



(43) 申请公布日 2024.03.15

权利要求书1页 说明书30页 附图10页

The diagram illustrates the internal structure of the frame prediction image generation unit (309). It includes the following components and data flow:

- Input:** A reference image storage unit (306) provides input to the motion compensation unit (3091).
- Frame Prediction Parameter Decoder (303):** Provides parameters $predFlagLX$, $refIdxLX$, and $mvLX, \dots$ to the motion compensation unit (3091) and the OOB processing unit (3092).
- Motion Compensation Unit (3091):** Receives input from the reference image storage unit (306) and the frame prediction parameter decoder (303). It outputs to the synthesis unit (3095).
- OOB Processing Unit (3092):** Receives input from the frame prediction parameter decoder (303) and outputs $isOOB[X]$ and $OOBMask[X]$ to the motion compensation unit (3091).
- Sub-units of OOB Processing (3092):**
 - OOB Judgment Unit (30921)
 - OOB Coding Guidance Unit (30922)
- Synthesis Unit (3095):** Receives input from the motion compensation unit (3091) and the OOB processing unit (3092). It contains:
 - Frame-to-frame Synthesis Unit (30951): Receives input from the frame prediction image generation unit (310).
 - GPM Synthesis Unit (30952): Receives input from the addition unit (312).
 - Weighted Prediction Unit (30954)
 - BIO Unit (30954)
- Output:** The synthesis unit (3095) outputs to the addition unit (312).

1. 一种运动图像解码装置, 具备: 00B判定部, 通过将参照块的坐标与图片的坐标进行比较来判定参照块是否是00B处理的对象; 以及00B掩码推导部, 通过将参照块中所包括的像素的坐标与图片边界或区域边界的坐标进行比较来推导出表示各像素可否利用的掩码数据, 所述运动图像解码装置的特征在于,

所述00B判定部使用应用于包括所述参照块的参照图片的处理的有无来判定所述参照块是否是00B处理对象。

2. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B判定部将L0参照区域的有效区域与L1参照区域的有效区域进行比较, 针对参照区域的有效区域更大的参照图片, 将00B判定设为无效。

3. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B判定部通过L0参照区域与L1参照区域的有效区域的面积的比较来判定所述有效区域的大小。

4. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B判定部通过L0参照区域与L1参照区域的有效区域的宽度或高度的比较来判定所述有效区域的大小。

5. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B判定部在应用以块为单位的加权双向预测, 即具有CU级权重的双向预测BCW的块中, 将00B判定设为无效。

6. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B掩码推导部以 $N \times M$ 像素为单位作为掩码进行推导, 其中, N 、 M 为正整数。

7. 根据权利要求1所述的运动图像解码装置, 其特征在于,

所述00B判定部在对象块为规定的尺寸以下的情况下, 将00B判定设为无效。

8. 一种运动图像编码装置, 具备: 00B判定部, 通过将参照块的坐标与图片的坐标进行比较来判定参照块是否是00B处理的对象; 以及

00B掩码推导部, 通过将参照块中所包括的像素的坐标与图片的坐标进行比较来推导出表示各像素可否利用的掩码数据, 所述运动图像编码装置的特征在于,

所述00B判定部使用应用于包括所述参照块的参照图片的处理的有无来判定所述参照块是否是00B处理对象。

运动图像解码装置以及运动图像编码装置

技术领域

[0001] 本发明的实施方式涉及运动图像解码装置以及运动图像编码装置。

背景技术

[0002] 为了高效地传输或记录运动图像,使用通过对运动图像进行编码而生成编码数据的运动图像编码装置,以及通过对该编码数据进行解码而生成解码图像的运动图像解码装置。

[0003] 作为具体的运动图像编码方式,例如可以列举出H.264/AVC、HEVC (High-Efficiency Video Coding:高效视频编码)、多功能视频编码(Versatile Video Coding: VVC)方式等。

[0004] 在这样的运动图像编码方式中,构成运动图像的图像(图片)通过分级结构来管理,并按每个CU进行编码/解码,所述分级结构包括通过分割图像而得到的切片、通过分割切片而得到的编码树单元(CTU:Coding Tree Unit)、通过分割编码树单元而得到的编码单元(有时也称为编码单元(Coding Unit:CU))以及通过分割编码单元而得到的变换单元(TU:Transform Unit)。

[0005] 此外,在这样的运动图像编码方式中,通常,基于通过对输入图像进行编码/解码而得到的局部解码图像而生成预测图像,对从输入图像(原图像)中减去该预测图像而得到的预测误差(有时也称为“差分图像”或“残差图像”)进行编码。作为预测图像的生成方法,可列举出画面间预测(帧间预测)和画面内预测(帧内预测)。在VVC中,如图8的(a)所示,能对运动矢量的坐标进行环绕式(wrap-around)处理,使水平方向的坐标系在图片的左端和右端连续地进行运动补偿。因此,在将环绕式处理应用于像360度全景图像、360度图像这样图片的左右端在空间上连续的影像时,能提高编码效率。此外,在VVC中,如图8的(b)所示,能进行以图片为单位使分辨率变化来进行运动补偿的参照图片重采样(RPR:Reference Picture Resampling)处理。在将RPR处理应用于像互联网上的运动图像分发这样的传输速率变动的服务时,能提高画质。

[0006] 此外,非专利文献1中公开了一种将包括参照图像的图片外的区域的双向预测的一方的参照块中的一部或全部的区域用另一方的参照块中的一部或全部的区域代替的OOB (Out-Of-Boundary:出界)处理技术。

现有技术文献

非专利文献

[0007] 非专利文献1

“AHG12:Enhanced bi-directional motion compensation”,Joint Video Experts Team(JVET)of ITU-T SG 16WP 3and ISO/IEC JTC 1/SC 29 25th Meeting,by teleconference,JVET-Y0125

发明内容

发明要解决的问题

[0008] 在非专利文献1中记载的方法中,在参照块包括图片外的区域的情况下,一边推导出表示图片外的区域的掩码,一边推导出双向预测图像。因此,存在计算量多的问题。

技术方案

[0009] 为了解决上述的问题,本发明的一个方案的运动图像解码装置具备:

[0010] 00B判定部,通过将参照块的坐标与图片的坐标进行比较来判定参照块是否是00B处理的对象;以及00B掩码推导部,通过将参照块中所包括的像素的坐标与图片边界或区域边界的坐标进行比较来推导出表示各像素可否利用的掩码数据,所述运动图像解码装置的特征在于,所述00B判定部使用应用于包括上述参照块的参照图片的处理的有无来判定上述参照块是否是00B处理对象。

[0011] 在本发明的一个方案的运动图像解码装置中,其特征在于,所述00B判定部将L0参照区域的有效区域与L1参照区域的有效区域进行比较,针对参照区域的有效区域较大的参照图片,将00B判定设为无效。

有益效果

[0012] 根据本发明的方案,能在运动图像编码/解码处理中削减00B处理的计算量。

附图说明

[0013] 图1是表示本实施方式的图像传输系统的构成的概略图。

图2是表示编码流的数据的分级结构的图。

图3是表示运动图像解码装置的构成的概略图。

图4是表示帧间预测图像生成部的构成的概略图。

图5是表示帧间预测参数推导部的构成的概略图。

图6是对运动图像解码装置的概略动作进行说明的流程图。

图7是表示运动图像编码装置的构成的框图。

图8是表示现有技术的环境式处理和参照图片重采样处理的示例的图。

图9是对00B处理的概略的动作进行说明的流程图。

图10是表示00B判定的示例的概率图。

具体实施方式

(第一实施方式)

[0014] 以下,参照附图对本发明的实施方式进行说明。

[0015] 图1是表示本实施方式的图像传输系统1的构成的概略图。

[0016] 图像传输系统1是传输对对象图像进行编码而得到的编码流,对所传输的编码流进行解码并显示图像的系统。图像传输系统1构成为包括:运动图像编码装置(图像编码装置)11、网络21、运动图像解码装置(图像解码装置)31以及运动图像显示装置(图像显示装置)41。

[0017] 运动图像编码装置11被输入图像T。

[0018] 网络21将运动图像编码装置11生成的编码流Te传输至运动图像解码装置31。网络

21是互联网(Internet)、广域网(WAN:Wide Area Network)、小型网络(LAN:Local Area Network,局域网)或它们的组合。网络21不一定限定于双向的通信网,也可以是传输地面数字广播、卫星广播等广播波的单向的通信网。此外,网络21也可以用DVD(Digital Versatile Disc:数字通用光盘,注册商标)、BD(Blue-ray Disc:蓝光光盘,注册商标)等记录有编码流Te的存储介质代替。

[0019] 运动图像解码装置31对网络21所传输的编码流Te分别进行解码,生成解码后的一个或多个解码图像Td。

[0020] 运动图像显示装置41显示运动图像解码装置31所生成的一个或多个解码图像Td的全部或一部分。运动图像显示装置41例如具备液晶显示器、有机EL(Electro-luminescence:电致发光)显示器等显示设备。作为显示器的形式,可以列举出固定式、移动式、HMD(Head Mounted Display:头戴显示器)等。此外,在运动图像解码装置31具有高处理能力的情况下,显示画质高的图像,在仅具有低处理能力的情况下,显示不需要高处理能力、高显示能力的图像。

<运算符>

[0021] 以下对在本说明书中使用的运算符进行描述。

[0022] >>为向右位移,<<为向左位移,&为逐位AND,|为逐位OR,|=为OR代入运算符,||表示逻辑和。

[0023] $x?y:z$ 是在x为真(0以外)的情况下取y、在x为假(0)的情况下取z的3项运算符。

[0024] Clip3(a,b,c)是将c限幅于a以上b以下的值的函数,是在 $c < a$ 的情况下返回a、在 $c > b$ 的情况下返回b、在其他情况下返回c的函数(其中 $a \leq b$)。

[0025] ClipH(o,W,x)是在 $x < 0$ 的情况下返回x,在 $x > W-1$ 的情况下返回 $x-o$,在除此以外的情况下返回x的函数。

[0026] sign(a)是在 $a > 0$ 的情况下返回1,在 $a = 0$ 的情况下返回0,在 $a < 0$ 的情况下返回-1的函数。

[0027] abs(a)是返回a的绝对值的函数。

[0028] Int(a)是返回a的整数值的函数。

[0029] floor(a)是返回a以下的最大整数的函数。

[0030] ceil(a)是返回a以上的最小整数的函数。

[0031] a/d表示a除以d(舍去小数点以下)。

<编码流Te的结构>

[0032] 在对本实施方式的运动图像编码装置11和运动图像解码装置31进行详细说明之前,对由运动图像编码装置11生成并由运动图像解码装置31进行解码的编码流Te的数据结构进行说明。

[0033] 图2是表示编码流Te中的数据的分级结构的图。编码流Te示例性地包括序列和构成序列的多张图片。图2的(a)~(f)分别是表示既定序列SEQ的编码视频序列、规定图片PICT的编码图片、规定切片S的编码切片、规定切片数据的编码切片数据、编码切片数据中所包括的编码树单元以及编码树单元中所包括的编码单元的图。

(编码视频序列)

[0034] 在编码视频序列中,规定有供运动图像解码装置31参照以便对处理对象的序列

SEQ进行解码的数据的集合。如图2所示,序列SEQ包括视频参数集(Video Parameter Set)、序列参数集SPS(Sequence Parameter Set)、图片参数集PPS(Picture Parameter Set)、自适应参数集(Adaptation Parameter Set:APS)、图片PICT以及补充增强信息SEI(Supplemental Enhancement Information)。

[0035] 视频参数集VPS在由多层构成的运动图像中,规定有多个运动图像通用的编码参数的集合,以及运动图像中包括的多层和与各层关联的编码参数的集合。

[0036] 在序列参数集SPS中,规定有供运动图像解码装置31参照以便对对象序列进行解码的编码参数的集合。例如,规定有图片的宽度、高度。需要说明的是,SPS可以存在多个。在该情况下,从PPS中选择多个SPS中的任一个SPS。

[0037] 在图片参数集PPS中,规定有供运动图像解码装置31参照以便对对象序列内的各图片进行解码的编码参数的集合。例如包括用于图片的解码的量化宽度的基准值(pic_init_qp_minus26)、指示加权预测的应用的标志(weighted_pred_flag)。需要说明的是,PPS可以存在多个。在该情况下,从对象序列内的各图片中选择多个PPS中的任一个PPS。

(编码图片)

[0038] 在编码图片中,规定有供运动图像解码装置31参照以便对处理对象的图片PICT进行解码的数据的集合。图片PICT如图2所示,包括切片0~切片NS-1(NS为图片PICT中所包括的切片的总数)。

[0039] 需要说明的是,以下,在无需对各切片0~切片NS-1进行区别的情况下,有时会省略代码的下标来进行记述。此外,对于以下说明的编码流Te中包括的数据、即标注有后缀的其他数据而言也是同样的。

(编码切片)

[0040] 在编码切片中,规定有供运动图像解码装置31参照以便对处理对象的切片S进行解码的数据的集合。切片如图2所示包括切片报头和切片数据。

[0041] 切片报头中包括供运动图像解码装置31参照以便确定对象切片的解码方法的编码参数组。指定切片类型的切片类型指定信息(slice_type)是切片报头中包括的编码参数的一个示例。

[0042] 作为能由切片类型指定信息指定的切片类型,可列举出:(1)在进行编码时仅使用帧内预测的I切片、(2)在进行编码时使用单向预测或帧内预测的P切片以及(3)在进行编码时使用单向预测、双向预测或帧内预测的B切片等。需要说明的是,帧间预测不限于单向预测、双向预测,也可以使用更多的参照图片来生成预测图像。以下,称为P、B切片的情况是指包括能使用帧间预测的块的切片。

[0043] 需要说明的是,切片报头中也可以包括对图片参数集PPS的参照(pic_parameter_set_id)。

(编码切片数据)

[0044] 在编码切片数据中,规定有供运动图像解码装置31参照以便对处理对象的切片数据进行解码的数据的集合。切片数据如图2的(d)所示包括CTU。CTU是构成切片的固定尺寸(例如64×64)的块,也称为最大编码单位(LCU:Largest Coding Unit)。

(编码树单元)

[0045] 在图2中,规定有供运动图像解码装置31参照以便对处理对象的CTU进行解码的数

据的集合。CTU通过递归的四叉树分割 (QT (Quad Tree) 分割)、二叉树分割 (BT (Binary Tree) 分割) 或三叉树分割 (TT (Ternary Tree) 分割) 分割成作为编码处理的基本单位的编码单元CU。将BT分割和TT分割统称为多叉树分割 (MT (Multi Tree) 分割)。将通过递归的四叉树分割而得到的树形结构的节点称为编码节点 (Coding Node)。四叉树、二叉树以及三叉树的中间节点为编码节点,CTU本身也被规定为最上层的编码节点。此外,最下层的编码节点被规定为编码单元。

(编码单元)

[0046] 在图2中,规定有供运动图像解码装置31参照以便对处理对象的编码单元进行解码的数据的集合。具体而言,CU由CU报头CUH、预测参数、变换参数、量化变换系数等构成。在CU报头中规定有预测模式等。

[0047] 预测处理存在以CU为单位进行的情况和以进一步分割CU而得到的子CU为单位进行的情况。在CU与子CU的尺寸相等的情况下,CU中的子CU为1个。在CU的尺寸大于子CU的尺寸的情况下,CU被分割成子CU。例如,在CU为 8×8 、子CU为 4×4 的情况下,CU被分割成4个子CU,包括水平分割的两部分和垂直分割的两部分。

[0048] 预测的种类(预测模式)有帧内预测 (MODE_INTRA)、帧间预测 (MODE_INTER) 以及帧内块复制 (MODE_IBC)。帧内预测是同一图片内的预测,帧间预测是指在互不相同的图片间(例如显示时刻间、层图像间)进行的预测处理。

[0049] 变换/量化处理以CU为单位来进行,但量化变换系数也可以以 4×4 等子块为单位来进行熵编码。

(预测参数)

[0050] 预测图像由附加于块的预测参数推导出。预测参数中存在帧内预测和帧间预测的预测参数。

(帧间预测的预测参数)

[0051] 对帧间预测的预测参数进行说明。帧间预测参数由预测列表利用标志predFlagL0和predFlagL1、参照图片索引refIdxL0和refIdxL1以及运动矢量mvL0和mvL1构成。predFlagL0、predFlagL1是表示是否使用参照图片列表(L0列表、L1列表)的标志,在值为1的情况下,使用对应的参照图片列表。需要说明的是,在本说明书中记为“表示是否为 $\times \times$ 的标志”的情况下,将标志为0以外(例如1)设为是 $\times \times$ 的情况,将标志为0设为不是 $\times \times$ 的情况,在逻辑非、逻辑积等中将1视为真,将0视为假(以下同样)。不过,在实际的装置、方法中也可以使用其他值作为真值、假值。

[0052] 用于推导帧间预测参数的语法要素例如有合并标志merge_flag (general_merge_flag)、合并索引merge_idx、merge_subblock_flag、regulare_merge_flag、ciip_flag、merge_gpm_partition_idx、merge_gpm_idx0、merge_gpm_idx1、inter_pred_idc、参照图片索引refIdxLX、mvp_LX_idx、差分矢量mvdLX、运动矢量精度模式amvr_mode。merge_subblock_flag是表示是否使用以子块为单位的帧间预测的标志。regulare_merge_flag是表示是否使用通常合并模式或MMVD的标志。ciip_flag是表示是否使用CIIP (combined inter-picture merge and intra-picture prediction:组合帧间图片合并和帧内图片预测) 模式的标志。merge_gpm_partition_idx是表示GPM模式的分割形状的索引。merge_gpm_idx0、merge_gpm_idx1是表示GPM模式的合并索引的索引。inter_pred_idc是用于选择在

AMVP模式下使用的参照图片的帧间预测标识符。mvp_LX_idx是用于推导运动矢量的预测矢量索引。

(参照图片列表)

[0053] 参照图片列表是由存储于参照图片存储器306的参照图片构成的列表。在各CU中,由refIdxLX指定实际上以参照图片列表RefPicListX(X=0或1)中的哪一张图片来进行参照。需要说明的是,LX是在不对L0预测和L1预测进行区分的情况下使用的记述方法,以下,通过将LX置换为L0、L1针对L0列表的参数和针对L1列表的参数进行区分。

(合并预测和AMVP预测)

[0054] 预测参数的解码(编码)方法中有合并预测(merge)模式(合并模式)和AMVP(Advanced Motion Vector Prediction:先进运动矢量预测、自适应运动矢量预测)模式,general_merge_flag是用于识别它们的标志。合并模式是省略运动矢量差分中的一部分或全部的预测模式,是不将预测列表利用标志predFlagLX、参照图片索引refIdxLX、运动矢量mvLX包括于编码数据而从已处理的附近块的预测参数等进行推导的模式。AMVP模式是将inter_pred_idc、refIdxLX、mvLX包括在编码数据中的模式。需要说明的是,mvLX被编码为识别预测矢量mvpLX的mvp_LX_idx和差分矢量mvdLX。此外,也可以将省略/简化运动矢量差分的预测模式的总称称为一般合并模式(general merge mode),根据general_merge_flag来选择一般合并模式和AMVP预测。

[0055] 在general_merge_flag为1的情况下,可以另行传输regular_merge_flag。也可以是,在regular_merge_flag为1的情况下,选择通常合并模式或MMVD,在除此以外的情况下选择CIIP模式或GPM模式。CIIP模式通过帧间预测图像与帧内预测图像的加权和来生成预测图像。GPM模式将对象CU作为通过线段分割的两个非矩形的预测单位来生成预测图像。

[0056] inter_pred_idc是表示参照图片的种类和数量的值,取PRED_L0、PRED_L1、PRED_BI中的任一值。PRED_L0、PRED_L1分别表示使用在L0列表、L1列表中进行管理的一张参照图片的单向预测。PRED_BI表示使用在L0列表和L1列表中进行管理的两张参照图片的双向预测。

[0057] merge_idx是表示是否将从处理已完成的块推导出的预测参数候选(合并候选)中的任一个预测参数用作对象块的预测参数的索引。

(运动矢量)

[0058] mvLX表示不同的两张图片上的块间的移位量。将与mvLX有关的预测矢量、差分矢量分别称为mvpLX、mvdLX。

(帧间预测标识符inter_pred_idc和预测列表利用标志predFlagLX)

[0059] inter_pred_idc与predFlagL0、predFlagL1的关系能如下进行相互转换。

$$\text{inter_pred_idc} = (\text{predFlagL1} \ll 1) + \text{predFlagL0}$$

$$\text{predFlagL0} = \text{inter_pred_idc} \& 1$$

$$\text{predFlagL1} = \text{inter_pred_idc} \gg 1$$

[0060] 需要说明的是,帧间预测参数可以使用预测列表利用标志,也可以使用帧间预测标识符。此外,使用了预测列表利用标志的判定可以替换为使用了帧间预测标识符的判定。反之,使用了帧间预测标识符的判定也可以替换为使用了预测列表利用标志的判定。

(双向预测biPred的判定)

[0061] 是否为双向预测的标志biPred能根据两个预测列表利用标志是否均为1而推导出。例如能通过以下算式推导出。

$$\text{biPred} = (\text{predFlagL0} == 1 \&\& \text{predFlagL1} == 1)$$

[0062] 或者,biPred也能根据帧间预测标识符是否是表示使用两个预测列表(参照图片)的值而推导出。例如能通过以下算式推导出。

$$\text{biPred} = (\text{inter_pred_idc} == \text{PRED_BI}) ? 1 : 0$$

(运动图像解码装置的构成)

[0063] 对本实施方式的运动图像解码装置31(图3)的构成进行说明。

[0064] 运动图像解码装置31构成为包括:熵解码部301、参数解码部(预测图像解码装置)302、环路滤波器305、参照图片存储器306、预测参数存储器307、预测图像生成部(预测图像生成装置)308、逆量化/逆变换部311以及加法部312、预测参数推导部320。需要说明的是,根据后述的运动图像编码装置11,也存在运动图像解码装置31中不包括环路滤波器305的构成。

[0065] 参数解码部302还具备报头解码部3020、CT信息解码部3021以及CU解码部3022(预测模式解码部),CU解码部3022还具备TU解码部3024。也可以将它们统称为解码模块。报头解码部3020从编码数据解码VPS、SPS、PPS、APS等参数集信息、切片报头(切片信息)。CT信息解码部3021从编码数据解码CT。CU解码部3022从编码数据解码CU。TU解码部3024从编码数据解码CU。

[0066] TU解码部3024在TU中包括预测误差的情况下,从编码数据解码QP更新信息和量化变换系数。QP更新信息是与作为量化参数QP的预测值的量化参数预测值qPpred的差分值。

[0067] 预测图像生成部308构成为包括帧间预测图像生成部309(图4)和帧内预测图像生成部310。

[0068] 预测参数推导部320构成为包括帧间预测参数推导部303(图5)和帧内预测参数推导部。

[0069] 此外,在下文中对将CTU、CU用作处理单位的示例进行了记载,但不限于该示例,也可以以子CU为单位进行处理。或者,也可以将CTU、CU替换为块,将子CU替换为子块,进行以块或者子块为单位的处理。

[0070] 熵解码部301对从外部输入的编码流Te进行熵解码,对各个代码(语法要素)进行解码。熵编码中存在如下方式:使用根据语法要素的种类、周围的状况而适当选择出的上下文(概率模型)对语法要素进行可变长度编码的方式;以及使用预定的表或计算式对语法要素进行可变长度编码的方式。

[0071] 熵解码部301将解码后的代码输出至参数解码部302。解码后的代码例如是指预测模式predMode、general_merge_flag、merge_idx、inter_pred_idc、refIdxLX、mvp_LX_idx、mvdLX、amvr_mode等。基于参数解码部302的指示来进行对哪一个代码进行解码的控制。

(基本流程)

[0072] 图6是对运动图像解码装置31的概略动作进行说明的流程图。

[0073] (S1100:参数集信息解码)报头解码部3020从编码数据解码VPS、SPS、PPS等参数集信息。

[0074] (S1200:切片信息解码)报头解码部3020从编码数据解码切片报头(切片信息)。

[0075] 以下,运动图像解码装置31通过针对对象图片中包括的各CTU反复进行S1300至S5000的处理来推导出各CTU的解码图像。

[0076] (S1300:CTU信息解码) CT信息解码部3021从编码数据解码CTU。

[0077] (S1400:CT信息解码) CT信息解码部3021从编码数据解码CT。

[0078] (S1500:CU解码) CU解码部3022实施S1510、S1520,从编码数据解码CU。

[0079] (S1510:CU信息解码) CU解码部3022从编码数据解码CU信息、预测信息、TU分割标志、CU残差标志等。

[0080] (S1520:TU信息解码) TU解码部3024在TU中包括预测误差的情况下,从编码数据解码量化预测误差等。

[0081] (S2000:预测图像生成) 预测图像生成部308针对对象CU中包括的各块,基于预测信息生成预测图像。

[0082] (S3000:逆量化/逆变换) 逆量化/逆变换部311针对对象CU中包括的各TU,执行逆量化/逆变换处理。

[0083] (S4000:解码图像生成) 加法部312通过将由预测图像生成部308提供的预测图像与由逆量化/逆变换部311提供的预测误差相加来生成对象CU的解码图像。

[0084] (S5000:环路滤波) 环路滤波器305对解码图像施加去块滤波、SAO (Sample Adaptive Offset:取样自适应偏移)、ALF (Adaptive Loop Filter:自适应环路滤波) 等环路滤波,生成解码图像。

[0085] 环路滤波器305是设于编码环路内的滤波器,是去除块失真、振铃失真来改善画质的滤波器。环路滤波器305对加法部312所生成的CU的解码图像实施去块滤波、取样自适应偏移(SAO)、自适应环路滤波(ALF)等滤波。

[0086] 参照图片存储器306将加法部312所生成的CU的解码图像按每个对象图片和对象CU存储于预定的位置。

[0087] 预测参数存储器307将预测参数按每个解码对象的CTU或CU存储于预定的位置。具体而言,预测参数存储器307存储由参数解码部302解码后的参数和由熵解码部301分离后的预测模式predMode等。

[0088] 预测图像生成部308被输入predMode、预测参数等。此外,预测图像生成部308从参照图片存储器306中读出参照图片。预测图像生成部308在predMode所示的预测模式下,使用预测参数和读出的参照图片(参照块)来生成块或子块的预测图像。在此,参照块是指参照图片上的像素的集合(通常为矩形,因此称为块),是为了生成预测图像而参照的区域。(帧间预测参数推导部的构成)

[0089] 如图5所示,帧间预测参数推导部303基于从参数解码部302输入的语法要素,参照存储于预测参数存储器307的预测参数来推导出帧间预测参数。此外,将帧间预测参数输出至帧间预测图像生成部309、预测参数存储器307。帧间预测参数推导部303和作为其内部要素的AMVP预测参数推导部3032、合并预测参数推导部3036、MMVD预测部30376、MV加法部3038是在运动图像编码装置、运动图像解码装置中通用的单元,因此也可以将它们统称为运动矢量推导部(运动矢量推导装置)。

[0090] 在general_merge_flag为1,即表示合并预测模式的情况下,推导出merge_idx,输出至合并预测参数推导部3036。

[0091] 在general_merge_flag为0,即表示AMVP预测模式的情况下,AMVP预测参数推导部3032从inter_pred_idc、refIdxLX或mvp_LX_idx推导出mvpLX。

(MV加法部)

[0092] 在MV加法部3038中将推导出的mvpLX与mvdLX相加,推导出mvLX。

(合并预测)

[0093] 合并预测参数推导部3036推导出构成为包括预测参数(predFlagLX、mvLX、refIdxLX)的合并候选,构成合并候选列表。而且,合并预测参数推导部3036将合并候选列表中所包括的合并候选中的merge_idx所示的合并候选N的运动信息(mvLXN[0]、mvLXN[1])、predFlagLXN、refIdxLXN选作对象块的帧间预测参数。合并预测参数推导部3036将所选出的合并候选的帧间预测参数存储于预测参数存储器307,并且输出至帧间预测图像生成部309。

(帧间预测图像生成部309)

[0094] 在predMode指示帧间预测的情况下,帧间预测图像生成部309使用从帧间预测参数推导部303输入的帧间预测参数和参照图片,通过帧间预测来生成块或子块的预测图像。

[0095] 图4是表示本实施方式的预测图像生成部308中所包括的帧间预测图像生成部309的构成的概略图。帧间预测图像生成部309构成为包括:运动补偿部(预测图像生成装置)3091、00B处理部3092以及合成部3095。合成部3095构成为包括加权预测部3094。

(运动补偿)

[0096] 运动补偿部3091(内插图像生成部3091)通过基于从帧间预测参数推导部303输入的帧间预测参数(predFlagLX、refIdxLX、mvLX),从参照图片存储器306读出参照块来生成内插图像(运动补偿图像)。参照块是在由refIdxLX指定的参照图片RefPicLX上从对象块的位置移位了mvLX的位置的块。在此,在mvLX并非整数精度的情况下,实施称为运动补偿滤波的用于生成小数位置的像素的滤波,生成内插图像。

[0097] 运动补偿部3091通过以下的算式推导出bW*bH尺寸的块的左上坐标(xPb,yPb)、预测块内坐标(xL,yL)、与运动矢量(mvLX[0],mvLX[1])对应的整数位置(xInt,yInt)以及相位(xFrac,yFrac)(MC-P1)。

$$xInt = xPb + (mvLX[0] \gg (\log2MVPREC)) + xL$$

$$xFrac = mvLX[0] \& (MVPREC - 1)$$

$$yInt = yPb + (mvLX[1] \gg (\log2MVPREC)) + yL$$

$$yFrac = mvLX[1] \& (MVPREC - 1)$$

[0098] 在此,MVPREC表示mvLX的精度(1/MVPREC像素精度)、 $\log2MVPREC = \log2(MVPREC)$ 、 $x = 0 \cdots bW - 1$ 、 $y = 0 \cdots bH - 1$ 。例如可以是MVPREC=16。

[0099] 需要说明的是,也可以如后文所述的(MC-P2)那样进行推导以进行RPR。而且,运动补偿部3091也可以对通过(MC-P1)、(MC-P2)推导出的(xInt,yInt)修正位置以进行环绕。

[0100] 在将子图片边界作为图片边界的标志有效的情况下(sps_subpic_treated_as_pic_flag=1),并且参照图像的子图片的个数大于1的情况下(用于参照图像refPicLX的sps_num_subpics_minus1>0),下式成立(MC-P2)。

$$xInt = Clip3(SubpicLeftBoundaryPos, SubpicRightBoundaryPos, refWraparoundEnabledFlag ? ClipH((PpsRefWraparoundOffset) * MinCbSizeY, picW,$$

xInt):xInt)

yInt=Clip3(SubpicTopBoundaryPos,SubpicBotBoundaryPos,yInt)

[0101] 在此,SubpicLeftBoundaryPos、SubpicRightBoundaryPos、SubpicTopBoundaryPos、SubpicBottomBoundaryPos为各子图片的左、右、上、下的边界位置。

[0102] 在上述以外的情况下(在将子图片边界作为图片边界的标志无效的情况下(sps_subpic_treated_as_pic_flag==0)或参照图像的子图片的个数为1的情况下(用于参照图像refPicLX的sps_num_subpics_minus1==0)),下式成立(MC-P2)。

xInt=Clip3(0,picW-1,refWraparoundEnabledFlag?ClipH((PpsRefWraparoundOffset)*MinCbSizeY,picW,xInt):xInt)

yInt=Clip3(0,picH-1,yInt)

[0103] 在此,如下所示地设定各参数。

refWraparoundEnabledFlag=pps_ref_wraparound_enabled_flag&&!refPicIsScaled

PpsRefWraparoundOffset=pps_pic_width_in_luma_samples/MinCbSizeY-pps_pic_width_minus_wraparound_offset

[0104] MinCbSizeY是规定的常数或变量(例如4),pps_pic_width_minus_wraparound_offset是从表示环绕的位置的编码数据解码的偏移。

[0105] 运动补偿部3091使用内插滤波器对参照图片refImg进行水平内插处理,由此推导出临时的图像temp[][]。以下的 Σ 是与 $k=0..NTAP-1$ 的 k 有关的和,mcFilter[Frac][k]是相位Frac中的第 k 个内插滤波器系数,shift1是调整值的区间的归一化参数,offset1= $1 \ll (\text{shift1}-1)$ 。

temp[x][y]=

$(\Sigma \text{mcFilter}[x\text{Frac}][k] * \text{refImg}[x\text{Int}+k-NTAP/2+1][y\text{Int}]+\text{offset1}) \gg \text{shift1}$

[0106] 接着,运动补偿部3091通过对临时的图像temp[][]进行垂直内插处理,推导出内插图像Pred[][]。以下的 Σ 是与 $k=0..NTAP-1$ 的 k 有关的和,shift2是调整值的区间的归一化参数,offset2= $1 \ll (\text{shift2}-1)$ 。

Pred[x][y]= $(\Sigma \text{mcFilter}[y\text{Frac}][k] * \text{temp}[x][y+k-NTAP/2+1]+\text{offset2}) \gg \text{shift2}$ (算式MC-1)

(加权预测)

[0107] 加权预测部3094通过将内插图像PredLX与加权系数相乘来生成块的预测图像。

(图片范围外(OOB)处理)

[0108] OOB处理部3092包括OOB判定部30921和OOB掩码推导部30922。OOB判定部30921基于从帧间预测参数推导部303输入的帧间预测参数(predFlagLX、refIdxLX、mvLX等)来判定双向预测模式的各参照块是否包括参照图片的有效范围外(OOB:Out-Of-Boundary)的区域。

[0109] 需要说明的是,上述判定也可以判定是否包括图片或子图片的有效范围外的区域。此外,也可以使用是否包括切片外的区域来进行判定。

[0110] 参照图9进行说明。

[0111] 在00B判定部30921判定为参照块的一部分的像素在图片外(有效范围外),即参照块是00B的情况下(is00B==true,S1002),00B掩码推导部30922推导出表示参照块的哪个区域在有效范围外的掩码数据(掩码值、00B掩码)(S1004)。帧间预测图像生成部309对作为00B的参照块应用00B掩码生成预测图像(S1006)。帧间预测图像生成部309对不是00B的参照块生成通常的预测图像(S1008)。以下,对详细内容进行说明。

(00B判定)

[0112] 00B判定部30921在双向预测中针对两个(第一和第二)参照块判定各参照块的一部分是否在图片外(00B)。与参照图像refPicLX对应的LX参照块(X=0或1)的区域由块左上坐标(xRefLX,yRefLX)和块尺寸(宽度和高度:bW,bH)表示。将图片的宽度和高度分别设为picW、picH。以下,根据mvLX的精度MVPREC将mvLX整数化后的值用于坐标系。00B判定部30921使用这些值,通过以下算式推导出表示LX参照块是否包括图片外(有效范围外)的真值is00B[X]。此外,将有效范围内的区域记为有效区域。

[0113] 图10表示一个示例。图10是以整数为单位的图,运动矢量mvLX'、参照像素位置(xRefLX',yRefLX')是将运动矢量mvLX、参照像素位置(xRefLX,yRefLX)整数表达的参数。图10的示例为X=0,但在X=1的情况下也是同样的。

$$\begin{aligned} xRefLX &= (xPb \ll \log 2 MVPREC) + mvLX[0] \text{ (Decide00B1)} \\ yRefLX &= (yPb \ll \log 2 MVPREC) + mvLX[1] \\ picMinX &= 0 - MVPREC / 2 \\ picMinY &= 0 - MVPREC / 2 \\ picMaxX &= ((picW - 1) \ll \log 2 MVPREC) + MVPREC / 2 \\ picMaxY &= ((picH - 1) \ll \log 2 MVPREC) + MVPREC / 2 \\ is00B[X] &= (xRefLX < picMinX) \parallel (yRefLX < picMinY) \parallel \\ &\quad (xRefLX + (bW \ll \log 2 MVPREC) - 1 > picMaxX) \parallel \\ &\quad (yRefLX + (bH \ll \log 2 MVPREC) - 1 > picMaxY) \end{aligned}$$

[0114] 也可以通过与MVPREC相乘(“*MVPREC”)对log2MVPREC比特左移位(“<<log2MVPREC”)进行处理,也可以通过“<<log2MVPREC”对后述的“*MVPREC”进行处理。

[0115] 或者,(Decide00B1)中的is00B[X]的推导式也可以为下式。

$$\begin{aligned} is00B[X] &= (xRefLX \leq picMinX) \parallel (yRefLX \leq picMinY) \parallel \\ &\quad (xRefLX + (bW \ll \log 2 MVPREC) - 1 \geq picMaxX) \parallel \\ &\quad (yRefLX + (bH \ll \log 2 MVPREC) - 1 \geq picMaxY) \end{aligned}$$

[0116] 或者,00B判定部30921也可以省略用于舍入的项,如下所示地简略进行判定。

$$\begin{aligned} picMaxX &= (picW - 1) \ll \log 2 MVPREC \text{ (Decide00B2)} \\ picMaxY &= (picH - 1) \ll \log 2 MVPREC \\ is00B[X] &= (xRefLX < 0) \parallel (yRefLX < 0) \parallel \\ &\quad (xRefLX + (bW \ll \log 2 MVPREC) - 1 > picMaxX) \parallel \\ &\quad (yRefLX + (bH \ll \log 2 MVPREC) - 1 > picMaxY) \end{aligned}$$

[0117] 此外,00B判定部30921也可以不通过以小数为单位的边界坐标而通过以整数为单位的坐标进行判定。

$$xRefLX = xPb + (mvLX[0] \gg \log 2 MVPREC) \text{ (Decide00B3)}$$

```

yRefLX=yPb+(mvLX[1]>>log2MVPREC)
is00B[X]=(xRefLX<0)|| (yRefLX<0)||
(xRefLX+bW-1>=picW-1)||
(yRefLX+bH-1>=picH-1)

```

[0118] 或者, (Decide00B3) 中的is00B[X]的推导式也可以为下式。

```

is00B[X]=(xRefLX<=0)|| (yRefLX<=0)||
(xRefLX+bW-1>=picW-1)||
(yRefLX+bH-1>=picH-1)

```

[0119] 根据上述内容,使用通过基于运动矢量mvLX[]的MV精度的右移位得到的整数坐标,因此,起到能比以小数精度进行判定简化处理量的效果。

[0120] 00B判定部30921可以根据块尺寸来切换是否应用00B处理。例如,可以是,00B判定部30921将以下算式用于is00B[X]的推导,仅对宽度和高度均为BLKTHW1、BLKTHH1像素以上的块设定is00B=true(真),应用00B处理。

```

is00B[X]=(bW>=BLKTHW1)&&(bH>=BLKTHH1)&&
((xRefLX<picMinX)|| (yRefLX<picMinY)||
(xRefLX+bW*MVPREC-1>=picMaxX)||
(yRefLX+bH*MVPREC-1>=picMaxY))

```

[0121] 在此,应用00B处理是指设为is00B[X]=true。是在加权预测部3094中不加权而优先使用任一个参照图像来推导出预测图像Pred[][]的意思。

[0122] 或者,也可以通过下式推导出is00B[X]。

```

is00B[X]=(bW>=BLKTHW1)&&(bH>=BLKTHH1)&&
((xRefLX<=picMinX)|| (yRefLX<=picMinY)||
(xRefLX+(bW<<log2MVPREC)-1>=picMaxX)||
(yRefLX+(bH<<log2MVPREC)-1>=picMaxY))

```

[0123] 或者,也可以通过下式推导出is00B[X]。

```

picMaxX=(picW-1)<<log2MVPREC
picMaxY=(picH-1)<<log2MVPREC
is00B[X]=(bW>=BLKTHW1)&&(bH>=BLKTHH1)&&
((xRefLX<0)|| (yRefLX<0)||
(xRefLX+(bW<<log2MVPREC)-1>picMaxX)||
(yRefLX+(bH<<log2MVPREC)-1>picMaxY))

```

[0124] 或者,也可以通过下式推导出is00B[X]。

```

xRefLX=xPb+(mvLX[0]>>log2MVPREC)
yRefLX=yPb+(mvLX[1]>>log2MVPREC)
is00B[X]=(bW>=BLKTHW1)&&(bH>=BLKTHH1)&&
((xRefLX<0)|| (yRefLX<0)||
(xRefLX+bW-1>=picW-1)||
(yRefLX+bH-1>=picH-1))

```

[0125] 或者,也可以通过下式推导出is00B[X]。

```
is00B[X] = (bW >= BLKTHW1) && (bH >= BLKTHH1) &&
((xRefLX <= 0) || (yRefLX <= 0) ||
(xRefLX + bW - 1 >= picW - 1) ||
(yRefLX + bH - 1 >= picH - 1))
```

[0126] 例如,可以是BLKTHW1=4、8、16、32,BLKTHH1=4、8、16、32等。应用00B的块尺寸的阈值可以是8x8、4x4、4x8、8x4、16x16等。

[0127] 或者,也可以在大的块中禁止00B处理。

```
is00B[X] = (bW <= BLKTHW2) && (bH <= BLKTHW2) &&
((xRefLX < picMinX) || (yRefLX < picMinY) ||
(xRefLX + bW * MVPREC - 1 >= picMaxX) ||
(yRefLX + bH * MVPREC - 1 >= picMaxY))
```

[0128] 或者,也可以通过下式推导出is00B[X]。

```
is00B[X] = (bW <= BLKTHW2) && (bH <= BLKTHW2) &&
((xRefLX <= picMinX) || (yRefLX <= picMinY) ||
(xRefLX + (bW << log2MVPREC) - 1 >= picMaxX) ||
(yRefLX + (bH << log2MVPREC) - 1 >= picMaxY))
```

[0129] 或者,也可以通过下式推导出is00B[X]。

```
picMaxX = (picW - 1) << log2MVPREC
picMaxY = (picH - 1) << log2MVPREC
is00B[X] = (bW <= BLKTHW2) && (bH <= BLKTHW2) &&
((xRefLX < 0) || (yRefLX < 0) ||
(xRefLX + (bW << log2MVPREC) - 1 >= picMaxX) ||
(yRefLX + (bH << log2MVPREC) - 1 >= picMaxY))
```

[0130] 或者,也可以通过下式推导出is00B[X]。

```
xRefLX = xPb + (mvLX[0] >> log2MVPREC)
yRefLX = yPb + (mvLX[1] >> log2MVPREC)
is00B[X] = (bW <= BLKTHW2) && (bH <= BLKTHW2) &&
((xRefLX < 0) || (yRefLX < 0) ||
(xRefLX + bW - 1 >= picW - 1) ||
(yRefLX + bH - 1 >= picH - 1))
```

[0131] 或者,也可以通过下式推导出is00B[X]。

```
xRefLX = xPb + (mvLX[0] >> log2MVPREC)
yRefLX = yPb + (mvLX[1] >> log2MVPREC)
is00B[X] = (bW <= BLKTHW2) && (bH <= BLKTHW2) &&
((xRefLX <= 0) || (yRefLX <= 0) ||
(xRefLX + bW - 1 >= picW - 1) ||
(yRefLX + bH - 1 >= picH - 1))
```

[0132] 例如,可以是BLKTHW2=16、32、64,BLKTHH2=16、32、64等。

[0133] 或者也可以设为仅对特定的范围的大小的块应用00B处理。

$is00B[X] = (bW > BLKTHW3) \&\& (bW \leq BLKTHW4) \&\& (bH > BLKTHH3) \&\& (bH \leq BLKTHH4) \&\&$

$((xRefLX < picMinX) || (yRefLX < picMinY) ||$
 $(xRefLX + bW * MVPREC - 1 > picMaxX) ||$
 $(yRefLX + bH * MVPREC - 1 > picMaxY))$

[0134] 或者,也可以通过下式推导出is00B[X]。

$is00B[X] = (bW > BLKTHW3) \&\& (bW \leq BLKTHW4) \&\& (bH > BLKTHH3) \&\& (bH \leq BLKTHH4) \&\&$

$((xRefLX \leq picMinX) || (yRefLX \leq picMinY) ||$
 $(xRefLX + (bW < \log 2 MVPREC) - 1 > picMaxX) ||$
 $(yRefLX + (bH < \log 2 MVPREC) - 1 > picMaxY))$

[0135] 或者,也可以通过下式推导出is00B[X]。

$picMaxX = (picW - 1) < \log 2 MVPREC$

$picMaxY = (picH - 1) < \log 2 MVPREC$

$is00B[X] = (bW > BLKTHW3) \&\& (bW \leq BLKTHW4) \&\& (bH > BLKTHH3) \&\& (bH \leq BLKTHH4) \&\&$

$((xRefLX < 0) || (yRefLX < 0) ||$
 $(xRefLX + (bW < \log 2 MVPREC) - 1 > picMaxX) ||$
 $(yRefLX + (bH < \log 2 MVPREC) - 1 > picMaxY))$

[0136] 或者,也可以通过下式推导出is00B[X]。

$xRefLX = xPb + (mvLX[0] >> \log 2 MVPREC)$

$yRefLX = yPb + (mvLX[1] >> \log 2 MVPREC)$

$is00B[X] = (bW > BLKTHW3) \&\& (bW \leq BLKTHW4) \&\& (bH > BLKTHH3) \&\& (bH \leq BLKTHH4) \&\&$

$((xRefLX < 0) || (yRefLX < 0) ||$
 $(xRefLX + bW - 1 > picW - 1) ||$
 $(yRefLX + bH - 1 > picH - 1))$

[0137] 或者,也可以通过下式推导出is00B[X]。

$is00B[X] = (bW > BLKTHW3) \&\& (bW \leq BLKTHW4) \&\& (bH > BLKTHH3) \&\& (bH \leq BLKTHH4) \&$

$((xRefLX \leq 0) || (yRefLX \leq 0) ||$
 $(xRefLX + bW - 1 > picW - 1) ||$
 $(yRefLX + bH - 1 > picH - 1))$

[0138] 例如,可以是 $(BLKTHW3, BLKTHW4) = \{(4, 16), (8, 32), (16, 32), (32, 64)\}$,
 $(BLKTHH3, BLKTHH4) = \{(4, 16),$

$(8, 32), (16, 32), (32, 64)\}$ 。

[0139] 或者也可以设为根据面积的大小来判定并应用00B处理。

$is00B[X] = (bW * bH > BLKTHW5 * BLKTHH5) \&\&$

$((xRefLX < picMinX) || (yRefLX < picMinY) ||$
 $(xRefLX + bW * MVPREC - 1 > picMaxX) ||$

(yRefLX+bH*MVPREC-1>=picMaxY))

[0140] 或者,也可以通过下式推导出is00B[X]。

is00B[X] = (bW*bH>=BLKTHW5*BLKTHH5)&&
((xRefLX<=picMinX) || (yRefLX<=picMinY) ||
(xRefLX+(bW<<log2MVPREC)-1>=picMaxX) |
(yRefLX+(bH<<log2MVPREC)-1>=picMaxY))

[0141] 或者,也可以通过下式推导出is00B[X]。

picMaxX = (picW-1) <<log2MVPREC
picMaxY = (picH-1) <<log2MVPREC
is00B[X] = (bW*bH>=BLKTHW5*BLKTHH5)&&
((xRefLX<0) || (yRefLX<0) ||
(xRefLX+(bW<<log2MVPREC)-1>picMaxX) ||
(yRefLX+(bH<<log2MVPREC)-1>picMaxY))

[0142] 或者,也可以通过下式推导出is00B[X]。

xRefLX = xPb + (mvLX[0]>>log2MVPREC)
yRefLX = yPb + (mvLX[1]>>log2MVPREC)
is00B[X] = (bW*bH>=BLKTHW5*BLKTHH5)&&
((xRefLX<0) || (yRefLX<0) ||
(xRefLX+bW-1>=picW-1) ||
(yRefLX+bH-1>=picH-1))

[0143] 或者,也可以通过下式推导出is00B[X]。

is00B[X] = (bW*bH>=BLKTHW5*BLKTHH5)&&
((xRefLX<=0) || (yRefLX<=0) ||
(xRefLX+bW-1>=picW-1) ||
(yRefLX+bH-1>=picH-1))

[0144] 例如,可以是BLKTHW5=4、8、16、32, BLKTHH5=4、8、16、32等。或者也可以使用BLKTHS5来代替BLKTHW5*BLKTHH5。例如,可以是BLKTHS5=32、64、128等。

[0145] 在上述的例子中,基于(Decide00B1)的00B判定式追加了限制块尺寸的条件,但也可以基于(Decide00B2)或者(Decide00B3)的00B判定式追加限制块尺寸的条件。

[0146] 在上述的示例中,使用了(xRefLX<picMinX) || (yRefLX<picMinY)作为is00B[X]的推导式的条件的一部分,但也可以使用(xRefLX<=picMinX) || (yRefLX<=picMinY)。

[0147] 如上所述,能限制应用00B处理的块尺寸来降低计算量。

[0148] 00B判定部30921也可以是避免is00B[X]和is00B[1-X] (X=0,1) 两方均为1的构成。就是说,在参照块双方中禁止帧间预测图像生成部309进行使用了00B掩码的预测图像生成。由此,参照块的一方一定是应用is00B[X]=false (假) 即不使用00B处理,能在后段省略00B掩码的生成和掩码的应用。因此,能削减计算量。

<00B判定省略构成的示例1>

[0149] 00B判定部30921可以通过与L0参照区域和L1参照区域的有效范围的大小相应的判定来推导出is00B[X] (X=0,1)。将LX参照块中的图片中所包括的有效面积InsideAreaX

较大的一方设为 $X=K$ ($K=0,1$)。将有效面积较大的参照图片的OOB判定设为无效 (OOB判定有无标志 $isOOB[K]=false$)，将有效面积较小的另一方 ($isOOB[!K]$ 或 $isOOB[1-K]$) 的OOB判定设为有效 ($isOOB[1-K]=true$)。OOB判定部30921例如通过以下算式求出InsideAreaX。需要说明的是，可以使用(0,1)作为(false,true)的值，也可以使用其他的值(1,0)作为(false,true)的值。LX参照区域的有效范围可以通过LX参照块和图片的共同部分来推导。

```

xRefLX=xPb*MVPREC+mvLX[0]
yRefLX=yPb*MVPREC+mvLX[1]
picMinX=0-MVPREC/2
picMinY=0-MVPREC/2
picMaxX=(picW-1)*MVPREC+MVPREC/2
picMaxY=(picH-1)*MVPREC+MVPREC/2
InsideWidthX=(min(picMaxX,xRefLX+bW*MVPREC-1)-max(picMinX,xRefLX))
InsideHeightX=(min(picMaxY,yRefLX+bH*MVPREC-1)-max(picMinY,yRefLX))
InsideAreaX=(InsideWidthX>0&&InsideHeightX>0)?InsideWidthX*

```

InsideHeightX:0

[0150] 在InsideWidthX或InsideHeightX的值为负的情况下，参照块在图片中没有共同部分(参照块不包括于图片)，因此可以设为InsideAreaX=0。

[0151] 尺寸(picW,picH,bW,bH)与精度MVPREC的相乘可以如下所示地以MVPREC的对数对尺寸进行左比特移位“<<”处理。

```

xRefLX=(xPb<<log2MVPREC)+mvLX[0]
yRefLX=(yPb<<log2MVPREC)+mvLX[1]

```

[0152] 或者,picMaxX,picMaxY也可以通过以下算式来推导。

```

picMaxX=(picW-1)*MVPREC
picMaxY=(picH-1)*MVPREC

```

[0153] 或者,xRefLX,yRefLX也可以不通过以小数为单位的边界坐标而通过以整数为单位的坐标来推导。

```

xRefLX=xPb+mvLX[0]*MVPREC
yRefLX=yPb+mvLX[1]*MVPREC

```

[0154] OOB判定部30921针对L0参照块和L1参照块分别推导出InsideArea0和InsideArea1,如下所述地推导出isOOB[X]。

```

if (InsideArea0>=InsideArea1) {
    isOOB[0]=false, isOOB[1]=true
} else {
    isOOB[0]=true, isOOB[1]=false
}

```

[0155] 此外,如下所述地先推导出一方的isOOB[X]的判定值,再从isOOB[X]的判定值推导出另一方isOOB[Y] ($Y=1-X$) 的判定值。

```

isOOB[0]=(InsideArea0>=InsideArea1)?0:1

```

isOOB[1]=1-isOOB[0]

[0156] 上述条件式也可以是“InsideArea0>InsideArea1”。

[0157] 根据上述构成,能将有效区域较大的一方的参照图片中的OOB判定设为无效(标志isOOB[K]=false),省略(不应用OOB处理)OOB判定,因此能削减预测图像生成中的计算量。特别是,在有效区域较大的情况下,即使将OOB判定设为无效画质也几乎不会降低。

<OOB判定省略构成的示例2>

[0158] 对于OOB判定部30921,可以在与两个参照图像有关的有效区域InsideArea的尺寸相等的情况下,对两方均将OOB判定设为无效。具体而言,可以推导出isOOB[X],使其为表示省略OOB判定的值(false)。

```
if (InsideArea0>InsideArea1) {  
    isOOB[0]=false, isOOB[1]=true  
} else if (InsideArea0<InsideArea1) {  
    isOOB[0]=true, isOOB[1]=false  
} else {  
    isOOB[0]=false, isOOB[1]=false  
}
```

[0159] 根据上述构成,能削减预测图像生成中的计算量。

<OOB判定省略构成的示例3>

[0160] OOB判定部30921可以使用表示LX参照区域的有效范围(例如,LX参照块与图片的共同部分)的纵横的大小的InsideWidthX、InsideHeightX的大小关系来推导出OOB判定有无标志isOOB[X]。例如,在某个LX参照块的有效区域的宽度和高度两方比另一方的LY参照块(Y=1-X)的宽度和高度更大的情况下,也可以将LX参照块的OOB判定设为无效(isOOB=false)。

```
if (InsideWidth0>InsideWidth1 && InsideHeight0>InsideHeight1) {  
    isOOB[0]=false, isOOB[1]=true  
} else {  
    isOOB[0]=true, isOOB[1]=false  
}
```

[0161] 或者也可以如下。

```
if (InsideWidth0>=InsideWidth1 && InsideHeight0>InsideHeight1) {  
    isOOB[0]=false, isOOB[1]=true  
} else {  
    isOOB[0]=true, isOOB[1]=false  
}
```

[0162] 或者也可以如下。

```
if (InsideWidth0>=InsideWidth1 && InsideHeight0>=InsideHeight1) {  
    isOOB[0]=false, isOOB[1]=true  
} else {  
    isOOB[0]=true, isOOB[1]=false  
}
```

[0163] 或者,也可以是,在某个LX参照块的有效区域的宽度和高度两方比另一方的LY参照块($Y=1-X$)的宽度和高度更大的情况下,将LX参照块的OOB判定设为无效(isOOB=false),在两个有效区域的宽度或高度相等的情况下,对双方将OOB判定均设为无效。

```
if (InsideWidth0>InsideWidth1 && InsideHeight0>InsideHeight1) {  
    isOOB[0]=false, isOOB[1]=true  
} else if (InsideWidth0<InsideWidth1 &&  
InsideHeight0<InsideHeight1) {  
    isOOB[0]=true, isOOB[1]=false  
} else {  
    isOOB[0]=false, isOOB[1]=false  
}
```

[0164] 或者,也可以对上述判定仅使用宽度或仅使用高度来进行OOB判定。

```
if (InsideWidth0>InsideWidth1) {  
    isOOB[0]=false, isOOB[1]=true  
} else {  
    isOOB[0]=true, isOOB[1]=false  
}
```

[0165] 或者也可以如下。

```
if (InsideHeight0>InsideHeight1) {  
    isOOB[0]=false, isOOB[1]=true  
} else {  
    isOOB[0]=true, isOOB[1]=false  
}
```

[0166] 根据上述构成,能省略有效区域较大的参照块(或两方的参照块)的OOB判定,因此能削减预测图像生成中的计算量的削减。而且,无需推导面积就能削减乘法的计算量。

<总结>

[0167] 综上所述,可以将L0参照区域的有效区域与L1参照区域的有效区域进行比较,将

参照区域较大的一方的参照图片的00B判定设为无效。此外,也可以通过L0与L1的有效区域的面积的比较来进行参照区域是否较大的判定。此外,也可以通过L0与L1的有效区域的宽度或高度的比较来进行参照区域是否较大的判定。

(00B掩码推导处理)

[0168] 00B掩码推导部30922针对参照块推导出00B掩码(00BMask[X])。00BMask[X]是bWxbH像素的二值的图像(掩码),如果00BMask[X][px][py](px=0..bW-1,py=0..bH-1)的值为0(false),则表示像素(px,py)在图片内,如果该值为1(true)则表示像素(px,py)在图片外。需要说明的是,false和true不限于0、1,也可以是其他的值,例如是2进制数且由BitDepth以上的长度的0或1的串构成的比特掩码值“0000000000b”、“1111111111b”等。此外,也可以使用16进制数的“000”、“3FF”、“000”、“FFF”、“0000”、“FFFF”。00B掩码推导部30922针对00B判定部30921判定为不是00B处理对象(is00B[X]==false)的块,将00BMask[X]的所有的值设为false。此外,针对00B判定部30921判定为是00B处理对象(is00B[X]==true)的块,使用如下算式推导出掩码。需要说明的是,不等号(“>”或“<”)也可以不包括等号(“>”或“<”)。

$$\begin{aligned} xRefLX &= (xPb \ll \log 2 MVPREC) + mvLX[0] \\ yRefLX &= (yPb \ll \log 2 MVPREC) + mvLX[1] \\ 00BMask[X][px][py] &= ((xRefLX + px < picMinX) \parallel (yRefLX + py < picMinY) \parallel \\ &\quad (xRefLX + px \geq picMaxX) \parallel (yRefLX + py \geq picMaxY)) ? 1 : 0 \end{aligned}$$

[0169] 其中,px=0..bW-1,py=0..bH-1。如上所述基于(Decide00B1)中的is00B[X]的判定式推导出00BMask。除此以外,也可以基于(Decide00B2)、(Decide00B3)中的is00B[X]的判定式来推导出00BMask。而且,也可以基于对(Decide00B1)、(Decide00B2)、(Decide00B3)附加有效范围的限制的is00B[X]的判定式来推导出00BMask。

[0170] 或者,00BMask的推导式也可以为下式。

$$\begin{aligned} 00BMask[X][px][py] &= ((xRefLX + px < picMinX) \parallel (yRefLX + py < picMinY) \parallel \\ &\quad (xRefLX + px + (bW \ll \log 2 MVPREC) - 1 \geq picMaxX) \parallel \\ &\quad (yRefLX + py + (bH \ll \log 2 MVPREC) - 1 \geq picMaxY)) ? 1 : 0 \end{aligned}$$

[0171] 或者,00BMask的推导式也可以为下式。

$$\begin{aligned} 00BMask[X][px][py] &= ((xRefLX + px \leq picMinX) \parallel (yRefLX + py \leq picMinY) \parallel \\ &\quad (xRefLX + px + (bW \ll \log 2 MVPREC) - 1 \geq picMaxX) \parallel \\ &\quad (yRefLX + py + (bH \ll \log 2 MVPREC) - 1 \geq picMaxY)) ? 1 : 0 \end{aligned}$$

[0172] 或者,也可以省略用于舍入的项,如下所示地推导出00BMask。

$$\begin{aligned} picMaxX &= (picW - 1) \ll \log 2 MVPREC \\ picMaxY &= (picH - 1) \ll \log 2 MVPREC \\ 00BMask[X][px][py] &= ((xRefLX + px < 0) \parallel (yRefLX + py < 0) \parallel \\ &\quad (xRefLX + (bW \ll \log 2 MVPREC) - 1 \geq picMaxX) \parallel \\ &\quad (yRefLX + (bH \ll \log 2 MVPREC) - 1 \geq picMaxY)) ? 1 : 0 \end{aligned}$$

[0173] 或者,也可以使用以整数为单位的运动矢量来推导出00BMask。

$$\begin{aligned} xRefLX &= xPb + (mvLX[0] \gg \log 2 MVPREC) \\ yRefLX &= yPb + (mvLX[1] \gg \log 2 MVPREC) \end{aligned}$$

$$\begin{aligned} \text{OOBMask}[X][px][py] = & ((x\text{RefLX}+px < 0) \parallel (y\text{RefLX}+py < 0) \parallel \\ & (x\text{RefLX}+px+bW-1 \geq \text{picW}-1) \parallel \\ & (y\text{RefLX}+py+bH-1 \geq \text{picH}-1)) ? 1:0 \end{aligned}$$

[0174] 或者,OOBMask的推导式也可以为下式。

$$\begin{aligned} \text{OOBMask}[X] = & ((x\text{RefLX}+px < 0) \parallel (y\text{RefLX}+py < 0) \parallel \\ & (x\text{RefLX}+px+bW-1 \geq \text{picW}-1) \parallel \\ & (y\text{RefLX}+py+bH-1 \geq \text{picH}-1)) ? 1:0 \end{aligned}$$

[0175] OOB掩码推导部30922可以将OOBMask[X]设为以NxM像素为单位的掩码来进行推导。即,使一个掩码值与NxM像素的OOB判定对应。此时,OOBMask[X]的大小为(bW/N)*(bH/M)。各掩码值OOBMask[X][x][y] (x=0..bW/N-1,y=0..bH/M-1)例如如下所示地求出。

$$\begin{aligned} x\text{RefLX} = & (xPb < \log 2MVPREC) + mvLX[0] \\ y\text{RefLX} = & (yPb < \log 2MVPREC) + mvLX[1] \\ \text{OOBMask}[X][px][py] = & ((x\text{RefLX}+px*N < \text{picMinX}) \parallel \\ & (y\text{RefLX}+py*M < \text{picMinY}) \parallel \\ & (x\text{RefLX}+px*N+N \geq \text{picMaxX}) \parallel \\ & (y\text{RefLX}+py*M+M \geq \text{picMaxY})) ? 1:0 \end{aligned}$$

[0176] 在此,px=0..bW/N-1,py=0..bH/M-1。N和M的项可以省略。如上所述基于(Decide00B1)中的is00B[X]的判定式推导出OOBMask。除此以外,也可以基于(Decide00B2)、(Decide00B3)中的is00B[X]的判定式来推导出OOBMask。而且,也可以基于对(Decide00B1)、(Decide00B2)、(Decide00B3)附加有效范围的限制的is00B[X]的判定式来推导出OOBMask。

[0177] 或者,OOBMask的推导式也可以为下式。

$$\begin{aligned} \text{OOBMask}[X][px][py] = & ((x\text{RefLX}+px*N < \text{picMinX}) \parallel (y\text{RefLX}+py*N < \text{picMinY}) \parallel \\ & (x\text{RefLX}+px*N+N+bW*MVPREC-1 \geq \text{picMaxX}) \parallel \\ & (y\text{RefLX}+py*N+M+bH*MVPREC-1 \geq \text{picMaxY})) ? 1:0 \end{aligned}$$

[0178] 或者,也可以省略用于舍入的项,如下所示地推导出OOBMask。

$$\begin{aligned} \text{picMaxX} = & (\text{picW}-1) < \log 2MVPREC \\ \text{picMaxY} = & (\text{picH}-1) < \log 2MVPREC \\ \text{OOBMask}[X][px][py] = & ((x\text{RefLX}+px*N < 0) \parallel (y\text{RefLX}+py*N < 0) \parallel \\ & (x\text{RefLX}+px*N+N+bW*MVPREC-1 \geq \text{picMaxX}) \parallel \\ & (y\text{RefLX}+py*N+M+bH*MVPREC-1 \geq \text{picMaxY})) ? 1:0 \end{aligned}$$

[0179] 或者,也可以使用以整数为单位的运动矢量来推导出OOBMask。

$$\begin{aligned} x\text{RefLX} = & xPb + (mvLX[0] >> \log 2MVPREC) \\ y\text{RefLX} = & yPb + (mvLX[1] >> \log 2MVPREC) \\ \text{OOBMask}[X][px][py] = & ((x\text{RefLX}+px*N < 0) \parallel (y\text{RefLX}+py*N < 0) \parallel \\ & (x\text{RefLX}+px*N+N+bW-1 \geq \text{picW}-1) \parallel \\ & (y\text{RefLX}+py*N+M+bH-1 \geq \text{picH}-1)) ? 1:0 \end{aligned}$$

[0180] 由此,以NxM像素为单位,掩码为相同值,因此,能削减掩码所需的存储器,并且削减计算量。

(OOB预测图像生成)

[0181] 运动补偿部3091(内插图像生成部3091)在双向预测的情况下将通过上述的(算式MC-1)推导出的Pred[] []按每个L0列表、L1列表设定为内插图像PredL0[] []和PredL1[] []。然后,加权预测部3094从PredL0[] []和PredL1[] []生成内插图像Pred[] []。在双向预测的参照块为OOB处理的对象的情况下,在预测图像的生成中使用与该参照块对应的OOBMask[X]。此时,Pred[] []如下所示地进行推导。

```

if (OOBMask[0][px][py]&&! OOBMask[1][px][py]) { (算式 WP-1)
    Pred[px][py]= (PredL1[px][py]+ offset1) >>shift1
} else if (! OOBMask[0][px][py]&& OOBMask[1][px][py]) {
    Pred[px][py]= (PredL0[px][py]+offset1) >>shift1
} else {
    Pred[px][py]= (PredL0[px][py]+PredL1[px][py]+offset2) >>shift2
}

```

[0182] 在此,px=0..bW-1,py=0..bH-1。此外,shift1=Max(2,14-bitDepth),shift2=Max(3,15-bitDepth),offset1=1<<(shift1-1),offset2=1<<(shift2-1)。需要说明的是,shift1是用于将内插图像比特深度(例如14bit)返回原来的比特深度的移位(例如14-bitDepth)。shift2是返回原来的比特的同时兼作平均的右移位置量,offset1、offset2是右移位shift1、shift2时的舍入值。

[0183] 需要说明的是,可以根据是否使用以CU为单位的明确的加权双向预测(BCW)或帧内帧间结合预测CIIP(combined inter-picture merge and intra-picture prediction)来切换上述处理。bcwIdx是表示BCW的权重的索引。ciip_flag是表示CIIP的打开关闭的标志。在bcwIdx==0或CIIP标志(ciip_flag)为1的情况下进行上述(算式WP-1)的处理。在除此以外的情况下进行以下的处理。

```

w1=bcwWLut[bcwIdx]
w0=(8-w1)

```

[0184] 在此,bcwWLut[k]={4,5,3,10,-2}

```

if (OOBMask[0][px][py]&&! OOBMask[1][px][py]) {
    Pred[px][py]= (PredL1[px][py]+ offset1) >>shift1
} else if (! OOBMask[0][px][py]&& OOBMask[1][px][py]) {
    Pred[px][py]= (PredL0[px][py]+offset1) >>shift1
} else {
    Pred[px][py]= ( w0*PredL0[px][py]+w1*PredL1[px][py]+offset3 ) >>
    (shift1+3) )
}

```

[0185] 在此,offset3=1<<(shift1+2)。

[0186] 在将表示是否以块为单位使用LX参照图片的标志predFlagLX与基于OOBMask的处理集中于一个分支并如下所示地进行处理时,起到能减轻处理量的效果。

```

if ( (predFlagL0==1 && predFlagL1==0) || (! OOBMask[0][px][py]&&
OOBMask[1][px][py]) ) { (算式 WP-2)
    Pred[px][py]= (PredL0[px][py]+offset1) >>shift1
} else if ( ( predFlagL0==0 && predFlagL1==1 ) ||
(OOBMask[0][px][py]&&! OOBMask[1][px][py]) ) {
    Pred[px][py]= (PredL1[px][py]+ offset1) >>shift1
} else if (bcwIdx==0||ciip_flag==1) {
    Pred[px][py]= (PredL0[px][py]+PredL1[px][py]+offset2) >>shift2
} else { //bcwIdx!=0 and ciip_flag==0)
    Pred[px][py]= ( w0*PredL0[px][py]+w1*PredL1[px][py]+offset3 ) >>
(shift1+3) )
}

```

<省略构成>

[0187] OOB判定部30921在根据两张参照图片上的参照块的加权和来生成预测图像的加权预测中,也可以不对块应用OOB处理。即,应用加权预测的块将OOB判定有无标志is00B[X]设为false。由此,能防止由未使用应用双向的加权预测的参照块的一方引起的失真的产生或编码效率的降低。在这样的加权预测中有BCW (Bi-prediction with CU-level Weight: 具有CU级权重的双向预测) 预测和WP (Weighted prediction: 加权预测) 预测。

[0188] BCW预测以块为单位选择加权系数(w0,w1),将双向预测中的L0内插图像和L1内插图像乘以权重来生成预测图像。参数解码部302从编码数据解码在编码单元CU中表示加权系数的参数bcw_idx。参数编码部111对bcw_idx进行编码。在合并预测的情况下,将参照块(邻接块)的bcwIdx设定为对象块的bcwIdx。在除此以外的情况下,对对象块的bcwIdx设定bcw_idx。如果bcwIdx!=0,则对该CU应用BCW预测。OOB判定部30921在应用BCW预测的情况下(bcwIdx!=0),将is00B[X]设为false。

[0189] WP预测以切片为单位切换每个参照图片的加权系数(we0,we1)和偏移(offset0,offset1),以CU为单位基于参照图片索引来选择加权系数和偏移,将L0内插图像和L1内插图像乘以权重来生成预测图像。参数解码部302从PPS的编码数据解码表示是否对B图片应用WP预测的pps_weighted_bipred_flag。在pps_weighted_bipred_flag==1的情况下,对该PPS的B图片应用WP预测。

[0190] (we0,we1)、(offset0,offset1)是以切片为单位从编码数据解码的加权系数、偏移,按照参照图片索引来进行推导。

[0191] OOB判定部30921在应用明确的WP预测的情况下(pps_weighted_bipred_flag==1),将is00B[X]设为false。需要说明的是,也可以在不使用两张参照图片的情况下(predFlagLX(X=0,1)中的任一个为0),应用OOB处理。

[0192] 在利用上述的省略构成的情况下,OOBMask[0]、OOBMask[1]不会都是1,因此,加权预测部3094能通过以下的分支处理来推导出内插图像。

```

if (OOBMask[0][px][py]) {
    Pred[px][py]= (we1*PredL1[px][py]+offset1) >>shift1
} else if (OOBMask[1][px][py]) {
    Pred[px][py]= (we0*PredL0[px][py]+offset0) >>shift1
} else {
    if (bcwIdx==0 && pps_weighted_bipred_flag==1) {
        Pred[px][py]=
        (we0*PredL0[px][py]+we1*PredL1[px][py]+offset2) >>shift2
    } else { //bcwIdx !=0 and pps_weighted_bipred_flag==0)
        Pred[px][py]= (w0*PredL0[px][py]+w1*PredL1[px][py]+offset3)
        >> (shift1+3) )
    }
}

```

[0193] 此外,预先计算出PredL0、PredL1、PredBI,通过以下的掩码处理推导出最终的Pred是适当的。

[0194] 首先,计算出PredL0、PredL1、PredBI。

PredL0[px][py] = (we0*PredL0[px][py]+offset0) >>shift1

PredL1[px][py] = (we1*PredL1[px][py]+offset1) >>shift1

PredBI[px][py] = (we0*PredL0[px][py]+we1*PredL1[px][py]+offset2) >>shift2 (WP预测的情况)

PredBI[px][py] = (w0*PredL0[px][py]+w1*PredL1[px][py]+offset3) >>(shift1+3) (BCW预测的情况)

[0195] 接着,在使用2进制数且BitDepth长度为“111111111b”、“000000000b”的比特掩码作为OOBMask[0][px][py]、OOBMask[1][px][py]的值的的情况下,通过以下的处理推导出最终的Pred。Pred[px][py] =

(PredBI[px][py] & (!OOBMask[0][px][py] & !OOBMask[1][px][py])) +
 (PredL0[px][py] & OOBMask[0][px][py]) +
 (PredL1[px][py] & OOBMask[1][px][py])

[0196] 需要说明的是,也可以是,在为true=1、false=0的值的的情况下,从0、1生成比特掩码值“000000000b”、“111111111b”并进行以下的处理。在此,“+”也可以是“|”(在以下的掩码运算处理中也是同样的)。Pred[px][py] = (算式exc1-1)

(PredBI[px][py] & (0 - ((!OOBMask[0][px][py] & !OOBMask[1][px][py]) ? 1:0))) +
 (PredL0[px][py] & (0 - (OOBMask[0][px][py] ? 1:0))) +

$(\text{PredL1}[\text{px}][\text{py}] \& (0 - (\text{OoBMask}[1][\text{px}][\text{py}] ? 1 : 0)))$

[0197] 上述的算式包括为了生成具有像素值的比特深度以上的比特宽度的掩码而将二值(0或1)的掩码转换成0(所有的比特为0,像素值为无效)或-1(所有的比特为1,像素值为有效)的计算。预测图像的推导不限于此,只要使用与PredBI、PredL0、PredL1对应的3项中的通过OoB掩码而无效的项为0的计算即可。例如,在将像素值的比特深度设为最大16比特时,可以使用如下算式。

```
Pred[px][py] =
(PredBI[px][py] & ((!OoBMask[0][px][py] & !OoBMask[1][px][py]) ? 0xFFFF : 0)) +
(PredL0[px][py] & (OoBMask[0][px][py] ? 0xFFFF : 0)) +
(PredL1[px][py] & (OoBMask[1][px][py] ? 0xFFFF : 0))
```

[0198] 也可以不对3项的结合使用加法,而如下所述地使用比特OR。

```
Pred[px][py] =
(PredBI[px][py] & ((!OoBMask[0][px][py] & !OoBMask[1][px][py]) ? 0xFFFF : 0)) |
(PredL0[px][py] & (OoBMask[0][px][py] ? 0xFFFF : 0)) |
(PredL1[px][py] & (OoBMask[1][px][py] ? 0xFFFF : 0))
```

[0199] 或者也可以使用乘法。

```
Pred[px][py] =
(PredBI[px][py] * ((!OoBMask[0][px][py] & !OoBMask[1][px][py]) ? 1 : 0)) +
(PredL0[px][py] * (OoBMask[0][px][py] ? 1 : 0)) +
(PredL1[px][py] * (OoBMask[1][px][py] ? 1 : 0))
```

[0200] 根据上述内容,能简化排他处理。通过排他处理,能比(算式excl-1)的运算简化处理。

[0201] 加权预测部3094可以参照isOoB的值预先分配处理。例如,可以设为如下。若如此,则在isOoB[X]为false的情况下起到无需按每个像素参照OoBMask[X][px][py]就能削减计算量的效果。例如在不使用权重的情况下,可以如下所示地生成预测图像。

```
if (isOoB[0] && ! isOoB[1]) {
    if (OoBMask[0][px][py]) {
        Pred[px][py] = (PredL1[px][py] + offset1) >> shift1
    } else {
        Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2) >> shift2
    }
}
```

```

    }
} else if ( ! isOOB[0]&& isOOB[1] ) {
    if ( OOBMask[1][px][py] ) {
        Pred[px][py]= ( PredL0[px][py]+offset1 ) >>shift1
    } else {
        Pred[px][py]=( PredL0[px][py]+PredL1[px][py]+offset2 )>>shift2
    }
} else if ( isOOB[0]&& isOOB[1] ) {
    if ( OOBMask[0][px][py]&& ! OOBMask[1][px][py] ) {
        Pred[px][py]= ( PredL1[px][py]+ offset1 ) >>shift1
    } else if ( ! OOBMask[0][px][py]&& OOBMask[1][px][py] ) {
        Pred[px][py]= ( PredL0[px][py]+offset1 ) >>shift1 } else {
        Pred[px][py]=( PredL0[px][py]+PredL1[px][py]+offset2 )>>shift2
    }
} else {
    Pred[px][py]= ( PredL0[px][py]+PredL1[px][py]+offset2 ) >>shift2
}

```

[0202] 例如,在以CU为单位的明确的加权双向预测(BCW)的情况下($bcwIdx \neq 0$ 且 $pps_weighted_bipred_flag = 0$),可以如下所示地生成预测图像。

```

    if ( isOOB[0]&& ! isOOB[1] ) {
        if ( OOBMask[0][px][py] ) {
            Pred[px][py]= ( PredL1[px][py]+ offset1 ) >>shift1
        } else {
            Pred[px][py]= ( w0*PredL0[px][py]+w1*PredL1[px][py]+offset3 )
>> (shift1+3)
        }
    } else if ( ! isOOB[0]&& isOOB[1] ) {
        if ( OOBMask[1][px][py] ) {
            Pred[px][py]= ( PredL0[px][py]+offset1 ) >>shift1

```

```

        } else {
            Pred[px][py]=(w0*PredL0[px][py]+w1*PredL1[px][py]+offset3)
>> (shift1+3)
        }
    } else if (isOOB[0]&& isOOB[1]) {
        if (OOBMask[0][px][py]&&! OOBMask[1][px][py]) {
            Pred[px][py]= (PredL1[px][py]+ offset1) >>shift1
        } else if (! OOBMask[0][px][py]&& OOBMask[1][px][py]) {
            Pred[px][py]= (PredL0[px][py]+offset1) >>shift1
        } else {
            Pred[px][py]= (w0*PredL0[px][py]+w1*PredL1[px][py]+offset3)
>> (shift1+3)
        }
    } else {
        Pred[px][py]= (w0*PredL0[px][py]+w1*PredL1[px][py]+offset3) >>
(shift1+3)
    }
}

```

[0203] 例如,在WP预测的情况下($bcwIdx==0$ 且 $pps_weighted_bipred_flag==1$),可以如下所示地生成预测图像。

```

    if (isOOB[0]&&! isOOB[1]) {
        if (OOBMask[0][px][py]) {
            Pred[px][py]= (we1*PredL1[px][py]+offset1) >>shift1
        } else {
            Pred[px][py]= (we0*PredL0[px][py]+we1*PredL1[px][py]+offset2)
>>shift2
        }
    } else if (! isOOB[0]&& isOOB[1]) {
        if (OOBMask[1][px][py]) {
            Pred[px][py]= (we0*PredL0[px][py]+offset1) >>shift1
        } else {

```

```

        Pred[px][py]=
        (we0*PredL0[px][py]+we1*PredL1[px][py]+offset2) >>shift2
    }
    } else if (isOOB[0]&& isOOB[1]) {
        if (OOBMask[0][px][py]&& ! OOBMask[1][px][py]) {
            Pred[px][py]= (we1*PredL1[px][py]+offset1) >>shift1
        } else if (! OOBMask[0][px][py]&& OOBMask[1][px][py]) {
            Pred[px][py]= (we0*PredL0[px][py]+offset1) >>shift1
        } else {
            Pred[px][py]= ( we0*PredL0[px][py]+we1*PredL1[px][py]+offset2 )
            >>shift2
        }
    } else {
        Pred[px][py]= ( we0*PredL0[px][py]+we1*PredL1[px][py]+offset2 )
        >>shift2
    }
}

```

[0204] 帧间预测图像生成部309将生成的块的预测图像输出至加法部312。

(运动图像编码装置的构成)

[0205] 接着,对本实施方式的运动图像编码装置11的构成进行说明。图7是表示本实施方式的运动图像编码装置11的构成的框图。运动图像编码装置11构成为包括:预测图像生成部101、减法部102、变换/量化部103、逆量化/逆变换部105、加法部106、环路滤波器107、预测参数存储器(预测参数存储部、帧存储器)108、参照图片存储器(参照图像存储部、帧存储器)109、编码参数确定部110、参数编码部111、预测参数推导部120以及熵编码部104。

[0206] 预测图像生成部101按每个CU生成预测图像。预测图像生成部101包括已经说明的帧间预测图像生成部309和帧内预测图像生成部,省略其说明。

[0207] 减法部102从图像T的像素值中减去从预测图像生成部101输入的块的预测图像的像素值,生成预测误差。减法部102将预测误差输出至变换/量化部103。

[0208] 变换/量化部103对从减法部102输入的预测误差,通过频率变换计算出变换系数,并通过量化推导出量化变换系数。变换/量化部103将量化变换系数输出至参数编码部111和逆量化/逆变换部105。

[0209] 变换/量化部103具备:分离变换部(第一变换部)、非分离变换部(第二变换部)以及缩放部。

[0210] 分离变换部对预测误差应用分离变换。缩放部对变换系数进行基于量化矩阵的缩放。

[0211] 逆量化/逆变换部105与运动图像解码装置31中的逆量化/逆变换部311相同,省略

其说明。计算出的预测误差输出至加法部106。

[0212] 参数编码部111具备报头编码部1110、CT信息编码部1111、CU编码部1112(预测模式编码部)。CU编码部1112还具备TU编码部1114。以下,对各模块的概略动作进行说明。

[0213] 报头编码部1110进行报头信息、分割信息、预测信息、量化变换系数等参数的编码处理。

[0214] CT信息编码部1111对QT、MT(BT、TT)分割信息等进行编码。

[0215] CU编码部1112对CU信息、预测信息、分割信息等进行编码。

[0216] TU编码部1114在TU中包括预测误差的情况下,对QP更新信息和量化预测误差进行编码。

[0217] CT信息编码部1111、CU编码部1112将帧间预测参数(predMode、general_merge_flag、merge_idx、inter_pred_idc、refIdxLX、mvp_LX_idx、mvdLX)、帧内预测参数、量化变换系数等语法要素供给至参数编码部111。

[0218] 熵编码部104被从参数编码部111输入量化变换系数和编码参数(分割信息、预测参数)。熵编码部104对它们进行熵编码,生成并输出编码流Te。

[0219] 预测参数推导部120是包括帧间预测参数编码部112、帧内预测参数编码部的单元,根据从编码参数确定部110输入的参数来推导出帧内预测参数和帧间预测参数。推导出的帧内预测参数和帧间预测参数输出至参数编码部111。

(帧间预测参数编码部的构成)

[0220] 帧间预测参数编码部112构成为包括参数编码控制部1121、帧间预测参数推导部303。帧间预测参数推导部303是与运动图像解码装置共同的构成。参数编码控制部1121包括合并索引推导部11211和矢量候选索引推导部11212。

[0221] 合并索引推导部11211推导出合并候选等,输出至帧间预测参数推导部303。矢量候选索引推导部11212推导出预测矢量候选等,输出至帧间预测参数推导部303和参数编码部111。

(帧内预测参数编码部的构成)

[0222] 帧内预测参数编码部具备参数编码控制部和帧内预测参数推导部。帧内预测参数推导部是与运动图像解码装置共同的构成。

[0223] 不过,与运动图像解码装置不同,对帧间预测参数推导部303、帧内预测参数推导部进行输入的是编码参数确定部110、预测参数存储器108,输出至参数编码部111。

[0224] 加法部106将从预测图像生成部101输入的预测块的像素值和从逆量化/逆变换部105输入的预测误差按每个像素相加来生成解码图像。加法部106将生成的解码图像存储于参照图片存储器109。

[0225] 环路滤波器107对加法部106所生成的解码图像,实施去块滤波器、SAO、ALF。需要说明的是,环路滤波器107不一定包括上述三种滤波器,例如也可以是仅包括去块滤波器的构成。

[0226] 预测参数存储器108将编码参数确定部110所生成的预测参数按每个对象图片和CU存储于预定的位置。

[0227] 参照图片存储器109将环路滤波器107所生成的解码图像按每个对象图片和CU存储于预定的位置。

[0228] 编码参数确定部110选择编码参数的多个集合中的一个集合。编码参数是指上述的QT、BT或TT分割信息、预测参数或与它们关联生成的作为编码对象的参数。预测图像生成部101使用这些编码参数来生成预测图像。

[0229] 编码参数确定部110对多个集合的每一个集合计算出表示信息量的尺寸和编码误差的RD成本值。RD成本值例如是代码量与平方误差乘以系数 λ 而得到的值之和。代码量是对量化误差和编码参数进行熵编码而得到的编码流Te的信息量。平方误差是在减法部102中计算出的预测误差的平方和。系数 λ 是大于预先设定的零的实数。编码参数确定部110选择计算出的成本值为最小的编码参数的集合。编码参数确定部110将所确定的编码参数输出至预测参数推导部120。

[0230] 需要说明的是,可以通过计算机实现上述的实施方式中的运动图像编码装置11、运动图像解码装置31的一部分、例如,熵解码部301、参数解码部302、环路滤波器305、预测图像生成部308、逆量化/逆变换部311、加法部312、预测参数推导部320、预测图像生成部101、减法部102、变换/量化部103、熵编码部104、逆量化/逆变换部105、环路滤波器107、编码参数确定部110、参数编码部111以及预测参数推导部120。在该情况下,可以通过将用于实现该控制功能的程序记录于计算机可读记录介质,使计算机系统读入记录于该记录介质的程序并执行来实现。需要说明的是,在此提到的“计算机系统”是指内置于运动图像编码装置11、运动图像解码装置31中的任一者的计算机系统,采用包括OS、外围设备等硬件的计算机系统。此外,“计算机可读记录介质”是指软盘、磁光盘、ROM、CD-ROM等可移动介质、内置于计算机系统的硬盘等存储装置。而且,“计算机可读记录介质”也可以包括:像经由互联网等网络或电话线路等通信线路来发送程序的情况下的通信线那样短时间内、动态地保存程序的记录介质;以及像作为该情况下的服务器、客户端的计算机系统内部的易失性存储器那样保存程序固定时间的记录介质。此外,上述程序可以是用于实现上述功能的一部分的程序,也可以是能通过与已记录在计算机系统程序中的程序进行组合来实现上述功能的程序。

[0231] 此外,也可以将上述的实施方式中的运动图像编码装置11、运动图像解码装置31中的一部分或全部作为LSI (Large Scale Integration:大规模集成电路)等集成电路而实现。运动图像编码装置11、运动图像解码装置31的各功能块可以单独地处理器化,也可以将一部分或全部集成来处理器化。此外,集成电路化的方法并不限于LSI,也可以通过专用电路或通用处理器来实现。此外,在随着半导体技术的进步而出现代替LSI的集成电路化的技术的情况下,也可以使用基于该技术的集成电路。

[0232] 以上,参照附图对该发明的一个实施方式详细地进行了说明,但具体构成并不限于上述实施方式,在不脱离该发明的主旨的范围内,可以进行各种设计变更等。

[0233] 本发明的实施方式并不限于上述的实施方式,能在权利要求所示的范围内进行各种变更。即,将在权利要求所示的范围内经过适当变更的技术方案组合而得到的实施方式也包括在本发明的技术范围内。

工业上的可利用性

[0234] 本发明的实施方式能优选地应用于对将图像数据编码而得到的编码数据进行解码的运动图像解码装置,以及生成将图像数据编码而得到的编码数据的运动图像编码装置。此外,能优选地应用于由运动图像编码装置生成并被运动图像解码装置参照的编码数据的数据结构。

附图标记说明
31运动图像解码装置
301熵解码部
302参数解码部
3022 CU解码部
3024 TU解码部
303帧间预测参数推导部
305、107环路滤波器
306、109参照图片存储器
307、108预测参数存储器
308、101预测图像生成部
309帧间预测图像生成部
3092 OOB处理部
30921 OOB判定部
30922 OOB掩码推导部
311、105逆量化/逆变换部
312、106加法部
320预测参数推导部
11运动图像编码装置
102减法部
103变换/量化部
104熵编码部
110编码参数确定部
111参数编码部
112帧间预测参数编码部
120预测参数推导部

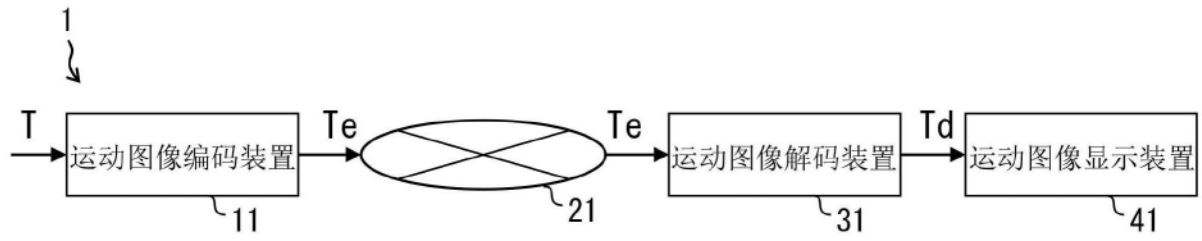


图1

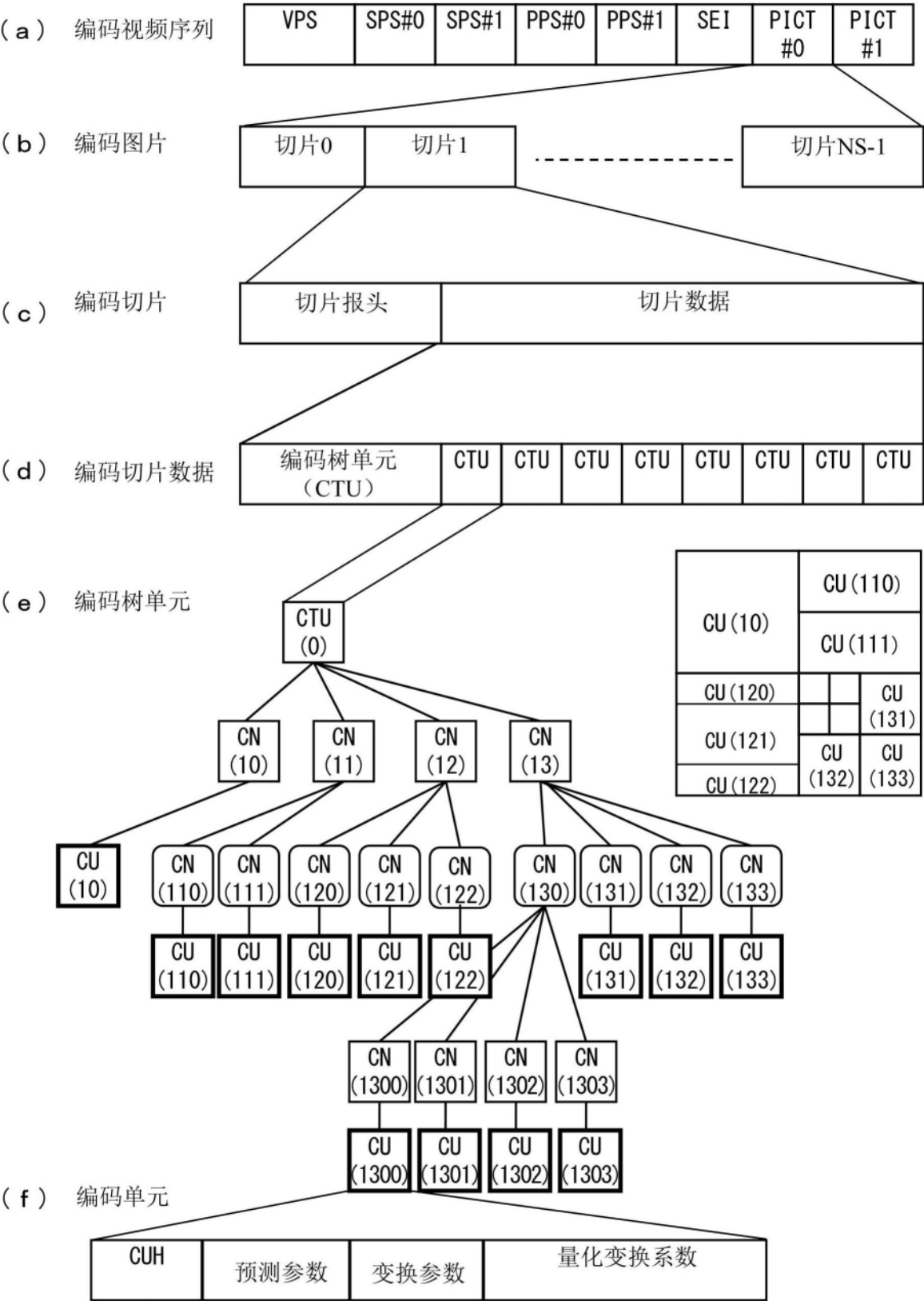


图2

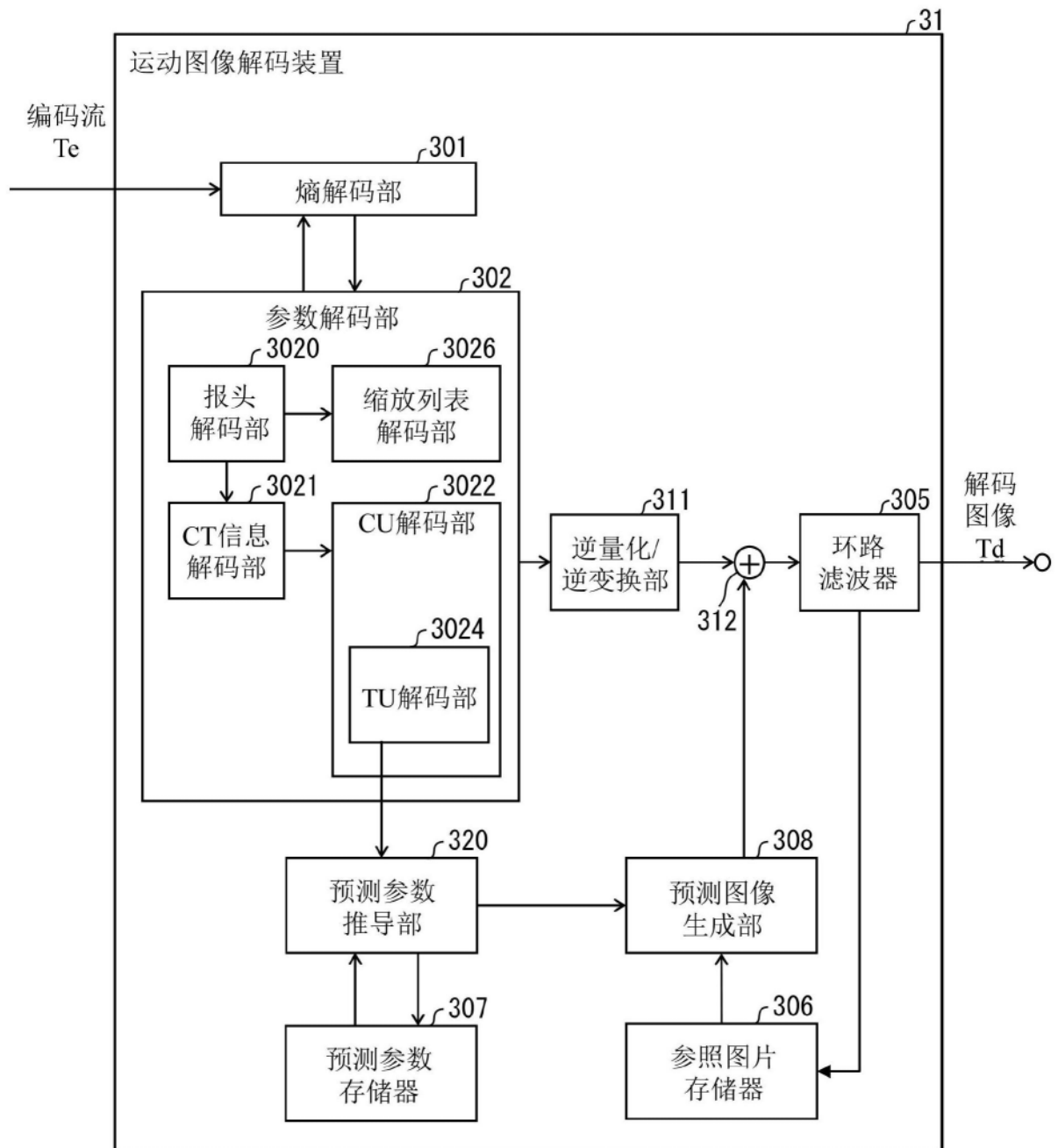


图3

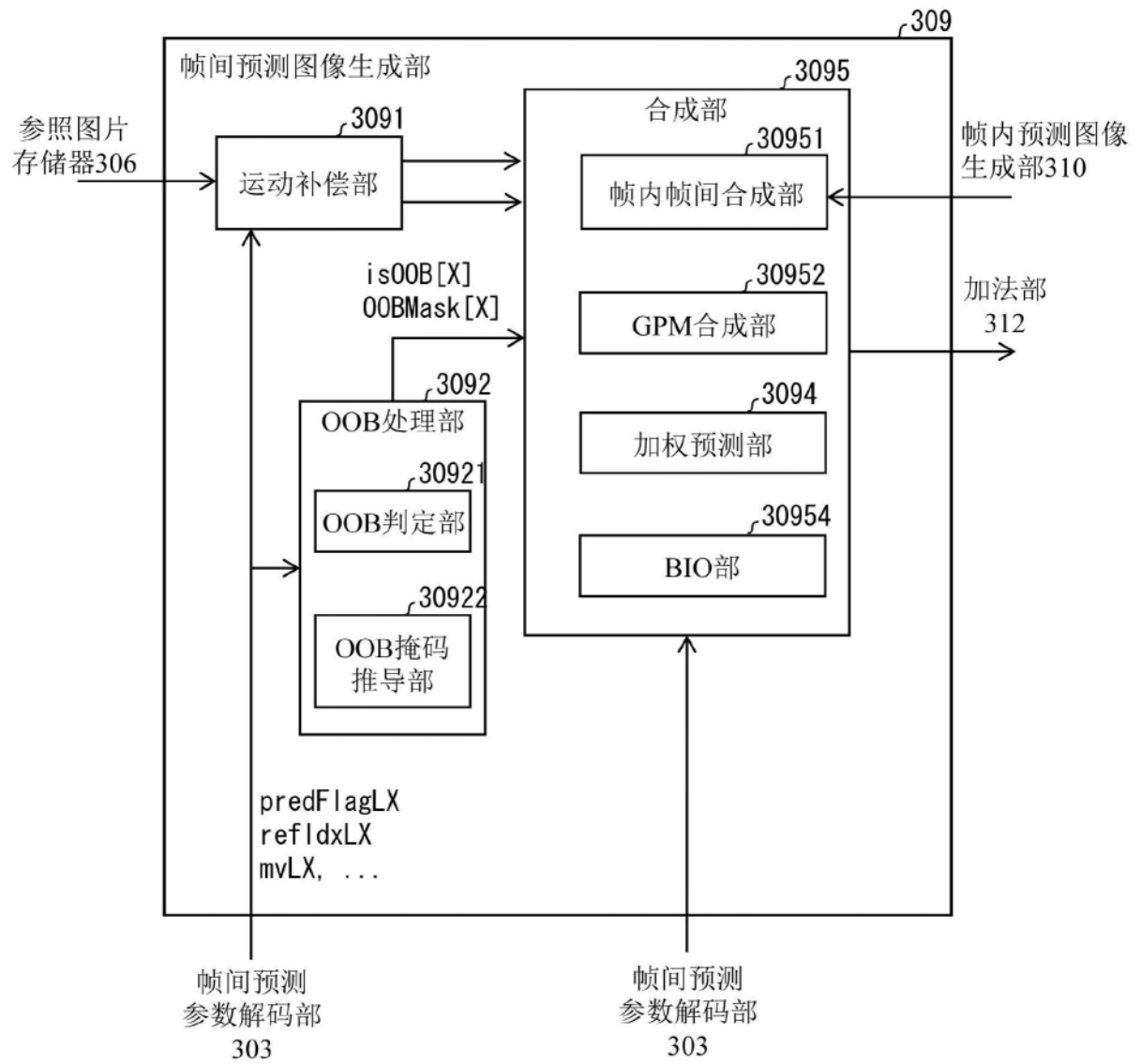


图4

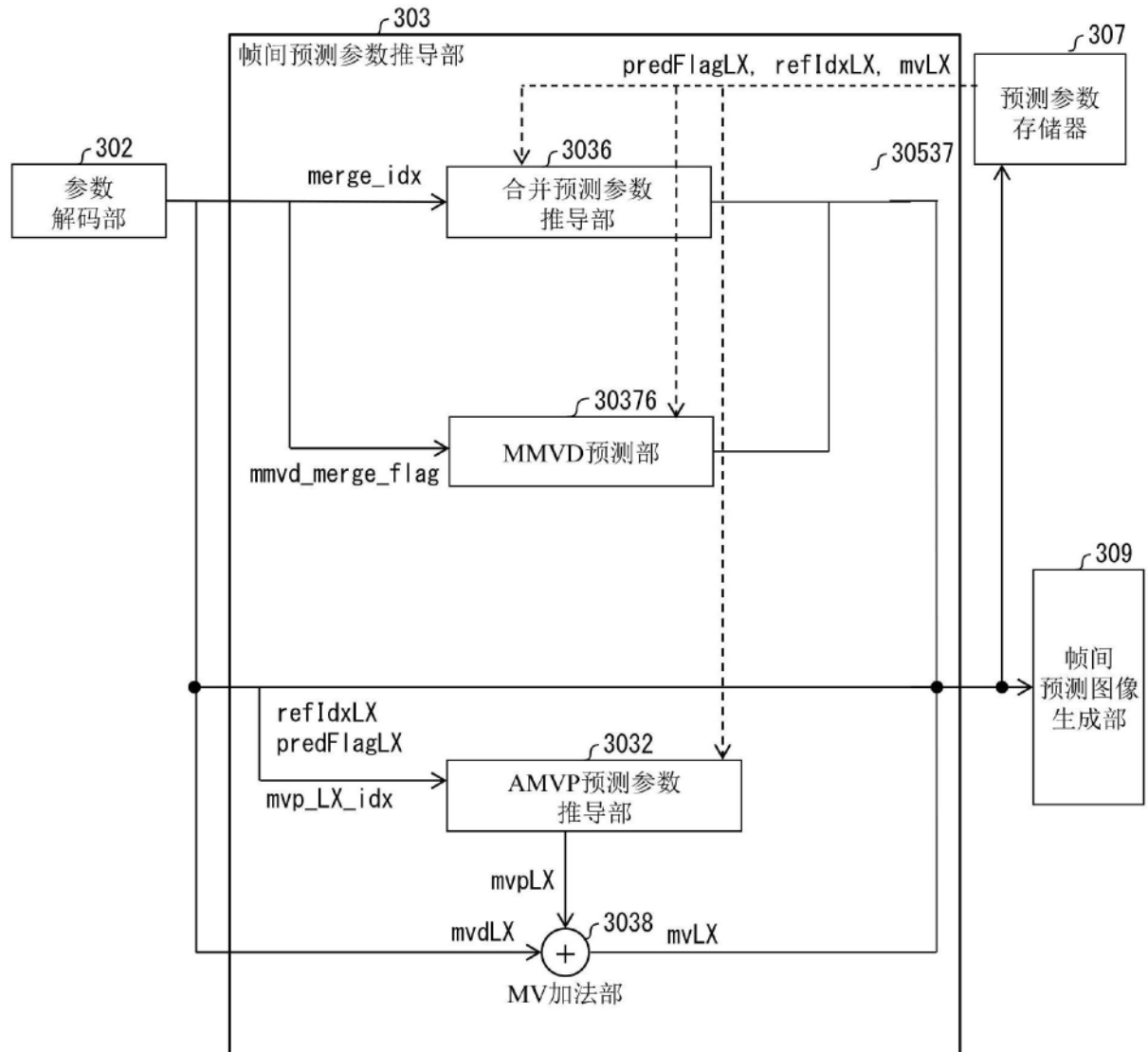


图5

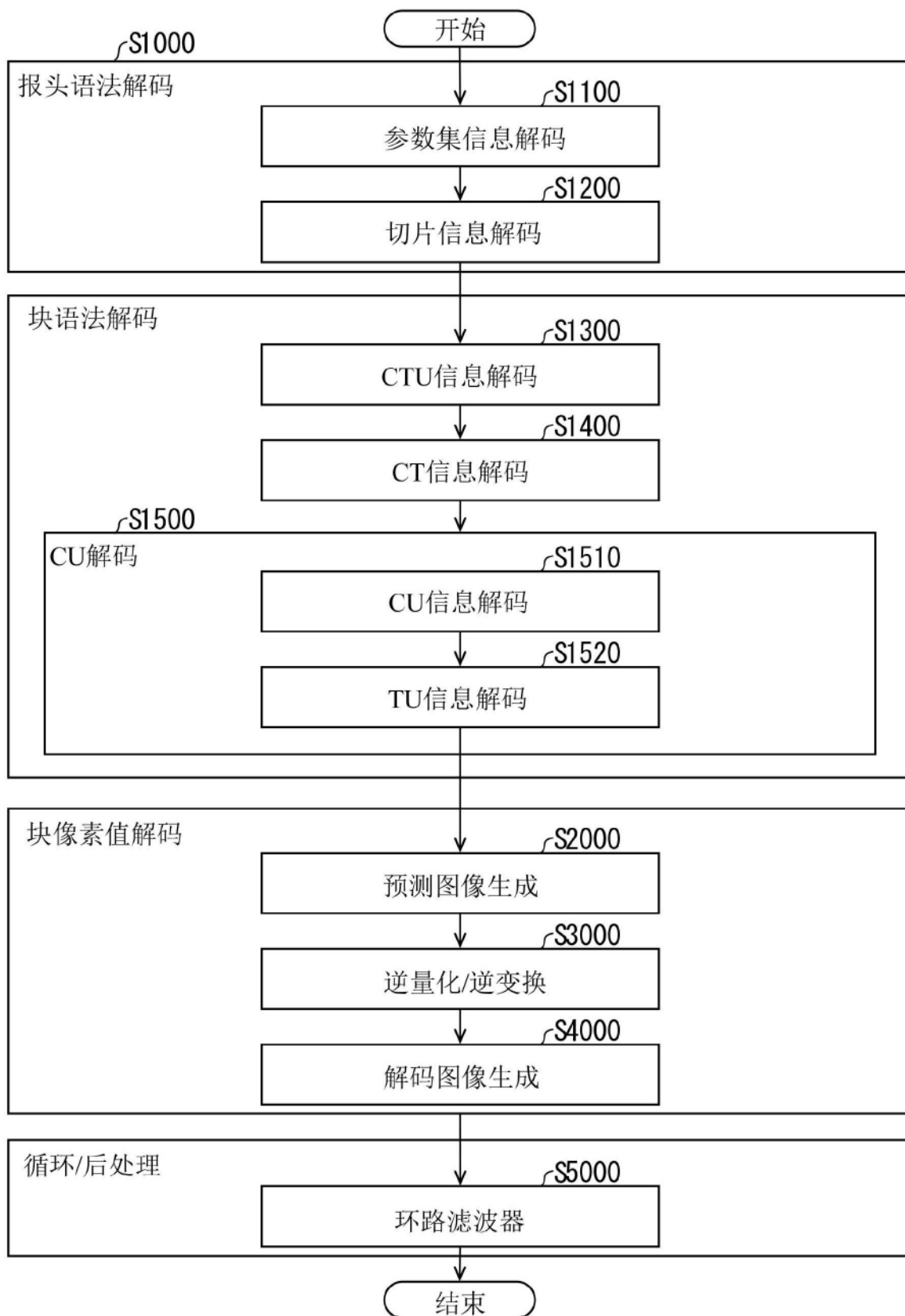


图6

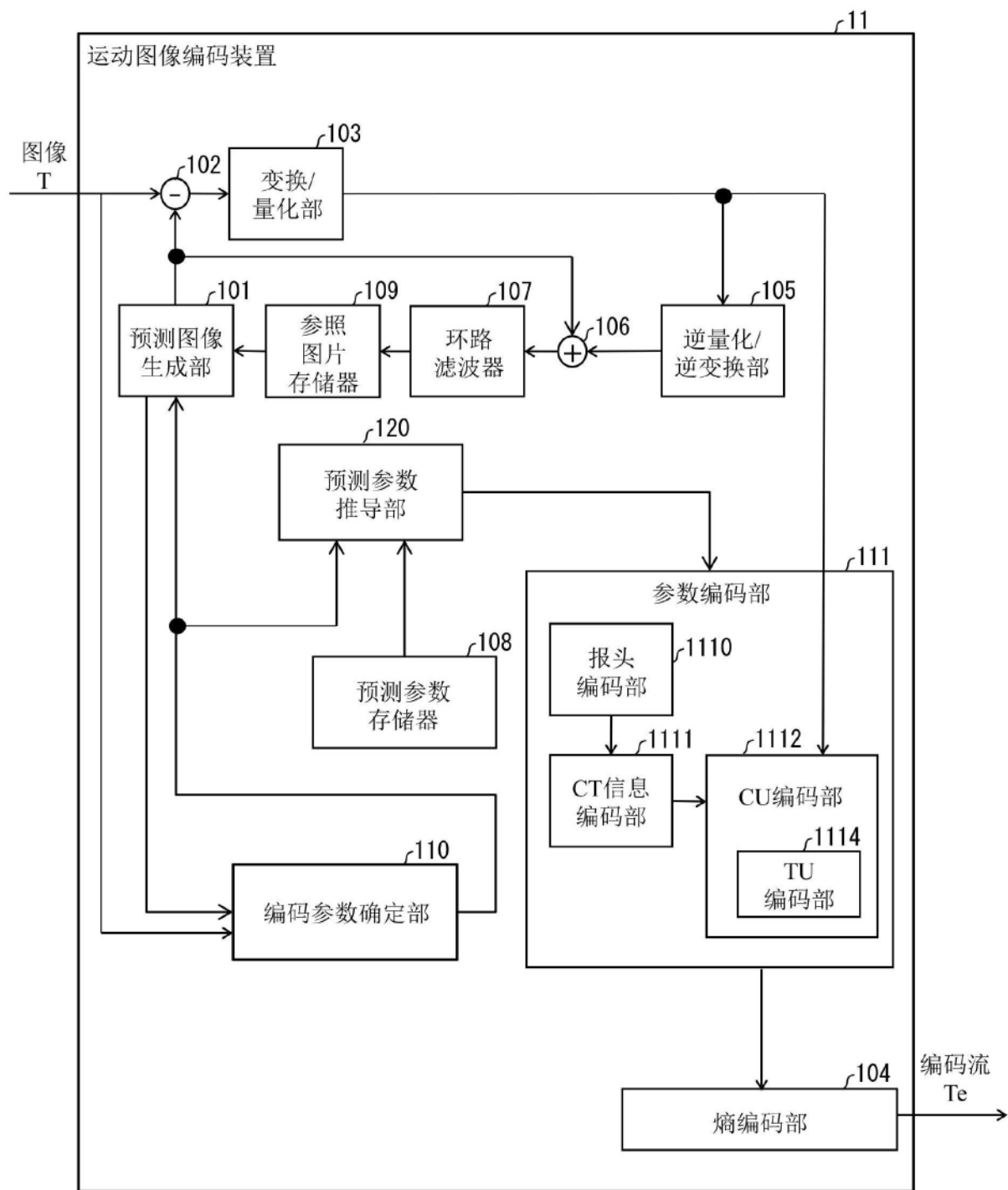
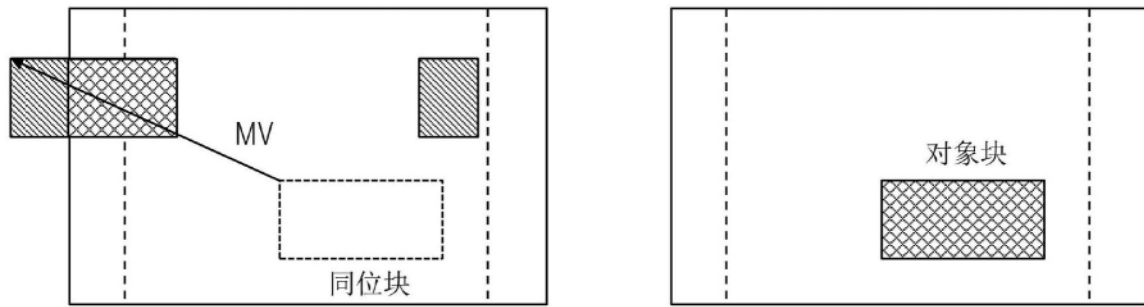


图7

(a)



(b)

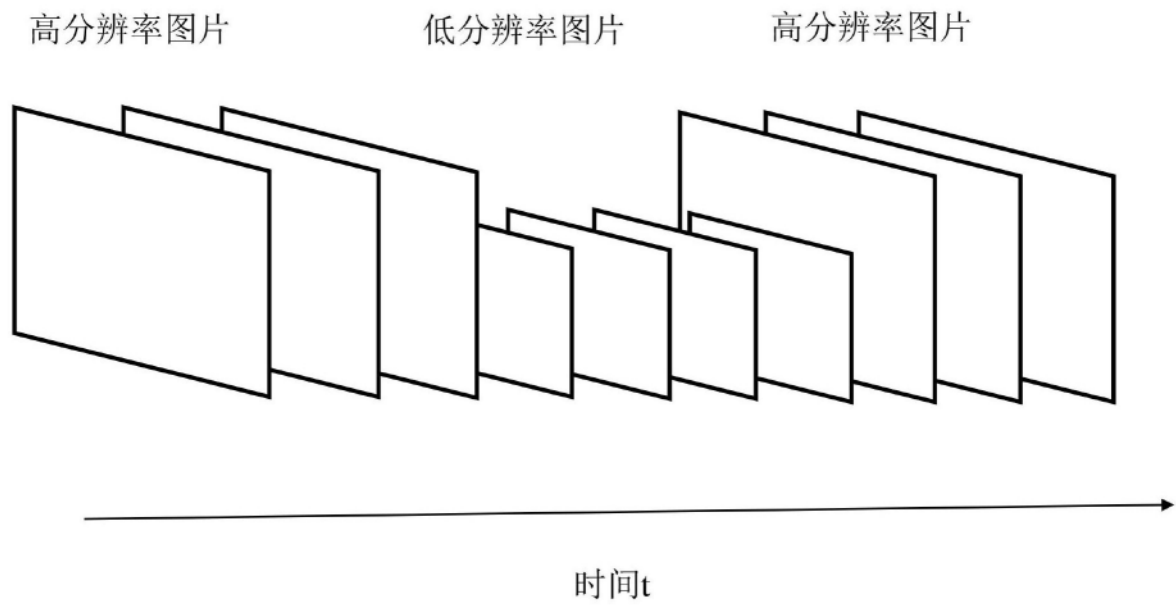


图8

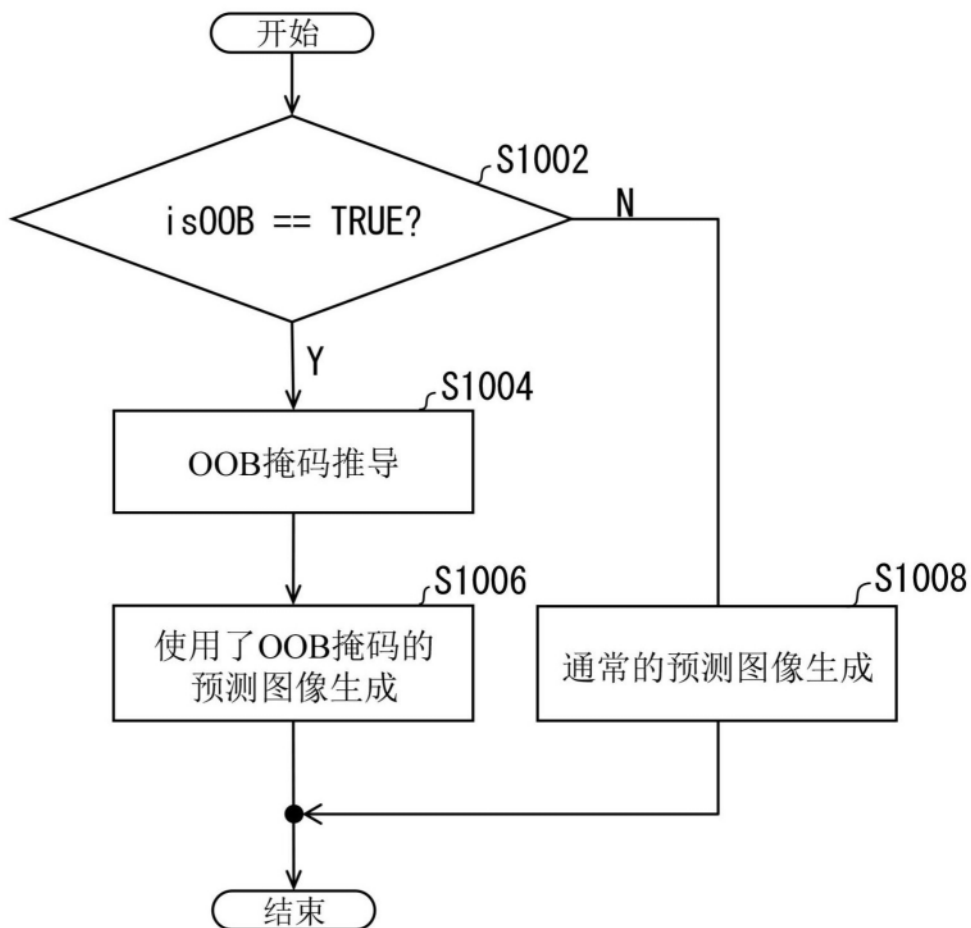


图9

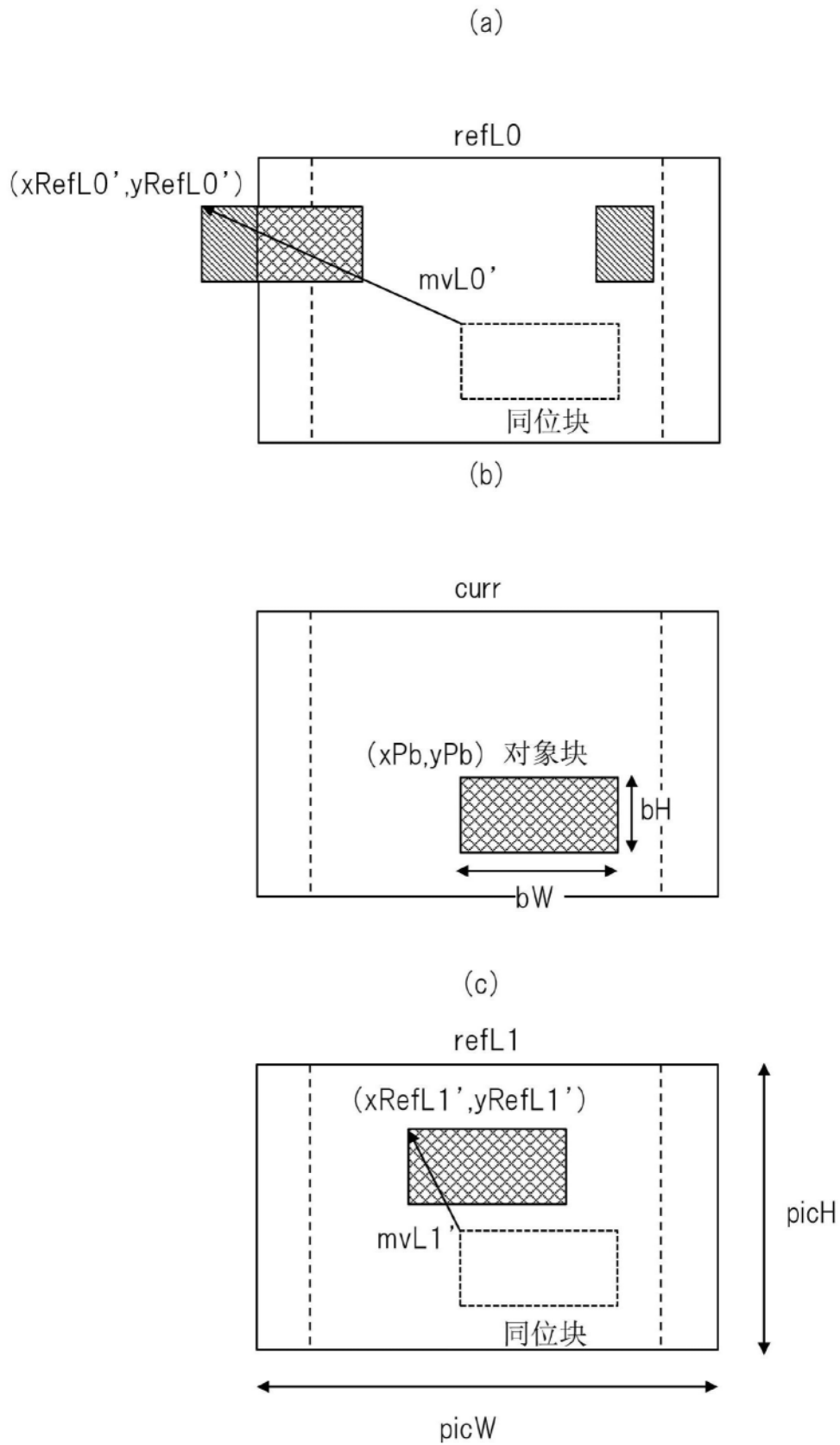


图10