



(12) 发明专利申请

(10) 申请公布号 CN 116250236 A

(43) 申请公布日 2023. 06. 09

(21) 申请号 202180063932.9

(74) 专利代理机构 永新专利商标代理有限公司  
72002

(22) 申请日 2021.08.30

专利代理师 刘瑜

(30) 优先权数据

63/083,747 2020.09.25 US

17/201,944 2021.03.15 US

(51) Int.Cl.

H04N 19/147 (2006.01)

(85) PCT国际申请进入国家阶段日

2023.03.17

(86) PCT国际申请的申请数据

PCT/US2021/048245 2021.08.30

(87) PCT国际申请的公布数据

W02022/066368 EN 2022.03.31

(71) 申请人 高通股份有限公司

地址 美国加利福尼亚

(72) 发明人 T·J·范罗森达尔

I·A·M·胡伊本 T·S·科亨

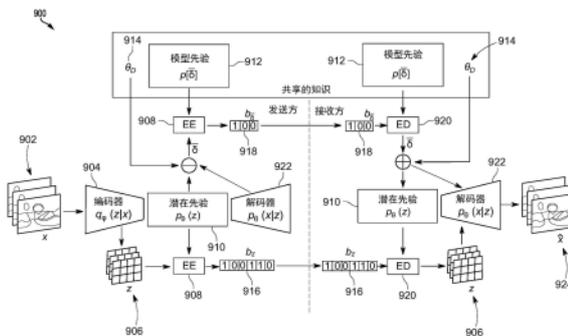
权利要求书4页 说明书37页 附图16页

(54) 发明名称

使用机器学习系统的实例自适应图像和视频压缩

(57) 摘要

描述了用于使用机器学习系统来压缩数据以及调谐机器学习系统以压缩数据的技术。示例过程可以包括：通过神经网络压缩系统(例如，在训练数据集上训练的)接收输入数据以便通过所述神经网络压缩系统压缩。所述过程可以包括：确定用于所述神经网络压缩系统的一组更新，所述一组更新包括使用所述输入数据调谐的经更新的模型参数。所述过程可以包括：通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的第二比特流。所述过程还可以包括：通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流。所述过程可以包括：输出所述第一比特流和所述第二比特流以传输到接收机。



1. 一种装置,包括:  
存储器;以及  
耦合至所述存储器的一个或多个处理器,所述一个或多个处理器被配置为:  
通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;  
确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;  
通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的  
第一比特流;  
通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模  
型参数的经压缩的版本的第二比特流;以及  
输出所述第一比特流和所述第二比特流以传输到接收机。
2. 根据权利要求1所述的装置,其中,所述第二比特流还包括所述潜在先验的经压缩的  
版本和所述模型先验的经压缩的版本。
3. 根据权利要求1所述的装置,其中,所述一个或多个处理器被配置为:  
生成包括所述第一比特流和所述第二比特流的经级联的比特流;以及  
将所述经级联的比特流发送到所述接收机。
4. 根据权利要求1所述的装置,其中,为了生成所述第二比特流,所述一个或多个处理  
器被配置为:  
通过所述神经网络压缩系统使用所述模型先验对所述潜在先验进行熵编码;以及  
通过所述神经网络压缩系统使用所述模型先验对所述经更新的模型参数进行熵编码。
5. 根据权利要求1所述的装置,其中,所述经更新的模型参数包括解码器模型的一个或  
多个经更新的参数,所述一个或多个经更新的参数使用所述输入数据被调谐。
6. 根据权利要求1所述的装置,其中,所述经更新的模型参数包括编码器模型的一个或  
多个经更新的参数,所述一个或多个经更新的参数使用所述输入数据被调谐,其中,所述第  
一比特流是通过所述神经网络压缩系统使用所述一个或多个经更新的参数生成的。
7. 根据权利要求6所述的装置,其中,为了生成所述第二比特流,所述一个或多个处理  
器被配置为:  
通过所述神经网络压缩系统使用所述一个或多个经更新的参数将所述输入数据编码  
为所述输入数据的潜在空间表示;以及  
通过所述神经网络压缩系统使用所述潜在先验将所述潜在空间表示熵编码为所述第  
一比特流。
8. 根据权利要求1所述的设备,其中,所述一个或多个处理器被配置为:  
基于用于训练所述神经网络压缩系统的训练数据集来生成所述神经网络压缩系统的  
模型参数;  
使用所述输入数据来调谐所述神经网络压缩系统的所述模型参数;以及  
基于所述模型参数与经调谐的模型参数之间的差异来确定所述一组更新。
9. 根据权利要求8所述的装置,其中,所述模型参数是基于所述输入数据、所述输入数  
据的所述经压缩的版本的比特大小、所述一组更新的比特大小、以及所述输入数据与从所  
述输入数据的所述经压缩的版本生成的重构的数据之间的失真被调谐的。

10. 根据权利要求8所述的装置,其中,所述模型参数是基于所述输入数据以及发送所述一组更新的成本与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真的比率被调谐的,所述成本基于所述一组更新的比特大小。

11. 根据权利要求8所述的装置,其中,为了调谐所述模型参数,所述一个或多个处理器被配置为:

基于以下确定将一个或多个参数包括在所述经调谐的模型参数中:将所述一个或多个参数包括在所述经调谐的模型参数中伴随着以下各项中的至少一项的减少:所述输入数据的所述经压缩的版本的比特大小以及所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真。

12. 根据权利要求1所述的装置,其中,为了确定用于所述神经网络压缩系统的所述一组更新,所述一个或多个处理器被配置为:

处理所述神经网络压缩系统处的所述输入数据;

基于经处理的输入数据确定所述神经网络压缩系统的一个或多个损失;以及

基于所述一个或多个损失来调谐所述神经网络压缩系统的模型参数,经调谐的模型参数包括用于所述神经网络压缩系统的所述一组更新。

13. 根据权利要求12所述的装置,其中,所述一个或多个损失包括:与用于基于所述第一比特流的大小来发送所述输入数据的所述经压缩的版本的速率相关联的速率损失,与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真相关联的失真损失,以及与用于基于所述第二比特流的大小来发送所述经更新的模型参数的所述经压缩的版本的速率相关联的模型速率损失。

14. 权利要求1的装置,其中,所述接收机包括编码器,并且其中,所述一个或多个处理器被配置为:

通过所述编码器接收包括所述第一比特流和所述第二比特流的数据;

通过所述解码器对基于所述第二比特流的所述经更新的模型参数的所述经压缩的版本进行解码;以及

通过所述解码器使用一组经更新的参数,基于所述第一比特流中所述输入数据的所述经压缩的版本生成所述输入数据的经重构的版本。

15. 根据权利要求1所述的装置,其中,所述一个或多个处理器被配置为:

通过降低速率失真和模型速率损失来训练所述神经网络压缩系统,其中,模型速率反映了用于发送模型更新的比特流的长度。

16. 根据权利要求1所述的装置,其中,所述模型先验包括以下各项中的至少一项:独立高斯网络先验、独立拉普拉斯网络先验,以及独立Spike和Slab网络先验。

17. 根据权利要求1所述的装置,其中,所述装置包括移动设备。

18. 根据权利要求1所述的装置,还包括被配置为捕获所述输入数据的相机。

19. 一种方法,包括:

通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;

确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;

通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的

第一比特流；

通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流；以及

输出所述第一比特流和所述第二比特流以传输到接收机。

20. 根据权利要求19所述的方法，其中，所述第二比特流还包括所述潜在先验的经压缩的版本和所述模型先验的经压缩的版本。

21. 根据权利要求19所述的方法，其中，所述一个或多个处理器被配置为：生成包括所述第一比特流和所述第二比特流的经级联的比特流；以及将所述经级联的比特流发送到所述接收机。

22. 根据权利要求19所述的方法，其中，生成所述第二比特流包括：通过所述神经网络压缩系统使用所述模型先验对所述潜在先验进行熵编码；以及通过所述神经网络压缩系统使用所述模型先验对所述经更新的模型参数进行熵编码。

23. 根据权利要求19所述的方法，其中，所述经更新的模型参数包括解码器模型的一个或多个经更新的参数，所述一个或多个经更新的参数使用所述输入数据被调谐。

24. 根据权利要求19所述的方法，其中，所述经更新的模型参数包括编码器模型的一个或多个经更新的参数，所述一个或多个经更新的参数使用所述输入数据被调谐，其中，所述第一比特流是通过所述神经网络压缩系统使用所述一个或多个经更新的参数生成的。

25. 根据权利要求24所述的方法，其中，生成所述第二比特流包括：

通过所述神经网络压缩系统使用所述一个或多个经更新的参数将所述输入数据编码为所述输入数据的潜在空间表示；以及

通过所述神经网络压缩系统使用所述潜在先验将所述潜在空间表示熵编码为所述第一比特流。

26. 根据权利要求19所述的方法，其中，所述一个或多个处理器被配置为：

基于用于训练所述神经网络压缩系统的训练数据集来生成所述神经网络压缩系统的模型参数；

使用所述输入数据来调谐所述神经网络压缩系统的所述模型参数；以及基于所述模型参数与经调谐的模型参数之间的差异来确定所述一组更新。

27. 根据权利要求26所述的方法，其中，所述模型参数是基于所述输入数据以及以下各项中的至少一项被调谐的：所述输入数据的所述经压缩的版本的比特大小，所述一组更新的比特大小，所述输入数据与从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真，发送所述一组更新的成本与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真的比率，所述成本基于所述一组更新的比特大小。

28. 根据权利要求26所述的方法，其中，调谐所述模型参数包括：基于以下确定将一个或多个参数包括在所述经调谐的模型参数中：将所述一个或多个参数包括在所述经调谐的模型参数中伴随着以下各项中的至少一项的减少：所述输入数据的所述经压缩的版本的比特大小以及所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真。

29. 根据权利要求19所述的方法，其中，确定用于所述神经网络压缩系统的所述一组更新包括：

处理所述神经网络压缩系统处的所述输入数据；

基于经处理的输入数据确定所述神经网络压缩系统的一个或多个损失；以及

基于所述一个或多个损失来调谐所述神经网络压缩系统的模型参数，经调谐的模型参数包括用于所述神经网络压缩系统的所述一组更新，

其中，所述一个或多个损失包括：与用于基于所述第一比特流的大小来发送所述输入数据的所述经压缩的版本的速率相关联的速率损失，与所述输入数据和从所述输入数据的经压缩的版本生成的重构的数据之间的失真相关联的失真损失，以及与用于基于所述第二比特流的大小来发送所述经更新的模型参数的所述经压缩的版本的速率相关联的模型速率损失。

30. 一种其上存储有指令的非暂时性计算机可读介质，所述指令在由一个或多个处理器执行时，使所述一个或多个处理器用于：

通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩；

确定用于所述神经网络压缩系统的一组更新，所述一组更新包括使用所述输入数据调谐的经更新的模型参数；

通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的所述第一比特流；

通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流；以及

输出所述第一比特流和所述第二比特流以传输到接收机。

## 使用机器学习系统的实例自适应图像和视频压缩

### 技术领域

[0001] 概括地说,本公开内容涉及数据压缩,并且更具体地说,本公开内容涉及使用机器学习系统来压缩图像和/或视频内容。

### 背景技术

[0002] 许多设备和系统允许图像/视频数据被处理并输出以供消费。数字图像/视频数据包括大量的数据,以满足对图像/视频质量、性能和特征不断增长的需求。例如,视频数据的消费者通常期望具有高保真度、高分辨率、高帧速率等的高质量视频。满足这些需求通常所需要的大量视频数据给处理和存储视频数据的通信网络和设备带来了沉重的负担。视频编解码(coding)技术可以用于压缩视频数据。视频编解码的一个示例目标是将视频数据压缩为使用较低比特率的形式,同时避免视频质量的下降或使视频质量的下降最小。随着不断发展的视频服务变得可用以及对大量视频数据的需求不断增加,需要具有更好性能和效率的编解码技术。

### 发明内容

[0003] 在一些示例中,描述了用于使用一个或多个机器学习系统来进行数据压缩和/或解压缩的系统和技术。在一些示例中,提供了用于对图像/视频数据进行压缩和/或解压缩的机器学习系统。根据至少一个说明性示例,提供了一种对图像/视频数据进行压缩和/或解压缩的方法。在一些示例中,所述方法可以包括:通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的第二比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0004] 根据至少一个说明性示例,提供了一种用于对图像/视频数据进行压缩和/或解压缩的非暂时性计算机可读介质。在一些方面,所述非暂时性计算机可读介质可以包括指令,所述指令在由一个或多个处理器执行时,使所述一个或多个处理器用于:通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的第二比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0005] 根据至少一个说明性示例,提供了一种用于对图像/视频数据进行压缩和/或解压缩的装置。在一些方面,该装置可以包括其上存储有计算机可读指令的存储器以及一个或多个处理器,所述处理器被配置为:通过神经网络压缩系统接收输入数据以便通过所述神

神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的所述第一比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0006] 根据另一说明性示例,一种用于对图像/视频数据进行压缩和/或解压缩的装置可以包括用于进行以下操作的单元:通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的所述第一比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0007] 在一些方面中,上述方法、装置以及计算机可读介质可以生成包括所述第一比特流和所述第二比特流的经级联的比特流;以及将所述经级联的比特流发送到所述接收机。

[0008] 在一些示例中,所述第二比特流还包括所述潜在先验的经压缩的版本和所述模型先验的经压缩的版本。

[0009] 在一些情况下,生成所述第二比特流可以包括:通过所述神经网络压缩系统使用所述模型先验对所述潜在先验进行熵编码;以及通过所述神经网络压缩系统使用所述模型先验对所述经更新的模型参数进行熵编码。

[0010] 在一些示例中,所述经更新的模型参数包括解码器模型的一个或多个经更新的参数。在一些情况下,所述一个或多个经更新的参数可以是使用所述输入数据被调谐的。

[0011] 在一些示例中,所述经更新的模型参数包括编码器模型的一个或多个经更新的参数。在一些情况下,所述一个或多个经更新的参数可以是使用所述输入数据被调谐的。在一些情况下,所述第一比特流是通过所述神经网络压缩系统使用所述一个或多个经更新的参数生成的。

[0012] 在一些示例中,生成所述第二比特流可以包括:通过所述神经网络压缩系统使用所述一个或多个经更新的参数将所述输入数据编码为所述输入数据的潜在空间表示;以及通过所述神经网络压缩系统使用所述潜在先验将所述潜在空间表示熵编码为所述第一比特流。

[0013] 在一些方面中,上述方法、装置和计算机可读介质可以基于用于训练所述神经网络压缩系统的训练数据集来生成所述神经网络压缩系统的模型参数;使用所述输入数据来调谐所述神经网络压缩系统的所述模型参数;以及基于所述模型参数与经调谐的模型参数之间的差异来确定所述一组更新。

[0014] 在一些示例中,所述模型参数是基于所述输入数据、所述输入数据的所述经压缩的版本的比特大小、所述一组更新的比特大小、以及所述输入数据与从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真而被调谐的。

[0015] 在一些示例中,所述模型参数是基于所述输入数据以及发送所述一组更新的成本与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真的比率而被调谐的,所述成本基于所述一组更新的比特大小。

[0016] 在一些示例中,调谐所述模型参数可以包括:基于以下确定将一个或多个参数包括在所述经调谐的模型参数中:将所述一个或多个参数包括在所述经调谐的模型参数中伴随着以下各项中的至少一项的减少:所述输入数据的所述经压缩的版本的比特大小以及所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真。

[0017] 在一些示例中,确定用于所述神经网络压缩系统的所述一组更新可以包括:处理所述神经网络压缩系统处的所述输入数据;基于经处理的输入数据确定所述神经网络压缩系统的一个或多个损失;以及基于所述一个或多个损失来调谐所述神经网络压缩系统的模型参数,经调谐的模型参数包括用于所述神经网络压缩系统的所述一组更新。

[0018] 在一些情况下,所述一个或多个损失包括:与用于基于所述第一比特流的大小来发送所述输入数据的所述经压缩的版本的速率相关联的速率损失,与所述输入数据和从所述输入数据的经压缩的版本生成的重构的数据之间的失真相关联的失真损失,以及与用于基于所述第二比特流的大小来发送经更新的模型参数的所述经压缩的版本的速率相关联的模型速率损失。

[0019] 在一些示例中,所述接收机包括编码器。在一些方面,上述方法、装置和计算机可读介质可以通过所述编码器接收包括所述第一比特流和所述第二比特流的数据;通过所述解码器对基于所述第二比特流的所述经更新的模型参数的所述经压缩的版本进行解码;以及通过所述解码器使用所述一组经更新的参数,基于所述第一比特流中所述输入数据的所述经压缩的版本生成所述输入数据的经重构的版本。

[0020] 在一些方面,上述方法、装置和计算机可读介质可以通过降低速率失真和模型速率损失来训练所述神经网络压缩系统,其中,模型速率反映了用于发送模型更新的比特流的长度。

[0021] 在一些示例中,所述模型先验包括独立高斯网络先验、独立拉普拉斯网络先验和/或独立Spike和Slab网络先验。

[0022] 在一些方面,装置可以是下列各项或者可以是下列各项的一部分:相机(例如,IP相机)、移动设备(例如,移动电话或所谓的“智能电话”或其他类型的移动设备)、智能可穿戴设备、扩展现实设备(例如,虚拟现实(VR)设备、增强现实(AR)设备或混合现实(MR)设备)、个人计算机、膝上型计算机、服务器计算机、3D扫描仪、多相机系统,或者其他设备。在一些方面,所述装置包括用于捕获一个或多个图像的相机或多个相机。在一些方面,所述装置还包括用于显示一个或多个图像、通知和/或其他可显示数据的显示器。在一些方面,上述装置可以包括一个或多个传感器。

[0023] 本发明内容并不旨在标识所要求保护的发明主题的关键或重要特征,也不旨在单独用于确定要求保护的发明主题的范围。应当通过参考本专利的整个说明书的适当部分、任何或所有附图以及每项权利要求来理解发明主题。

[0024] 通过参考以下说明书、权利要求书和附图,前述以及其他特征和实施例将变得更加显而易见。

## 附图说明

[0025] 参考下列附图在下文中对本申请的说明性实施例进行了详细的描述,其中:

[0026] 图1是示出根据本公开内容的一些示例的图像处理系统的示例的图;

- [0027] 图2A是示出根据本公开内容的一些示例的全连接神经网络的示例的图；
- [0028] 图2B是示出根据本公开内容的一些示例的局部连接的神经网络的示例的图；
- [0029] 图2C是示出根据本公开内容的一些示例的卷积神经网络的示例的图；
- [0030] 图2D是示出根据本公开内容的一些示例的、用于从图像中识别视觉特征的深度卷积网络(DCN)的示例的图；
- [0031] 图3是示出根据本公开内容的一些示例的示例深度卷积网络(DCN)的框图；
- [0032] 图4是示出根据本公开内容的一些示例的、包括用于对视频内容进行压缩的发送设备和用于将接收到的比特流解压缩成视频内容的接收设备的系统的示例的图；
- [0033] 图5A和图5B是示出根据本公开内容的一些示例的、示例速率失真自动编码器系统的图；
- [0034] 图6是示出根据本公开内容的一些示例的、用于实例自适应数据压缩的示例神经网络压缩系统的图；
- [0035] 图7是示出根据本公开内容的一些示例的、使用模型先验微调(例如,实例适配)的神经网络压缩系统的示例架构的图；
- [0036] 图8是示出根据本公开内容的一些示例的、由使用模型先验微调的示例神经网络压缩系统实现的示例推测过程的图；
- [0037] 图9是示出根据本公开内容的一些示例的、由使用模型先验微调的示例神经网络压缩系统执行的编码和解码任务的图；
- [0038] 图10是示出根据本公开内容的一些示例的、在将被发送到接收机的数据点上微调的示例速率失真自动编码器模型以及没有在将被发送到接收机的数据点上微调的速率失真自动编码器模型的示例速率失真；
- [0039] 图11是示出根据本公开内容的一些示例的、用于使用适用于被压缩的输入数据(例如,针对被压缩的输入数据进行微调)的神经网络压缩系统的实例自适应压缩的示例过程1100的流程图；
- [0040] 图12是示出根据本公开内容的一些示例的、用于对一个或多个图像进行压缩的过程的示例的流程图；
- [0041] 图13是示出根据本公开内容的一些示例的、用于对一个或多个图像进行解压缩的过程的示例的流程图；以及
- [0042] 图14示出了根据本公开内容的一些示例的示例计算系统。

### 具体实施方式

[0043] 下文提供了本公开内容的某些方面和实施例。对于本领域技术人员来说将显而易见的是,这些方面和实施例中的一些可以独立地应用,并且它们中的一些也可以组合地应用。在下文的描述中,出于解释的目的,阐述了具体细节以便提供对本申请的实施例的透彻理解。然而,将显而易见的是,可以在不使用这些具体细节的情况下来实施各种实施例。附图和描述不旨在是限制性的。

[0044] 随后的描述仅提供示例实施例,而并不旨在限制本公开内容的范围、适用性或配置。相反,对示例实施例的随后描述将向本领域技术人员提供能够实施用于实现示例实施例的描述。应该理解的是,在不脱离所附权利要求书所阐述的本申请的精神和范围的前提

下,可以对元素的功能和布置进行各种改变。

[0045] 如上所述,数字图像和视频数据可以包括大量数据,特别是随着对高质量视频数据的需求持续增长。例如,图像和视频数据的消费者通常希望获得越来越高的视频质量,例如高保真度、分辨率、帧速率等。然而,满足此类需求所需的大量数据可能会给通信网络带来沉重的负担(例如高带宽和网络资源要求)以及对处理和存储视频数据的设备带来沉重的负担。因此,用于减少存储和/或传输图像和视频数据所需的数据量的压缩算法(也被称为编解码算法或工具)是有利的。

[0046] 可以使用各种技术来压缩图像数据和视频数据。图像数据的压缩是使用诸如联合图像专家组(JPEG)、更好的便携式图形(BPG)等算法完成的。近年来,基于神经网络的压缩方法在压缩图像数据方面显示出巨大的前景。可以根据特定的视频编解码标准来执行视频编解码。示例视频编解码标准包括高效视频编解码(HEVC)、基本视频编解码(EVC)、高级视频编解码(AVC)、运动图像专家组(MPEG)编解码和通用视频编解码(VVC)。然而,这样的传统图像和视频编解码技术会在执行解码之后在重构图像中造成伪影。

[0047] 在一些方面,本文描述了用于使用一个或多个机器学习系统来执行数据(例如,图像、视频、音频等)压缩和解压缩(也被称为编码和解码,统称为编解码)的系统、装置、过程(也被称为方法)以及计算机可读介质(在本文中统称为“系统和技术”)。一个或多个机器学习系统可以按照本文所述被训练,并用于执行数据压缩和/或解压缩,例如图像、视频和/或音频压缩和解压缩。本文描述的机器学习系统可以执行训练和压缩/解压缩技术,其产生高质量的数据输出。

[0048] 本文描述的系统和技术可以执行对任何类型的数据的压缩和/或解压缩。例如,在一些情况下,本文描述的系统和技术可以执行对图像数据的压缩和/或解压缩。作为另一示例,在一些情况下,本文描述的系统和技术可以执行对视频数据的压缩和/或解压缩。作为另一示例,在一些情况下,本文描述的系统和技术可以执行对音频数据的压缩和/或解压缩。为了简单、说明和解释的目的,本文描述的系统和技术是参考对图像数据(例如,图像、视频等)的压缩和/或解压缩来讨论的。然而,如上所述,本文中描述的概念也可以应用于其他模态,例如音频数据和任何其他类型的数据。

[0049] 在一些示例中,用于数据压缩和/或解压缩的机器学习系统可以在训练数据集(例如,图像、视频、音频等)上进行训练,并且可以针对将被发送到接收机并由接收机解码的数据被进一步微调(例如,训练、拟合)。在一些情况下,机器学习系统的编码器可以将使用将被发送到解码器并由解码器解码的数据而被微调的压缩模型的经更新的参数发送到解码器。在一些示例中,编码器可以在没有其他模型参数的情况下发送经更新的模型参数(和/或而不是发送完整的一组模型参数),以减少发送到解码器的数据的量和/或比特率。在一些情况下,可以使用模型先验对经更新的模型参数进行量化和压缩以减少发送到解码器的数据量和/或比特率。

[0050] 编码器和/或解码器使用的压缩模型可以推广到不同类型的数据。此外,通过针对正在被发送和解码的数据来对压缩模型进行微调,机器学习系统可以提高针对该特定数据的压缩和/或解压缩性能、质量和/或效率。在一些情况下,可以使用速率和失真损失以及额外的速率损失来训练机器学习系统的模型,该额外的速率损失反映和/或解释了发送经更新的模型参数时的额外开销和比特率。可以训练模型以使速率(例如,比特流的大小/长

度)、失真(例如,输入和重构输出之间的失真)和模型速率损失(例如,反映发送经更新的模型参数的成本的损失)最小化。在一些示例中,机器学习系统可以在速率、失真和模型速率(例如,发送经更新的模型参数所需的比特流的大小/长度)之间进行折衷。

[0051] 在一些示例中,机器学习系统可以包括一个或多个神经网络。机器学习(ML)是人工智能(AI)的子集。机器学习系统包括算法和统计模型,计算机系统可以使用该算法和统计模型通过依赖模式和推测来执行各种任务,而无需使用显式指令。ML系统的一个示例是神经网络(也被称为人工神经网络),该神经网络可以由互连的人工神经元(例如神经元模型)的群组组成。神经网络可用于各种应用和/或设备,例如图像分析和/或计算机视觉应用、互联网协议(IP)相机、物联网(IoT)设备、自动驾驶车辆、服务机器人等。

[0052] 神经网络中的各个节点可以通过获取输入数据并对数据执行简单操作来模拟生物神经元。对输入数据执行的简单操作的结果被选择性地传递给其他神经元。权重值与网络中的每个向量和节点相关联,并且这些值限制了输入数据与输出数据相关的方式。例如,可以将每个节点的输入数据乘以相对应的权重值,并且可以将乘积相加。乘积之和可以通过可选的偏差进行调谐,并且可以将激活函数应用于结果,产生节点的输出信号或“输出激活”(有时被称为激活图或特征图)。权重值最初可以由通过网络的训练数据的迭代流来确定(例如,权重值是在网络学习如何通过其典型输入数据特性识别特定类的训练阶段期间建立的)。

[0053] 存在不同类型的神经网络,例如深度生成神经网络模型(例如,生成式对抗网络(GAN)、循环神经网络(RNN)模型、多层感知器(MLP)神经网络模型、卷积神经网络(CNN)模型、自动编码器(AE)等。例如,GAN是一种生成神经网络,它可以学习输入数据中的模式,以便神经网络模型可以生成新的合成输出,这些新的合成输出合理地可能来自原始数据集。GAN可以包括一起操作的两个神经网络。其中一个神经网络(被称为生成式神经网络或生成器,表示为 $G(z)$ )生成合成输出,而另一个神经网络(被称为鉴别式神经网络或鉴别器,表示为 $D(X)$ )针对真实性来评估输出(输出是来自原始数据集,例如训练数据集,还是由生成器生成)。训练输入和输出可以包括图像作为说明性示例。生成器被训练以试图并且欺骗鉴别器,使其确定由生成器生成的合成图像是来自数据集的真实图像。训练过程继续进行,并且生成器在生成看起来像真实图像的合成图像方面变得更好。鉴别器继续在合成图像中发现缺陷,并且生成器找出鉴别器正在查看的内容以确定图像中的缺陷。一旦网络经过训练,生成器就能够产生鉴别器无法与真实图像区分开来的逼真图像。

[0054] RNN针对以下原理工作:保存层的输出并将该输出反馈给输入,以帮助预测该层的结果。在MLP神经网络中,数据可以被馈送到输入层,并且一个或多个隐藏层为数据提供抽象级别。然后可以基于抽象数据对输出层进行预测。MLP可以特别适用于为输入分配类别或标签的分类预测问题。卷积神经网络(CNN)是一种类型的前馈人工神经网络。CNN可以包括人工神经元的集合,每个人工神经元都具有相应的感受域(例如,输入空间的局部区域)并且共同平铺输入空间。CNN有许多应用,包括模式识别和分类。

[0055] 在分层神经网络架构中(当存在多个隐藏层时被称为深度神经网络),第一层人工神经元的输出成为第二层人工神经元的输入,第二层人工神经元的输出成为第三层人工神经元的输入,以此类推。卷积神经网络可以被训练以识别特征的层次结构。卷积神经网络架构中的计算可以分布在一组处理节点上,这些节点可以被配置在一个或多个运算链中。这

些多层架构可以一次训练一层,并且可以使用后向传播进行微调。

[0056] 自动编码器(AE)可以以无监督的方式学习有效的数据编解码。在一些示例中,AE可以通过训练网络来忽略信号噪声而学习一组数据的表示(例如,数据编解码)。AE可以包括编码器和解码器。编码器可以将输入数据映射到代码中,而解码器可以将代码映射到输入数据的重构。在一些示例中,可以训练速率失真自动编码器(RD-AE)以使数据点的数据集(例如图像和/或视频数据点)上的平均速率失真损失最小化。在一些情况下,RD-AE可以在推测时进行前向传递以编码新的数据点。

[0057] 在一些情况下,可以在要被发送到接收机(例如解码器)的数据上对RD-AE进行微调。在一些示例中,通过在数据点上微调RD-AE,RD-AE可以获得高压缩(例如,速率/失真)性能。与RD-AE相关联的编码器可以将RD-AE模型或RD-AE模型的一部分发送到接收机(例如,解码器),以供接收机对包括由编码器发送的压缩数据的比特流进行解码。

[0058] 在一些情况下,AE模型可能很大,这可能会增加比特率和/或降低速率失真增益。在一些示例中,可以使用模型先验(例如,RDM-AE先验)对RD-AE模型进行微调。模型先验可以被定义并用于生成针对接收机(例如,解码器)的模型更新,以用于实现模型以对发送的数据进行解压缩。模型先验可以通过在模型先验下生成的模型更新来减少被发送到解码器的数据量。在一些示例中,模型先验可以被设计为降低用于发送模型更新的成本。例如,模型先验可用于减少和/或限制模型更新的比特率开销和/或产生较小的模型更新。在一些情况下,随着更多参数被微调,模型更新的比特率开销可能会增加。在一些示例中,可以减少被微调的参数,这也可以减少或限制比特率开销。

[0059] 在一些情况下,可以使用RDM-AE损失对模型进行微调。可以将损失项添加到用于模型速率的比特率中。添加的损失项可以补偿模型更新上“花费”的比特。例如,在微调期间,模型更新上“花费”的任何比特都可以通过速率失真(R/D)的改进得到补偿。在一些示例中,在微调期间,在模型更新上“花费”的任何比特都可以通过至少同样多的速率失真的改进得到补偿。

[0060] 在一些情况下,可以如本文进一步描述的那样来改进模型先验的设计。在一些说明性示例中,模型先验设计可以包括独立的高斯模型先验。在其他说明性示例中,模型先验设计可以包括独立的拉普拉斯模型先验。在其他说明性示例中,模型先验设计可以包括独立的Spike和Slab先验。在一些说明性示例中,可以联合训练模型先验和全局AE模型。在一些说明性示例中,模型先验可以包括由神经网络学习的复杂依赖关系。

[0061] 图1是示出了根据本公开内容的一些示例的图像处理系统100的示例的图。在一些情况下,图像处理系统100可以包括中央处理单元(CPU)102或多核CPU,该多核CPU被配置为执行本文描述的一个或多个功能。变量(例如,神经信号和突触权重)、与计算设备相关联的系统参数(例如,具有权重的神经网络)、延迟、频率仓信息、任务信息以及其他信息,可以存储在与神经处理单元(NPU)108相关联的存储器块中、存储在与CPU 102相关联的存储器块中、存储在与图形处理单元(GPU)104相关联的存储器块中、存储在与数字信号处理器(DSP)106相关联的存储器块中,存储在存储器块118中,或分布在多个块中。在CPU 102处执行的指令可以从与CPU 102和/或存储器块118相关联的程序存储器加载。

[0062] 图像处理系统100可以包括为特定功能定制的附加处理块,例如GPU 104;DSP 106;连接块110,其可以包括第五代(5G)连接、第四代长期演进(4G LTE)连接、Wi-Fi连接、

USB连接、蓝牙连接等；和/或可以例如检测和识别特征的多媒体处理器112。在一种实施方式中，NPU 108在CPU102、DSP 106和/或GPU 104中实施。图像处理系统100还可以包括传感器处理器114、一个或多个图像信号处理器 (ISP) 116和/或存储装置120。在一些示例中，图像处理系统100可以基于ARM指令集。

[0063] 图像处理系统100可以是计算设备或多个计算设备的一部分。在一些示例中，图像处理系统100可以是电子设备(或设备)的一部分，例如相机系统(例如，数码相机、IP相机、摄像机、安全相机等)、电话系统(例如，智能手机、蜂窝电话、会议系统等)、桌面式计算机、XR设备(例如，头戴式显示器等)、智能可穿戴设备(例如，智能手表、智能眼镜等)、膝上型计算机或笔记本电脑、平板电脑、机顶盒、电视机、显示设备、数字媒体播放器、游戏控制台、视频流式传输设备、无人机、车内计算机、片上系统(SOC)、物联网(IoT)设备或任何其他合适的电子设备。

[0064] 尽管示出图像处理系统100包括某些组件，但是本领域普通技术人员将理解，图像处理系统100可以包括比图1所示的组件更多或更少的组件。例如，图像处理系统100在一些情况下还可以包括一个或多个存储器设备(例如RAM、ROM、高速缓存器等)、一个或多个网络接口(例如，有线和/或无线通信接口等)、一个或多个显示设备和/或图1中未示出的其他硬件或处理设备。下面针对图14描述可以用图像处理系统100实现的计算设备和硬件组件的说明性示例。

[0065] 图像处理系统100和/或其组件可以被配置为使用本文描述的机器学习系统和/或技术来执行压缩和/或解压缩(也被称为编码和/或解码，统称为图像编解码)。在一些情况下，图像处理系统100和/或其组件可以被配置为使用本文描述的技术来执行图像或视频压缩和/或解压缩。在一些示例中，机器学习系统可以利用深度学习神经网络架构来执行对图像、视频和/或音频数据的压缩和/或解压缩。通过使用深度学习神经网络架构，机器学习系统可以提高对设备上内容的压缩和/或解压缩的效率和速度。例如，使用所描述的压缩和/或解压缩技术的设备可以使用基于机器学习的技术有效地压缩一个或多个图像，可以将经压缩的一个或多个图像发送到接收设备，并且接收设备可以使用本文描述的基于机器学习的技术更有效地对这一个或多个经压缩的图像进行解压缩。本文中使用的，图像可以指代与帧序列(例如，视频)相关联的静止图像和/或视频帧。

[0066] 如上所述，神经网络是机器学习系统的示例。神经网络可以包括输入层、一个或多个隐藏层，以及输出层。从输入层的输入节点提供数据，由一个或多个隐藏层的隐藏节点执行处理，并通过输出层的输出节点产生输出。深度学习网络通常包括多个隐藏层。神经网络的每一层都可以包括特征图或激活图，特征图或激活图可以包括人工神经元(或节点)。特征图可以包括滤波器、内核等。节点可以包括一个或多个权重，用于指示一个或多个层的节点的重要性。在一些情况下，深度学习网络可以具有一系列许多隐藏层，其中，早期层用于确定输入的简单和低级别特征，而后面的层则建立更复杂和抽象特性的层次结构。

[0067] 深度学习架构可以学习特征的层次结构。例如，如果用视觉数据呈现，则第一层可以学习以识别输入流中相对简单的特征，例如边。在另一个示例中，如果用听觉数据呈现，则第一层可以学习以识别特定频率中的频谱功率。第二层以第一层的输出作为输入，可以学习识别特征的组合，例如视觉数据的简单形状或听觉数据的声音组合。例如，更高的层可以学习在视觉数据中表示复杂的形状或在听觉数据中表示词语。更高的层可以学习识别常

见的视觉对象或口语短语。

[0068] 因此,当应用于具有自然层次结构的问题时,深度学习架构可以表现得尤其出色。例如,机动车辆的分类可以受益于首先学习识别车轮、挡风玻璃和其他特征。这些特征可以在更高层以不同的方式进行组合,以识别汽车、卡车和飞机。

[0069] 神经网络可以被设计成具有多种连接模式。在前馈网络中,信息从较低层传递到较高层,其中,给定层中的每个神经元都与较高层中的神经元进行通信。如上所述,可以在前馈网络的连续层中建立分层表示。神经网络也可以具有循环或反馈(也被称为自上而下)连接。在循环连接中,来自给定层中的神经元的输出可以被传送到同一层中的另一个神经元。循环架构可以有助于识别跨越按顺序被传递给神经网络的输入数据块中的多于一个输入数据块的模式。从给定层中的神经元到较低层中的神经元的连接被称为反馈(或自上而下)连接。当对高级别概念的识别可能有助于区分输入的特定低级别特征时,具有许多反馈连接的网络可能会有所帮助。

[0070] 神经网络各层之间的连接可以是全连接的,也可以是局部连接的。图2A示出了全连接的神经网络202的示例。在全连接的神经网络202中,第一层中的神经元可以将其输出传送到第二层中的每个神经元,使得第二层中的每个神经元将接收到来自第一层中每个神经元的输入。图2B示出了局部连接的神经网络204的示例。在局部连接的神经网络204中,第一层中的神经元可以连接到第二层中有限数量的神经元。更一般地,局部连接的神经网络204的局部连接层可以被配置为使得层中的每个神经元将具有相同或相似的连接模式,但其中,连接强度可以具有不同的值(例如,210、212、214和216)。局部连接的模式可能会在较高层中产生空间上不同的感受域,因为给定区域中的较高层神经元可以接收输入,该输入通过训练被调谐到网络的总输入的受限部分的属性。

[0071] 局部连接的神经网络的一个示例是卷积神经网络。图2C示出了卷积神经网络206的示例。卷积神经网络206可以被配置为使得与第二层中的每个神经元的输入相关联的连接强度是共享的(例如,208)。卷积神经网络可能非常适合于输入的空间位置是有意义的问题。根据本公开内容的方面,卷积神经网络206可用于执行视频压缩和/或解压缩的一个或多个方面。

[0072] 一种类型的卷积神经网络是深度卷积网络(DCN)。图2D示出了DCN 200的详细示例,该DCN 200被设计为从图像捕获设备230(例如车载相机)输入的图像226中识别视觉特征。当前示例的DCN 200可以被训练以识别交通标志和交通标志上提供的数字。当然,可以针对其他任务对DCN 200进行训练,例如识别车道标记或识别交通信号灯。

[0073] 可以通过监督学习对DCN 200进行训练。在训练期间,可以向DCN 200呈现图像,例如限速标志的图像226,并且然后可以计算前向传递以产生输出222。DCN 200可以包括特征提取部分和分类部分。在接收到图像226时,卷积层232可以将卷积核(未示出)应用于图像226以生成第一组特征图218。作为示例,卷积层232的卷积核可以是生成28x28特征图的5x5核。在本示例中,因为在第一组特征图218中生成了四个不同的特征图,因此在卷积层232处将四个不同的卷积核应用于图像226。卷积核也可以被称为滤波器或卷积滤波器。

[0074] 第一组特征图218可以由最大池化层(未示出)二次采样以生成第二组特征图220。最大池化层减小了第一组特征图218的大小。也就是说,第二组特征图220的大小,例如14x14,小于第一组特征图218的大小,例如28x28。减小的大小为后续层提供了类似的信息,

同时减少了存储器消耗。第二组特征图220可以通过一个或多个后续卷积层(未示出)被进一步卷积以生成一个或多个后续组的特征图(未示出)。

[0075] 在图2D的示例中,对第二组特征图220进行卷积以生成第一特征向量224。此外,对第一特征向量224进行进一步卷积以生成第二特征向量228。第二特征向量228的每个特征可以包括与图像226的可能特征相对应的数字,例如“符号”、“60”和“100”。Softmax函数(未示出)可以将第二特征向量228中的数字转换为概率。这样,DCN 200的输出222是图像226包括一个或多个特征的概率。

[0076] 在本示例中,输出222中“符号”和“60”的概率高于输出222中的其他(例如,“30”、“40”、“50”、“70”、“80”、“90”和“100”)的概率。在训练之前,由DCN 200产生的输出222可能是不正确的。因此,可以计算输出222和目标输出之间的误差。目标输出是图像226的地面真值(例如,“符号”和“60”)。然后可以调整DCN 200的权重,使得DCN 200的输出222更接近于目标输出。

[0077] 为了调整权重,学习算法可以计算权重的梯度向量。梯度可以指示如果调整权重时误差将增加或减少的量。在顶层,梯度可以直接对应于连接倒数第二层中激活的神经元和输出层中的神经元的权重值。在较低层中,梯度可以取决于权重的值和较高层的计算的误差梯度。然后可以调整权重以减少误差。这种调整权重的方式可以被称为“后向传播”,因为它涉及通过神经网络的“后向传递”。

[0078] 在实践中,可以在少量示例来计算权重的误差梯度,使计算出的梯度接近真实的误差梯度。这种近似方法可以被称为随机梯度下降。可以重复随机梯度下降,直到整个系统的可实现错误率停止下降或直到错误率达到目标水平。在学习之后,可以向DCN呈现新图像,并且通过网络的前向传递可以产生可以被认为是DCN的推测或预测的输出222。

[0079] 深度置信网络(DBN)是包括多层隐藏节点的概率模型。DBN可用于提取训练数据集的分层表示。DBN可以通过对受限玻尔兹曼机(RBM)的层进行堆叠来获得。RBM是一种类型的人工神经网络,其可以通过一组输入学习概率分布。由于RBM可以在没有关于每个输入应归入的类别的信息的情况下学习概率分布,因此RBM通常用于无监督学习。使用混合的无监督范式和受监督范式,DBN的底部RBM可以以无监督的方式被训练,并可以用作特征提取器,而顶部RBM可以以受监督的方式被训练(在来自前一层的输入和目标类的联合分布上)并且可以用作分类器。

[0080] 深度卷积网络(DCN)是卷积网络的网络,其被配置有额外的池化层和归一化层。DCN在许多任务上都达到了最先进的性能。可以使用监督学习对DCN进行训练,其中,输入和输出目标二者对于许多示例都是已知的,并用于通过使用梯度下降方法来修改网络的权重。

[0081] DCN可以是前馈网络。此外,如上所述,从DCN的第一层中的神经元到下一个更高层中的神经元的群组的连接在第一层中的神经元之间共享。DCN的前馈和共享连接可用于快速处理。例如,DCN的计算负担可能比包含循环或反馈连接的类似大小的神经网络的计算负担小得多。

[0082] 卷积网络的每一层的处理可以被认为是空间不变的模板或基础投影。如果输入首先被分解为多个通道,例如彩色图像的红、绿和蓝通道,那么在该输入上被训练的卷积网络可以被认为是三维的,其中,两个空间维度沿着图像的轴,第三维捕获颜色信息。卷积连接

的输出可以被认为在后续层中形成特征图,其中,特征图的每个元素(例如,220)接收来自前一层中的一系列神经元(例如,特征图218)以及来自多个通道中的每个通道的输入。特征图中的值可以用非线性进一步处理,例如校正,  $\max(0, x)$ 。来自相邻神经元的值可以进一步池化,这对应于下采样,并且可以提供额外的局部不变性和维度降低。

[0083] 图3是示出深度卷积网络350的示例的框图。深度卷积网络350可以包括基于连通性和权重共享的多个不同类型的层。如图3所示,深度卷积网络350包括卷积块354A、354B。卷积块354A、354B中的每个可以被配置有卷积层(CONV) 356、归一化层(LNorm) 358和最大池化层(MAX POOL) 360。

[0084] 卷积层356可包括一个或多个卷积滤波器,其可以应用于输入数据352以生成特征图。尽管仅示出了两个卷积块354A、354B,但本公开内容不限于此,而是可以根据设计偏好将任意数量的卷积块(例如,块354A、354B)包括在深度卷积网络350中。归一化层358可以对卷积滤波器的输出进行归一化。例如,归一化层358可以提供白化或横向抑制。最大池化层360可以在空间上提供下采样聚合以用于局部不变性和维度下降。

[0085] 例如,深度卷积网络的并行滤波器组可以加载到图像处理系统100的CPU 102或GPU 104上以实现高性能和低功耗。在替代实施例中,并行滤波器组可以加载在图像处理系统100的DSP 106或ISP 116上。此外,深度卷积网络350可以访问可能存在于图像处理系统100上的其他处理块,例如传感器处理器114。

[0086] 深度卷积网络350还可以包括一个或多个全连接层,例如层362A(标记为“FC1”)和层362B(标记为“FC2”)。深度卷积网络350还可以包括逻辑回归(LR)层364。在深度卷积网络350的每一层356、358、360、362、364之间是待更新的权重(未示出)。每一层(例如,356、358、360、362、364)的输出可以作为深度卷积网络350中层(例如,356、358、360、362、364)的后续一个层的输入,以从在卷积块354A中的第一个卷积块354A处提供的输入数据352(例如,图像、音频、视频、传感器数据和/或其他输入数据)学习分层特征表示。深度卷积网络350的输出是输入数据352的分类得分366。分类得分366可以是一组概率,其中每个概率是输入数据包括来自一组特征的特征的概率。

[0087] 图像和视频内容可以被存储在设备中和/或可以在设备之间共享。例如,图像和视频内容可以上传到媒体托管服务和共享平台,并可以被发送到各种设备。录制未压缩的图像和视频内容通常会导致大文件大小,该大文件大小随着图像和视频内容的分辨率的增加而大大增加。例如,以1080p/24格式录制的未压缩的每通道16比特视频(例如,分辨率为1920像素宽和1080像素高,其中,每秒捕获24帧)可以占用每帧12.4兆字节,或每秒297.6兆字节。以每秒24帧的4K分辨率录制的未压缩的每通道16比特视频可能占用每帧49.8兆字节,或每秒1195.2兆字节。

[0088] 因为未压缩的图像和视频内容可能会导致大文件,这些大文件可能涉及用于物理存储的相当大的存储器和用于传输的相当大的带宽,因此可以利用一些技术来对此类视频内容进行压缩。例如,为了减小图像内容的大小,并且从而减少存储图像内容所涉及的存储量以及传送视频内容所涉及的带宽量,可以将各种压缩算法应用于图像和视频内容。

[0089] 在一些情况下,可以使用先验定义的压缩算法来压缩图像内容,压缩算法例如联合图像专家组(JPEG)、更好的便携式图形(BPG)等。例如,JPEG是一种基于离散余弦变换(DCT)的有损压缩形式。例如,执行图像的JPEG压缩的设备可以将图像转换为最佳颜色空间

(例如, YCbCr颜色空间, 包括亮度(Y)、色度-蓝色(Cb)、色度-红色(Cr)), 可以通过将像素的群组平均在一起来对色度分量进行下采样, 并且可以将DCT函数应用于像素块以去除冗余图像数据, 并且从而压缩图像数据。压缩基于识别图像内部的相似区域并将这些区域转换为相同的颜色代码(基于DCT函数)。也可以使用先验定义的压缩算法来对视频内容进行压缩, 该压缩算法例如电影专家组(MPEG)算法、H.264或高效视频编解码算法。

[0090] 这些先验定义的压缩算法可能能够保留原始图像和视频内容中的大部分信息, 并且可以基于信号处理和信息论的思想进行先验定义。然而, 虽然这些预定义的压缩算法可能通常适用(例如, 适用于任何类型的图像/视频内容), 但压缩算法可能不会考虑内容的相似性、视频捕获和传输的新分辨率或帧速率、非自然图像(例如, 雷达图像或经由各种传感器捕获的其他图像)等。

[0091] 先验定义的压缩算法被认为是有损压缩算法。在输入图像(或视频帧)的有损压缩中, 输入图像不能被编码并且然后被解码/重构, 以便重构精确的输入图像。相反, 在有损压缩中, 输入图像的近似版本是在对经压缩的输入图像的解码/重构之后生成的。有损压缩会导致比特率降低, 但会以失真为代价, 从而导致重构图像中出现伪影。因此, 在有损压缩系统中存在速率失真折衷。对于某些压缩方法(例如JPEG、BPG等), 基于失真的伪影可能采取块状或其他伪影的形式。在一些情况下, 可以使用基于神经网络的压缩, 并且可以对图像数据和视频数据进行高质量压缩。在一些情况下, 模糊和颜色偏移是伪影的示例。

[0092] 每当比特率低于输入数据的真实熵时, 可能很难或不可能重构准确的输入数据。然而, 从数据的压缩/解压缩中实现的失真/损失这一事实并不意味着重构的图像或帧不一定具有伪影。实际上, 可以将经压缩的图像重构为另一个类似但不同的具有高视觉质量的图像。

[0093] 在如前所述, 本文描述的系统和技术可以使用一个或多个机器学习(ML)系统来执行压缩和解压缩。在一些示例中, 机器学习技术可以提供产生高质量视觉输出的图像和/或视频压缩。在一些示例中, 本文描述的这些系统和技术可以使用诸如速率失真自动编码器(RD-AE)的深度神经网络来执行内容(例如, 图像内容、视频内容、音频内容等)的压缩和解压缩。深度神经网络可以包括将图像映射到潜在代码空间(例如, 包括一组代码 $z$ )的自动编码器(AE)。潜在代码空间可以包括编码器和解码器使用的代码空间, 并且其中内容已经被编码成代码 $z$ 。代码(例如, 代码 $z$ )也可以被称为潜在量、潜在变量或潜在表示。深度神经网络可以包括概率模型(也被称为模型先验或代码模型), 该概率模型可以对来自潜在代码空间的代码 $z$ 进行无损压缩。概率模型可以在代码集 $z$ 上生成概率分布, 该代码集 $z$ 可以表示基于输入数据的经编码的数据。在一些情况下, 概率分布可以表示为 $(P(z))$ 。

[0094] 在一些示例中, 深度神经网络可以包括算术编码器, 其基于概率分布 $P(z)$ 和/或代码集 $z$ 来生成包括要输出的经压缩的数据的比特流。包括经压缩的数据的比特流可以被存储和/或可以被发送到接收设备。接收设备可以使用例如算术解码器、概率(或代码)模型以及AE的解码器来执行逆过程以便对比特流进行解码或解压缩。生成包括经压缩的数据的比特流的设备在从存储中取回经压缩的数据时也可以执行类似的解码/解压缩过程。可以执行类似的技术来对经更新的模型参数进行压缩/编码和解压缩/解码。

[0095] 在一些示例中, 可以训练和操作RD-AE以执行作为多速率AE(包括高速率和低速率操作)。例如, 由多速率AE的编码器生成的潜在代码空间可以被划分为两个或更多个块(例

如,代码 $z$ 被划分为块 $z_1$ 和 $z_2$ )。在高速率操作中,多速率AE可以发送基于整个潜在空间(例如代码 $z$ ,包括 $z_1$ 、 $z_2$ 等)的比特流,接收设备可以使用该比特流来对数据进行解压缩,类似于上文针对RD-AE描述的操作。在低速率操作中,发送到接收设备的比特流基于潜在空间的子集(例如,块 $z_1$ 而不是 $z_2$ )。接收设备可以基于被发送的子集来推测潜在空间的剩余部分,并且可以使用潜在空间的子集和潜在空间的推测的剩余部分来生成重构的数据。

[0096] 通过使用RD-AE或多速率AE来对内容进行压缩(和解压缩),编码和解码机制可以适用于各种用例。基于机器学习的压缩技术可以生成具有高质量和/或降低比特率的经压缩的内容。在一些示例中,可以训练RD-AE以使数据点的数据集(例如图像和/或视频数据点)上的平均速率失真损失最小化。在一些情况下,RD-AE还可以针对要发送到接收机并由接收机解码的特定数据点进行微调。在一些示例中,通过在数据点上微调RD-AE,RD-AE可以获得高压缩(速率/失真)性能。与RD-AE相关联的编码器可以将AE模型或AE模型的一部分发送到接收机(例如,解码器)以对比特流进行解码。

[0097] 在一些情况下,神经网络压缩系统可以从(经量化的)潜在表示重构输入实例(例如,输入图像、视频、音频等)。神经网络压缩系统还可以使用先验来对潜在表示进行无损压缩。在一些情况下,神经网络压缩系统可以确定测试时数据分布是已知的且熵相对较低(例如,观看静态场景的相机、自动驾驶汽车中的行车记录仪等),并且可以被微调或者被适应于这样的分布。微调或适应可以造成改进的速率/失真(RD)性能。在一些示例中,神经网络压缩系统的模型可以适用于待压缩的单个输入实例。神经网络压缩系统可以提供模型更新,在一些示例中,可以使用参数空间先验以及潜在表示对模型更新进行量化和压缩。

[0098] 微调可以考虑模型量化的影响以及发送模型更新所导致的额外成本。在一些示例中,可以使用RD损失以及附加模型速率项 $M$ 来对神经网络压缩系统进行微调,该模型速率项 $M$ 测量在模型先验下发送模型更新所需的比特数,从而导致组合的RDM损失。

[0099] 图4是示出根据本公开内容的一些示例的、包括发送设备410和接收设备420的系统400的图。发送设备410和接收设备420中的每个在一些情况下可以被称为RD-AE。发送设备410可以压缩图像内容,并且可以存储经压缩的图像内容和/或将经压缩的图像内容发送到接收设备420以进行解压缩。接收设备420可以对经压缩的图像内容进行解压缩,并且可以在接收设备420上输出经解压缩的图像内容(例如,用于显示、编辑等)和/或可以将经解压缩的图像内容输出到连接到接收设备420的其他设备(例如,电视、移动设备或其他设备)。在一些情况下,接收设备420可以通过以下操作变成发送设备:对图像内容进行压缩(使用编码器422)并将经压缩的图像内容存储和/或发送到另一个设备,例如发送设备410(在这种情况下,发送设备410将变成接收设备)。尽管在本文中针对图像压缩和解压缩描述了系统400,但本领域技术人员将理解:系统400可以使用本文描述的技术来对视频内容进行压缩和解压缩。

[0100] 如图4所示,发送设备410包括图像压缩管线,而接收设备420包括图像比特流解压缩管线。根据本公开内容的方面,发送设备410中的图像压缩管线和接收设备420中的比特流解压缩管线通常使用一个或多个神经网络来压缩图像内容和/或将接收的比特流解压缩成图像内容。发送设备410中的图像压缩管线包括自动编码器401、代码模型404和算术编码器406。在一些实施方式中,算术编码器406是可选的并且在一些情况下可以被省略。接收设备420中的图像解压缩管线包括自动编码器421、代码模型424和算术解码器426。在一

些实施方式中,算术解码器426是可选的并且在一些情况下可以被省略。发送设备410的自动编码器401和代码模型404在图4中被示为机器学习系统,该机器学习系统先前已经被训练并且因此被配置用于在训练的机器学习系统的推测或操作期间执行操作。自动编码器421、代码模型424以及完成模型425也被示为先前已经被训练的机器学习系统。

[0101] 自动编码器401包括编码器402和解码器403。编码器402可以通过将未经压缩的图像内容的一个或多个图像中的像素映射到潜在代码空间(包括代码 $z$ )来对接收到的未经压缩的图像内容执行有损压缩。通常,编码器402可以被配置为使得表示经压缩(或编码)的图像的代码 $z$ 是离散的或二进制的。这些代码可以基于随机扰动技术、软向量量化,或者可以生成不同代码的其他技术来生成。在一些方面,自动编码器401可以将未压缩的图像映射到具有可压缩(低熵)分布的代码。这些代码的交叉熵可以接近于预定义或学习的先验分布。

[0102] 在一些示例中,可以使用卷积架构来实现自动编码器401。例如,在一些情况下,自动编码器401可以被配置为二维卷积神经网络(CNN),以便自动编码器401学习用于将图像内容映射到潜在代码空间的空间滤波器。在系统400用于对视频数据进行编解码的示例中,自动编码器401可以被配置为三维CNN,使得自动编码器401学习用于将视频映射到潜在代码空间的时空滤波器。在这样的网络中,自动编码器401可以根据关键帧(例如,标记帧序列开始的初始帧,其中序列中的后续帧被描述为相对于序列中的初始帧的差异)、关键帧与视频中其他帧之间的翘曲(或差异)以及残差因子来对视频进行编码。在其他方面,自动编码器401可以实现为以先前帧、帧之间的残差因子为条件的二维神经网络,并通过堆叠通道或包括循环层来进行调节。

[0103] 自动编码器401的编码器402可以接收第一图像(在图4中标示为图像 $x$ )作为输入,并且可以将第一图像 $x$ 映射到潜在代码空间中的代码 $z$ 。如上所述,编码器402可以被实现为二维卷积网络,使得潜在代码空间在每个 $(x, y)$ 位置处具有描述以该位置为中心的图像 $x$ 的块的向量。 $x$ 坐标可以表示图像 $x$ 的该块中的水平像素位置, $y$ 坐标可以表示图像 $x$ 的该块中的垂直像素位置。当对视频数据进行编解码时,潜在代码空间可以具有 $t$ 变量或位置,其中, $t$ 变量表示视频数据的块中的时间戳(除了空间 $x$ 和 $y$ 坐标之外)。通过使用水平和垂直像素位置的两个维度,向量可以描述图像 $x$ 中的图像补丁。

[0104] 自动编码器401的解码器403随后可以对代码 $z$ 进行解压缩以获得第一图像 $x$ 的重构 $\hat{x}$ 。通常,重构 $\hat{x}$ 可以是未经压缩的第一图像 $x$ 的近似,而不需要是第一图像 $x$ 的精确副本。在一些情况下,重构的图像 $\hat{x}$ 可以被输出为经压缩的图像文件,以存储在发送设备中。

[0105] 代码模型404接收表示经编码的图像或其部分的代码 $z$ ,并在可用于表示代码 $z$ 的一组经压缩的码字上生成概率分布 $P(z)$ 。在一些示例中,代码模型404可以包括概率自回归生成模型。在一些情况下,可以针对其生成概率分布的代码包括基于算术编解码器406来控制比特分配的经学习的分布。例如,使用算术编解码器406,可以单独预测第一代代码 $z$ 的压缩代码;可以基于第一代代码 $z$ 的压缩代码来预测第二代代码 $z$ 的压缩代码;可以基于第一代代码 $z$ 和第二代代码 $z$ 的压缩代码来预测第三代代码 $z$ 的压缩代码,以此类推。压缩代码通常表示待压缩的给定图像的不同时空块。

[0106] 在一些方面, $z$ 可以被表示为三维张量。张量的三个维度可以包括特征通道维度,以及高度和宽度空间维度(例如,表示为代码 $z_{c,w,h}$ )。每个代码 $z_{c,w,h}$ (表示由通道和水平以及垂直位置索引的代码)都可以基于先前代码来进行预测,先前代码可以是固定排序的和

理论上任意排序的代码。在一些示例中,可以通过从头到尾对给定图像文件进行分析并以光栅扫描顺序对图像中的每个块分析来生成代码。

[0107] 代码模型404可以使用概率自回归模型来学习输入代码 $z$ 的概率分布。概率分布可以以其先前的值为条件(如上所述)。在一些示例中,概率分布可以由以下公式表示:

$$[0108] \quad P(z) = \prod_{c=0}^C \prod_{w=0}^W \prod_{h=0}^H p(z_{c,w,h} | z_{0:c,0:w,0:h}) \quad \text{公式 (1)},$$

[0109] 其中 $c$ 是所有图像通道 $C$ (例如R、G和B通道,Y、Cb和Cr通道或其他通道)的通道索引, $w$ 是总图像帧宽度 $W$ 的宽度索引,而 $h$ 是总图像帧高度 $H$ 的高度索引。

[0110] 在一些示例中,可以通过因果卷积的全卷积神经网络来预测概率分布 $P(z)$ 。在一些方面,卷积神经网络的每一层的内核可以被屏蔽,使得卷积网络在计算概率分布时知晓先前值 $z_{0:c,0:w,0:h}$ 并且可能不知晓其他值。在一些方面,卷积网络的最后一层可以包括确定潜在空间中的代码适用于输入值的概率的softmax函数(例如,给定代码可以用于压缩给定输入的可能性)。

[0111] 算术编码器406使用由代码模型404生成的概率分布 $P(z)$ 来生成与代码 $z$ 的预测相对应的比特流415(在图4中示为“0010011……”)。代码 $z$ 的预测可以表示为在一组可能的代码上生成的概率分布 $P(z)$ 中具有最高概率得分的代码。在一些方面,算术编码器406可以基于代码 $z$ 的预测的准确性和由自动编码器401生成的实际代码 $z$ 来输出可变长度的比特流。例如,如果预测准确,则比特流415可以对应于短码字,而随着代码 $z$ 和代码 $z$ 的预测之间的差异幅度增加,比特流415可以对应于更长的码字。

[0112] 在一些情况下,比特流415可以由算术编码器406输出以存储在经压缩的图像文件中。还可以输出比特流415以传输到请求设备(例如,接收设备420,如图4所示)。通常,算术编码器406输出的比特流415可以无损地对 $z$ 进行编码,使得可以 $z$ 在对经压缩的图像文件应用的解压缩过程期间被准确地恢复。

[0113] 由算术编码器406生成并从发送设备410发送的比特流415可以被接收设备420接收。发送设备410和接收设备420之间的传输可以使用各种合适的有线或无线通信技术中的任何一种来发生。发送设备410和接收设备420之间的通信可以是直接的或者可以通过一个或多个网络基础设施组件(例如,基站、中继站、移动站、网络集线器、路由器和/或其他网络基础设施组件)来执行。

[0114] 如图所示,接收设备420可以包括算术解码器426、代码模型424以及自动编码器421。自动编码器421包括编码器422和解码器423。对于给定的输入,解码器423可以产生与解码器403相同或相似的输出。虽然自动编码器421被示为包括编码器422,但在解码过程期间不需要使用编码器422来从接收自发送设备410的代码 $z$ 中获得 $\hat{x}$ (例如,在发送设备410处压缩的原始图像 $x$ 的近似)。

[0115] 接收到的比特流415可以被输入到算术解码器426中以从比特流中获得一个或多个代码 $z$ 。算术解码器426可以基于由代码模型424在一组可能的代码上生成的概率分布 $P(z)$ 以及将每个所生成的代码 $z$ 与比特流相关联的信息来提取经解压缩的代码 $z$ 。给定比特流的接收部分和下一个代码 $z$ 的概率预测,算术解码器426可以产生新代码 $z$ ,因为它是由发送设备410处的算术编码器406编码的。使用新代码 $z$ ,算术解码器426可以针对连续代码 $z$ 进行概率预测,读取比特流的附加部分,并对连续代码 $z$ 进行解码直到整个接收到的比特流被解码。可以将经解压缩的代码 $z$ 提供给自动编码器421中的解码器423。解码器423对代码 $z$ 进

行解压缩并输出图像内容 $x$ 的近似 $\hat{x}$  (其可以被称为经重构或解码的图像)。在一些情况下,可以存储内容 $x$ 的近似 $\hat{x}$ 以供以后取回。在一些情况下,内容 $x$ 的近似 $\hat{x}$ 可以由接收设备420恢复并显示在与接收设备420通信地耦合或集成的屏幕上。

[0116] 如上所述,发送设备410的自动编码器401和代码模型404在图4中被示为先前已经被训练的机器学习系统。在一些方面,可以使用图像数据一起训练自动编码器401和代码模型404。例如,自动编码器401的编码器402可以接收第一训练图像 $n$ 作为输入,并且可以将第一训练图像 $n$ 映射到潜在码空间中的代码 $z$ 。代码模型404可以使用概率自回归模型(类似于上述技术)来学习代码 $z$ 的概率分布 $P(z)$ 。算术编码器406可以使用由代码模型404生成的概率分布 $P(z)$ 来生成图像比特流。使用来自代码模型404的比特流和概率分布 $P(z)$ ,算术编码器406可以生成代码 $z$ ,并且可以将代码 $z$ 输出到自动编码器401的解码器403。解码器403然后可以对代码 $z$ 进行解压缩以获得对第一训练图像 $n$ 的重构 $\hat{n}$  (其中重构 $\hat{n}$ 是未压缩的第一训练图像 $n$ 的近似)。

[0117] 在一些情况下,在发送设备410的训练期间使用的后向传播引擎可以执行后向传播过程以基于一个或多个损失函数来调谐自动编码器401和代码模型404的神经网络的参数(例如,权重、偏差等)。在一些情况下,后向传播过程可以基于随机梯度下降技术。后向传播可以包括前向传递、一个或多个损失函数、后向传递,以及权重(和/或其他参数)更新。针对一次训练迭代可以执行前向传递、损失函数、后向传递和参数更新。该过程可以针对每一组训练数据重复一定次数的迭代,直到神经网络的权重和/或其他参数被准确调谐。

[0118] 例如,自动编码器401可以比较 $n$ 和 $\hat{n}$ 以确定第一训练图像 $n$ 和重构的第一训练图像 $\hat{n}$ 之间的损失(例如,由距离向量或其他差值表示)。损失函数可用于分析输出中的误差。在一些示例中,损失可以基于最大似然。在使用未压缩的图像 $n$ 作为输入和重构图像 $\hat{n}$ 作为输出的一个说明性示例中,损失函数 $Loss = D + \beta * R$ 可用于训练自动编码器401和代码模型404的神经网络系统,其中, $R$ 是速率, $D$ 是失真, $*$ 表示乘法函数,而 $\beta$ 是折衷参数,其被设置为定义比特率的值。在另一示例中,损失函数 $Loss_0 = \sum_t distortion(n, \hat{n})$ 可用于训练自动编码器401和代码模型404的神经网络系统。在一些情况下,例如在使用其他训练数据时,可以使用其他损失函数。另一损失函数的一个示例包括均方误差(MSE),定义为 $E_{total} = \sum \frac{1}{2} (target - output)^2$ 。MSE计算二分之一乘以实际答案减去预测(输出)答案平方的总和。

[0119] 基于确定的损失(例如,距离向量或其他差值)并使用后向传播过程,自动编码器401和代码模型404的神经网络系统的参数(例如,权重、偏差等)可以被调整(有效地调整接收到的图像内容和潜在代码空间之间的映射)以减少输入的未压缩图像和作为输出由自动编码器401生成的经压缩的图像内容之间的损失。

[0120] 对第一训练图像,损失(或误差)可能很高,因为实际输出值(重构图像)可能与输入图像有很大不同。训练的目标是使预测的输出的损失量最小化。神经网络可以通过确定神经网络的哪些节点(具有相对应的权重)对神经网络的损失贡献最大来执行后向传递,并且可以调整权重(和/或其他参数)以使损失减少并最终达到最小化。可以计算损失相对于权重的导数(表示为 $dL/dw$ ,其中 $w$ 是特定层处的权重),以确定对神经网络损失贡献最大的权重。例如,可以更新权重,使它们在梯度的相反方向上变化。权重更新可以表示为 $w = w_i - \eta \frac{dL}{dw}$ ,其中 $w$ 表示权重, $w_i$ 表示初始权重,而 $\eta$ 表示学习率。学习率可以设置为任何合适的值,其中,高学习率包括较大的权重更新,而较低的值指示较小的权重更新。

[0121] 可以继续以这种方式训练自动编码器401和代码模型404的神经网络系统,直到获得期望的输出。例如,自动编码器401和代码模型404可以重复后向传播过程以最小化或以其他方式减少输入图像 $n$ 和由生成的代码 $z$ 的解压缩导致的重构图像 $\hat{n}$ 之间的差。

[0122] 可以使用与上述用于训练发送设备410的自动编码器401和代码模型404的技术类似的技术来训练自动编码器421和代码模型424。在一些情况下,可以使用用于训练发送设备410的自动编码器401和代码模型404的相同或不同的训练数据集来训练自动编码器421和代码模型424。

[0123] 在图4所示的示例中,速率失真自动编码器(发送设备410和接收设备420)被训练并根据比特率在推测中运行。在一些实施方式中,可以以多个比特率来训练速率失真自动编码器,以便当在潜在代码 $z$ 中提供不同量的信息时,允许生成和输出高质量的重构图像或视频帧(例如,没有或具有有限的由于相对于输入图像的失真而造成的伪影)。

[0124] 在一些实施方式中,潜在代码 $z$ 可以被划分为至少两个块 $z_1$ 和 $z_2$ 。当以高速率设置使用RD-AE模型时,两个块都被发送到设备以进行解码。当在低速率设置中使用速率失真自动编码器模型时,仅发送块 $z_1$ ,并且在解码器侧从 $z_1$ 推测块 $z_2$ 。可以使用各种技术来执行从 $z_1$ 推测 $z_2$ ,如下文更详细描述。

[0125] 在一些实施方式中,可以使用一组连续的潜在(例如,其可以传达大量信息)和相对应的经量化的离散潜在(例如,其包含较少的信息)。在训练完RD-AE模型之后,可以训练辅助解量化模型。在一些情况下,当使用RD-AE时,仅传输离散潜在,并且在解码器端使用辅助解量化模型从离散潜在中推测出连续潜在。

[0126] 尽管示出系统400包括某些组件,但是本领域普通技术人员将理解,系统400可以包括比图4所示的组件更多或更少的组件。例如,系统400的发送设备410和/或接收设备420在一些情况下还可以包括一个或多个存储器设备(例如RAM、ROM、高速缓存器等)、一个或多个网络接口(例如,有线和/或无线通信接口等)、一个或多个显示设备和/或图4中未示出的其他硬件或处理设备。图4所示的组件和/或系统400的其他组件可以使用一个或多个计算或处理组件来实现。一个或多个计算组件可以包括中央处理单元(CPU)、图形处理单元(GPU)、数字信号处理器(DSP)和/或图像信号处理器(ISP)。下面针对图14描述可以用系统1400实现的计算设备和硬件组件的说明性示例。

[0127] 系统400可以是单个计算设备或多个计算设备的一部分或由其实实现。在一些示例中,发送设备410可以是第一设备的一部分并且接收设备420可以是第二计算设备的一部分。在一些示例中,发送设备410和/或接收设备420可以作为诸如下列各项的电子设备(或多个设备)的一部分被包括在内:电话系统(例如智能手机、蜂窝电话、会议系统等)、桌面式计算机、膝上型计算机或笔记本电脑、平板电脑、机顶盒、智能电视、显示设备、游戏控制台、视频流式传输设备、SOC、IoT(物联网)设备、智能可穿戴设备(例如,头戴式显示器(HMD)、智能眼镜等)、相机系统(例如,数码相机、IP相机、摄像机、安全相机等)或任何其他合适的电子设备。在一些情况下,系统400可以由图1所示的图像处理系统100来实现。在其他情况下,系统400可以由一个或多个其他系统或设备来实现。

[0128] 图5A是示出示例神经网络压缩系统500的图。在一些示例中,神经网络压缩系统500可以包括RD-AE系统。在图5A中,神经网络压缩系统500包括编码器502、算术编码器508、算术解码器512和解码器514。在一些情况下,编码器502和/或解码器514可以分别与编码器

402和/或解码器403相同。在其他情况下,编码器502和/或解码器514可以分别与编码器402和/或解码器403不同。

[0129] 编码器502可以接收图像501(图像 $x_i$ )作为输入并且可以将图像501(图像 $x_i$ )映射和/或转换为潜在代码空间中的潜在代码504(潜在 $z_i$ )。图像501可以表示与帧序列(例如,视频)相关联的静止图像和/或视频帧。在一些情况下,编码器502可以执行前向传递以生成潜在代码504。在一些示例中,编码器502可以实现可学习函数。在一些情况下,编码器502可以实现由 $\varphi$ 参数化的可学习函数。例如,编码器502可以实现函数 $q_\varphi(z|x)$ 。在一些示例中,可学习函数不需要与解码器514共享或由解码器514知晓。

[0130] 算术编码器508可以基于潜在代码504(潜在 $z_i$ )和潜在先验506来生成比特流510。在一些示例中,潜在先验506可以实现可学习函数。在一些情况下,潜在先验506可以实现由 $\psi$ 参数化的可学习函数。例如,潜在先验506可以实现函数 $p_\psi(z)$ 。潜在先验506可用于使用无损压缩将潜在代码504(潜在 $z_i$ )转换为比特流510。潜在先验506可以在发送方(例如,编码器502和/或算术编码器508)和接收方(例如,算术解码器512和/或解码器514)两者处被共享和/或可用。

[0131] 解码器514可以从算术编码器508接收经编码的比特流510,并且使用潜在先验506来对经编码的比特流510中的潜在代码504(潜在 $z_i$ )进行解码。解码器514可以将潜在代码504(潜在 $z_i$ )解码为近似重构图像516(重构 $\hat{x}_i$ )。在一些情况下,解码器514可以实现由 $\theta$ 参数化的可学习函数。例如,解码器514可以实现函数 $p_\theta(x|z)$ 。由解码器514实现的可学习函数可以在发送方(例如,编码器502和/或算术编码器508)和接收方(例如,算术解码器512和/或解码器)两者处被共享和/或可用。

[0132] 可以训练神经网络压缩系统500以使速率失真最小化。在一些示例中,速率反映比特流510(比特流 $b$ )的长度,而失真反映图像501(图像 $x_i$ )和重构图像516(重构 $\hat{x}_i$ )之间的失真。参数 $\beta$ 可用于针对特定的速率失真比来训练模型。在一些示例中,参数 $\beta$ 可以用于定义和/或实现速率和失真之间的某种折衷。

[0133] 在一些示例中,损失可以表示如下 $L_{RD}(x; \varphi, \psi, \theta) = E_{q_\varphi(z|x)}[-\log p_\theta(x|z) - \beta \log p_\psi(z)]$ ,其中函数 $E$ 是期望值。可以基于例如均方误差(MSE)的损失函数来确定失真 $(x|z; \theta)$ 。在一些示例中,术语 $-\log p_\theta(x|z)$ 可以指示和/或表示失真 $D(x|z; \theta)$ 。

[0134] 用于发送潜在值的速率可以表示为 $R_z(z; \psi)$ 。在一些示例中,术语 $\log p_\psi(z)$ 可以指示和/或表示速率 $R_z(z; \psi)$ 。在一些情况下,可以在整个数据集 $D$ 上使损失最小化,如下所示:  
 $\varphi_D^*, \psi_D^*, \theta_D^* = \operatorname{argmin} E_{x \sim D}[L_{RD}(x; \varphi, \psi, \theta)]$ 。

[0135] 图5B是示出由神经网络压缩系统500执行的推测过程530的图。如图所示,编码器502可以将图像501转换为潜在代码504。在一些示例中,图像501可以表示与帧序列(例如,视频)相关联的静止图像和/或视频帧。

[0136] 在一些示例中,编码器502可以使用单个前向传递 $z = q_{\varphi_D^*}(z|x)$ 对图像501进行编码。算术编码器508然后可以在潜在先验506下执行对潜在代码504(潜在 $z_i$ )的算术编码以生成比特流520( $b_2^i$ )。在一些示例中,算术编码器508可以如下生成比特流520: $b_2^i = AE(z|p_{\psi_D^*}(z))$ 。

[0137] 算术解码器512可以从算术编码器508接收比特流520并且在潜在先验506下执行潜在代码504(潜在 $z_i$ )的算术解码。在一些示例中,算术解码器512可以对来自比特流520的

潜在代码504进行解码如下： $z = \text{AD}(b_z^i | p_{\psi_D}(z))$ 。解码器514可以对潜在代码504(潜在 $z_i$ )进行解码并生成重构图像516(重构 $\hat{x}_i$ )。在一些示例中，解码器514可以使用单个前向传递来对潜在代码504(潜在 $z_i$ )解码如下： $\hat{x} = p_{\theta_D}(x|z)$ 。

[0138] 在一些示例中，RD-AE系统可以使用一组训练数据被训练，并针对将被发送到接收机(例如，解码器)并由其解码的数据点(例如，图像数据、视频数据、音频数据)进行进一步微调。例如，在推测时，RD-AE系统可以在被发送到接收机的图像数据上进行微调。由于压缩模型通常很大，因此将与模型相关联的参数发送到接收机在诸如网络(例如带宽等)、存储和计算资源等资源方面可能非常昂贵。在一些情况下，RD-AE系统可以在被压缩的单个数据点上进行调整并被发送到接收机进行解压缩。这可以限制发送到接收机的信息量(和相关成本)，同时保持和/或提高压缩/解压缩效率、性能和/或质量。

[0139] 图6是示出用于实例自适应数据压缩的示例神经网络压缩系统600的图。神经网络压缩系统600可以被训练和针对被压缩数据进一步的微调以提供适应于/微调到被压缩数据的压缩(例如，实例适配)。在该示例中，神经网络压缩系统600被示为使用可变自动编码器(VAE)框架实现平均尺度超先验模型架构。在一些情况下，共享的超解码器可用于预测平均尺度超先验模型的均值和尺度参数。

[0140] 如图6所示，可以使用训练数据集602对神经网络压缩系统600进行训练。训练数据集602可由编解码器604的编码器606处理，以生成训练数据集602的潜在空间表示608( $z_2$ )。编码器606可以向编解码器604的解码器610和超编解码器612的超编码器614提供潜在空间表示608( $z_2$ )。

[0141] 超编码器614可以使用潜在空间表示608( $z_2$ )和超先验620的潜在先验来生成训练数据集602的超潜在空间表示616( $z_1$ )。在一些示例中，超潜在空间表示616( $z_1$ )和超潜在空间表示616( $z_1$ )可以为潜在空间 $z = \{z_1, z_2\}$ 提供分层潜在变量模型。

[0142] 超编解码器612的超解码器618可以使用超潜在空间表示616( $z_1$ )来生成超先验模型624。超解码器618可以预测超先验模型624的均值和标度参数。在一些示例中，超先验模型624可以包括潜在空间表示608( $z_2$ )和超潜在空间表示616( $z_1$ )的参数上的概率分布。在一些示例中，超先验模型624可以包括潜在空间表示608( $z_2$ )、超潜在空间表示616( $z_1$ )和超解码器618的参数上的概率分布。

[0143] 解码器610可以使用超先验模型624和潜在空间表示608( $z_2$ )来生成用于训练数据集602的重构的数据626(重构 $\hat{x}$ )。

[0144] 在一些示例中，在训练期间，神经网络压缩系统600可以实现混合的量化策略，其中经量化的潜在空间表示608( $z_2$ )用于计算失真损失，并且在计算速率损失时使用用于潜在空间表示608( $z_2$ )和超潜在空间表示616( $z_1$ )的噪声样本。

[0145] 图7是示出使用模型先验微调(例如，实例适配)的神经网络压缩系统700的示例架构的图。在一些示例中，神经网络压缩系统700可以包括使用RDM-AE模型先验微调的RD-AE系统。神经网络压缩系统700可以包括编码器702、算术编码器706、算术解码器716和解码器718。在一些情况下，编码器702可以与编码器402、编码器502或编码器606相同或不同，而且解码器718可以与解码器403、解码器514或解码器610相同或不同。算术编码器706可以与算术编解码器406或算术编码器508相同或不同，而算术解码器716可以与算术解码器426或508相同或不同。

[0146] 编码器702可以接收图像701(图像 $x_i$ )作为输入并且可以将图像701(图像 $x_i$ )映射和/或转换为潜在代码空间中的潜在代码704(潜在 $z_i$ )。在一些示例中,图像701可以表示与帧序列(例如,视频)相关联的静止图像和/或视频帧。在一些示例中,在处理图像701之前,可以在训练数据集(例如,图6的训练数据集602)上训练编码器702。此外,可以针对图像701对编码器702进行进一步训练或微调(例如,实例适配)。例如,图像701可用于针对图像701对编码器702进行微调。针对图像701对编码器702进行微调可以造成图像701的高压缩性能。例如,编码器702的微调可以允许神经网络压缩系统700改善经压缩的图像701的速率失真。

[0147] 在一些情况下,编码器702可以使用单个前向传递来生成潜在代码704。在一些示例中,编码器702可以实现可学习函数。在一些情况下,可学习函数可由 $\varphi$ 参数化。例如,编码器702可以实现函数 $q_\varphi(z|x)$ 。在一些示例中,可学习函数不需要与算术解码器716和/或解码器718共享或被其所知。

[0148] 算术编码器706可以使用潜在先验708对潜在代码704进行熵编码,并生成将被发送到算术解码器716的比特流710(例如,比特流 $b_z^i$ )。比特流710可以包括表示潜在代码704的经压缩的数据。潜在先验708可用于使用无损压缩将潜在代码704(潜在 $z_i$ )转换为比特流710。潜在先验708可以在发送方(例如,编码器702和/或算术编码器706)和接收方(例如,算术解码器716和/或解码器718)两者处被共享和/或可用。在一些示例中,潜在先验708可以实现可学习函数。在一些情况下,可学习函数可由 $\psi$ 参数化。例如,潜在先验708可以实现函数 $p_\psi(z)$ 。

[0149] 神经网络压缩系统700还可以包括模型先验714。模型先验714可以包括潜在先验708和解码器718的参数上的概率分布。在一些示例中,模型先验714可以实现可学习函数。在一些情况下,可学习函数可由 $\omega$ 参数化。例如,模型先验714可以实现函数 $p_\omega(\psi|\theta)$ 。神经网络压缩系统700可以使用模型先验714以将编码器702和潜在先验708的经微调参数转换为将要发送到算术解码器716的比特流712(例如,比特流 $b_{\psi,\theta}^i$ )。

[0150] 模型先验714可以在发送方(例如,编码器702和/或算术编码器706)和接收方(例如,算术解码器716和/或解码器718)两者处被共享和/或可用。例如,算术编码器706可以将比特流712发送到算术解码器716,以用于对从潜在代码704生成的比特流710进行解码。比特流712可以包括表示编码器702和潜在先验708的经微调参数的经压缩的数据,算术解码器716可以使用这些经压缩的数据来获得编码器702的经微调参数和潜在先验708。算术解码器716和解码器718可以基于从比特流710获得的潜在代码704以及从比特流712获得的编码器702的经微调参数和潜在先验708来重构图像701。

[0151] 算术解码器716可以使用潜在先验708和模型先验714将比特流710转换为潜在码704(潜在 $z_i$ )。例如,算术解码器716可以使用潜在先验708和模型先验714以对比特流710进行解码。解码器718可以使用由算术解码器716解码的潜在码704(潜在 $z_i$ )来生成经重构的图像720(重构 $\hat{x}_i$ )。例如,解码器718可以将潜在码704(潜在 $z_i$ )解码为近似经重构的图像720(重构 $\hat{x}_i$ )。在一些示例中,解码器718可以使用潜在码704、潜在先验708和/或模型先验714来生成经重构的图像720。

[0152] 在一些情况下,解码器718可以实现由 $\theta$ 参数化的可学习函数。例如,解码器718可以实现可学习函数 $p_\theta(x|z)$ 。由解码器718实现的可学习函数可以在发送方(例如,编码器

702和/或算术编码器706)和接收方(例如,算术解码器716和/或解码器718)两者处被共享和/或可用。

[0153] 在一些示例中,在编码器侧,可以使用速率失真模型(RDM)损失在图像701(图像 $x_i$ )上对模型进行微调如下: $\varphi_D^*, \psi_D^*, \theta_D^* = \operatorname{argmin} L_{RD}(x; \varphi, \psi, \theta, \omega)$ 。在一些示例中,可以使用经微调编码器702的单个前向传递来对图像701(图像 $x_i$ )编码如下 $z = q_{\varphi_x^*}(z|x)$ 。在一些情况下,经微调潜在先验708可以被熵编解码如下 $b_{\psi}^i = \operatorname{AE}(\psi_x^*|p_{\omega}(\psi))$ ,并且经微调解码器718和/或算术编码器706可以被熵编解码如下 $b_{\theta}^i = \operatorname{AE}(\theta_x^*|p_{\omega}(\theta))$ 。在一些情况下,潜在代码704(潜在 $z_i$ )可以被熵编解码如下 $b_z^i = \operatorname{AE}(z|p_{\psi_x^*}(z))$ 。

[0154] 在解码器侧,在一些示例中,经微调潜在先验708可以被熵编解码如下 $\psi_x^* = \operatorname{AD}(b_{\psi}^i|p_{\omega}(\psi))$ ,并且经微调解码器718和/或算术解码器716可以被熵编解码如下 $\theta_x^* = \operatorname{AD}(b_{\theta}^i|p_{\omega}(\theta))$ 。在一些情况下,潜在代码704(潜在 $z_i$ )也可以被熵编解码如下 $z = \operatorname{AD}(b_z^i|p_{\psi_x^*}(z))$ 。

[0155] 在一些示例中,解码器718可以使用经微调解码器(例如,解码器718)的单个前向传递来将潜在代码704(潜在 $z_i$ )解码为近似重构图像720(重构 $\hat{x}_i$ )如下 $\hat{x} = p_{\theta_x^*}(x|z)$ 。

[0156] 图8是示出由使用模型先验微调的示例神经网络压缩系统800实现的示例推测过程的图。在一些示例中,神经网络压缩系统800可以包括使用RDM-AE模型先验微调的RD-AE系统。在一些情况下,神经网络压缩系统800可以包括使用模型先验微调的AE模型。

[0157] 在该说明性示例中,神经网络压缩系统800包括编码器802、算术编码器808、算术解码器812、解码器814、模型先验816和潜在先验806。在一些情况下,编码器802可以与编码器402、编码器502、编码器606或编码器702相同或不同,而且解码器814可以与解码器403、解码器514、解码器610或解码器718相同或不同。算术编码器808可以与算术编解码器406、算术编码器508或算术编码器706相同或不同,而算术解码器812可以与算术解码器426、算术解码器508或算术解码器716相同或不同。

[0158] 神经网络压缩系统800可以为图像801生成潜在代码804(潜在 $z_i$ )。神经网络压缩系统800可以使用潜在代码804和潜在先验806来对图像801(图像 $x_i$ )进行编码,并生成可以被接收机用来生成重构图像820(重构 $\hat{x}_i$ )的比特流810。在一些示例中,图像801可以表示与帧序列(例如,视频)相关联的静止图像和/或视频帧。

[0159] 在一些示例中,可以使用RDM-AE损失对神经网络压缩系统800进行微调。可以通过使速率-失真-模型速率(RDM)损失最小化来训练神经网络压缩系统800。在一些示例中,在编码器侧,可以使用RDM损失在图像801(图像 $x_i$ )上对AE模型进行微调如下:

$$\varphi_x^*, \psi_x^*, \theta_x^* = \operatorname{argmin} L_{RD}(x; \varphi, \psi, \theta, \omega)。$$

[0160] 经微调的编码器802可以对图像801(图像 $x_i$ )进行编码以生成潜在代码804。在一些情况下,经微调的编码器802可以使用如下的单个前向传递来对图像801(图像 $x_i$ )进行编码 $z = q_{\psi_x^*}(z|x)$ 。算术编码器808可以使用潜在先验806将潜在代码804转换为用于算术解码器812的比特流810。算术编码器808可以在模型先验816下对经微调的解码器814和经微调的潜在先验806的参数进行熵编码,并且生成包括经微调的解码器814和经微调的潜在先验806的压缩参数的比特流811。在一些示例中,比特流811可以包括经微调的解码器814和经微调的潜在先验806的更新的参数。更新的参数可以包括例如相对于基线解码器和潜在先验(例如微调之前的解码器814和潜在先验806)的参数更新。

[0161] 在一些情况下,经微调的潜在先验806可以在模型先验816下进行熵编码如下 $b_{\psi}^i = AE(\psi_x^* | p_{\omega}(\psi))$ ,经微调的解码器814可以在模型先验816下进行熵编码如下 $b_{\theta}^i = AE(\theta_x^* | p_{\omega}(\theta))$ ,并且潜在代码804(潜在 $z_i$ )可以在经微调的潜在先验806下进行熵编码如下 $b_z^i = AE(z | p_{\psi_x}(z))$ 。在一些情况下,在解码器侧,经微调的潜在先验806可以在模型先验816下进行熵编码如下 $\psi_x^* = AD(b_{\psi}^i | p_{\omega}(\psi))$ ,经微调的解码器814可以在模型先验816下进行熵编码如下 $\theta_x^* = AD(b_{\theta}^i | p_{\omega}(\theta))$ ,并且潜在代码804(潜在 $z_i$ )可以在经微调的潜在先验806下进行熵编码如下 $z = AD(b_z^i | p_{\psi_x}(z))$ 。

[0162] 解码器814可以将潜在代码804(潜在 $z_i$ )解码为近似重构图像820(重构 $\hat{x}_i$ )。在一些示例中,解码器814可以使用经微调的解码器的单个前向传递来将潜在代码804解码如下 $\hat{x} = p_{\theta_x}(x | z)$ 。

[0163] 如前所述,可以通过使RDM损失最小化来训练神经网络压缩系统800。在一些情况下,速率可以反映比特流 $b$ (例如,比特流810和/或811)的长度,失真可以反映输入图像801(图像 $x_i$ )和重构图像820(重构 $\hat{x}_i$ )之间的失真,并且模型-速率可以反映用于将模型更新(例如,经更新的参数)发送到接收机(例如,到解码器814)和/或该操作所需的比特流的长度。参数 $\beta$ 可用于针对特定的速率失真比来训练模型。

[0164] 在一些示例中,可以在推测时使数据点 $x$ 的损失最小化,如下所示:

$\varphi_x^*, \psi_x^*, \theta_x^* = \operatorname{argmin} \mathcal{L}_{\text{RDM}}(x; \varphi, \psi, \theta, \omega)$ 。在一些示例中,RDM损失可以表示如下:

$L_{\text{RDM}}(x; \varphi, \psi, \theta, \omega) = E_{q_{\varphi}} q_{\psi}(z | x) [-\log p_{\theta}(x | z) - \beta \log p_{\psi}(z) - \beta \log p_{\omega}(\psi, \theta)]$ 。在一些情况下,可以基于例如均方误差(MSE)的损失函数来确定失真 $D(x | z; \theta)$ 。

[0165] 术语 $-\log p_{\theta}(x | z)$ 可以指示和/或表示失真 $D(x | z; \theta)$ 。术语 $\beta \log p_{\psi}(z)$ 可以指示和/或表示用于发送潜在 $R_z(z; \psi)$ 的速率,而术语 $\beta \log p_{\omega}(\psi, \theta)$ 可以指示和/或表示用于发送经微调的模型更新 $R_{\psi, \theta}(\psi, \theta; \omega)$ 的速率。

[0166] 在一些情况下,模型先验816可以反映用于发送模型更新的比特率开销的长度。在一些示例中,用于发送模型更新的比特率可以描述如下: $|b_{\psi, \theta}^i| = R_{\psi, \theta}(\psi, \theta; \omega) = -\log p_{\omega}(\psi, \theta)$ 。在一些情况下,可以选择模型先验,以使发送没有更新的模型很便宜,即比特长度(模型-速率-损失)很小: $R_{\psi, \theta}(\psi_D^*, \theta_D^*; \omega)$ 。

[0167] 在一些情况下,使用RDM损失函数,神经网络压缩系统800可以仅在潜在速率或失真随着至少同样多的比特降低时将比特添加到用于模型更新 $b_{\psi, \theta}^i$ 的比特流中。这可以提高速率-失真(R/D)性能。例如,如果速率或失真还可以随着至少相同数量的比特而下降,神经网络压缩系统800可以增加比特流811中用于发送模型更新的比特的数量。在其他情况下,神经网络压缩系统800可以将比特添加到用于模型更新 $b_{\psi, \theta}^i$ 的比特流中,即使潜在速率或失真没有随着至少同样多的比特而下降。

[0168] 可以端到端地训练神经网络压缩系统800。在一些情况下,可以在端到端推测时使RDM损失最小化。在一些示例中,可以一次性花费一定量的计算(例如,对模型进行微调)并且随后可以获得高压缩比,而无需接收机侧的额外成本。例如,内容提供者可能花费大量计算来针对将提供给大量接收机的视频更广泛地训练和微调神经网络压缩系统800。经高度训练和微调的神经网络压缩系统800可以为该视频提供高压缩性能。在花费大量计算之后,视频提供者可以存储模型先验的经更新的参数,并高效地提供给经压缩的视频的每个接收

机以对视频进行解压缩。视频提供者可以在每次传输视频时在压缩(以及减少网络和计算资源)方面获得巨大收益,这大大超过了对模型进行训练和微调的初始计算成本。

[0169] 由于视频和高分辨率图像中的大量像素,本文中的学习和微调方法对于视频压缩和/或高分辨率图像可以非常有利。在一些情况下,复杂度和/或解码器计算可以用作整体系统设计和/或实现的附加考虑因素。例如,可以对可以快速进行推测的非常小的网络进行微调。作为另一示例,可以针对接收机复杂度添加成本项,这可以强制和/或导致模型移除一个或多个层。在某些示例中,可以使用机器学习来学习更复杂的模型先验,以获得更大的增益。

[0170] 模型先验设计可以包括各种属性。在一些示例中,实现的模型先验可以包括这样的模型先验:它分配高概率 $p_{\omega}(\psi_D^*, \theta_D^*)$ 用于发送没有任何更新的模型,并且因此比特率低: $R_{\psi, \theta}(\psi_D^*, \theta_D^*; \omega)$ 。在一些情况下,模型先验可以包括这样的模型先验:它为 $\psi_D^*, \theta_D^*$ 周围的值分配非零概率,因此在实践中可以对经微调的模型的不同实例进行编码。在一些情况下,模型先验可以包括可以在推测时被量化并用于进行熵编解码的模型先验。

[0171] 图9是示出由使用模型先验微调的示例神经网络压缩系统900执行的编码和解码任务的图。在该示例中,编码器904接收图像902并将图像902压缩到潜在空间906(z)。算术编码器908可以使用在训练集D上预训练的潜在先验910、解码器922的参数和全局模型参数914( $\theta_D$ ),来计算经量化的模型参数( $\bar{\theta}$ )。

[0172] 在一些示例中,经量化的模型参数( $\bar{\theta}$ )可以表示在训练集D上预训练的全局模型参数914( $\theta_D$ )和经量化的模型参数更新( $\delta$ )的总和。在一些示例中,经量化的模型参数更新( $\delta$ )可由量化器( $Q_t$ )基于在训练集D上预训练的全局模型参数914( $\theta_D$ )和模型参数更新( $\theta$ )之间的差来生成。

[0173] 算术编码器908可以对经量化的模型参数更新( $\delta$ )进行熵编码以生成比特流918,用于用信号向接收机(例如,算术解码器920)传送模型参数更新。算术编码器908可以使用经量化的模型先验912( $p[\bar{\delta}]$ )对经量化的模型参数更新( $\delta$ )进行熵编码。在一些示例中,算术编码器908可以实现连续模型先验( $p[\delta]$ )来规范模型参数更新( $\delta$ ),并使用经量化的模型先验912( $p[\bar{\delta}]$ )以对经量化的模型参数更新( $\delta$ )进行熵编码。

[0174] 算术编码器908还可以对潜在空间906(z)进行熵编码以生成比特流916,用于用信号向接收机(例如,算术解码器920)传送潜在空间906(z)。在一些示例中,算术编码器908可以对比特流916和比特流918进行级联,并将级联的比特流发送到接收机(例如,算术解码器920)。

[0175] 算术解码器920可以使用经量化的模型先验912( $p[\bar{\delta}]$ )对比特流918进行熵解码并获得经量化的模型参数更新( $\delta$ )。算术解码器920可以使用经微调的潜在先验910对来自比特流916的潜在空间906(z)进行熵解码。解码器922可以使用潜在空间906和模型参数更新来生成重构图像924。

[0176] 在一些示例中,神经网络压缩系统900可以在编码过程期间实现下面的算法1,并在解码过程期间实现下面的算法2。

**算法 1 x 的编码**

**输入:** 在训练集合  $D$  上被预训练的全局模型参数  $\{\phi_D, \theta_D\}$ 。模型参数量化器  $Q_t$ ，模型先验  $p[\bar{\delta}]$ ，以及待压缩数据点  $x$ 。

**输出:** 经压缩的比特流  $b$ 。

- 1: 对模型参数进行初始化:  $\phi = \phi_D$  并且  $\theta = \theta_D$
- 2: **for** step in MAX STEPS **do**
- 3: 对来自  $x_f \sim x$  的一个或多个帧的选择进行采样
- 4: 对可发送参数进行量化:  $\bar{\theta} \leftarrow Q_t(\delta) + \theta_D$ , 其中  $\delta = \theta - \theta_D$
- 5: 前向传送:  $z \sim q_\phi(z | x_f)$  并且评估  $p_\phi(x_f | z)$  和  $p_\phi(z)$
- 6: 根据下文的公式 (2) 计算  $x_f$  上的损失  $L_{RDM}(\phi, \delta)$ 。
- 7: 使用针对  $Q_t$  的 STE 后向传播, 然后使用梯度  $\frac{\partial L_{RDM}}{\partial \phi}$  和  $\frac{\partial L_{RDM}}{\partial \theta}$  来更新  $\phi, \theta$
- 8: **end for**
- 9: 将  $x$  压缩为  $z \sim q_\phi(x)$
- 10: 计算经量化的模型参数:  $\bar{\theta} = \theta_D + \bar{\delta}$ , 其中  $\bar{\delta} = Q_t(\bar{\theta} - \theta_D)$
- 11: 熵编码:  $b_{\bar{\delta}} = \text{enc}(\bar{\delta}; p[\bar{\delta}])$  并且  $b_z = \text{enc}(z; p_{\bar{\theta}}(z))$
- 12: 对比特流进行级联  $b = (b_{\bar{\delta}}, b_z)$

[0177]

**算法 2 x 的解码**

**输入:** 模型先验  $p[\bar{\delta}]$ , 比特流  $b = (b_{\bar{\delta}}, b_z)$ , 全局模型参数  $\theta_D$

**输出:** 经解码的数据点  $\hat{x}$

- 1: 熵解码:  $\bar{\delta} = \text{dec}(b_{\bar{\delta}}; p[\bar{\delta}])$
- 2: 计算经更新的模型参数:  $\bar{\theta} = \theta_D + \bar{\delta}$
- 3: 在经微调的先验  $z = \text{dec}(b_z; p_{\bar{\theta}}(z))$  下对潜在进行熵解码
- 4: 将实例解码为经微调的解码器的均值:  $p_{\bar{\theta}}(x | z)$

[0178]

[0179] 在一些示例中, 上文引用的公式 (2) 可用于计算 RDM 损失如下:

$$[0180] \quad L_{RDM}(\phi, \delta) = L_{RD}(\phi, \theta_D + \bar{\delta}) - \beta \log p(\delta) \quad \text{公式 2}$$

[0181] 其中,  $\log p(\delta)$  项表示模型更新  $M$ , 并且  $\beta$  是折衷参数。

[0182] 在一些示例中, 可以将模型先验 912 设计为概率密度函数 (PDF)。在一些情况下, 参数更新  $\delta = \theta - \theta_D$  可以使用以零更新为中心的高斯分布进行建模。在一些情况下, 模型先验可以在更新上被定义为协方差为零的多元零中心高斯, 并且单个共享 (超参数)  $\sigma$  表示标准差  $p(\delta) = \mathcal{N}(\delta | \mathbf{0}, \sigma \mathbf{I})$ , 其可以等价于通过  $p(\theta) = \mathcal{N}(\theta | \theta_D, \sigma \mathbf{I})$  对  $\theta$  取模。

[0183] 在一些情况下, 当在  $p[\bar{\delta}]$  下对经量化更新  $\bar{\delta}$  进行熵编解码时, 即使零更新也可能不

是免费的。在一些情况下,这些初始静态成本可以被定义为 $\overline{M}_0 = -\log p[\bar{\delta} = \mathbf{0}]$ 。由于定义的模型先验的模式为零,因此这些初始成本 $\overline{M}_0$ 可以等于最小成本。上述公式(2)的最小化可以确保在克服这些静态成本后,在模型更新上花费的任何额外比特都可以伴随着RD性能的相应改进。

[0184] 在一些情况下,模型先验设计可以基于独立高斯网络先验,其中参数之间的独立性可以假定如下: $p_\omega(\psi | \theta) = \prod_{\theta^{(i)} \in \theta} p_\omega(\theta^{(i)}) \prod_{\psi^{(i)} \in \psi} p_\omega(\psi^{(i)})$ 。在本文的各个示例中,提供了用于单个参数 $p_\omega(\theta^{(i)})$ 的方程和示例。但是,其他示例可以涉及用于多个参数的方程。

[0185] 在一些示例中,先验高斯模型可以包括以全局模型为中心的高斯RDAE模型,其具有可变标准差 $\sigma$ 如下: $p_\omega(\theta^{(i)}) = \mathcal{N}(\theta_D^*, \sigma)$ 。

[0186] 在一些情况下,可以在较高的 $\sigma$ (例如,用于发送未调谐模型的较高成本(初始成本))和较低的 $\sigma$ (例如,用于微调模型的更有限的概率质量)之间进行折衷。

[0187] 在一些示例中,模型先验设计可以基于独立拉普拉斯模型先验。在一些示例中,拉普拉斯模型先验可以包括以全局模型为中心的高斯RDAE模型,其具有可变标准差 $\sigma$ 如下:

$$p_\omega(\theta^{(i)}) = L(\theta_D^*, \sigma)$$

[0188] 在一些情况下,可以在较高的 $\sigma$ (例如,用于发送未调谐模型的较高成本(初始成本))和较低的 $\sigma$ (例如,用于微调模型的更有限的概率质量)之间进行折衷。在一些示例中,模型可以实施参数更新的稀疏性。

[0189] 在一些情况下,模型先验设计可以基于独立的Spike和Slab模型先验。在一些示例中,Spike和Slab模型先验可以包括以全局模型为中心的高斯RDAE模型,其具有针对slab分量的可变标准差 $\sigma_{slab}$ ,针对spike分量的可变标准差 $\sigma_{spike} \ll \sigma_{slab}$ ,以及Spike/Slab比 $\alpha$ ,如下: $p_\omega(\theta^{(i)}) = \frac{1}{c} \left( \mathcal{N}(\theta_D^*, \sigma_{slab}) + \alpha \mathcal{N}(\theta_D^*, \sigma_{spike}) \right)$ 。

[0190] 在一些示例中,当 $\alpha$ 很大,并且对微调模型广泛支持时,用于发送未调模型的成本(初始成本)可能较低。

[0191] 在一些示例中,可以联合训练模型先验和全局AE模型。在一些情况下,可以与实例RDM-AE模型一起端到端地训练模型先验,而不是手动设置参数 $\omega$ 。

[0192] 如上所述,在被发送到接收方并由接收方解码的数据点上对神经压缩系统进行微调可以提供速率和/或失真的优点和好处。图10是示出在要被发送到接收机的数据点(例如,图像或帧、视频、音频数据等)上微调的示例RD-AE模型以及没有在要被发送到接收机的数据点上微调的RD-AE模型的示例速率失真的图1000。

[0193] 在该示例中,图1000示出了未如本文所述微调的RD-AE模型的速率失真1002,以及经微调的RD-AE模型的速率失真1004。如速率失真1002和速率失真1004所示,经微调的RD-AE模型比其他RD-AE模型具有明显更高的压缩(速率失真)性能。

[0194] 可以使用要被压缩并被发送到接收机的数据点对与速率失真相关联的RD-AE模型1004进行微调。在一些示例中,可以在数据点 $x$ 上对压缩AE微调如下:

$\varphi_D^*, \psi_D^*, \theta_D^* = \operatorname{argmin} L_{RD}(x; \varphi, \psi, \theta, \omega)$ 。在一些情况下,这可以允许数据点 $x$ 上的高压压缩(速率-失真或R/D)性能。经微调的先验的参数(例如, $\psi_D^*$ )和经微调的解码器的参数(例如, $\psi_D^*, \theta_D^*$ )

可以与接收机共享,以用于对比特流进行解码。接收机可以使用经微调的先验(例如, $\psi_D^*$ )和经微调的解码器的参数(例如, $\psi_D^*, \theta_D^*$ )来对经微调的RD-AE系统产生的比特流进行解码。

[0195] 下面的表1示出了示例调谐细节:

[0196] 表1

		参数数量	比特数量 (16 比特)
[0197] 模型大小	先验	800 万	16 MB
模型大小	解码器	290 万	5.8 MB
模型大小	传输大小	1090 万	21.8 MB
图像大小		1600x1216 像素	
用于发送经微调的模型参数的速率中的开销		89 bpp (每像素比特数量)	

[0198] 表1中的调谐细节只是说明性示例。本领域的普通技术人员将认识到,其他示例可以包括更多、更少、相同和/或不同的调谐细节。

[0199] 图11是示出用于使用适配于正在被压缩的输入数据的神经网络压缩系统(例如,神经网络压缩系统500、神经网络压缩系统600、神经网络压缩系统700、神经网络压缩系统800、神经网络压缩系统900)的实例自适应压缩的示例过程1100的流程图。

[0200] 在框1102处,过程1100可以包括:通过神经网络压缩系统接收输入数据以便由神经网络压缩系统进行压缩。神经网络压缩系统可以在训练数据集(例如,图6的训练数据集602)上被训练。在一些示例中,神经网络压缩系统可以包括在训练集D(例如,训练数据集)上被预训练的全局模型参数 $\{\phi_D, \theta_D\}$ 。输入数据可以包括图像数据、视频数据、音频数据和/或任何其他数据。

[0201] 在框1104处,过程1100可以包括:确定用于神经网络压缩系统的一组更新。在一些示例中,一组更新可以包括使用输入数据调谐的经更新的模型参数(例如,模型参数更新 $(\theta)$ 和/或经量化的模型参数更新 $(\delta)$ )。

[0202] 在一些示例中,确定用于神经网络压缩系统的一组更新可以包括:处理(例如,编解码)神经网络压缩系统处的输入数据;基于经处理的输入数据确定神经网络压缩系统的一个或多个损失(例如,RDM损失);以及基于一个或多个损失来调谐神经网络压缩系统的模型参数。经调谐的模型参数可以包括用于神经网络压缩系统的一组更新。

[0203] 在一些示例中,一个或多个损失可以包括:与用于基于第一比特流的大小来发送输入数据的经压缩的版本的速率相关联的速率损失,与输入数据和从输入数据的经压缩的版本生成的重构的数据之间的失真相关联的失真损失,以及与用于基于第二比特流的大小来发送经更新的模型参数的经压缩的版本的速率相关联的模型速率损失。在一些情况下,可以使用上述公式2来计算一个或多个损失。

[0204] 在框1106处,过程1100可以包括:通过神经网络压缩系统使用潜在先验(例如,潜在先验506、潜在先验622、潜在先验708、潜在先验806、潜在先验910)生成包括输入数据的经压缩的版本的第二比特流(例如,比特流510、比特流520、比特流710、比特流810、比特流

916)。在一些示例中,可以通过将输入数据编码为潜在空间表示并对该潜在空间表示进行熵编码来生成第一比特流。

[0205] 在框1108处,过程1100可以包括:通过神经网络压缩系统使用潜在先验和模型先验(例如,模型先验714、模型先验816、模型先验912、模型先验 $p[\bar{\delta}]$ )生成包括经更新的模型参数(例如,经量化的模型参数更新 $\bar{\delta}$ )的经压缩的版本的第二比特流(例如,比特流510、比特流520、比特流712、比特流811、比特流918)。

[0206] 在一些示例中,模型先验可以包括独立高斯网络先验、独立拉普拉斯网络先验和/或独立Spike和Slab网络先验。

[0207] 在一些示例中,生成第二比特流可以包括:通过神经网络压缩系统使用模型先验对潜在先验进行熵编码;以及通过神经网络压缩系统使用模型先验对经更新的模型参数进行熵编码。

[0208] 在一些示例中,经更新的模型参数可以包括神经网络参数,例如权重、偏置等。在一些示例中,经更新的模型参数可以包括解码器模型的一个或多个经更新的参数。可以使用输入数据来调谐一个或多个经更新的参数(例如,调谐/调整以减少输入数据的一个或多个损失)。

[0209] 在一些示例中,经更新的模型参数可以包括编码器模型的一个或多个经更新的参数。一个或多个经更新的参数可以是使用输入数据被调谐的。在一些情况下,第一比特流可由神经网络压缩系统使用一个或多个经更新的参数生成。

[0210] 在一些示例中,生成第二比特流可以包括:通过神经网络压缩系统使用一个或多个经更新的参数将输入数据编码为输入数据的潜在空间表示;以及通过神经网络压缩系统使用潜在先验将潜在空间表示熵编码为第一比特流。

[0211] 在框1110处,过程1100可以包括:输出第一比特流和第二比特流以传输到接收机。在一些示例中,接收机可以包括解码器(例如,解码器514、解码器610、解码器718、解码器814、解码器922)。在一些示例中,第二比特流还可以包括潜在先验的经压缩的版本和模型先验的经压缩的版本。

[0212] 在一些情况下,过程1100还可以包括:生成包括第一比特流和第二比特流的经级联的比特流;以及将经级联的比特流发送到接收机。例如,神经网络压缩系统可以对第一比特流和第二比特流进行级联,并将经级联的比特流发送到接收机。在其他情况下,过程1100可以包括:将第一比特流和第二比特流分别发送到接收机。

[0213] 在一些情况下,过程1100可以包括:基于训练数据集来生成神经网络压缩系统的模型参数;使用输入数据来调谐神经网络压缩系统的模型参数;以及基于模型参数与经调谐的模型参数之间的差异来确定一组更新。在一些示例中,模型参数可以基于输入数据、输入数据的经压缩的版本的比特大小、一组更新的比特大小、以及输入数据与从输入数据的经压缩的版本生成的重构的数据之间的失真被调谐。

[0214] 在一些情况下,模型参数可以基于输入数据以及发送一组更新的成本与输入数据和从输入数据的经压缩的版本生成的重构的数据之间的失真的比率(例如,速率/失真比)被调谐。在一些示例中,该成本可以基于一组更新的比特大小。在一些示例中,调谐模型参数可以包括:基于以下确定将一个或多个参数添加/包括在经调谐的模型参数中:将一个或多个参数包括在经调谐的模型参数中伴随着输入数据的经压缩的版本的比特大小和/或输

入数据和从输入数据的经压缩的版本生成的重构的数据之间的失真的减少。

[0215] 在一些示例中,接收机可以包括解码器(例如,解码器514、解码器610、解码器718、解码器814、解码器922)。在一些情况下,过程1100可以包括:通过编码器接收包括第一比特流和第二比特流的数据。在一些情况下,过程1100还可以包括:通过解码器对基于第二比特流的一组经更新的参数的经压缩的版本进行解码;以及通过解码器使用一组经更新的参数,基于第一比特流中输入数据的经压缩的版本生成输入数据的经重构的版本。

[0216] 在一些示例中,过程1100可以包括:通过降低速率失真和模型速率损失来训练(例如,基于训练数据集)神经网络压缩系统。在一些示例中,模型速率反映了用于发送模型更新的比特流的长度。

[0217] 在一些示例中,过程1100可以实现前文描述的算法1和/或算法2。

[0218] 图12是示出用于对一个或多个图像进行压缩的过程1200的示例的流程图。在框1202处,过程1200可以包括:接收图像内容以进行压缩。在框1204处,过程1200可以包括:通过训练神经网络压缩系统来更新神经网络压缩系统,以提高用于压缩图像的压缩性能。在一些示例中,训练神经网络压缩系统可以包括:训练与神经网络压缩系统相关联的自动编码器系统。

[0219] 在框1206处,过程1200可以包括:使用经更新神经网络压缩系统将接收到的图像内容编码为潜在空间表示。

[0220] 在框1208处,过程1200可以包括:基于概率模型和第一代码子集,使用经更新神经网络压缩系统生成编码图像内容的经压缩的版本。第一代码子集可以包括潜在空间表示的一部分。例如,潜在空间表示可以分为第一代码子集和一个或多个附加代码子集。

[0221] 在框1210处,生成过程1200可以包括:使用概率模型和经更新神经网络压缩系统生成经更新神经网络压缩系统的经压缩的版本。在一些情况下,经更新神经网络压缩系统的经压缩的版本可以包括神经网络压缩系统的经量化参数更新,并且可以排除神经网络压缩系统的未更新参数。

[0222] 在框1212处,过程1200可以包括:输出经更新神经网络压缩系统的经压缩的版本以及编码图像内容的经压缩的版本用于传输。经更新神经网络压缩系统的经压缩的版本和编码图像内容的经压缩的版本可以被发送到接收机(例如,解码器)以进行解码。神经网络压缩系统的经压缩的版本可以包括神经网络压缩系统的完整模型或神经网络压缩系统的经更新的模型参数。

[0223] 在一些示例中,过程1200可以实现先前描述的算法1。

[0224] 图13是示出用于使用本文描述的技术对一个或多个图像进行解压缩的过程1300的示例的流程图。在框1302处,过程1300包括:接收经更新神经网络压缩系统的经压缩的版本(和/或神经网络压缩系统的一个或多个参数)以及编码图像内容的经压缩的版本。经更新神经网络压缩系统的经压缩的版本和编码图像内容的经压缩的版本可以如前文在图11或12中描述的那样被生成和发送。在一些情况下,编码图像内容可以包括从中生成编码图像内容的图像内容的潜在空间表示的第一代码子集。

[0225] 在一些情况下,经更新神经网络压缩系统的经压缩的版本可以包括模型参数。在一些示例中,模型参数可以包括经更新的模型参数,并排除一个或多个其他模型参数。

[0226] 在框1304处,过程1300可以包括:使用共享概率模型将经更新神经网络压缩系统

的经压缩的版本解压缩为经更新神经网络压缩系统模型。

[0227] 在框1306处,解压缩过程1300可以包括:使用经更新概率模型和经更新神经网络压缩系统模型将编码图像内容的经压缩的版本解压缩为潜在空间表示。

[0228] 在框1308处,处理过程1300可以包括:使用经更新神经网络压缩系统模型和潜在空间表示来生成重构的图像内容。

[0229] 在框1310处,过程1300可以包括:输出重构的图像内容。

[0230] 在一些示例中,过程1300可以实现上述算法2。

[0231] 在一些示例中,本文描述的过程(例如,过程1100、过程1200、过程1300和/或本文描述的任何其他过程)可由计算设备或装置执行。在一个示例中,过程1100和/或1200可由图4所示的系统400的发送设备410执行。在另一示例中,过程1100、1200和/或1300可以由根据图4所示的系统400或图14所示的计算系统1400的计算设备执行。

[0232] 计算设备可以包括任何合适的设备,例如移动设备(例如,移动电话)、桌面计算设备、平板计算设备、可穿戴设备(例如,VR耳机、AR耳机)、AR眼镜、联网手表或智能手表或其他可穿戴设备)、服务器计算机、自动驾驶汽车或自动驾驶汽车的计算设备、机器人设备、电视机和/或具有资源能力来执行本文中描述的过程的任何其他计算设备,包括过程1100、1200和1300。在一些情况下,计算设备或装置可以包括各种组件,例如一个或多个输入设备、一个或多个输出设备、一个或多个处理器、一个或多个微处理器、一个或多个微型计算机、一个或多个相机、一个或多个传感器和/或被配置为执行在本文中描述的过程的步骤的其他组件。在一些示例中,计算设备可以包括显示器、被配置为传送和/或接收数据的网络接口、它们的任意组合,和/或其他组件。网络接口可以被配置为传送和/或接收基于互联网协议(IP)的数据或其他类型的数据。

[0233] 计算设备的组件可以在电路中实现。例如,组件可以包括电子电路或其他电子硬件和/或可以使用其来实现,所述电子电路或其他电子硬件可以包括一个或多个可编程电子电路(例如,微处理器、图形处理单元(GPU)、数字信号处理器(DSP)、中央处理单元(CPU)和/或其他合适的电子电路),和/或可以包括计算机软件、固件或其任何组合和/或使用其来实现,以执行本文所述的各种操作。

[0234] 过程1100、1200和1300被示为逻辑流程图,其操作表示可以用硬件、计算机指令或其组合来实现的一系列操作。在计算机指令的上下文中,操作表示存储在一个或多个计算机可读存储介质上的计算机可执行指令,所述计算机可执行指令在由一个或多个处理器执行时执行所阐述的操作。通常,计算机可执行指令包括执行特定功能或实现特定数据类型的例程、程序、对象、组件、数据结构等。对操作进行描述的顺序不应当不解释为限制,并且可以以任何顺序和/或并行地对任何数量的所描述的操作进行组合以实现过程。

[0235] 另外,本文中描述的过程1100、1200和1300和/或任何其他过程可以在配置有可执行指令的一个或多个计算机系统的控制下执行,并且可以被实现为通过硬件或其组合在一个或多个处理器上共同执行的代码(例如,可执行指令、一个或多个计算机程序或者一个或多个应用)。如上所述,代码可以例如以包括可由一个或多个处理器执行的多个指令的计算机程序的形式存储在计算机可读或机器可读存储介质上。计算机可读或机器可读存储介质可以是非暂时性的。

[0236] 图14是示出用于实施本文技术的某些方面的系统的示例的图。具体而言,图14示

出了计算系统1400的示例,其可以是例如构成内部计算系统的任何计算设备、远程计算系统、相机或其任何组件,在其中系统的组件使用连接1405相互通信。连接1405可以是使用总线的物理连接,或者是到处理器1410的直接连接,例如在芯片组架构中。连接1405也可以是虚拟连接、网络连接或者逻辑连接。

[0237] 在一些实施例中,计算系统1400是分布式系统,其中本公开内容中描述的功能可以分布在数据中心、多个数据中心、对等网络等之内。在一些实施例中,所描述的系统组件中的一个或多个代表许多这样的组件,每个组件执行描述该组件所针对的功能的一些或全部。在一些实施例中,组件可以是物理或虚拟设备。

[0238] 示例系统1400包括至少一个处理单元(CPU或处理器)1410以及将包括系统存储器1415(例如只读存储器(ROM)1420和随机存取存储器(RAM)1425)的各种系统组件耦合至处理器1410的连接1405。计算系统1400可以包括与处理器1410直接连接、紧邻或集成为处理器610的一部分的高速存储器的高速缓存器1412。

[0239] 处理器1410可以包括任何通用处理器和硬件服务或软件服务,例如存储在存储设备1430中的服务1432、1434和1436,它们被配置为控制处理器1410以及其中软件指令被并入实际处理器设计的专用处理器。处理器1410可以本质上是一个完全独立的计算系统,包含多个核或处理器、总线、存储器控制器、高速缓存等。多核处理器可以是对称的或非对称的。

[0240] 为了实现用户交互,与计算系统1400包括输入设备1445,其可以表示任何数量的输入机制,例如用于语音的麦克风、用于手势或图形输入的触敏屏幕、键盘、鼠标、运动输入、语音,等等。计算系统1400还可以包括输出设备1435,其可以是多个输出机制中的一个或多个。在一些情况下,多模式系统可以使用户能够提供多种类型的输入/输出以与计算系统1400通信。计算系统1400可以包括通信接口1440,其通常可以支配和管理用户输入和系统输出。

[0241] 通信接口可以使用有线和/或无线收发机来执行或促进接收和/或发送有线或无线通信,包括使用下列各项的收发机:音频插孔/插头、麦克风插孔/插头、通用串行总线(USB)端口/插头、Apple® Lightning®端口/插头、以太网端口/插头、光纤端口/插头、专有有线端口/插头、BLUETOOTH®无线信号传输、BLUETOOTH®低功耗(BLE)无线信号传输、IBEAICON®无线信号传输、射频识别(RFID)无线信号传输、近场通信(NFC)无线信号传输、专用短距离通信(DSRC)无线信号传输、802.11Wi-Fi无线信号传输、无线局域网(WLAN)信号传输、可见光通信(VLC)、全球微波接入互操作性(WiMAX)、红外(IR)通信无线信号传输、公共交换电话网络(PSTN)信号传输、综合业务数字网络(ISDN)信号传输、3G/4G/5G/LTE蜂窝数据网络无线信号传输、自组织网络信号传输、无线电波信号传输、微波信号传输、红外信号传输、可见光信号传输、紫外光信号传输、沿电磁频谱的无线信号传输,或者它们的某种组合。

[0242] 通信接口1440还可以包括一个或多个全球导航卫星系统(GNSS)接收机或收发机,用于基于从与一个或多个GNSS系统相关联的一个或多个卫星接收的一个或多个信号来确定计算系统1400的位置。GNSS系统包括但不限于美国的全球定位系统(GPS)、俄罗斯的全球导航卫星系统(GLONASS)、中国的北斗导航卫星系统(BDS)以及欧洲的伽利略(Galileo)GNSS。对在任何特定硬件布置上的操作没有限制,因此这里的基本特征可以很容易地替换

为改进的硬件或固件布置(当它们在开发中)。

[0243] 存储设备1630可以是非易失性和/或非暂时性和/或计算机可读存储器设备,并且可以是硬盘或可以存储计算机可存取数据的其他类型的计算机可读介质,例如磁带盒、闪存卡、固态存储器设备、数字多功能磁盘、盒式磁带、软盘、软磁盘、硬盘、磁带、磁条/磁条带、任何其他磁性存储介质、闪存、存储器存储、任何其他固态存储器、压缩盘只读存储器(CD-ROM)光盘、可重写压缩盘(CD)光盘、数字视频盘(DVD)光盘、蓝光盘(BDD)光盘、全息光盘、另一种光学介质、安全数字(SD)卡、微型安全数字(microSD)卡、MemoryStick®卡、智能卡芯片、EMV芯片、用户身份模块(SIM)卡、微小/微型/纳米/微微SIM卡、另一种集成电路(IC)芯片/卡、随机存取存储器(RAM)、静态RAM(SRAM)、动态RAM(DRAM)、只读存储器(ROM)、可编程只读存储器(PROM)、可擦除可编程只读存储器(EPROM)、电可擦除可编程只读存储器(EEPROM)、闪存EPROM(FLASH EPROM)、高速缓存器(L1/L2/L3/L4/L5/L#)、电阻式随机存取存储器(RRAM/ReRAM)、相变存储器(PCM)、自旋转移矩RAM(STT-RAM)、其他存储器芯片或盒式存储器,和/或它们的组合。

[0244] 存储设备1430可以包括软件服务、服务器、服务等,当处理器1410执行定义这种软件的代码时,它使系统执行某种功能。在一些实施例中,执行特定功能的硬件服务可以包括存储在计算机可读介质中的与必要的硬件组件(例如处理器1410、连接1405、输出设备1435等)连接的软件组件,以执行该功能。术语“计算机可读介质”包括但不限于便携式或非便携式存储设备、光学存储设备以及能够存储、包含或携带指令和/或数据的各种其他介质。计算机可读介质可以包括其中可以存储数据的非暂时性介质,并且不包括无线地或通过有线连接传播的载波和/或暂时性电子信号。非暂时性介质的示例可以包括但不限于磁盘或磁带、诸如光盘(CD)或数字通用光盘(DVD)的光学存储介质、闪存器、存储器或存储设备。计算机可读介质上可以存储有代码和/或机器可执行指令,其可以表示过程、功能、子程序、程序、例程、模块、软件包、类、或者指令、数据结构或程序语句的任意组合。代码段可以通过传递和/或接收信息、数据、自变量、参数或存储器内容来连接到另一个代码段或硬件电路。信息、自变量、参数数据等可经由包括存储器共享、消息传递、令牌传递、网络传输等的任何合适的手段来传递、转发或发送。

[0245] 在一些实施例中,计算机可读存储设备、介质和存储器可以包括包含比特流等的电缆或无线信号。然而,当提及,非暂时性计算机可读存储介质明确地排除诸如能量、载波信号、电磁波和信号本身之类的介质。

[0246] 在上文的说明中提供了具体细节来提供对本文中提供的实施例和示例的透彻理解。然而,本领域技术的普通技术人员将意理解的是,可以在不使用这些具体细节的情况下来实施这些实施例。为了解释清楚起见,在一些情况下,本文的技术可以被呈现为包括单独的功能块,这些功能块包括设备、设备组件、以软件体现的方法中的步骤或例程,或者硬件和软件的组合。除了图中所示和/或本文所述的组件外,还可以使用其他组件。例如,可以将电路、系统、网络、过程和其他组件以框图形式示为组件,以免在不必要的细节上模糊实施例。例如,可以在没有非必要细节的情况下示出公知的电路、过程、算法、结构和技术以避免模糊实施例。

[0247] 各个实施例可以作为过程或方法来描述,过程或方法被描绘成了流程图、流图、数据流图、结构图或框图。虽然流程图可以将操作描述为顺序过程,但操作中的许多操作可以

并行或并发地执行。此外,可以对这些操作的顺序进行重新布置。过程在其操作完成后会终止,但是过程可以具有图中没的其他步骤。过程可以对应于方法、函数、过程、子例程,子程序等。当过程对应于函数时,其终止可以对应于该函数返回到调用函数或主函数。

[0248] 可以使用存储在计算机可读介质中或从计算机可读介质中可获得的计算机可执行指令来实现根据上述示例的过程和方法。例如,这样的指令可以包括使通用计算机、专用计算机或处理设备或以其他方式配置通用计算机、专用计算机或处理设备执行某个功能或功能的群组的指令和数据。可以通过网络访问所使用的计算机资源的一些部分。计算机可执行指令可以是例如二进制文件、如汇编语言的中间格式指令、固件、源代码等。可用于存储指令、所用信息和/或在根据所述示例的方法期间创建的信息的计算机可读介质的示例包括磁盘或光盘、闪存、配备有非易失性存储器的USB设备、网络存储设备等等。

[0249] 实现根据这些公开内容的过程和方法的设备可以包括硬件、软件、固件、中间件、微代码、硬件描述语言或其任意组合,并且可以采用多种外形尺寸中的任意一种。当以软件、固件、中间件,或微代码来实现时,用于执行必要任务的程序代码或代码段(例如,计算机程序产品)可以存储在计算机可读或机器可读介质中。处理器可以执行必要的任务。外形尺寸的典型示例包括膝上型计算机、智能电话、移动电话、平板电脑设备或其他外形小巧的个人计算机、个人数字助理、机架安装设备、独立设备等等。本文描述的功能也可以体现在外围设备或附加卡中。作为进一步的示例,这种功能还可以在单个设备中执行的不同芯片或不同过程之间的电路板上实现。

[0250] 指令、用于传达这些指令的介质、用于执行它们的计算资源以及用于支持这样的计算资源的其他结构是用于提供本公开内容中描述的功能的示例单元。

[0251] 在前文的描述中,参考本申请的特定实施例描述了本申请的各个方面,但是本领域技术人员将认识到,本申请不限于此。因此,尽管在本文中已经详细描述了本申请的说明性实施例,但是应当理解,这些发明构思可以用其他方式被不同地体现和采用,并且所附权利要求旨在被解释为包括这样的变型,除了现有技术所限制的以外。可以单独或联合使用上述申请的各种特征和方面。此外,在不脱离本说明书的更广泛的精神和范围的情况下,实施例可以在本文所述之外的任何数量的环境和应用中使用。因此,应该认为本说明和附图是说明性而非限制性的。为了说明的目的,以特定顺序描述了方法。应当认识到的是,在替代实施例中,可以用与所描述的顺序不同的顺序来执行这些方法。

[0252] 普通技术人员将理解,本文中使用的小于("<")和大于(">")符号或术语可以在不脱离本说明书的范围的情况下替换为小于或等于("<=")和大于或等于(">=")符号。

[0253] 在将组件描述为“被配置为”执行某些操作的情况下,可以例如通过设计电子电路或其他硬件以执行该操作,通过对可编程电子电路(例如微处理器或其他合适的电子电路)进行编程以执行该操作或其任意组合来实现这种配置。

[0254] 短语“耦合至”是指直接或间接物理连接到另一个组件的任何组件,和/或直接或间接与另一个组件进行通信的任何组件(例如,通过有线或无线连接和/或其他合适的通信接口连接到另一个组件)。

[0255] 引用集合中的“至少一个”和/或集合中的“一个或多个”的权利要求语言或其他语言表示该集合中的一个成员或该集合中的多个成员(以任何组合)满足权利要求。例如,引用“A和B中的至少一个”或“A或B中的至少一个”的声明语言意指A、B或A和B。在另一个示例

中,引用“A、B和C中的至少一个”或“A、B或C中的至少一个”的权利要求语言是指A,B,C或A和B,或A和C,或B和C,或A和B和C。语言集合中的“至少一个”和/或集合中的“一个或多个”不将集合限制为该集合中列出的项目。例如,引用“A和B中的至少一个”或“A或B中的至少一个”的声明语言可以意指A,B或A和B,并且可以另外包括未在A和B的集合中列出的项目。

[0256] 结合本文公开的示例描述的各个说明性的逻辑框、模块、电路和算法步骤均可以实现成电子硬件、计算机软件或它们的组合。为了清楚地表示硬件和软件之间的该可交换性,上文对各个说明性的组件、框、模块、电路和步骤均围绕其功能进行了总体描述。至于这种功能是实现为硬件还是实现成软件,取决于特定的应用和对整个系统所施加的设计约束。本领域技术人员可以针对每个特定应用,以变通的方式实现所描述的功能,但是这种实现决策不应解释为造成对本申请的范围的背离。

[0257] 本文描述的技术还可以以电子硬件、计算机软件、固件或其任何组合来实现。可以在诸如通用计算机、无线通信手持设备或具有包括在无线通信手持设备和其他设备中的应用的多种用途的集成电路设备等各种设备中的任何一种设备中实现这种技术。描述为模块或组件的任何特征可以一起实现在集成逻辑器件中,或者单独实现为分立但可互操作的逻辑器件。如果以软件实现,则可以至少部分通过包括程序代码的计算可读数据存储介质来实现这些技术,程序代码包括指令,当被执行时,所述指令执行上述方法、算法和/或操作中的一项或多项。计算机可读数据存储介质可以形成计算机程序产品的一部分,计算机程序产品可以包括封装材料。计算机可读介质可以包括存储器或数据存储介质,例如随机存取存储器(RAM)(如同步动态随机存取存储器(SDRAM))、只读存储器(ROM)、非易失性随机存取存储器(NVRAM)、电可擦除可编程只读存储器(EEPROM)、FLASH存储器、磁或光数据存储介质等。附加地或替代地,这些技术可以至少部分由计算机可读通信介质实现,所述通信介质承载或传送指令或数据结构形式的程序代码并且可以由计算机访问、读取和/或执行,例如传播信号或波形。

[0258] 程序代码可由处理器执行,处理器可以包括诸如一个或多个数字信号处理器(DSP)、通用微处理器、专用集成电路(ASIC)、现场可编程逻辑阵列(FPGA)或其他等效集成或离散逻辑电路的一个或多个处理器。这样的处理器可以被配置为执行本公开内容中描述的任何技术。通用处理器可以是微处理器,或者,处理器可以是任何常规的处理器、控制器、微控制器或者状态机。处理器也可以实现为计算设备的组合,例如,DSP和微处理器的组合、多个微处理器、一个或多个微处理器与DSP内核的结合、或者任何其他这种配置。因此,本文中所使用的术语“处理器”可以指代任意前述结构,或者前述结构的任意组合,或者适于本文中描述的技术的实现的任何其他结构或装置。

[0259] 本公开内容的说明性示例包括:

[0260] 方面1:一种装置,包括:存储器;以及耦合至所述存储器的一个或多个处理器,所述一个或多个处理器被配置为:通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的第二比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0261] 方面2:根据方面1所述的装置,其中,所述第二比特流还包括所述潜在先验的经压缩的版本和所述模型先验的经压缩的版本。

[0262] 方面3:根据方面1至2中任意方面所述的装置,其中,所述一个或多个处理器被配置为:生成包括所述第一比特流和所述第二比特流的经级联的比特流;以及将所述经级联的比特流发送到所述接收机。

[0263] 方面4:根据方面1至3中任意方面所述的装置,其中,为了生成所述第二比特流,所述一个或多个处理器被配置为:通过所述神经网络压缩系统使用所述模型先验对所述潜在先验进行熵编码;以及通过所述神经网络压缩系统使用所述模型先验对所述经更新的模型参数进行熵编码。

[0264] 方面5:根据方面1至4中任意方面所述的装置,其中,所述经更新的模型参数包括解码器模型的一个或多个经更新的参数,所述一个或多个经更新的参数使用所述输入数据被调谐。

[0265] 方面6:根据方面1至5中任意方面所述的装置,其中,所述经更新的模型参数包括所述编码器模型的一个或多个经更新的参数,所述一个或多个经更新的参数使用所述输入数据被调谐,其中,所述第一比特流是通过所述神经网络压缩系统使用所述一个或多个经更新的参数生成的。

[0266] 方面7:根据方面6所述的装置,其中,为了生成所述第二比特流,所述一个或多个处理器被配置为:通过所述神经网络压缩系统使用所述一个或多个经更新的参数将所述输入数据编码为所述输入数据的潜在空间表示;以及通过所述神经网络压缩系统使用所述潜在先验将所述潜在空间表示熵编码为所述第一比特流。

[0267] 方面8:根据方面1至7中任意方面所述的装置,其中,所述一个或多个处理器被配置为:基于用于训练所述神经网络压缩系统的训练数据集来生成所述神经网络压缩系统的模型参数;使用所述输入数据来调谐所述神经网络压缩系统的所述模型参数;以及基于所述模型参数与经调谐的模型参数之间的差异来确定所述一组更新。

[0268] 方面9:根据方面8所述的装置,其中,所述模型参数是基于所述输入数据、所述输入数据的所述经压缩的版本的比特大小、所述一组更新的比特大小、以及所述输入数据与从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真被调谐的。

[0269] 方面10:根据方面8所述的装置,其中,所述模型参数是基于所述输入数据以及发送所述一组更新的成本与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真的比率被调谐的,所述成本基于所述一组更新的比特大小。

[0270] 方面11:根据方面8所述的装置,其中,为了调谐所述模型参数,所述一个或多个处理器配置为:基于以下确定将一个或多个参数包括在所述经调谐的模型参数中:将所述一个或多个参数包括在所述经调谐的模型参数中伴随着以下各项中的至少一项的减少:所述输入数据的所述经压缩的版本的比特大小以及所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真。

[0271] 方面12:根据方面1至11中任意方面所述的装置,其中,为了确定用于所述神经网络压缩系统的所述一组更新,所述一个或多个处理器被配置为:处理所述神经网络压缩系统处的所述输入数据;基于经处理的输入数据确定所述神经网络压缩系统的一个或多个损失;以及基于所述一个或多个损失来调谐所述神经网络压缩系统的模型参数,经调谐的模

型参数包括用于所述神经网络压缩系统的所述一组更新。

[0272] 方面13:根据方面12所述的装置,其中,所述一个或多个损失包括:与用于基于所述第一比特流的大小来发送所述输入数据的所述经压缩的版本的速率相关联的速率损失,与所述输入数据和从所述输入数据的经压缩的版本生成的重构的数据之间的失真相关联的失真损失,以及与用于基于所述第二比特流的大小来发送经更新的模型参数的所述经压缩的版本的速率相关联的模型速率损失。

[0273] 方面14:根据方面1至13中任意方面所述的装置,其中,所述接收机包括编码器,并且其中,所述一个或多个处理器被配置为:通过所述编码器接收包括所述第一比特流和所述第二比特流的数据;通过所述解码器对基于所述第二比特流的所述经更新的模型参数的所述经压缩的版本进行解码;以及通过所述解码器使用所述一组经更新的参数,基于所述第一比特流中所述输入数据的所述经压缩的版本生成所述输入数据的经重构的版本。

[0274] 方面15:根据方面1至14中任意方面所述的装置,其中,所述一个或多个处理器被配置为:通过降低速率失真和模型速率损失来训练所述神经网络压缩系统,其中,模型速率反映了用于发送模型更新的比特流的长度。

[0275] 方面16:根据方面1至15中任意方面所述的装置,其中,所述模型先验包括以下各项中的至少一项:独立高斯网络先验、独立拉普拉斯网络先验,以及独立Spike和Slab网络先验。

[0276] 方面17:根据方面1至16中任意方面所述的装置,其中,所述装置包括移动设备。

[0277] 方面18:根据方面1至17中任意方面所述的装置,还包括被配置为捕获所述输入数据的相机。

[0278] 方面19:一种方法,包括:通过神经网络压缩系统接收输入数据以便通过所述神经网络压缩系统进行压缩;确定用于所述神经网络压缩系统的一组更新,所述一组更新包括使用所述输入数据调谐的经更新的模型参数;通过所述神经网络压缩系统使用潜在先验生成包括所述输入数据的经压缩的版本的所述第一比特流;通过所述神经网络压缩系统使用所述潜在先验和模型先验生成包括所述经更新的模型参数的经压缩的版本的第二比特流;以及输出所述第一比特流和所述第二比特流以传输到接收机。

[0279] 方面20:根据方面19所述的方法,其中,所述第二比特流还包括所述潜在先验的经压缩的版本和所述模型先验的经压缩的版本。

[0280] 方面21:根据方面19至20中任意方面所述的方法,其中,所述一个或多个处理器被配置为:生成包括所述第一比特流和所述第二比特流的经级联的比特流;以及将所述经级联的比特流发送到所述接收机。

[0281] 方面22:根据方面19至21中任意方面所述的方法,其中,生成所述第二比特流包括:通过所述神经网络压缩系统使用所述模型先验对所述潜在先验进行熵编码;以及通过所述神经网络压缩系统使用所述模型先验对所述经更新的模型参数进行熵编码。

[0282] 方面23:根据方面19至22中任意方面所述的方法,其中,所述经更新的模型参数包括解码器模型的一个或多个经更新的参数,所述一个或多个经更新的参数使用所述输入数据被调谐。

[0283] 方面24:根据方面19至23中任意方面所述的方法,其中,所述经更新的模型参数包括所述编码器模型的一个或多个经更新的参数,所述一个或多个经更新的参数使用所述输

入数据被调谐,其中,所述第一比特流是通过所述神经网络压缩系统使用所述一个或多个经更新的参数生成的。

[0284] 方面25:根据方面24所述的方法,其中,生成所述第二比特流包括:通过所述神经网络压缩系统使用所述一个或多个经更新的参数将所述输入数据编码为所述输入数据的潜在空间表示;以及通过所述神经网络压缩系统使用所述潜在先验将所述潜在空间表示熵编码为所述第一比特流。

[0285] 方面26:根据方面19至25中任意方面所述的方法,其中,所述一个或多个处理器被配置为:基于用于训练所述神经网络压缩系统的训练数据集来生成所述神经网络压缩系统的模型参数;使用所述输入数据来调谐所述神经网络压缩系统的所述模型参数;以及基于所述模型参数与经调谐的模型参数之间的差异来确定所述一组更新。

[0286] 方面27:根据方面26所述的方法,其中,所述模型参数是基于所述输入数据、所述输入数据的所述经压缩的版本的比特大小、所述一组更新的比特大小、以及所述输入数据与从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真被调谐的。

[0287] 方面28:根据方面26所述的方法,其中,所述模型参数是基于所述输入数据以及发送所述一组更新的成本与所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真的比率被调谐的,所述成本基于所述一组更新的比特大小。

[0288] 方面29:根据方面26所述的方法,其中,调谐所述模型参数包括:基于以下确定将一个或多个参数包括在所述经调谐的模型参数中:将所述一个或多个参数包括在所述经调谐的模型参数中伴随着以下各项中的至少一项的减少:所述输入数据的所述经压缩的版本的比特大小以及所述输入数据和从所述输入数据的所述经压缩的版本生成的重构的数据之间的失真。

[0289] 方面30:根据方面19至29中任意方面所述的方法,其中,确定用于所述神经网络压缩系统的所述一组更新包括:处理所述神经网络压缩系统处的所述输入数据;基于经处理的输入数据确定所述神经网络压缩系统的一个或多个损失;以及基于所述一个或多个损失来调谐所述神经网络压缩系统的模型参数,经调谐的模型参数包括用于所述神经网络压缩系统的所述一组更新。

[0290] 方面31:根据方面30所述的方法,其中,所述一个或多个损失包括:与用于基于所述第一比特流的大小来发送所述输入数据的所述经压缩的版本的速率相关联的速率损失,与所述输入数据和从所述输入数据的经压缩的版本生成的重构的数据之间的失真相关联的失真损失,以及与用于基于所述第二比特流的大小来发送经更新的模型参数的所述经压缩的版本的速率相关联的模型速率损失。

[0291] 方面32:根据方面19至31中任意方面所述的方法,其中,所述接收机包括编码器,并且其中,所述一个或多个处理器被配置为:通过所述编码器接收包括所述第一比特流和所述第二比特流的数据;通过所述解码器对基于所述第二比特流的所述经更新的模型参数的所述经压缩的版本进行解码;以及通过所述解码器使用所述一组经更新的参数,基于所述第一比特流中所述输入数据的所述经压缩的版本生成所述输入数据的经重构的版本。

[0292] 方面33:根据方面19至32中任意方面所述的方法,其中,所述一个或多个处理器被配置为:通过降低速率失真和模型速率损失来训练所述神经网络压缩系统,其中,模型速率反映了用于发送模型更新的比特流的长度。

[0293] 方面34:根据方面19至33中任意方面所述的方法,其中,所述模型先验包括以下各项中的至少一项:独立高斯网络先验、独立拉普拉斯网络先验,以及独立Spike和Slab网络先验。

[0294] 方面35:一种非暂时性计算机可读介质,其上存储有指令,所述指令当由一个或多个处理器执行时,使所述一个或多个处理器执行根据方面19至33中任意方面的方法。

[0295] 方面36:一种装置,包括用于执行根据方面19至33中任意方面所述的方法的单元。

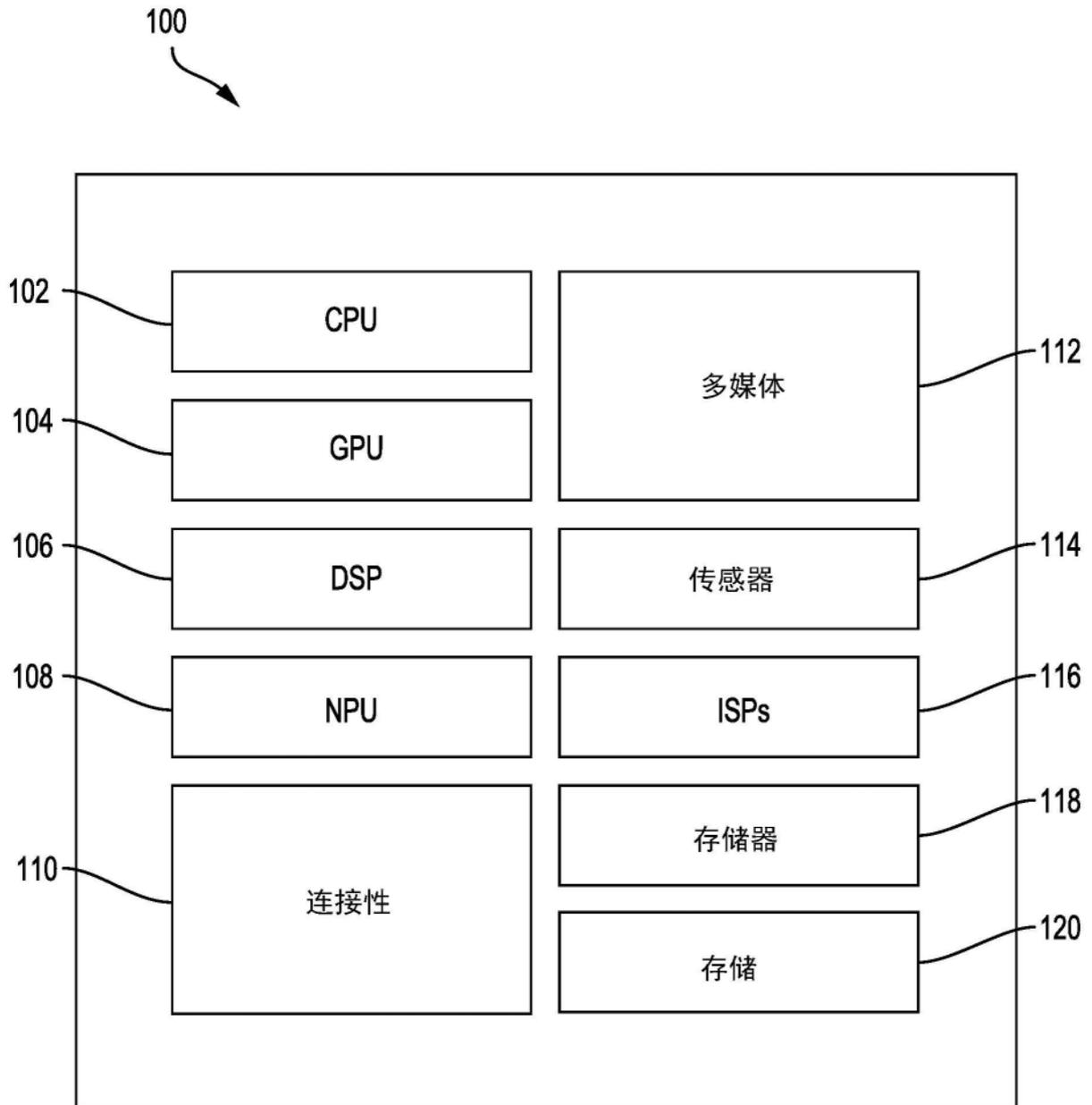


图1

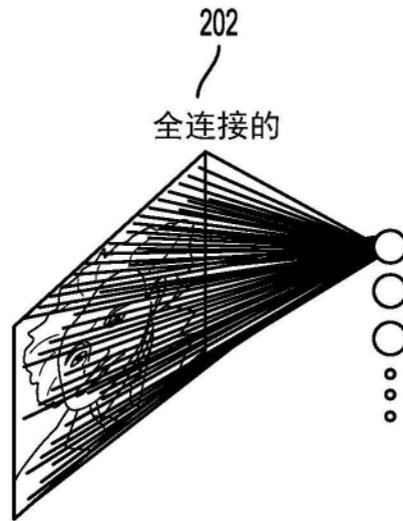


图2A

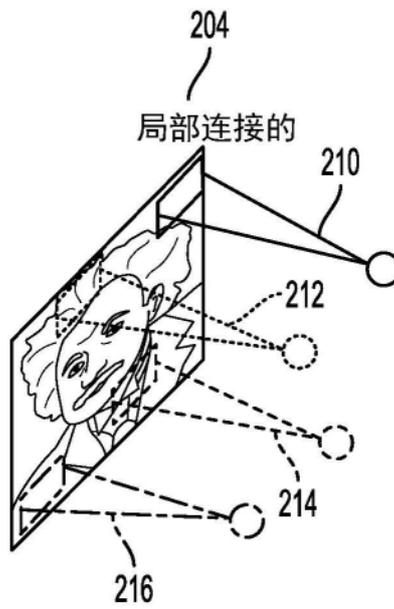


图2B

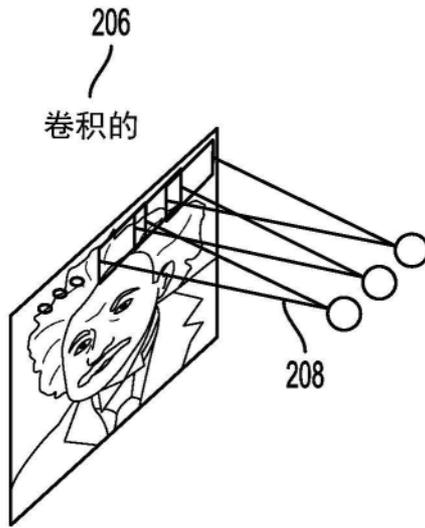


图2C

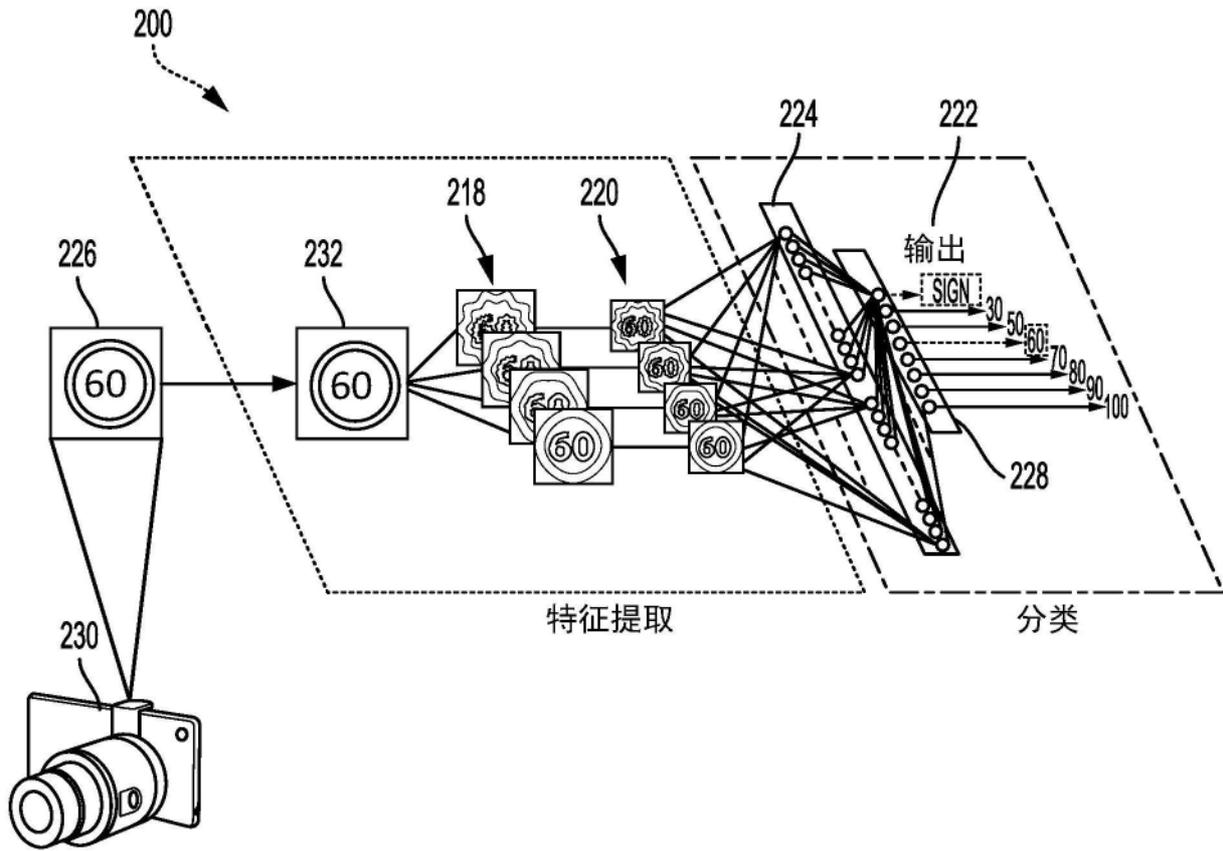


图2D

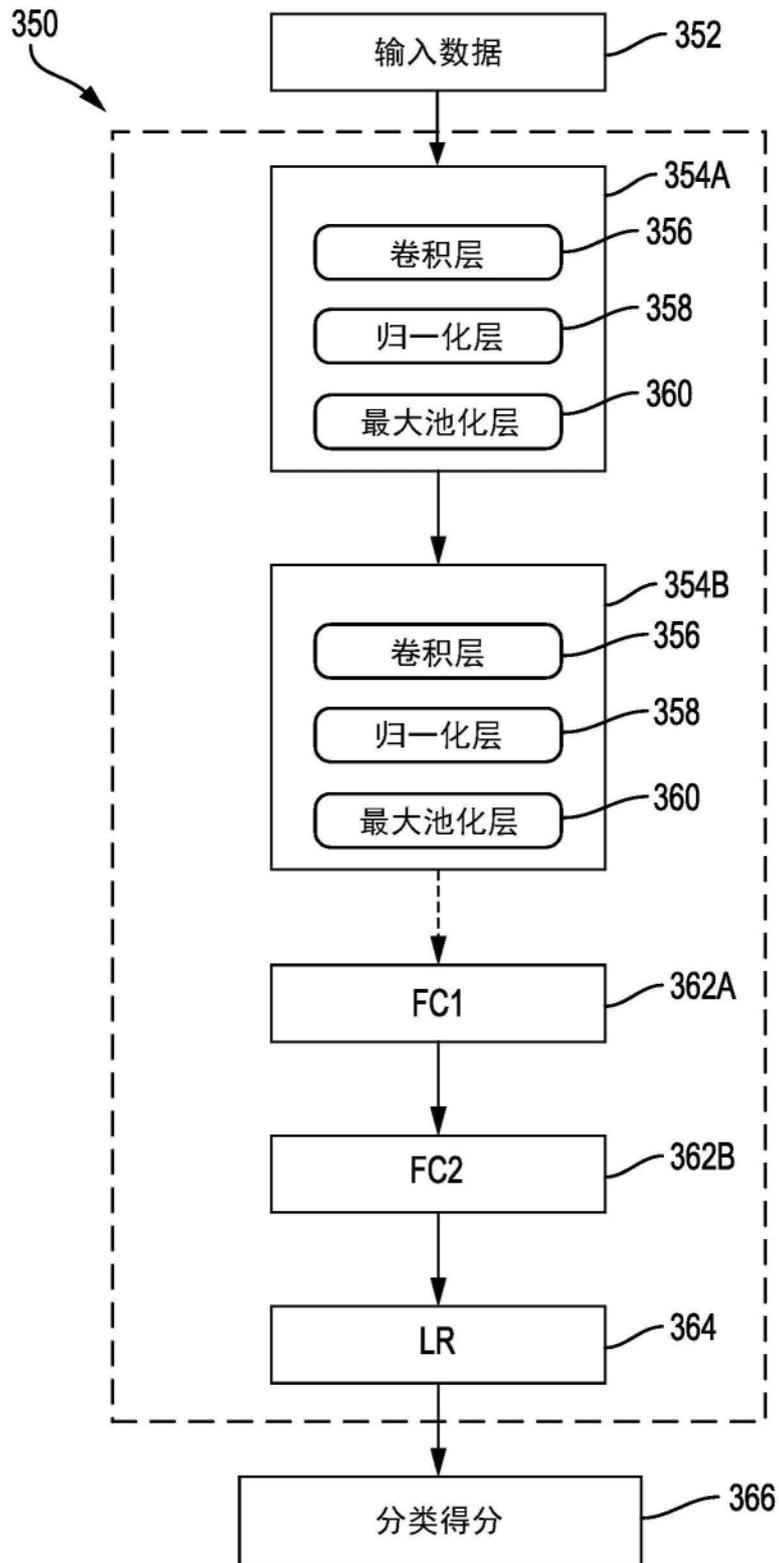


图3

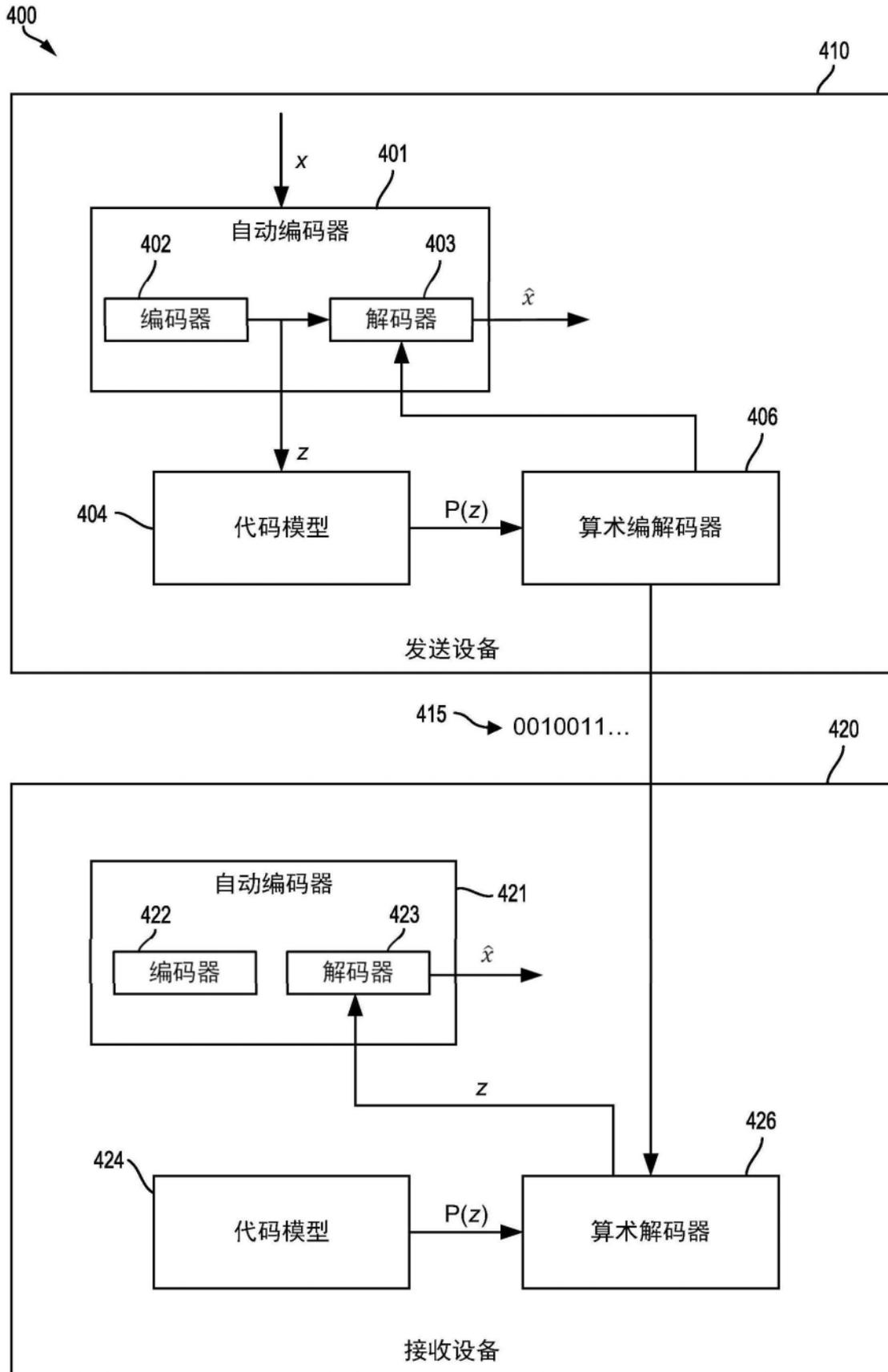


图4

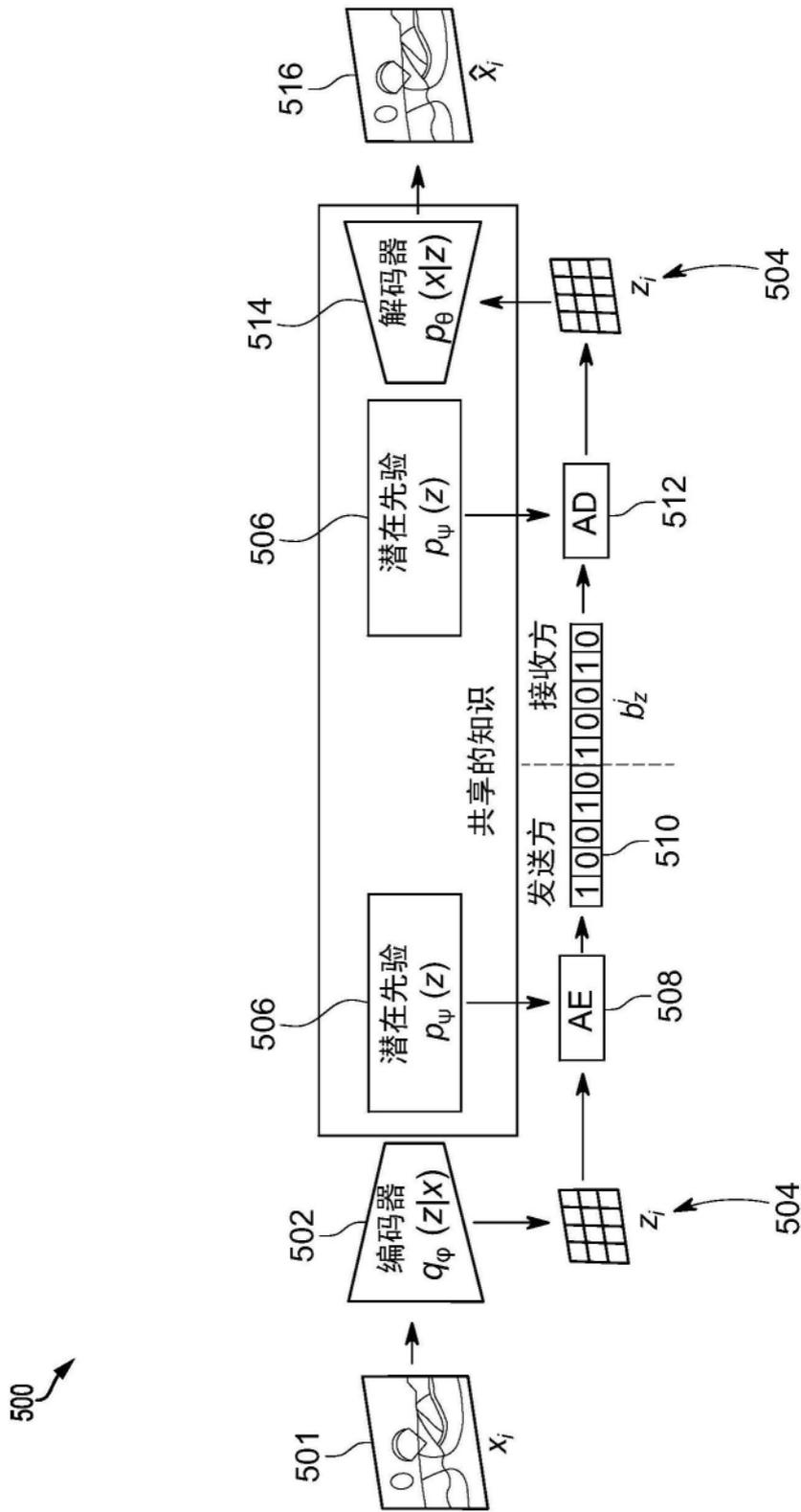


图5A



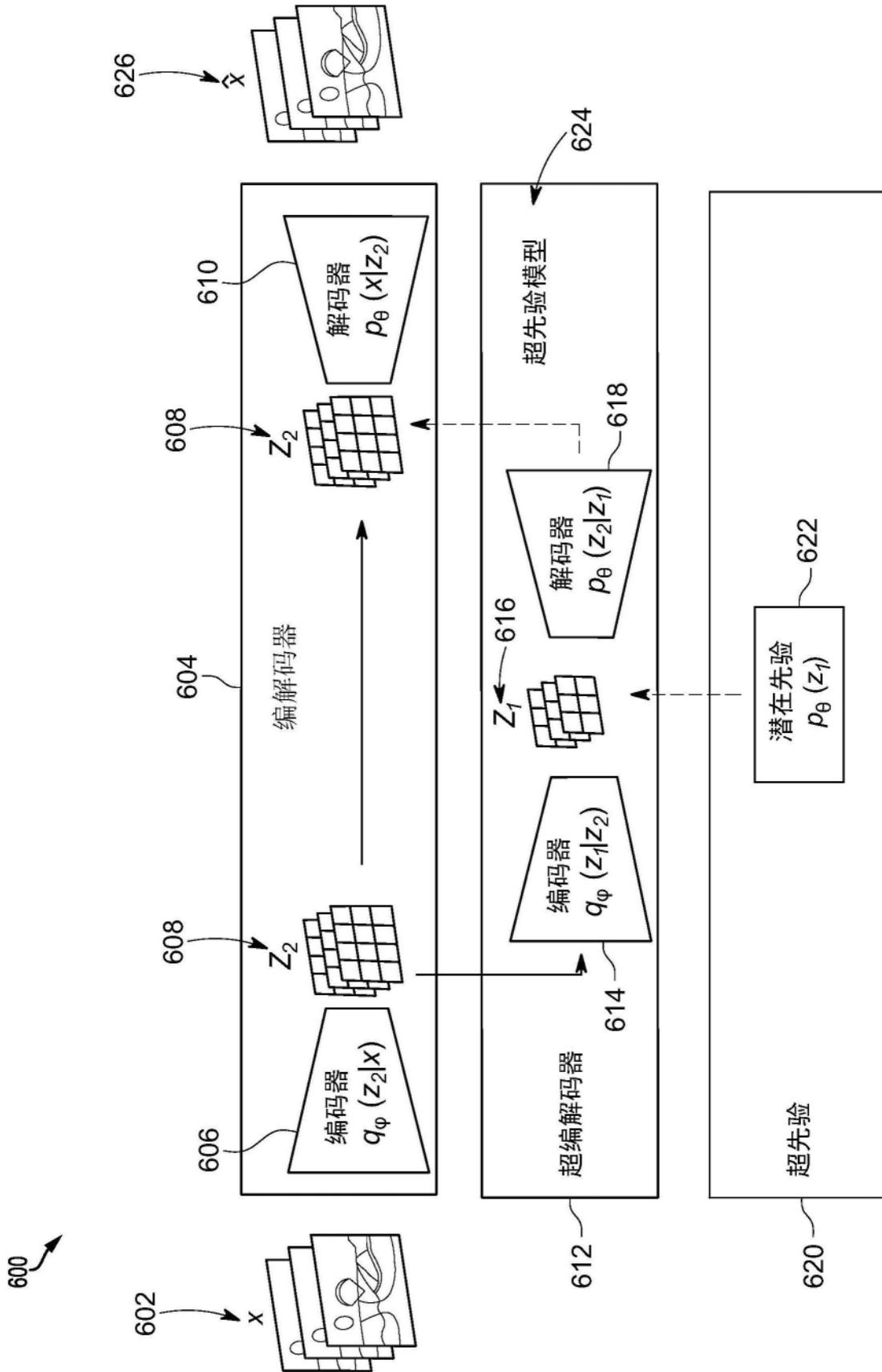


图6

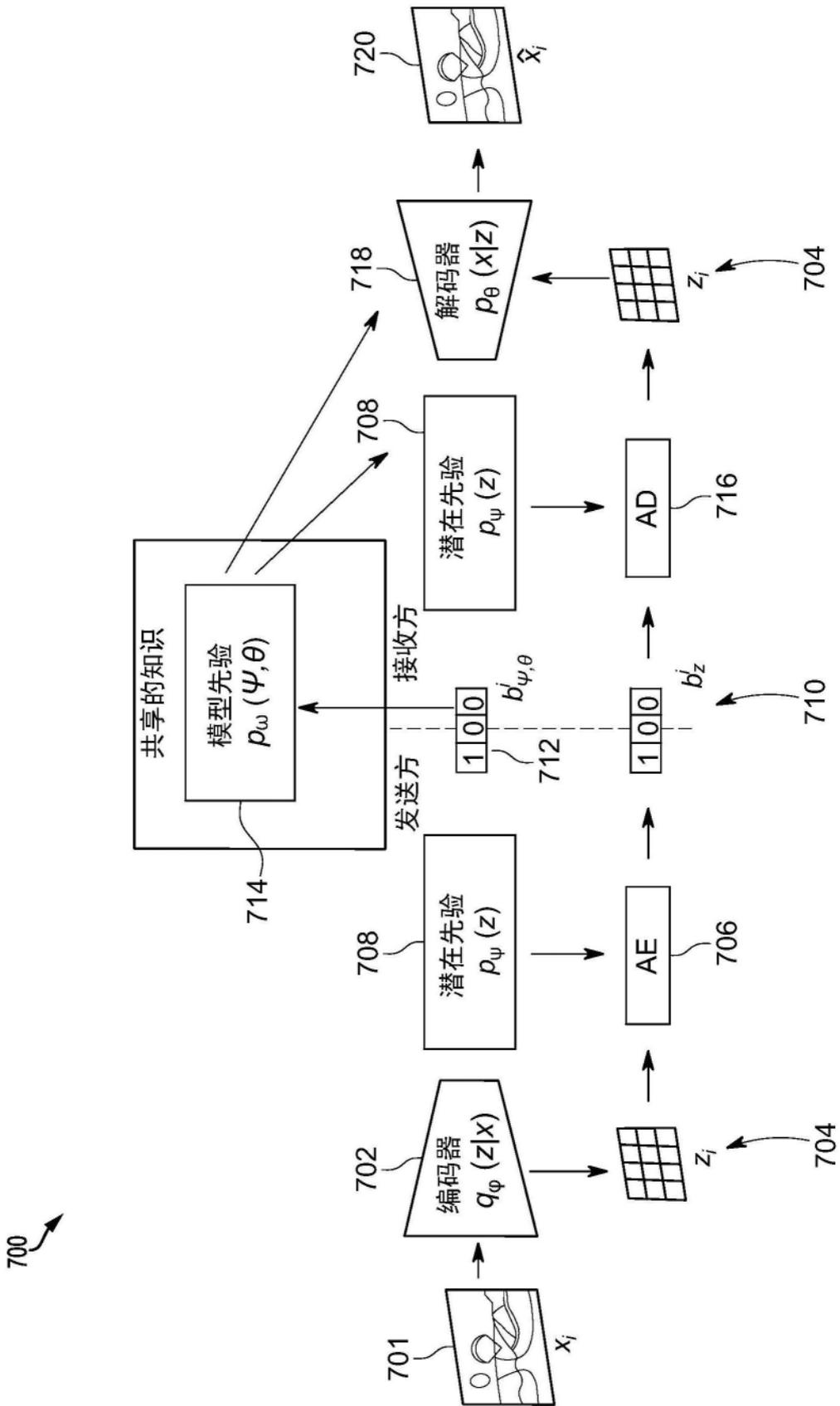


图7



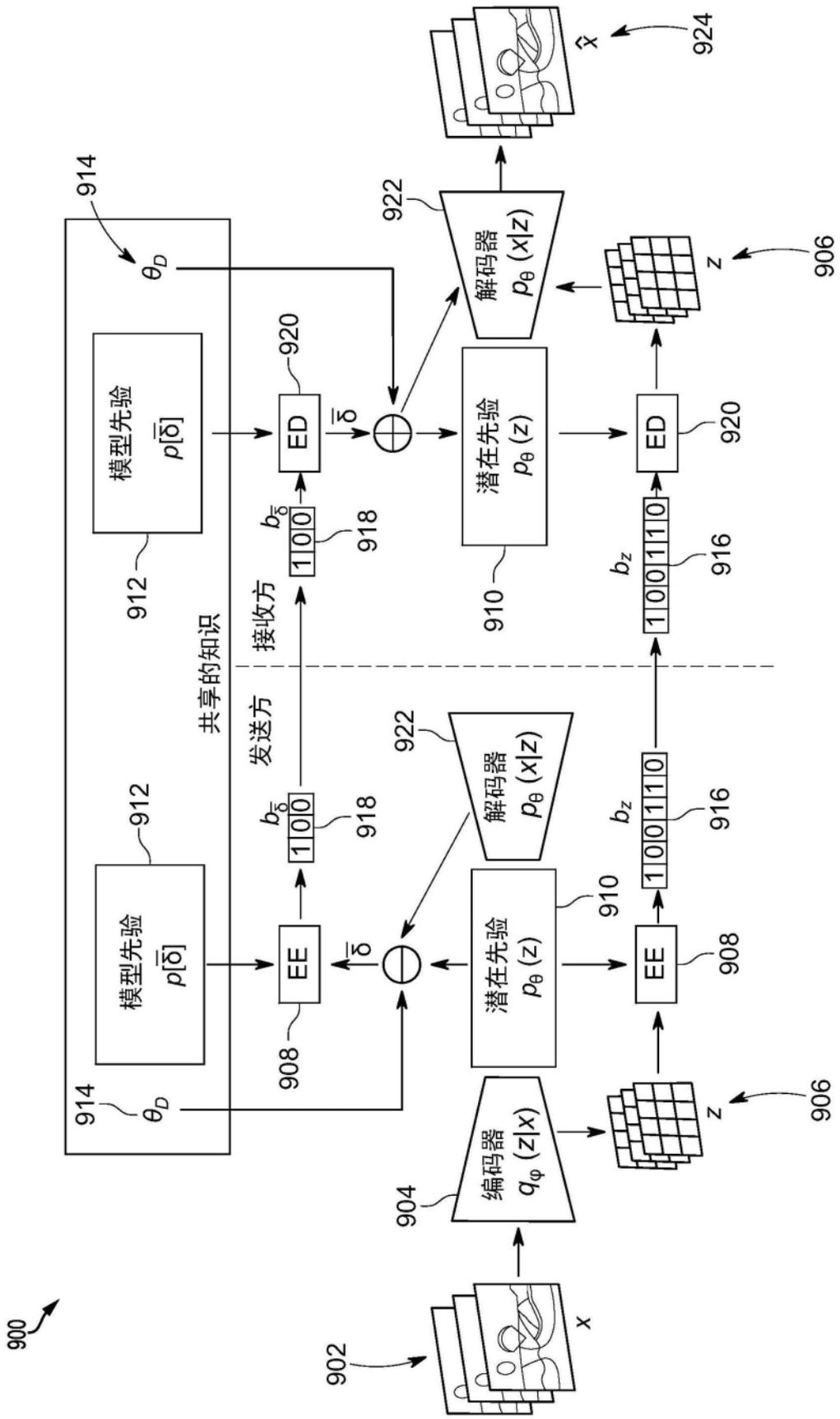


图9

1000 ↗

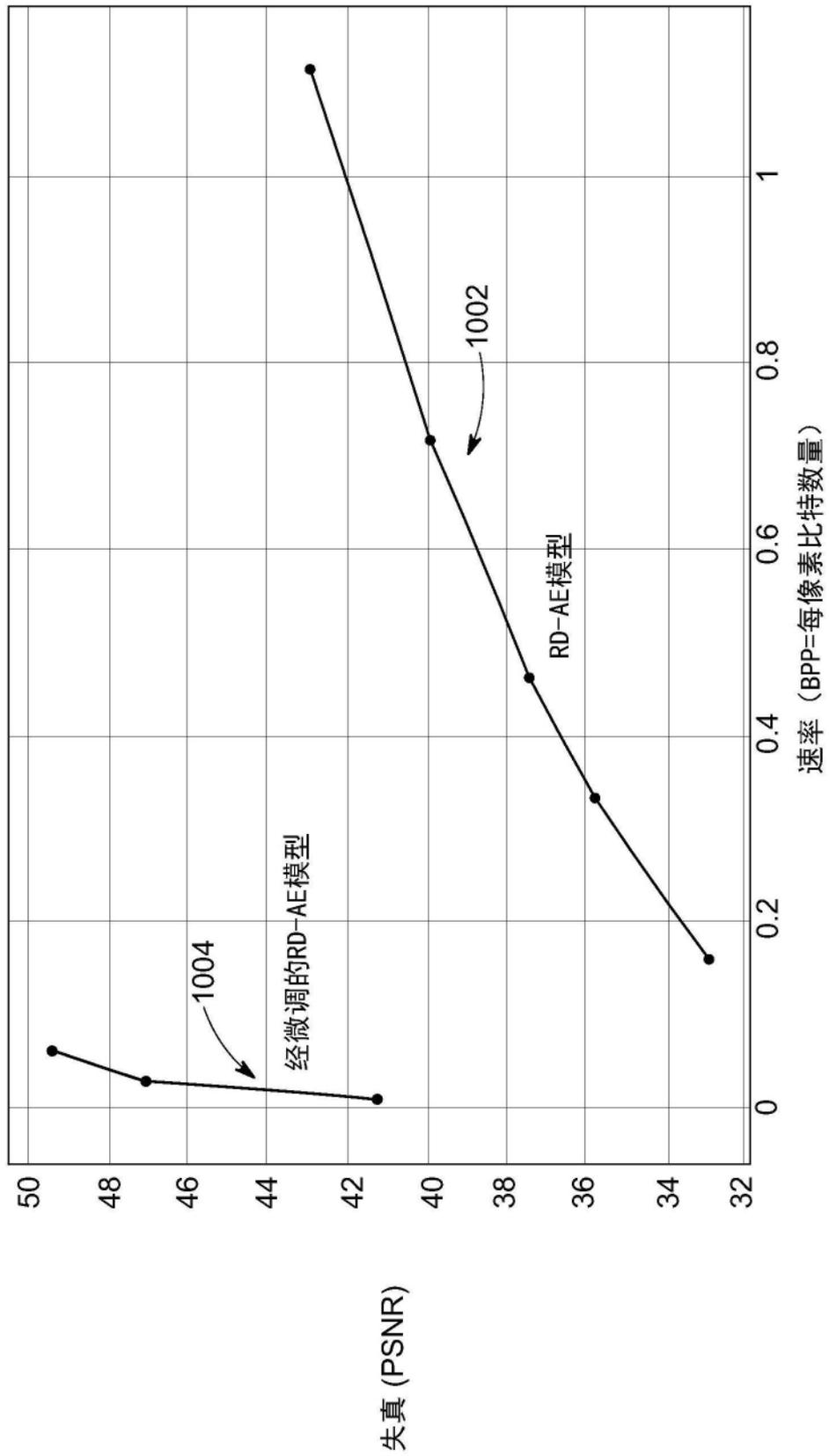


图10

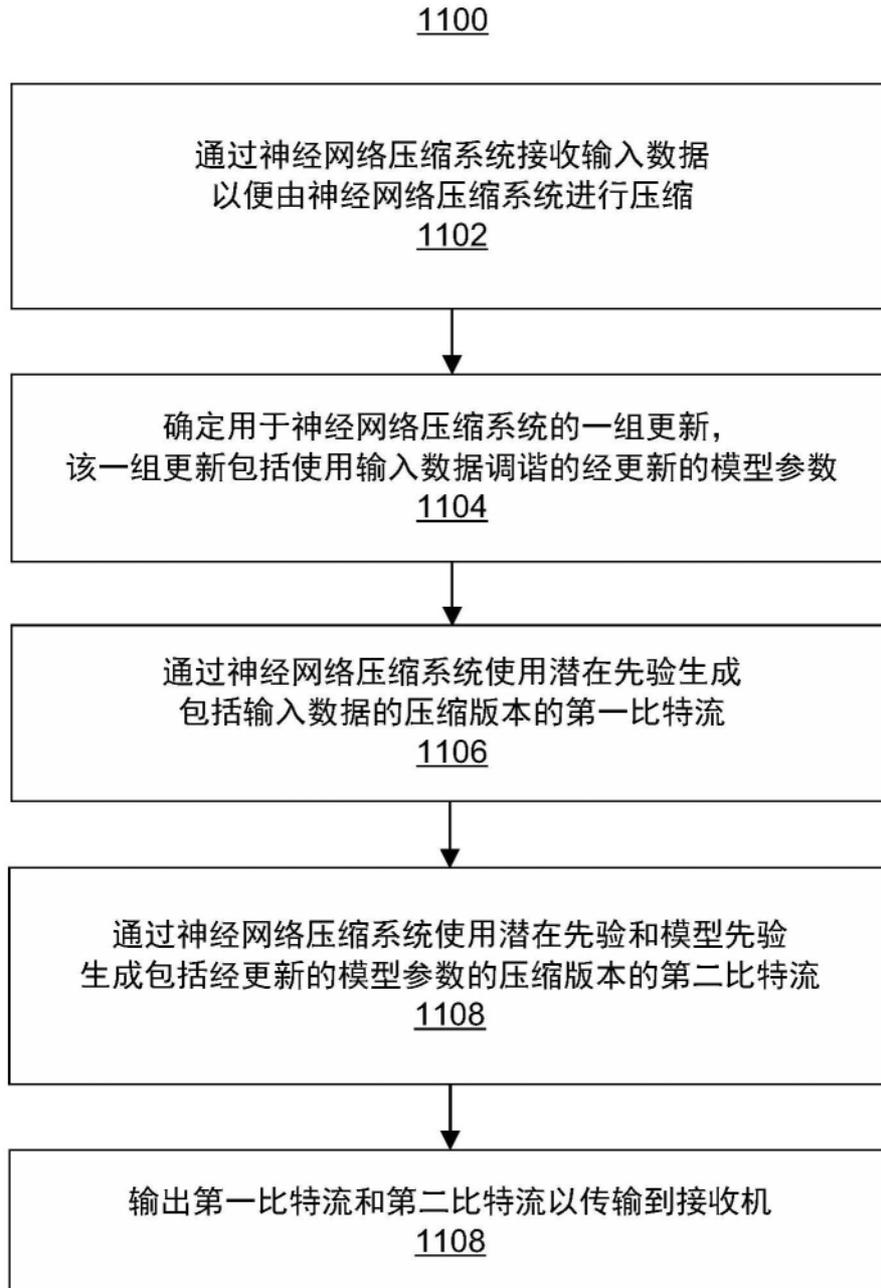


图11

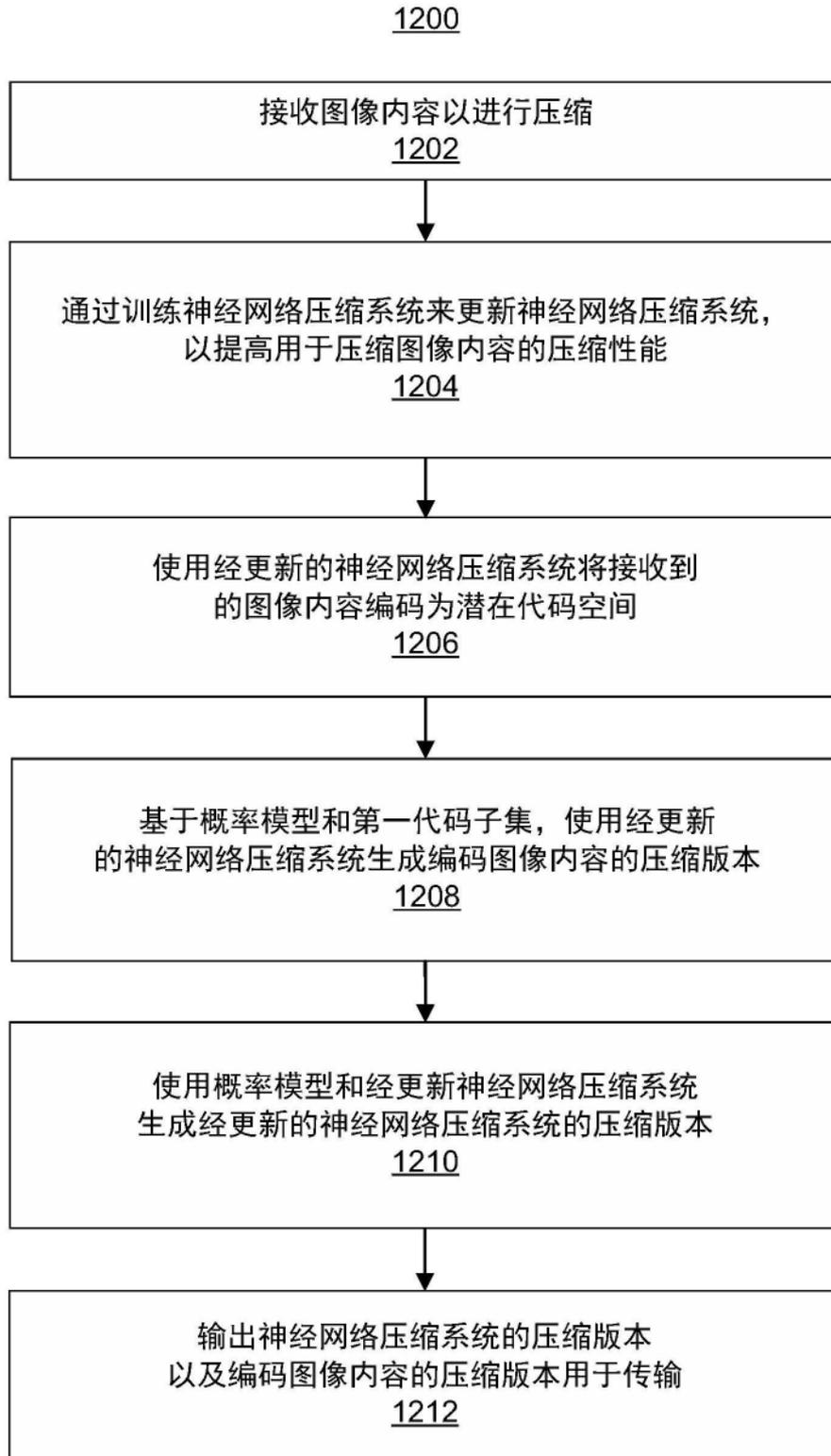


图12

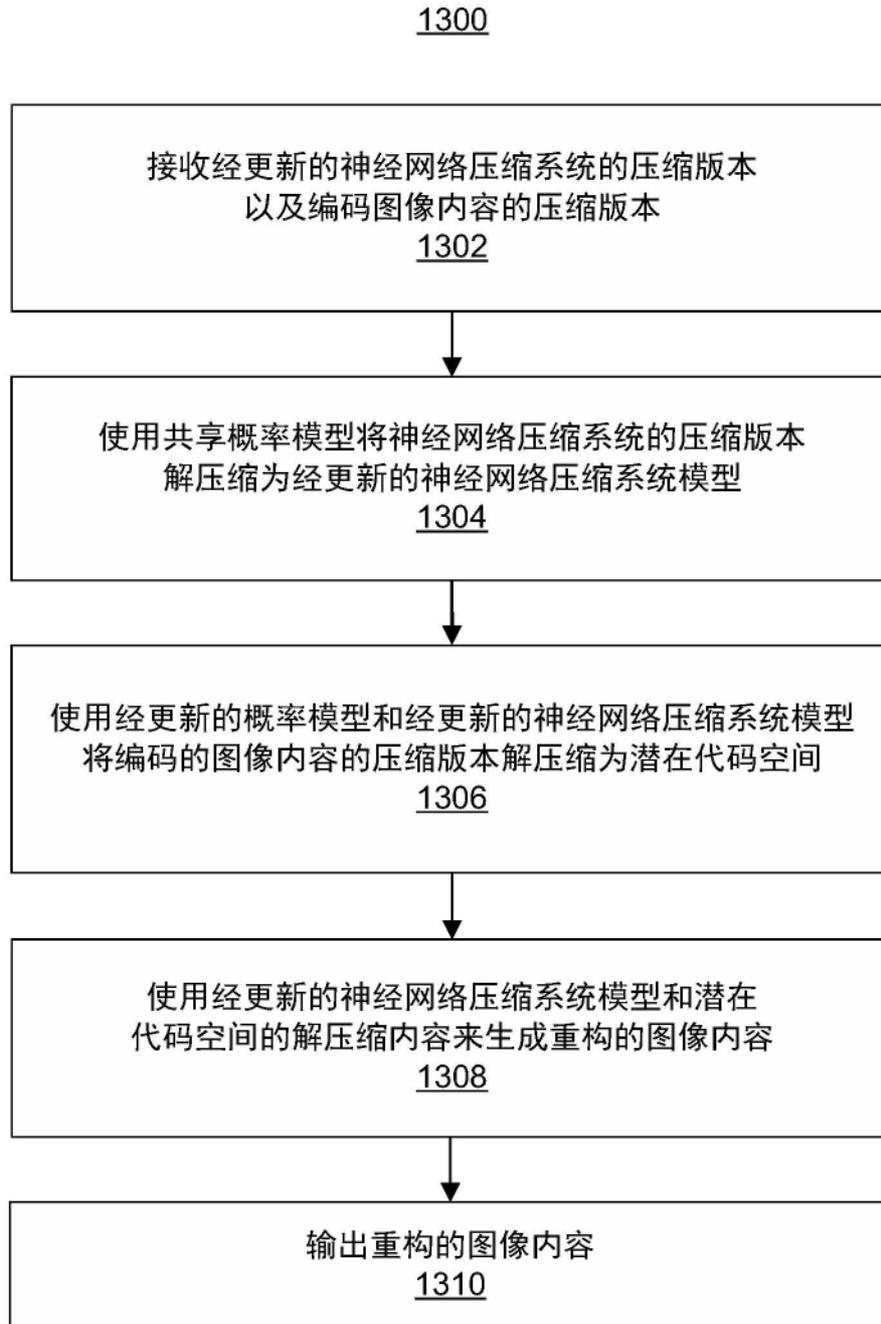


图13

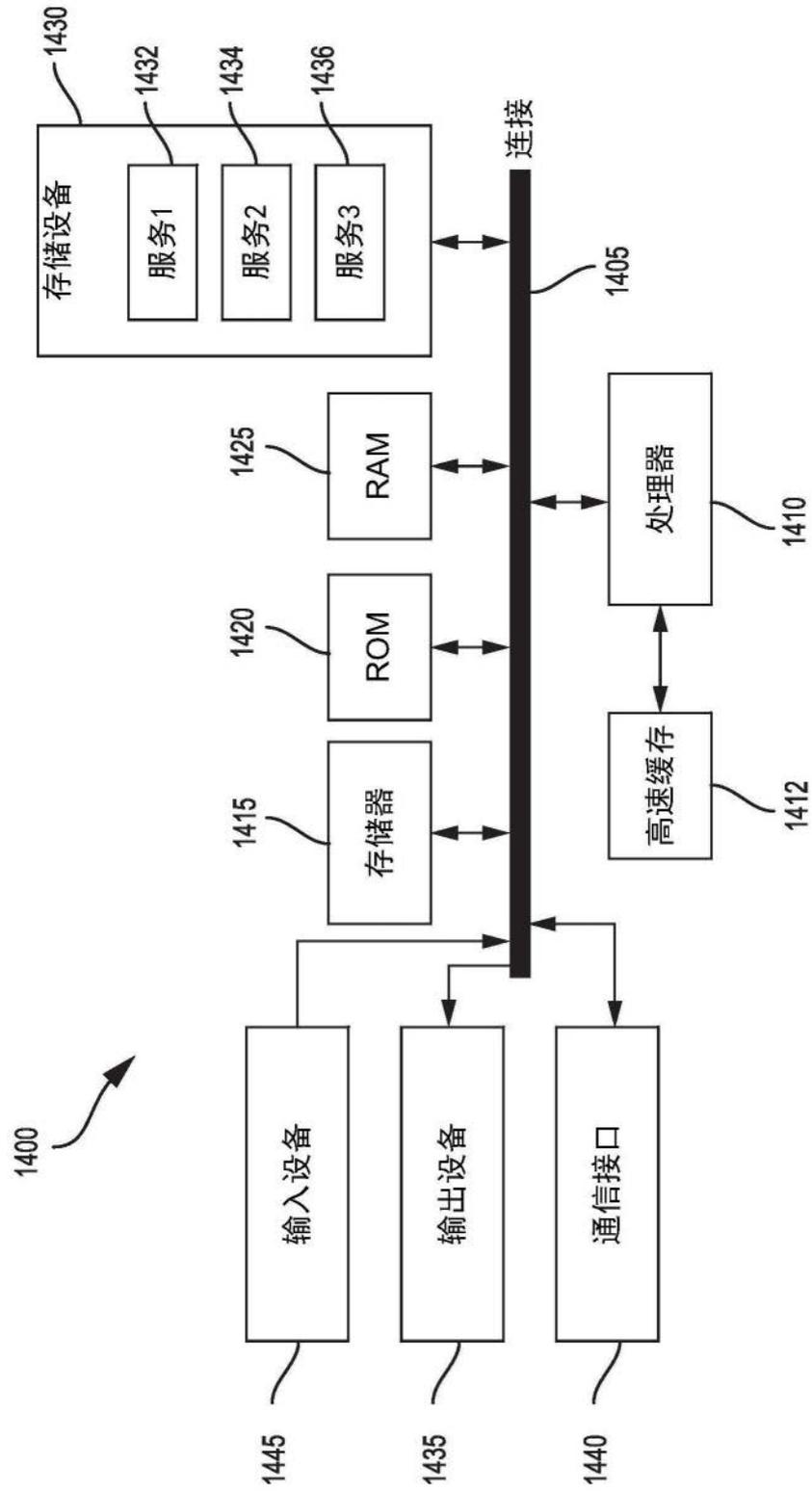


图14