(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117837149 A
(43)申请公布日 2024.04.05

(21)申请号 202280055257.X
(22)申请日 2022.07.20
(30)优先权数据
10-2021-0108041 2021.08.17 KR
10-2022-0089020 2022.07.19 KR
(85)PCT国际申请进入国家阶段日
2024.02.06
(86)PCT国际申请的申请数据
PCT/KR2022/010603 2022.07.20
(87)PCT国际申请的公布数据
WO2023/022376 KO 2023.02.23
(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
(72)发明人 姜制远 李订炅 朴胜煜 许镇
(74)专利代理机构 北京康信知识产权代理有限责任公司 11240
专利代理师 陈鹏
(51)Int.Cl.
H04N 19/82(2014.01)
H04N 19/597(2014.01)
H04N 19/117(2014.01)
H04N 19/423(2014.01)
H04N 19/70(2014.01)
H04N 19/86(2014.01)
G06T 3/40(2024.01)
G06T 9/00(2006.01)

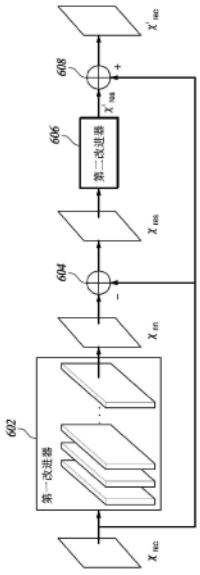
权利要求书2页 说明书15页 附图9页

(54)发明名称

使用改进的环内滤波器的视频编码方法和设备

(57)摘要

公开了一种使用改进的环内滤波器的视频编码方法和设备,并且在实施方式中提供了一种视频编码方法和设备,其使用深度学习模型从重构帧生成残差帧以改进视频编码效率并改进视频质量,并将生成的残差帧应用于线性模型以近似原始残差帧,从而改进环内滤波器的性能。



CN 117837149 A

1.一种由计算装置执行的用于改进重构帧的视频质量的方法,所述方法包括:

获得所述重构帧,并将所述重构帧输入到基于深度学习的改进模型中以生成近似原始帧的输出,其中,所述重构帧是通过重构所述原始帧而获得的帧并且所述重构帧由所述计算装置预先重构;

通过从所述基于深度学习的改进模型的输出中减去所述重构帧来生成第一残差帧;

将所述第一残差帧输入到线性模型以生成第二残差帧,其中,所述线性模型包括表示所述第一残差帧与所述第二残差帧之间的线性关系的参数;以及

通过将所述第二残差帧与所述重构帧相加来生成改进的重构帧。

2.根据权利要求1所述的方法,其中,所述第一残差帧近似于原始残差帧,并且

其中,所述原始残差帧是所述原始帧与所述重构帧之间的差。

3.根据权利要求2所述的方法,其中,所述基于深度学习的改进模型是包括多个层的深度学习模型,并且所述基于深度学习的改进模型使用基于所述第一残差帧与所述原始残差帧之间的差的损失函数来训练。

4.根据权利要求1所述的方法,其中,所述重构帧是存储在经解码图片缓冲区(DPB)中的重构信号、去块滤波器的输出、样点自适应偏移(SAO)滤波器的输出、以及自适应环路滤波器(ALF)的输出中的一者,并且

其中,所述重构信号是经预测的信号和经逆变换的信号之和。

5.根据权利要求1所述的方法,进一步包括:

用信号发送指示是否应用改进视频质量的所述方法的标记。

6.根据权利要求2所述的方法,进一步包括:

基于所述第一残差帧中的像素值和所述原始残差帧中的像素值,通过使用线性最小二乘方程来估计所述线性模型的参数。

7.根据权利要求1所述的方法,进一步包括:

通过视频编码设备导出所述线性模型的所述参数;

将所述线性模型的所述参数编码成比特流;以及

将所述线性模型的所述参数传输到视频解码设备。

8.根据权利要求1所述的方法,进一步包括:

将预设值用作所述线性模型的参数,其中,所述预设值是基于编码和解码信息的统计上频繁出现的值。

9.根据权利要求1所述的方法,其中,所述线性模型的参数包括预先设置的所述线性模型的参数值和对应的索引,

其中,视频编码设备将索引用信号发送至视频解码设备,并且其中,所述视频解码设备使用所述索引来获得预先设置的所述参数值。

10.一种包括在视频解码设备中的视频质量改进装置,所述视频质量改进装置包括:

第一改进器,包括基于深度学习的改进模型并且被配置为获得重构帧并将所述重构帧输入到所述基于深度学习的改进模型中以生成近似原始帧的输出,其中,所述重构帧是通过重构所述原始帧而获得的帧并且所述重构帧由所述视频解码设备预先重构;

减法器,被配置为从所述基于深度学习的改进模型的输出中减去所述重构帧以生成第一残差帧;

第二改进器,包括线性模型并且被配置为将所述第一残差帧输入到所述线性模型以生成第二残差帧,其中,所述线性模型包括表示所述第一残差帧与所述第二残差帧之间的线性关系的参数;以及

加法器,被配置为将所述第二残差帧与所述重构帧相加以生成改进的重构帧。

11.根据权利要求10所述的视频质量改进装置,其中,所述第一残差帧近似于原始残差帧,并且

其中,所述原始残差帧是所述原始帧与所述重构帧之间的差。

12.根据权利要求11所述的视频质量改进装置,其中,所述基于深度学习的改进模型是包括多个层的深度学习模型,并且所述基于深度学习的改进模型使用基于所述第一残差帧与所述原始残差帧之间的差的损失函数来训练。

13.根据权利要求11所述的视频质量改进装置,其中,基于所述第一残差帧中的像素值和所述原始残差帧中的像素值,使用线性最小二乘方程来估计所述线性模型的参数。

14.一种由计算装置执行的用于改进重构帧的视频质量的方法,所述方法包括:

获得所述重构帧,并将所述重构帧输入到基于深度学习的改进模型中以生成近似原始帧的输出,其中,所述重构帧是通过重构所述原始帧而获得的帧并且所述重构帧由所述计算装置预先重构;

通过从所述基于深度学习的改进模型的输出中减去所述重构帧来生成第一残差帧;

获得编码/解码信息,并将所述编码/解码信息输入到基于深度学习的嵌入函数中以生成嵌入矢量,其中,所述编码/解码信息是量化参数、在速率失真优化处理中使用的拉格朗日常数、图片组(GOP)中的时间层、以及帧的类型中的至少一者;

通过将所述嵌入矢量与所述第一残差帧进行矩阵相乘来生成经补充的残差帧;

通过将所述经补充的残差帧输入到线性模型来生成第二残差帧,其中,所述线性模型包括表示所述经补充的残差帧与所述第二残差帧之间的线性关系的参数;以及

通过将所述第二残差帧与所述重构帧相加来生成改进的重构帧。

15.根据权利要求14所述的方法,其中,所述第一残差帧近似于原始残差帧,并且所述原始残差帧是所述原始帧与所述重构帧之间的差。

16.根据权利要求15所述的方法,其中,所述基于深度学习的改进模型是包括多个层的深度学习模型,并且所述基于深度学习的改进模型使用基于所述第一残差帧与所述原始残差帧之间的差的损失函数来训练。

17.根据权利要求15所述的方法,其中,基于所述经补充的残差帧中的像素值和所述原始残差帧中的像素值,使用线性最小二乘方程来估计所述线性模型的参数。

使用改进的环内滤波器的视频编码方法和设备

技术领域

[0001] 本公开涉及使用改进的环路滤波器的视频编码方法和设备。

背景技术

[0002] 此部分中的陈述仅提供与本公开相关的背景信息,并且不一定构成现有技术。

[0003] 由于视频数据与音频或静态图像数据相比具有大量的数据,因此视频数据需要大量硬件资源(包括存储器)来存储或传输视频数据,而无需用于压缩的处理。

[0004] 因而,编码器通常用于压缩和存储或传输视频数据。解码器接收压缩视频数据,对接收到的压缩视频数据进行解压缩,并播放解压缩后的视频数据。视频压缩技术包括H.264/AVC、高效率视频编码(HEVC)和多功能视频编码(VVC),VVC与HEVC相比具有约30%以上的改进的编码效率。

[0005] 然而,由于图像尺寸、分辨率和帧速率逐渐增加,要编码的数据量也增加。因而,需要提供比现有压缩技术更高的编码效率和改进的图像增强效果的新压缩技术。

[0006] 基于深度学习技术的一般视频质量改进算法试图通过根据跳过路径使用残差信号来改进视频质量并减少计算量。此外,基于深度学习的图像处理技术已经被应用到现有的编码元素技术中。在现有的编码技术中,通过将基于深度学习的图像处理技术应用于诸如帧间预测、帧内预测、环内滤波器和变换的压缩技术,可以改进编码效率。代表性的应用实施例包括基于虚拟参考帧的帧间预测和基于重构帧的环内滤波器,虚拟参考帧基于深度学习模型而生成。因此,在视频编码/解码中,需要考虑使用残差信号的基于深度学习的图像处理技术的应用,以改进编码效率和视频质量。

发明内容

[0007] 技术问题

[0008] 本发明试图提供一种用于使用深度学习模型从重构帧生成残差帧以改进视频编码效率和视频质量的视频编码方法和设备。该视频编码方法和设备通过将生成的残差帧应用于线性模型来近似原始残差帧,从而改进环内滤波器的性能。

[0009] 技术方案

[0010] 本发明的至少一个方面提供一种由计算装置执行的用于改进重构帧的视频质量的方法。该方法包括获得重构帧并将重构帧输入到基于深度学习的改进模型中以生成近似原始帧的输出。在此,重构帧是通过重构原始帧而获得的帧并且其由计算装置预先重构。该方法还包括通过从基于深度学习的改进模型的输出中减去重构帧来生成第一残差帧。该方法还包括将第一残差帧输入到线性模型以生成第二残差帧。在此,线性模型包括表示第一残差帧与第二残差帧之间的线性关系的参数。该方法还包括通过将第二残差帧与重构帧相加来生成改进的重构帧。

[0011] 本发明的另一方面提供一种包括在视频解码设备中的视频质量改进装置。该视频质量改进装置包括第一改进器,该第一改进器包括基于深度学习的改进模型。第一改进器

用于获得重构帧,并将重构帧输入到基于深度学习的改进模型中,以生成与原始帧近似的输出。在此,重构帧是通过重构原始帧而获得的帧并且其由视频解码设备预先重构。视频质量改进装置还包括减法器,该减法器被配置为从改进模型的输出中减去重构帧以生成第一残差帧。该视频质量改进装置还包括第二改进器,第二改进器包括线性模型。第二改进器用于将第一残差帧输入到线性模型以生成第二残差帧。在此,线性模型包括表示第一残差帧与第二残差帧之间的线性关系的参数。该视频质量改进装置还包括加法器,加法器被配置为将第二残差帧与重构帧相加以生成改进的重构帧。

[0012] 本发明的又一方面提供一种由计算装置执行的用于改进重构帧的视频质量的方法。该方法包括获得重构帧并将重构帧输入到基于深度学习的改进模型中以生成近似原始帧的输出。在此,重构帧是通过重构原始帧而获得的帧并且其由计算装置预先重构。该方法还包括通过从基于深度学习的改进模型的输出中减去重构帧来生成第一残差帧。该方法还包括获得编码/解码信息并将该编码/解码信息输入到基于深度学习的嵌入函数中以生成嵌入矢量。在此,编码/解码信息是量化参数、在速率失真优化处理中使用的拉格朗日常数、图片组(GOP)中的时间层、以及帧的类型中的至少一个。该方法还包括:通过将该嵌入矢量与该第一残差帧进行矩阵相乘来生成经补充的残差帧。该方法还包括:通过将该经补充的残差帧输入到线性模型来生成第二残差帧。在此,线性模型包括表示经补充的残差帧与第二残差帧之间的线性关系的参数。该方法还包括通过将第二残差帧与重构帧相加来生成改进的重构帧。

[0013] 有益效果

[0014] 如上所述,本公开提供一种用于使用深度学习模型从重构帧生成残差帧,并通过将生成的残差帧应用于线性模型来近似原始残差帧的视频编码方法和设备。由此,视频编码方法和设备根据环内滤波器的性能改进来改进视频编码效率和视频质量。

附图说明

[0015] 图1是可以实现本发明的技术的视频编码设备的框图。

[0016] 图2示出了使用四叉树加二叉树三叉树(QTBTTT)结构来划分块的方法。

[0017] 图3a和图3b示出了包括广角帧内预测模式的多个帧内预测模式。

[0018] 图4示出了当前块的相邻块。

[0019] 图5是可以实现本发明的技术的视频解码设备的框图。

[0020] 图6是示出根据本公开的实施方式的视频质量改进装置的示图。

[0021] 图7是示出根据本发明另一实施方式的视频质量改进装置的示图。

[0022] 图8是示出根据本公开的另一实施方式的视频质量改进装置的示图。

[0023] 图9是示出根据本公开的实施方式的用于改进视频质量的方法的流程图。

[0024] 图10是示出根据本公开的实施方式的用于改进视频质量的方法的流程图。

具体实施方式

[0025] 在下文中,参照附图详细描述本公开的一些实施方式。在以下描述中,尽管元件在不同的附图中示出,但是相同的参考标号表示相同的元件。此外,在一些实施方式的以下描述中,为了清楚和简洁的目的,当被认为使本公开的主题模糊时,相关已知组件和功能的详

细描述可以被省略。

[0026] 图1是可以实现本公开的技术的视频编码设备的框图。在下文中,参照图1的图示,描述了视频编码设备和该设备的组件。

[0027] 编码设备可包括图片分割器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、环路滤波器单元180和存储器190。

[0028] 编码设备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。此外,每个组件的功能可以实现为软件,并且还可以实现微处理器以执行与每个组件相对应的软件的功能。

[0029] 一个视频由包括多个图像的一个或多个序列构成。每个图片被分割为多个区域,并对每个区域执行编码。例如,将一个图片分割成一个或多个瓦片或/和切片。在此,一个或多个瓦片可以被定义为瓦片组。每个瓦片或/和切片被分割成一个或多个编码树单元(CTU)。另外,每个CTU通过树结构分割成一个或多个编码单元(CU)。将应用于每个编码单元(CU)的信息编码为CU的语法,并且将共同应用于包含在一个CTU中的CU的信息编码为CTU的语法。此外,共同应用于一个切片中的所有块的信息被编码为切片报头的语法,并且应用于构成一个或更多个图片的所有块的信息被编码为图片参数集(PPS)或图片报头。此外,多个图片共同引用的信息被编码到序列参数集(SPS)。另外,一个或多个SPS共同引用的信息被编码到视频参数集(VPS)。进一步,共同应用于一个瓦片或瓦片组的信息也可被编码为瓦片或瓦片组报头的语法。包括在SPS、PPS、切片报头、瓦片或瓦片组报头中的语法可以被称为高级语法。

[0030] 图片分割器110确定编码树单元(CTU)的尺寸。关于CTU的尺寸的信息(CTU尺寸)被编码为SPS或PPS的语法并且被递送到视频解码设备。

[0031] 图片分割器110将构成视频的每个图片分割成具有预定尺寸的多个编码树单元(CTUs),然后通过使用树结构递归分割CTU。树结构中的叶节点变成编码单元(CU),其是编码的基本单元。

[0032] 树结构可以是四叉树(QT),其中,较高节点(或父节点)被分割成具有相同尺寸的四个较低节点(或子节点)。树结构也可以是二进制树(BT),其中较高节点被分割成两个较低节点。树结构还可以是三叉树(TT),其中,较高节点以1:2:1的比率被分割成三个较低节点。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,可以使用四叉树加二叉树(QTBT)结构或者可以使用四叉树加二叉树三叉树(QTBTTT)结构。这里,将二叉树三叉树(BTTT)添加到树结构被称为多类型树(MTT)。

[0033] 图2是描述通过使用QTBTTT结构来分割块的方法的示图。

[0034] 如图2中所示,CTU可以首先被分割成QT结构。四叉树分割可以是递归的,直到分割的块的尺寸达到在QT中允许的叶节点的最小块尺寸(MinQTSize)。指示QT结构的每个节点是否被分割成下层的四个节点的第一标记(QT_split_flag)由熵编码器155编码,并用信号发送给视频解码设备。当QT的叶节点不大于在BT中允许的根节点的最大块尺寸(MaxBTSize)时,叶节点可进一步被分割为BT结构和TT结构中的至少一者。在BT结构和/或TT结构中可存在多个分割方向。例如,可存在两个方向,即,对应节点的块被水平分割的方向和对应节点的块被垂直分割的方向。如图2所示,当MTT分割开始时,指示节点是否被分割

的第二标记(mtt_split_flag)以及另外指示分割方向(垂直或水平)的标记和/或如果节点被分割则指示分割类型(二叉或三叉)的标记由熵编码器155编码并用信号发送给视频解码设备。

[0035] 可替代地,在编码指示每个节点是否被分割成下层的四个节点的第一标记(QT_split_flag)之前,指示该节点是否被分割的CU分割标记(split_cu_flag)也可被编码。当CU分割标记(split_cu_flag)的值指示每个节点未分割时,对应节点的块变为分割树结构中的叶节点且变为作为编码的基本单元的CU。当CU分割标记(split_cu_flag)的值指示每个节点被分割时,视频编码设备首先通过上述方案开始编码第一标记。

[0036] 当QTBT被用作树结构的另一实施例时,可存在两种类型,即,对应节点的块被水平分割成具有相同尺寸的两个块的类型(即,对称水平分割)和对应节点的块被垂直分割成具有相同尺寸的两个块的类型(即,对称垂直分割)。指示BT结构的每个节点是否被分割为下层的块的分割标记(split_flag)和指示分割类型的分割类型信息由熵编码器155编码,并被递送到视频解码设备。同时,可以另外呈现将对应节点的块分割为彼此不对称的两个块的类型。非对称形式可包括其中对应节点的块被分割为具有1:3的尺寸比的两个矩形块的形式,或者还可包括其中对应节点的块被沿对角线方向分割的形式。

[0037] CU可根据来自CTU的QTBT或QTBTTT分割而具有不同尺寸。在下文中,对应于待编码或待解码的CU(即,QTBTTT的叶节点)的块被称为“当前块”。由于采用QTBTTT分割,当前块的形状除了正方形形状外,还可以是矩形形状。

[0038] 预测器120预测当前块以生成预测块。预测器120包括帧内预测器122和帧间预测器124。

[0039] 通常,图片中的每个当前块可被预测性编码。通常,当前块的预测可以通过使用帧内预测技术(使用来自包括当前块的图片的数据)或者帧间预测技术(使用来自在包括当前块的图片之前被编码的图片的数据)来执行。帧间预测包括单向预测和双向预测两者。

[0040] 帧内预测器122通过使用在包括当前块的当前图片中位于当前块的邻域上的像素(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模式和DC模式的2个非定向模式,并且可以包括65个定向模式。邻近像素和将被使用的算术等式根据每个预测模式而被不同地定义。

[0041] 为了对具有矩形形状的当前块进行有效的方向预测,可以另外使用如图3b中的虚线箭头所示的定向模式(#67至#80,帧内预测模式#-1至#-14)。定向模式可被称作“广角帧内预测模式”。在图3b中,箭头指示用于预测的对应参考样本并且不表示预测方向。预测方向与箭头所示的方向相反。在当前块具有矩形形状时,广角帧内预测模式是在没有附加比特传输的情况下以与特定定向模式相反的方向执行预测的模式。在这种情况下,在广角帧内预测模式中,可通过具有矩形形状的当前块的宽度和高度的比率确定可用于当前块的一些广角帧内预测模式。例如,在当前块具有高度小于宽度的矩形形状时,可使用具有小于45度的角度的广角帧内预测模式(帧内预测模式#67至#80)。在当前块具有宽度大于高度的矩形形状时,可使用具有大于-135度的角度的广角帧内预测模式。

[0042] 帧内预测器122可以确定将用于对当前块进行编码的帧内预测。在一些实施例中,帧内预测器122可通过使用多个帧内预测模式对当前块进行编码,且还可从测试的模式中选择将使用的适当帧内预测模式。例如,帧内预测器122可通过使用针对多个测试的帧内预

测模式的速率-失真分析来计算速率-失真值,并且还可在测试的模式中选择具有最佳速率-失真特征的帧内预测模式。

[0043] 帧内预测器122在多个帧内预测模式当中选择一个帧内预测模式,并通过使用相邻像素(参考像素)和根据所选择的帧内预测模式确定的算术等式来预测当前块。关于所选择的帧内预测模式的信息由熵编码器155编码,并被递送到视频解码设备。

[0044] 帧间预测器124通过使用运动补偿处理来生成对于当前块的预测块。帧间预测器124在比当前图片更早地编码和解码的参考图片中搜索与当前块最相似的块,并通过使用搜索到的块生成对于当前块的预测块。另外,生成运动矢量(MV),其对应于当前图片中的当前块与参考图片中的预测块之间的位移。通常,对亮度分量执行运动估计,并且基于亮度分量计算的运动矢量被用于亮度分量和色度分量两者。包括关于参考图片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由熵编码器155编码,并被递送到视频解码设备。

[0045] 帧间预测器124还可对参考图片或参考块执行插值以便增加预测的准确性。换句话说,通过将滤波器系数应用于包括两个整数样本的多个连续整数样本,来对两个连续整数样本之间的子样本进行插值。当针对插值的参考图片执行搜索与当前块最相似的块的处理时,可以针对运动矢量表达小数单位精度,而不是整数样本单位精度。可针对待编码的每个目标区域(例如,诸如切片、瓦片、CTU、CU等的单元)不同地设定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(AMVR)时,应针对每个目标区域用信号发送关于待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,用信号发送关于对于每个CU应用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表示将在下面描述的运动矢量差的精度的信息。

[0046] 同时,帧间预测器124可通过使用双向预测来执行帧间预测。在双向预测的情况下,使用两个参考图片和表示在每个参考图片中最类似于当前块的块位置的两个运动矢量。帧间预测器124分别从参考图片列表0(RefPicList0)和参考图片列表1(RefPicList1)选择第一参考图片和第二参考图片。帧间预测器124还在相应参考图片中搜索与当前块最相似的块以生成第一参考块和第二参考块。另外,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。另外,包括关于用于预测当前块的两个参考图片的信息并且包括关于两个运动矢量的信息的运动信息被递送到熵编码器155。在此,参考图片列表0可以由预重构的图片当中按照显示顺序在当前图片之前的图片构成,并且参考图片列表1可以由预重构的图片当中按照显示顺序在当前图片之后的图片构成。然而,尽管不特别限于此,但是按照显示顺序在当前图片之后的预重构的图片可以附加地包括在参考图片列表0中。相反,在当前图片之前的预重构的图片也可以被附加地包括在参考图片列表1中。

[0047] 为了最小化用于编码运动信息所消耗的比特数量,可以使用各种方法。

[0048] 例如,当当前块的参考图片和运动矢量与相邻块的参考图片和运动矢量相同时,能够识别相邻块的信息被编码以将当前块的运动信息递送到视频解码设备。这种方法被称为合并模式。

[0049] 在合并模式中,帧间预测器124从当前块的相邻块中选择预定数量的合并候选块(在下文中,称为“合并候选”)。

[0050] 作为用于导出合并候选的相邻块,如图4所示,可以使用在当前图片中邻近于当前块的左块A0、左下块A1、上块B0、右上块B1、以及左上块B2中的全部或者一些。此外,除当前

块所位于的当前图片之外,位于参考图片(其可以与用于预测当前块的参考图片相同或不同)内的块也可以用作合并候选。例如,在参考图片内与当前块位于同一地点的块或邻近于该位于同一地点的块的块可另外用作合并候选。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。

[0051] 帧间预测器124通过使用相邻块来配置包括预定数量的合并候选的合并列表。从包括在合并列表中的合并候选中选择要用作当前块的运动信息的合并候选,并且生成用于识别所选择的候选的合并索引信息。所生成的合并索引信息由熵编码器155编码,并被递送到视频解码设备。

[0052] 合并跳过模式是合并模式的特殊情况。在量化之后,当用于熵编码的所有变换系数接近于零时,仅传输相邻块选择信息而不传输残差信号。通过使用合并跳过模式,对于具有轻微运动的图像、静止图像、屏幕内容图像等可以实现相对高的编码效率。

[0053] 在下文中,合并模式和合并跳过模式被统称为合并/跳过模式。

[0054] 用于编码运动信息的另一方法是高级运动矢量预测(AMVP)模式。

[0055] 在AMVP模式下,帧间预测器124通过使用当前块的相邻块来导出用于当前块的运动矢量的运动矢量预测值候选(predictor candidate)。作为用于导出运动矢量预测值候选的相邻块,可以使用与在图4中示出的当前图片中的当前块邻近的左块A0、左下块A1、上块B0、右上块B1、以及左上块B2中的全部或一些。此外,除当前块所位于的当前图片之外,位于参考图片(其可以与用于预测当前块的参考图片相同或不同)内的块也可以用作用于导出运动矢量预测值候选的相邻块。例如,可以使用在参考图片内与当前块位于同一地点的块或者邻近于该位于同一地点的块的块。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,则零矢量被添加到运动矢量候选。

[0056] 帧间预测器124通过使用相邻块的运动矢量导出运动矢量预测值候选,并通过使用运动矢量预测值候选来确定对于当前块的运动矢量的运动矢量预测值。另外,通过从当前块的运动矢量减去运动矢量预测值来计算运动矢量差。

[0057] 运动矢量预测值可通过将预定义的函数(例如,中心值和平均值计算等)应用于运动矢量预测值候选来获得。在这种情况下,视频解码设备还知道预定义的函数。此外,由于用于导出运动矢量预测值候选的相邻块是编码和解码已经完成的块,所以视频解码设备还可能已经知道相邻块的运动矢量。因此,视频编码设备不需要对用于识别运动矢量预测值候选的信息进行编码。因而,在这种情况下,关于运动矢量差的信息和关于用于预测当前块的参考图片的信息被编码。

[0058] 同时,还可以通过选择运动矢量预测值候选中的任一者的方案来确定运动矢量预测值。在这种情况下,用于识别所选择的运动矢量预测值候选的信息与关于运动矢量差的信息和关于用于预测当前块的参考图片的信息被联合地另外编码。

[0059] 减法器130通过从当前块中减去由帧内预测器122或帧间预测器124生成的预测块来生成残差块。

[0060] 变换器140将具有空间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可通过将残差块的总尺寸用作变换单元来对残差块中的残差信号进行变换,或者也可将残差块分割成多个子块,并可通过将子块用作变换单元来执行变换。可替代地,残差块被划分为两个子块,这两个子块分别为变换区域和非变换区域,以仅使用变换区域

子块作为变换单元对残差信号进行变换。在此,变换区域子块可以是两个矩形块之一,其基于水平轴(或垂直轴)的尺寸比为1:1。在这种情况下,标记(cu_sbt_flag)指示仅子块被变换,并且定向(垂直/水平)信息(cu_sbt_horizontal_flag)和/或位置信息(cu_sbt_pos_flag)由熵编码器155编码并用信号发送给视频解码设备。此外,变换区域子块的尺寸可基于水平轴(或垂直轴)具有1:3的尺寸比。在这种情况下,划分对应分割的标记(cu_sbt_quad_flag)另外由熵编码器155编码,并用信号发送给视频解码设备。

[0061] 同时,变换器140可在水平方向和垂直方向上单独地对残差块执行变换。对于变换,可使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一对变换函数可定义为多重变换集(MTS)。变换器140可选择在MTS中具有最高变换效率的一个变换函数对,并可在水平和垂直方向中的每一者上变换残差块。关于MTS中的变换函数对的信息(mts_idx)由熵编码器155编码,并用信号发送给视频解码设备。

[0062] 量化器145使用量化参数量化从变换器140输出的变换系数,并将经量化的变换系数输出到熵编码器155。量化器145还可以立即量化相关的残差块,而无需对任何块或帧的变换。量化器145还可根据变换系数在变换块中的位置而应用不同的量化系数(缩放值)。应用于以2维布置的经量化的变换系数的量化矩阵可被编码并用信号发送给视频解码设备。

[0063] 重排单元150可以对经量化的残差值的系数值进行重新排列。

[0064] 重排单元150可以通过使用系数扫描将2D系数阵列改变为1D系数序列。例如,重排单元150可以通过使用Z字形扫描或者对角线扫描将DC系数扫描为高频域系数,来输出ID系数序列。根据变换单元的尺寸和帧内预测模式,也可使用沿列方向扫描2D系数阵列的垂直扫描和沿行方向扫描2D块类型系数的水平扫描来代替Z字形扫描。换句话说,根据变换单元的尺寸和帧内预测模式,可在Z字形扫描、对角线扫描、垂直扫描和水平扫描当中确定要使用的扫描方法。

[0065] 熵编码器155通过使用各种编码方案(包括基于上下文的自适应二进制算术码(CABAC)、指数哥伦布等),对从重排单元150输出的ID量化的变换系数的序列进行编码来生成比特流。

[0066] 此外,熵编码器155对与块分割相关的信息(诸如,CTU尺寸、CTU分割标记、QT分割标记、MTT划分类型、MTT分割方向等)进行编码,以允许视频解码设备将块均等分割到视频编码设备。此外,熵编码器155对关于预测类型的信息进行编码,该信息指示当前块是通过帧内预测还是帧间预测而编码的。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、合并索引的情况下,以及在AMVP模式的情况下,关于参考图片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。

[0067] 逆量化器160对从量化器145输出的经量化的变换系数进行去量化,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间域,以重构残差块。

[0068] 加法器170将重构的残差块和由预测器120生成的预测块相加,以重构当前块。当对下一次序块进行帧内预测时,经重构的当前块中的像素可用作参考像素。

[0069] 环路滤波器单元180对重构的像素执行滤波,以便减少由于基于块的预测和变换/量化而引起的块效应、振铃伪影、模糊伪影等。作为环路滤波器的环路滤波器单元180可包括去块滤波器182、样点自适应偏移(SAO)滤波器184和自适应环路滤波器(ALF)186中的全

部或一些。

[0070] 去块滤波器182对重构块之间的边界进行滤波以便去除由于块单位编码/解码而引起的块效应,并且SAO滤波器184和ALF 186对经去块的经滤波视频执行附加滤波。SAO滤波器184及ALF 186是用于补偿重构像素与原始像素之间的差的滤波器,该差由于有损编码导致。SAO滤波器184作为CTU单元来应用偏移以增强主观图像质量和编码效率。另一方面,ALF 186通过划分对应块的边界和变化量的程度来应用不同的滤波器,从而执行块单位滤波并补偿失真。关于将用于ALF的滤波器系数的信息可被编码并用信号发送给视频解码设备。

[0071] 通过去块滤波器182、SAO滤波器184和ALF 186滤波的重构块被存储在存储器190中。当一个图片中的所有块都被重构时,重构的图片可以用作用于对之后要编码的图片内的块进行帧间预测的参考图片。

[0072] 图5是可以实现本公开的技术的视频解码设备的功能框图。在下文中,参考图5,描述视频解码设备和该设备的组件。

[0073] 视频解码设备可包括熵解码器510、重排单元515、逆量化器520、逆变换器530、预测器540、加法器550、环路滤波器单元560和存储器570。

[0074] 与图1的视频编码设备相似,视频解码设备的每个组件可实现为硬件或软件或实现为硬件和软件的组合。此外,每个组件的功能可以实现为软件,并且还可以实现微处理器以执行与每个组件对应的软件的功能。

[0075] 熵解码器510通过对由视频编码设备生成的比特流进行解码来提取与块分割相关的信息,以确定待解码的当前块,并且提取重构当前块所需的预测信息和关于残差信号的信息。

[0076] 熵解码器510通过从序列参数集(SPS)或图片参数集(PPS)提取关于CTU尺寸的信息来确定CTU的尺寸,并且将图片分割成具有确定尺寸的CTU。另外,确定CTU为树结构的最高层,即根节点,并且可以提取CTU的分割信息,以使用树结构对CTU进行分割。

[0077] 例如,当使用QTBTTT结构对CTU进行分割时,首先提取与QT的分割相关的第一标记(QT_split_flag),以将每个节点分割成下层的四个节点。此外,针对与QT的叶节点相对应的节点,提取与MTT的分割相关的第二标记(mtt_split_flag)、分割方向(垂直/水平)和/或分割类型(二叉/三叉),以将对应的叶节点分割为MTT结构。结果,在QT的叶节点下面的每个节点被递归地分割成BT或TT结构。

[0078] 作为另一实施例,当通过使用QTBTTT结构来分割CTU时,提取指示CU是否被分割的CU分割标记(split_cu_flag)。当对应块被分割时,也可以提取第一标记(QT_split_flag)。在分割处理期间,对于每个节点,在0次或更多次的递归QT分割之后,可以发生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,或相反,也可以仅发生多次QT分割。

[0079] 作为另一实施例,当使用QTBT结构对CTU进行分割时,提取与QT的分割相关的第一标记(QT_split_flag),以将每个节点分割成下层的四个节点。另外,提取指示与QT的叶节点相对应的节点是否被进一步分割成BT的分割标记(split_flag)以及分割方向信息。

[0080] 同时,当熵解码器510通过使用树结构的分割而确定将被解码的当前块时,熵解码器510提取关于指示该当前块是帧内预测的还是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提取表示用于帧间预测信息的语法元素(即,运动矢量和该运动矢量引用的参考图片)的信息。

[0081] 此外,熵解码器510提取量化相关信息并且提取关于当前块的经量化的变换系数的信息,作为关于残差信号的信息。

[0082] 重排单元515可以再次以与由视频编码设备执行的系数扫描顺序相反的顺序,将由熵解码器510熵解码的经1D量化的变换系数的序列改变为2D系数阵列(即,块)。

[0083] 逆量化器520对经量化的变换系数去量化,并且通过使用量化参数对经量化的变换系数去量化。逆量化器520还可将不同的量化系数(缩放值)应用于以2D布置的经量化的变换系数。逆量化器520可通过将来自视频编码设备的量化系数(缩放值)的矩阵应用于经量化的变换系数的2D阵列来执行去量化。

[0084] 逆变换器530通过将去量化的变换系数从频域逆变换到空间域中来重构残差信号,从而生成对于当前块的残差块。

[0085] 此外,当逆变换器530逆变换变换块的部分区域(子块)时,逆变换器530提取变换块中仅子块被变换的标记(cu_sbt_flag)、子块的定向(垂直/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间域中以重构残差信号,并且利用值“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。

[0086] 此外,当应用MTS时,逆变换器530通过使用从视频编码设备发信号通知的MTS信息(mts_idx)来确定在水平方向和垂直方向中的每一个方向上应用的变换索引或变换矩阵。逆变换器530还通过使用所确定的变换函数在水平方向和垂直方向上对变换块中的变换系数执行逆变换。

[0087] 预测器540可包括帧内预测器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,并且在当前块的预测类型是帧间预测时激活帧间预测器544。

[0088] 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,在多个帧内预测模式当中确定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使用当前块的相邻参考像素来预测当前块。

[0089] 帧间预测器544通过使用从熵解码器510提取的用于帧间预测模式的语法元素来确定当前块的运动矢量和该运动矢量引用的参考图片。

[0090] 加法器550通过将从逆变换器530输出的残差块和从帧间预测器544或帧内预测器542输出的预测块相加来重构当前块。在对之后要解码的块进行帧内预测时,经重构的当前块内的像素被用作参考像素。

[0091] 作为环内滤波器的环路滤波器单元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对重构的块之间的边界执行去块滤波,以便去除由于块单位解码而引起的块效应。SAO滤波器564和ALF 566在去块滤波之后对重构块执行附加滤波,以便补偿重构像素和原始像素之间的差,该差由于有损编码导致。通过使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。

[0092] 通过去块滤波器562、SAO滤波器564和ALF 566滤波的重构块被存储在存储器570中。当一个图片中的所有块被重构时,重构的图片可以用作参考图片,用于对之后要编码的图片内的块进行帧间预测。

[0093] 在一些实施方式中,本发明涉及如上所述的对视频图像进行编码和解码。更具体地讲,本公开提供一种用于使用深度学习模型从重构帧生成残差帧的视频编码方法和设备。视频编码方法和设备通过将生成的残差帧应用于线性模型来近似原始残差帧,从而改进环内滤波器的性能。

[0094] 以下实施方式可共同应用于视频编码设备中的环路滤波器单元180和视频解码设备中的环路滤波器单元560。另外,以下实施方式可共同应用于视频编码设备和视频解码设备中使用深度学习技术的部分。

[0095] 在以下描述中,可以以与如上所述的当前块或者编码单元(CU)相同的含义使用要被编码或解码的术语“目标块”,或者“目标块”可以指编码单元的部分区域。

[0096] 图6是示出根据本公开的实施方式的视频质量改进装置的示图。

[0097] 根据本实施方式的视频质量改进装置使用深度学习模型从重构帧生成残差帧,将生成的残差帧应用于线性模型以近似原始残差帧,并通过使用近似的原始残差帧生成改进的重构帧。视频质量改进装置包括第一改进器602、减法器604、第二改进器606和加法器608中的全部或一些。另外,第一改进器602包括作为深度学习模型的改进模型,并且第二改进器606包括线性模型。

[0098] 当根据本实施方式的视频质量改进装置包括在视频编码设备中时,包括在视频质量改进装置中的组件不必限于图6中的实施例。例如,视频质量改进装置可另外包括用于训练改进模型的训练单元(未示出),或者可实现为链接到外部训练单元的形式。

[0099] 第一改进器602获得重构帧$x_{rec}[i,j]$作为输入,将重构帧$x_{rec}[i,j]$输入到改进模型,并输出通过对原始帧$x_{ori}[i,j]$进行近似而获得的$x_{nn}[i,j]$。在此,[i,j]指示像素在帧中的位置。改进模型可以使用多个层来生成近似输出x_{nn}。在此,多个层可以是适合于基于深度学习的视频信号处理的卷积层。

[0100] 减法器604从近似输出$x_{nn}[i,j]$中减去重构帧$x_{rec}[i,j]$以生成残差帧$x_{res}[i,j]$,如等式1所示。

[0101] [等式1]

[0102] $$x_{res}[i,j]=x_{nn}[i,j]-x_{rec}[i,j]$$

[0103] 因此,残差帧x_{res}近似于原始帧x_{ori}与重构帧x_{rec}之间的差。在下文中,该差被称为“原始残差帧x_{o_res}”。原始残差帧$x_{o_res}[i,j]$可被表示为等式2。

[0104] [等式2]

[0105] $$x_{o_res}[i,j]=x_{ori}[i,j]-x_{rec}[i,j]$$

[0106] 第二改进器606再次使用线性模型从残差帧x_{res}对原始残差帧x_{o_res}进行近似。当α和β是由线性模型提供的表示线性关系的参数时,线性近似的残差帧$x'_{res}[i,j]$可以表示为等式3。

[0107] [等式3]

[0108] $$x'_{res}[i,j]=\alpha \cdot x_{res}[i,j]+\beta$$

[0109] 在下文中,残差帧x_{res}与‘第一残差帧’可互换地使用,并且线性近似的残差帧x'_{res}与‘第二残差帧’可互换地使用。

[0110] 加法器608将第二残差帧$x'_{res}[i,j]$与重构帧$x_{rec}[i,j]$相加以生成改进的重构帧$x'_{rec}[i,j]$,如等式4所示。

[0111] [等式4]

[0112] $x'_{rec}[i,j]=x_{rec}[i,j]+\alpha \cdot x_{rec}[i,j]+\beta$

[0113] 从等式3和等式4,第二残差帧$x'_{res}[i,j]$可表示为等式5。

[0114] [等式5]

[0115] $x'_{res}[i,j]=x'_{rec}[i,j]-x_{rec}[i,j]$

[0116] 同时,训练单元可以使用原始残差帧x_{o_res}作为目标来训练改进模型,使得改进模型可以学习对原始帧x_{ori}进行近似。训练单元可使用如等式6中所示的损失函数来更新包括在多个层中的参数。

[0117] [等式6]

[0118] $$\begin{aligned} 损失 &= \|x_{res} - x_{o_{res}}\|^2 \\ &= \|x_{res} - (x_{ori} - x_{rec})\|^2 \end{aligned}$$

[0119] 这里,//·//表示L2度量。同时,如等式6所示,改进模型的损失函数表示为L2度量,但是不必限于此。可表示第一残差帧x_{res}与原始残差帧x_{o_res}之间的距离的任何度量可用作损失函数。

[0120] 线性模型的参数α和β可使用第一残差帧$x_{res}[i,j]$和第二残差帧$x'_{res}[i,j]$来计算。此时,由于第二残差帧$x'_{res}[i,j]$是尚未被确定或者接近原始残差帧的值,因此可以替代地使用原始残差帧$x_{o_res}[i,j]$。使用线性最小二乘方程,可以如方程7中所示估计α和β。

[0121] [等式7]

[0122] $$\alpha = \frac{K \cdot \sum_{k=1}^{K} x_{o_res}(k)x_{res}(k) - \sum_{k=1}^{K} x_{o_res}(k) \cdot \sum_{k=1}^{K} x_{res}(k)}{K \cdot \sum_{k=1}^{K} x_{res}(k)x_{res}(k) - (\sum_{k=1}^{K} x_{res}(k))^2}$$

[0123] $$\beta = \frac{\sum_{k=1}^{K} x_{o_res}(k) - \alpha \sum_{k=1}^{K} x_{res}(k)}{K}$$

[0124] 在此,K表示包括在帧中的像素的总数。换言之,为了计算α和β,[i,j]可以被[k]替换。因此,线性模型提供第一残差帧x_{res}中的像素与原始残差帧x_{o_res}中的像素之间的线性关系。

[0125] 同时,当α或β之一被设置为0时,可根据等式7估计非零参数。此后,第二改进器606可使用包括一个估计的参数的等式3来生成第二残差帧$x'_{res}[i,j]$。

[0126] 图7为示出根据本公开的另一实施方式的视频质量改进装置的示图。

[0127] 根据本实施方式的视频质量改进装置可使用视频编码/解码信息的嵌入矢量来补充第一残差帧x_{res}。视频质量改进装置可以另外包括嵌入矢量生成器702和矩阵乘法器704。

[0128] 嵌入矢量生成器702使用嵌入函数e(·)从视频编码/解码信息P生成嵌入矢量λ_p。在此,嵌入函数是包括全连接层或卷积层的深度学习模型。

[0129] 作为编码/解码信息P,可以使用量化参数、在速率失真优化处理中使用的拉格朗日常数、图片组(GOP)中的时间层、以及帧的类型(帧内/预测/双向预测(I/P/B))等。可替代地,它们中的全部或者一些的组合可以用作编码/解码信息P。

[0130] 矩阵乘法器704执行嵌入矢量λ_p和第一残差帧x_{res}的矩阵乘法,以生成尺寸与x_{res}相同的经补充的残差帧$x_{h_res}[i,j]$。在此,矩阵乘法可以是例如嵌入矢量与第一残差帧中的

行之间的逐分量乘法。为此,嵌入函数可被定义成使得嵌入矢量的尺寸等于第一残差帧中的行的尺寸。作为另一实施例,矩阵乘法可为嵌入矢量与第一残差帧中的列之间的逐分量乘法。为此,嵌入函数可以被定义成使得嵌入矢量的尺寸等于第一残差帧中的列的尺寸。

[0131] 第二改进器606再次使用线性模型从经补充的残差帧$x_{h_res}[i,j]$对原始残差帧x_{o_res}进行近似。对于表示线性模型的参数α和β,线性近似的残差帧$x'_{res}[i,j]$可以表示为等式8。

[0132] [等式8]

[0133] $x'_{res}[i,j]=\alpha \cdot x_{h_res}[i,j]+\beta$

[0134] 此时,可以使用原始残差帧$x_{o_res}[i,j]$和经补充的残差帧$x_{h_res}[i,j]$如等式9中所示来估计线性模型的参数α和β。

[0135] [等式9]

[0136] $$\alpha = \frac{K \cdot \sum_{k=1}^{K} x_{o_res}(k)x_{h_res}(k) - \sum_{k=1}^{K} x_{o_res}(k) \cdot \sum_{k=1}^{K} x_{h_res}(k)}{K \cdot \sum_{k=1}^{K} x_{h_res}(k)x_{h_res}(k) - \left(\sum_{k=1}^{K} x_{h_res}(k)\right)^2}$$

[0137] $$\beta = \frac{\sum_{k=1}^{K} x_{o_res}(k) - \alpha \sum_{k=1}^{K} x_{h_res}(k)}{K}$$

[0138] 作为实施例,视频编码设备可导出线性模型的参数,将参数编码为比特流,并将比特流发送到视频解码设备。此时,可以在帧、图片、子图片、切片、瓦片、块单元、以及像素单元的级别上确定是否应用视频质量改进。在块单元的情况下,可基于CTU和CU/预测单元(PU)来确定是否应用视频质量改进。可替代地,可基于子CU的尺寸来确定是否应用视频质量改进。可基于作为块集合的瓦片或子图片的尺寸来确定是否应用视频质量改进。

[0139] 作为另一实施例,视频编码设备可使用标记将是否改进视频质量用信号发送给视频解码设备。视频编码设备可从高级别接收此标记,并且视频解码设备可通过对比特流进行解码来获得该标记。同时,如果不考虑应用单元来对参数信息进行编码的方法在速率失真方面比仅传输重构的信号的现有方法更具有破坏性,则可以根据标记值而不应用视频质量改进方法。

[0140] 作为另一实施例,除了传输用于所有应用单元的线性模型的参数值的方法之外,还可以在统计上利用根据视频编码/解码信息的参数值。例如,当量化参数是22时,可以预先计算频繁出现的线性模型的参数值,然后可以使用所计算的值预先设置参数值。可替代地,可基于根据编码/解码信息的参数值的统计数据来生成查找表,然后使用查找表。此时,可以根据量化参数值、帧类型、GOP内的时间标识符(时间ID)、应用的块的尺寸、块的帧内/帧间编码等生成一个或多个查找表,然后可以在应用视频质量改进方法时使用该查找表。

[0141] 作为另一实施例,可基于根据编码/解码信息的参数值的统计数据预先设置多个(α,β)对和对应的索引。视频编码设备用信号发送索引,并且视频解码设备可使用该索引来获得预定义的α和β。此时,当α或β固定为0时,可针对除了0之外的参数值用信号发送索引。

[0142] 同时,视频质量改进装置可以是包括在视频编码设备中的环路滤波器单元180或视频解码设备中的环路滤波器单元560内的滤波器。因此,重构帧x_{rec}是通过重构原始帧而获得的帧,并且可使用由视频编码设备或视频解码设备先前重构的任何帧。例如,重构帧x_{rec}可以是存储在存储器190或570中的经解码图片缓冲区(DPB)中的信号、去块滤波器182

或562的输出、SAO滤波器184或564的输出、双边滤波器(BF)的输出、以及ALF 186或566的输出中的一者。在此,BF是环内滤波器之一并且可以被应用于改进重构帧的视频质量。另外,存储在DPB中的信号被定义为加法器170和550的输出,即,对应于预测信号和逆变换信号之和的重构信号。同时,DBP可以存储通过使所有环内滤波器通过而生成的重构图片,如图1和图5所示。

[0143] 作为另一实施例,除了环内滤波器之外,x_{rec}可以是经帧间预测的信号或经帧内预测的信号。通过根据本实施方式的视频质量改进方法可以执行预测信号的改进、压缩之后的视频质量的改进等。

[0144] 在图6和图7的实施例中,根据等式1的x_{nn}是改进模型的输出,改进模型是深度学习模型。然而,当不使用深度学习模型时,x_{nn}是去块滤波器182或562的输出、SAO滤波器184或564的输出、双边滤波器的输出、以及ALF 186或566的输出中的一者。

[0145] 图8是示出根据本公开的另一实施方式的视频质量改进装置的示图。

[0146] 例如,如图8所示,x_{rec}可用作SAO 564的输出和ALF 566的输入,且x_{nn}可用作ALF 566的输出。此后,x_{nn}可用以生成重构信号x'_{rec}。在图8中示出的视频质量改进装置基于视频解码设备中的环内滤波器而配置,但不限于此。例如,可基于视频编码设备中的环内滤波器来类似地配置视频质量改进装置。

[0147] 在下文中,使用图9和图10描述使用线性模型改进视频质量的方法。

[0148] 图9是示出根据本公开的实施方式的用于改进视频质量的方法的流程图。

[0149] 视频质量改进装置获得重构帧(S900)。

[0150] 视频质量改进装置将重构帧输入到基于深度学习的改进模型中并生成近似原始帧的输出(S902)。在此,重构帧是通过重构原始帧而获得的帧,并且通过视频编码设备或视频解码设备预先重构。例如,重构帧可以是存储在存储器190或570中的DPB中的信号、去块滤波器182或562的输出、SAO滤波器184或564的输出、BF的输出、以及ALF 186或566的输出中的一者。

[0151] 视频质量改进装置通过从改进模型的输出中减去重构帧来生成第一残差帧(S904)。第一残差帧近似于原始残差帧,原始残差帧是原始帧与重构帧之间的差。

[0152] 改进模型是包括多个层并被训练以学习对原始帧进行近似的深度学习模型。如等式6所示,可以使用基于第一残差帧与原始残差帧之间的差的损失函数来训练改进模型。

[0153] 视频质量改进装置将第一残差帧输入到线性模型中以生成第二残差帧(S906)。在此,线性模型包括表示第一残差帧与第二残差帧之间的线性关系的参数α和β。

[0154] 该线性模型的参数可以使用该第一残差帧和该第二残差帧来计算。此时,第二残差帧是尚未确定的值,但是由于其近似于原始残差帧,因此可以替代地使用原始残差帧。可以使用线性最小二乘方程如等式7所示来计算线性模型的参数。该线性模型可提供第一残差帧中的像素值与原始残差帧中的像素值之间的线性关系。

[0155] 视频质量改进装置将第二残差帧和重构帧相加以生成改进的重构帧(S908)。

[0156] 图10是示出根据本公开的实施方式的用于改进视频质量的方法的流程图。

[0157] 视频质量改进装置获得重构帧(S1000)。

[0158] 视频质量改进装置将重构帧输入到基于深度学习的改进模型中并生成近似于原始帧的输出(S1002)。在此,重构帧是通过重构原始帧而获得的帧,并且通过视频编码设备

或视频解码设备预先重构。

[0159] 视频质量改进装置通过从改进模型的输出中减去重构帧来生成第一残差帧(S1004)。

[0160] 视频质量改进装置获得编码/解码信息(S1006)。在此,量化参数、在速率失真优化处理中使用的拉格朗日常数、GOP内的时间层、帧的类型(I/P/B)等可以用作编码/解码信息。可替代地,可以将它们中的全部或者一些的组合用作编码/解码信息。

[0161] 视频质量改进装置通过将编码/解码信息输入到基于深度学习的嵌入函数中来生成嵌入矢量(S1008)。嵌入函数是包括全连接层或卷积层的深度学习模型。

[0162] 视频质量改进装置执行嵌入矢量和第一残差帧的矩阵乘法,以生成经补充的残差帧(S1010)。在此,矩阵乘法可以是例如嵌入矢量与第一残差帧中的行之间的逐分量乘法。作为另一实施例,矩阵乘法可以是嵌入矢量与第一残差帧中的列之间的逐分量乘法。

[0163] 视频质量改进装置通过将经补充的残差帧输入到线性模型来生成第二残差帧(S1012)。在此,线性模型包括表示经补充的残差帧与第二残差帧之间的线性关系的参数α和β。线性模型的参数可以如等式9所示使用原始残差帧和经补充的残差帧来计算。

[0164] 视频质量改进装置通过将第二残差帧和重构帧相加来生成改进的重构帧(S1014)。

[0165] 尽管各个流程图中的步骤被描述为被顺序地执行,但是这些步骤仅仅例示本公开的一些实施方式的技术构思。因此,本公开所属领域的普通技术人员可通过改变在各个附图中描述的顺序或者通过并行执行两个以上的步骤来执行步骤。所以,各个流程图中的步骤不限于所示的时间序列顺序。

[0166] 应当理解,以上描述呈现了可以以不同其他方式实现的说明性实施方式。在一些实施方式中描述的功能可以通过硬件、软件、固件和/或其组合来实现。还应该理解的是,本发明中所描述的功能组件被标注为“……单元”,以强烈强调其独立实现的可能性。

[0167] 同时,在一些实施方式中描述的各种方法或功能可以被实现为存储在能够由一个或多个处理器读取和执行的非暂时性记录介质中的指令。例如,非暂时性记录介质可包括各种类型的记录装置,其中,数据以计算机系统可读的形式存储。例如,非暂时性记录介质可包括存储介质,诸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。

[0168] 尽管出于说明性的目的已经描述了本公开的实施方式,然而,本公开所属领域的普通技术人员应当认识到,在不背离本公开的构思和范围的情况下,各种修改、添加、以及替换是可能的。因此,为了简洁和清晰起见,本公开的实施方式已经进行描述。本公开的实施方式的技术构思的范围不受图示的限制。因此,本公开所属领域的普通技术人员应当理解,本公开的范围不应受上述明确描述的实施方式的限制,而是受权利要求及其等同物的限制。

[0169] (参考标号)

[0170] 180 环路滤波器单元

[0171] 560 环路滤波器单元

[0172] 602 第一改进器

[0173] 606 第二改进器

[0174] 702 嵌入矢量生成器

[0175] 相关申请的交叉引用

[0176] 本申请要求于2021年8月17日提交的韩国专利申请第10-2021-0108041号以及2022年7月19日提交的韩国专利申请第10-2022-0089020号的优先权和权益,通过引用将每一个申请的全部公开内容结合于此。

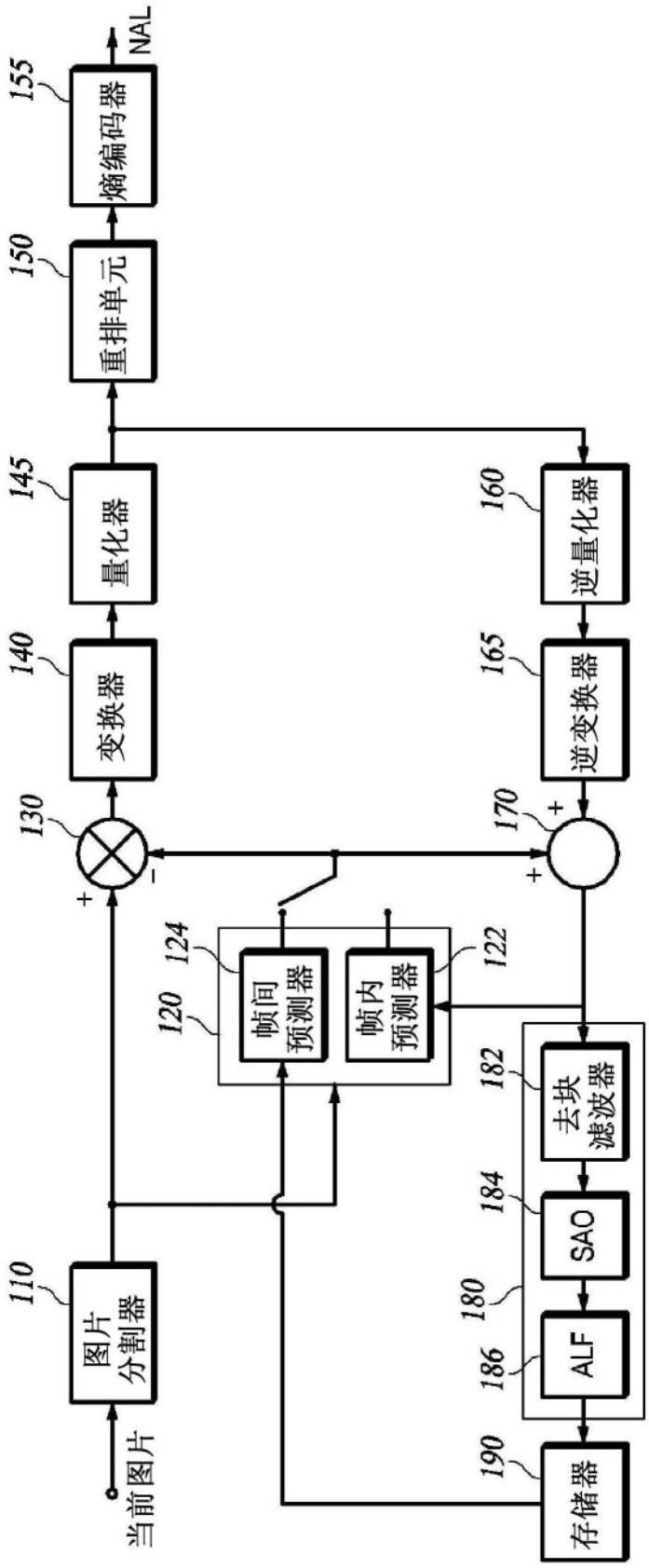

155
熵编码器
NAL
150
重排单元
145
量化器
140
变换器
130
160
逆量化器
165
逆变换器
170
124
帧间预测器
122
帧内预测器
120
182
去块滤波器
184
SAO
180
186
ALF
190
存储器
110
图片分割器
当前图片

图1

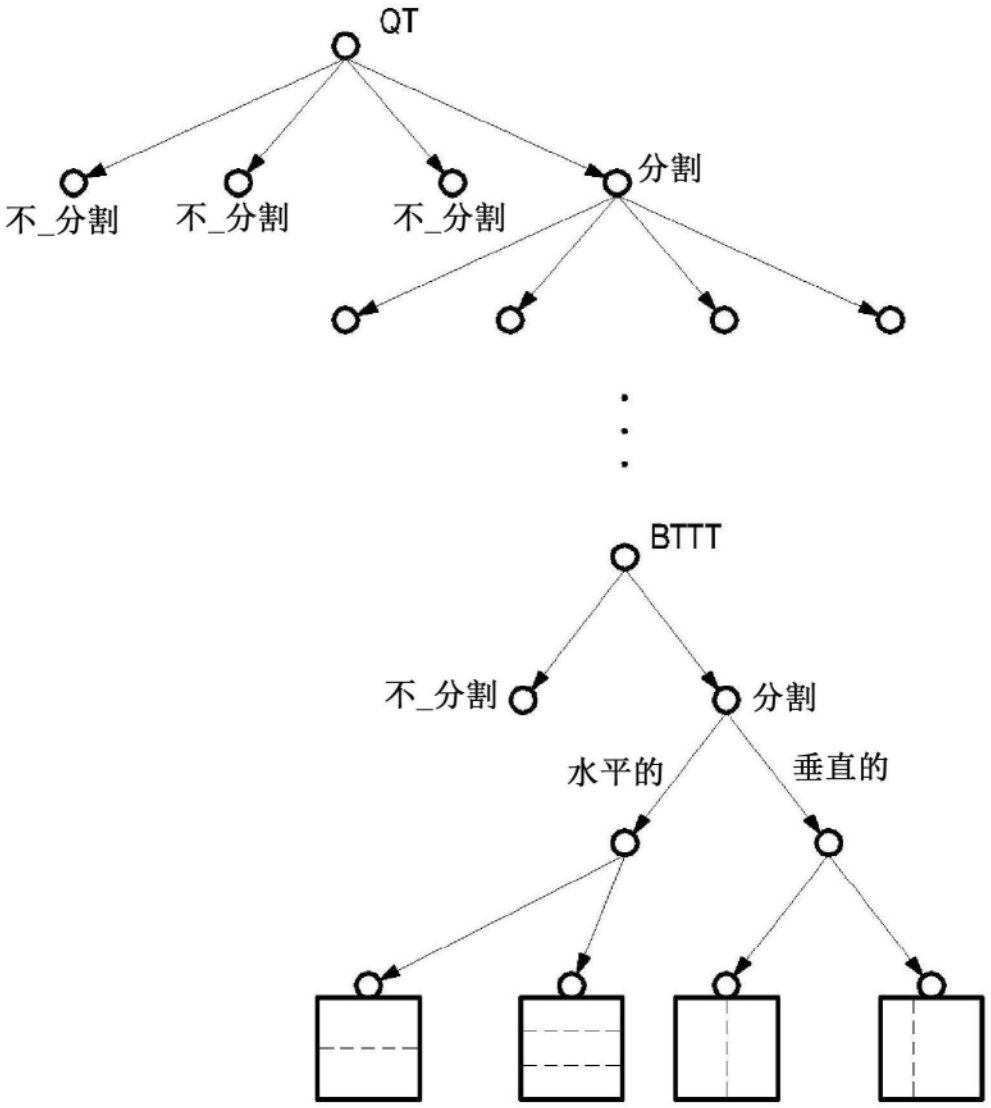


图2

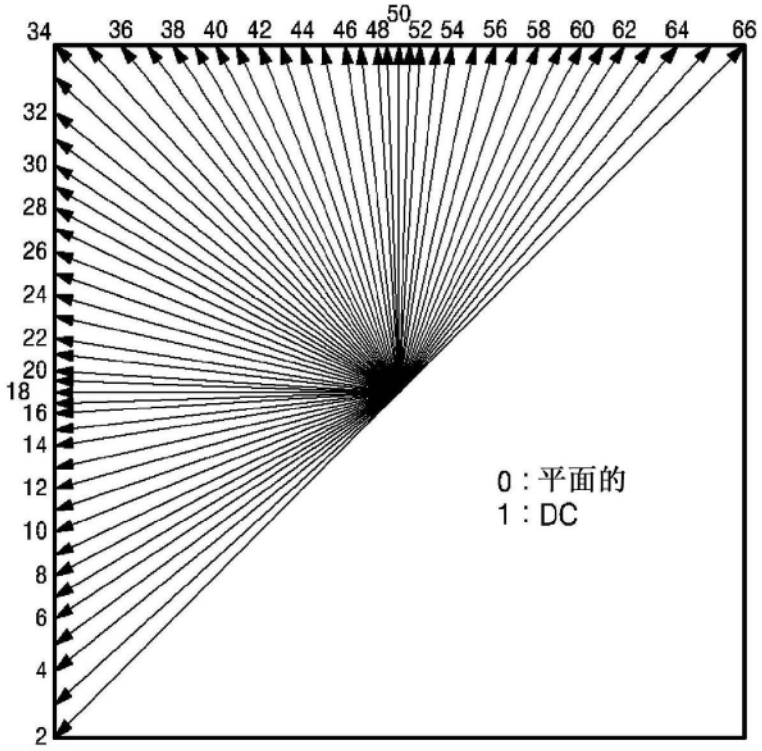

34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2
0：平面的
1：DC

图3a

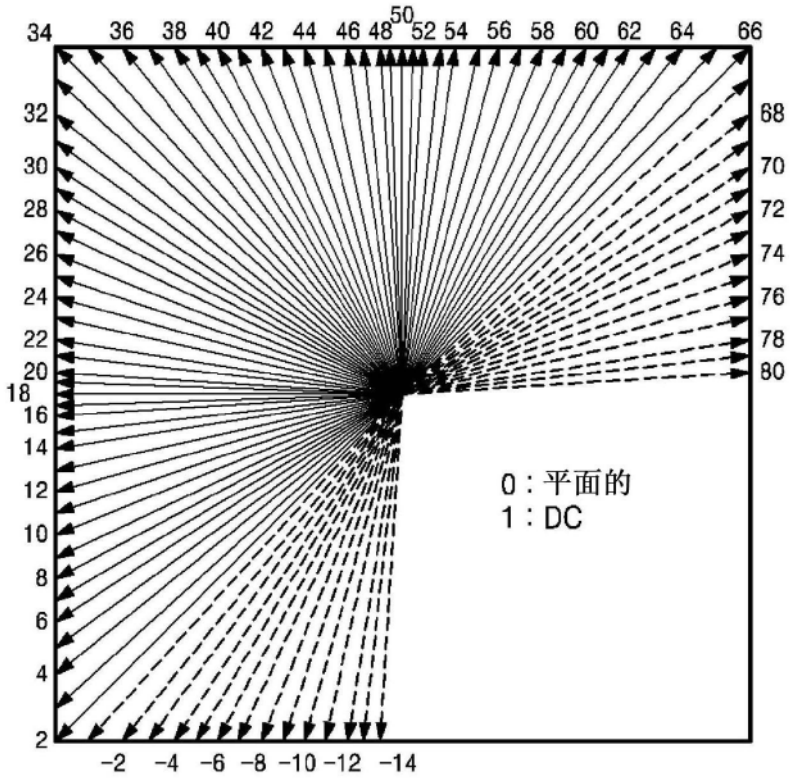

34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66
32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2
68 70 72 74 76 78 80
-2 -4 -6 -8 -10 -12 -14
0：平面的
1：DC

图3b

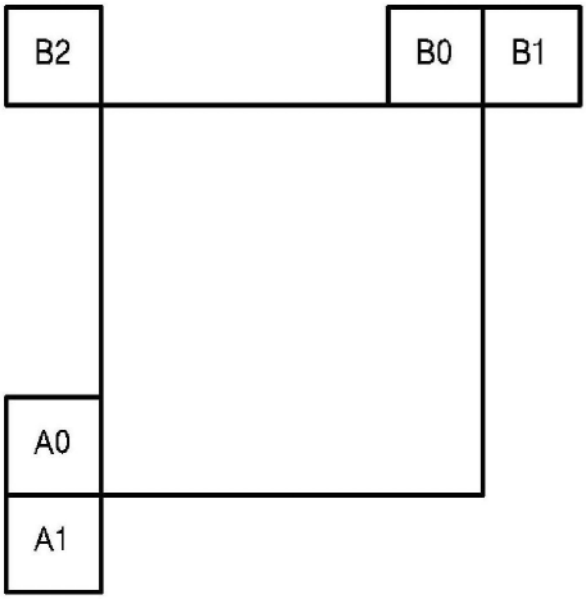

B2
B0
B1
A0
A1

图4

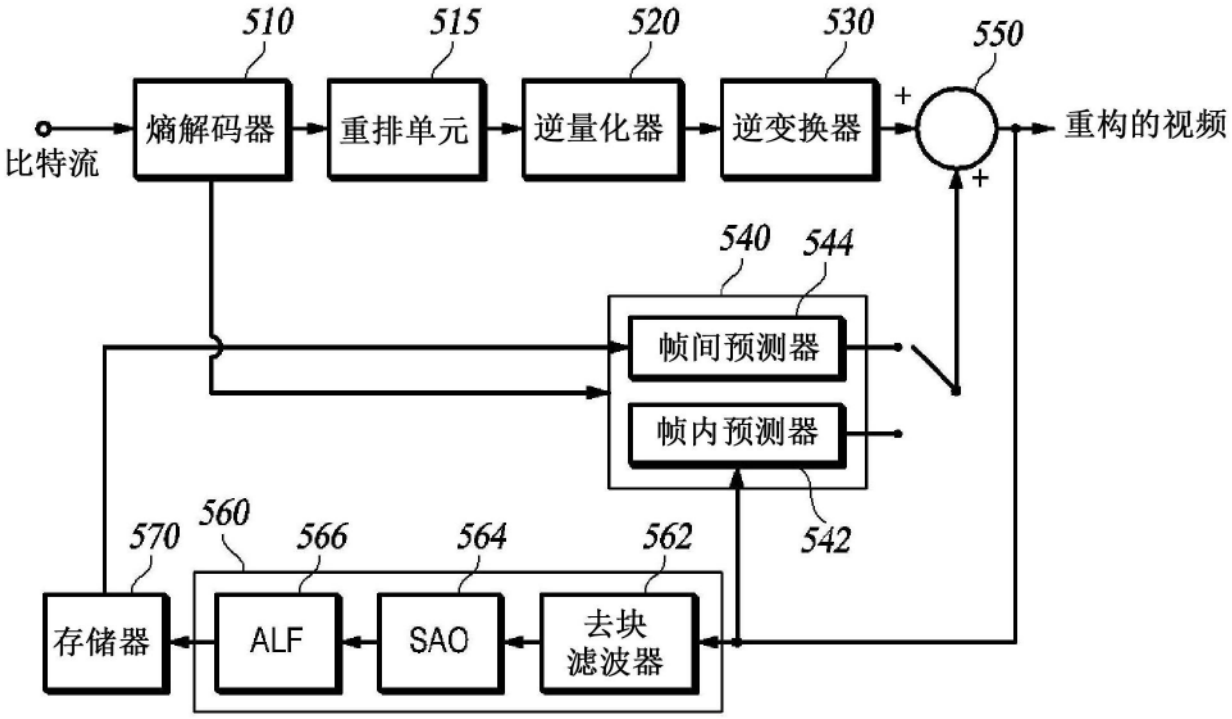

510
515
520
530
550
比特流
熵解码器
重排单元
逆量化器
逆变换器
+
+
重构的视频
540
544
帧间预测器
帧内预测器
542
570
560
566
564
562
存储器
ALF
SAO
去块
滤波器

图5

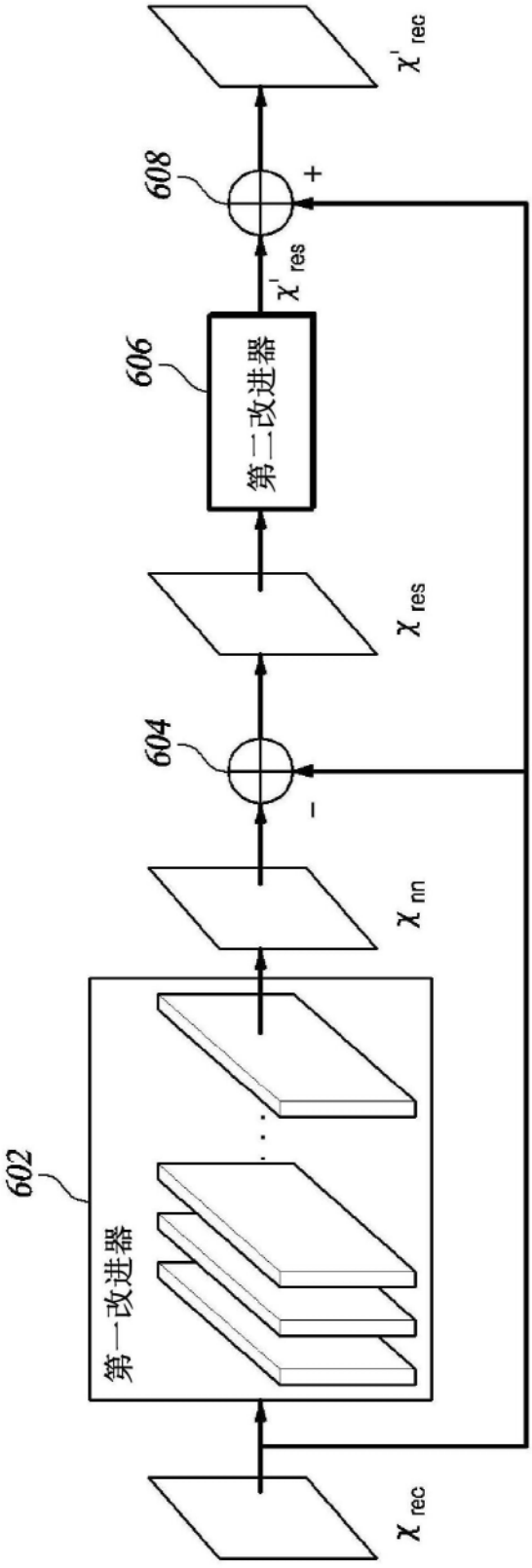


图6

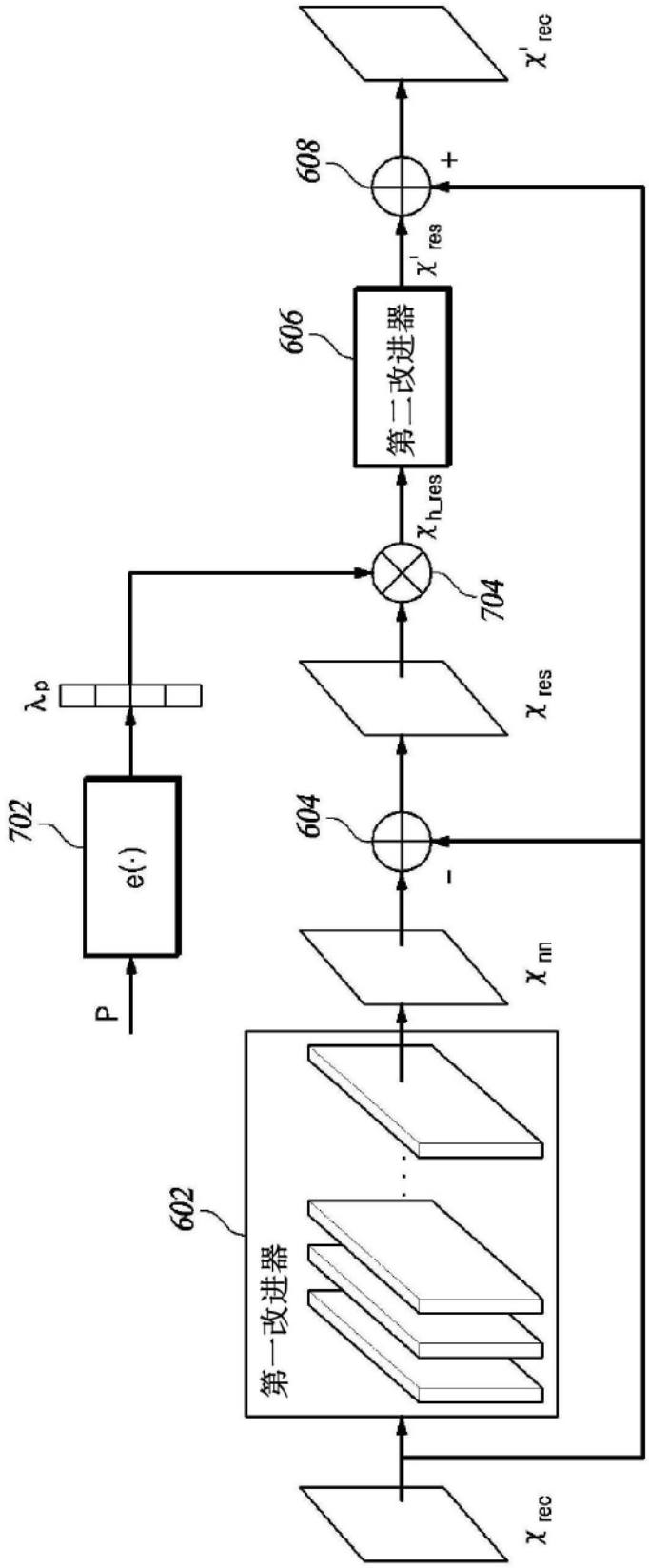

χ'rec
608
+
χ'res
606
第二改进器
χh_res
704
λp
χres
702
e(·)
604
−
P
χnn
602
第一改进器
χrec

图7

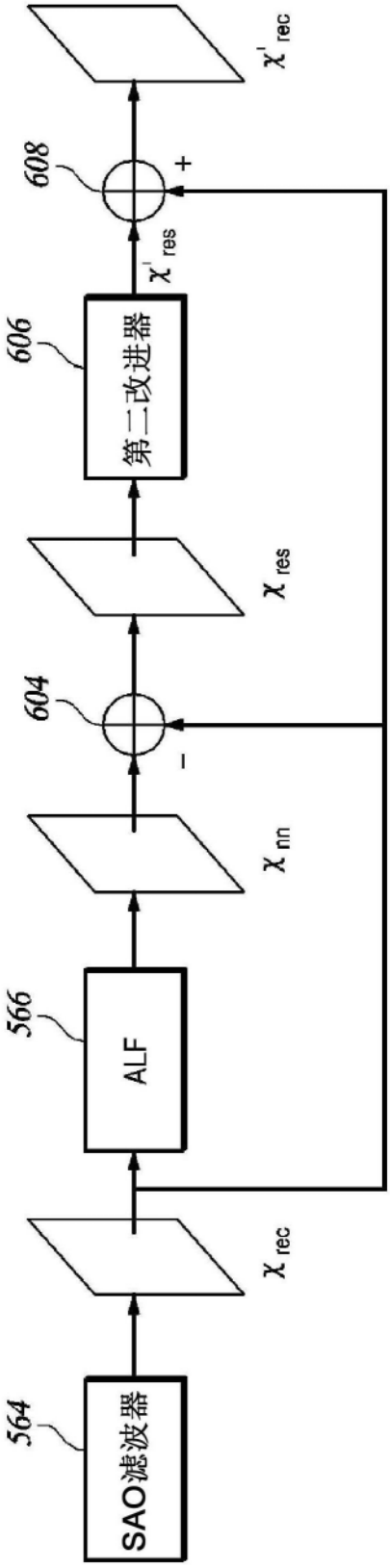


图8

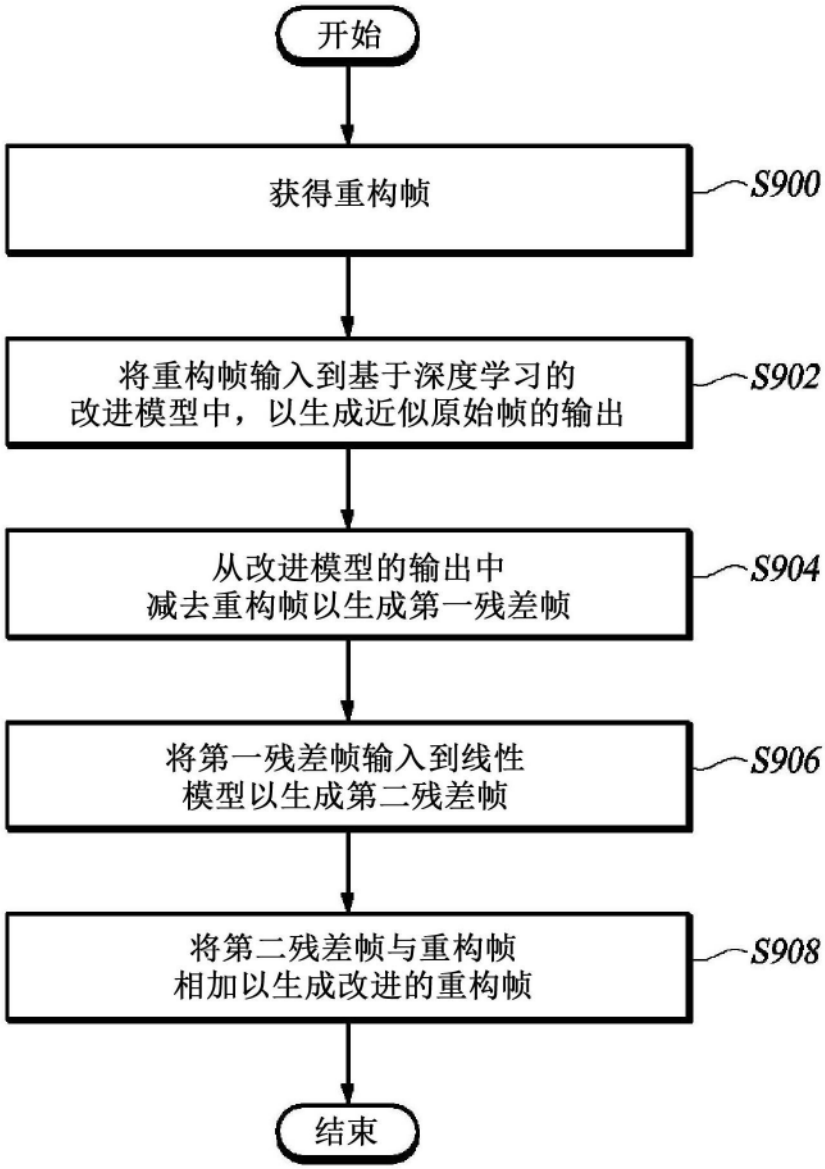

开始
获得重构帧
S900
将重构帧输入到基于深度学习的
改进模型中，以生成近似原始帧的输出
S902
从改进模型的输出中
减去重构帧以生成第一残差帧
S904
将第一残差帧输入到线性
模型以生成第二残差帧
S906
将第二残差帧与重构帧
相加以生成改进的重构帧
S908
结束

图9

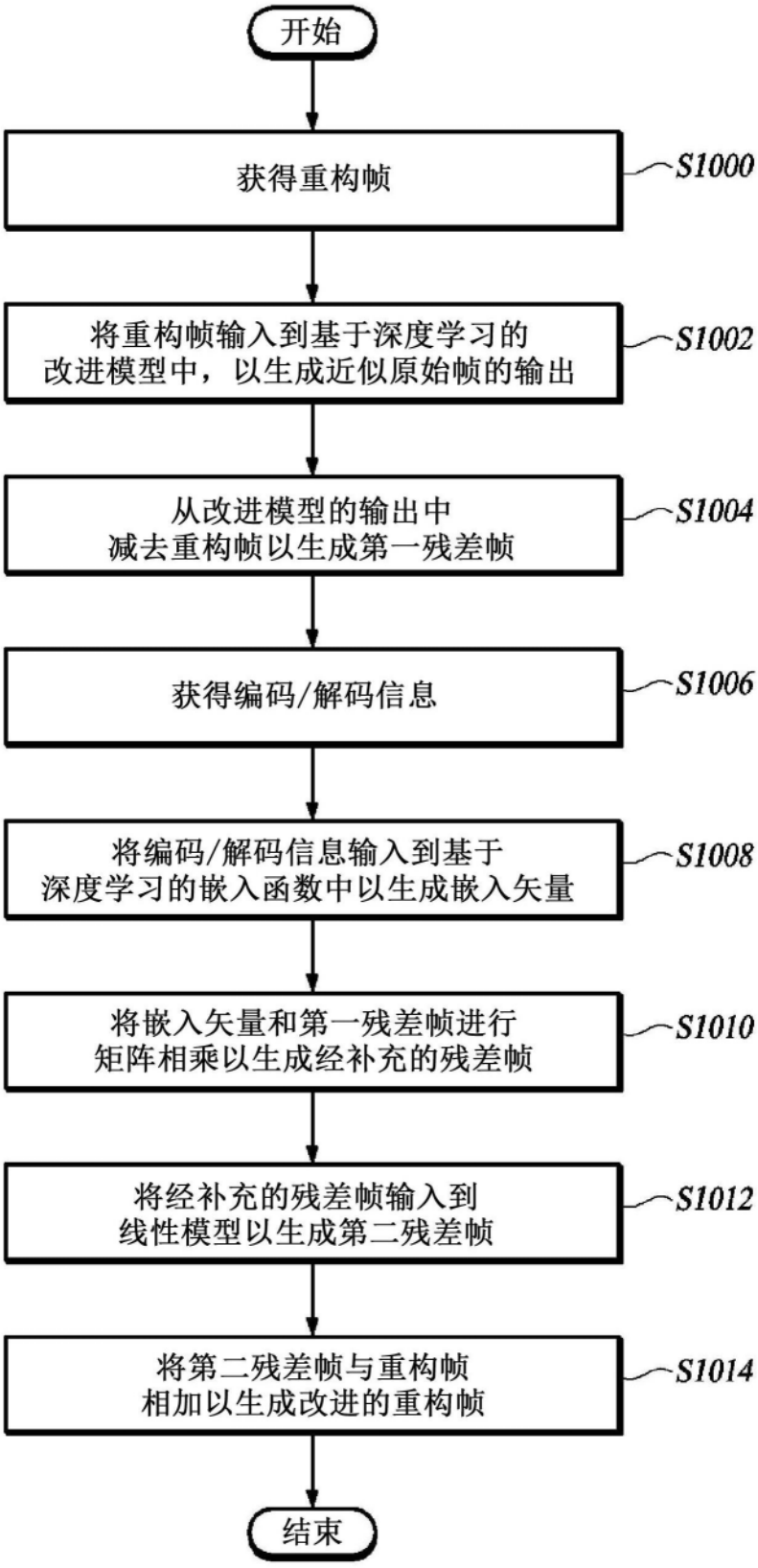

开始
获得重构帧
S1000
将重构帧输入到基于深度学习的
改进模型中，以生成近似原始帧的输出
S1002
从改进模型的输出中
减去重构帧以生成第一残差帧
S1004
获得编码/解码信息
S1006
将编码/解码信息输入到基于
深度学习的嵌入函数中以生成嵌入矢量
S1008
将嵌入矢量和第一残差帧进行
矩阵相乘以生成经补充的残差帧
S1010
将经补充的残差帧输入到
线性模型以生成第二残差帧
S1012
将第二残差帧与重构帧
相加以生成改进的重构帧
S1014
结束

图10