



(12) 发明专利申请

(10) 申请公布号 CN 116530084 A

(43) 申请公布日 2023. 08. 01

(21) 申请号 202180080772.9

(22) 申请日 2021.12.02

(30) 优先权数据

10-2020-0166450 2020.12.02 KR

(85) PCT国际申请进入国家阶段日

2023.05.31

(86) PCT国际申请的申请数据

PCT/KR2021/018073 2021.12.02

(87) PCT国际申请的公布数据

W02022/119333 KO 2022.06.09

(71) 申请人 现代自动车株式会社

地址 韩国首尔

申请人 起亚株式会社

梨花女子大学校产学协力团

(74) 专利代理机构 北京戈程知识产权代理有限公司 11314

专利代理师 程伟 甄雁翔

(51) Int.Cl.

H04N 19/577 (2006.01)

(72) 发明人 姜制远 朴胜煜

权利要求书2页 说明书17页 附图15页

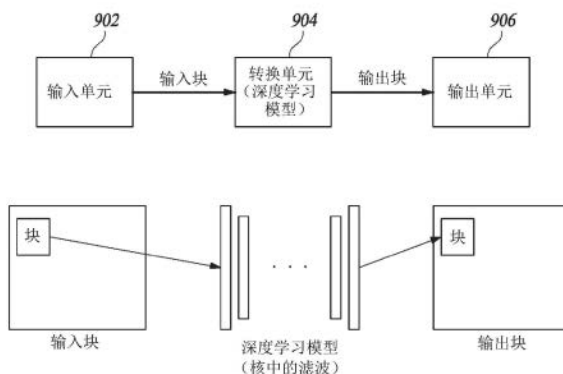
(54) 发明名称

利用基于块的深度学习模型的视频编解码器

(57) 摘要

本公开涉及利用基于块的深度学习模型的视频编解码器。本实施方案提供了一种视频编解码器,其在通过利用深度学习模型处理视频块期间,通过层叠或封装相应的YUV视频块来生成超级块,然后将生成的超级块输入到深度学习模型中,其中,在深度学习模型内部的卷积期间,根据构成超级块的YUV块的特征,对视频块进行不同的处理。

900



1. 一种由计算装置执行的用于基于深度学习技术处理视频块的方法,所述方法包括:
获得视频输入块,所述视频输入块包括Y块、U块和V块,所述Y块、U块和V块分别具有采样率格式为4:2:0或4:4:4的Y信号、U信号和V信号;
通过堆叠或组合Y块、U块和V块来生成输入块;
将输入块输入到至少一个深度学习模型;
通过基于至少一个或更多个深度学习模型执行卷积运算来从输入块生成输出块;以及
从输出块生成视频输出块。
2. 根据权利要求1所述的方法,其中,生成输入块包括:
当Y信号、U信号和V信号为4:2:0格式时,
扩大U块和V块以在大小上匹配Y块;以及
堆叠Y块、扩大的U块和扩大的V块。
3. 根据权利要求2所述的方法,其中,扩大包括:
通过上下或从左到右镜像U块并且组合由镜像产生的四个U块来重复U块4次,从而扩大U块以在大小上匹配Y块。
4. 根据权利要求2所述的方法,其中,扩大包括:
将U块居中,然后通过填充U块的外周来扩大U块以在大小上匹配Y块,同时用与U块同位的Y块的值填补U块的外周。
5. 根据权利要求2所述的方法,其中,扩大包括:
将U块定位在扩大的U块的一个象限中,然后利用与U块同位的Y块的值来填补扩大的U块的剩余象限。
6. 根据权利要求1所述的方法,其中,生成输入块包括:
当Y信号、U信号和V信号为4:2:0格式时,
将Y块四等分以在大小上匹配U块;以及
堆叠四等分的Y块、U块和V块。
7. 根据权利要求6所述的方法,其中,四等分包括:
在水平和垂直方向上抽取构成Y块的样本,以生成四等分的Y块。
8. 根据权利要求1所述的方法,其中,生成输入块包括:
当Y信号、U信号和V信号为4:2:0格式时,
通过利用U块和V块生成大小与Y块相等的超级块;以及
堆叠超级块和Y块。
9. 根据权利要求8所述的方法,其中,生成超级块包括:
上下组合U块和V块,然后在水平方向上对U块和V块进行上采样,或者从一侧到另一侧组合U块和V块,然后在垂直方向上对U块和V块进行上采样。
10. 根据权利要求1所述的方法,其中,生成输入块包括:
当Y信号、U信号和V信号为4:2:0格式时,通过组合Y块、U块和V块来生成超级块。
11. 根据权利要求1所述的方法,其中,生成输入块包括:
当Y信号、U信号和V信号为4:4:4格式时,通过堆叠Y块、U块和V块来生成输入块。
12. 根据权利要求1所述的方法,其中,将输入块输入包括:
利用三个深度学习模型,并且将Y块、U块和V块的每个输入到三个深度学习模型的每个

中。

13. 根据权利要求1所述的方法,其中,将输入块输入包括:

利用两个深度学习模型,并且将Y块输入到一个深度学习模型中,通过堆叠U块和V块来生成输入块,然后将输入块输入到剩余的深度学习模型中。

14. 根据权利要求1所述的方法,其中,将输入块输入包括:

利用两个深度学习模型,并且将Y块输入到一个深度学习模型中,通过利用U块和V块来生成超级块,然后将超级块输入到剩余的深度学习模型中。

15. 根据权利要求1所述的方法,其中,生成输出块包括以下各项,以执行卷积运算:

填充输入块的外周;

设置输入块的步长值;以及

通过利用预设的核来对输入块进行滤波。

16. 根据权利要求15所述的方法,其中,填充包括:

当输入块是帧内预测块时,利用来自与输入块邻近的周边样本中已经编码的样本来填充输入块。

17. 根据权利要求15所述的方法,其中,填充包括:

当输入块是帧间预测块时,利用已经编码的先前帧中的同位块的样本来填充输入块。

18. 根据权利要求15所述的方法,其中,设置步长值包括:

基于输入块的大小或色度分量来设置输入块的步长值。

19. 根据权利要求15所述的方法,其中,预设的核具有基于输入块的大小或色度分量而设置的核大小。

20. 根据权利要求1所述的方法,进一步包括:

除输入块外,输入额外信息,所述额外信息为块分区结构或编码的图,所述编码的图为量化参数、图像顺序计数(POC)或预测模式。

21. 根据权利要求20所述的方法,其中,输入额外信息是响应于额外信息确定为块分区结构,堆叠输入块和块分区结构并将输入块和块分区结构的堆叠输入到深度学习模型。

22. 根据权利要求20所述的方法,其中,输入额外信息是响应于额外信息确定为编码的图,堆叠输入块和编码的图并将输入块和编码的图的堆叠输入到深度学习模型。

23. 一种用于基于深度学习技术处理视频块的装置,其包括:

输入单元,其配置为获得视频输入块,并且通过堆叠或组合Y块、U块和V块来生成输入块,所述Y块、U块和V块包括在视频输入块中并且分别具有采样率格式为4:2:0或4:4:4的Y信号、U信号和V信号;

转换单元,其配置为通过基于至少一个或更多个深度学习模型执行卷积运算来从输入块生成输出块;以及

输出单元,其配置为从输出块生成视频输出块。

利用基于块的深度学习模型的视频编解码器

技术领域

[0001] 本发明涉及利用基于块的深度学习模型的视频编解码器。

背景技术

[0002] 本部分中的陈述仅提供与本发明有关的背景技术信息,并不一定构成现有技术。

[0003] 由于视频数据与音频数据或静止影像数据相比具有较大的数据量,视频数据需要大量的硬件资源(包括存储器)来存储或发送未经压缩处理的视频数据。

[0004] 相应地,编码器通常用于压缩并存储或发送视频数据。解码器接收压缩的视频数据,解压接收到的压缩的视频数据,并且播放解压的视频数据。视频压缩技术包括H.264/AVC、高效率视频编码(High Efficiency Video Coding, HEVC)和多功能视频编码(Versatile Video Coding, VVC),所述多功能视频编码(VVC)比HEVC的编码效率提高了大约30%或更多。

[0005] 然而,由于影像大小、分辨率和帧速率逐渐增加,要编码的数据量也在增多。相应地,需要一种与现有的压缩技术相比提供更高的编码效率和改善的影像增强效果的新的压缩技术。

[0006] 近年来,基于深度学习的图像处理技术已经应用于现有的编码基本技术。通过将基于深度学习的图像处理技术应用于现有的编码技术(特别是,诸如帧间预测、帧内预测、环内滤波、变换等的压缩技术),可以提高编码效率。代表性的应用示例包括基于由深度学习模型生成的虚拟参考帧的帧间预测,以及基于去噪模型的环内滤波。因此,需要进一步采用基于深度学习的图像处理技术来提高影像编码/解码的编码效率。

发明内容

[0007] 技术课题

[0008] 在一些实施方案中的本发明致力于提供一种视频编解码器,其在通过利用深度学习模型处理视频块时,通过堆叠或封装YUV视频块的相应视频块来生成超级块,并且将生成的超级块输入到深度学习模型。视频编解码器根据超级块的构成的YUV块的特征,在深度学习模型内执行卷积的过程中不同地处理输入。

[0009] 解决方法

[0010] 本发明的至少一个方面提供了由计算装置执行的用于基于深度学习技术来处理视频块的方法。该方法包括:获得视频输入块,所述视频输入块包括Y块、U块和V块,所述Y块、U块和V块分别具有采样率格式为4:2:0或4:4:4的Y信号、U信号和V信号。该方法还包括:通过堆叠或组合Y块、U块和V块来生成输入块。该方法还包括将输入块输入到至少一个深度学习模型。该方法还包括通过基于至少一个或更多个深度学习模型执行卷积运算来从输入块生成输出块。该方法还包括从输出块生成视频输出块。

[0011] 本发明的另一个方面提供了用于基于深度学习技术来处理视频块的装置。用于处理视频块的装置包括输入单元,所述输入单元配置为获得视频输入块,并且通过堆叠或组

合Y块、U块和V块来生成输入块,所述Y块、U块和V块包括在视频输入块中并且分别具有采样率格式为4:2:0或4:4:4的Y信号、U信号和V信号。用于处理视频块的装置还包括转换单元,所述转换单元配置为通过基于至少一个或更多个深度学习模型执行卷积运算来从输入块生成输出块。用于处理视频块的装置还包括输出单元,所述输出单元配置为从输出块生成视频输出块。

[0012] 发明效果

[0013] 如上所述,本实施方案提供了一种视频编解码器,其在通过堆叠或封装YUV视频块的相应视频块来生成超级块,并且将生成的超级块输入到深度学习模型。视频编解码器根据超级块的构成的YUV块的特征,在深度学习模型内执行卷积的过程中不同地处理输入,以提高编码效率并降低复杂度和存储要求。

附图说明

[0014] 图1是可以实现本发明技术的视频编码装置的框图。

[0015] 图2示出利用四叉树加二叉树三叉树(QTBT)结构来对块进行分区的方法。

[0016] 图3a和图3b示出包括宽角度帧内预测模式的多个帧内预测模式。

[0017] 图4示出当前块的相邻块。

[0018] 图5是可以实现本发明技术的视频解码装置的框图。

[0019] 图6是示出根据本发明的至少一个实施方案的卷积层的操作的示意图。

[0020] 图7是示出根据本发明的至少一个实施方案的反卷积层的操作的示意图。

[0021] 图8是示出根据本发明的至少一个实施方案的池化层的操作的示意图。

[0022] 图9是概念性地示出根据本发明的至少一个实施方案的视频块处理装置的框图。

[0023] 图10是示出根据本发明的至少一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0024] 图11是示出根据本发明的至少一个实施方案的当构造输入块时扩大U块的方法的示意图。

[0025] 图12是示出根据本发明的另一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0026] 图13是示出根据本发明的另一个实施方案的四等分Y块的方法的示意图。

[0027] 图14是示出根据本发明的另一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0028] 图15是示出根据本发明的另一个实施方案的当构造输入块时构造超级块的方法的示意图。

[0029] 图16是示出根据本发明的另一个实施方案的通过利用Y、U和V块来构造超级块的方法的示意图。

[0030] 图17是示出根据本发明的又一个实施方案的通过利用Y、U和V块来构造超级块的方法的示意图。

[0031] 图18是示出根据本发明的另一个实施方案的利用块分区结构作为深度学习模型的输入的方法的示意图。

[0032] 图19是示出根据本发明的另一个实施方案的利用块分区结构作为深度学习模型

的分支输入的方法的示意图。

[0033] 图20是示出根据本发明的另一个实施方案的利用编码的图作为深度学习模型的分支输入的方法的示意图。

[0034] 图21是根据本发明的至少一个实施方案的处理视频块的方法的流程图。

具体实施方式

[0035] 在下文中,将参考所附说明性附图对本发明的一些实施方案进行详细描述。在下面的描述中,相同的附图标记表示相同的元件,尽管元件在不同的附图中示出。此外,在一些实施方案的以下描述中,当认为相关的已知组件和功能的详细描述模糊了本发明的主题时,为了清楚和简洁起见,可以省略对相关的已知组件和功能的详细描述。

[0036] 图1是可以实现本发明技术的视频编码装置的框图。在下文中,参考图1的图示,对视频编码装置以及该装置的子组件进行描述。

[0037] 编码装置可以包括:图像分割器110、预测器120、减法器130、变换器140、量化器145、重排单元150、熵编码器155、逆量化器160、逆变换器165、加法器170、环路滤波单元180和存储器190。

[0038] 编码装置的每个组件可以实现为硬件或软件,或者实现为硬件和软件的组合。另外,每个组件的功能可以实现为软件,并且微处理器也可以实现为执行对应于每个组件的软件的功能。

[0039] 一个视频由包括多个图像的一个或更多个序列组成。每个图像分割为多个区域,并且对每个区域执行编码。例如,一个图像分割为一个或更多个瓦片(tile)或/和切片(slice)。这里,一个或更多个瓦片可以定义为瓦片组。每个瓦片或/和切片分割为一个或更多个编码树单元(coding tree unit,CTU)。另外,每个CTU通过树结构分割为一个或更多个编码单元(coding unit,CU)。应用于每个CU的信息被编码为CU的语法,并且共同应用于包括在一个CTU中的CU的信息被编码为CTU的语法。另外,共同应用于一个切片中的所有块的信息被编码为切片头的语法,而应用于构成一个或更多个图像的所有块的信息被编码为图像参数集(Picture Parameter Set,PPS)或图像头。此外,多个图像共同参考的信息被编码为序列参数集(Sequence Parameter Set,SPS)。另外,一个或更多个SPS共同参考的信息被编码为视频参数集(Video Parameter Set,VPS)。此外,共同应用于一个瓦片或瓦片组的信息也可以被编码为瓦片或瓦片组头的语法。包括在SPS、PPS、切片头、瓦片或瓦片组头中的语法可以称为高级语法。

[0040] 图像分割器110确定CTU的大小。关于CTU的大小(CTU尺寸)的信息被编码为SPS或PPS的语法,并且被传送至视频解码装置。

[0041] 图像分割器110将构成视频的每个图像分割为具有预定大小的多个CTU,然后通过利用树结构递归地分割CTU。树结构中的叶节点成为CU,所述CU是编码的基本单元。

[0042] 树结构可以是四叉树(quadtrees,QT),其中较高节点(或父节点)被分割为具有相同大小的四个较低节点(或子节点)。树结构也可以是二叉树(binarytree,BT),其中较高的节点分割为两个较低的节点。树结构还可以是三叉树(ternarytree,TT),其中较高的节点以1:2:1的比率分割为三个较低的节点。树结构还可以是QT结构、BT结构和TT结构中的两个或更多个结构混合的结构。例如,可以利用四叉树加二叉树(quadtrees plus binarytree,

QTBT) 结构, 或者可以利用四叉树加二叉树三叉树 (quadtree plus binarytree ternarytree, QTBT) 结构。这里, BTBT被添加到树结构以称为多类型树 (multiple-type tree, MTT)。

[0043] 图2是用于描述通过利用QTBT结构来对块进行分割的方法的示意图。

[0044] 如图2所示, CTU可以首先分割为QT结构。四叉树分割可以是递归的, 直到分割块的大小达到QT中允许的叶节点的最小块大小 (MinQTSIZE)。由熵编码器155对指示QT结构的每个节点是否被分割为下层的四个节点的第一标志 (QT_split_flag) 进行编码, 并将其用信号通知视频解码装置。当QT的叶节点不大于BT中允许的根节点的最大块大小 (MaxBTSIZE) 时, 可以将叶节点进一步分割为BT结构或TT结构的至少一个。在BT结构和/或TT结构中可能存在多个分割方向。例如, 可以存在两个方向, 即, 水平地分割相应节点的块的方向以及竖直地分割相应节点的块的方向。如图2所示, 当MTT分割开始时, 由熵编码器155对指示节点是否被分割的第二标志 (mtt_split_flag)、以及在节点被分割的情况下额外指示分割方向 (竖直或水平) 的标志和/或指示分割类型 (二叉或三叉) 的标志进行编码, 并将其用信号通知视频解码装置。

[0045] 替代地, 在对指示每个节点是否被分割为下层的四个节点的第一标志 (QT_split_flag) 进行编码之前, 还可以对指示节点是否被分割的CU分割标志 (split_cu_flag) 进行编码。当CU分割标志 (split_cu_flag) 的值指示出每个节点没有被分割时, 相应节点的块成为分割树结构中的叶节点并成为CU, CU是编码的基本单元。当CU分割标志 (split_cu_flag) 的值指示出每个节点被分割时, 视频编码装置以上述方案对第一标志首先开始编码。

[0046] 当QTBT用作树结构的另一个示例时, 可以存在两种类型, 即将相应节点的块水平地分割为具有相同大小的两个块的类型 (即, 对称水平分割) 以及将相应节点的块竖直地分割为具有相同大小的两个块的类型 (即, 对称竖直分割)。由熵编码器155对指示BT结构的每个节点是否被分割为下层的块的分割标志 (split_flag) 和指示分割类型的分割类型信息进行编码, 并将其传送至视频解码装置。另一方面, 可以另外存在相应节点的块被分割为彼此非对称形式的两个块的类型。非对称形式可以包括相应节点的块被分割为大小比率为1:3的两个矩形块的形式, 或者还可以包括相应节点的块在对角线方向上被分割的形式。

[0047] CU可以根据从CTU分割的QTBT或QTBTBT而具有各种大小。在下文中, 与要编码或解码的CU (即, QTBTBT的叶节点) 相对应的块称为“当前块”。当采用QTBTBT分割时, 除了正方形形状之外, 当前块的形状也可以是矩形形状。

[0048] 预测器120对当前块进行预测以生成预测块。预测器120包括帧内预测器122和帧间预测器124。

[0049] 通常, 可以对图像中的当前块的每个进行预测地编码。通常, 可以通过利用帧内预测技术 (其利用来自包括当前块的图像的数据) 或帧间预测技术 (其利用来自在包括当前块的图像之前编码的图像的数据) 执行当前块的预测。帧间预测包括单向预测和双向预测两者。

[0050] 帧内预测器122通过利用在包括当前块的当前图像中位于当前块邻近的像素 (参考像素) 来预测当前块中的像素。根据预测方向, 存在多个帧内预测模式。例如, 如图3a所示, 多个帧内预测模式可以包括包含平面 (planar) 模式和DC模式的两种非方向模式, 并且可以包括65种方向模式。根据每种预测模式不同地定义要利用的相邻像素和算法等式。

[0051] 为了对具有矩形形状的当前块进行高效的方向预测,可以额外地利用图3b中虚线箭头所示的方向模式(#67至#80,帧内预测模式#-1至#-14)。方向模式可以称为“宽角度帧内预测模式(wide angle intra-prediction modes)”。在图3b中,箭头指示用于预测的相应参考样本,而非代表预测方向。预测方向与箭头指示的方向相反。在当前块具有矩形形状时,宽角度帧内预测模式是在没有额外比特传输的情况下在与特定方向模式相反的方向上执行预测的模式。在这种情况下,在宽角度帧内预测模式中,可以通过具有矩形形状的当前块的宽度与高度的比率来确定可用于当前块的一些宽角度帧内预测模式。例如,在当前块具有高度小于宽度的矩形形状时,具有小于45度的角度的宽角度帧内预测模式(帧内预测模式#67至#80)是可用的。在当前块具有宽度大于高度的矩形形状时,具有大于-135度的角度的宽角度帧内预测模式是可用的。

[0052] 帧内预测器122可以确定要用于对当前块进行编码的帧内预测。在一些示例中,帧内预测器122可以通过利用多个帧内预测模式来对当前块进行编码,并且还可以从测试模式中选择要利用的适当的帧内预测模式。例如,帧内预测器122可以通过利用对多个测试的帧内预测模式的率失真(rate-distortion)分析来计算率失真值,并且还可以在测试模式中选择具有最佳率失真特性的帧内预测模式。

[0053] 帧内预测器122在多个帧内预测模式中选择一个帧内预测模式,并且通过利用根据选择的帧内预测模式确定的相邻像素(参考像素)和算法等式来预测当前块。由熵编码器155对关于选择的帧内预测模式的信息进行编码并将其传送至视频解码装置。

[0054] 帧间预测器124通过利用运动补偿处理来生成当前块的预测块。帧间预测器124在比当前图像更早已被编码和解码的参考图像中搜索与当前块最相似的块,并且通过利用搜索到的块来生成当前块的预测块。另外,生成运动矢量(motion vector, MV),所述运动矢量与当前图像中的当前块和参考图像中的预测块之间的位移(displacement)相对应。通常,对亮度(luma)分量执行运动估计,并且基于亮度分量计算的运动矢量用于亮度分量和色度分量两者。由熵编码器155对包括参考图像的信息和关于用于预测当前块的运动矢量的信息的运动信息进行编码,并将其传送至视频解码装置。

[0055] 帧间预测器124还可以执行参考图像或参考块的插值,以增加预测的准确性。换句话说,通过向包括两个整数样本的多个连续整数样本应用滤波器系数来在两个连续整数样本之间插值子样本。在对插值的参考图像执行搜索与当前块最相似的块的处理时,对于运动矢量可以表示小数单位精度而不是整数样本单位精度。对于要编码的每个目标区域,例如,诸如切片、瓦片、CTU、CU等的单元,可以不同地设置运动矢量的精度或分辨率。当应用这种自适应运动矢量分辨率(adaptive motion vector resolution, AMVR)时,应该针对每个目标区域用信号通知关于要应用于每个目标区域的运动矢量分辨率的信息。例如,当目标区域是CU时,用信号通知关于应用于每个CU的运动矢量分辨率的信息。关于运动矢量分辨率的信息可以是代表要在以下描述的运动矢量差的精度的信息。

[0056] 另一方面,帧间预测器124可以通过利用双向预测来执行帧间预测。在双向预测的情况下,利用两个参考图像和代表与每个参考图像中的当前块最相似的块位置的两个运动矢量。帧间预测器124分别从参考图像列表0(RefPicList0)和参考图像列表1(RefPicList1)选择第一参考图像和第二参考图像。帧间预测器124还搜索与相应参考图像中的当前块最相似的块,以生成第一参考块和第二参考块。此外,通过对第一参考块和第二

参考块进行平均或加权平均来生成当前块的预测块。此外,包括关于用于预测当前块的两个参考图像的信息和关于两个运动矢量的信息的运动信息被传送至熵编码器155。这里,参考图像列表0可以由预恢复图像中按显示顺序在当前图像之前的图像构成,并且参考图像列表1可以由预恢复图像中按显示顺序在当前图像之后的图像构成。然而,尽管不特别限于此,但可以将显示顺序中在当前图像之后的预恢复图像额外地包括在参考图像列表0中。相反,在当前图像之前的预恢复图像也可以额外地包括在参考图像列表1中。

[0057] 为了最小化用于编码运动信息所消耗的比特量,可以利用各种方法。

[0058] 例如,在当前块的参考图像和运动矢量与相邻块的参考图像和运动矢量相同时,能够识别相邻块的信息被编码以将当前块的运动信息传送至视频解码装置。这种方法称为合并模式(merge mode)。

[0059] 在合并模式中,帧间预测器124从当前块的相邻块选择预定数量的合并候选块(在下文中,称为“合并候选”)。

[0060] 作为用于推导合并候选的相邻块,可以利用与当前图像中的当前块邻近的左侧块A0、左下侧块A1、上侧块B0、右上侧块B1和左上侧块B2的全部或一些,如图4所示。此外,除了当前块所在的当前图像之外,位于参考图像内的块(其可以与用于预测当前块的参考图像相同或不同)也可以用作合并候选。例如,参考图像内的当前块的同位块(co-located block)或与该同位块邻近的块可以额外地用作合并候选。如果通过上述方法选择的合并候选的数量小于预设数量,则将零矢量添加到合并候选。

[0061] 帧间预测器124通过利用相邻块来配置包括预定数量的合并候选的合并列表。从合并列表中包括的合并候选之中选择要用作当前块的运动信息的合并候选,并且生成用于识别选择的候选的合并索引信息。由熵编码器155对生成的合并索引信息进行编码并将其传送至视频解码装置。

[0062] 合并跳过模式是合并模式的特例。在量化之后,在用于熵编码的所有变换系数都接近零时,仅传输相邻块选择信息而不传输残差信号。通过利用合并跳过模式,对于具有轻微运动的影像、静止影像、屏幕内容影像等可以实现相对较高的编码效率。

[0063] 此后,合并模式和合并跳过模式统称为合并/跳过模式。

[0064] 用于编码运动信息的另一种方法是高级运动矢量预测(advanced motion vector prediction,AMVP)模式。

[0065] 在AMVP模式中,帧间预测器124通过利用当前块的相邻块来推导用于当前块的运动矢量的运动矢量预测候选。作为用于推导运动矢量预测候选的相邻块,可以利用与图4所示的当前图像中的当前块邻近的左侧块A0、左下侧块A1、上侧块B0、右上侧块B1和左上侧块B2的全部或一些。此外,除了当前块所在的当前图像之外,位于参考图像(其可以与用于预测当前块的参考图像相同或不同)内的块也可以用作用于推导运动矢量预测候选的相邻块。例如,可以利用参考图像内的当前块的同位块或与该同位块邻近的块。如果通过上述方法选择的运动矢量候选的数量小于预设数量,则将零矢量添加到运动矢量候选。

[0066] 帧间预测器124通过利用相邻块的运动矢量来推导运动矢量预测候选,并且通过利用运动矢量预测候选来确定当前块的运动矢量的运动矢量预测。另外,通过将当前块的运动矢量减去运动矢量预测来计算运动矢量差。

[0067] 可以通过将预定义函数(例如,中值和平均值计算等)应用于运动矢量预测候选来

获取运动矢量预测。在这种情况下,视频解码装置还知道预定义功能。此外,由于用于推导运动矢量预测候选的相邻块是已经完成编码和解码的块,所以视频解码装置也可能已经知道相邻块的运动矢量。因此,视频编码装置不需要对用于识别运动矢量预测候选的信息进行编码。相应地,在这种情况下,对关于运动矢量差的信息和关于用于预测当前块的参考图像的信息进行编码。

[0068] 另一方面,还可以通过选择运动矢量预测候选的任何一方的方案来确定运动矢量预测。在这种情况下,用于识别选择的运动矢量预测候选的信息与用于预测当前块的关于运动矢量差的信息以及关于参考图像的信息共同进行额外的编码。

[0069] 减法器130通过将当前块减去由帧内预测器122或帧间预测器124生成的预测块来生成残差块。

[0070] 变换器140将具有空域的像素值的残差块中的残差信号变换为频域的变换系数。变换器140可以通过利用残差块的整个大小作为变换单元来变换残差块中的残差信号,或者也可以将残差块分割为多个子块,并且通过利用子块作为变换单元来执行变换。替选地,将残差块划分成两个子块,即变换区域和非变换区域,以通过仅利用变换区域子块作为变换单元来变换残差信号。这里,变换区域子块可以是基于水平轴线(或垂直轴线)的具有大小比率为1:1的两个矩形块的一个。在这种情况下,由熵编码器155对指示出仅变换子块的标志(cu_sbt_flag)、以及方向(竖直/水平)信息(cu_sbt_horizontal_flag)和/或位置信息(cu_sbt_pos_flag)进行编码,并将其用信号通知视频解码装置。另外,变换区域子块的大小基于水平轴线(或垂直轴线)可以具有1:3的大小比率。在这种情况下,由熵编码器155对划分相应分割的标志(cu_sbt_quad_flag)进行额外地编码,并将其用信号通知视频解码装置。

[0071] 另一方面,变换器140可以在水平方向和垂直方向上单独地执行残差块的变换。对于该变换,可以利用各种类型的变换函数或变换矩阵。例如,用于水平变换和垂直变换的成对变换函数可以定义为多变换集(multiple transform set,MTS)。变换器140可以选择MTS中具有最高变换效率的一个变换函数对,并且在水平方向和垂直方向的每个上变换残差块。由熵编码器155对关于MTS中的变换函数对的信息(mts_idx)进行编码,并将其用信号通知视频解码装置。

[0072] 量化器145利用量化参数对从变换器140输出的变换系数进行量化,并且将量化的变换系数输出到熵编码器155。量化器145还可以对相关的残差块立即进行量化,而无需对任何块或帧进行变换。量化器145还可以根据变换块中的变换系数的位置来应用不同的量化系数(缩放值)。可以对应用于布置为二维的量化的变换系数的量化矩阵进行编码,并将其用信号通知视频解码装置。

[0073] 重排单元150可以对量化的残差值执行系数值的重新排列。

[0074] 重排单元150可以通过利用系数扫描将2D系数阵列改变为1D系数序列。例如,重排单元150可以利用锯齿形扫描(zig-zag scan)或对角线扫描(diagonal scan)来扫描DC系数到高频区域的系数以输出1D系数序列。根据变换单元的大小和帧内预测模式,也可以利用在列方向上扫描2D系数阵列的垂直扫描和在行方向上扫描2D块类型系数的水平扫描来代替锯齿形扫描。换句话说,根据变换单元的大小和帧内预测模式,可以在锯齿形扫描、对角线扫描、垂直扫描和水平扫描中确定要利用的扫描方法。

[0075] 熵编码器155通过利用包括基于上下文的自适应二进制算术编码(Context-based Adaptive Binary Arithmetic Code,CABAC)、指数哥伦布(Exponential Golomb)等的各种编码方案来对从重排单元150输出的1D量化的变换系数的序列进行编码,以生成比特流。

[0076] 此外,熵编码器155对与块分割相关的信息(例如,CTU大小、CTU分割标志、QT分割标志、MTT分割类型和MTT分割方向等)进行编码,以使得视频解码装置能够等同于视频编码装置来分割块。此外,熵编码器155对关于指示当前块是由帧内预测编码还是由帧间预测编码的预测类型的信息进行编码。熵编码器155根据预测类型对帧内预测信息(即,关于帧内预测模式的信息)或帧间预测信息(在合并模式的情况下的合并索引,以及在AMVP模式的情况下的关于参考图像索引和运动矢量差的信息)进行编码。此外,熵编码器155对与量化相关的信息(即,关于量化参数的信息和关于量化矩阵的信息)进行编码。

[0077] 逆量化器160对从量化器145输出的量化的变换系数进行逆量化,以生成变换系数。逆变换器165将从逆量化器160输出的变换系数从频域变换到空域,以恢复残差块。

[0078] 加法器170将恢复的残差块和由预测器120生成的预测块相加,以恢复当前块。在对下一个块进行帧内预测时,恢复的当前块中的像素可以用作参考像素。

[0079] 环路滤波单元180对恢复的像素执行滤波,以减少由于基于块的预测和变换/量化而发生的块伪影(blocking artifacts)、振铃伪影(ringing artifacts)、模糊伪影(blurring artifacts)等。作为环内滤波器的环路滤波单元180可以包括去块滤波器182、样本自适应偏移(sample adaptive offset,SAO)滤波器184和自适应环路滤波器(adaptive loop filter,ALF)186的全部或一些。

[0080] 去块滤波器182对恢复的块之间的边界进行滤波,以去除由于块单元编码/解码而发生的块伪影(blocking artifacts),并且SAO滤波器184和ALF 186对去块滤波的视频进行额外的滤波。SAO滤波器184和ALF 186是用于补偿由于有损编码(lossy coding)而发生的恢复的像素与原始像素之间的差异的滤波器。SAO滤波器184应用偏移作为CTU单位,以增强主观图像质量和编码效率。另一方面,ALF 186执行块单元滤波,并且通过划分相应块的边界和变化量的程度来应用不同的滤波器以补偿失真。可以对关于要用于ALF的滤波器系数的信息进行编码,并将其用信号通知视频解码装置。

[0081] 通过去块滤波器182、SAO滤波器184和ALF 186滤波的恢复的块存储在存储器190中。当一个图像中的所有块被恢复时,恢复的图像可以用作用于对随后要被编码的图像内的块进行帧间预测的参考图像。

[0082] 图5是可以实现本发明技术的视频解码装置的功能框图。在下文中,参考图5,描述了视频解码装置和该装置的子组件。

[0083] 视频解码装置可以包括熵解码器510、重排单元515、逆量化器520、逆变换器530、预测器540、加法器550、环路滤波单元560和存储器570。

[0084] 类似于图1的视频编码装置,视频解码装置的每个组件可以实现为硬件或软件,或者实现为硬件和软件的组合。另外,每个组件的功能可以实现为软件,并且微处理器也可以实现为执行对应于每个组件的软件的功能。

[0085] 熵解码器510通过解码由视频编码装置生成的比特流来提取与块分割相关的信息,以确定要解码的当前块,并且提取恢复当前块所需的预测信息和关于残差信号的信息。

[0086] 熵解码器510通过从序列参数集(SPS)或图像参数集(PPS)提取关于CTU大小的信

息来确定CTU的大小,并且将图像分割为具有确定大小的CTU。此外,CTU被确定为树结构的最高层(即,根节点),并且提取CTU的分割信息以通过利用树结构来分割CTU。

[0087] 例如,当通过利用QTBT结构来分割CTU时,首先提取与QT的分割相关的第一标志(QT_split_flag),以将每个节点分割为下层的四个节点。另外,相对于与QT的叶节点相对应的节点来提取与MTT的分割相关的第二标志(mtt_split_flag)、分割方向(竖直/水平)和/或分割类型(二叉/三叉),以将相应的叶节点分割为MTT结构。结果,QT的叶节点下方的每个节点被递归地分割为BT或TT结构。

[0088] 作为另一个示例,当通过利用QTBT结构来分割CTU时,提取指示是否分割CU的CU分割标志(split_cu_flag)。当分割相应块时,还可以提取第一标志(QT_split_flag)。在分割过程期间,对于每个节点,在递归QT分割0次或更多次之后可能发生0次或更多次的递归MTT分割。例如,对于CTU,MTT分割可以立即发生,或者相反,也可以仅发生多次QT分割。

[0089] 作为另一个示例,当通过利用QTBT结构来分割CTU时,提取与QT的分割相关的第一标志(QT_split_flag),以将每个节点分割为下层的四个节点。另外,提取指示将与QT的叶节点相对应的节点是否进一步分割为BT的分割标志(split_flag)以及分割方向信息。

[0090] 另一方面,当熵解码器510通过利用树结构的分割来确定要解码的当前块时,熵解码器510提取关于指示当前块是被帧内预测还是被帧间预测的预测类型的信息。当预测类型信息指示帧内预测时,熵解码器510提取用于当前块的帧内预测信息(帧内预测模式)的语法元素。当预测类型信息指示帧间预测时,熵解码器510提取代表帧间预测信息的语法元素的信息,即,运动矢量和运动矢量参考的参考图像。

[0091] 此外,熵解码器510提取量化相关的信息并且提取关于量化的当前块的变换系数的信息作为关于残差信号的信息。

[0092] 重排单元515可以以由视频编码装置执行的系数扫描次序的相反顺序,将由熵解码器510熵解码的1D量化的变换系数的序列再次改变为2D系数阵列(即,块)。

[0093] 逆量化器520对量化的变换系数进行逆量化,并且通过利用量化参数对量化的变换系数进行逆量化。逆量化器520还可以向布置为2D的量化的变换系数应用不同的量化系数(缩放值)。逆量化器520可以通过将来自视频编码装置的量化系数(缩放值)的矩阵应用于量化的变换系数的2D阵列来执行逆量化。

[0094] 逆变换器530通过将逆量化的变换系数从频域逆变换到空域来恢复残差信号,以生成当前块的残差块。

[0095] 此外,当逆变换器530对变换块的部分区域(子块)进行逆变换时,逆变换器530提取仅对变换块的子块进行变换的标志(cu_sbt_flag)、子块的方向(竖直/水平)信息(cu_sbt_horizontal_flag)和/或子块的位置信息(cu_sbt_pos_flag)。逆变换器530还将相应子块的变换系数从频域逆变换到空域以恢复残差信号,并且用值“0”填充未被逆变换的区域作为残差信号,以生成当前块的最终残差块。

[0096] 此外,当应用MTS时,逆变换器530通过利用从视频编码装置用信号通知的MTS信息(mts_idx)来确定在水平方向和竖直方向的每个上要应用的变换索引或变换矩阵。逆变换器530还通过利用确定的变换函数在水平方向和竖直方向上对变换块中的变换系数执行逆变换。

[0097] 预测器540可以包括帧内预测器542和帧间预测器544。在当前块的预测类型是帧

内预测时,激活帧内预测器542,而在当前块的预测类型是帧间预测时,激活帧间预测器544。

[0098] 帧内预测器542根据从熵解码器510提取的帧内预测模式的语法元素来确定多个帧内预测模式中当前块的帧内预测模式。帧内预测器542还根据帧内预测模式通过利用当前块的相邻参考像素来预测当前块。

[0099] 帧间预测器544通过利用从熵解码器510提取的帧间预测模式的语法元素来确定当前块的运动矢量和运动矢量参考的参考图像。

[0100] 加法器550通过将逆变换器530输出的残差块与从帧间预测器544或帧内预测器542输出的预测块相加来恢复当前块。在对后续要被解码的块进行帧内预测时,恢复的当前块内的像素用作参考像素。

[0101] 作为环内滤波器的环路滤波单元560可以包括去块滤波器562、SAO滤波器564和ALF 566。去块滤波器562对恢复的块之间的边界执行去块滤波,以便去除由于块单元解码而发生的块伪影。SAO滤波器564和ALF 566在去块滤波之后对恢复的块执行额外的滤波,以便补偿由于有损编码而发生的恢复的像素与原始像素之间的差异。通过利用关于从比特流解码的滤波器系数的信息来确定ALF的滤波器系数。

[0102] 通过去块滤波器562、SAO滤波器564和ALF 566滤波的恢复的块存储在存储器570中。当一个图像中的所有块被恢复时,恢复的图像可以用作用于对随后要被编码的图像内的块进行帧间预测的参考图像。

[0103] 本发明在一些实施方案中涉及如上所述的编码和解码视频影像。更具体地,本发明提供了一种视频编解码器,其在通过利用深度学习模型处理视频块时,通过堆叠或封装相应的YUV视频块来生成超级块,然后将生成的超级块输入到深度学习模型中。视频编解码器在深度学习模型内部执行卷积的过程中,根据构成超级块的YUV块的特征,对超级块进行不同的处理。

[0104] 当将深度学习技术应用于现有的视频编码框架(例如,环内滤波、帧内预测、帧间预测、变换和量化)时,可以将整个视频帧输入到深度学习模型中进行处理。然而,当在每个块上应用深度学习技术时,本发明描述了与深度学习模型相关的块输入、滤波处理和输出的过程中的考虑因素。

[0105] 以下实施方案可以共同应用于视频编码装置和视频解码装置中利用深度学习技术的部分。

[0106] I. 卷积神经网络(Convolutional Neural Network, CNN)

[0107] CNN是指由多个卷积层和池化层组成的神经网络,并且CNN是已知最适合图像处理的深度学习技术。卷积层通过利用多个核或滤波器来提取特征图。构成滤波器的核系数是在学习过程期间确定的参数。

[0108] 在CNN的卷积层中,靠近输入的前层提取对简单的、低级影像特征(例如,线、点或面)做出响应的特征图,并且靠近输出的后层提取对高级特征(例如,纹理和对象部分)做出响应的特征图。

[0109] 图6是示出根据本发明的至少一个实施方案的卷积层的操作的示意图。

[0110] 卷积层利用卷积运算从输入影像生成特征图。图6的示例描绘了核大小为 3×3 的核(或滤波器)。核大小也称为滤波器大小。核具有核参数或滤波器参数,也称为权重。图6所

示的核具有总共九个核参数。核参数可以最初设置为随机值,并且它们的值可以基于训练来更新。

[0111] 卷积层通过利用与输入影像中的核大小相等的块来执行卷积运算。在这种情况下,相当于输入影像中的核大小的块称为窗口。

[0112] 当以光栅扫描(raster-scan)顺序对输入影像执行滤波时,窗口的移动量称为步长。在图6的示例中,步长为1。如果步长设置为2,则通过将窗口移动2个样本来执行卷积运算,这导致特征图的水平和垂直尺寸是输入影像的水平和垂直尺寸的一半。

[0113] 如上所述,卷积层可以包括多个滤波器。滤波器的数量或核的数量称为通道的数量,即通道的数量等于滤波器的数量。滤波器的数量也确定了特征图的维数。

[0114] 填充是指在执行卷积运算之前,通过在输入数据周围填补某个值来扩展输入数据的方法。填充通常用于控制输出数据的空间大小。用于填充的值可以由超参数确定,但是通常利用零填充。如果不利用填充,输出数据的空间大小会随着每次通过卷积层而减小,这可能会导致边界信息丢失。因此,为了避免这个问题,利用填充,即,可以利用填充来均衡卷积层的输出数据和输入数据的空间大小。

[0115] 图7是示出根据本发明的至少一个实施方案的反卷积层的操作的示意图。

[0116] 反卷积层执行卷积层的逆运算。反卷积层从作为输入的特征图生成期望的数据图像作为输出。反卷积层利用与卷积层相同的术语。当步长为1时,特征图的水平和垂直尺寸与输出数据的水平和垂直尺寸相同,如图7中的示例所示。对于步长2,输出数据的水平和垂直尺寸是特征图的水平和垂直尺寸的两倍。

[0117] 图8是示出根据本发明的至少一个实施方案的池化层的操作的示意图。

[0118] 池化层执行池化,这是对由卷积层生成的特征图进行子采样的过程。池化层利用 2×2 窗口来选择样本,使得输出是输入的宽度的一半和高度的一半。换句话说,通过将 2×2 区域压缩成单个样本,利用池化层来减小输入影像或输入特征图的大小。如图8所示,池化方法包括选择 2×2 区域中的最大值的最大池化,以及生成 2×2 区域的平均值的平均池化等。与卷积层不同,池化层不包括需要学习的变量,并且输入中的通道数量在输出中保持不变。

[0119] 与池化层相反的层被定义为非池化层。与池化层相反,非池化层用于增加维数,并且通常在反卷积层之后利用。

[0120] 卷积编码器-解码器结构是由成对的卷积层和反卷积层组成的网络结构。卷积编码器由卷积层和池化层组成,以从输入影像输出特征图(或特征向量)。卷积编码器的最终输出向量也称为本征向量(latent vector)。卷积解码器由反卷积层和去池化层组成,以从特征图或本征向量生成输出影像。

[0121] 卷积编码器-解码器的输入和输出可以根据应用和网络的目的而变化。例如,输入和输出可以是光流图(optical flow map)、显著性图(saliency map)、影像帧等。

[0122] II. 以块为单位的深度学习技术

[0123] 图9是概念性地示出根据本发明的至少一个实施方案的视频块处理装置的框图。

[0124] 根据本发明的视频块处理装置900(在下文中,称为“块处理装置”)可以通过利用深度学习模型从输入块生成输出块。块处理装置900包括输入单元902、转换单元904和输出单元906的全部或部分。

[0125] 输入单元902获得视频输入块。这里,视频输入块包括三个通道,即Y块、U块和V块。此外,用作视频输入块的块可以是这样的块,所述块的Y块的Y信号、U块的U信号和V块的V信号的采样率格式为4:2:0或4:4:4。因此,深度学习模型的结构也需要设置为符合这些格式。

[0126] 输入单元902通过堆叠或组合视频输入块来生成输入块,使得它们适合于由包括在转换单元904中的深度学习模型进行处理。

[0127] 另一方面,当深度学习模型仅应用于Y信号时,不考虑U和V信号。

[0128] 转换单元904通过在深度学习模型的内核中利用滤波(即,卷积运算)而从输入块生成输出块。这里,深度学习模型可以是如上所述的CNN,并且可以在传统的视频编码框架内执行操作,例如环内滤波、帧内预测、帧间预测、变换和量化。此外,深度学习模型可以被预训练以执行这些任务,并且可以在视频编码装置与视频解码装置之间共享。

[0129] 输出单元906参考应用于输入块的方法,从输出块生成变换的视频块。

[0130] 下面描述输入单元902根据YUV采样率不同地组织深度学习模型的输入块的示例。输入块可以以CU、变换单元(transform unit,TU)、预测单元(prediction unit,PU)或CTU为基础输入,但不一定限于此,并且可以具有由深度学习模型利用的单独大小。

[0131] 首先描述的是从YUV 4:2:0格式的块来构造深度学习模型的输入块的方法。

[0132] 在下面的描述中,Y块的水平尺寸由W表示,并且Y块的竖直尺寸由H表示。在4:2:0格式中,U块和V块的宽度为W/2,并且U块和V块的长度为H/2。此外,在4:4:4格式中,U块和V块的宽度为W,并且高度为H。因此,Y块是大小为W×H的块(在下文中,称为“W×H块”),并且U块和V块是大小为W/2×H/2的块。

[0133] 图10是示出根据本发明的至少一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0134] 输入单元902可以扩大U块和V块以匹配Y块的大小,然后堆叠Y块和扩大的U块和V块以生成W×H×3输入块。在表达式W×H×3中,最后的数字代表输入通道的数量。为了扩大U块和V块,输入单元902可以利用以下方法的一个。由于相同的方法可以应用于U块和V块两者,下面仅描述用于扩大U块的方法。

[0135] 图11是示出根据本发明的至少一个实施方案的当构造输入块时扩大U块的方法的示意图。

[0136] 输入单元902可以通过重复相同的U块4次来生成W×H块。在重复中,输入单元902可以通过上下或左右镜像U块来组合U块,以减少重复的块的边界之间的变化。

[0137] 输入单元902可以将U块定位在W×H块的中心。在这种情况下,输入单元902可以利用来自U块的一些信号来填充U块周围。替选地,输入单元902可以用同位的Y块的值填补U块的外周。替选地,输入单元902可以用来自先前编码的块的值来填充U块。例如,在帧间预测的情况下,输入单元902可以用来自先前帧的同位块的值来填补U块周围。替选地,输入单元902可以利用任意像素值来填补U块的外周。

[0138] 输入单元902可以将U块定位在W×H块的一个象限中,包括W×H块的左上角。此时,输入单元902可以利用来自U块的一些信号来填充W×H块的剩余部分。替选地,输入单元902可以利用同位的Y块的值来填补W×H块的剩余部分。替选地,输入单元902可以利用来自先前编码的块的值来填补W×H块的剩余部分。例如,在帧间预测的情况下,输入单元902可以利用来自先前帧的同位块的值来填补W×H块的剩余部分。

[0139] 输入单元902可以通过对U块进行上采样来生成 $W \times H$ 块。输入单元902可以通过利用用于上采样的单独的滤波器或者通过重复相同的像素两次来对U块进行上采样。

[0140] 图12是示出根据本发明的另一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0141] 在根据本发明的另一个实施方案中,如图12所示,输入单元902可以对Y块的大小进行四等分,以匹配U块和V块的大小。在这种情况下,输入单元902可以将Y块的四分块与U块和V块堆叠,以生成 $W/2 \times H/2 \times 6$ 输入块。

[0142] 输入单元902可以将Y块划分到四个象限中以四等分Y块。

[0143] 替代地,如图13所示,输入单元902可以通过在水平和垂直方向上抽取(decimation)构成Y块的样本来四等分Y块。

[0144] 图14是示出根据本发明的另一个实施方案的构造深度学习模型的输入块的方法的示意图。

[0145] 在根据本发明的另一个实施方案中,如图14所示,输入单元902可以通过利用U块和V块构造超级块来匹配Y块的大小,从而生成 $W \times H \times 2$ 的输入块。此时,输入单元902可以利用以下方法的一个。

[0146] 图15是示出根据本发明的另一个实施方案的当构造输入块时构造超级块的方法的示意图。

[0147] 输入单元902通过在水平或垂直方向上重复U、V块两次来构造 $W \times H$ 超级块。为了重复相同的块但减少边界之间的变化,输入单元902可以通过上下或左右镜像相同色度分量的块来组合相同色度分量的块。

[0148] 输入单元902可以将U块和V块定位在 $W \times H$ 块的每个象限中,包括 $W \times H$ 块的左上角。此时,输入单元902可以利用来自U、V块的一些信号来填充 $W \times H$ 块的剩余部分。替代地,输入单元902可以利用同位的Y块的值来填补 $W \times H$ 块的剩余部分。替代地,输入单元902可以利用先前编码的块的值来填补 $W \times H$ 块的剩余部分。例如,在帧间预测的情况下,输入单元902可以利用来自先前帧的同位块的值来填补 $W \times H$ 块的剩余部分。

[0149] 输入单元902可以上下组合U、V块并且在水平方向上对U、V块进行上采样,或者左右组合U、V块并且在垂直方向上对U、V块进行上采样,以生成 $W \times H$ 块。此时,输入单元902可以利用单独的滤波器进行上采样,或者可以通过重复相同的像素两次来对U、V块进行上采样。

[0150] 在根据本发明的其他实施方案中,输入单元902可以利用Y、U和V块以形成单个超级块。如图16所示,输入单元902可以排列Y、U和V块。例如,输入单元902可以将U块和V块放置在Y块的顶部、底部、左侧或右侧。

[0151] 在根据本发明的其他实施方案中,超级块的大小可以是2的幂或4的倍数。如图17所示的输入单元902可以通过将U块和V块的每个相加来构造超级块。为了构造超级块,可以重复相同的U块和相同的V块,但是为了减少边界之间的变化,输入单元902可以通过上下或从左到右镜像相同色度分量的块来组合相同色度分量的块。替代地,输入单元902可以对U块和V块进行上采样。

[0152] 下面描述了从YUV 4:4:4格式的块构造深度学习模型的输入块的方法。

[0153] 在4:4:4格式中,U、V块的大小与Y块的大小相同。因此,如图10所示,输入单元902

可以通过将U、V块与Y块堆叠来生成 $W \times H \times 3$ 输入块。替选地,输入单元902可以组合Y、U和V块以生成大小为 $3W \times H$ 的超级块。

[0154] 以上描述涉及当转换单元904包括单个深度学习模型时生成输入块的方法。下面描述了当转换单元904包括两个或更多个深度学习模型时生成输入块的方法。

[0155] 转换单元904可以通过利用三个不同的深度学习模型来处理Y、U和V块。此时,输入单元902可以通过利用以下方法的一个来生成输入块。

[0156] 输入单元902可以从Y、U和V块的每个生成 $W \times H \times 1$ 输入块,以将生成的 $W \times H \times 1$ 输入块输入到三个深度学习模型。例如,输入单元902可以为每个输入通道分配一个深度学习模型。在这种情况下,如图11所示,输入单元902可以扩大U块和V块的每个,以生成 $W \times H \times 1$ 输入块。对于相应的Y、U和V块,输入单元902可以将 $W \times H \times 1$ 输入块堆叠三次,以将堆叠变换成 $W \times H \times 3$ 输入块,然后可以将其输入到深度学习模型。

[0157] 输入单元902可以从Y块生成 $W \times H \times 1$ 输入块,并且从U块和V块的每个生成 $W/2 \times H/2 \times 1$ 输入块。例如,由用于处理Y块的深度学习模型处理的输入块在大小上不同于由用于处理U块和V块的每个的深度学习模型处理的输入块。在这种情况下,对于Y块,输入单元902可以将 $W \times H \times 1$ 输入块堆叠三次,以将堆叠变换成 $W \times H \times 3$ 输入块,然后可以将其输入到用于处理Y块的深度学习模型。对于U块和V块的每个,输入单元902可以将 $W/2 \times H/2 \times 1$ 输入块堆叠三次,以将堆叠变换成 $W/2 \times H/2 \times 3$ 输入块。

[0158] 输入单元902可以从Y、U和V块的每个生成 $W \times H \times 3$ 输入块,以输入到三个深度学习模型。在这种情况下,输入单元902可以通过将相同的色度分量堆叠三次来生成一个神经网络的输入块。以这种方式,输入单元902可以利用设置为适合三个输入通道的深度学习模型。

[0159] 在根据本发明的另一个实施方案中,转换单元904可以利用两个不同的深度学习模型来处理Y、U和V块。转换单元904可以利用用于处理Y块的一个深度学习模型(在下文中,称为“Y块深度学习模型”),以及用于处理U、V块的另一个深度学习模型(在下文中,称为“UV块深度学习模型”)。此时,输入单元902可以通过利用以下方法的一个来生成输入块。

[0160] 输入单元902可以从Y块生成 $W \times H \times 1$ 输入块并将其输入到Y块深度学习模型。此外,输入单元902可以堆叠U块和V块以生成 $W/2 \times H/2 \times 2$ 输入块,该输入块然后可以输入到UV块深度学习模型中。

[0161] 在另一个实施方案中,输入单元902可以从Y块生成 $W \times H \times 1$ 输入块并将其输入到Y块深度学习模型中。此外,输入单元902可以堆叠U块和V块以生成 $W \times H \times 2$ 输入块,该输入块然后可以输入到UV块深度学习模型中。在这种情况下,如图11所示,输入单元902可以扩大U块和V块的每个,以生成 $W \times H \times 2$ 输入块。

[0162] 在又一个实施方案中,输入单元902可以从Y块生成 $W \times H \times 1$ 输入块并将其输入到Y块深度学习模型中。此外,输入单元902可以从U块和V块生成超级块并将其输入到UV块深度学习模型。在这种情况下,如图15所示,输入单元902可以通过利用U、V块来生成 $W \times H \times 1$ 输入块。

[0163] 下面描述转换单元904通过利用深度学习模型来执行卷积运算(即,滤波)的过程。为了执行卷积运算,转换单元904可以在输入块周围填充样本,设置步长值,并且根据预设的核大小通过利用深度学习模型中的核来对输入块进行滤波。考虑到以下内容,转换单元

904根据输入视频块执行卷积运算。此外,如上所述,转换单元904可以将执行卷积运算的方法与生成一个或多个输入块的方法结合。

[0164] 与涉及通过利用零填充进行填充、复制周边样本或镜像周边样本的传统卷积运算相比,根据本实施方案的转换单元904可以根据输入块的预测模式通过考虑以下方法中的一个或多个来填充输入块。

[0165] 首先要描述的是当输入块是帧内预测块时的填充。

[0166] 转换单元904可以通过利用来自与输入块邻近的周边样本的已经编码的样本来填充输入块。

[0167] 转换单元904可以利用与由常规帧内预测使用的一样多的行作为填充样本。例如,对于HEVC,转换单元904可以利用来自块的顶部和左侧的一行的样本,并且对于VVC,转换单元904可以利用来自块顶部和左侧的三行(第一、第二和第四)的样本。

[0168] 转换单元904可以在对由帧内预测利用的边界样本执行滤波之后填充样本。

[0169] 转换单元904可以填充样本,而不对由帧内预测利用的边界样本执行任何滤波。

[0170] 转换单元904可以不在CTU边界、切片边界和虚拟处理单元边界处填充相邻样本。

[0171] 转换单元904可以通过利用相邻样本来填充块的顶部和左侧,并且可以通过复制或镜像当前块的值来填充块的底部和右侧。

[0172] 下面描述当输入块是帧间预测块时的填充。

[0173] 转换单元904可以通过利用来自与输入块邻近的周边样本的已经编码的样本来填充输入块。

[0174] 转换单元904可以利用来自已经被编码的先前帧中的同位块的样本来填充输入块。

[0175] 转换单元904可以利用通过利用来自先前编码的帧的运动向量获得的参考预测块的样本来填充输入块。

[0176] 转换单元904可以在应用由帧间预测利用的插值滤波器之后填充样本。

[0177] 转换单元904可以在不应用由帧间预测利用的插值滤波器的情况下填充样本。

[0178] 除了块的顶部和左侧之外,转换单元904可以填充块的底部和右侧。

[0179] 对于360度的视频,为了填充帧的最右侧的块,转换单元904可以利用相同行中的最左侧的块。另外,为了填充最下方的块,转换单元904可以利用相同列中的最上方的块。

[0180] 如果填充不可行,可以利用0作为填充值,但不限于此。例如,转换单元904可以利用预定值作为填充值。例如, $2^{\text{Bit}-1}$ 可以用作填充值。这里,Bit代表表示视频样本所需的位的数量。

[0181] 另一方面,如果一个或多个块构成超级块,则转换单元904可以填充共有块之间的边界值。

[0182] 转换单元904可以利用相同的步长和相同的核大小,以通过利用深度学习模型来对输入块进行滤波。例如,转换单元904将步长设置为1或2,并且将核设置为3、5或7。然而,不限于此,转换单元904可以如下地基于块的大小和色度分量来设置步长值和核大小。

[0183] 转换单元904可以根据块的大小不同地设置步长值。例如,对于顶部或左侧大小为16或更大的块,转换单元904可以将步长设置为2,并且对于小于16的块,转换单元904可以将步长设置为1。

[0184] 转换单元904可以基于块的大小不同地设置核大小。例如,对于顶部或左侧大小为16或更大的块,转换单元904可以将核大小设置为5,并且对于小于16的块,转换单元904可以将核大小设置为3。

[0185] 转换单元904可以根据块的色度分量不同地设置步长值。例如,转换单元904可以将Y块的步长设置为2,并且将U块和V块的步长设置为1。相反,转换单元904可以将Y块的步长设置为1,并且将U块、V块的步长设置为2。

[0186] 转换单元904可以根据块的色度分量不同地设置核大小。例如,转换单元904可以将Y块的核大小设置为5,并且将U、V块的核大小设置为3。

[0187] 在围绕输入块填充样本并设置步长值之后,转换单元904可以通过向输入块应用滤波器系数和非线性激活函数来生成输出块。

[0188] 另一方面,如上所述,输出单元906可以通过将应用于输入块的组合方法应用于输出块来生成Y、U和V块的每个。

[0189] 在根据本发明的其他实施方案中,除了Y、U和V块的像素值之外,块处理单元900可以利用与Y、U、V块的编码一起输出的额外信息作为深度学习模型的输入,Y、U、V块的编码如块分区结构、量化参数、预测模式、参考块的时间信息(例如,计数图像(a picture of count,POC))等。

[0190] 例如,当利用块分区结构作为额外信息时,块处理单元900可以通过堆叠除上述Y、U和V块的输入方法之外的块分区结构来使得输入通道的维数的进一步增加。块处理单元900可以提供YUV输入块和块分区结构作为深度学习模型的输入,如图18所示。替选地,块处理单元900可以封装YUV输入块和块分区结构以生成超级块,该超级块然后可以作为输入提供给深度学习模型。

[0191] 在另一个实施方案中,块处理器900可以将额外信息输入到深度学习模型的另一个分支中。在图19的示例中,块处理单元900将块分区结构输入到深度学习模型的另一个分支中。

[0192] 在如图20所示的另一个实施方案中,块处理单元900将编码的图输入到深度学习模型的另一个分支中。在这种情况下,块处理单元900可以利用以下信息作为编码的图。

[0193] 块处理单元900可以利用量化参数作为编码的图。例如,块处理装置900可以将量化参数从最小值归一化到最大值以供使用,然后缩放它们以用作编码的图。

[0194] 块处理器900可以利用POC作为编码的图。块处理单元900可以利用当前块与参考块之间的POC差异作为编码的图。

[0195] 块处理单元900可以利用帧内预测模式作为编码的图。块处理单元900可以利用指示帧内预测模式的一个的值作为编码的图。

[0196] 块处理单元900可以利用帧间预测模式作为编码的图。块处理单元900可以利用指示帧间预测模式的一个的值作为编码的图。

[0197] 在根据本发明的其他实施方案中,块处理单元900可以将YUV输入块与编码的图堆叠,类似于图18所示的方式,并且将堆叠作为深度学习模型的输入。

[0198] 图21是根据本发明的至少一个实施方案的处理视频块的方法的流程图。

[0199] 块处理单元900获得视频输入块(S2100)。这里,视频输入块包括三个通道,即Y块、U块和V块。此外,用作视频输入块的块可以是包括Y块、U块和V块的块,所述Y块、U块和V块分

别具有采样率格式为4:2:0或4:4:4的Y信号、U信号和V信号。

[0200] 块处理单元900堆叠或组合Y块、U块和V块以生成输入块(S2102)。

[0201] 块处理单元900可以根据YUV采样率以不同的方式构造深度学习模型的输入块。对于块处理单元900如何从YUV 4:2:0格式和YUV 4:4:4格式的块构造深度学习模型的输入块,可以查阅图10到图17的示例,因此没有提供进一步的详细描述。

[0202] 块处理单元900将输入块输入到至少一个深度学习模型中(S2104)。这里,深度学习模型可以在现有的视频编码框架内执行运算,例如环内滤波、帧内预测、帧间预测、变换、量化等。

[0203] 块处理单元900基于至少一个深度学习模型来执行卷积运算,以从输入块生成输出块(S2106)。为了执行卷积运算,块处理单元900可以在输入块周围填充样本,设置步长值,并且根据预设的核大小通过利用深度学习模型中的核来对输入块进行滤波。

[0204] 块处理单元900从输出块生成视频输出块(S2108)。块处理单元900可以通过参考应用于输入块的方法从输出块生成转换后的视频块。

[0205] 尽管描述了顺序执行的各个流程图中的步骤,但这些步骤仅仅例示了本发明的一些实施方案的技术思想。因此,本发明所属领域的普通技术人员可以通过改变在各个附图中描述的顺序或通过并行地执行两个或更多个步骤来执行步骤。因此,各个流程图中的步骤不限于所示的按发生时间排列的顺序。

[0206] 应当理解,上述描述呈现了可以以各种其他方式实现的说明性实施方案。在一些实施方案中描述的功能可以通过硬件、软件、固件和/或它们的组合来实现。还应当理解,本说明书中描述的功能组件标记为“.....单元”,以突出强调它们独立实现的可能性。

[0207] 另一方面,在一些实施方案中描述的各种方法或功能可以实现为存储在非易失性记录介质中的指令,所述指令可以由一个或更多个处理器读取和执行。非易失性记录介质可以包括例如以计算机系统可读取的形式存储数据的各种类型的记录装置。例如,非易失性记录介质可以包括存储介质,例如可擦除可编程只读存储器(EPROM)、闪存驱动器、光盘驱动器、磁性硬盘驱动器和固态驱动器(SSD)等等。

[0208] 尽管出于说明的目的描述了本发明的示例性实施方案,但是本发明所属领域的普通技术人员应当理解,在不脱离本发明的思想和范围的情况下,可以进行各种修改、添加和替换。因此,出于简洁和清楚起见描述了本发明的实施方案。本发明的实施方案的技术思想的范围不受例示的限制。相应地,本发明所属领域的普通技术人员应当理解的是,本发明的范围不受上述明确描述的实施方案的限制,而是受权利要求及其等同形式的限制。

[0209] (附图标记说明)

[0210] 900: 视频块处理装置

[0211] 902: 输入单元

[0212] 904: 转换单元

[0213] 906: 输出单元。

[0214] 相关申请的交叉引用

[0215] 本申请要求于2020年12月2日提交的韩国专利申请No.10-2020-0166450以及于2021年12月2日提交的韩国专利申请No.10-2021-0170547的优先权,其全部内容通过引用结合于本文中。

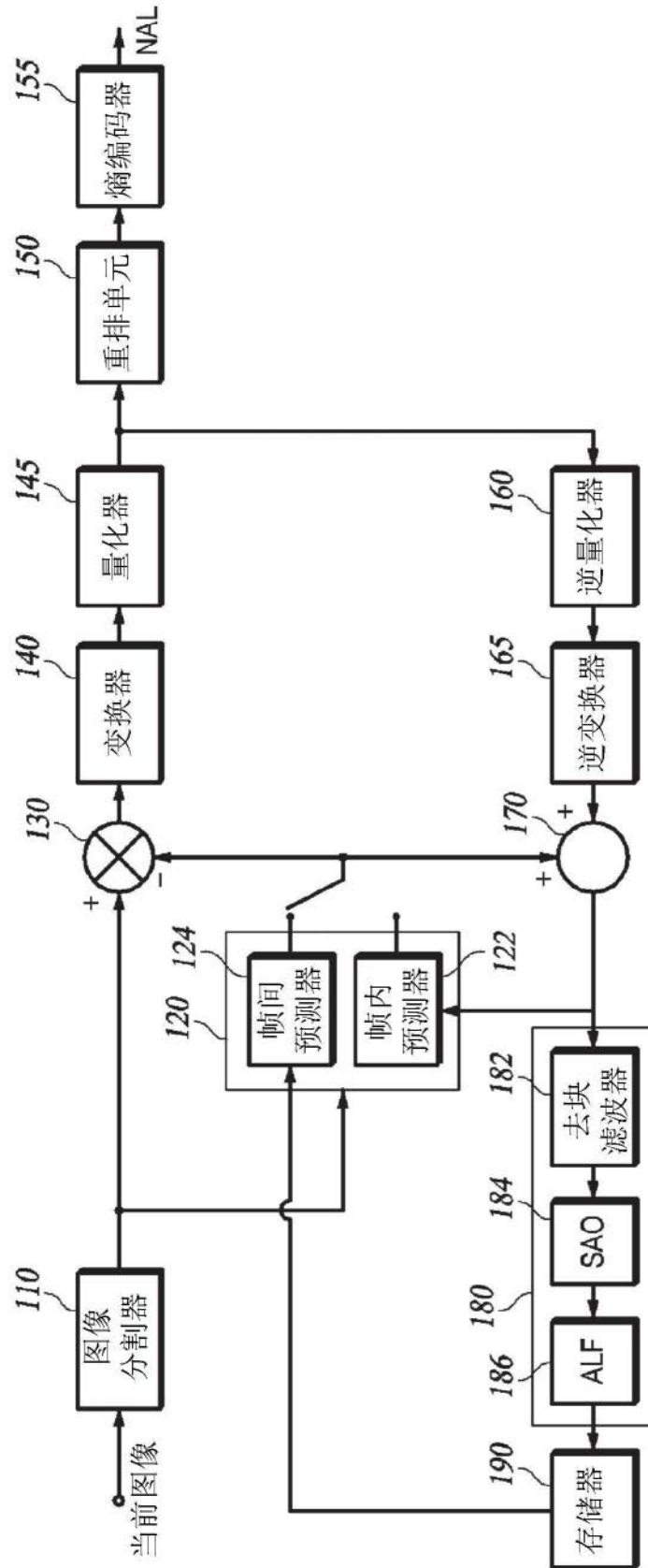


图1

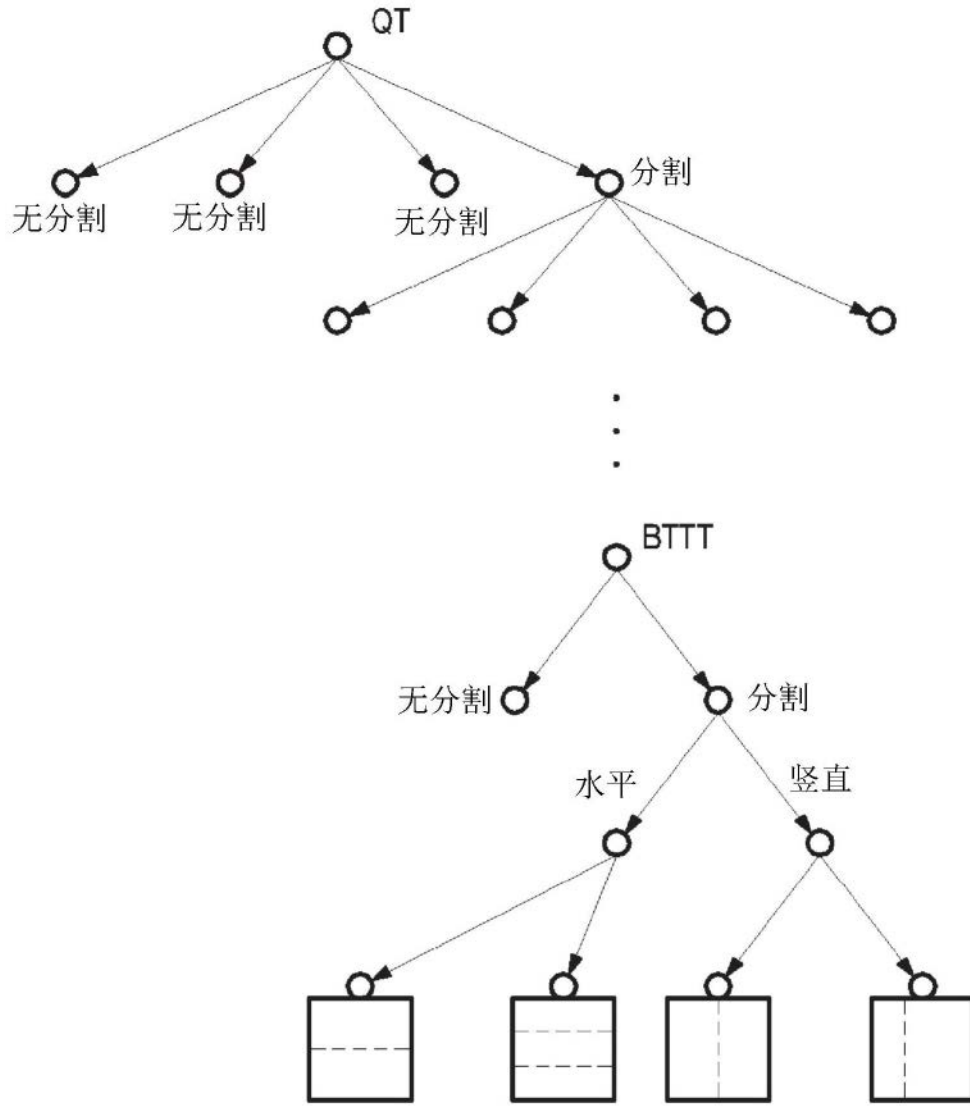


图2

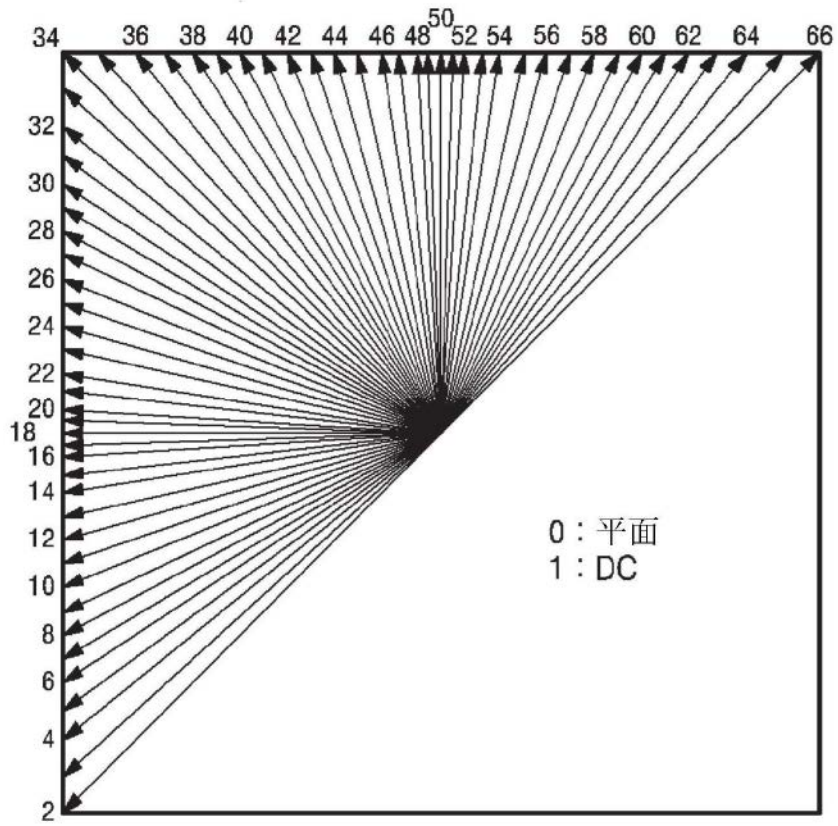


图3a

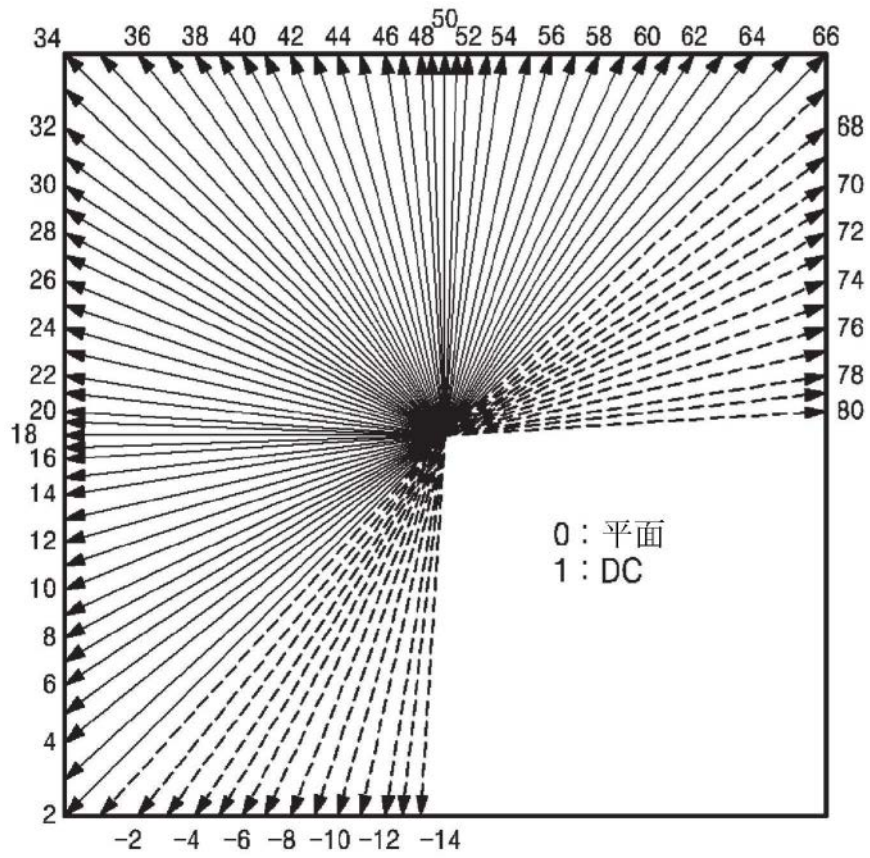


图3b

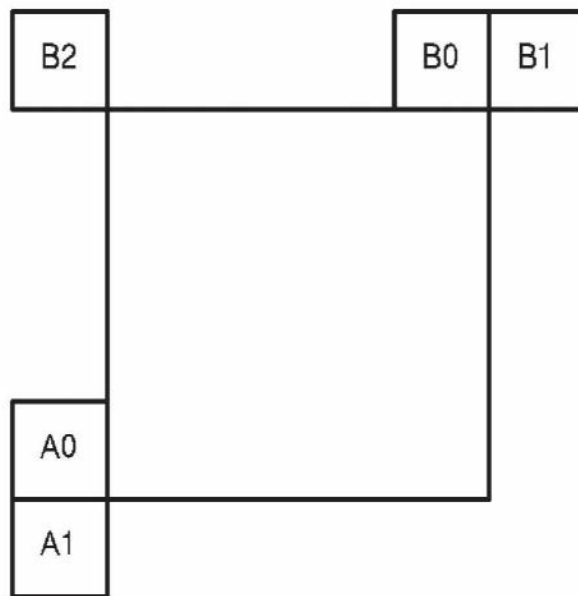


图4

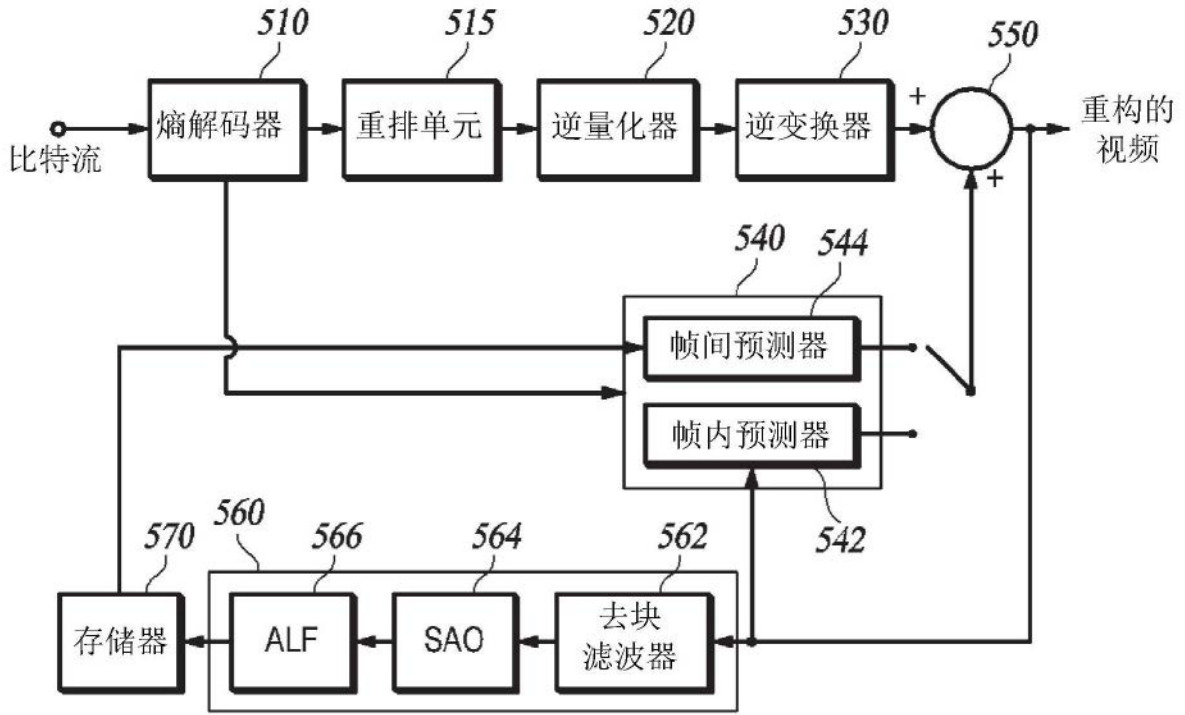


图5

卷积层

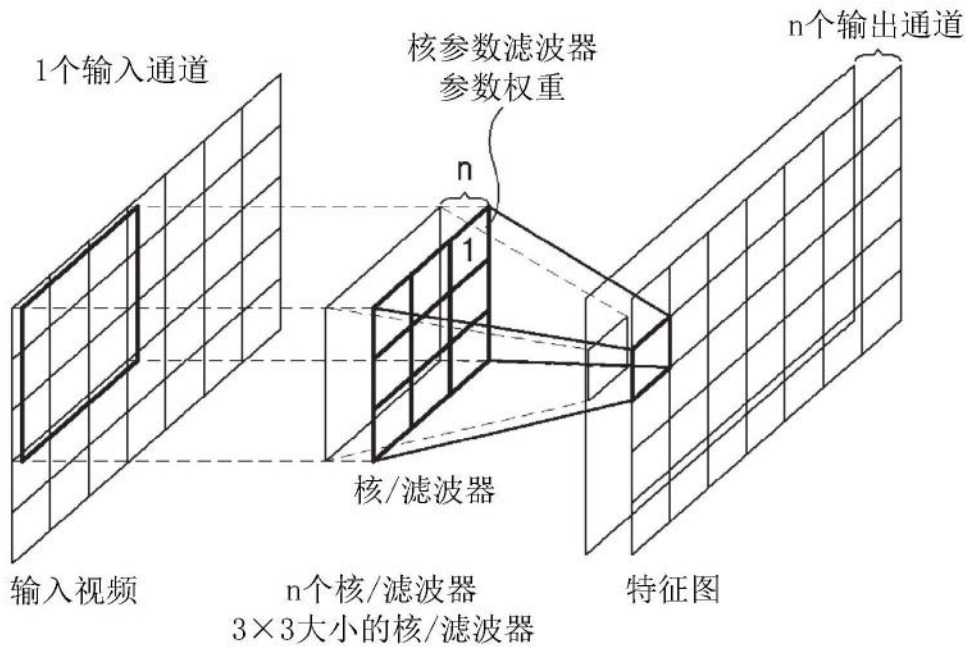


图6

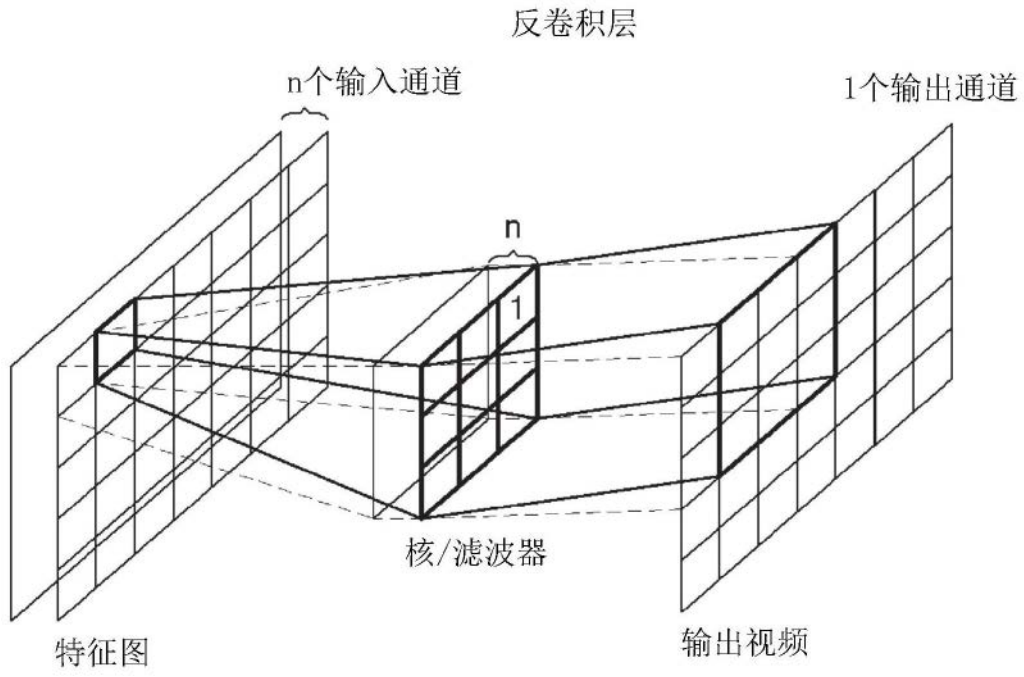


图7

池化层

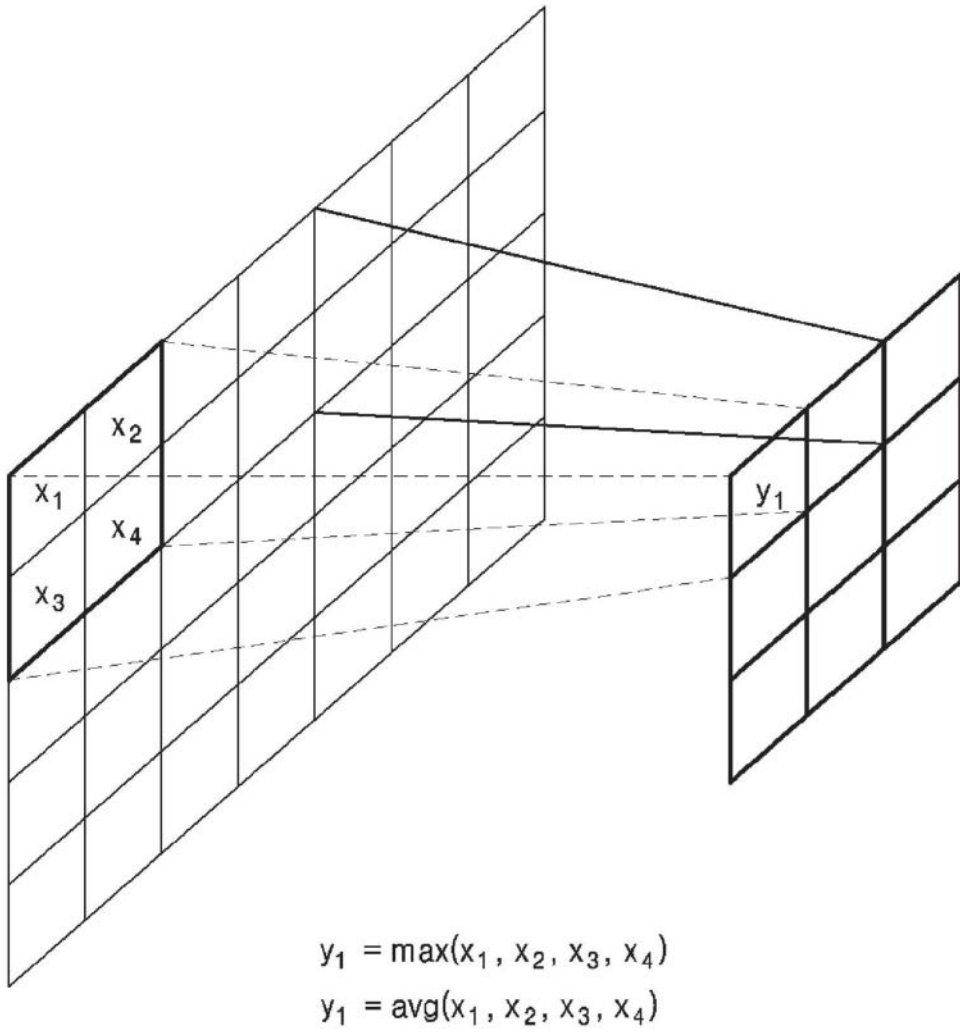


图8

900

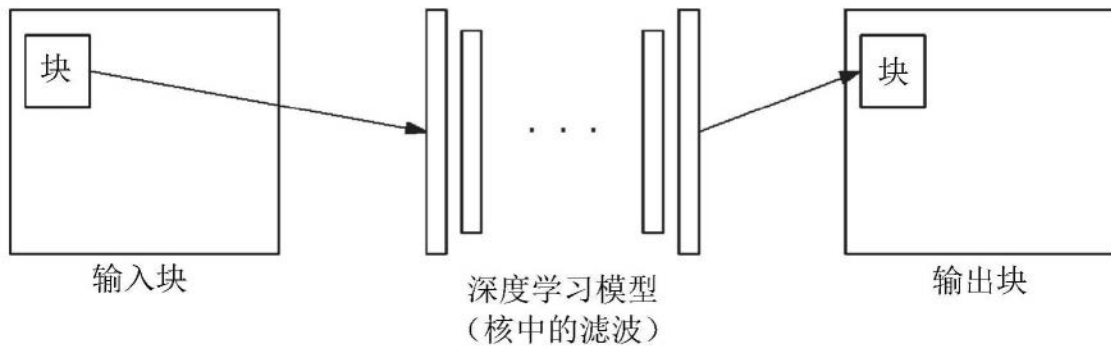
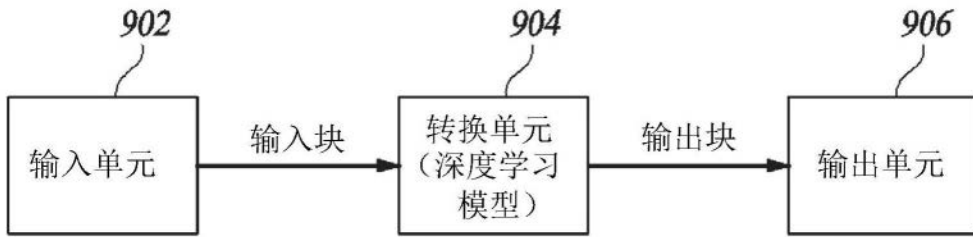


图9

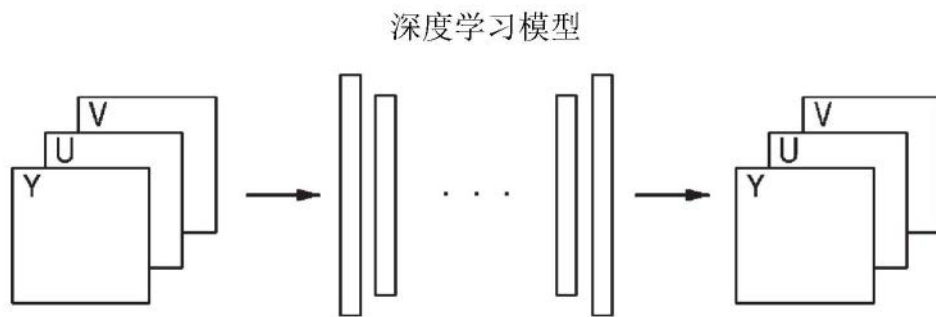


图10

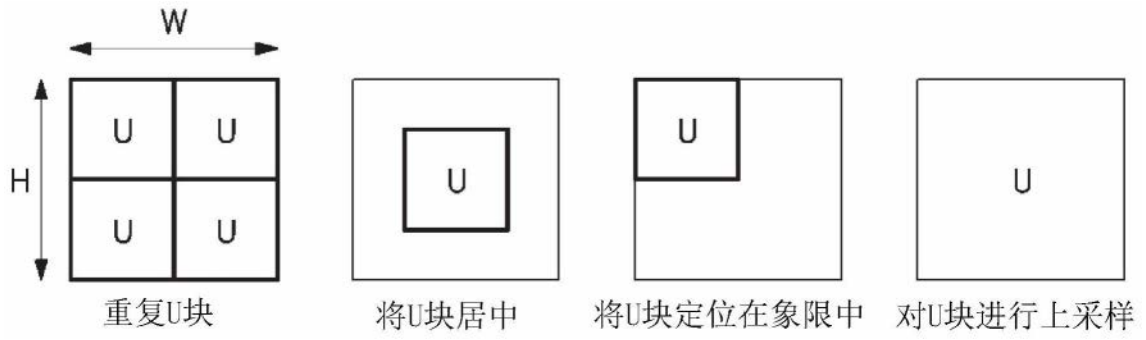


图11

深度学习模型

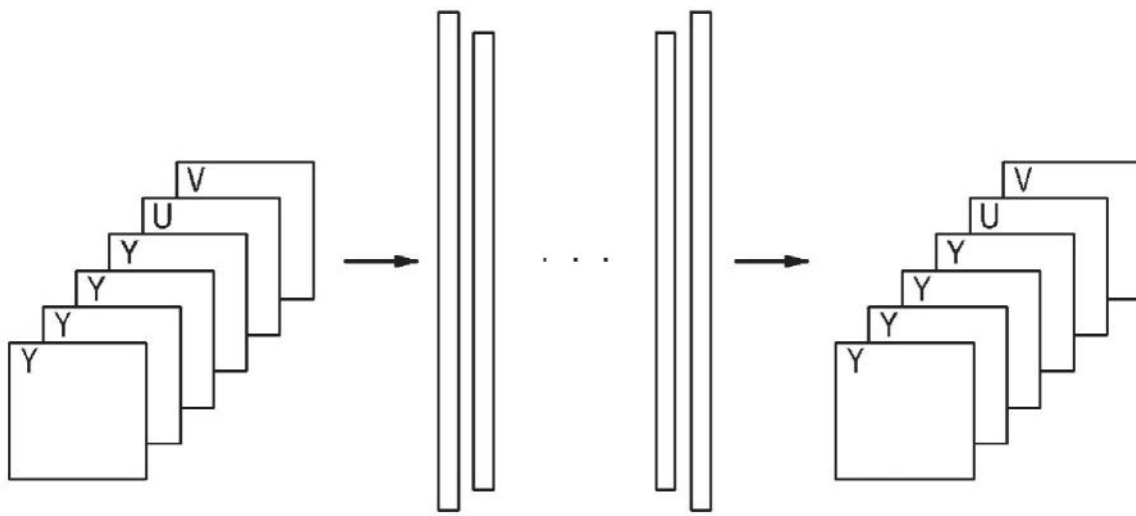


图12

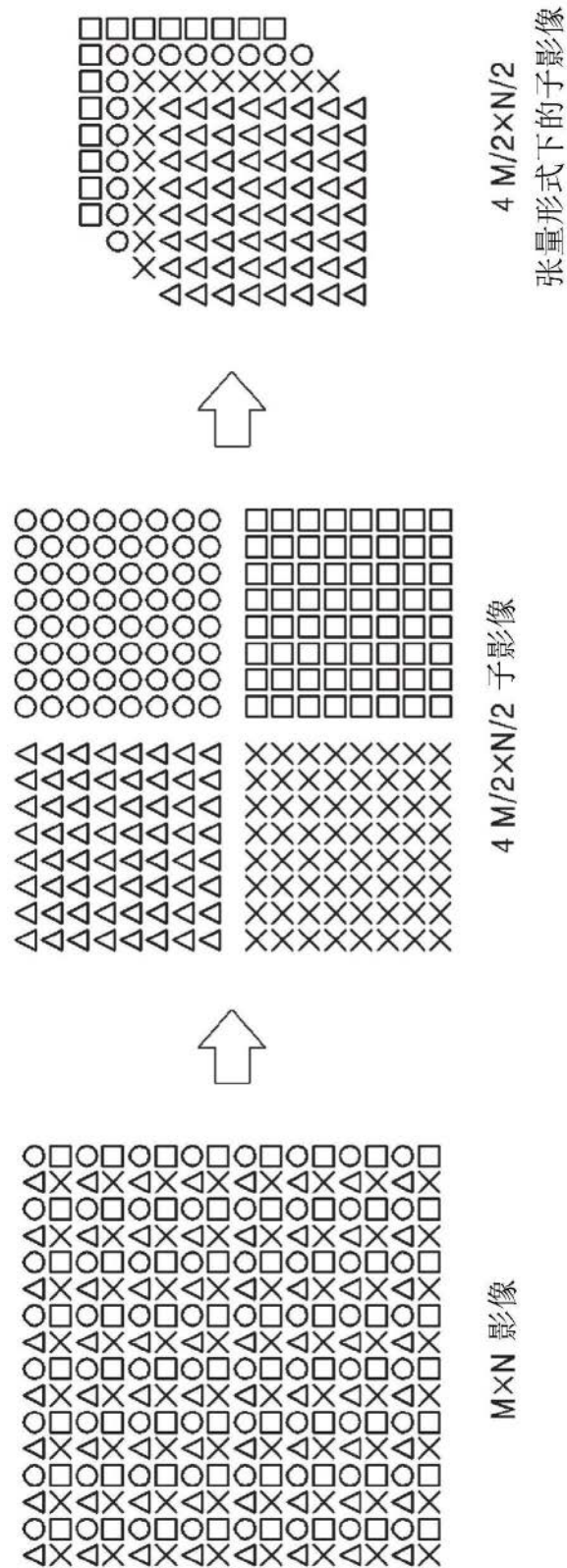


图13

深度学习模型

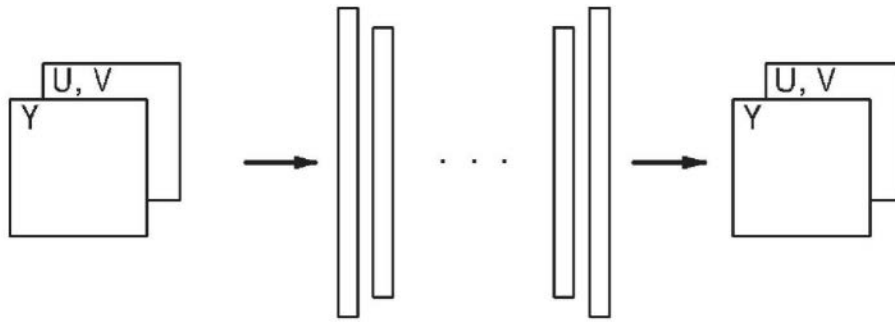


图14



图15

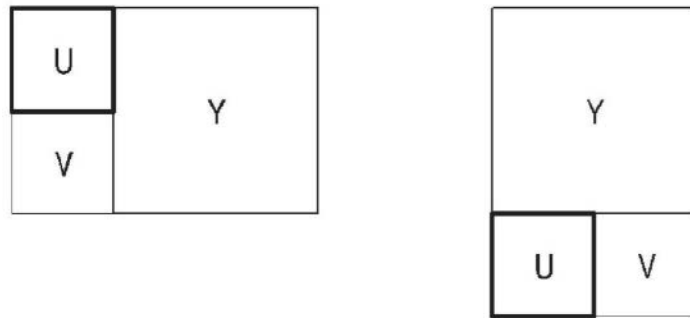
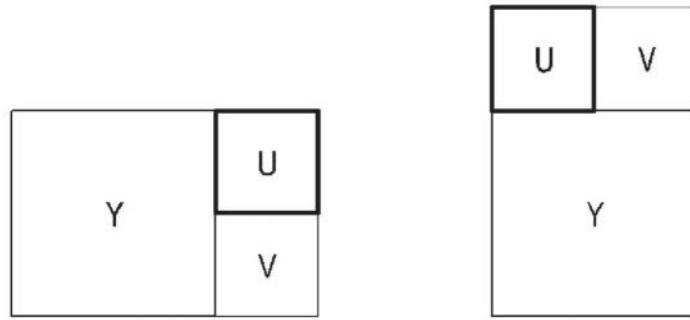


图16

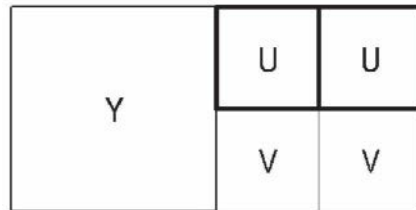


图17

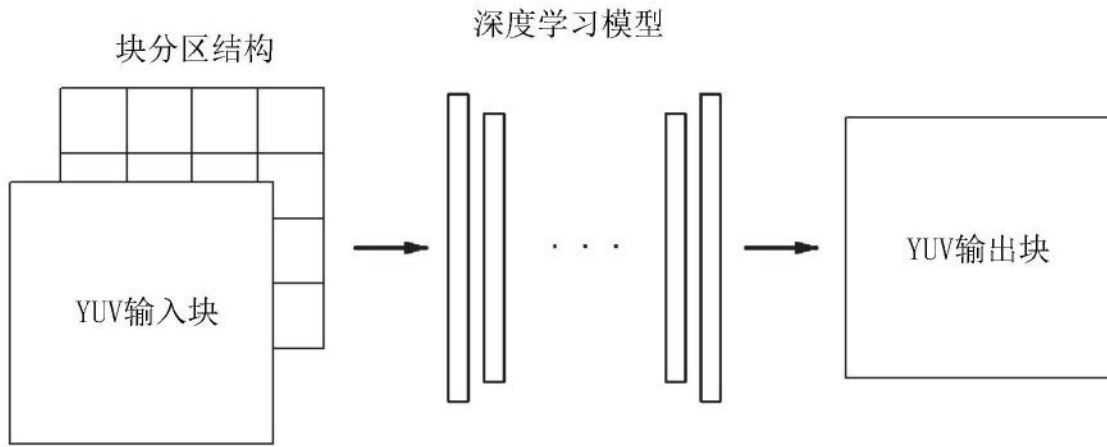


图18

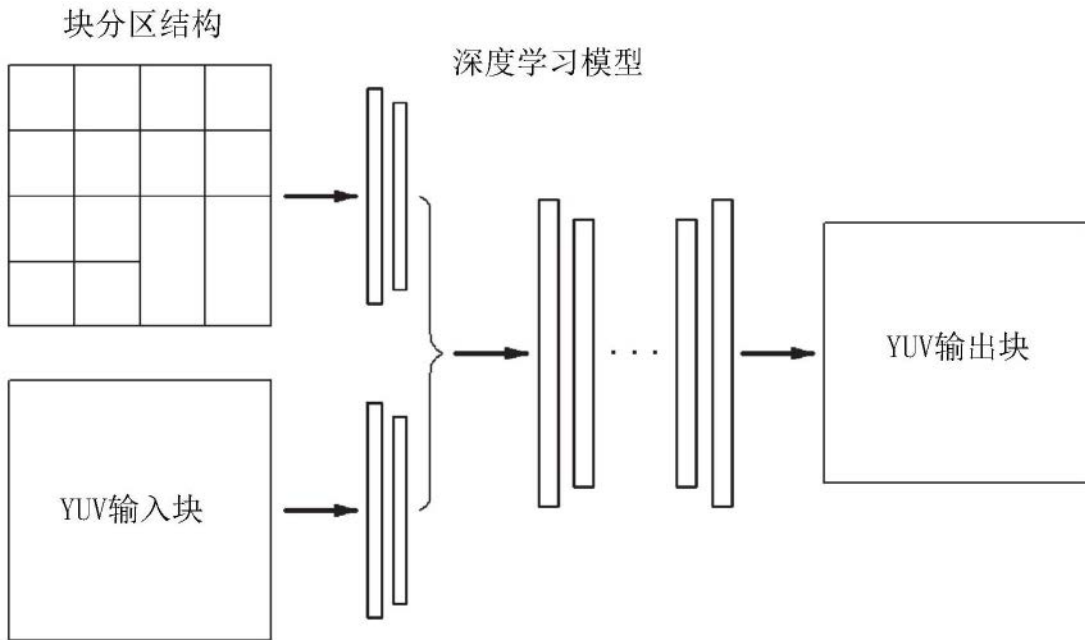


图19

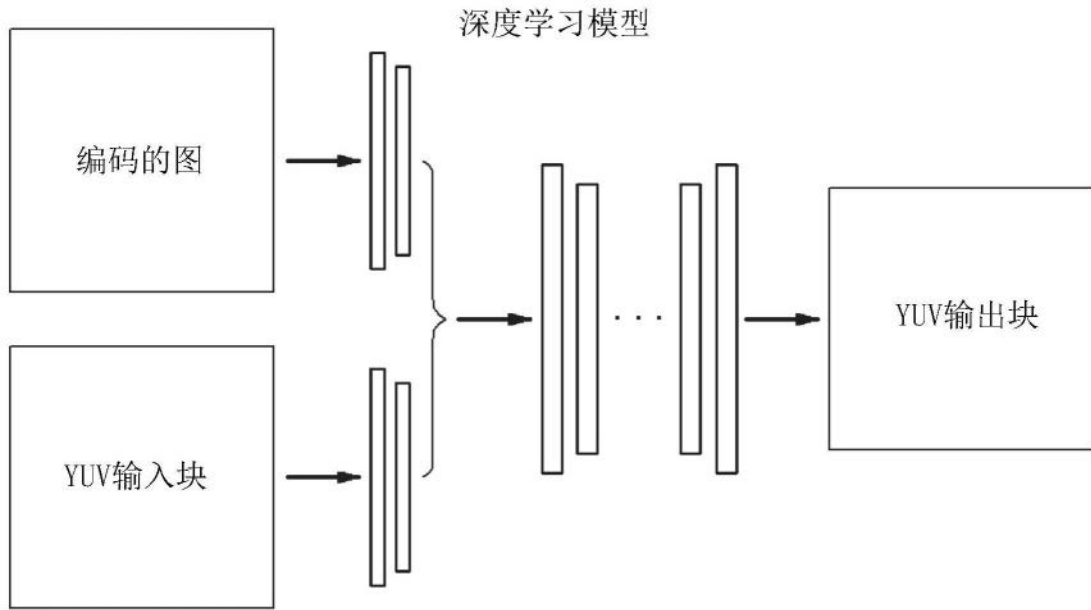


图20

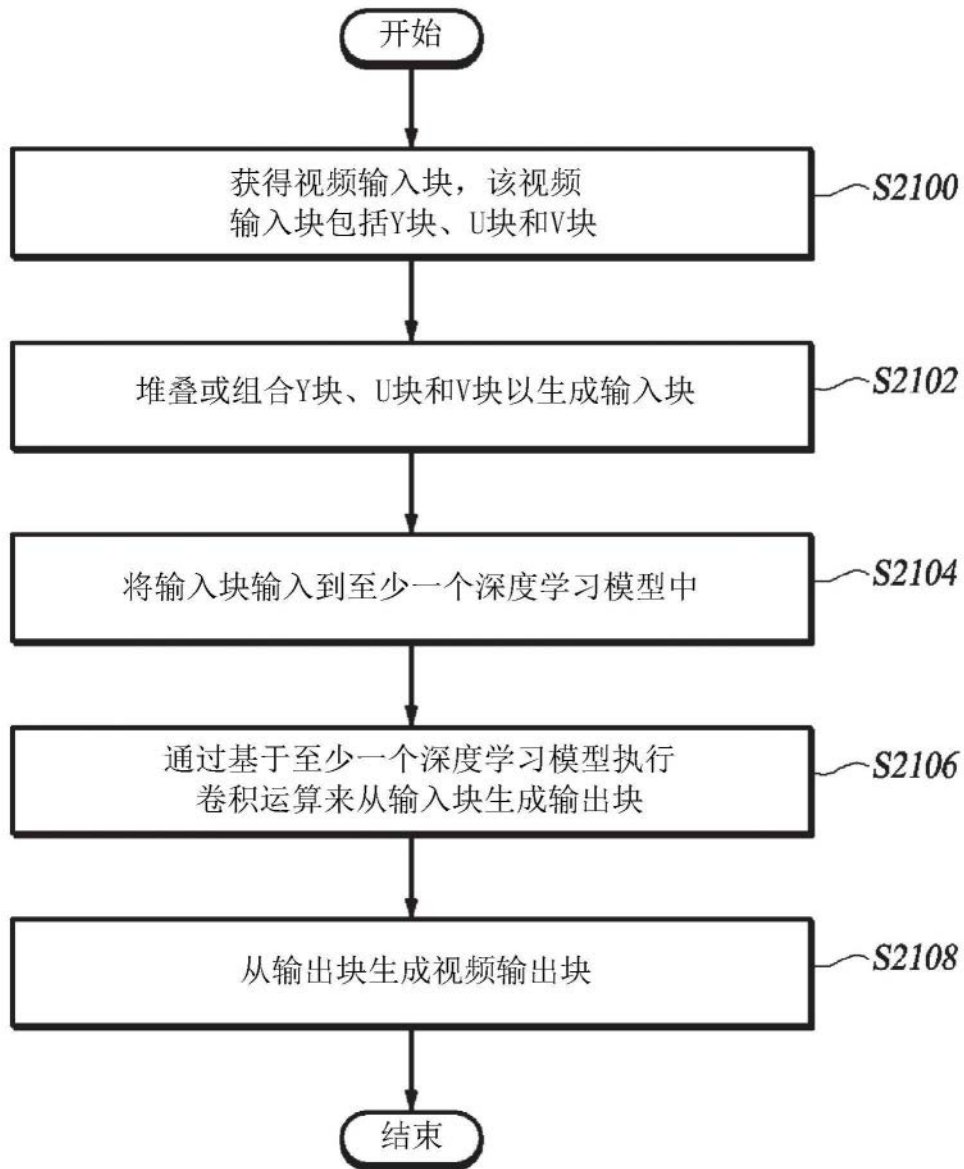


图21