

图像编解码方法和装置

5 本申请要求于 2021 年 12 月 3 日提交中国专利局、申请号为 202111470979.5、申请名称为“图像编解码方法和装置”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及图像处理技术领域，尤其涉及图像编解码方法和装置。

10 背景技术

随着卷积神经网络（convolution neural network，CNN）在图像识别、目标检测等计算机视觉任务中的表现远超传统算法，越来越多的研究者开始探索基于深度学习的图像/视频压缩方法。一些研究者设计出端到端的深度学习图像/视频压缩算法，例如，将编码网络、熵估计网络、熵编码网络、熵解码网络、解码网络等模块作为一个整体同时优化，其中，15 编码网络和解码网络也可称为变换模块和逆变换模块，一般由卷积层和非线性变换单元组成。

非线性变换单元是图像/视频压缩网络的基础组件之一，其非线性特性的优劣直接影响了压缩算法的率失真性能。因此设计更高效的非线性变换单元是进一步提升图像/视频压缩算法中的率失真性能的关键。

20

发明内容

本申请实施例提供一种图像编解码方法和装置，可以实现编/解码网络中的高效的非线性变换处理，进一步提升图像/视频压缩算法中的率失真性能。

25 第一方面，本申请实施例提供一种图像编码方法，包括：获取待处理的第一图像特征；对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；根据所述经处理的图像特征进行编码得到码流。

30 第一图像特征是编码端在获取到待处理图像后，对其进行从图像域到特征域的转换后得到的，这里的转换可以包括但不限于：1、卷积处理，使用卷积层来进行提取特征，具有局部感受野，权重共享机制（即每一个滤波器滑动处理输入特征）。2、使用多层感知机（multi-layer perceptron，MLP）或全连接层提取特征，具有全局感受野特性，权重不共享。3、变换器（Transformer）处理，其中包括矩阵相乘、MLP 和归一化处理，具有全局感受野特性，捕捉长距离依赖能力强。

35 在一种可能的实现方式中，对第一图像特征中的每个特征值进行非线性运算得到第二图像特征；对第二图像特征进行卷积处理得到第三图像特征，第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到经处理的图像特征。

第一非线性运算是针对上述第一图像特征中的每个特征值进行的运算，可以包括取绝对值运算、线性整流函数（Rectified Linear Unit，ReLU）系列、Sigmoid、Tanh、分段线性

(piecewise linear, PWL) 运算等, 其中线性整流函数又称为修正线性单元。

经过第一非线性运算, 第一图像特征转换成第二图像特征, 第二图像特征和第一图像特征可以表示成矩阵的形式, 由于第一非线性运算是针对第一图像特征中的每个特征值的, 因此第一图像特征的每个特征值在第二图像特征值中均有一个特征值与其相对应, 因此第二图像特征对应的矩阵和第一图像特征对应的矩阵的尺寸相同, 且相同位置的特征值 (矩阵元素的值) 相对应, 例如, 第一图像特征表示成 3×3 的矩阵, 那么第二图像特征也可以表示成 3×3 的矩阵, 但由于第一图像特征和第二图像特征经过了第一非线性运算, 因此第一图像特征和第二图像特征中的特征值不完全相同, 相应的, 二者分别对应的矩阵中的元素值也不完全相同。

对第二图像特征进行卷积处理, 输出第三图像特征, 第三图像特征可以认为是第二图像特征的局部响应 (即校正后的值), 即第三图像特征是第二图像特征经过卷积处理得到的响应信号, 由于卷积处理的感受野是有限的, 经卷积处理输出的图像特征中的每个位置的响应值只和与该位置相邻位置的输入特征值有关, 因此称为局部响应。

由此可见, 经过上述非线性运算、卷积处理和逐点相乘运算实现了局部注意力机制, 局部是指非线性运算是逐点进行的, 输入的各个特征值只需要自身的特性即可得到输出值, 不用考虑周边特征值的影响; 注意力机制是指, 第一图像特征中的所有特征值, 有的是重要的, 有的是冗余的, 卷积处理的输出可以是图像特征中的各个特征值的权重, 可以对原特征值进行修正, 把重要的特征值突出, 把冗余的特征值抑制; 逐点相乘运算利用前述局部信息校正第一图像特征上每个特征值的值, 不必要求卷积参数必须是正数, 避免了对卷积参数的取值空间的限定, 可以在更广阔的取值空间中得到更优的卷积参数, 从而取得对图像更好的压缩性能。

在一种可能的实现方式中, 非线性变换处理还包括在逐点相乘运算之后的逐点相加运算。

在一种可能的实现方式中, 对第一图像特征中的每个特征值进行非线性运算得到第二图像特征; 对第二图像特征进行卷积处理得到第三图像特征, 第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应; 将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到第四图像特征, 第四图像特征中的多个特征值和第一图像特征中的多个特征值相对应; 将第一图像特征和第四图像特征中相对应的多个特征值进行逐点相加运算得到经处理的图像特征。

逐点相加运算是一个首位相加的残差结构, 可以使使用上述处理过程的编解码网络在训练时更容易收敛。卷积处理 $\text{conv1}(x)$ 和卷积处理 $\text{conv2}(x)$ 相似, 区别在于卷积处理 $\text{conv2}(x)$ 中的卷积的偏置参数 β 参数多加了 1。这样可以通过对卷积处理中的卷积中的偏置参数 β 的微调, 实现上述两种实施方式之间的转换, 即如果非线性变换单元包含卷积处理 $\text{conv1}(x)$ 和逐点相加运算, 则可以把两者融合成为卷积处理 $\text{conv2}(x)$, 从而省略逐点相加运算。得到经处理的图像特征后, 编码侧可以继续对其进行卷积处理, 也可以在卷积处理之后再次对卷积处理输出的特征进行非线性变换处理, 然后再对前述处理的结果特征进行编码以得到码流。

本申请实施例, 通过对编码网络中的非线性变换处理进行改变, 可以使输入的各个特征值只需要自身的特性即可得到输出值, 不用考虑周边特征值的影响, 并对原特征值进行

修正,把重要的特征值突出,把冗余的特征值抑制,此外还可以校正第一图像特征上每个特征值的值,避免了对卷积参数的限定,从而实现编码网络中的高效的非线性变换处理,进一步提升图像/视频压缩算法的率失真性能。

非线性运算是针对上述第一图像特征中的每个特征值进行的运算,可以包括采用分段线性映射方法,该方法可以是取绝对值(针对输入的特征值取其绝对值),也可以是修正线性单元(ReLU)或者泄漏修正线性单元(LeakyReLU),也可以是分段线性(PWL)运算。其中,ReLU是一种分段线性映射方法,针对输入的特征值,小于0的特征值输出为0,大于或等于0的特征值恒等输出;LeakyReLU是一种分段线性映射方法,其在ReLU的基础上,将小于0的输入特征值用预先设定的权重进行缩放,权重通常为0.01;PWL运算也是一种分段线性映射方法,PWL运算中的分段数可以更多,其具体定义请参照下文实施例部分。除此以外,非线性运算还可以包括其它方法,例如分段非线性运算、Tanh、Sigmoid,本申请实施例对此不做具体限定。

在一种可能的实现方式中,上述非线性变换处理还包括在上述卷积处理和逐点相乘运算之间的第二非线性运算,第二非线性运算和第一非线性运算相同或不同。例如,第一非线性运算可以是取绝对值运算,第二非线性运算可以仍然是取绝对值运算,第二非线性运算也可以是其他分段线性映射方法或其他非线性运算;或者,第一非线性运算可以是ReLU,第二非线性运算可以仍然是ReLU,第二非线性运算也可以是Sigmoid或其他非线性运算;或者,第一非线性运算可以是LeakyReLU,第二非线性运算可以仍然是LeakyReLU,第二非线性运算也可以是Tanh或其他非线性运算;或者,第一非线性运算可以是PWL,第二非线性运算可以仍然是PWL或其他非线性运算。当第二非线性运算使用PWL实现时,分段线性映射可以采用不同的分段数量;在各分段上的映射斜率可以通过训练的方式或者直接指定的方式确定;对输入特征图像的每个通道可以采用不同的分段线性函数,或者对所有通道都是用同一个分段线性函数,或者对若干个通道采用同一个分段线性函数进行处理。在这种实现方式下,模型训练完毕之后,残差结构不再与卷积融合,而是可与分段线性函数融合,即在原来的分段线性函数的输出上+1,组合成一个新的分段线性函数。

在一种可能的实现方式中,还可以构建训练阶段的非线性变换单元,所述训练阶段的非线性变换单元包括非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层;

根据预先获取的训练数据训练得到经训练的非线性变换单元,所述经训练的非线性变换单元用于实现所述非线性变换处理。

第二方面,本申请实施例提供一种图像解码方法,包括:获取待处理的第一图像特征;对所述第一图像特征进行非线性变换处理得到经处理的图像特征,所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算;根据所述经处理的图像特征获取重建图像。

在一种可能的实现方式中,所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征,包括:对所述第一图像特征中的每个特征值进行所述非线性运算得到第二图像特征;对所述第二图像特征进行卷积处理得到第三图像特征,所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应;将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

在一种可能的实现方式中,所述非线性变换处理还包括逐点相加运算。

在一种可能的实现方式中，所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征，包括：对所述第一图像特征中的每个特征值进行所述非线性运算得到第二图像特征；对所述第二图像特征进行卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性运算包括分段线性映射，例如 ReLU, LeakyReLU, PWL, Abs 等。在另一种可能的实现方式中，所述非线性运算包括连续函数，例如 Tanh, Sigmoid 等。在另一种可能的实现方式中，所述非线性运算包括分段的非线性运算。

上述第二方面及其可能的实现方式中提供的图像解码方法的技术效果可以参考上述第一方面及其对应的可能的实现方式中提供的图像编码方法的技术效果，此处不再赘述。

第三方面，本申请实施例提供一种编码装置，包括：获取模块，用于获取待处理的第一图像特征；变换模块，用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；编码模块，用于根据所述经处理的图像特征进行编码得到码流。

在一种可能的实现方式中，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

在一种可能的实现方式中，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

例如，第一非线性运算可以是取绝对值运算，第二非线性运算可以仍然是取绝对值运算，第二非线性运算也可以说分段线性映射或其他非线性运算。

在一种可能的实现方式中，所述非线性运算包括分段线性映射，例如 ReLU, LeakyReLU, PWL, Abs 等。在另一种可能的实现方式中，所述非线性运算包括连续函数，例如 Tanh, Sigmoid 等。在另一种可能的实现方式中，所述非线性运算包括分段的非线性

运算。

在一种可能的实现方式中,还包括:训练模块,用于构建训练阶段的非线性变换单元,所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层;根据预先获取的训练数据训练得到经训练的非线性变换单元,所述经训练的非线性变换单元用于实现所述非线性变换处理。

第四方面,本申请实施例提供一种解码装置,包括:获取模块,用于获取待处理的第一图像特征;变换模块,用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征,所述非线性变换处理包括第一非线性运算、卷积处理和逐点相乘运算;重建模块,用于根据所述经处理的图像特征获取重建图像。

在一种可能的实现方式中,所述变换模块,具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征;对所述第二图像特征进行所述卷积处理得到第三图像特征,所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应;将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

在一种可能的实现方式中,所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

在一种可能的实现方式中,所述变换模块,具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征;对所述第二图像特征进行所述卷积处理得到第三图像特征,所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应;将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征,所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应;将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

在一种可能的实现方式中,所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算,所述第二非线性运算和所述第一非线性运算相同或不同。

在一种可能的实现方式中,所述非线性运算包括分段线性映射,例如 ReLU, LeakyReLU, PWL, Abs 等。在另一种可能的实现方式中,所述非线性运算包括连续函数,例如 Tanh, Sigmoid 等。在另一种可能的实现方式中,所述非线性运算包括分段的非线性运算。

在一种可能的实现方式中,还包括:训练模块,用于构建训练阶段的非线性变换单元,所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层;根据预先获取的训练数据训练得到经训练的非线性变换单元,所述经训练的非线性变换单元用于实现所述非线性变换处理。

第五方面,本申请实施例提供一种编码器,包括:一个或多个处理器;非瞬时性计算机可读存储介质,耦合到所述处理器并存储由所述处理器执行的程序,其中所述程序在由所述处理器执行时,使得所述编码器执行上述第一方面中任一项所述的方法。

第六方面,本申请实施例提供一种解码器,包括:一个或多个处理器;非瞬时性计算机可读存储介质,耦合到所述处理器并存储由所述处理器执行的程序,其中所述程序在由所述处理器执行时,使得所述解码器执行上述第二方面中任一项所述的方法。

第七方面，本申请实施例提供一种包括程序代码的计算机程序产品，当所述程序代码在计算机或处理器上执行时，用于执行上述第一至二方面中任一项所述的方法。

第八方面，本申请实施例提供一种计算机可读存储介质，包括指令，当其在计算机上运行时，使得计算机执行上述第一至二方面中任一项所述的方法。

5 第九方面，本申请实施例提供一种码流，该码流由处理器执行第一方面任一项所述的方法所生成。

第十方面，本申请实施例提供一种存储码流的装置，该装置包括：接收器和至少一个存储介质，接收器用于接收码流；至少一个存储介质用于存储码流；码流为根据第一方面任一项所述的方法生成的码流。

10 第十一方面，本申请实施例提供一种传输码流的装置，该装置包括：发送器和至少一个存储介质，至少一个存储介质用于存储码流，码流包括处理器根据第一方面的任一项所述的方法生成的码流；发送器用于将码流发送给其他电子设备。

第十二方面，本申请实施例提供一种分发码流的系统，该系统包括：至少一个存储介质和流媒体设备，至少一个存储介质用于存储至少一个码流，至少一个码流包括根据第一方面的任意一种实现方式生成的码流；流媒体设备用于从至少一个存储介质中获取目标码流，并将目标码流发送给端侧设备，其中，流媒体设备包括内容服务器或内容分发服务器。

第十三方面，本申请实施例提供一种分发码流的系统，该系统包括：通信接口，用于接收用户获取目标码流的请求；处理器，用于响应于用户请求，确定目标码流的存储位置；通信接口还用于将目标码流的存储位置发送给用户，以使得用户从目标码流的存储位置获取目标码流，其中，目标码流由处理器执行第一方面任一项所述的方法所生成。

附图说明

- 图 1A 为示例性译码系统 10 的示意性框图；
- 图 1B 是视频译码系统 40 的实例的说明图；
- 25 图 2 为本发明实施例提供的视频译码设备 400 的示意图；
- 图 3 为示例性实施例提供的装置 500 的简化框图；
- 图 4 为端到端的深度学习图像编解码框架的示例图；
- 图 5 为端到端的深度学习视频编解码框架的示例图；
- 图 6 为本申请实施例的应用场景的示例图；
- 30 图 7 为本申请实施例的应用场景的示例图；
- 图 8 为本申请实施例的应用场景的示例图；
- 图 9 为本申请实施例的图像编码方法的过程 900 的流程图；
- 图 10a 为具备局部注意力机制的非线性变换单元的结构图；
- 图 10b 为具备局部注意力机制的残差非线性变换单元的结构图；
- 35 图 10c 为具备注意力机制的残差非线性变换单元的结构图；
- 图 10d 为具备注意力机制的非线性变换单元的结构图；
- 图 11 为 PWL 函数示意图；
- 图 12 为卷积处理的一个示意图；
- 图 13 为编码网络的结构图；

- 图 14 为本申请实施例的图像解码方法的过程 1300 的流程图；
图 15 为解码网络的结构图；
图 16a 为 ResAU 的一个示例性的结构图；
图 16b 为 ResAU 的一个示例性的结构图；
5 图 16c 为 ResAU 的一个示例性的结构图；
图 16d 为 ResAU 的一个示例性的结构图；
图 16e 为 ResAU 的一个示例性的结构图；
图 17a 为 ResAU 在 Kodak 测试集的 24 张图像上的整体表现；
图 17b 为 ResAU 在 Kodak 测试集的 24 张图像上的整体表现；
10 图 18 为本申请实施例编码装置 1700 的一个示例性的结构示意图；
图 19 为本申请实施例解码装置 1800 的一个示例性的结构示意图。

具体实施方式

- 为使本申请的目的、技术方案和优点更加清楚，下面将结合本申请中的附图，对本申
15 请中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本申请一部分实施例，
而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有作出创造性劳
动前提下所获得的所有其他实施例，都属于本申请保护的范围。

- 本申请的说明书实施例和权利要求书及附图中的术语“第一”、“第二”等仅用于区分描
述的目的，而不能理解为指示或暗示相对重要性，也不能理解为指示或暗示顺序。此外，
20 术语“包括”和“具有”以及他们的任何变形，意图在于覆盖不排他的包含，例如，包含了一
系列步骤或单元。方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元，而是
可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

- 应当理解，在本申请中，“至少一个（项）”是指一个或者多个，“多个”是指两个或两
个以上。“和/或”，用于描述关联对象的关联关系，表示可以存在三种关系，例如，“A 和/
25 或 B”可以表示：只存在 A，只存在 B 以及同时存在 A 和 B 三种情况，其中 A，B 可以是
单数或者复数。字符“/”一般表示前后关联对象是一种“或”的关系。“以下至少一项（个）”
或其类似表达，是指这些项中的任意组合，包括单项（个）或复数项（个）的任意组合。
例如，a，b 或 c 中的至少一项（个），可以表示：a，b，c，“a 和 b”，“a 和 c”，“b 和 c”，
或“a 和 b 和 c”，其中 a，b，c 可以是单个，也可以是多个。

- 30 以下是本申请实施例涉及到的术语说明：

- 1、码率：在图像压缩中，编码单位像素所需要的平均编码长度。
2、率失真性能：用于衡量压缩算法性能的指标，综合考虑了码率和解码图像失真度
两项数据。
3、注意力机制：利用有限的注意力资源从大量信息中筛选出高价值信息的手段。可
35 以使神经网络更多关注输入中的相关部分，更少关注不相关的部分。
4、残差结构：神经网络中的一种常用连接结构，计算方式可表示为 $H(x) = x + f(x)$ 。
该结构可防止网络深度增加时可能出现的梯度消失和梯度爆炸问题。
5、非线性变换单元：包含非线性运算（例如，ReLU、Sigmoid、Tanh、PWL 等运算）
的网络单元，该单元的整体计算方式不符合线性特性。

由于本申请实施例涉及神经网络的应用，为了便于理解，下面先对本申请实施例所使用到的相关名词或术语进行解释说明：

1、神经网络

神经网络（neural network, NN）是机器学习模型，神经网络可以由神经元组成的，神经元可以是指以 x_s 和截距 1 为输入的运算单元，该运算单元的输出可以为：

$$h_{w,b}(x) = f(W^T x) = f(\sum_{s=1}^n W_s x_s + b) \quad (1-1)$$

其中， $s=1, 2, \dots, n$ ， n 为大于 1 的自然数， W_s 为 x_s 的权重， b 为神经单元的偏置。 f 为神经单元的激活函数（activation functions），用于将非线性特性引入神经网络中，来将神经元中的输入信号转换为输出信号。该激活函数的输出信号可以作为下一层卷积层的输入。激活函数可以是 ReLU 等非线性函数。神经网络是将许多个上述单一的神经元联结在一起形成的网络，即一个神经元的输出可以是另一个神经元的输入。每个神经元的输入可以与前一层的局部感受野（local receptive field）相连，来提取局部感受野的特征，局部感受野可以由若干个神经元组成的区域。

2、多层感知器（multi-layer perception, MLP）

MLP 是一种简单的深度神经网络（deep neural network, DNN）（不同层之间是全连接的），也称多层神经网络，可以理解为具有很多层隐含层的神经网络，这里的“很多”并没有特别的度量标准。从 DNN 按不同层的位置划分，DNN 内部的神经网络可以分为三类：输入层，隐含层，输出层。一般来说第一层是输入层，最后一层是输出层，中间的层数都是隐含层。层与层之间是全连接的，也就是说，第 i 层的任意一个神经元一定与第 $i+1$ 层的任意一个神经元相连。虽然 DNN 看起来很复杂，但是就每一层的工作来说，其实并不复杂，简单来说就是如下线性关系表达式： $\vec{y} = \alpha(W\vec{x} + \vec{b})$ ，其中， $\vec{x}$ 是输入向量， $\vec{y}$ 是输出向量， $\vec{b}$ 是偏移向量， W 是权重矩阵（也称系数）， $\alpha()$ 是激活函数。每一层仅仅是对输入向量 $\vec{x}$ 经过如此简单的操作得到输出向量 $\vec{y}$ 。由于 DNN 层数多，则系数 W 和偏移向量 $\vec{b}$ 的数量也就很多了。这些参数在 DNN 中的定义如下所述：以系数 W 为例：假设在一个三层的 DNN 中，第二层的第 4 个神经元到第三层的第 2 个神经元的线性系数定义为 w_{24}^3 。上标 3 代表系数 W 所在的层数，而下标对应的是输出的第三层索引 2 和输入的第二层索引 4。总结就是：第 $L-1$ 层的第 k 个神经元到第 L 层的第 j 个神经元的系数定义为 w_{jk}^{L-1} 。需要注意的是，输入层是没有 W 参数的。在深度神经网络中，更多的隐含层让网络更能够刻画现实世界中的复杂情形。理论上而言，参数越多的模型复杂度越高，“容量”也就越大，也就意味着它能完成更复杂的学习任务。训练深度神经网络的也就是学习权重矩阵的过程，其最终目的是得到训练好的深度神经网络的所有层的权重矩阵（由很多层的向量 W 形成的权重矩阵）。

3、卷积神经网络

卷积神经网络（convolutional neuron network, CNN）是一种带有卷积结构的深度神经网络，是一种深度学习（deep learning）架构，深度学习架构是指通过机器学习的算法，在不同的抽象层级上进行多个层次的学习。作为一种深度学习架构，CNN 是一种前馈（feed-forward）人工神经网络，该前馈人工神经网络中的各个神经元可以对输入其中的图像作出响应。卷积神经网络包含了一个由卷积层和池化层构成的特征抽取器。该特征抽取器可以看作是滤波器，卷积过程可以看作是使用一个可训练的滤波器与一个输入的图像或者卷积

特征平面 (feature map) 做卷积。

卷积层是指卷积神经网络中对输入信号进行卷积处理的神经元层。卷积层可以包括很多个卷积算子，卷积算子也称为核，其在图像处理中的作用相当于一个从输入图像矩阵中提取特定信息的过滤器，卷积算子本质上可以是一个权重矩阵，这个权重矩阵通常被预先定义，在对图像进行卷积操作的过程中，权重矩阵通常在输入图像上沿着水平方向一个像素接着一个像素(或两个像素接着两个像素……这取决于步长 stride 的取值)的进行处理，从而完成从图像中提取特定特征的工作。该权重矩阵的大小应该与图像的大小相关，需要注意的是，权重矩阵的纵深维度 (depth dimension) 和输入图像的纵深维度是相同的，在进行卷积运算的过程中，权重矩阵会延伸到输入图像的整个深度。因此，和一个单一的权重矩阵进行卷积会产生一个单一纵深维度的卷积化输出，但是大多数情况下不使用单一权重矩阵，而是应用多个尺寸 (行×列) 相同的权重矩阵，即多个同型矩阵。每个权重矩阵的输出被堆叠起来形成卷积图像的纵深维度，这里的维度可以理解为由上面所述的“多个”来决定。不同的权重矩阵可以用来提取图像中不同的特征，例如一个权重矩阵用来提取图像边缘信息，另一个权重矩阵用来提取图像的特定颜色，又一个权重矩阵用来对图像中不需要的噪点进行模糊化等。该多个权重矩阵尺寸 (行×列) 相同，经过该多个尺寸相同的权重矩阵提取后的特征图的尺寸也相同，再将提取到的多个尺寸相同的特征图合并形成卷积运算的输出。这些权重矩阵中的权重值在实际应用中需要经过大量的训练得到，通过训练得到的权重值形成的各个权重矩阵可以用来从输入图像中提取信息，从而使得卷积神经网络进行正确的预测。当卷积神经网络有多个卷积层的时候，初始的卷积层往往提取较多的一般特征，该一般特征也可以称之为低级别的特征；随着卷积神经网络深度的加深，越往后的卷积层提取到的特征越来越复杂，比如高级别的语义之类的特征，语义越高的特征越适用于待解决的问题。

由于常常需要减少训练参数的数量，因此卷积层之后常常需要周期性的引入池化层，可以是一层卷积层后面跟一层池化层，也可以是多层卷积层后面接一层或多层池化层。在图像处理过程中，池化层的唯一目的就是减少图像的空间大小。池化层可以包括平均池化算子和/或最大池化算子，以用于对输入图像进行采样得到较小尺寸的图像。平均池化算子可以在特定范围内对图像中的像素值进行计算产生平均值作为平均池化的结果。最大池化算子可以在特定范围内取该范围内值最大的像素作为最大池化的结果。另外，就像卷积层中用权重矩阵的大小应该与图像尺寸相关一样，池化层中的运算符也应该与图像的大小相关。通过池化层处理后输出的图像尺寸可以小于输入池化层的图像的尺寸，池化层输出的图像中每个像素点表示输入池化层的图像的对应子区域的平均值或最大值。

在经过卷积层/池化层的处理后，卷积神经网络还不足以输出所需要的输出信息。因为如前所述，卷积层/池化层只会提取特征，并减少输入图像带来的参数。然而为了生成最终的输出信息 (所需要的类信息或其他相关信息)，卷积神经网络需要利用神经网络层来生成一个或者一组所需要的类的数量的输出。因此，在神经网络层中可以包括多层隐含层，该多层隐含层中所包含的参数可以根据具体的任务类型的相关训练数据进行预先训练得到，例如该任务类型可以包括图像识别，图像分类，图像超分辨率重建等等。

可选的，在神经网络层中的多层隐含层之后，还包括整个卷积神经网络的输出层，该输出层具有类似分类交叉熵的损失函数，具体用于计算预测误差，一旦整个卷积神经网络

的前向传播完成，反向传播就会开始更新前面提到的各层的权重值以及偏差，以减少卷积神经网络的损失，及卷积神经网络通过输出层输出的结果和理想结果之间的误差。

4、循环神经网络

循环神经网络（recurrent neural networks, RNN）是用来处理序列数据的。在传统的神经网络模型中，是从输入层到隐含层再到输出层，层与层之间是全连接的，而对于每一层内层之间的各个节点是无连接的。这种普通的神经网络虽然解决了很多难题，但是却仍然对很多问题却无能为力。例如，你要预测句子的下一个单词是什么，一般需要用前面的单词，因为一个句子中前后单词并不是独立的。RNN之所以称为循环神经网络，即一个序列当前的输出与前面的输出也有关。具体的表现形式为网络会对前面的信息进行记忆并应用于当前输出的计算中，即隐含层本层之间的节点不再无连接而是有连接的，并且隐含层的输入不仅包括输入层的输出还包括上一时刻隐含层的输出。理论上，RNN能够对任何长度的序列数据进行处理。对于RNN的训练和对传统的CNN或DNN的训练一样。同样使用误差反向传播算法，不过有一点区别：即，如果将RNN进行网络展开，那么其中的参数，如W，是共享的；而如上举例上述的传统神经网络却不是这样。并且在使用梯度下降算法中，每一步的输出不仅依赖当前步的网络，还依赖前面若干步网络的状态。该学习算法称为基于时间的反向传播算法（Back propagation Through Time, BPTT）。

既然已经有了卷积神经网络，为什么还要循环神经网络？原因很简单，在卷积神经网络中，有一个前提假设是：元素之间是相互独立的，输入与输出也是独立的，比如猫和狗。但现实世界中，很多元素都是相互连接的，比如股票随时间的变化，再比如一个人说了：我喜欢旅游，其中最喜欢的地方是云南，以后有机会一定要去。这里填空，人类应该都知道是填“云南”。因为人类会根据上下文的内容进行推断，但如何让机器做到这一步？RNN就应运而生了。RNN旨在让机器像人一样拥有记忆的能力。因此，RNN的输出就需要依赖当前的输入信息和历史的记忆信息。

5、损失函数

在训练深度神经网络的过程中，因为希望深度神经网络的输出尽可能的接近真正想要预测的值，所以可以通过比较当前网络的预测值和真正想要的目标值，再根据两者之间的差异情况来更新每一层神经网络的权重向量（当然，在第一次更新之前通常会有初始化的过程，即为深度神经网络中的各层预先配置参数），比如，如果网络的预测值高了，就调整权重向量让它预测低一些，不断的调整，直到深度神经网络能够预测出真正想要的目标值或与真正想要的目标值非常接近的值。因此，就需要预先定义“如何比较预测值和目标值之间的差异”，这便是损失函数（loss function）或目标函数（objective function），它们是用于衡量预测值和目标值的差异的重要方程。其中，以损失函数举例，损失函数的输出值（loss）越高表示差异越大，那么深度神经网络的训练就变成了尽可能缩小这个loss的过程。

6、反向传播算法

卷积神经网络可以采用误差反向传播（back propagation, BP）算法在训练过程中修正初始的超分辨率模型中参数的大小，使得超分辨率模型的重建误差损失越来越小。具体地，前向传递输入信号直至输出会产生误差损失，通过反向传播误差损失信息来更新初始的超分辨率模型中参数，从而使误差损失收敛。反向传播算法是以误差损失为主导的反向传播

运动，旨在得到最优的超分辨率模型的参数，例如权重矩阵。

7、生成式对抗网络

生成式对抗网络 (generative adversarial networks, GAN) 是一种深度学习模型。该模型中至少包括两个模块：一个模块是生成模型 (Generative Model)，另一个模块是判别模型 (Discriminative Model)，通过这两个模块互相博弈学习，从而产生更好的输出。生成模型和判别模型都可以是神经网络，具体可以是深度神经网络，或者卷积神经网络。GAN 的基本原理如下：以生成图片的 GAN 为例，假设有两个网络，G (Generator) 和 D (Discriminator)，其中 G 是一个生成图片的网络，它接收一个随机的噪声 z ，通过这个噪声生成图片，记做 $G(z)$ ；D 是一个判别网络，用于判别一张图片是不是“真实的”。它的输入参数是 x ， x 代表一张图片，输出 $D(x)$ 代表 x 为真实图片的概率，如果为 1，就代表 100% 是真实的图片，如果为 0，就代表不可能是真实的图片。在对该生成式对抗网络进行训练的过程中，生成网络 G 的目标就是尽可能生成真实的图片去欺骗判别网络 D，而判别网络 D 的目标就是尽量把 G 生成的图片和真实的图片区分开来。这样，G 和 D 就构成了一个动态的“博弈”过程，也即“生成式对抗网络”中的“对抗”。最后博弈的结果，在理想的状态下，G 可以生成足以“以假乱真”的图片 $G(z)$ ，而 D 难以判定 G 生成的图片究竟是不是真实的，即 $D(G(z))=0.5$ 。这样就得到了一个优异的生成模型 G，它可以用来生成图片。

随着卷积神经网络 (convolution neural network, CNN) 在图像识别、目标检测等计算机视觉任务中的表现远超传统算法，越来越多的研究者开始探索基于深度学习的图像/视频压缩方法。一些研究者设计出端到端的深度学习图像/视频压缩算法，例如，将编码网络、熵估计网络、熵编码网络、熵解码网络、解码网络等模块作为一个整体同时优化，其中，编码网络和解码网络也可称为变换模块和逆变换模块，一般由卷积层和非线性变换单元组成。

图 1A 为示例性译码系统 10 的示意性框图，例如可以利用本申请技术的视频译码系统 10 (或简称为译码系统 10)。视频译码系统 10 中的视频编码器 20 (或简称为编码器 20) 和视频解码器 30 (或简称为解码器 30) 代表可用于根据本申请中描述的各种示例执行各技术的设备等。

如图 1A 所示，译码系统 10 包括源设备 12，源设备 12 用于将编码图像等编码图像数据 21 提供给用于对编码图像数据 21 进行解码的目的设备 14。

源设备 12 包括编码器 20，另外即可选地，可包括图像源 16、图像预处理器等预处理器 (或预处理单元) 18、通信接口 (或通信单元) 22。

图像源 16 可包括或可以为任意类型的用于捕获现实世界图像等的图像捕获设备，和/或任意类型的图像生成设备，例如用于生成计算机动画图像的计算机图形处理器或任意类型的用于获取和/或提供现实世界图像、计算机生成图像 (例如，屏幕内容、虚拟现实 (virtual reality, VR) 图像和/或其任意组合 (例如增强现实 (augmented reality, AR) 图像) 的设备。所述图像源可以为存储上述图像中的任意图像的任意类型的内存或存储器。

为了区分预处理器 (或预处理单元) 18 执行的处理，图像 (或图像数据) 17 也可称为原始图像 (或原始图像数据) 17。

预处理器 18 用于接收 (原始) 图像数据 17，并对图像数据 17 进行预处理，得到预处

理图像（或预处理图像数据）19。例如，预处理器 18 执行的预处理可包括修剪、颜色格式转换（例如从 RGB 转换为 YCbCr）、调色或去噪。可以理解的是，预处理单元 18 可以为可选组件。

5 视频编码器（或编码器）20 用于接收预处理图像数据 19 并提供编码图像数据 21（下面将根据图 2 等进一步描述）。

源设备 12 中的通信接口 22 可用于：接收编码图像数据 21 并通过通信信道 13 向目的设备 14 等另一设备或任何其它设备发送编码图像数据 21（或其它任意处理后的版本），以便存储或直接重建。

10 目的设备 14 包括解码器 30，另外即可选地，可包括通信接口（或通信单元）28、后处理器（或后处理单元）32 和显示设备 34。

目的设备 14 中的通信接口 28 用于直接从源设备 12 或从存储设备等任意其它源设备接收编码图像数据 21（或其它任意处理后的版本），例如，存储设备为编码图像数据存储设备，并将编码图像数据 21 提供给解码器 30。

15 通信接口 22 和通信接口 28 可用于通过源设备 12 与目的设备 14 之间的直连通信链路，例如直接有线或无线连接等，或者通过任意类型的网络，例如有线网络、无线网络或其任意组合、任意类型的私网和公网或其任意类型的组合，发送或接收编码图像数据（或编码数据）21。

20 例如，通信接口 22 可用于将编码图像数据 21 封装为报文等合适的格式，和/或使用任意类型的传输编码或处理来处理所述编码后的图像数据，以便在通信链路或通信网络上进行传输。

通信接口 28 与通信接口 22 对应，例如，可用于接收传输数据，并使用任意类型的对应传输解码或处理和/或解封装对传输数据进行处理，得到编码图像数据 21。

25 通信接口 22 和通信接口 28 均可配置为如图 1A 中从源设备 12 指向目的设备 14 的对应通信信道 13 的箭头所指示的单向通信接口，或双向通信接口，并且可用于发送和接收消息等，以建立连接，确认并交换与通信链路和/或例如编码后的图像数据传输等数据传输相关的任何其它信息，等等。

视频解码器（或解码器）30 用于接收编码图像数据 21 并提供解码图像数据（或解码图像数据）31（下面将根据图 3 等进一步描述）。

30 后处理器 32 用于对解码后的图像等解码图像数据 31（也称为重建后的图像数据）进行后处理，得到后处理后的图像等后处理图像数据 33。后处理单元 32 执行的后处理可以包括例如颜色格式转换（例如从 YCbCr 转换为 RGB）、调色、修剪或重采样，或者用于产生供显示设备 34 等显示的解码图像数据 31 等任何其它处理。

35 显示设备 34 用于接收后处理图像数据 33，以向用户或观看者等显示图像。显示设备 34 可以为或包括任意类型的用于表示重建后图像的显示器，例如，集成或外部显示屏或显示器。例如，显示屏可包括液晶显示器（liquid crystal display, LCD）、有机发光二极管（organic light emitting diode, OLED）显示器、等离子显示器、投影仪、微型 LED 显示器、硅基液晶显示器（liquid crystal on silicon, LCoS）、数字光处理器（digital light processor, DLP）或任意类型的其它显示屏。

译码系统 10 还包括训练引擎 25，训练引擎 25 用于训练编码器 20 或解码器 30 以实

现图像域和特征域的转变。

本申请实施例中训练数据可以存入数据库（未示意）中，训练引擎 25 基于训练数据训练得到编/解码网络。需要说明的是，本申请实施例对于训练数据的来源不做限定，例如可以从云端或其他地方获取训练数据进行模型训练。

5 尽管图 1A 示出了源设备 12 和目的设备 14 作为独立的设备，但设备实施例也可以同时包括源设备 12 和目的设备 14 或同时包括源设备 12 和目的设备 14 的功能，即同时包括源设备 12 或对应功能和目的设备 14 或对应功能。在这些实施例中，源设备 12 或对应功能和目的设备 14 或对应功能可以使用相同硬件和/或软件或通过单独的硬件和/或软件或其任意组合来实现。

10 根据描述，图 1A 所示的源设备 12 和/或目的设备 14 中的不同单元或功能的存在和（准确）划分可能根据实际设备和应用而有所不同，这对技术人员来说是显而易见的。

编码器 20（例如视频编码器 20）或解码器 30（例如视频解码器 30）或两者都可通过如图 1B 所示的处理电路实现，例如一个或多个微处理器、数字信号处理器（digital signal processor, DSP）、专用集成电路（application-specific integrated circuit, ASIC）、现场可编程门阵列（field-programmable gate array, FPGA）、离散逻辑、硬件、视频编码专用处理器或其任意组合。编码器 20 可以通过处理电路 46 实现，以包含参照图 2 编码器 20 论述的各种模块和/或本文描述的任何其它编码器系统或子系统。解码器 30 可以通过处理电路 46 实现，以包含参照图 3 解码器 30 论述的各种模块和/或本文描述的任何其它解码器系统或子系统。所述处理电路 46 可用于执行下文论述的各种操作。如图 8 所示，如果部分技术在软件中实施，则设备可以将软件的指令存储在合适的非瞬时性计算机可读存储介质中，并且使用一个或多个处理器在硬件中执行指令，从而执行本发明技术。视频编码器 20 和视频解码器 30 中的其中一个可作为组合编解码器（encoder/decoder, CODEC）的一部分集成在单个设备中，如图 1B 所示。

25 源设备 12 和目的设备 14 可包括各种设备中的任一种，包括任意类型的手持设备或固定设备，例如，笔记本电脑或膝上型电脑、手机、智能手机、平板或平板电脑、相机、台式计算机、机顶盒、电视机、显示设备、数字媒体播放器、视频游戏控制台、视频流设备（例如，内容业务服务器或内容分发服务器）、广播接收设备、广播发射设备，等等，并可以不使用或使用任意类型的操作系统。在一些情况下，源设备 12 和目的设备 14 可配备用于无线通信的组件。因此，源设备 12 和目的设备 14 可以是无线通信设备。

30 在一些情况下，图 1A 所示的视频译码系统 10 仅仅是示例性的，本申请提供的技术可适用于视频编码设置（例如，视频编码或视频解码），这些设置不一定包括编码设备与解码设备之间的任何数据通信。在其它示例中，数据从本地存储器中检索，通过网络发送，等等。视频编码设备可以对数据进行编码并将数据存储在存储器中，和/或视频解码设备可以从存储器中检索数据并对数据进行解码。在一些示例中，编码和解码由相互不通信而只是编码数据到存储器和/或从存储器中检索并解码数据的设备来执行。

35 图 1B 是视频译码系统 40 的实例的说明图。视频译码系统 40 可以包含成像设备 41、视频编码器 20、视频解码器 30（和/或藉由处理电路 46 实施的视频编/解码器）、天线 42、一个或多个处理器 43、一个或多个内存存储器 44 和/或显示设备 45。

如图 1B 所示，成像设备 41、天线 42、处理电路 46、视频编码器 20、视频解码器 30、

处理器 43、内存存储器 44 和/或显示设备 45 能够互相通信。在不同实例中，视频译码系统 40 可以只包含视频编码器 20 或只包含视频解码器 30。

5 在一些实例中，天线 42 可以用于传输或接收视频数据的经编码比特流。另外，在一些实例中，显示设备 45 可以用于呈现视频数据。处理电路 46 可以包含专用集成电路（application-specific integrated circuit, ASIC）逻辑、图形处理器、通用处理器等。视频译码系统 40 也可以包含可选的处理器 43，该可选处理器 43 类似地可以包含专用集成电路（application-specific integrated circuit, ASIC）逻辑、图形处理器、通用处理器等。另外，内存存储器 44 可以是任何类型的存储器，例如易失性存储器（例如，静态随机存取存储器（static random access memory, SRAM）、动态随机存储器（dynamic random access memory, DRAM）等）或非易失性存储器（例如，闪存等）等。在非限制性实例中，内存存储器 44 10 可以由超速缓存内存实施。在其它实例中，处理电路 46 可以包含存储器（例如，缓存等）用于实施图像缓冲器等。

在一些实例中，通过逻辑电路实施的视频编码器 20 可以包含（例如，通过处理电路 46 或内存存储器 44 实施的）图像缓冲器和（例如，通过处理电路 46 实施的）图形处理单元。图形处理单元可以通信耦合至图像缓冲器。图形处理单元可以包含通过处理电路 46 15 实施的视频编码器 20，以实施参照图 2 和/或本文中所描述的任何其它编码器系统或子系统所论述的各种模块。逻辑电路可以用于执行本文所论述的各种操作。

在一些实例中，视频解码器 30 可以以类似方式通过处理电路 46 实施，以实施参照图 3 的视频解码器 30 和/或本文中所描述的任何其它解码器系统或子系统所论述的各种模块。20 在一些实例中，逻辑电路实施的视频解码器 30 可以包含（通过处理电路 46 或内存存储器 44 实施的）图像缓冲器和（例如，通过处理电路 46 实施的）图形处理单元。图形处理单元可以通信耦合至图像缓冲器。图形处理单元可以包含通过处理电路 46 实施的视频解码器 30，以实施参照图 3 和/或本文中所描述的任何其它解码器系统或子系统所论述的各种模块。

25 在一些实例中，天线 42 可以用于接收视频数据的经编码比特流。如所论述，经编码比特流可以包含本文所论述的与编码视频帧相关的数据、指示符、索引值、模式选择数据等，例如与编码分割相关的数据（例如，变换系数或经量化变换系数，（如所论述的）可选指示符，和/或定义编码分割的数据）。视频译码系统 40 还可包含耦合至天线 42 并用于解码经编码比特流的视频解码器 30。显示设备 45 用于呈现视频帧。

30 应理解，本申请实施例中对于参考视频编码器 20 所描述的实例，视频解码器 30 可以用于执行相反过程。关于信令语法元素，视频解码器 30 可以用于接收并解析这种语法元素，相应地解码相关视频数据。在一些例子中，视频编码器 20 可以将语法元素熵编码成经编码视频比特流。在此类实例中，视频解码器 30 可以解析这种语法元素，并相应地解码相关视频数据。

35 为便于描述，参考通用视频编码（Versatile video coding, VVC）参考软件或由 ITU-T 视频编码专家组（Video Coding Experts Group, VCEG）和 ISO/IEC 运动图像专家组（Motion Picture Experts Group, MPEG）的视频编码联合工作组（Joint Collaboration Team on Video Coding, JCT-VC）开发的高性能视频编码（High-Efficiency Video Coding, HEVC）描述本发明实施例。本领域普通技术人员理解本发明实施例不限于 HEVC 或 VVC。

图 2 为本发明实施例提供的视频译码设备 400 的示意图。视频译码设备 400 适用于实现本文描述的公开实施例。在一个实施例中，视频译码设备 400 可以是解码器，例如图 1A 中的视频解码器 30，也可以是编码器，例如图 1A 中的视频编码器 20。

视频译码设备 400 包括：用于接收数据的入端口 410（或输入端口 410）和接收单元（receiver unit, Rx）420；用于处理数据的处理器、逻辑单元或中央处理器（central processing unit, CPU）430；例如，这里的处理器 430 可以是神经网络处理器 430；用于传输数据的发送单元（transmitter unit, Tx）440 和出端口 450（或输出端口 450）；用于存储数据的存储器 460。视频译码设备 400 还可包括耦合到入端口 410、接收单元 420、发送单元 440 和出端口 450 的光电（optical-to-electrical, OE）组件和电光（electrical-to-optical, EO）组件，用于光信号或电信号的出口或入口。

处理器 430 通过硬件和软件实现。处理器 430 可实现为一个或多个处理器芯片、核（例如，多核处理器）、FPGA、ASIC 和 DSP。处理器 430 与入端口 410、接收单元 420、发送单元 440、出端口 450 和存储器 460 通信。处理器 430 包括译码模块 470（例如，基于神经网络 NN 的译码模块 470）。译码模块 470 实施上文所公开的实施例。例如，译码模块 470 执行、处理、准备或提供各种编码操作。因此，通过译码模块 470 为视频译码设备 400 的功能提供了实质性的改进，并且影响了视频译码设备 400 到不同状态的切换。或者，以存储在存储器 460 中并由处理器 430 执行的指令来实现译码模块 470。

存储器 460 包括一个或多个磁盘、磁带机和固态硬盘，可以用作溢出数据存储设备，用于在选择执行程序时存储此类程序，并且存储在程序执行过程中读取的指令和数据。存储器 460 可以是易失性和/或非易失性的，可以是只读存储器（read-only memory, ROM）、随机存取存储器（random access memory, RAM）、三态内容寻址存储器（ternary content-addressable memory, TCAM）和/或静态随机存取存储器（static random-access memory, SRAM）。

图 3 为示例性实施例提供的装置 500 的简化框图，装置 500 可用作图 1A 中的源设备 12 和目的设备 14 中的任一个或两个。

装置 500 中的处理器 502 可以是中央处理器。或者，处理器 502 可以是现有的或今后将研发出的能够操控或处理信息的任何其它类型设备或多个设备。虽然可以使用如图所示的处理器 502 等单个处理器来实施已公开的实现方式，但使用一个以上的处理器速度更快和效率更高。

在一种实现方式中，装置 500 中的存储器 504 可以是只读存储器（ROM）设备或随机存取存储器（RAM）设备。任何其它合适类型的存储设备都可以用作存储器 504。存储器 504 可以包括处理器 502 通过总线 512 访问的代码和数据 506。存储器 504 还可包括操作系统 508 和应用程序 510，应用程序 510 包括允许处理器 502 执行本文所述方法的至少一个程序。例如，应用程序 510 可以包括应用 1 至 N，还包括执行本文所述方法的视频译码应用。

装置 500 还可以包括一个或多个输出设备，例如显示器 518。在一个示例中，显示器 518 可以是将显示器与可用于感测触摸输入的触敏元件组合的触敏显示器。显示器 518 可以通过总线 512 耦合到处理器 502。

虽然装置 500 中的总线 512 在本文中描述为单个总线，但是总线 512 可以包括多个总

线。此外，辅助储存器可以直接耦合到装置 500 的其它组件或通过网络访问，并且可以包括存储卡等单个集成单元或多个存储卡等多个单元。因此，装置 500 可以具有各种各样的配置。

图 4 为端到端的深度学习图像编解码框架的示例图，如图 4 所示，该图像编解码框架包括编码端：编码网络（Encoder）、量化模块、熵编码网络；解码端：熵解码网络、解码网络（Decoder），以及熵估计网络。

在编码端，原图经过编码网络的处理从图像域变换到特征域，变换后的图像特征经过量化模块和熵编码网络的处理被编码成待传输或待存储的码流。在解码端，码流经过熵解码网络的处理被解码成图像特征，图像特征经过解码网络的处理从特征域变换到图像域，从而得到重建图像。熵估计网络根据图像特征估计得到每个特征元素的估计概率值，以用于熵编码网络和熵解码网络的处理。

本实施例中，编码网络（Encoder）和解码网络（Decoder）中都具有非线性变换单元。

图 5 为端到端的深度学习视频编解码框架的示例图，如图 5 所示，该视频编解码框架包括预测模块（predict model）和残差压缩（residual compress）模块，

预测模块利用前一帧的重建图像对当前帧进行预测得到预测图像，残差压缩模块一方面对当前帧的原始图像和预测图像之间的残差进行压缩，另一方面解压缩得到重建残差，重建残差和预测图像求和得到当前帧的重建图像。预测模块和残差压缩模块中的编码子网络和解码子网络中均具有非线性变换单元。

本实施例中，预测模块（predict model）和残差压缩（residual compress）模块中都具有非线性变换单元。

图 6 为本申请实施例的应用场景的示例图，如图 6 所示，该应用场景可以是终端、云服务器、视频监控中涉及图像/视频采集、存储或传输的业务，例如，终端拍照/录像、相册、云相册、视频监控等。

编码端：摄像头（Camera）采集图像/视频。人工智能（artificial intelligence, AI）图像/视频编码网络对图像/视频进行从特征提取得到冗余度较低的图像特征，进而基于图像特征进行压缩得到码流/图像文件。

解码端：当需要输出图像/视频时，AI 图像/视频解码网络对码流/图像文件进行解压得到图像特征，再对图像特征进行反特征提取得到重建的图像/视频。

存储/传输模块将压缩得到的码流/图像文件针对不同业务进行存储（例如，终端拍照、视频监控、云服务器等）或传输（例如，云服务，直播技术等）。

图 7 为本申请实施例的应用场景的示例图，如图 7 所示，该应用场景可以是终端、视频监控中涉及图像/视频采集、存储或传输的业务，例如，终端相册、视频监控、直播等。

编码端：编码网络将图像/视频变换成冗余度更低的图像特征，其通常包含非线性变换单元，具有非线性特性。熵估计网络负责计算图像特征中各个数据的编码概率。熵编码网络根据各个数据对应的概率，对图像特征进行无损编码得到码流/图像文件，进一步降低图像压缩过程中的数据传输量。

解码端：熵解码网络根据各个数据对应的概率，对码流/图像文件进行无损解码得到重建的图像特征。解码网络对熵解码输出的图像特征进行反变换，将其解析为图像/视频。和编码网络对应，其通常包含非线性变换单元，具有非线性特性。保存模块将码流/图像文件

保存至终端的对应存储位置。加载模块从终端的对应存储位置加载码流/图像文件，并输入到熵解码网络。

图 8 为本申请实施例的应用场景的示例图，如图 8 所示，该应用场景可以是云、视频监控中涉及图像/视频采集、存储或传输的业务，例如，云相册、视频监控、直播等。

5 编码端：本地获取图像/视频，对其进行图像（JPEG）编码得到压缩图像/视频，之后向云端发送压缩图像/视频。云端对压缩图像/视频进行 JPEG 解码得到图像/视频，之后对图像/视频进行压缩得到码流/图像文件并存储。

10 解码端：当本地需要从云端获取图像/视频时，云端对码流/图像文件进行解压缩得到图像/视频，之后对图像/视频进行 JPEG 编码，得到压缩图像/视频，向本地发送压缩图像/视频。本地对压缩图像/视频进行 JPEG 解码得到图像/视频。云端的结构以及各个模块的用途可以参考图 7 的结构以及各个模块的用途，本申请实施例在此不做赘述。

基于上述编/解码网络和应用场景，本申请实施例提供了一种图像编/解码方法，以实现高效的非线性变换处理，提升图像/视频压缩算法中的率失真性能。

15 图 9 为本申请实施例的图像编码方法的过程 900 的流程图。过程 900 可由上述实施例中的编码端执行。过程 900 描述为一系列的步骤或操作，应当理解的是，过程 900 可以以各种顺序执行和/或同时发生，不限于图 9 所示的执行顺序。过程 900 包括如下步骤：

步骤 901、获取待处理的第一图像特征。

20 第一图像特征是编码端在获取到待处理图像后，对其进行从图像域到特征域的转换后得到的，这里的转换可以包括：1、卷积处理，使用卷积层来进行提取特征，具有局部感受野，权重共享机制（即每一个滤波器滑动处理输入特征）。2、使用 MLP 或全连接层提取特征，具有全局感受野特性，权重不共享。3、变换器（Transformer）处理，其中包括矩阵相乘、MLP 和归一化处理，具有全局感受野特性，捕捉长距离依赖能力强。

25 第一图像特征可以表示成二维矩阵（ $L \times C$ ， L 表示长度， C 表示通道（channel））或三维矩阵（ $C \times H \times W$ ， C 表示通道数， H 表示高度， W 表示宽度）的形式，具体的形式与前述转换方式相关联，例如，采用卷积处理或者 MLP 提取得到的第一图像特征一般对应于三维矩阵，而采用变换器处理得到的第一图像特征一般对应于二维矩阵。

例如，第一图像特征表示为一个二维矩阵：

$$A = \begin{bmatrix} a_{0,2} & a_{1,2} & a_{2,2} \\ a_{0,1} & a_{1,1} & a_{2,1} \\ a_{0,0} & a_{1,0} & a_{2,0} \end{bmatrix}$$

30 该二维矩阵 A 是 3×3 的矩阵，包含 9 个元素，每个元素 $a(i,j)$ 对应第一图像特征的一个特征值，其中， i 表示元素 $a(i,j)$ 对应的长度， j 表示元素 $a(i,j)$ 所在的通道。

又例如，第一图像特征表示为一个三维矩阵：

$$B = \begin{bmatrix} b_{0,2,0} & b_{1,2,0} & b_{2,2,0} \\ b_{0,1,0} & b_{1,1,0} & b_{2,1,0} \\ b_{0,0,0} & b_{1,0,0} & b_{2,0,0} \end{bmatrix}, \begin{bmatrix} b_{0,2,1} & b_{1,2,1} & b_{2,2,1} \\ b_{0,1,1} & b_{1,1,1} & b_{2,1,1} \\ b_{0,0,1} & b_{1,0,1} & b_{2,0,1} \end{bmatrix}$$

35 该三维矩阵 B 是 $3 \times 3 \times 2$ 的矩阵，包含 18 个元素，每个元素 $a(i,j,l)$ 对应第一图像特征的一个特征值，其中， i 表示元素 $a(i,j,l)$ 所在的行， j 表示元素 $a(i,j,l)$ 所在的列， l 表示元素 $a(i,j,l)$ 所在的通道。

需要说明的是，本申请实施例对第一图像特征的获取方式不做具体限定。

待处理图像可以是一张图片，也可以是视频中的一帧图像帧，还可以是前述图片或图像帧分割得到的图像块，对此不做具体限定。

步骤 902、对第一图像特征进行非线性变换处理得到经处理的图像特征。

- 5 在一种可能的实现方式中，对第一图像特征中的每个特征值进行第一非线性运算得到第二图像特征；对第二图像特征进行卷积处理得到第三图像特征，第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到经处理的图像特征。

- 10 图 10a 为具备注意力机制的非线性变换单元的结构图，如图 10a 所示，本申请实施例中，非线性变换单元用于实现上述非线性变换处理，其包括第一非线性运算、卷积处理和逐点相乘运算。

第一非线性运算是针对上述第一图像特征中的每个特征值进行的运算，可以包括取绝对值运算、ReLU 系列、Sigmoid、Tanh、PWL 运算等。其中，

取绝对值是指针对输入的特征值取其绝对值。可以用如下公式表示：

$$y = |x| = \begin{cases} x, x \geq 0 \\ -x, x < 0 \end{cases}$$

- 15 分段线性映射包括修正线性单元 (ReLU)，或者泄漏修正线性单元 (LeakyReLU)，或者 PWL 运算。其中，ReLU 是一种分段线性映射方法，针对输入的特征值，小于 0 的特征值输出为 0，大于或等于 0 的特征值恒等输出。可以用如下公式表示：

$$y = \text{Relu}(x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases}$$

- 20 LeakyReLU 是一种分段线性映射方法，其在 ReLU 的基础上，将小于 0 的输入特征值用预先设定的权重进行缩放，权重通常为 0.01。可以用如下公式表示：

$$y = \text{LeakyReLU}(x) = \begin{cases} x, x \geq 0 \\ ax, x < 0 \end{cases}$$

其中，a 表示预先设定值，通常设置为 0.01。

Sigmoid 可以表示为如下运算：

$$y = \text{Sigmoid}(x) = \frac{1}{1 + e^{-x}}$$

- 25 Tanh 可以表示为如下运算：

$$y = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

PWL 可以表示为如下运算：

$$PWL_N(x, B_L, B_R, Y_P, K_L, K_R) = \begin{cases} (x - B_L) * K_L + Y_P^0, x < B_L \\ (x - B_R) * K_R + Y_P^N, x \geq B_R \\ (x - B_{idx}) * K_{idx} + Y_P^{idx}, B_L \leq x < B_R \end{cases} \quad (1)$$

- 30 其中，N 表示分段数； B_L 表示左边界； B_R 表示右边界； Y_P 表示 N+1 个分界点对应的 y 轴坐标值； K_L 表示最左斜率； K_R 表示最右斜率；idx 表示 x 所属分段的索引号； B_{idx} 和 K_{idx} 是该分段的相应左边界和斜率。 $d = \frac{B_R - B_L}{N}$ 表示分段长度，这些值的计算如下：

$$idx = \left\lfloor \frac{x - B_L}{d} \right\rfloor \quad (2)$$

$$B_{idx} = B_L + idx * d \quad (3)$$

$$K_{idx} = \frac{y_P^{idx+1} - y_P^{idx}}{d} \quad (4)$$

需要说明的是，公式(1)可用于不同粒度(按层甚至按通道均可)。按通道应用时， B_L, B_R, Y_P, K_L, K_R 是逐通道参数。可选的，可以在所有的PWL中使用相同的超参N。

5 PWL的优点有：作为一个通用逼近器，PWL可以逼近任何连续的有界标量函数；PWL随参数(超参N除外)连续变化，非常利于基于梯度优化；灵活性集中在有界区域内，因此可以最大限度地利用可学习参数；由于分段均匀，PWL计算效率高，尤其是在推理方面。

10 图11为PWL函数示意图，如图11所示，分段数N是一个超参，影响PWL函数的拟合能力。分段数越大，自由度越高，模型容量也越大。左右边界 B_L 和 B_R 限定了PWL关注的主要有效区域。 $[B_L, B_R]$ 被均匀划分为N个分段，得到N+1个分界点。每个分界点都有对应的y轴坐标值 Y_P ，这些坐标就决定了PWL的曲线形状。 $[B_L, B_R]$ 之外的区域，可以用两个斜率 K_L 和 K_R 来控制边界外的区域形状。

需要说明的是，分段线性映射还可以采用ReLU的其它变形方式，也可以采用其它新的第一非线性运算方式，本申请实施例对此均不作具体限定。

15 经过上述第一非线性运算，第一图像特征转换成第二图像特征，第二图像特征也可以如第一图像特征表示成矩阵的形式，由于第一非线性运算是针对第一图像特征中的每个特征值的，因此第一图像特征的每个特征值在第二图像特征值中均有一个特征值与其相对应，因此第二图像特征对应的矩阵和第一图像特征对应的矩阵的尺寸相同，且相同位置的特征值(矩阵元素的值)相对应，例如，第一图像特征表示成上述矩阵A那样的3×3的矩阵，
20 那么第二图像特征也可以表示成3×3的矩阵，但由于第一图像特征和第二图像特征经过了第一非线性运算，因此第一图像特征和第二图像特征中的特征值不完全相同，相应的，二者分别对应的矩阵中的元素值也不完全相同。

对第二图像特征进行卷积处理，输出第三图像特征，第三图像特征可以认为是第二图像特征的局部响应(即校正后的值)，即第三图像特征是第二图像特征经过卷积处理得到的响应信号，由于卷积处理的感受野是有限的，经卷积处理输出的图像特征中的每个位置的响应值只和与该位置相邻位置的输入特征值有关，因此称为局部响应。卷积处理可以表示为如下公式：

$$\text{conv1}(x) = \beta + \sum \gamma \times x$$

其中， γ 表示卷积层的权重， β 表示卷积层的偏置参数。

30 经过卷积处理，第三图像特征对应的矩阵和第一图像特征对应的矩阵的尺寸也相同。图12为卷积处理的一个示意图，如图12所示，输入一个1×2的矩阵，经过卷积层的处理输出1×4的矩阵，该卷积层包括两个滤波器，一个是2×50的矩阵W1，另一个是50×4的矩阵W2。先由矩阵W1对输入矩阵进行卷积运算，得到1×50的矩阵，然后由矩阵W2对该矩阵进行卷积运算得到1×4的输出矩阵。

35 将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算，即为将非线性变换单元的初始输入和卷积处理的输出进行逐点相乘，可以表示为如下公式：

$$c(i, j) = a(i, j) \times b(i, j)$$

其中，(i, j)表示特征值在其所在图像特征中的索引，a(i, j)表示第一图像特征的特征值，

$b(i,j)$ 表示第二图像特征的特征值， $c(i,j)$ 表示经处理的图像特征的特征值。

由此可见，经处理的图像特征对应的矩阵和第一图像特征对应的矩阵的尺寸也相同。

需要说明的是，本申请实施例中，两个图形特征中的多个特征值相对应可以是指该两个图像特征分别表示为矩阵后，两个矩阵中的相同位置上的元素的值具有运算关系，该二者相对应。例如，第一图像特征和第三图像特征均表示成上述矩阵 A 的形式，同位于 $a_{0,2}$ 的位置的元素相对应。

此外，卷积处理也可以得到与第一图像特征尺寸不同的第三图像特征，这取决于卷积处理的内部结构，尤其是在进行卷积处理时的滤波器的长、宽以及通道数，如果第三图像特征和第一图像特征的尺寸不同，相应的第三图像特征对应的矩阵中的元素和第一图像特征对应的矩阵中的元素也不是一一对应的，此时可以考虑第一图像特征对应的矩阵中的多个元素与第三图像特征对应的矩阵中的同一个元素相乘。例如，第一图像特征对应的矩阵的通道数为 3，而第三图像特征对应的矩阵的通道数为 1，可以将该第三图像特征对应的矩阵中的元素与第一图像特征对应的矩阵中各个通道中的同一位置的元素分别相乘。本申请实施例对此不做具体限定。

上述第一非线性运算、卷积处理和逐点相乘运算可以表示为如下公式 (1)：

$$y_i = x_i \times (\beta_i + \sum_j \gamma_{ij} \phi(x_j)) \quad (1)$$

其中， x 表示输入的特征值， y 表示输出的特征值， $\phi(x_j)$ 表示第一非线性运算， γ 表示卷积层的权重， β 表示卷积层的偏置参数。

由于上述公式中采用了相乘运算，相较于相关技术 (GDN 除法归一化) 的卷积处理中， $y_i = \frac{x_i}{\sqrt{\beta_i + \sum_j \gamma_{ij} x_j^2}}$ 的除法运算，一方面，不必为了迁就开平方根必须是正数的限定，而要求卷积参数 γ 和 β 必须是正数，另一方面，不必为了迁就除法要求分母不能为 0 的限定，也对卷积参数 γ 和 β 构成了取值上的限定。

经过上述第一非线性运算、卷积处理和逐点相乘运算实现了局部注意力机制，局部是指第一非线性运算是逐点进行的，输入的各个特征值只需要自身的特性即可得到输出值，不用考虑周边特征值的影响；注意力机制是指，第一图像特征中的所有特征值，有的是重要的，有的是冗余的，卷积处理的输出可以是图像特征中的各个特征值的权重，可以对原特征值进行修正，把重要的特征值突出，把冗余的特征值抑制；逐点相乘运算利用前述局部信息校正第一图像特征上每个特征值的值，避免了对卷积参数的限定。

在一种可能的实现方式中，对第一图像特征中的每个特征值进行第一非线性运算得到第二图像特征；对第二图像特征进行卷积处理得到第三图像特征，第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到第四图像特征，第四图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第四图像特征中相对应的多个特征值进行逐点相加运算得到经处理的图像特征。

图 10b 为具备注意力机制的残差非线性变换单元的结构图，如图 10b 所示，本申请实施例中，非线性变换单元用于实现上述非线性变换处理，其包括第一非线性运算、卷积处理、逐点相乘运算和逐点相加运算。

本申请实施例中，第一非线性运算、卷积处理和逐点相乘运算均可以参照上一种实施

方式中的描述，此处不再赘述。

得到第四图像特征后，再将第一图像特征和第四图像特征中相对应的多个特征值进行逐点相加运算，即为将非线性变换单元的初始输入和逐点相乘运算的输出进行逐点相加，可以表示为如下公式：

$$\text{sum}(i, j) = a(i, j) + c(i, j) \quad (1)$$

其中， (i, j) 表示特征值在其所在图像特征中的索引， $a(i, j)$ 表示第一图像特征的特征值， $c(i, j)$ 表示第四图像特征的特征值， $\text{sum}(i, j)$ 表示经处理的图像特征的特征值。

逐点相加运算是一个首位相加的残差结构，可以使使用上述处理过程的编解码网络在训练时更容易收敛。

上述第一非线性运算、卷积处理、逐点相乘运算和逐点相加运算可以表示为如下公式(2)：

$$y_i = x_i \times (\beta_i + \sum_j \gamma_{ij} \phi(x_j)) + x_i \quad (2)$$

其中， x 表示输入的特征值， y 表示输出的特征值， $\phi(x_j)$ 表示第一非线性运算， γ 表示卷积层的权重， β 表示卷积层的偏置参数。

上述公式(2)经变形可以得到： $y_i = x_i \times (\beta_i + 1 + \sum_j \gamma_{ij} \phi(x_j))$ ，与公式(1)对比可以得到卷积处理可以表示为如下公式：

$$\text{conv2}(x) = \beta + 1 + \sum \gamma \times x$$

由此可见，卷积处理 $\text{conv1}(x)$ 和卷积处理 $\text{conv2}(x)$ 相似，区别在于卷积处理 $\text{conv2}(x)$ 中的卷积参数多加了 1。这样可以通过对卷积处理中的卷积参数 β 的微调，实现上述两种实施方式之间的转换，即如果非线性变换单元中不包含逐点相加运算，则可以采用卷积处理 $\text{conv1}(x)$ ；如果非线性变换单元中包含逐点相加运算，则可以采用卷积处理 $\text{conv2}(x)$ 。

在一种可能的实现方式中，在上述第一种实施方式的基础上，还可以对第三图像特征进行第二非线性运算，再对输出的经处理的第三图像特征和第一图像特征进行逐点相乘运算，或者，在上述第二种实施方式的基础上，还可以对第三图像特征进行第二非线性运算，再对输出的经处理的第三图像特征和第一图像特征进行逐点相乘运算。即在非线性变换处理中增加了第二非线性运算，该第二非线性运算的输入均是非线性变换处理中的卷积处理的输出，该第二非线性运算的输出均是作为逐点相乘运算的一个输入。

图 10c 为具备注意力机制的残差非线性变换单元的结构图，如图 10c 所示，本申请实施例中，非线性变换单元用于实现上述非线性变换处理，其包括第一非线性运算、卷积处理、第二非线性运算和逐点相乘运算。

其中，第一非线性运算、卷积处理和逐点相乘运算可以参照图 10a 所示实施例的描述，此处不再赘述。第二非线性运算可以与第一非线性运算采用相同的运算方法，也可以采用不同的运算方法，其可以包括取绝对值、ReLU、LeakyReLU 等，本申请实施例对此不做具体限定。

上述第一非线性运算、卷积处理、第二非线性运算和逐点相乘运算可以表示为如下公式(3)：

$$y_i = x_i * \phi_2(\beta_i + \sum_j \gamma_{ij} \phi_1(x_j)) \quad (3)$$

其中， x 表示输入的特征值， y 表示输出的特征值， $\phi_1(x_j)$ 表示第一非线性运算， $\phi_2(x_j)$ 表示第二非线性运算， γ 表示卷积层的权重， β 表示卷积层的偏置参数。

图 10d 为具备注意力机制的非线性变换单元的结构图, 如图 10d 所示, 本申请实施例中, 非线性变换单元用于实现上述非线性变换处理, 其包括第一非线性运算、卷积处理、第二非线性运算、逐点相乘运算和逐点相加运算。

其中, 第一非线性运算、卷积处理、逐点相乘运算和逐点相加运算可以参照图 10b 所示实施例的描述, 此处不再赘述。第二非线性运算可以与第一非线性运算采用相同的运算方法, 也可以采用不同的运算方法, 其可以包括取绝对值、ReLU、LeakyReLU 等, 本申请实施例对此不做具体限定。

当第二非线性运算使用分段线性映射实现时, 分段线性映射可以采用不同的分段数量; 在各分段上的映射斜率可以通过训练的方式或者直接指定的方式确定; 对输入特征图像的每个通道可以采用不同的分段线性函数, 或者对所有通道都是用同一个分段线性函数, 或者对若干个通道采用同一个分段线性函数进行处理。在这种实现方式下, 模型训练完毕之后, 残差结构不再与卷积融合, 而是可与分段线性函数融合, 即在原来的分段线性函数的输出上+1, 组合成一个新的分段线性函数。

上述第一非线性运算、卷积处理、第二非线性运算、逐点相乘运算和逐点相加运算可以表示为如下公式 (4):

$$y_i = x_i * \phi_2(\beta_i + \sum_j \gamma_{ij} \phi_1(x_j)) + x_i \quad (4)$$

其中, x 表示输入的特征值, y 表示输出的特征值, $\phi_1(x_j)$ 表示第一非线性运算, $\phi_2(x_j)$ 表示第二非线性运算, γ 表示卷积层的权重, β 表示卷积层的偏置参数。

步骤 903、根据经处理的图像特征进行编码得到码流。

得到经处理的图像特征后, 编码侧可以继续对其进行卷积处理, 也可以在卷积处理之后再次对卷积处理的输出进行非线性变换处理, 然后再对前述处理的结果进行熵编码以得到码流, 熵编码可以采用图 4~图 8 所示实施例中的熵编码网络实现, 此处不再赘述。也可以采用其他的编码方式对前述处理的结果进行编码以得到码流, 本申请对此不做限定。

图 13 为编码网络的结构图, 如图 13 所示, 该编码网络包括 4 个卷积层 (conv) 和 3 个非线性变换单元, 卷积层和非线性变换单元交叉排列, 即对输入图像依次进行卷积处理、非线性变换处理、卷积处理、非线性变换处理、卷积处理、非线性变换处理以及卷积处理得到输出图像特征。非线性变换单元可以采用图 10a 或图 10b 所示实施例的结构。而后输出图像特征再进行熵编码, 此处不再赘述。

本申请实施例, 通过对解码网络中的非线性变换处理进行改变, 可以使输入的各个特征值只需要自身的特性即可得到输出值, 不用考虑周边特征值的影响, 并对原特征值进行修正, 把重要的特征值突出, 把冗余的特征值抑制, 此外还可以校正第一图像特征上每个特征值的值, 避免了对卷积参数的限定, 从而实现编码网络中的高效的非线性变换处理, 进一步提升图像/视频压缩算法中的率失真性能。

图 14 为本申请实施例的图像解码方法的过程 1300 的流程图。过程 1300 可由上述实施例中的解码端执行。过程 1300 描述为一系列的步骤或操作, 应当理解的是, 过程 1300 可以以各种顺序执行和/或同时发生, 不限于图 14 所示的执行顺序。过程 1300 包括如下步骤:

步骤 1301、获取待处理的第一图像特征。

解码端与编码端相对应, 第一图像特征可以是解码端对码流进行熵解码后, 熵解码可

以采用图 4~图 8 所示实施例中的熵解码网络实现，此处不再赘述。之后再经过卷积处理、反卷积处理、转置卷积处理、插值+卷积处理、Transformer 处理等方式得到的。应理解，经过前述处理，输出的第一图像特征的尺寸将复原（和编码端是镜像对称的），也可能使输入的图像特征发生尺寸变化、通道数量改变等，对此不做具体限定。前述处理与图 9 所示实施例中的步骤 901 中的转换方式相反。

同样的，第一图像特征也可以表示成二维矩阵或三维矩阵的形式，原理参照步骤 901 的描述，此处不再赘述。

步骤 1302、对第一图像特征进行非线性变换处理得到经处理的图像特征。

10 在一种可能的实现方式中，对第一图像特征中的每个特征值进行第一非线性运算得到第二图像特征；对第二图像特征进行卷积处理得到第三图像特征，第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到经处理的图像特征。

本实施例可以参照图 10a 所示实施例，此处不再赘述。

15 在一种可能的实现方式中，对第一图像特征中的每个特征值进行第一非线性运算得到第二图像特征；对第二图像特征进行卷积处理得到第三图像特征，第三图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第三图像特征中相对应的多个特征值进行逐点相乘运算得到第四图像特征，第四图像特征中的多个特征值和第一图像特征中的多个特征值相对应；将第一图像特征和第四图像特征中相对应的多个特征值进行逐点相加运算得到经处理的图像特征。

20 本实施例可以参照图 10b 所示实施例，此处不再赘述。

步骤 1303、根据经处理的图像特征获取重建图像。

得到经处理的图像特征后，解码侧也可以继续对其进行卷积处理，也可以在卷积处理之后再次对卷积处理的输出进行非线性变换处理，从而将其从特征域转换到图像域，得到重建图像。

25 图 15 为解码网络的结构图，如图 15 所示，该解码网络包括 4 个反卷积层（Deconv）和 3 个非线性变换单元，反卷积层和非线性变换单元交叉排列，即对输入图像依次进行反卷积处理、非线性变换处理、反卷积处理、非线性变换处理、反卷积处理、非线性变换处理以及反卷积处理得到重建图像。非线性变换单元可以采用图 10a 或图 10b 所示实施例的结构。而后输出图像特征在进行熵解码，此处不再赘述。

30 本申请实施例，通过对解码网络中的非线性变换处理进行改变，可以使输入的各个特征值只需要自身的特性即可得到输出值，不用考虑周边特征值的影响，并对原特征值进行修正，把重要的特征值突出，把冗余的特征值抑制，此外还可以校正第一图像特征上每个特征值的值，避免了对卷积参数的限定，从而实现编码网络中的高效的非线性变换处理，进一步提升图像/视频压缩算法中的率失真性能。

35 需要说明的是，本申请实施例还提供了一种编/解码网络的训练方式，可以包括：首先构建端到端的编/解码网络，包含编码器 Encoder、解码器 Decoder、熵估计单元。训练时将编解码网络当作一个整体，训练数据（图像或视频）输入到编码器得到特征数据；特征数据一方面经过熵估计单元计算编码码率开销，得到码率 loss；特征数据另一方面经过解码端输出重建数据，重建数据和输入数据计算失真程度，得到失真 loss。反向传播算法通

过码率 loss 和失真 loss 组成的加权 loss 去更新模型中的可学习参数。训练完毕之后，模型中的所有子模块的参数被固定。

此后，编码器 Encoder 和熵估计单元被划分到编码端，用于将待编码数据编码成码流文件；解码器和熵估计单元被划分到解码端，用于从码流文件中重建数据。

- 5 以下通过几个具体的实施例对上述图像编码方法进行说明，下文中将本申请实施例提供的非线性变换单元称作 ResAU。

实施例一

- 10 图 16a 为 ResAU 的一个示例性的结构图，如图 16a 所示，ResAU 采用图 10b 所示的结构，包括第一非线性运算、卷积处理、逐点相乘运算和逐点相加运算，其中第一非线性运算采用取绝对值 (abs)。该 ResAU 可以应用于图 13 所示的编码网络或者图 15 所示的解码网络。

在公式 (2) 的基础上，本实施例的 ResAU 可以表示为如下公式：

$$y_i = x_i * (\beta_i + \sum_j \gamma_{ij} |x_j|) + x_i$$

对图 16a 所示的 ResAU 进行压缩性能测试。

- 15 测试集：Kodak 测试集，包含 24 张分辨率为 768×512 或 512×768 的便携式网络图形 (portable network graphics, PNG) 图像。

实验：在混合高斯超先验熵估计的编/解码网络结构中使用图 16a 所示的 ResAU。

- 20 实验对应的性能表现：图 17a 为 ResAU 在 Kodak 测试集的 24 张图像上的整体表现。与当前主流使用的 GDN 非线性单元相比，在相同的计算复杂度基础上，使用 ResAU 的 GMM 网络 RD 性能更优。更具体地，在达到相同的解码重建质量下，在 GMM 编/解码网络下使用 ResAU 可比使用 ReLU 节省约 12% 的编码码率开销，可比使用 GDN 节省约 5% 的编码码率开销。

- 25 性能分析：在本申请实施例提供的 ResAU 中，通过取绝对值引入非线性特性是一种可行的方案。在基础的 GMM 结构上，使用 ResAU 能比使用 GDN 取得更好的率失真性能。

实施例二

- 30 图 16b 为 ResAU 的一个示例性的结构图，如图 16b 所示，ResAU 采用图 10b 所示的结构，包括第一非线性运算、卷积处理、逐点相乘运算和逐点相加运算，其中第一非线性运算采用 ReLU 运算。该 ResAU 可以应用于图 13 所示的编码网络或者图 15 所示的解码网络。

在公式 (2) 的基础上，本实施例的 ResAU 可以表示为如下公式：

$$y_i = x_i * (\beta_i + \sum_j \gamma_{ij} \phi(x_j)) + x_i$$

对图 16b 所示的 ResAU 进行压缩性能测试。

测试集：24 张 Kodak 测试图像。

- 35 实验：在混合高斯超先验熵估计的编/解码网络结构中使用图 16b 所示的 ResAU。在本次实验中，ResAU 中的非线性运算分别替换成了 ReLU、LeakyReLU，对比实验包括无非线性运算的 Identity 方案以及实施例一所描述的取绝对值的 ResAU 方案。

实验对应的性能表现：图 17b 为 ResAU 在 Kodak 测试集的 24 张图像上的整体表现。从 RD 曲线可知，相比于不使用非线性运算的 Identity 方案，使用逐点非线性运算可以较大幅度提升图像压缩网络的率失真性能。而 ReLU 类的非线性运算效果会略优于取绝对值的非线性运算。

5 实施例三

图 16c 为 ResAU 的一个示例性的结构图，如图 16c 所示，ResAU 采用图 10d 所示的结构，包括第一非线性运算、卷积处理、第二非线性运算、逐点相乘运算和逐点相加运算，其中第一非线性运算和第二非线性运算均采用 PWL。该 ResAU 可以应用于图 13 所示的编码网络或者图 15 所示的解码网络。

- 10 ResAU 通过分段线性函数 PWL 引入非线性特性；通过卷积、分段线性函数 PWL 和逐点相乘运算来实现局部注意力机制，利用局部信息校正特征图上各个位置各个通道上的响应；逐点相乘操作可以避免 GDN 中存在的可学习参数取值空间受限的问题；另外，首尾相加的残差形式的连接可使网络在训练时更容易收敛。

在公式（4）的基础上，本实施例的 ResAU 可以表示为如下公式：

$$15 \quad y_i = x_i * \phi_2(\beta_i + \sum_j \gamma_{ij} \phi_1(x_j)) + x_i$$

其中， $\phi_1(\cdot)$ 和 $\phi_2(\cdot)$ 均为 PWL 函数， γ 为卷积层的权重， β 为卷积层的偏置参数，两者都是取值范围不受限制的可学习参数。

该结构包含的各个运算的功能如下：

- 20 第一个 PWL 运算：该函数为分段线性函数，可以为整体变换提供非线性特性。在该运算的作用下，输入数据的每个值都会依据其所在数值区间采用不同的映射关系计算得到输出值。对于一个输入特征，其不同通道维度的特征图可以采用相同或者不同的分段线性映射函数进行运算。其参数可以为预设值，也可以通过训练学习得到。

卷积处理：在 ResAU 中，卷积处理的输入为非线性运算的输出。输入张量通过卷积处理得到尺寸不变的张量，输出张量可以认为是输入张量的局部响应。

- 25 第二个 PWL 运算：该函数为分段线性函数，可以对卷积的输出进行缩放映射，也为整体变换提供非线性特性。在该运算的作用下，输入数据的每个值都会依据其所在数值区间采用不同的映射关系计算得到输出值。对于一个输入特征，其不同通道维度的特征图可以采用相同或者不同的分段线性映射函数进行运算。其参数可以为预设值，也可以通过训练学习得到。

- 30 逐点相乘运算：在 ResAU 中，逐点相乘运算的输入为该单元的原始输入和卷积处理的输出。其两项输入张量的尺寸相同，执行对应位置数据的相乘运算，输出张量的尺寸和输入张量的尺寸也相同。卷积处理和逐点相乘运算结合实现了局部注意力机制，利用输入特征的局部信息校正特征图上每个位置的响应。同时，逐点相乘运算避免了当前用于图像视频压缩网络中的主流非线性单元 GDN 存在的可学习参数取值空间受限的问题。

- 35 逐点相加运算：在 ResAU 中，逐点相加运算的输入为该单元的原始输入和逐点相乘运算的输出。该运算是一个首位相加的残差结构，可以使使用该非线性单元的编解码网络在训练时更容易收敛。

本实施例所描述的使用分段线性函数 PWL 的残差注意力非线性单元可用于基于深度

神经网络的端到端图像压缩网络和视频压缩网络之中。更具体地，一般用于端到端图像压缩网络中的编码模块（Encoder）和解码模块（Decoder）中，用于端到端视频压缩网络预测子网络和残差压缩子网络的编码模块和解码模块中。

5 在一个常用的基于超先验结构的端到端编解码网络上进行了实验：当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度分组粒度为 1（即所有通道的特征使用不同的分段线性函数计算输出值）时，能够比采用卷积运算之后使用 Tanh 实现非线性运算的方案提升 0.506% 的压缩率失真性能；当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度分组粒度为 8（即以每 8 个通道为单位对特征进行分组，同一组内的所有通道使用相同的分段线性函数计算输出值）时，比采用卷积运算之后使用 Tanh 实现非线性运算的方案降低 0.488% 的压缩率失真性能；当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度不分组（即所有通道的特征使用相同的分段线性函数计算输出值）时，比采用卷积运算之后使用 Tanh 实现非线性运算的方案降低 0.659% 的压缩率失真性能。

15 可选的，图 16d 为 ResAU 的一个示例性的结构图，如图 16d 所示，ResAU 采用图 10d 所示的结构，包括第一非线性运算、卷积处理、第二非线性运算、逐点相乘运算和逐点相加运算，其中第一非线性运算采用 LeakyReLU，第二非线性运算采用 tanh。卷积处理可以是 conv1×1，该 conv1×1 表示卷积核（或卷积算子）的尺寸为 1×1。该 ResAU 可以应用于图 13 所示的编码网络或者图 15 所示的解码网络。

20 实施例四

图 16e 为 ResAU 的一个示例性的结构图，如图 16e 所示，ResAU 采用图 10c 所示的结构，包括第一非线性运算、卷积处理、第二非线性运算和逐点相乘运算，其中第一非线性运算和第二非线性运算均采用 PWL。该 ResAU 可以应用于图 13 所示的编码网络或者图 15 所示的解码网络。

25 在公式（3）的基础上，本实施例的 ResAU 可以表示为如下公式：

$$y_i = x_i * \phi_2(\beta_i + \sum_j \gamma_{ij} \phi_1(x_j))$$

其中， $\phi_1(\cdot)$ 和 $\phi_2(\cdot)$ 均为 PWL 函数， γ 为卷积层的权重， β 为卷积层的偏置参数，两者都是取值范围不受限制的可学习参数。

该结构包含的各个运算的功能如下：

30 第一个 PWL 运算：该函数为分段线性函数，可以为整体变换提供非线性特性。在该运算的作用下，输入数据的每个值都会依据其所在数值区间采用不同的映射关系计算得到输出值。对于一个输入特征，其不同通道维度的特征图可以采用相同或者不同的分段线性映射函数进行运算。其参数可以为预设值，也可以通过训练学习得到。

35 卷积处理：在 ResAU 中，卷积处理的输入为非线性运算的输出。输入张量通过卷积处理得到尺寸不变的张量，输出张量可以认为是输入张量的局部响应。

第二个 PWL 运算：该函数为分段线性函数，可以对卷积的输出进行缩放映射，也为整体变换提供非线性特性。在该运算的作用下，输入数据的每个值都会依据其所在数值区间采用不同的映射关系计算得到输出值。对于一个输入特征，其不同通道维度的特征图

可以采用相同或者不同的分段线性映射函数进行运算。其参数可以为预设值，也可以通过训练学习得到。

逐点相乘运算：在 ResAU 中，逐点相乘运算的输入为该单元的原始输入和卷积处理的输出。其两项输入张量的尺寸相同，执行对应位置数据的相乘运算，输出张量的尺寸和输入张量的尺寸也相同。卷积处理和逐点相乘运算结合实现了局部注意力机制，利用输入特征的局部信息校正特征图上每个位置的响应。同时，逐点相乘运算避免了当前用于图像视频压缩网络中的主流非线性单元 GDN 存在的可学习参数取值空间受限的问题。

本实施例所描述的使用分段线性函数的无残差结构的注意力非线性单元可用于基于深度神经网络的端到端图像压缩网络和视频压缩网络之中。更具体地，一般用于端到端图像压缩网络中的编码模块 Encoder 和解码模块 Decoder 中，用于端到端视频压缩网络预测子网络和残差压缩子网络的编码模块和解码模块中。并且，本实施例所描述的使用分段线性函数的无残差结构的注意力非线性单元可以通过使用分段线性函数的残差注意力非线性单元转换得到。在训练完成之后，可以将使用分段线性函数的残差注意力非线性单元中的第二个 PWL 运算和逐点相加运算合并（即将第二个 PWL 运算的输出整体加 1，形成新的 PWL 函数，并去除逐点相加运算），得到一个对应的使用分段线性函数的无残差结构的注意力非线性单元。

在一个常用的基于超先验结构的端到端编解码网络上进行了实验，通过将实施例四对应的实验中的残差注意力非线性单元在训练完毕后转换为本实施例所描述的无残差结构的非线性单元，可以获得和实施例四对应方案相同的编解码效果。

当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度分组粒度为 1（即所有通道的特征使用不同的分段线性函数计算输出值）时，能够比采用卷积运算之后使用 Tanh 实现非线性运算的方案提升 0.506% 的压缩率失真性能。并且，本实施例方案能够节省在逐点相加运算上的计算耗时和功耗。

当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度分组粒度为 8（即以每 8 个通道为单位对特征进行分组，同一组内的所有通道使用相同的分段线性函数计算输出值）时，比采用卷积运算之后使用 Tanh 实现非线性运算的方案降低 0.488% 的压缩率失真性能。并且，本实施例方案能够节省在逐点相加运算上的计算耗时和功耗。

当本实施例中的第一个 PWL 运算预设值为 Leaky ReLU 函数、且第二个 PWL 运算分段数为 6、通道维度不分组（即所有通道的特征使用相同的分段线性函数计算输出值）时，比采用卷积运算之后使用 Tanh 实现非线性运算的方案降低 0.659% 的压缩率失真性能。并且，本实施例方案能够节省在逐点相加运算上的计算耗时和功耗。

图 18 为本申请实施例编码装置 1700 的一个示例性的结构示意图，如图 18 所示，本实施例的装置 1700 可以应用于编码端。该装置 1700 可以包括：获取模块 1701、变换模块 1702、编码模块 1703 和训练模块 1704。其中，

获取模块 1701，用于获取待处理的第一图像特征；变换模块 1702，用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；编码模块 1703，用于根据所述经处理的图像特征进行编码得到码流。

在一种可能的实现方式中，所述变换模块 1702，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

在一种可能的实现方式中，所述变换模块 1702，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

在一种可能的实现方式中，所述非线性运算包括分段线性映射，例如 ReLU，LeakyReLU，PWL，Abs 等。在另一种可能的实现方式中，所述非线性运算包括连续函数，例如 Tanh，Sigmoid 等。在另一种可能的实现方式中，所述非线性运算包括分段的非线性运算。

在一种可能的实现方式中，还包括：训练模块 1704，用于构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层；根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。

本实施例的装置，可以用于执行图 9 所示方法实施例的技术方案，其实现原理和技术效果类似，此处不再赘述。

图 19 为本申请实施例解码装置 1800 的一个示例性的结构示意图，如图 19 所示，本实施例的装置 1800 可以应用于解码端。该装置 1800 可以包括：获取模块 1801、变换模块 1802、重建模块 1803 和训练模块 1804。其中，

获取模块 1801，用于获取待处理的第一图像特征；变换模块 1802，用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；重建模块 1803，用于根据所述经处理的图像特征获取重建图像。

在一种可能的实现方式中，所述变换模块 1802，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述逐点相乘运算之后的逐

点相加运算。

在一种可能的实现方式中，所述变换模块 1802，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

在一种可能的实现方式中，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

在一种可能的实现方式中，所述非线性运算包括分段线性映射，例如 ReLU, LeakyReLU, PWL, Abs 等。在另一种可能的实现方式中，所述非线性运算包括连续函数，例如 Tanh, Sigmoid 等。在另一种可能的实现方式中，所述非线性运算包括分段的非线性运算。

在一种可能的实现方式中，训练模块 1804，用于构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层；根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。

本实施例的装置，可以用于执行图 14 所示方法实施例的技术方案，其实现原理和技术效果类似，此处不再赘述。

本申请实施例提供一种码流，该码流由处理器执行上述实施例中任一种编码方法所生成。

本申请实施例提供一种存储码流的装置，该装置包括：接收器和至少一个存储介质，接收器用于接收码流；至少一个存储介质用于存储码流；码流为上述实施例中任一种编码方法所生成的码流。

本申请实施例提供一种传输码流的装置，该装置包括：发送器和至少一个存储介质，至少一个存储介质用于存储码流，码流由处理器执行上述实施例中任一种编码方法所生成；发送器用于将码流发送给其他电子设备。可选的，该传输码流的装置还包括接收器和处理器，接收器用于接收用户请求，处理器用于响应于用户请求，以从存储介质中选择目标码流，并指示发送器发送目标码流。

本申请实施例提供一种分发码流的系统，该系统包括：至少一个存储介质和流媒体设备，至少一个存储介质用于存储至少一个码流，至少一个码流包括根据第一方面的任意一种实现方式生成的码流；流媒体设备用于从至少一个存储介质中获取目标码流，并将目标码流发送给端侧设备，其中，流媒体设备包括内容服务器或内容分发服务器。

本申请实施例提供一种分发码流的系统，该系统包括：通信接口，用于接收用户获取目标码流的请求；处理器，用于响应于用户请求，确定目标码流的存储位置；通信接口还用于将目标码流的存储位置发送给用户，以使得用户从目标码流的存储位置获取目标码流，其中，目标码流由处理器执行上述实施例中任一种编码方法所生成。

在实现过程中，上述方法实施例的各步骤可以通过处理器中的硬件的集成逻辑电路或

者软件形式的指令完成。处理器可以是通用处理器、数字信号处理器(digital signal processor, DSP)、特定应用集成电路(application-specific integrated circuit, ASIC)、现场可编程门阵列(field programmable gate array, FPGA)或其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。本申请实施例公开的方法的步骤可以直接体现为硬件编码处理器执行完成,或者用编码处理器中的硬件及软件模块组合执行完成。软件模块可以位于随机存储器,闪存、只读存储器,可编程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器,处理器读取存储器中的信息,结合其硬件完成上述方法的步骤。

- 10 上述各实施例中提及的存储器可以是易失性存储器或非易失性存储器,或可包括易失性和非易失性存储器两者。其中,非易失性存储器可以是只读存储器(read-only memory, ROM)、可编程只读存储器(programmable ROM, PROM)、可擦除可编程只读存储器(erasable PROM, EPROM)、电可擦除可编程只读存储器(electrically EPROM, EEPROM)或闪存。易失性存储器可以是随机存取存储器(random access memory, RAM),其用作
- 15 外部高速缓存。通过示例性但不是限制性说明,许多形式的RAM可用,例如静态随机存取存储器(static RAM, SRAM)、动态随机存取存储器(dynamic RAM, DRAM)、同步动态随机存取存储器(synchronous DRAM, SDRAM)、双倍数据速率同步动态随机存取存储器(double data rate SDRAM, DDR SDRAM)、增强型同步动态随机存取存储器(enhanced SDRAM, ESDRAM)、同步连接动态随机存取存储器(synchlink DRAM, SLDRAM)和直接内存总线随机存取存储器(direct rambus RAM, DR RAM)。应注意,本文描述的系统和方法的存储器旨在包括但不限于这些和任意其它适合类型的存储器。

- 20 本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、或者计算机软件和电子硬件的结合来实现。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本申请实施例的范围。
- 25 所属领域的技术人员可以清楚地了解到,为描述的方便和简洁,上述描述的系统、装置和单元的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。

- 30 在本申请实施例所提供的几个实施例中,应该理解到,所揭露的系统、装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,所述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

- 35 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

另外,在本申请实施例各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。

所述功能如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读取存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算机设备（个人计算机，服务器，或者网络设备等）执行本申请实施例各个实施例所述方法的全部或部分步骤。或者该计算机软件产品可以从一个计算机可读存储介质向另一个计算机可读存储介质传输，例如，所述计算机指令可以从一个网站站点、计算机、服务器或数据中心通过有线（例如同轴电缆、光纤、数字用户线（DSL））或无线（例如红外、无线、微波等）方式向另一个网站站点、计算机、服务器或数据中心进行传输。所述计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。所述可用介质可以是磁性介质，（例如，软盘、硬盘、磁带）、光介质（例如，DVD）、或者半导体介质（例如固态硬盘 Solid State Disk (SSD)、只读存储器（read-only memory，ROM）、随机存取存储器（random access memory，RAM））等。

以上所述，仅为本申请实施例的具体实施方式，但本申请实施例的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请实施例揭露的技术范围内，可轻易想到变化或替换，都应涵盖在本申请实施例的保护范围之内。因此，本申请实施例的保护范围应以所述权利要求的保护范围为准。

权 利 要 求 书

1. 一种图像编码方法，其特征在于，包括：
获取待处理的第一图像特征；
5 对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；
根据所述经处理的图像特征进行编码得到码流。
2. 根据权利要求1所述的方法，其特征在于，所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征，包括：
对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；
10 对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；
将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。
3. 根据权利要求1所述的方法，其特征在于，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。
- 15 4. 根据权利要求3所述的方法，其特征在于，所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征，包括：
对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；
对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；
20 将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；
将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。
- 25 5. 根据权利要求1-4中任一项所述的方法，其特征在于，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。
6. 根据权利要求1-5中任一项所述的方法，其特征在于，所述第一非线性运算包括
30 Sigmoid、Tanh 或者分段线性映射。
7. 根据权利要求5或6所述的方法，其特征在于，所述第二非线性运算包括 Sigmoid、Tanh 或者分段线性映射。
8. 根据权利要求1-7中任一项所述的方法，其特征在于，所述方法还包括：
构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运
35 算层、卷积处理层、逐点相乘运算层和逐点相加运算层；
根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。
9. 一种图像解码方法，其特征在于，包括：

获取待处理的第一图像特征；

对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；

根据所述经处理的图像特征获取重建图像。

- 5 10. 根据权利要求9所述的方法，其特征在于，所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征，包括：

对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；

对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；

- 10 将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

11. 根据权利要求9所述的方法，其特征在于，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

- 15 12. 根据权利要求11所述的方法，其特征在于，所述对所述第一图像特征进行非线性变换处理得到经处理的图像特征，包括：

对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；

对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；

- 20 将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；

将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

- 25 13. 根据权利要求9-12中任一项所述的方法，其特征在于，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

14. 根据权利要求9-13中任一项所述的方法，其特征在于，所述第一非线性运算包括Sigmoid、Tanh或者分段线性映射。

- 30 15. 根据权利要求13或14所述的方法，其特征在于，所述第二非线性运算包括Sigmoid、Tanh或者分段线性映射。

16. 根据权利要求9-15中任一项所述的方法，其特征在于，所述方法还包括：

构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层；

- 35 根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。

17. 一种编码装置，其特征在于，包括：

获取模块，用于获取待处理的第一图像特征；

变换模块，用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理依次包括第一非线性运算、卷积处理和逐点相乘运算；

编码模块，用于根据所述经处理的图像特征进行编码得到码流。

18. 根据权利要求 17 所述的装置，其特征在於，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

19. 根据权利要求 17 所述的装置，其特征在於，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

20. 根据权利要求 19 所述的装置，其特征在於，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

21. 根据权利要求 17-20 中任一项所述的装置，其特征在於，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

22. 根据权利要求 17-21 中任一项所述的装置，其特征在於，所述第一非线性运算包括 Sigmoid、Tanh 或者分段线性映射。

23. 根据权利要求 21 或 22 所述的装置，其特征在於，所述第二非线性运算包括 Sigmoid、Tanh 或者分段线性映射。

24. 根据权利要求 17-23 中任一项所述的装置，其特征在於，还包括：

- 训练模块，用于构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层；根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。

25. 一种解码装置，其特征在於，包括：

- 获取模块，用于获取待处理的第一图像特征；
变换模块，用于对所述第一图像特征进行非线性变换处理得到经处理的图像特征，所述非线性变换处理包括第一非线性运算、卷积处理和逐点相乘运算；
重建模块，用于根据所述经处理的图像特征获取重建图像。

26. 根据权利要求 25 所述的装置，其特征在於，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到所述经处理的图像特征。

27. 根据权利要求 25 所述的装置，其特征在於，所述非线性变换处理还包括在所述逐点相乘运算之后的逐点相加运算。

28. 根据权利要求 27 所述的装置，其特征在于，所述变换模块，具体用于对所述第一图像特征中的每个特征值进行所述第一非线性运算得到第二图像特征；对所述第二图像特征进行所述卷积处理得到第三图像特征，所述第三图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第三图像特征中相对应的多个特征值进行所述逐点相乘运算得到第四图像特征，所述第四图像特征中的多个特征值和所述第一图像特征中的多个特征值相对应；将所述第一图像特征和所述第四图像特征中相对应的多个特征值进行所述逐点相加运算得到所述经处理的图像特征。

29. 根据权利要求 25-28 中任一项所述的装置，其特征在于，所述非线性变换处理还包括在所述卷积处理和所述逐点相乘运算之间的第二非线性运算，所述第二非线性运算和所述第一非线性运算相同或不同。

30. 根据权利要求 25-29 中任一项所述的装置，其特征在于，所述第一非线性运算包括 Sigmoid、Tanh 或者分段线性映射。

31. 根据权利要求 29 或 30 所述的装置，其特征在于，所述第二非线性运算包括 Sigmoid、Tanh 或者分段线性映射。

32. 根据权利要求 25-31 中任一项所述的装置，其特征在于，所述装置还包括：

训练模块，用于构建训练阶段的非线性变换单元，所述训练阶段的非线性变换单元包括第一非线性运算层、卷积处理层、逐点相乘运算层和逐点相加运算层；根据预先获取的训练数据训练得到经训练的非线性变换单元，所述经训练的非线性变换单元用于实现所述非线性变换处理。

33. 一种编码器，其特征在于，包括：

一个或多个处理器；

非瞬时性计算机可读存储介质，耦合到所述处理器并存储由所述处理器执行的程序，其中所述程序在由所述处理器执行时，使得所述解码器执行权利要求 1-8 中任一项所述的方法。

34. 一种解码器，其特征在于，包括：

一个或多个处理器；

非瞬时性计算机可读存储介质，耦合到所述处理器并存储由所述处理器执行的程序，其中所述程序在由所述处理器执行时，使得所述解码器执行权利要求 9-16 中任一项所述的方法。

35. 一种计算机程序产品，其特征在于，包括程序代码，当所述程序代码在计算机或处理器上执行时，用于执行权利要求 1-16 中任一项所述的方法。

36. 一种计算机可读存储介质，包括指令，当其在计算机上运行时，使得计算机执行如权利要求 1-16 任意一项所述的方法。