每一分量仅使用8比特的比特深度时,可如表4中所示存储总共二十四个运动向量。

[0115] [表4]

[0116]

MV1-X	MV1-Y	MV2-X	MV2-Y	MV3-X	MV3-Y	MV4-X	MV4-Y
MV5-X	MV5-Y	MV6-X	MV6-Y	MV7-X	MV7-Y	MV8-X	MV8-Y
MV9-X	MV9-Y	MV10-X	MV10-Y	MV11-X	MV11-Y	MV12-X	MV12-Y

[0117]

MV13-X	MV13-Y	MV14-X	MV14-Y	MV15-X	MV15-Y	MV16-X	MV16-Y
MV17-X	MV17-Y	MV18-X	MV18-Y	MV19-X	MV19-Y	MV20-X	MV20-Y
MV21-X	MV21-Y	MV22-X	MV22-Y	MV23-X	MV23-Y	MV24-X	MV24-Y

[0118] 所以,根据本发明的示范实施例,当在编码器和/或解码器中重构的图像经受诸如解块滤波、自适应环路滤波等的环路内滤波处理并然后被存储在解码画面缓冲器 (DPB) 中时,限制运动向量的动态范围,使得存储参考画面的运动向量。解码画面缓冲器意味着图1或图2的参考画面缓冲器。

[0119] I. 剪切运动向量的处理

[0120] 在其中slice_type不等于I的情况下,调用剪切运动向量的每一分量的处理。在完成滤波处理之后,在树块或最大编码单元 (LCU) 中执行剪切运动向量的处理。

[0121] 剪切运动向量的处理中的输入是相对于当前画面的左上样本规定预测单元的左上样本的地点 (xP,yP)、以及运动向量索引MvL0和MvL1。该处理中的输出是剪切的运动向量索引CMvL0和CMvL1。

[0122] 对于索引MvL0、MvL1、CMvL0和CMvL1,执行等式4到7的操作。

[0123] [等式4]

[0124] $mvLX = MvLX[xP, yP]$

[0125] [等式5]

[0126] $cmvLX[0] = Clip3(-1 \ll (TMVBitWidth-1), 1 \ll (TMVBitWidth-1)-1, mvLX[0])$

[0127] [等式6]

[0128] $cmvLX[1] = Clip3(-1 \ll (TMVBitWidth-1), 1 \ll (TMVBitWidth-1)-1, mvLX[1])$

[0129] [等式7]

[0130] $CMvLX[xP, yP] = cmvLX$

[0131] 其中TMVBitWidth指示运动向量的比特深度,Clip3(a,b,c)意味着剪切c以便在a和b之间的范围中存在的函数。

[0132] II. 存储运动向量的处理

[0133] 图5到8是示出了用于存储参考画面的运动向量的方法的流程图。

[0134] 参考图5,可使用存储重构图像的图像缓冲器和存储运动向量的运动向量缓冲器两者,来存储参考画面的运动向量。这里,重构图像经受环路内滤波处理(S510),而运动向量经受限制动态范围处理(S520)并然后存储(S540)。

[0135] 另外,参考图6,使用图像缓冲器和运动向量缓冲器两者,并且运动向量经受限制动态范围处理(S620)和降低空间分辨率处理(S630)并然后存储(S640)。

[0136] 此外,参考图7,重构图像经受环路内滤波处理(S710)并然后存储在图像缓冲器中(S740),而运动向量经受限制动态范围处理(S720)并然后存储在运动向量缓冲器中(S750)。

[0137] 此外,参考图8,重构图像经受环路内滤波处理(S810)并然后存储在图像缓冲器中(S840),而运动向量受限制动态范围处理(S820)和降低空间分辨率处理(S830)并然后存储(S850)。

[0138] 其间,在图6和8的示范实施例中,限制动态范围处理S620或S820以及降低空间分辨率处理S630和S830的顺序不受限制,而是可以改变。

[0139] 另外,为了进一步降低存储器存取带宽,可以不同地限制运动向量的每一分量的动态范围。例如,可限制X分量的动态范围和Y分量的动态范围中的仅一个,或者与X分量的动态范围相比可进一步限制Y分量的动态范围。

[0140] 限制的运动向量的动态范围通过序列参数集、画面参数集(PPS)、片段报头等来传送,并且解码器类似地执行序列、画面、或片段中的时间运动向量的动态范围的限制。在该情况下,也可传送作为用于存储该动态范围中表示的运动向量所需的存储空间的尺寸的比特深度。另外,可能使用通过序列参数集、画面参数集、片段报头等传送的动态范围,来有效存储时间运动向量,以便与图像的运动特性匹配,而不使用具有固定尺寸的比特深度来存储运动向量。

[0141] 其间,可量化和存储运动向量。在其中量化和存储运动向量的情况下,降低运动向量的精度。作为量化方法,存在其中步长尺寸均匀的均匀量化、其中步长尺寸不均匀的不均匀量化等。量化中的步长尺寸被设置为在编码器和解码器之间预定义的固定值,或者通过序列参数集、画面参数集、片段报头等从编码器传送到解码器。解码器原样使用量化的运动向量,或者对量化的运动向量进行反量化和使用。图9示出了对运动向量进行量化的示例。参考图9,在其中运动向量具有32到48的分量值的情况下,运动向量被量化到40。

[0142] 另外,运动向量可被限制在代表分辨率中并然后存储。代表分辨率意味着整数像素单元(1像素单元)、分数像素单元(1/2像素单元、1/4像素单元等)。例如,在1/4像素单元中处理的运动向量的分辨率可被存储为整数像素。运动向量的代表分辨率被设置为在编码器和解码器之间预定义的固定值,或者通过序列参数集、画面参数集、片段报头等从编码器传送到解码器。

[0143] 另外,仅对于存储器中存储的时间运动向量之中的一些运动向量,可执行运动向量的限制动态范围处理、降低空间分辨率处理、和量化处理。

[0144] 在其中限制并存储运动向量的动态范围的情况下,可添加关于运动向量的动态范围的信息并存储在存储器中。例如,在其中运动向量的动态范围是-128到+127的情况下,可附加存储标记1,并且在其中运动向量的动态范围是-32到+31的情况下,可附加存储标记0。在该情况下,标记信息可与运动向量一起存储,或者可被存储在与其中存储运动向量的存

存储器不同的存储器中。在其中标记信息和运动向量被存储在不同存储器中的情况下,当识别在哪个动态范围中存储特定运动向量时,可允许标记信息的任意存取。另外,通过序列参数集、画面参数集、片段报头等来传送关于在哪个动态范围中存储一些运动向量的信息,由此使得可能允许解码器执行与编码器的操作类似的操作。

[0145] 在其中降低运动向量的空间分辨率并存储的情况下,关于运动向量的块尺寸的信息可被添加并存储在存储器中。例如,在其中运动向量的块尺寸是 4×4 的情况下,可附加存储标记1,并且在其中运动向量的块尺寸是 16×16 的情况下,可附加存储标记0。在该情况下,标记信息可与运动向量一起存储,或者可被存储在与其中存储运动向量的存储器不同的存储器中。在其中标记信息和运动向量被存储在不同存储器中的情况下,当识别在哪个块尺寸中存储特定运动向量时,可允许标记信息的任意存取。另外,通过序列参数集、画面参数集、片段报头等来传送关于在哪个块尺寸中存储一些运动向量的信息,由此使得可能允许解码器执行与编码器的操作类似的操作。

[0146] 在其中量化并存储运动向量的情况下,关于运动向量的精度的信息可被添加并存储在存储器中。例如,在其中量化的步长尺寸是4的情况下,可附加存储标记1,并且在其中量化的步长尺寸是1的情况下,可附加存储标记0。在该情况下,标记信息可与运动向量一起存储,或者可被存储在与其中存储运动向量的存储器不同的存储器中。在其中标记信息和运动向量被存储在不同存储器中的情况下,当识别按照哪个步长尺寸来量化特定运动向量并存储时,可允许标记信息的任意存取。另外,通过序列参数集、画面参数集、片段报头等来传送关于按照哪个步长尺寸来量化一些运动向量并存储的信息,由此使得可能允许解码器执行与编码器的操作类似的操作。

[0147] 此外,在其中将运动信息存储在存储器中的情况下,可降低运动向量的空间分辨率并存储。这里,运动信息包括在帧间预测时需要的指示参考画面索引、运动向量、单向、双向等的帧间预测模式信息、参考画面列表、和关于执行帧内预测模式还是执行帧间预测模式的预测模式信息中的至少一个。

[0148] 例如,特定区域的多个运动信息之中的具有最大分区尺寸的预测单元的运动信息可作为代表性运动信息存储在存储器中。这里,特定区域可包括编码树块中的区域和编码树块的相邻块的区域。另外,该特定区域可以是在其中按照预定尺寸划分整个画面或片段的情况下、包括其中存储运动信息的块的区域。

[0149] 例如,在从该特定区域中包括的多个运动信息中排除按照运动信息合并方法、编码信息跳过方法等编码的运动信息之后,可将该代表性运动信息存储在存储器中。

[0150] 例如,该特定区域中包括的多个运动信息之中的最频繁生成的运动信息可作为代表性运动信息存储在存储器中。在该情况下,可计算对于块的每一尺寸等的运动信息的生成次数。

[0151] 例如,可存储该特定区域中包括的多个运动信息之中的特定位置处的运动信息。这里,作为特定区域中包括的位置的特定位置可以是特定区域的固定位置。另外,该特定位置可以被选择为多个位置之一。当使用多个位置时,可确定每一位置的优先级,并且可根据该优先级在存储器中存储该运动信息。

[0152] 例如,当该特定区域中包括的多个运动信息被存储在存储器中时,由于在按照帧内预测模式编码的块、按照脉冲编码调制(PCM)模式编码的块、片段、或画面的边界外部不

存在运动信息,所以对应位置的运动信息可以不存储在存储器中。

[0153] 在上述示例中,当存储特定位置的运动信息时,在其中不存在对应位置的运动信息的情况下,可使用并置块的运动信息、已编码的块的运动信息、或相邻块的运动信息作为对应位置的运动信息。这里,特定位置可以是相邻块中的一个样本位置或者该块的位置。例如,在其中不存在特定位置的运动信息的情况下,按照帧间预测编码的相邻块的运动信息之中的中间值或平均值可被存储在存储器中。例如,在其中不存在特定位置的运动信息的情况下,相邻块的运动信息的平均值可被存储在存储器中。当计算中间值和平均值时,在其中相邻块的运动信息与参考画面索引、参考画面列表、和帧间预测模式信息中的至少一个不同的情况下,可根据参考画面索引、参考画面列表、帧间预测模式信息、画面顺序计数等,来调整运动向量的尺寸。

[0154] III. 导出运动向量的处理

[0155] 在其中使用上述运动信息方法在存储器中存储运动信息、并且按照运动向量预测方法、先进运动向量预测方法、或运动信息合并方法使用参考画面的运动信息的情况下,可取得存储的运动信息。

[0156] 例如,可取得参考画面中与编码树块的位置对应的位置的运动信息。在该情况下,参考画面中与编码树块的位置对应的位置可以是特定区域中的固定位置、或者是编码树块的位置的相对位置。

[0157] 图10到13示出了从参考画面取得运动信息的示例。

[0158] 在图10到13中,块X指示编码/解码目标画面1010、1110、1210、或1310中的编码树块,并且块A、块B、块C、块D和块E指示重构的相邻块。另外,参考画面1020、1120、1220、和1320中的块T指示与编码树块对应的并置块。图13的参考画面1320中的块Y指示与除了编码/解码目标块之外的位置对应的块。

[0159] 参考图10,可取得与以下位置对应的运动信息,该位置对应于参考画面中的编码树块X的位置之中的左上像素位置。

[0160] 参考图11,可取得与以下位置对应的运动信息,该位置对应于参考画面中的编码树块X的位置之中的中间像素位置。

[0161] 参考图12,可取得与以下位置对应的运动信息,该位置对应于参考画面中的编码树块X的位置之中的右下像素位置。

[0162] 参考图13,可取得与以下位置对应的运动信息,该位置对应于参考画面中除了编码树块X之外的像素位置。

[0163] 可使用存储器中存储的运动信息(即,参考画面的运动信息),来执行诸如运动向量预测、先进运动向量预测、运动信息合并、运动信息合并跳过等的编码/解码方法。

[0164] 可使用限制运动向量的动态范围的方法、降低运动向量的空间分辨率的方法、量化运动向量的方法、和降低运动向量的代表性分辨率的方法中的至少一个,来在存储器中存储该运动向量,并且可使用存储的运动向量用于预测编码树块的运动向量并合并其运动信息。

[0165] 用于从存储器取得参考画面的运动向量的处理被称为导出时间运动向量的处理。在导出时间运动向量的处理中,TMVbitWidth指示在存储器中存储的时间运动向量的比特宽度。

[0166] 在导出时间运动向量的处理中的输入是相对于当前画面的左上样本规定当前预测单元的左上亮度(luma)样本的地点(xP,yP)、规定用于亮度的预测单元的宽度和高度的变量nPSW和nPSH、当前预测单元分区的参考索引refIdxLX(X是0或1)。处理中的输出是运动向量预测mvLXCol和可用性标记availableFlagLXCol。

[0167] 通过作为pic的参考画面RefPicListX[refidx]的画面的值PicOrderCnt,来规定函数RefPicOrderCnt(pic,refidx,LX),其中X是0或1。应维持参考画面的PicOrderCnt,直到画面被标注为“不存在”为止。Clip3(a,b,c)意味着剪切c以便在a和b之间的范围中存在的函数。

[0168] 如果slice_type等于B并且collocated_from_10_flag等于0,则变量colPic如同RefPicList1[0]所规定的那样规定包括共处一地(co-located)的分区的画面。否则(slice_type等于B并且collocated_feom_10_flag等于1或者slice_type等于P),变量colPic如同RefPicList0[0]所规定的那样规定包括共处一地的分区的画面。

[0169] 变量colPu及其位置(xPCol,yPCol)按照以下顺序步骤来导出:

[0170] 1.如等式8和9表示的那样定义当前预测单元的右下亮度位置(xPRb,yPRb)。

[0171] [等式8]

[0172] $xPRb = xP + nPSW$

[0173] [等式9]

[0174] $yPRb = yP + nPSH$

[0175] 2.如果按照帧内预测模式编码colPu或者colPu不可用,则

[0176] (1)如等式10和11表示的那样定义当前预测单元的中心亮度位置。

[0177] [等式10]

[0178] $xPCtr = (xP + (nPSW > 1) - 1)$

[0179] [等式11]

[0180] $yPCtr = (yP + (nPSH > 1) - 1)$

[0181] (2)将变量colPu设置为覆盖colPic内部的($(xPCtr > 4) < 4$, ($yPCtr > 4) < 4$)所给出的修改位置的预测单元。

[0182] (3)将(xPCol,yPCol)设置为等于相对于colPic的左上亮度样本的colPu的左上亮度样本。

[0183] 如下导出变量mvLXCol和availableFlagLXCol。

[0184] 1.如果按照帧内预测模式编码colPu或者colPu不可用,则将mvLXCol的两个分量设置为等于0,并将availableFlagLXCol设置为等于0。

[0185] 2.否则(不按照帧内预测模式编码colPu并且colPu可用),如下导出变量mvCol和refIdxCol。

[0186] (1)如果PredFlagL0[xPCol][yPCol]等于0,则将运动向量mvCol和参考索引refIdxCol设置为分别等于MvL1[xPCol][yPCol]和RefIdxL1[xPCol][yPCol]。

[0187] (2)否则(PredFlagL0[xPCol][yPCol]等于1),应用以下。

[0188] 1)如果PredFlagL1[xPCol][yPCol]等于0,则将运动向量mvCol和参考索引refIdxCol设置为分别等于MvL0[xPCol][yPCol]和RefIdxL0[xPCol][yPCol]。

[0189] 2)否则(PredFlagL1[xPCol][yPCol]等于1),应用以下。

- [0190] a. 在X为0或1的情况下进行以下分配。
- [0191] i. $\text{RefIdxColLX} = \text{RefIdxLX}[\text{xPCol}][\text{yPCol}]$
- [0192] ii. 如果 $\text{PicOrderCnt}(\text{colPic})$ 小于 $\text{PicOrderCnt}(\text{currPic})$ 并且 $\text{RefPicOrderCnt}(\text{colPic}, \text{RefIdxColLX}, \text{LX})$ 大于 $\text{PicOrderCnt}(\text{currPic})$ 或者 $\text{PicOrderCnt}(\text{colPic})$ 大于 $\text{PicOrderCnt}(\text{currPic})$ 并且 $\text{RefPicOrderCnt}(\text{colPic}, \text{RefIdxColLX}, \text{LX})$ 小于 $\text{PicOrderCnt}(\text{currPic})$, 则变量 MvXCross 等于1
- [0193] iii. 否则 ($\text{PicOrderCnt}(\text{colPic})$ 小于 $\text{PicOrderCnt}(\text{currPic})$ 并且 $\text{RefPicOrderCnt}(\text{colPic}, \text{RefIdxColLX}, \text{LX})$ 小于或等于 $\text{PicOrderCnt}(\text{currPic})$ 或者 $\text{PicOrderCnt}(\text{colPic})$ 大于 $\text{PicOrderCnt}(\text{currPic})$ 并且 $\text{RefPicOrderCnt}(\text{colPic}, \text{RefIdxColLX}, \text{LX})$ 大于或等于 $\text{PicOrderCnt}(\text{currPic})$), 则变量 MvXCross 等于0
- [0194] b. 如果以下条件之一为真, 则运动向量 mvCol 、参考索引 refIdxCol 和 ListCol 被设置为分别等于 $\text{MvL1}[\text{xPCol}][\text{yPCol}]$ 、 RefIdxColL1 和 L1
- [0195] i. Mv0Cross 等于0并且 Mv1Cross 等于1
- [0196] ii. Mv0Cross 等于 Mv1Cross 并且参考索引列表等于 L1
- [0197] c. 否则, 运动向量 mvCol 、参考索引 refIdxCol 和 ListCol 被设置为分别等于 $\text{MvL0}[\text{xPCol}][\text{yPCol}]$ 、 RefIdxColL0 和 L0
- [0198] 3) 变量 $\text{availableFlagLXCol}$ 被设置为等于1并且应用等式12或等式13的操作。
- [0199] a. 如果 $\text{PicOrderCnt}(\text{colPic}) - \text{RefPicOrderCnt}(\text{colPic}, \text{refIdxCol}, \text{ListCol})$ 等于 $\text{PicOrderCnt}(\text{currPic}) - \text{RefPicOrderCnt}(\text{currPic}, \text{refIdxLX}, \text{LX})$, 则
- [0200] [等式12]
- [0201] $\text{mvLXCol} = \text{Clip3}(-1 \ll (\text{TMVBitWidth}-1), 1 \ll (\text{TMVBitWidth}-1)-1, \text{mvCol})$
- [0202] b. 否则,
- [0203] [等式13]
- [0204] $\text{tx} = (16384 + \text{Abs}(\text{td}/2)) / \text{td}$
- [0205] [等式14]
- [0206] $\text{DistScaleFactor} = \text{Clip3}(-1024, 1023, (\text{tb} * \text{tx} + 32) \gg 6)$
- [0207] [等式15]
- [0208] $\text{mvLXCol}' = \text{Clip3}(-1 \ll (\text{TMVBitWidth}-1), 1 \ll (\text{TMVBitWidth}-1)-1, \text{mvCol})$
- [0209] [等式16]
- [0210] $\text{mvLXCol} = \text{ClipMv}((\text{DistScaleFactor} * \text{mvLXCol}' + 128) \gg 8)$
- [0211] 其中 td 和 tb 从等式17和18导出。
- [0212] [等式17]
- [0213] $\text{td} = \text{Clip3}(-128, 127, \text{PicOrderCnt}(\text{colPic}) - \text{RefPicOrderCnt}(\text{colPic}, \text{refIdxCol}, \text{ListCol}))$
- [0214] [等式18]
- [0215] $\text{tb} = \text{Clip3}(-128, 127, \text{PicOrderCnt}(\text{currPic}) - \text{RefPicOrderCnt}(\text{currPic}, \text{refIdxLX}, \text{LX}))$
- [0216] 即, 参考等式13到16, 导出 mvLXCol 作为运动向量 mvCol 的缩放 (scaled) 版本。
- [0217] 其间, 即使在动态范围中剪切运动向量, 在其中缩放剪切的运动向量的情况下, 剪

切的运动向量可再次超出该动态范围。所以,在导出缩放的运动向量之后,可限制运动向量的动态范围。在该情况下,可用等式19和20来替换等式15和16中的每一个。

[0218] [等式19]

[0219] $mvLXCol' = ClipMv((DistScaleFactor * mvLXCol + 128) >> 8)$

[0220] [等式20]

[0221] $mvLXCol = Clip3(-1 << (TMVBitWidth - 1), 1 << (TMVBitWidth - 1) - 1, mvCol')$

[0222] IV. 在解码器中传送剪切时间运动向量的信息的方法

[0223] 其后,将描述使用与编码器中的方法相同的方法来在解码器中传送剪切时间运动向量所需的信息的方法。

[0224] 导出时间运动向量的处理中的TMVBitwidth可通过序列参数集、画面参数集、片段报头等从编码器传送到解码器。

[0225] [表5]

[0226]

seq_parameter_set_rbsp() {	C	描述符
.....		
bit_width_temporal_motion_vector_minus8	0	se(v)
.....		
rbsp_trailing_bits()	0	
}		

[0227] 表5的bit_width_temporal_motion_vector_minus8规定该时间运动向量分量的比特宽度。当不存在bit_width_temporal_motion_vector_minus8时,应推断其等于0。如下规定时间运动向量分量的比特宽度:

[0228] [等式21]

[0229] $TMVBitWidth = bit_width_temporal_motion_vector_minus8 + 8$

[0230] 1. 信息传送方法1-在其中压缩运动向量并且限制运动向量的比特深度的情况下

[0231] [表6]

[0232]

seq_parameter_set_rbsp() {	C	描述符
.....		
motion_vector_buffer_comp_flag	0	u(1)
if(motion_vector_buffer_comp_flag)		
motion_vector_buffer_comp_ratio_log2	0	u(8)
bit_depth_temporal_motion_vector_constraint_flag	0	u(1)
if(bit_depth_temporal_motion_vector_constraint_flag)		
bit_depth_temporal_motion_vector_minus8	0	se(v)
.....		
rbsp_trailing_bits()	0	
}		

[0233] 参考表6,在其中motion_vector_buffer_comp_flag等于1的情况下,规定应用运

动向量缓冲器压缩处理。

[0234] motion_vector_buffer_comp_ratio_log2规定该运动向量缓冲器压缩处理中的压缩比。当不存在motion_vector_buffer_comp_ratio_log2时,应推断其等于0。如下规定该运动向量缓冲器压缩比:

[0235] [等式22]

[0236] $MVBufferCompRatio = 1 \ll \text{motion_vector_buffer_comp_ratio_log2}$

[0237] 再次参考表6,在其中bit_depth_temporal_motion_vector_constraint_flag等于1的情况下,规定应用时间运动向量比特深度限制约束处理。

[0238] bit_depth_temporal_motion_vector_minus8规定时间运动向量的比特深度。当不存在bit_depth_temporal_motion_vector_minus8时,应推断其等于0。如下规定时间运动向量的比特深度:

[0239] [等式23]

[0240] $TMVBitDepth = \text{bit_depth_temporal_motion_vector_minus8} + 8$

[0241] 2.信息传送2-在其中限制运动向量的比特深度的情况下

[0242] [表7]

[0243]

seq_parameter_set_rbsp() {	C	描述符
.....		
bit_depth_temporal_motion_vector_constraint_flag	0	u(1)
if(bit_depth_temporal_motion_vector_constraint_flag)		
bit_depth_temporal_motion_vector_minus8	0	se(v)
.....		
rsbp_trailing_bits()	0	
}		

[0244] 参考表7,在其中bit_depth_temporal_motion_vector_constraint_flag等于1的情况下,规定应用时间运动向量比特深度约束处理。

[0245] bit_depth_temporal_motion_vector_minus8规定时间运动向量的比特深度。当不存在bit_depth_temporal_motion_vector_minus8时,应推断其等于0。如下规定时间运动向量的比特深度:

[0246] [等式24]

[0247] $TMVBitDepth = \text{bit_depth_temporal_motion_vector_minus8} + 8$

[0248] 3.信息传送方法3-在其中限制运动向量的比特深度的情况下

[0249] [表8]

[0250]

seq_parameter_set_rbsp() {	C	描述符
.....		
bit_depth_temporal_motion_vector_minus8	0	se(v)
.....		
rsbp_trailing_bits()	0	

}		
---	--	--

[0251] bit_depth_temporal_motion_vector_minus8规定时间运动向量的比特深度。当不存在bit_depth_temporal_motion_vector_minus8时,应推断其等于0。如下规定时间运动向量的比特深度:

[0252] [等式25]

[0253] $TMVBitDepth = bit_depth_temporal_motion_vector_minus8 + 8$

[0254] 4. 信息传送方法4-在其中针对运动向量的X和Y分量中的每一个限制比特深度的情况下

[0255] [表9]

[0256]

seq_parameter_set_rbsp() {	C	描述符
.....		
bit_depth_temporal_motion_vector_constraint_flag	0	u(1)
if (bit_depth_temporal_motion_vector_constraint_flag) {		
bit_depth_temporal_motion_vector_x_minus8	0	se(v)
bit_depth_temporal_motion_vector_y_minus8	0	se(v)
}		
.....		
rbp_trailing_bits()	0	
}		

[0257] 参考表9,在其中bit_depth_temporal_motion_vector_constraint_flag等于1的情况下,规定应用时间运动向量比特深度约束处理。

[0258] bit_depth_temporal_motion_vector_x_minus8规定时间运动向量分量x的比特深度。当不存在bit_depth_temporal_motion_vector_x_minus8时,应推断其等于0。如下规定时间运动向量分量x的比特深度:

[0259] [等式26]

[0260] $TMVXBitDepth = bit_depth_temporal_motion_vector_x_minus8 + 8$

[0261] bit_depth_temporal_motion_vector_y_minus8规定时间运动向量分量y的比特深度。当不存在bit_depth_temporal_motion_vector_y_minus8时,应推断其等于0。如下规定时间运动向量分量y的比特深度:

[0262] [等式27]

[0263] $TMVYBitDepth = bit_depth_temporal_motion_vector_y_minus8 + 8$

[0264] 5. 信息传送方法5-在其中压缩运动向量并限制运动向量的比特深度的情况下

[0265] [表10]

[0266]

seq_parameter_set_rbsp() {	C	描述符
.....		
motion_vector_buffer_comp_flag	0	u(1)
if (motion_vector_buffer_comp_flag) {		

motion_vector_buffer_comp_ratio_log2	0	u (8)
bit_depth_temporal_motion_vectorminus8	0	se (v)
}		
.....		
rbsp_trailing_bits()	0	
}		

[0267] 参考表10,在其中motion_vector_buffer_comp_flag等于1的情况下,规定应用运动向量缓冲器压缩处理。

[0268] motion_vector_buffer_comp_ratio_log2规定该运动向量缓冲器压缩处理中的压缩比。当不存在motion_vector_buffer_comp_ratio_log2时,应推断其等于0。如下规定该运动向量缓冲器压缩比:

[0269] [等式28]

[0270] $MVBufferCompRatio = 1 \ll \text{motion_vector_buffer_comp_ratio_log2}$

[0271] V. 通过视频编解码器的级别的动态范围的定义

[0272] 时间运动向量的动态范围可通过视频编解码器的级别来定义,而无需通过序列参数集、画面参数集、或片段报头来传送。编码器和解码器可使用级别信息,来确定运动向量的限制动态范围。

[0273] 此外,即使在这些级别中,也可以不同地定义运动向量的X和Y分量的每一个的动态范围和/或比特深度,并且可定义每一分量的最小值和最大值。

[0274] 表11和12示出了在这些级别中定义在上述导出时间运动向量的处理中的TMVBitWidth的情况的示例。

[0275] [表11]

级别号	MaxTMVBitWidth (最大时间MV分量比特宽度)
1	8
1b	8
1.1	8
1.2	8
1.3	8
2	8
2.1	8
2.2	8
3	8
3.1	10
3.2	10
4	10
4.1	10
4.2	10
5	10
5.1	10

[0276] 参考表11,将TMVBitWidth设置为在这些级别中定义的MaxTMVBitWidth。这里,当将时间运动向量存储在存储器中时,MaxTMVBitWidth指示时间运动向量的最大比特宽度。

[0278] 其间,也可以按照级别来定义TMVBitWidth,并且可通过序列参数集、画面参数集、

或片段报头来传送与定义的值之间的差(德尔塔值)。即, TMVBitWidth可被设置为通过向按照级别定义的MaxTMVBitWidth添加通过序列参数集、画面参数集、或片段报头所传送的差而获得的值。这里, TMVBitWidth指示当将时间运动向量存储在存储器中时该时间运动向量的比特宽度。

[0279] [表12]

级别号	MaxTMVBitWidth (最大时间MV分量比特深度)
1	8
1b	8
1.1	8
1.2	8
1.3	8
2	8
2.1	8
2.2	8
3	8
3.1	10
3.2	10
4	10
4.1	10
4.2	10
5	10
5.1	10

[0281] [表13]

[0282]

seq_parameter_set_rbsp() {	C	描述符
.....		
delta_bit_width_temporal_motion_vector_minus8	0	se(v)
.....		
rbp_trailing_bits()	0	
}		

[0283] delta_bit_width_temporal_motion_vector_minus8规定时间运动向量分量的德尔塔比特宽度。当不存在delta_bit_width_temporal_motion_vector_minus8时, 其应被推断为等于0。如下规定时间运动向量分量的比特宽度:

[0284] [等式29]

[0285] $TMVBitWidth = \text{delta_bit_width_temporal_motion_vector_minus8} + \text{MaxTMVBitWidth}$

[0286] 另外, 如表14中所示, 也可按照级别来定义时间运动向量的每一分量的动态范围。

[0287] [表14]

[0288]

级别号	MaxMbps (最大宏块处理速率) (MB/s)	MaxFS (最大画面尺寸) (MBs)	MaxDpbMbs (最大解码画面缓冲器尺寸) (MBs)	MaxBR (最大视频比特率) 1000比特/秒、 2000比特/秒、 或 cpbBrVclFactor 比特/秒、或 cpbBrNalFactor 比特/秒)	MaxCPB (最大CPB尺寸) (1000比特、 1200比特、 cpbBrVclFactor 比特、或 cpbBrNalFactor 比特)	MaxTimeR (最大时间 MV分量 范围) 亮度画面 样本)	MinCR (最小压缩比率)	MaxMvsPer2Mbs (每两个连续MB的运动向量的最大数目)
1	1 485	99	996	64	175	[164, +63.75]	2	-
1b	1 485	99	996	128	350	[164, +63.75]	2	-
1.1	3 000	996	900	192	530	[128, +127.75]	2	-
1.2	6 000	996	2 376	384	1 000	[128, +127.75]	2	-
1.3	11 880	996	2 376	768	2 000	[128, +127.75]	2	-
2	11 880	996	2 376	2 000	2 900	[128, +127.75]	2	-
2.1	19 800	792	4 752	4 000	4 000	[128, +255.75]	3	-
2.2	20 290	1 620	8 100	4 000	4 000	[128, +255.75]	2	-
3	40 500	1 620	8 100	10 000	10 000	[128, +255.75]	2	32
3.1	108 000	3 600	18 000	24 000	24 000	[112, +511.75]	4	18
3.2	216 000	5 120	20 480	20 000	20 000	[112, +511.75]	4	18
4	345 760	8 192	32 768	20 000	20 000	[112, +511.75]	4	18
4.1	345 760	8 192	32 768	80 000	80 000	[112, +511.75]	3	18
4.2	522 240	8 704	34 816	80 000	80 000	[112, +511.75]	2	18
5	589 624	22 080	110 400	135 000	135 000	[112, +511.75]	2	18
5.1	983 040	36 864	184 320	240 000	240 000	[112, +511.75]	2	18

[0289] 另外,如表15到17中所示,也可按照级别来定义时间运动向量的每一分量的比特宽度。

[0290] [表15]

级别号	MaxMbps (最大宏块处理速率) (MB/s)	MaxFS (最大画面尺寸) (MBs)	MaxDpbMbs (最大解码画面缓冲器尺寸) (MBs)	MaxBR (最大视频比特率) 1000比特/秒、 2000比特/秒、 或 cpbBrVclFactor 比特/秒、或 cpbBrNalFactor 比特/秒)	MaxCPB (最大CPB尺寸) (1000比特、 1200比特、 cpbBrVclFactor 比特、或 cpbBrNalFactor 比特)	MaxTimeR (最大时间 MV分量 比特宽度)	MinCR (最小压缩比率)	MaxMvsPer2Mbs (每两个连续MB的运动向量的最大数目)
1	1 485	99	996	64	175	8	2	-
1b	1 485	99	996	128	350	8	2	-
1.1	3 000	996	900	192	530	8	2	-
1.2	6 000	996	2 376	384	1 000	8	2	-
1.3	11 880	996	2 376	768	2 000	8	2	-
2	11 880	996	2 376	2 000	2 900	8	2	-
2.1	19 800	792	4 752	4 000	4 000	8	2	-
2.2	20 290	1 620	8 100	4 000	4 000	8	2	-
3	40 500	1 620	8 100	10 000	10 000	8	2	32
3.1	108 000	3 600	18 000	24 000	24 000	10	4	18
3.2	216 000	5 120	20 480	20 000	20 000	10	4	18
4	345 760	8 192	32 768	20 000	20 000	10	4	18
4.1	345 760	8 192	32 768	80 000	80 000	10	3	18
4.2	522 240	8 704	34 816	80 000	80 000	10	2	18
5	589 624	22 080	110 400	135 000	135 000	10	2	18
5.1	983 040	36 864	184 320	240 000	240 000	10	2	18

[0292] [表16]

[0293]

级别号	MaxMBPS (最大宏块处理速率) (MB/s)	MaxFS (最大画面尺寸) (MBs)	MaxDpbMbs (最大解码画面缓冲器尺寸) (MBs)	MaxBR (最大视频比特率) 1000比特/秒、 2000比特/秒、 pbrVclFactor 比特/秒、或 pbrNalFactor 比特/秒)	MaxCPB (最大CPB尺寸) (1000比特、 1200比特、 pbrVclFactor 比特、或 pbrNalFactor 比特)	MaxTMVBitWidth (最大时间MV分量 比特宽度)	MinCR (最小压缩比率)	MaxMvsPer2Mb (每两个连续MB的运动向量的最大数目)
3	1 485	99	996	64	375	8	2	-
1b	1 485	99	996	128	380	8	2	-
1.1	3 020	996	900	192	900	7	2	-
1.2	6 050	996	2 376	384	2 900	7	2	-
1.3	12 880	996	2 376	768	2 900	7	2	-
2	12 880	996	2 376	2 900	2 900	7	2	-
2.1	19 800	792	4 752	4 900	4 900	8	2	-
2.2	20 250	1 620	8 100	4 900	4 900	8	2	-
3	40 500	1 620	8 100	10 000	10 000	8	2	32
3.1	106 000	5 800	16 000	14 900	14 900	10	4	36
3.2	216 000	5 120	20 480	20 000	20 000	10	4	36
4	240 760	6 192	21 768	20 000	25 000	10	4	36
4.1	240 760	8 192	32 768	36 000	62 500	10	2	36
4.2	312 240	8 704	34 816	50 000	62 500	10	2	36
5	589 824	22 080	113 400	135 000	135 000	10	2	36
5.1	892 040	36 864	184 320	240 000	240 000	10	2	36

[0294] [表17]

[0295]

级别号	MaxTMVBitWidth (最大时间MV分量比特宽度)
1	8
1b	8
1.1	8
1.2	8
1.3	8
2	8
2.1	8
2.2	8
3	8
3.1	10
3.2	10
4	10
4.1	10
4.2	10
5	10
5.1	10

[0296] 另外,如表18中所示,也可按照级别来定义时间运动向量的Y分量的比特宽度。

[0297] [表18]

[0298]

级别号	MaxMSPS (最大宏块处理速率) (MB/s)	MaxFS (最大画面尺寸) (MBs)	MaxCpbMbs (最大解码画面缓冲器尺寸) (MBs)	MaxBR (最大视频比特率) (1000比特/秒、1200比特/秒、cphBrVclFactor比特/秒、或cphBrNalFactor比特/秒)	MaxCPB (最大CPB尺寸) (1000比特、1200比特、cphBrVclFactor比特、或cphBrNalFactor比特)	MaxTMVBitWidth (最大垂直时间MV分量比特深度)	MinCR (最小压缩比率)	MaxMvsPer2MB (每两个连续MB的逆运动向量的最大数目)
1	1 485	96	396	64	175	8	2	-
1b	1 485	96	396	128	350	8	2	-
1.1	3 000	396	900	192	500	8	2	-
1.2	6 000	396	2 376	384	1 000	8	2	-
1.3	11 880	396	2 376	768	2 000	8	2	-
2	11 880	396	2 376	1 000	2 000	8	2	-
2.1	18 880	792	4 752	4 000	4 000	8	2	-
2.2	20 290	1 620	8 100	4 000	4 000	8	2	-
3	40 500	1 620	8 100	10 000	10 000	8	2	32
3.1	108 000	3 800	18 000	14 000	14 000	10	4	16
3.2	114 000	5 120	20 480	10 000	20 000	10	4	16
4	245 760	8 192	32 768	20 000	25 000	10	4	16
4.1	245 760	8 192	32 768	30 000	62 800	10	2	16
4.2	622 040	8 704	34 816	30 000	62 800	10	2	16
5	398 624	13 080	110 400	130 000	130 000	10	2	16
5.1	398 640	18 364	164 320	240 000	240 000	10	2	16

[0299] 另外,时间运动向量的动态范围可以在无需传送关于运动向量的限制的信息的情况下被定义为编码器和解码器之间预定义的固定值,或者可以按照固定比特深度的形式来存储。

[0300] 在其中TMVBitWidth被固定为相同值并在编码器和解码器中使用的情况下, TMVBitWidth可以是诸如4、6、8、10、12、14、16等的正整数。这里, TMVBitWidth指示当将时间运动向量存储在存储器中时该时间运动向量的比特宽度。

[0301] 图14是示出了根据本发明示范实施例的用于编码图像的方法的流程图。参考图14,用于编码图像的方法包括剪切步骤(S1410)、存储步骤(S1420)、和编码步骤(S1430)。

[0302] 用于编码图像的设备 and/或用于解码图像的设备在预定动态范围中剪切参考画面的运动向量(S1410)。如上面通过“I. 剪切运动向量的处理”所描述的,动态范围之外的运动向量通过对应动态范围的最小值或最大值来表示。所以,如上面通过“IV. 在解码器中传送剪切时间运动向量的信息的方法”和“V. 通过视频编解码器的级别的动态范围的定义”所描述的,通过视频编解码器的级别、序列参数集等来限制比特深度,或者通过视频编解码器的级别来限制动态范围,由此使得可能在预定动态范围中剪切参考画面的运动向量。

[0303] 如上面通过“II. 存储运动向量的处理”所描述的,用于编码图像的设备 and/或用于解码图像的设备在缓冲器中存储所剪切的参考画面的运动向量(S1420)。运动向量可以连同重构图像一起或者单独存储在缓冲器中。

[0304] 用于编码图像的设备使用存储的参考画面的运动向量,来对编码树块的运动向量进行编码(S1430)。如上面通过“III. 导出运动向量的处理”所描述的,在HEVC中使用的先进运动向量预测方法中,可使用参考画面中在与编码树块的位置相同或对应的位置处存在的块的运动向量、以及位于编码树块周围的重构块的运动向量。所以,编码树块的运动向量也可以是参考画面的运动向量(即,时间运动向量)、以及与编码树块相邻的相邻块的运动向量。

[0305] 其间,由于可以不同地定义参考画面的运动向量的X分量和Y分量的动态范围,所以可在每一动态范围中剪切参考画面的运动向量的每一分量。

[0306] 另外,可使用压缩参考画面的运动向量的方法以及限制参考画面的运动向量的动态范围的方法。在限制参考画面的运动向量的动态范围或压缩参考画面的运动向量的情况下,可按照视频编解码器的级别、序列参数集等来定义指示动态范围和运动向量和与其相关的参数的标记。

[0307] 另外,可使用存储器中存储的运动信息(即,参考画面的运动信息),来执行诸如运动向量预测、先进运动向量预测、运动信息合并、运动信息合并跳过等编码方法。

[0308] 图15是示出了根据本发明示范实施例的用于解码图像的方法的流程图。参考图15,用于解码图像的方法包括剪切步骤(S1510)、存储步骤(S1520)、导出步骤(S1530)、和解码步骤(S1540)。

[0309] 图15的剪切步骤(S1510)和存储步骤(S1520)与使用上述“I.剪切运动向量的处理”和“II.存储运动向量的处理”的图14的剪切步骤(S1410)和存储步骤(S1420)类似。另外,图15的导出步骤(S1530)使用上述“III.导出运动向量的处理”,并与图14的编码步骤(S1430)对称。所以,将省略其详细描述。

[0310] 用于解码图像的设备使用编码树块的运动向量来执行帧间预测解码(S1540)。用于解码图像的设备可使用限制运动向量的动态范围的方法、降低运动向量的空间分辨率的方法、量化运动向量的方法、和降低运动向量的代表性分辨率的方法中的至少一个来在存储器中存储运动向量,并使用存储的运动向量,用于预测编码树块的运动向量并合并其运动信息。

[0311] 另外,用于解码图像的设备可使用存储器中存储的运动信息(即,参考画面的运动信息),来执行诸如运动向量预测、先进运动向量预测、运动信息合并、运动信息合并跳过等解码方法。

[0312] 尽管已通过由一连串步骤或块表示的流程图描述了上述示范实施例,但是本发明不限于上述步骤顺序。即,可按照与其他步骤不同的顺序或同时生成一些步骤。此外,本发明所属技术领域的技术人员可理解的是,流程图中示出的步骤是非排他的,使得可包括其他步骤,或者可删除一些步骤

[0313] 另外,上述实施例包括各个方面的示例。尽管没有描述示出各个方面的所有可能组合,但是本领域技术人员可理解的是,可进行其他组合。所以,本发明应被解释为包括属于以下权利要求的所有其他替换、修改和变型。

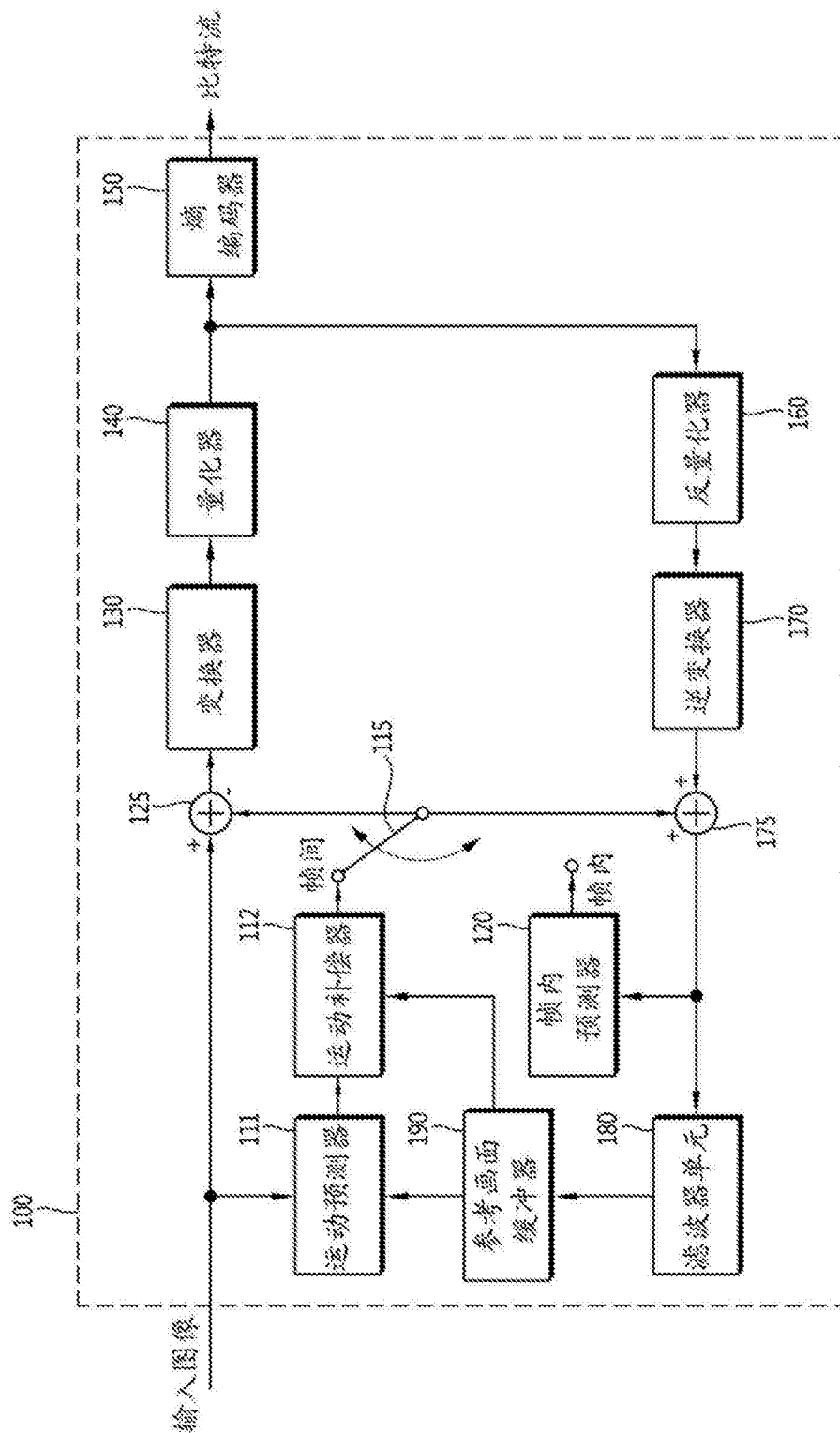


图1

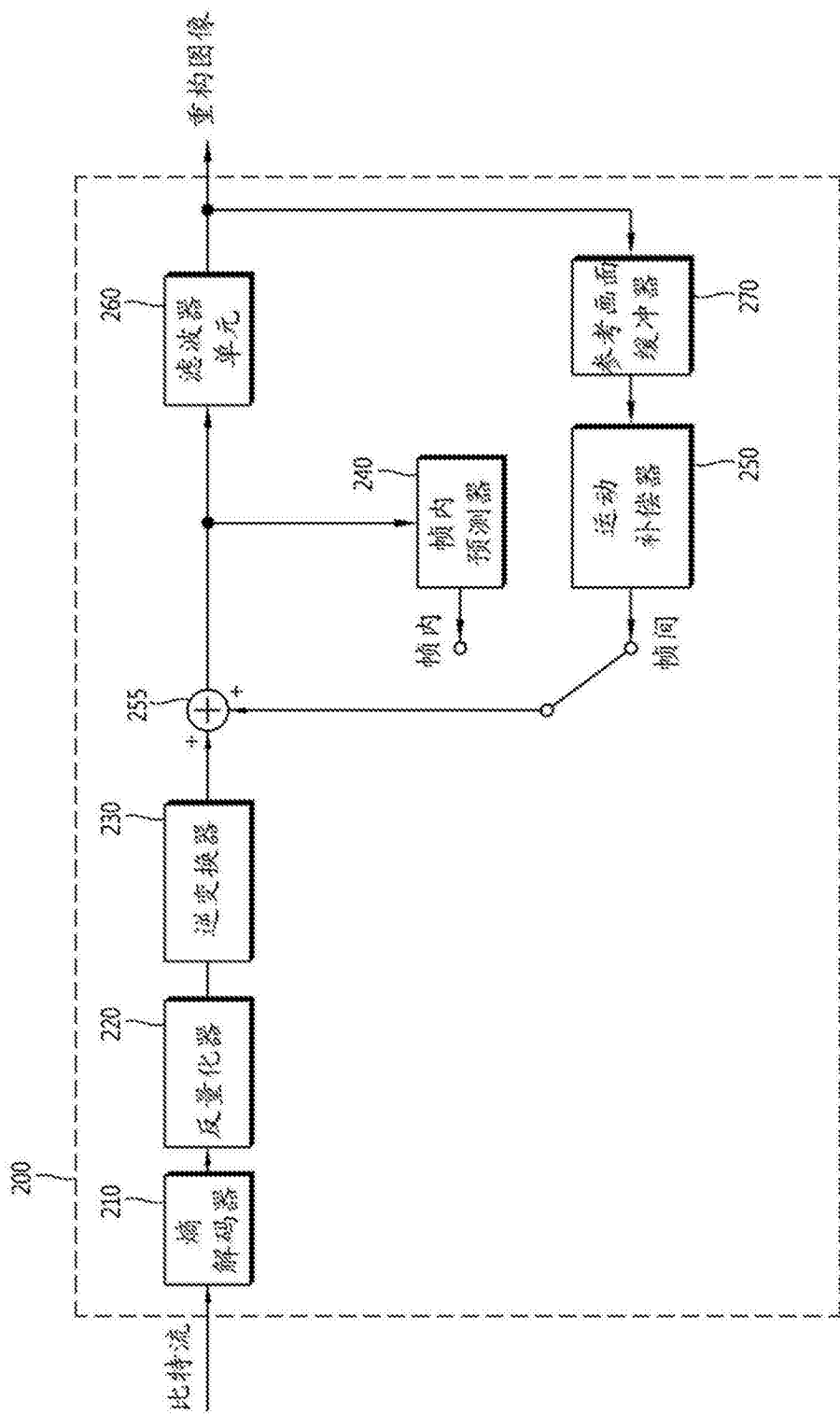


图2

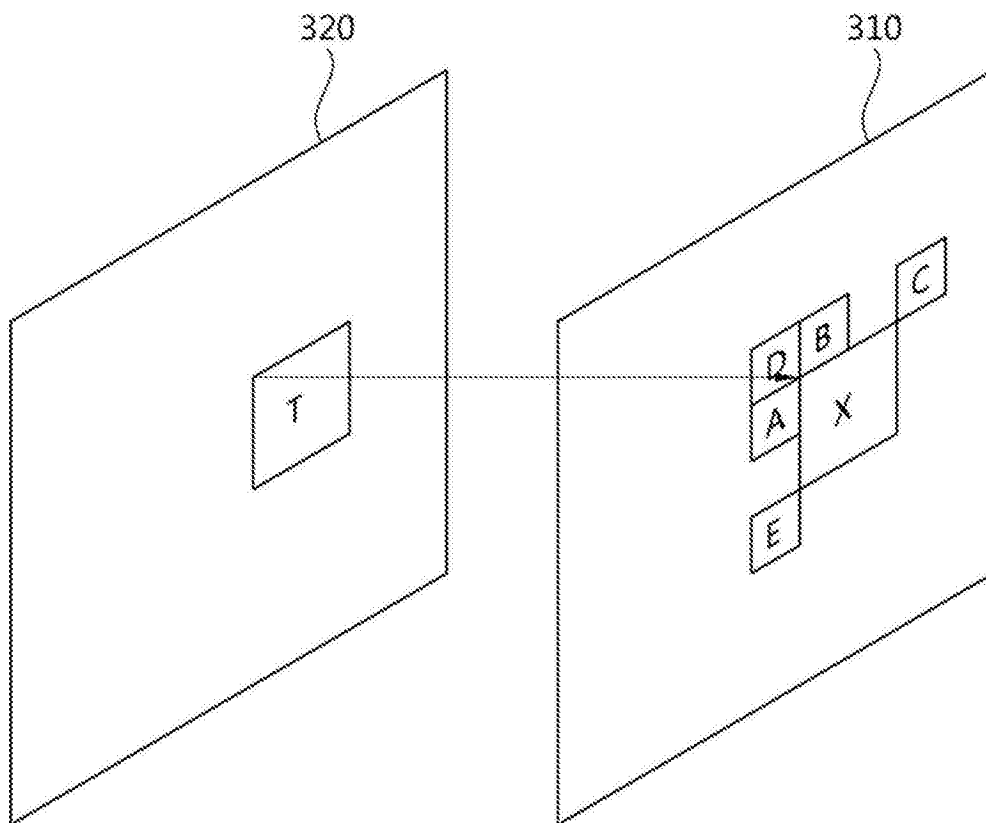


图3

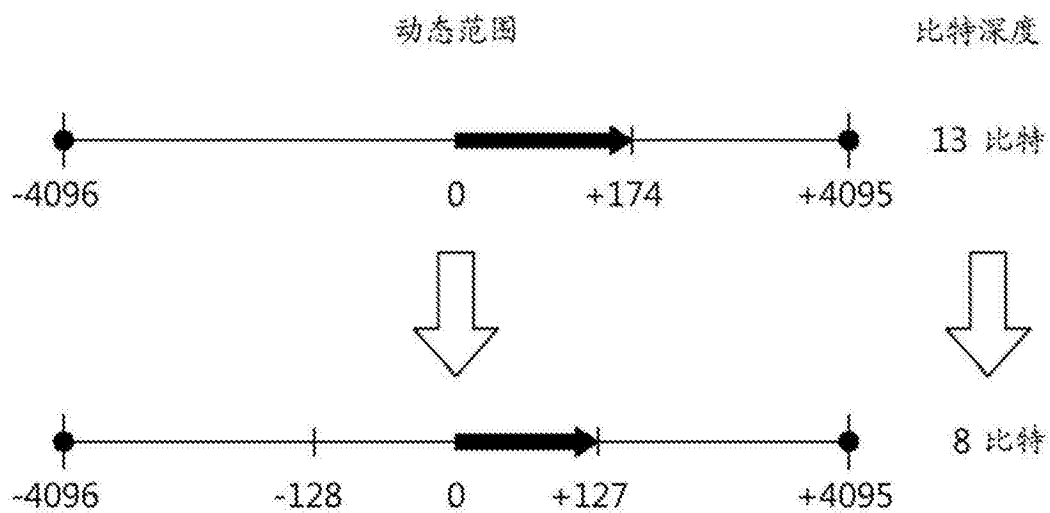


图4

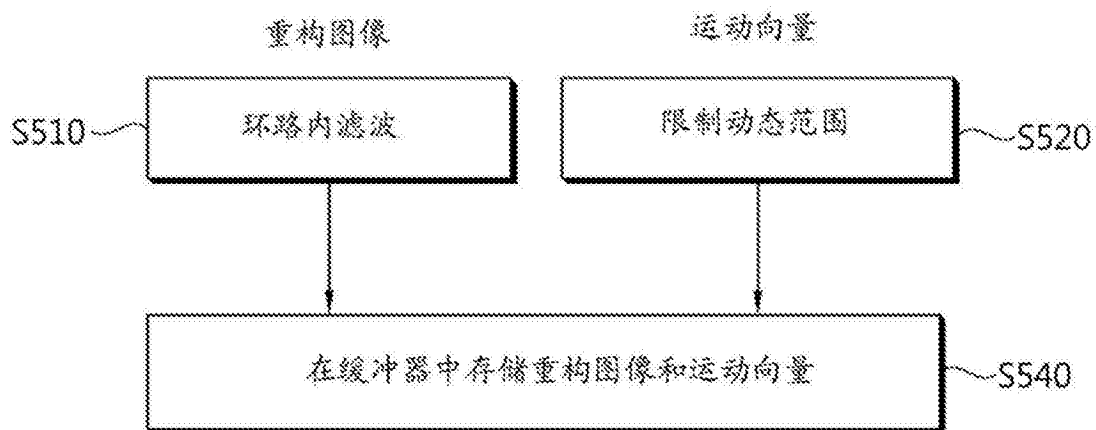


图5

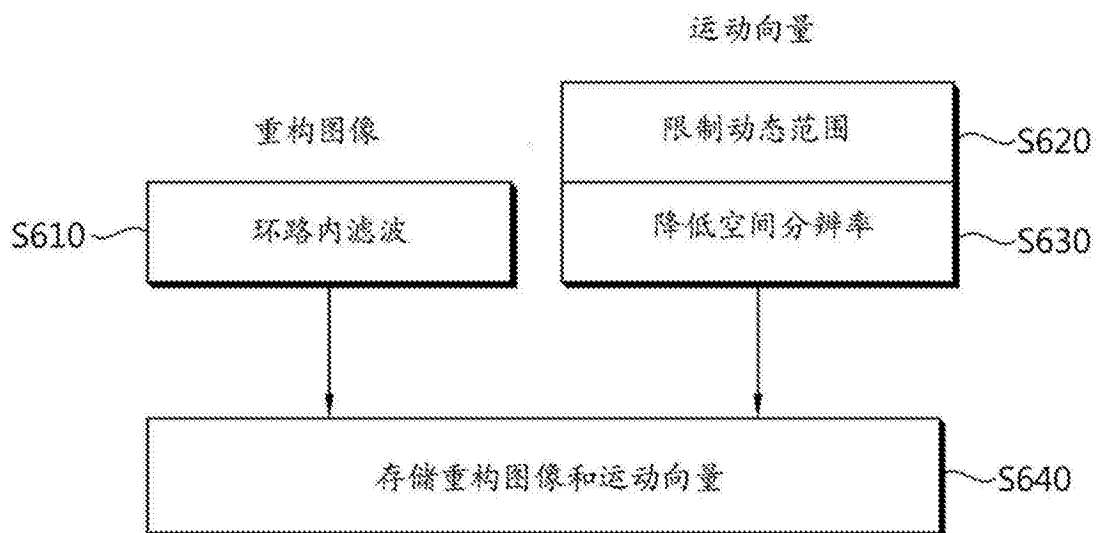


图6

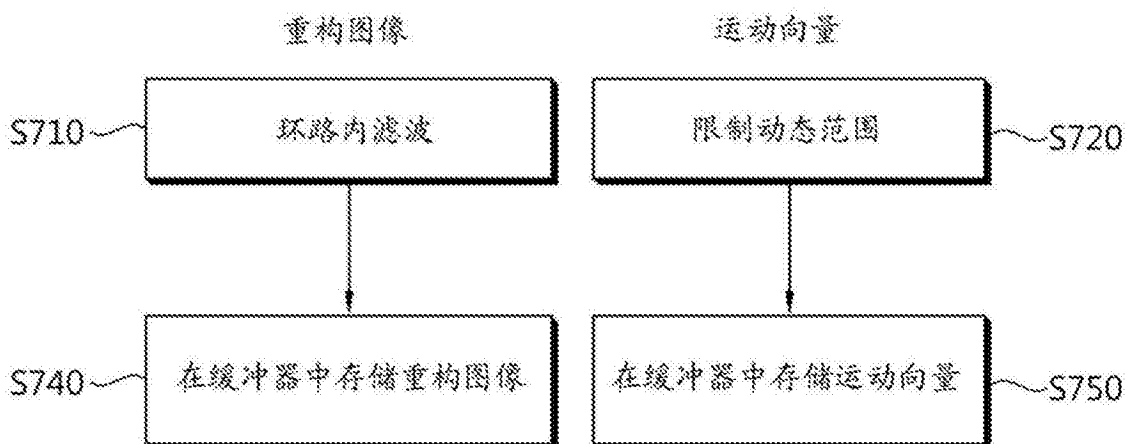


图7

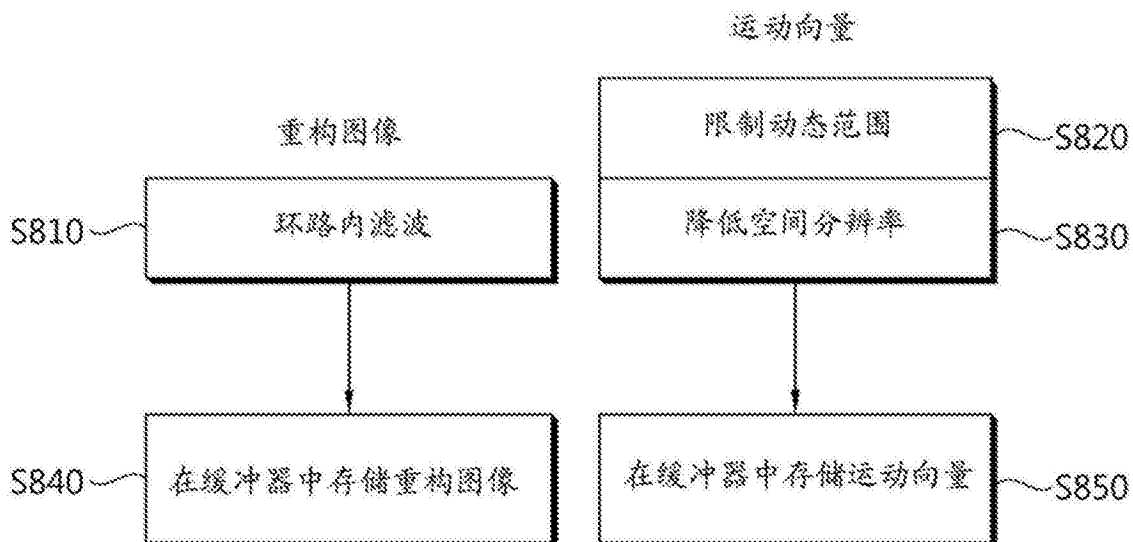


图8

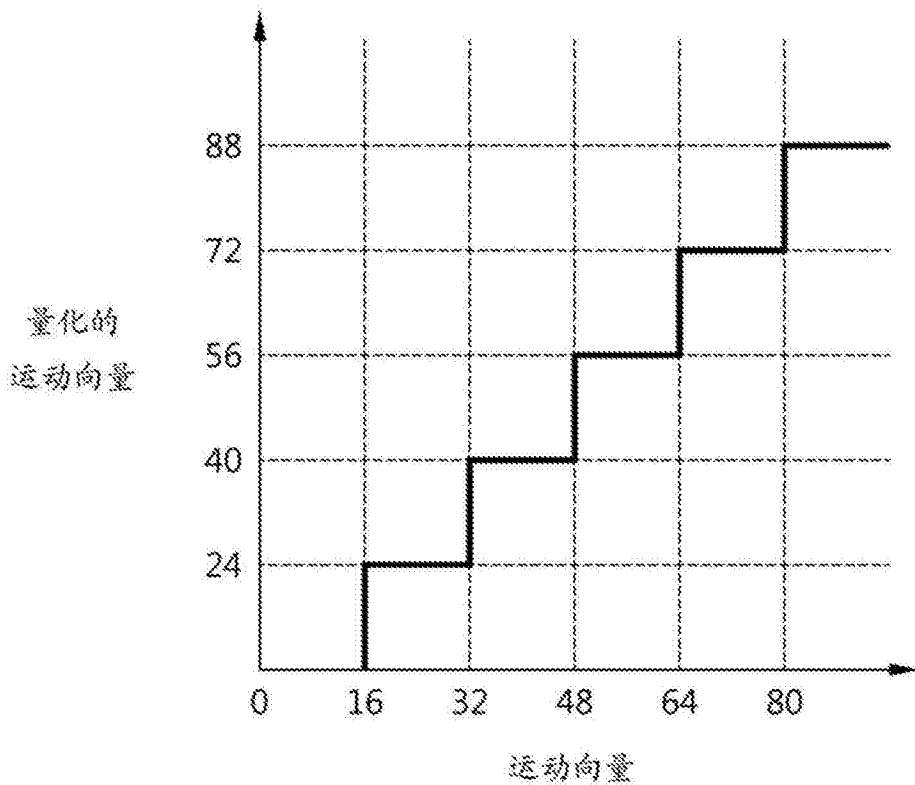


图9

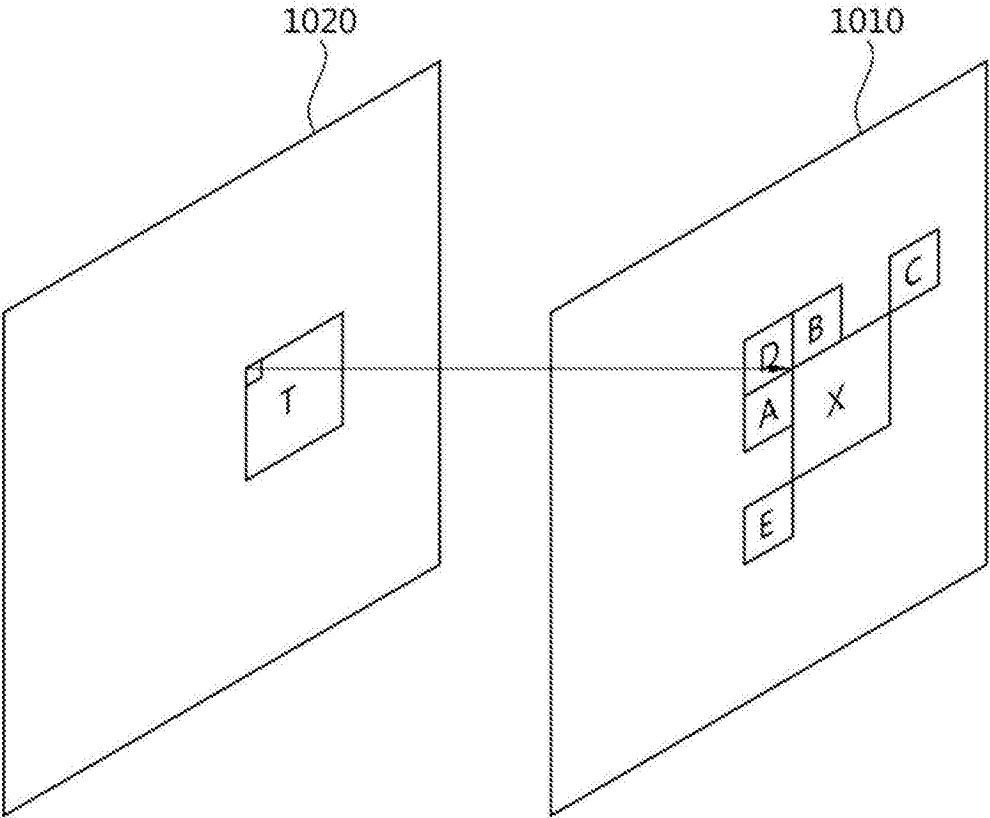


图10

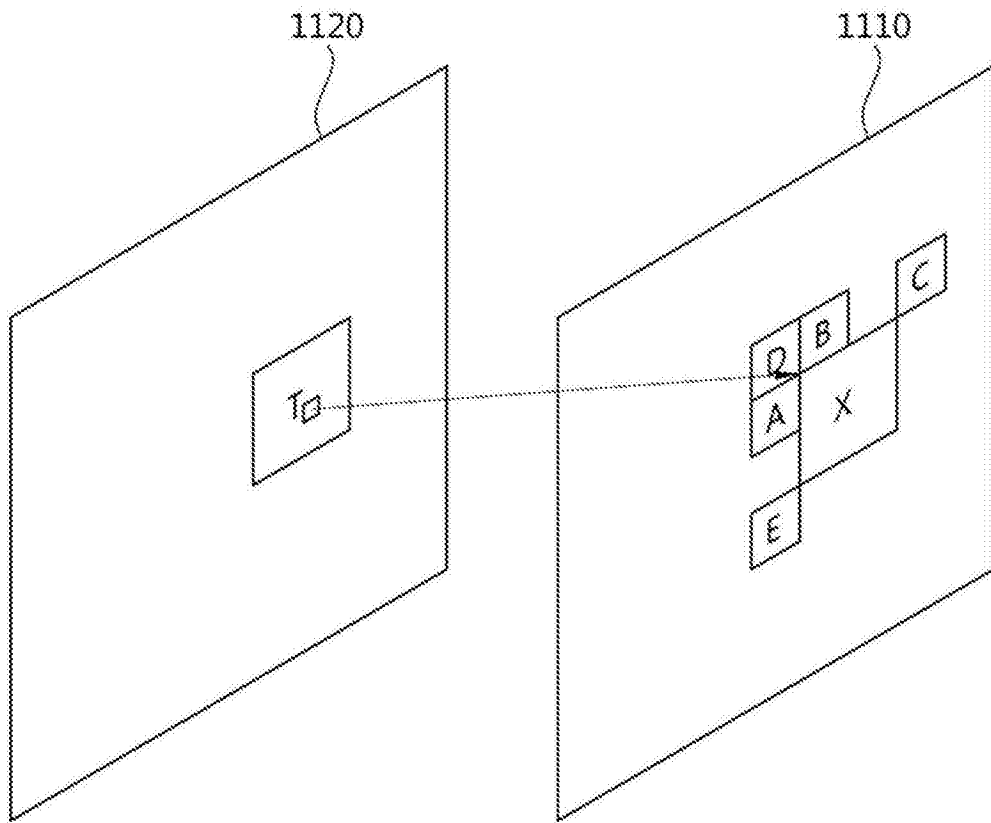


图11

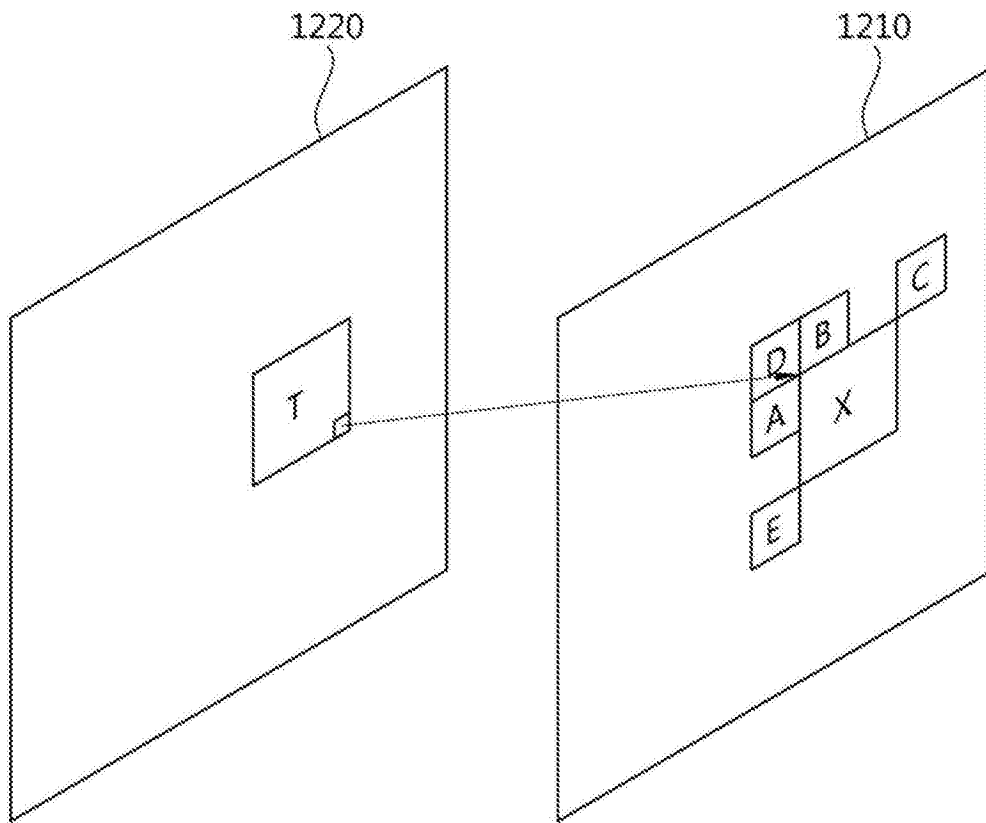


图12

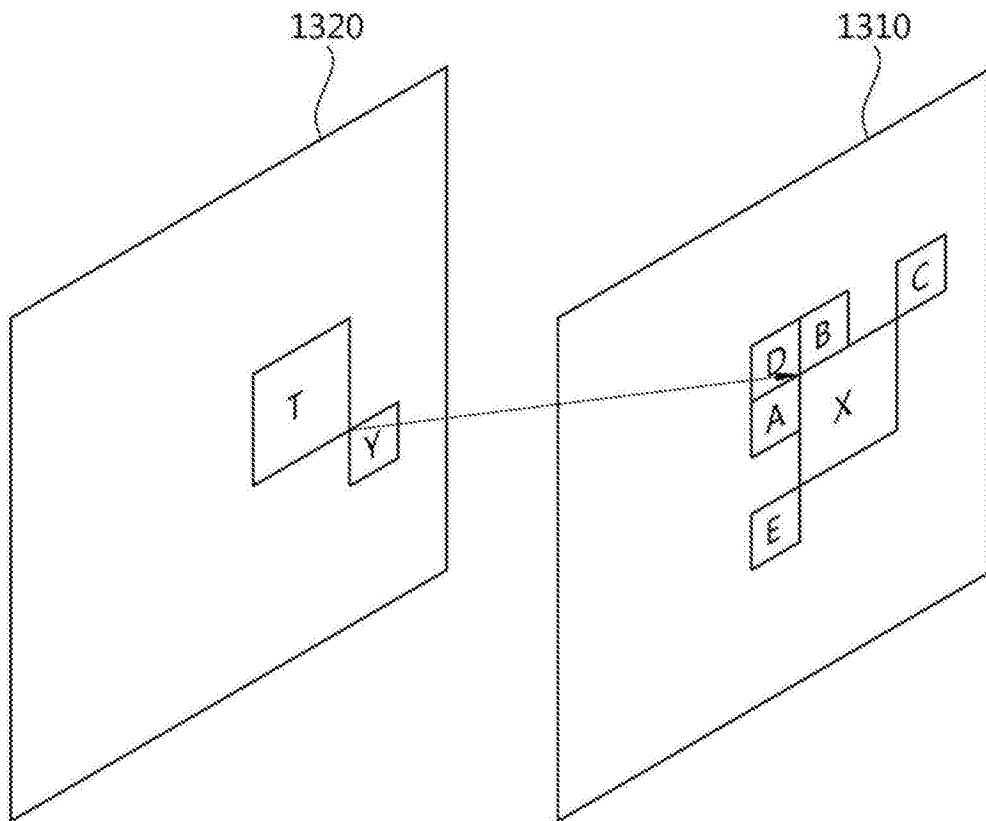


图13

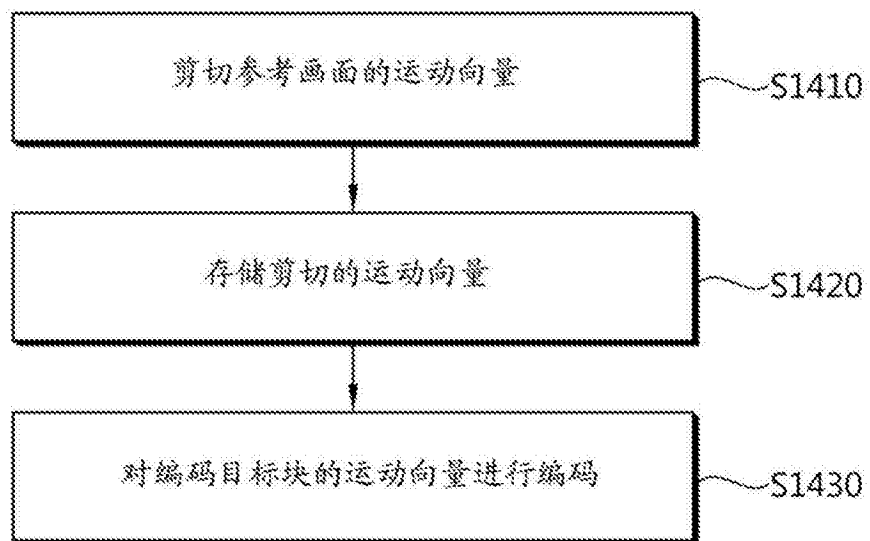


图14

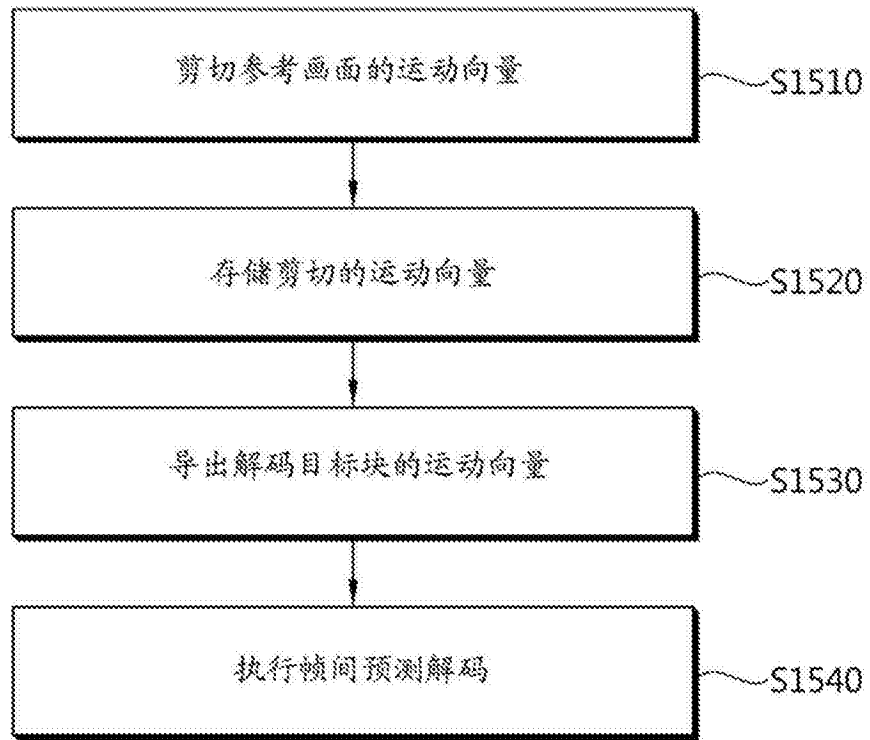


图15