(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0074137
(43) 공개일자 2023년05월26일

(51) 국제특허분류(Int. Cl.)
H04N 19/13 (2014.01) *G06N 3/045* (2023.01)
G06N 3/0455 (2023.01) *G06N 3/047* (2023.01)
G06N 3/08 (2023.01) *G06T 9/00* (2019.01)
H04N 19/147 (2014.01) *H04N 19/184* (2014.01)
H04N 19/463 (2014.01)
(52) CPC특허분류
H04N 19/13 (2015.01)
G06N 3/045 (2023.01)
(21) 출원번호 10-2023-7009609
(22) 출원일자(국제) 2021년08월30일
심사청구일자 없음
(85) 번역문제출일자 2023년03월20일
(86) 국제출원번호 PCT/US2021/048245
(87) 국제공개번호 WO 2022/066368
국제공개일자 2022년03월31일
(30) 우선권주장
63/083,747 2020년09월25일 미국(US)
17/201,944 2021년03월15일 미국(US)

(71) 출원인
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
판 로전달 티스 예한
미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775
하위번 이리스 안네 마리
미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775
코헌 타코 세바스티안
미국 92121 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

전체 청구항 수 : 총 30 항

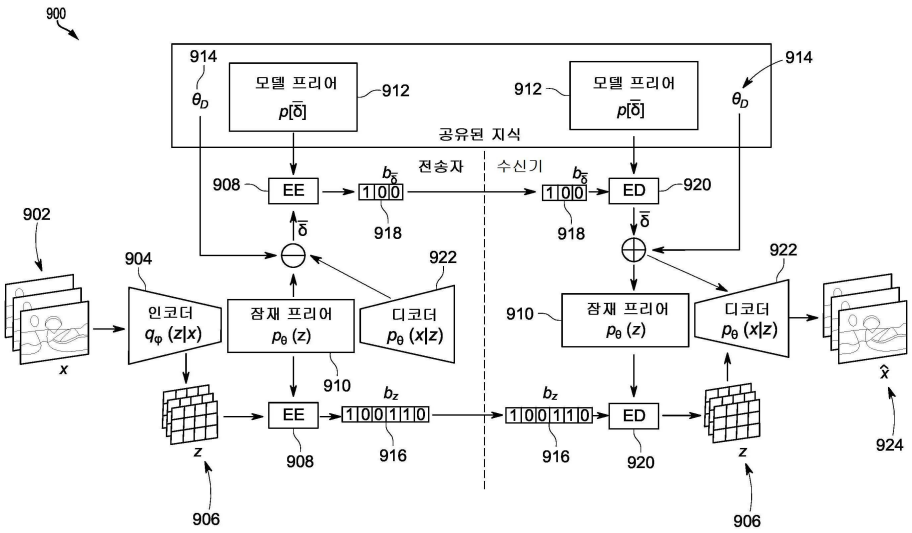
(54) 발명의 명칭 **머신 러닝 시스템들을 이용한 인스턴스 적응적 이미지 및 비디오 압축**

(57) 요 약

머신 러닝 시스템들을 사용하여 데이터를 압축하고, 데이터를 압축하기 위해 머신 러닝 시스템들을 튜닝하기 위한 기술들이 설명된다. 예시적인 프로세스는 (예컨대, 트레이닝 데이터세트에 대해 트레이닝되는) 신경망 압축 시스템에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하는 것을 포함할 수 있다.

(뒷면에 계속)

대 표 도



프로세스는 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것을 포함할 수 있고, 그 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함한다. 프로세스는 신경망 압축 시스템에 의해 잠재 프리어를 사용하여, 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하는 것을 포함할 수 있다. 프로세스는 신경망 압축 시스템에 의해 잠재 프리어 및 모델 프리어를 사용하여, 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하는 것을 더 포함할 수 있다. 프로세스는 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하는 것을 포함할 수 있다.

(52) CPC특허분류

G06N 3/0455 (2023.01)
G06N 3/047 (2023.01)
G06N 3/084 (2023.01)
G06N 3/088 (2023.01)
G06T 9/002 (2013.01)
H04N 19/147 (2015.01)
H04N 19/184 (2015.01)
H04N 19/463 (2015.01)
G06T 2207/20084 (2013.01)

명 세 서

청구범위

청구항 1

장치로서,

메모리; 및

상기 메모리에 커플링된 하나 이상의 프로세서들을 포함하고,

상기 하나 이상의 프로세서들은:

신경망 압축 시스템에 의해, 상기 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하고;

상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것으로서, 상기 업데이트들의 세트는 상기 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함하는, 상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 결정하는 것을 행하며;

상기 신경망 압축 시스템에 의해 잠재 프리어(latent prior)를 사용하여, 상기 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하고;

상기 신경망 압축 시스템에 의해 상기 잠재 프리어 및 모델 프리어(model prior)를 사용하여, 상기 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하고; 그리고

수신기로의 전송을 위해 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 출력하도록

구성되는, 장치.

청구항 2

제 1 항에 있어서,

상기 제 2 비트스트림은 상기 잠재 프리어의 압축된 버전 및 상기 모델 프리어의 압축된 버전을 더 포함하는, 장치.

청구항 3

제 1 항에 있어서,

상기 하나 이상의 프로세서들은:

상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 연접된 비트스트림을 생성하고; 그리고

상기 연접된 비트스트림을 상기 수신기로 전송하도록

구성되는, 장치.

청구항 4

제 1 항에 있어서,

상기 제 2 비트스트림을 생성하기 위해, 상기 하나 이상의 프로세서들은:

상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 잠재 프리어를 엔트로피 인코딩하고; 그리고

상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 업데이트된 모델 파라미터들을 엔트로피 인코딩하도록

구성되는, 장치.

청구항 5

제 1 항에 있어서,

상기 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 상기 입력 데이터를 사용하여 튜닝되는, 장치.

청구항 6

제 1 항에 있어서,

상기 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 상기 입력 데이터를 사용하여 튜닝되고, 상기 제 1 비트스트림은 상기 하나 이상의 업데이트된 파라미터들을 사용하여 상기 신경망 압축 시스템에 의해 생성되는, 장치.

청구항 7

제 6 항에 있어서,

상기 제 2 비트스트림을 생성하기 위해, 상기 하나 이상의 프로세서들은:

상기 신경망 압축 시스템에 의해 상기 하나 이상의 업데이트된 파라미터들을 사용하여, 상기 입력 데이터를 상기 입력 데이터의 잠재 공간 표현으로 인코딩하고; 그리고

상기 신경망 압축 시스템에 의해 상기 잠재 프리어를 사용하여, 상기 잠재 공간 표현을 상기 제 1 비트스트림으로 엔트로피 인코딩하도록

구성되는, 장치.

청구항 8

제 1 항에 있어서,

상기 하나 이상의 프로세서들은:

상기 신경망 압축 시스템을 트레이닝하기 위해 사용되는 트레이닝 데이터세트에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 생성하고;

상기 입력 데이터를 사용하여 상기 신경망 압축 시스템의 상기 모델 파라미터들을 튜닝하며; 그리고

상기 모델 파라미터들과 튜닝된 상기 모델 파라미터들 사이의 차이에 기초하여 상기 업데이트들의 세트를 결정하도록

구성되는, 장치.

청구항 9

제 8 항에 있어서,

상기 모델 파라미터들은, 상기 입력 데이터, 상기 입력 데이터의 상기 압축된 버전의 비트 사이즈, 상기 업데이트들의 세트의 비트 사이즈, 및 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에 기초하여 튜닝되는, 장치.

청구항 10

제 8 항에 있어서,

상기 모델 파라미터들은 상기 입력 데이터, 및 상기 업데이트들의 세트를 전송하는 비용 및 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡의 비율에 기초하여 튜닝되고, 상기 비용은 상기 업데이트들의 세트의 비트 사이즈에 기초하는, 장치.

청구항 11

제 8 항에 있어서,

상기 모델 파라미터들을 튜닝하기 위해, 상기 하나 이상의 프로세서들은:

튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함하는 것이 상기 입력 데이터의 상기 압축된 버전의 비트 사이즈 및 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡 중 적어도 하나에서의 감소를 수반한다는 결정에 기초하여 상기 튜닝된 모델 파라미터들에 상기 하나 이상의 파라미터들을 포함시키도록 구성되는, 장치.

청구항 12

제 1 항에 있어서,

상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 결정하기 위해, 상기 하나 이상의 프로세서들은:

상기 신경망 압축 시스템에서 상기 입력 데이터를 프로세싱하고;

프로세싱된 상기 입력 데이터에 기초하여 상기 신경망 압축 시스템에 대한 하나 이상의 손실들을 결정하며; 그리고

상기 하나 이상의 손실들에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하도록

구성되고,

튜닝된 상기 모델 파라미터들은 상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 포함하는, 장치.

청구항 13

제 12 항에 있어서,

상기 하나 이상의 손실들은 상기 제 1 비트스트림의 사이즈에 기초하여 상기 입력 데이터의 상기 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 상기 제 2 비트스트림의 사이즈에 기초하여 상기 업데이트된 모델 파라미터들의 상기 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함하는, 장치.

청구항 14

제 1 항에 있어서,

상기 수신기는 인코더를 포함하고,

상기 하나 이상의 프로세서들은:

상기 인코더에 의해, 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 데이터를 수신하고;

디코더에 의해, 상기 제 2 비트스트림에 기초하여 상기 업데이트된 모델 파라미터들의 상기 압축된 버전을 디코딩하며; 그리고

업데이트된 파라미터들의 세트를 사용하여 상기 디코더에 의해, 상기 제 1 비트스트림에서의 상기 입력 데이터의 상기 압축된 버전에 기초하여 상기 입력 데이터의 재구성된 버전을 생성하도록

구성되는, 장치.

청구항 15

제 1 항에 있어서,

상기 하나 이상의 프로세서들은:

레이트-왜곡 및 모델-레이트 손실을 감소시킴으로써 상기 신경망 압축 시스템을 트레이닝시키도록 구성되고, 여기서 모델-레이트는 모델 업데이트들을 전송하기 위한 비트스트림의 길이를 반영하는, 장치.

청구항 16

제 1 항에 있어서,

상기 모델 프리어는 독립 가우시안 네트워크 프리어, 독립 라플라스 네트워크 프리어, 및 독립 스파이크 앤 슬래브 네트워크 프리어 중 적어도 하나를 포함하는, 장치.

청구항 17

제 1 항에 있어서,

상기 장치는 모바일 디바이스를 포함하는, 장치.

청구항 18

제 1 항에 있어서,

상기 입력 데이터를 캡처하도록 구성된 카메라를 더 포함하는, 장치.

청구항 19

방법으로서,

신경망 압축 시스템에 의해, 상기 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하는 단계;

상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 단계로서, 상기 업데이트들의 세트는 상기 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함하는, 상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 결정하는 단계;

상기 신경망 압축 시스템에 의해 잠재 프리어(latent prior)를 사용하여, 상기 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하는 단계;

상기 신경망 압축 시스템에 의해 상기 잠재 프리어 및 모델 프리어(model prior)를 사용하여, 상기 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하는 단계; 및

수신기로의 전송을 위해 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 출력하는 단계를 포함하는, 방법.

청구항 20

제 19 항에 있어서,

상기 제 2 비트스트림은 상기 잠재 프리어의 압축된 버전 및 상기 모델 프리어의 압축된 버전을 더 포함하는, 방법.

청구항 21

제 19 항에 있어서,

하나 이상의 프로세서들이:

상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 연접된 비트스트림을 생성하고; 그리고

상기 연접된 비트스트림을 상기 수신기로 전송하도록

구성되는, 방법.

청구항 22

제 19 항에 있어서,

상기 제 2 비트스트림을 생성하는 단계는,

상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 잠재 프리어를 엔트로피 인코딩하는 단계; 및

상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 업데이트된 모델 파라미터들을 엔트로피 인코딩하는 단계를 포함하는, 방법.

청구항 23

제 19 항에 있어서,

상기 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 상기 입력 데이터를 사용하여 튜닝되는, 방법.

청구항 24

제 19 항에 있어서,

상기 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 상기 입력 데이터를 사용하여 튜닝되고, 상기 제 1 비트스트림은 상기 하나 이상의 업데이트된 파라미터들을 사용하여 상기 신경망 압축 시스템에 의해 생성되는, 방법.

청구항 25

제 24 항에 있어서,

상기 제 2 비트스트림을 생성하는 단계는,

상기 신경망 압축 시스템에 의해 상기 하나 이상의 업데이트된 파라미터들을 사용하여, 상기 입력 데이터를 상기 입력 데이터의 잠재 공간 표현으로 인코딩하는 단계; 및

상기 신경망 압축 시스템에 의해 상기 잠재 프리어를 사용하여, 상기 잠재 공간 표현을 상기 제 1 비트스트림으로 엔트로피 인코딩하는 단계를 포함하는, 방법.

청구항 26

제 19 항에 있어서,

하나 이상의 프로세서들이:

상기 신경망 압축 시스템을 트레이닝하기 위해 사용되는 트레이닝 데이터세트에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 생성하고;

상기 입력 데이터를 사용하여 상기 신경망 압축 시스템의 상기 모델 파라미터들을 튜닝하며; 그리고

상기 모델 파라미터들과 튜닝된 상기 모델 파라미터들 사이의 차이에 기초하여 상기 업데이트들의 세트를 결정하도록

구성되는, 방법.

청구항 27

제 26 항에 있어서,

상기 모델 파라미터들은, 상기 입력 데이터 및 상기 입력 데이터의 상기 압축된 버전의 비트 사이즈, 상기 업데이트들의 세트의 비트 사이즈, 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡, 상기 업데이트들의 세트를 전송하는 비용 및 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 상기 왜곡의 비율 중 적어도 하나에 기초하여 튜닝되고, 상기 비용은 상기 업데이트들의 세트의 비트 사이즈에 기초하는, 방법.

청구항 28

제 26 항에 있어서,

상기 모델 파라미터들을 튜닝하는 것은,

튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함하는 것이 상기 입력 데이터의 상기 압축된 버전의 비트 사이즈 및 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡 중 적어도 하나에서의 감소를 수반한다는 결정에 기초하여 상기 튜닝된 모델 파라미터들에 상기 하나 이상의 파라미터들을 포함시키는 것을 포함하는, 방법.

청구항 29

제 19 항에 있어서,

상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 결정하는 것은:

상기 신경망 압축 시스템에서 상기 입력 데이터를 프로세싱하는 것;

프로세싱된 상기 입력 데이터에 기초하여 상기 신경망 압축 시스템에 대한 하나 이상의 손실들을 결정하는 것; 및

상기 하나 이상의 손실들에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 것으로서, 튜닝된 상기 모델 파라미터들은 상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 포함하는, 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 것을 포함하고,

상기 하나 이상의 손실들은 상기 제 1 비트스트림의 사이즈에 기초하여 상기 입력 데이터의 상기 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 상기 입력 데이터와 상기 입력 데이터의 상기 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 상기 제 2 비트스트림의 사이즈에 기초하여 상기 업데이트된 모델 파라미터들의 상기 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함하는, 방법.

청구항 30

명령들이 저장된 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령들은, 하나 이상의 프로세서들에 의해 실행될 때, 상기 하나 이상의 프로세서들로 하여금:

신경망 압축 시스템에 의해, 상기 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하게 하고;

상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것으로서, 상기 업데이트들의 세트는 상기 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함하는, 상기 신경망 압축 시스템에 대한 상기 업데이트들의 세트를 결정하는 것을 행하게 하며;

상기 신경망 압축 시스템에 의해 잠재 프리어(latent prior)를 사용하여, 상기 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하게 하고;

상기 신경망 압축 시스템에 의해 상기 잠재 프리어 및 모델 프리어(model prior)를 사용하여, 상기 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하게 하고; 그리고

수신기로의 전송을 위해 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 출력하게 하는, 비일시적 컴퓨터 판독가능 저장 매체.

발명의 설명

기 술 분 야

[0001] **기술 분야**

[0002] 본 개시는 일반적으로 데이터 압축에 관한 것으로, 보다 구체적으로는, 이미지 및/또는 비디오 콘텐츠를 압축하기 위해 머신 러닝 시스템들을 사용하는 것에 관한 것이다.

배 경 기 술

[0003] **배경**

[0004] 많은 디바이스들 및 시스템들은 이미지/비디오 데이터가 소비를 위해 프로세싱되고 출력되도록 한다. 디지털 이미지/비디오 데이터는 이미지/비디오 품질, 성능 및 기능에 대한 증가하는 요구를 충족시키기 위해 대량의 데이터를 포함한다. 예를 들어, 비디오 데이터의 소비자들은 전형적으로 높은 충실도, 해상도들, 프레임 레이트들 등을 갖는 고품질 비디오들을 원한다. 이러한 요구들을 충족시키기 위해 종종 필요한 많은 양의 비디오 데이터는 비디오 데이터를 프로세싱하고 저장하는 통신 네트워크들 및 디바이스들에 상당한 부담을 준다.

비디오 코딩 기법들은 비디오 데이터를 압축하기 위해 사용될 수도 있다. 비디오 코딩의 하나의 예시적

인 목표는 비디오 품질에서의 열화들을 회피하거나 최소화하면서, 비디오 데이터를 더 낮은 비트 레이트를 사용하는 형태로 압축하는 것이다. 끊임없이 진화하는 비디오 서비스들이 가능해지고 대량의 비디오 데이터에 대한 요구들이 증가함에 따라, 더 나은 성능 및 효율성을 갖는 코딩 기술이 필요하다.

발명의 내용

[0005] **개요**

[0006] 일부 예들에서, 하나 이상의 머신 러닝 시스템들을 사용하여 데이터 압축(compression) 및/또는 압축해제(decompression)를 위한 시스템들 및 기법들이 설명된다. 일부 예들에서, 이미지/비디오 데이터를 압축 및/또는 압축해제하기 위한 머신 러닝(machine learning) 시스템들이 제공된다. 적어도 하나의 예시적인 예에 따르면, 이미지/비디오 데이터를 압축 및/또는 압축해제하는 방법이 제공된다. 일부 예들에서, 방법은, 신경망 압축 시스템(neural network compression system)에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하는 단계; 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 단계 - 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함함 -; 잠재 프리어(latent prior)를 사용하여 신경망 압축 시스템에 의해, 입력 데이터의 압축된 버전을 포함하는 제1 비트스트림을 생성하는 단계; 잠재 프리어 및 모델 프리어(model prior)를 사용하여 신경망 압축 시스템에 의해, 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하는 단계; 및 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하는 단계를 포함할 수 있다.

[0007] 적어도 하나의 예시적인 예에 따르면, 이미지/비디오 데이터를 압축 및/또는 압축해제하기 위한 비일시적 컴퓨터 판독가능 매체가 제공된다. 일부 양태들에서, 비일시적 컴퓨터 판독가능 매체는, 하나 이상의 프로세서들에 의해 실행될 때, 하나 이상의 프로세서들로 하여금, 신경망 압축 시스템에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하게 하고; 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하게 하고 - 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함함 -; 잠재 프리어의 모델을 사용하여 신경망 압축 시스템에 의해, 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하게 하고; 잠재 프리어 및 모델 프리어를 사용하여 신경망 압축 시스템에 의해, 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하게 하고; 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하게 하는 명령들을 포함할 수 있다.

[0008] 적어도 하나의 예시적인 예에 따르면, 이미지/비디오 데이터를 압축 및/또는 압축해제하기 위한 장치가 제공된다. 일부 양태들에서, 장치는, 신경망 압축 시스템에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하고; 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하고 - 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함함 -; 잠재 프리어를 사용하여 신경망 압축 시스템에 의해, 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하고; 잠재 프리어 및 모델 프리어를 사용하여 신경망 압축 시스템에 의해, 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하고; 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하도록 구성된 하나 이상의 프로세서들 및 컴퓨터 판독가능 명령들을 저장하고 있는 메모리를 포함할 수 있다.

[0009] 다른 예시적인 예에 따르면, 이미지/비디오 데이터를 압축 및/또는 압축해제하기 위한 장치는, 신경망 압축 시스템에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하기 위한 수단; 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하기 위한 수단으로서, 그 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함하는, 상기 업데이트들의 세트를 결정하기 위한 수단; 잠재적 프리어를 사용하여 신경망 압축 시스템에 의해, 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하기 위한 수단; 잠재적 프리어 및 모델 프리어를 사용하여 신경망 압축 시스템에 의해, 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하기 위한 수단; 및 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하기 위한 수단을 포함할 수 있다.

[0010] 일부 양태들에서, 상기 설명된 방법, 장치들, 및 컴퓨터 판독가능 매체는 제 1 비트스트림 및 제 2 비트스트림을 포함하는 연접된 비트스트림(concatenated bitstream)을 생성하고, 그 연접된 비트스트림을 수신기로 전송할 수 있다.

[0011] 일부 예들에서, 제 2 비트스트림은 잠재 프리어의 압축된 버전 및 모델 프리어의 압축된 버전을 포함한다.

[0012] 일부 경우들에서, 제 2 비트스트림을 생성하는 것은, 신경망 압축 시스템에 의해, 모델 프리어를 사용하여 잠재 프리어를 엔트로피 인코딩하는 것; 및 신경망 압축 시스템에 의해, 모델 프리어를 사용하여 업데이트된 모델 파

라미터들을 엔트로피 인코딩하는 것을 포함할 수 있다.

[0013] 일부 예들에서, 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함한다. 일부 경우들에서, 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝될 수 있다.

[0014] 일부 예들에서, 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함한다. 일부 경우들에서, 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝될 수 있다. 일부 경우들에서, 제 1 비트스트림은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해 생성된다.

[0015] 일부 예들에서, 제 2 비트스트림을 생성하는 것은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해, 입력 데이터를 입력 데이터의 잠재 공간 표현(latent space representation)으로 인코딩하는 것; 및 잠재 프리어를 사용하여 신경망 압축 시스템에 의해, 잠재 공간 표현을 제 1 비트스트림으로 엔트로피 인코딩하는 것을 포함할 수 있다.

[0016] 일부 양태들에서, 위에서 설명된 방법, 장치들, 및 컴퓨터 판독가능 매체는 신경망 압축 시스템을 트레이닝하는 데 사용되는 트레이닝 데이터세트에 기초하여 신경망 압축 시스템의 모델 파라미터들을 생성하고; 입력 데이터를 사용하여 신경망 압축 시스템의 모델 파라미터들을 튜닝하고; 모델 파라미터들과 튜닝된 모델 파라미터들 사이의 차이에 기초하여 업데이트들의 세트를 결정할 수 있다.

[0017] 일부 예들에서, 모델 파라미터들은 입력 데이터, 입력 데이터의 압축된 버전의 비트 사이즈, 업데이트들의 세트의 비트 사이즈, 및 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에 기초하여 튜닝된다.

[0018] 일부 예들에서, 모델 파라미터들은 입력 데이터, 및 업데이트들의 세트를 전송하는 비용과, 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡의 비율에 기초하여 튜닝되고, 비용은 업데이트들의 세트의 비트 사이즈에 기초한다.

[0019] 일부 예들에서, 모델 파라미터들을 튜닝(tuning)하는 것은 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함하는 것이 입력 데이터의 압축된 버전의 비트 사이즈, 및 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡 중 적어도 하나에서의 감소를 수반한다는 결정에 기초하여 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함시키는 것을 포함할 수 있다.

[0020] 일부 예들에서, 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것은 신경망 압축 시스템에서 입력 데이터를 프로세싱하는 것; 프로세싱된 입력 데이터에 기초하여 신경망 압축 시스템에 대한 하나 이상의 손실들을 결정하는 것; 및 하나 이상의 손실들에 기초하여 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 것을 포함할 수 있고, 튜닝된 모델 파라미터들은 신경망 압축 시스템에 대한 업데이트들의 세트를 포함한다.

[0021] 일부 경우들에서, 하나 이상의 손실들은 제 1 비트스트림의 사이즈에 기초하여 입력 데이터의 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 제 2 비트스트림의 사이즈에 기초하여 업데이트된 모델 파라미터들의 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함한다.

[0022] 일부 예들에서, 수신기는 인코더를 포함한다. 일부 양태들에서, 상기 설명된 방법, 장치들, 및 컴퓨터 판독가능 매체는 인코더에 의해, 제 1 비트스트림 및 제 2 비트스트림을 포함하는 데이터를 수신하고; 디코더에 의해, 제 2 비트스트림에 기초하여 업데이트된 모델 파라미터들의 압축된 버전을 디코딩하고; 디코더에 의해, 업데이트된 파라미터들의 세트를 사용하여, 제 1 비트스트림에서의 입력 데이터의 압축된 버전에 기초하여 입력 데이터의 재구성된 버전을 생성할 수 있다.

[0023] 일부 양태들에서, 상기 설명된 방법, 장치들, 및 컴퓨터 판독가능 매체는 레이트-왜곡 및 모델-레이트 손실을 감소시킴으로써 신경망 압축 시스템을 트레이닝시킬 수 있고, 여기서 모델-레이트는 모델 업데이트들을 전송하기 위한 비트스트림의 길이를 반영한다.

[0024] 일부 예들에서, 모델 프리어는 독립 가우시안 네트워크 프리어(independent Gaussian network prior), 독립 라플라스 네트워크 프리어(independent Laplace network prior), 및/또는 독립 스파이크 앤 슬래브 네트워크 프리어(independent Spike and Slab network prior)를 포함한다.

[0025] 일부 양태들에서, 장치는 카메라 (예를 들어, IP 카메라), 모바일 디바이스 (예를 들어, 모바일 전화 또는 소위

"스마트폰", 또는 다른 모바일 디바이스), 스마트 웨어러블 디바이스, 확장 현실 디바이스 (예를 들어, 가상 현실 (VR) 디바이스, 증강 현실 (AR) 디바이스, 또는 혼합 현실 (MR) 디바이스), 퍼스널 컴퓨터, 랩톱 컴퓨터, 서버 컴퓨터, 3D 스캐너, 멀티-카메라 시스템, 또는 다른 디바이스일 수 있거나, 또는 그 일부일 수 있다. 일부 양태들에서, 장치는 하나 이상의 이미지들을 캡처하기 위한 카메라 또는 다중의 카메라들을 포함한다. 일부 양태들에서, 장치는 하나 이상의 이미지들, 통지들, 및/또는 다른 디스플레이가능 데이터를 디스플레이하기 위한 디스플레이를 더 포함한다. 일부 양태들에서, 상기 설명된 장치들은 하나 이상의 센서들을 포함할 수 있다.

[0026] 이 개요는 청구된 주제의 핵심적인 또는 본질적인 특징들을 식별하도록 의도되지 않았고, 청구된 주제의 범위를 결정하는데 단독으로 사용되도록 의도되지도 않았다. 그 주제는 이 특허의 전체 명세서, 임의의 또는 모든 도면들, 및 각 청구항의 적절한 부분들을 참조하여 이해되어야 한다.

[0027] 전술한 내용은, 다른 특징들 및 실시양태들과 함께, 다음의 명세서, 청구항들 및 첨부 도면들을 참조하면 더욱 명백해질 것이다.

도면의 간단한 설명

[0028] **도면들의 간단한 설명**

본 출원의 예시적인 실시양태들이 이하 도면들을 참조하여 상세히 설명된다.

도 1은 본 개시의 일부 예들에 따른 이미지 프로세싱 시스템의 예를 나타내는 다이어그램이다.

도 2a는 본 개시의 일부 예들에 따른, 완전-연결 신경망의 예를 나타내는 다이어그램이다.

도 2b는 본 개시의 일부 예들에 따른, 국부적으로 연결된 신경망의 예를 나타내는 다이어그램이다.

도 2c는 본 개시의 일부 예들에 따른, 콘볼루션 신경망의 예를 나타내는 다이어그램이다.

도 2d 는 본 개시의 일부 예들에 따른, 이미지로부터 시각적 피처들을 인식하기 위한 딥 콘볼루션 네트워크 (DCN) 의 예를 나타내는 다이어그램이다.

도 3은 본 개시의 일부 예들에 따른, 예시적인 딥 콘볼루션 네트워크(DCN)를 예시하는 블록도이다.

도 4 는 본 개시의 일부 예들에 따른, 비디오 콘텐츠를 압축하기 위한 송신 디바이스 및 수신된 비트스트림을 비디오 콘텐츠로 압축해제하기 위한 수신 디바이스를 포함하는 시스템의 일 예를 나타내는 다이어그램이다.

도 5a 및 도 5b 는 본 개시의 일부 예들에 따른, 예시적인 레이트-왜곡 오토인코더 시스템들을 나타내는 다이어그램들이다.

도 6은 본 개시의 일부 예들에 따른, 인스턴스 적응적 데이터 압축을 위한 예시적인 신경망 압축 시스템을 나타내는 다이어그램이다.

도 7은 본 개시의 일부 예들에 따른, 모델 프리어를 사용하여 미세 튜닝된 (예를 들어, 인스턴스 적응된) 신경망 압축 시스템의 예시적인 아키텍처를 나타내는 다이어그램이다.

도 8은 본 개시의 일부 예들에 따른, 모델 프리어를 사용하여 미세 튜닝된 예시적인 신경망 압축 시스템에 의해 구현되는 예시적인 추론 프로세스를 나타내는 다이어그램이다.

도 9는 본 개시의 일부 예들에 따른, 모델 프리어를 사용하여 미세 튜닝된 예시적인 신경망 압축 시스템에 의해 수행되는 인코딩 및 디코딩 태스크들을 나타내는 다이어그램이다.

도 10은 본 개시의 일부 예들에 따른, 수신기로 송신될 데이터포인트 상에서 미세-튜닝되는 예시적인 레이트-왜곡 오토인코더 모델 및 수신기로 송신될 데이터포인트 상에서 미세-튜닝되지 않는 레이트-왜곡 오토인코더 모델의 예시적인 레이트-왜곡들을 나타내는 그래프이다.

도 11은 본 개시의 일부 예들에 따른, 압축되는 입력 데이터에 적응된 (예를 들어, 미세-튜닝된) 신경망 압축 시스템을 사용하는 인스턴스-적응적 압축에 대한 예시적인 프로세스(1100)를 나타내는 플로우차트이다.

도 12는 본 개시의 일부 예들에 따른, 하나 이상의 이미지들을 압축하기 위한 프로세스의 예를 나타내는 플로우차트이다.

도 13은 본 개시의 일부 예들에 따른, 하나 이상의 이미지들을 압축해제하기 위한 프로세스의 예를 나타내는 플

로우차트이다.

도 14 는 본 개시의 일부 예들에 따른, 예시적인 컴퓨팅 시스템을 나타낸다.

발명을 실시하기 위한 구체적인 내용

[0029] **상세한 설명**

[0030] 이 개시의 일부 양태들 및 실시양태들이 아래 제공된다. 이들 양태들 및 실시양태들의 일부가 독립적으로 적용될 수도 있고 그것들 중 일부는 본 기술분야의 숙련된 자들에게 명확하게 될 바와 같이 조합하여 적용될 수도 있다. 다음의 설명에 있어서, 설명의 목적들로, 특정 상세들이 본 출원의 실시양태들의 철저한 이해를 제공하기 위해 기술된다. 하지만, 여러 실시양태들은 이들 특정 상세들 없이 실시될 수도 있음이 명백할 것이다. 도면 및 설명은 제한하려는 것이 아니다.

[0031] 다음의 설명은 오직 예시적인 실시양태들을 제공할 뿐이고, 본 개시의 범위, 적용가능성, 또는 구성을 한정하도록 의도되지 않는다. 오히려, 예시적인 실시양태들의 설명은 예시적인 실시양태를 구현하기 위한 가능한 설명을 당업자에게 제공할 것이다. 첨부된 청구범위에 설명된 바와 같이 본 출원의 사상 및 범위를 벗어나지 않으면서 엘리먼트들의 기능 및 배열에 다양한 변경들이 이루어질 수도 있음이 이해되어야 한다.

[0032] 상기 언급된 바와 같이, 디지털 이미지 및 비디오 데이터는 특히 고품질 비디오 데이터에 대한 요구가 계속 증가함에 따라 많은 양의 데이터를 포함할 수 있다. 예를 들어, 이미지 및 비디오 데이터의 소비자들은 전형적으로, 높은 충실도, 해상도, 프레임 레이트들 등과 같은 점점 더 높은 비디오 품질을 원한다. 그러나, 이러한 요구들을 충족시키기 위해 필요한 많은 양의 데이터는 높은 대역폭 및 네트워크 자원 요건들과 같이 통신 네트워크들, 및 비디오 데이터를 프로세싱하고 저장하는 디바이스들에 상당한 부담을 줄 수 있다. 따라서, 이미지 및 비디오 데이터의 저장 및/또는 전송에 필요한 데이터의 양을 감소시키기 위한 압축 알고리즘들(코딩 알고리즘들 또는 툴들이라고도 지칭됨)이 유리하다.

[0033] 이미지 데이터 및 비디오 데이터를 압축하기 위해 다양한 기술이 사용될 수 있다. 이미지 데이터의 압축은 JPEG(Joint Photographic Experts Group), BPG(Better Portable Graphics) 등과 같은 알고리즘을 사용하여 달성되었다. 최근, 신경망 기반의 압축 방법들은 이미지 데이터를 압축하는데 상당한 가능성을 보여주었다.

비디오 코딩은 특정 비디오 코딩 표준에 따라 수행될 수 있다. 예시적인 비디오 코딩 표준들은 고효율 비디오 코딩 (HEVC), 필수 비디오 코딩 (EVC), 어드밴스드 비디오 코딩 (AVC), 동영상 전문가 그룹 (MPEG) 코딩, 및 다기능 비디오 코딩 (VVC) 을 포함한다. 그러나, 이러한 종래의 이미지 및 비디오 코딩 기법들은 디코딩이 수행된 후에 재구성된 이미지에서 아티팩트들을 초래할 수 있다.

[0034] 일부 양태들에서, 하나 이상의 머신 러닝 시스템들을 사용하여 데이터 (예를 들어, 이미지, 비디오, 오디오 등) 압축 및 압축해제 (또한, 인코딩 및 디코딩으로서 지칭되고, 집합적으로 코딩으로서 지칭됨) 를 수행하기 위한 시스템들, 장치들, 프로세스들 (또한 방법들로서 지칭됨), 및 컴퓨터 판독가능 매체들 (집합적으로 "시스템들 및 기법들" 로서 본 명세서에서 지칭됨) 이 본 명세서에서 설명된다. 하나 이상의 머신 러닝 시스템은 본 명세서에 설명된 바와 같이 트레이닝될 수 있고, 이미지, 비디오 및/또는 오디오 압축 및 압축해제와 같은 데이터 압축 및/또는 압축해제를 수행하는 데 사용될 수 있다. 본 명세서에 설명된 머신 러닝 시스템들은 고품질 데이터 출력들을 생성하는 트레이닝 및 압축/압축해제 기술들을 수행할 수 있다.

[0035] 본 명세서에 설명된 시스템들 및 기법들은 임의의 타입의 데이터의 압축 및/또는 압축해제를 수행할 수 있다. 예를 들어, 일부 경우들에서, 본 명세서에 설명된 시스템들 및 기법들은 이미지 데이터의 압축 및/또는 압축해제를 수행할 수 있다. 다른 예로서, 일부 경우들에서, 본 명세서에 설명된 시스템들 및 기법들은 비디오 데이터의 압축 및/또는 압축해제를 수행할 수 있다. 다른 예로서, 일부 경우들에서, 본 명세서에 설명된 시스템들 및 기법들은 오디오 데이터의 압축 및/또는 압축해제를 수행할 수 있다. 간략화, 예시 및 설명의 목적들을 위해, 본 명세서에 설명된 시스템들 및 기법들은 이미지 데이터(예를 들어, 이미지들, 비디오들 등)의 압축 및/또는 압축해제를 참조하여 논의된다. 그러나, 위에서 언급된 바와 같이, 본 명세서에 설명된 개념들은 또한 오디오 데이터 및 임의의 다른 타입의 데이터와 같은 다른 양식들에 적용될 수 있다.

[0036] 일부 예들에서, 데이터 압축 및/또는 압축해제를 위한 머신 러닝 시스템은 트레이닝 데이터의 세트(예를 들어, 이미지들, 비디오, 오디오 등)에 대해 트레이닝될 수 있고, 수신기로 전송되고 디코딩될 데이터에 대해 추가로 미세 튜닝(예를 들어, 트레이닝, 피팅)될 수 있다. 일부 경우들에서, 머신 러닝 시스템의 인코더는 디코더에 전송되고 디코더에 의해 디코딩될 데이터를 사용하여 미세 튜닝되는 압축 모델의 업데이트된 파라미터들을

디코더에 전송할 수 있다. 일부 예들에서, 인코더는 디코더로 전송된 데이터의 양 및/또는 비트레이트를 감소시키기 위해 다른 모델 파라미터들 없이 (및/또는 모델 파라미터들의 전체 세트를 전송하는 대신에) 업데이트된 모델 파라미터들을 전송할 수 있다. 일부 경우들에서, 업데이트된 모델 파라미터들은 디코더로 전송된 데이터의 양 및/또는 비트레이트를 감소시키기 전에 모델을 사용하여 양자화되고 압축될 수 있다.

[0037] 인코더 및/또는 디코더에 의해 사용되는 압축 모델은 상이한 타입들의 데이터로 일반화될 수 있다. 또한, 압축 모델을 전송 및 디코딩되는 데이터로 미세 튜닝함으로써, 머신 러닝 시스템은 그 특정 데이터에 대한 압축 및/또는 압축해제 성능, 품질 및/또는 효율을 증가시킬 수 있다. 일부 경우들에서, 머신 러닝 시스템의 모델은, 업데이트된 모델 파라미터들을 전송할 때 여분의 오버헤드 및 비트레이트들을 반영하고 그리고/또는 고려하는 레이트 및 왜곡 손실들 및 추가적인 레이트 손실로 트레이닝될 수 있다. 모델은 레이트 (예를 들어, 비트스트림의 사이즈/길이), 왜곡 (예를 들어, 입력과 재구성된 출력 사이의 왜곡), 및 모델-레이트 손실 (예를 들어, 업데이트된 모델 파라미터들을 전송하는 비용을 반영하는 손실) 을 최소화하도록 트레이닝될 수 있다.
일부 예들에서, 머신 러닝 시스템은 레이트, 왜곡 및 모델 레이트 (예를 들어, 업데이트된 모델 파라미터들을 전송하는데 필요한 비트스트림의 사이즈/길이) 를 트레이드 오프할 수 있다.

[0038] 일부 예들에서, 머신 러닝 시스템은 하나 이상의 신경망을 포함할 수 있다. 머신 러닝(machine learning; ML)은 인공 지능(artificial intelligence; AI)의 서브세트이다. ML 시스템들은 컴퓨터 시스템들이 명시적 명령어들의 사용 없이 패턴들 및 추론에 의존함으로써 다양한 작업들을 수행하는 데 사용할 수 있는 알고리즘들 및 통계 모델들을 포함한다. ML 시스템의 일 예는 인공 뉴런들(예를 들어, 뉴런 모델들)의 상호연결된 그룹으로 구성될 수도 있는 신경망(인공 신경망으로도 지칭됨)이다. 신경망들은 특히 이미지 분석 및/또는 컴퓨터 비전 애플리케이션들, 인터넷 프로토콜(IP) 카메라들, 사물 인터넷(IoT) 디바이스들, 자율 차량들, 서비스 로봇들과 같은 다양한 애플리케이션들 및/또는 디바이스들을 위해 사용될 수도 있다.

[0039] 신경망 내의 개별 노드들은 입력 데이터를 취하고 데이터에 대해 간단한 연산들을 수행함으로써 생물학적 뉴런들을 에뮬레이트(emulate)할 수도 있다. 입력 데이터에 대해 수행된 단순한 연산들의 결과들은 다른 뉴런들에 선택적으로 전달된다. 가중치 값들은 네트워크 내의 각각의 벡터 및 노드와 연관되고, 이들 값들은 입력 데이터가 출력 데이터와 어떻게 관련되는지를 제약한다. 예를 들어, 각 노드의 입력 데이터는 대응하는 가중치 값과 곱해질 수도 있고, 그 곱들은 합산될 수도 있다. 곱들의 합은 선택적인 바이어스에 의해 조정될 수도 있고, 활성화 함수가 결과에 적용되어, 노드의 출력 신호 또는 "출력 활성화"(때때로 활성화 맵 또는 피처 맵으로 지칭됨)를 산출할 수 있다. 가중치 값들은 초기에 네트워크를 통한 트레이닝 데이터의 반복 흐름에 의해 결정될 수 있다(예를 들어, 가중치 값들은 네트워크가 그들의 전형적인 입력 데이터 특성들에 의해 특정 클래스들을 식별하는 방법을 학습하는 트레이닝 단계 동안 설정된다).

[0040] 심층 생성 신경망 모델(예를 들어, 생성 적대 신경망(generative adversarial network; GAN)), 순환 신경망(recurrent neural network; RNN) 모델, 다층 퍼셉트론(multilayer perceptron; MLP) 신경망 모델, 콘볼루션 신경망(convolutional neural network; CNN) 모델, 오토인코더(autoencoder; AE)와 같은 상이한 타입들의 신경망이 존재한다. 예를 들어, GAN은 입력 데이터에서 패턴들을 학습할 수 있는 생성 신경망의 형태이며, 따라서 신경망 모델은 원래 데이터세트로부터 합리적으로 얻어질 수 있었을 새로운 합성 출력들을 생성할 수 있다.
GAN은 함께 동작하는 2개의 신경망을 포함할 수 있다. 신경망들 중 하나(G(z)로 표시된 생성 신경망 또는 생성기로 지칭됨)는 합성된 출력을 생성하고, 다른 신경망(D(X)로 표시된 판별 신경망 또는 판별기로 지칭됨)은 진본성에 대해 출력을 평가한다(출력이 트레이닝 데이터세트와 같은 원래 데이터세트로부터 온 것인지 또는 생성기에 의해 생성되었는지 여부). 트레이닝 입력 및 출력은 예시적인 예로서 이미지들을 포함할 수 있다. 생성기는 생성기에 의해 생성된 합성 이미지가 데이터세트로부터의 실제 이미지임을 결정하는 것으로 판별기를 속이려고 시도하도록 트레이닝된다. 트레이닝 프로세스가 계속되고 생성기는 실제 이미지들처럼 보이는 합성 이미지들을 생성하는 데 더 잘 된다. 판별기는 합성된 이미지들에서 결함들을 계속 찾고, 생성기는 이미지들에서의 결함들을 결정하기 위해 판별기가 무엇을 보고 있는지를 파악한다. 일단 네트워크가 트레이닝되면, 생성기는 판별기가 실제 이미지들과 구별할 수 없는 현실적인 외관의 이미지들을 생성할 수 있다.

[0041] RNN들은 계층의 결과를 예측하는 것을 돕기 위해 계층의 출력을 저장하고 이 출력을 입력에 다시 공급하는 원리에 대해 작동한다. MLP 신경망들에서, 데이터는 입력 계층에 공급될 수도 있고, 하나 이상의 은닉 계층들은 데이터에 대한 추상화의 레벨들을 제공한다. 그런 다음, 추상화된 데이터에 기초하여 출력 계층 상에서 예측들이 이루어질 수도 있다. MLP들은 입력들이 클래스 또는 라벨이 할당되는 분류 예측 문제들에 특히 적합할 수도 있다. 콘볼루션 신경망(CNN)은 피드-포워드 인공 신경망의 일종이다. CNN들은 수용 필드

(receptive field)(예를 들어, 입력 공간의 공간적으로 국부화된 영역)를 각각 갖고 입력 공간을 집합적으로 타일링하는 인공 뉴런들의 집합들을 포함할 수도 있다. CNN은 패턴 인식 및 분류를 포함하여 수많은 애플리케이션들을 가지고 있다.

[0042] 계층화된 신경망 아키텍처들(다수의 은닉 층들이 존재할 때 심층 신경망들로 지칭됨)에서, 인공 뉴런들의 제 1 층의 출력은 인공 뉴런들의 제 2 층에 대한 입력이 되고, 인공 뉴런들의 제 2 층의 출력은 인공 뉴런들의 제 3 층에 대한 입력이 되는 등등이다. 콘볼루션 신경망들은 피처들의 계층을 인식하도록 트레이닝될 수도 있다. 콘볼루션 신경망 아키텍처들에서의 계산은 하나 이상의 계산 체인으로 구성될 수도 있는 프로세싱 노드들의 집단에 걸쳐 분산될 수도 있다. 이들 다중-계층화된 아키텍처들은 한번에 하나의 계층씩 트레이닝될 수도 있고, 역 전파 (back propagation) 를 이용하여 미세-튜닝될 수도 있다.

[0043] 오토인코더(AE)는 비지도 방식(unsupervised manner)으로 효율적인 데이터 코딩들을 학습할 수 있다. 일부 예들에서, AE 는 신호 노이즈를 무시하도록 네트워크를 트레이닝함으로써 데이터의 세트에 대한 표현 (예를 들어, 데이터 코딩) 을 학습할 수 있다. AE는 인코더 및 디코더를 포함할 수 있다. 인코더는 입력 데이터를 코드로 맵핑할 수 있고 디코더는 코드를 입력 데이터의 재구성에 맵핑할 수 있다. 일부 예들에서, 레이트-왜곡 오토인코더(RD-AE)는 이미지 및/또는 비디오 데이터포인트들과 같은 데이터포인트들의 데이터세트에 걸쳐 평균 레이트-왜곡 손실을 최소화하도록 트레이닝될 수 있다. 일부 경우들에서, RD-AE는 새로운 데이터포인트를 인코딩하기 위해 추론 시간에 순방향 패스를 수행할 수 있다.

[0044] 일부 경우들에서, RD-AE는 디코더와 같은 수신기에 송신될 데이터에 대해 미세-튜닝될 수 있다. 일부 예들에서, 데이터포인트 상에서 RD-AE 를 미세-튜닝함으로써, RD-AE 는 높은 압축 (예를 들어, 레이트/왜곡) 성능을 획득할 수 있다. RD-AE와 연관된 인코더는 RD-AE 모델 또는 RD-AE 모델의 일부를 수신기(예를 들어, 디코더)에 전송하여, 수신기가 인코더에 의해 전송된 압축된 데이터를 포함하는 비트스트림을 디코딩하도록 할 수 있다.

[0045] 일부 경우들에서, AE 모델들은 클 수 있으며, 이는 비트레이트를 증가시키고/시키거나 레이트-왜곡 이득들을 감소시킬 수 있다. 일부 예들에서, RD-AE 모델은 모델 프리어(예를 들어, RDM-AE 프리어)를 사용하여 미세 튜닝될 수 있다. 모델 프리어는 수신기(예를 들어, 디코더)가 송신된 데이터를 압축해제하기 위해 모델을 구현하기 위해 사용할 모델 업데이트들을 생성하기 위해 정의되고 사용될 수 있다. 모델 프리어는 모델 프리어 하에서 생성된 모델 업데이트들을 통해 디코더로 전송되는 데이터의 양을 감소시킬 수 있다. 일부 예들에서, 모델 프리어는 모델 업데이트들을 송신하기 위한 비용을 감소시키도록 설계될 수 있다. 예를 들어, 모델 프리어는 모델 업데이트들의 비트레이트 오버헤드를 감소 및/또는 제한하고 및/또는 더 작은 모델 업데이트들을 생성하는 데 사용될 수 있다. 일부 경우들에서, 더 많은 파라미터들이 미세-튜닝됨에 따라, 모델 업데이트들의 비트레이트 오버헤드가 증가할 수 있다. 일부 예들에서, 미세-튜닝되는 파라미터들은 감소될 수 있으며, 이는 또한 비트레이트 오버헤드를 감소시키거나 제한할 수 있다.

[0046] 일부 경우들에서, 모델은 RDM-AE 손실들을 사용하여 미세 튜닝될 수 있다. 손실 항은 모델-레이트에 대한 비트레이트에 추가될 수 있다. 추가된 손실 항은 모델 업데이트들에서 "소비된" 비트를 보상할 수 있다. 예를 들어, 미세-튜닝 동안, 모델 업데이트들에 대해 "소비된" 임의의 비트는 레이트-왜곡(rate-distortion; R/D)의 개선에 의해 보상될 수 있다. 일부 예들에서, 미세-튜닝 동안, 모델 업데이트들에 대해 "소비된" 임의의 비트는 적어도 많은 레이트-왜곡의 개선에 의해 보상될 수 있다.

[0047] 일부 경우들에서, 모델 프리어의 설계는 본 명세서에 추가로 설명되는 바와 같이 개선될 수 있다. 일부 예시적인 예들에서, 모델 프리어 설계는 독립 가우시안 모델 프리어(independent Gaussian model prior)를 포함할 수 있다. 다른 예시적인 예들에서, 모델 프리어 설계는 독립 라플라스 모델 프리어(independent Laplace model prior)를 포함할 수 있다. 다른 예시적인 예들에서, 모델 프리어 설계는 독립 스파이크 앤 슬래브 프리어(independent Spike and Slab prior)를 포함할 수 있다. 일부 예시적인 예들에서, 모델 프리어 및 글로벌 AE 모델은 공동으로 트레이닝될 수 있다. 일부 예시적인 예들에서, 모델 프리어는 신경망에 의해 학습되는 복잡한 종속성들을 포함할 수 있다.

[0048] 도 1은 본 개시의 일부 예들에 따른, 이미지 프로세싱 시스템(100)의 예를 나타내는 다이어그램이다. 일부 경우들에서, 이미지 프로세싱 시스템(100)은 본 명세서에 설명된 기능들 중 하나 이상을 수행하도록 구성된 중앙 프로세싱 유닛(CPU)(102) 또는 멀티-코어 CPU를 포함할 수 있다. 다른 정보 중에서도, 변수들(예를 들어, 신경 신호들 및 시냅스 가중치들), 계산 디바이스와 연관된 시스템 파라미터들(예를 들어, 가중치들을 갖는 신경망), 지연들, 주파수 빈 정보, 태스크 정보는 신경 프로세싱 유닛(NPU)(108)과 연관된 메모리 블록에,

CPU(102)와 연관된 메모리 블록에, 그래픽 프로세싱 유닛(GPU)(104)과 연관된 메모리 블록에, 디지털 신호 프로세서(DSP)(106)와 연관된 메모리 블록에, 메모리 블록(118)에 저장될 수도 있거나, 또는 다수의 블록들에 걸쳐 분산될 수도 있다. CPU(102)에서 실행되는 명령들은 CPU(102)와 연관된 프로그램 메모리 및/또는 메모리 블록(118)으로부터 로딩될 수도 있다.

[0049] 이미지 프로세싱 시스템(100)은 GPU(104); DSP(106); 5세대(5G) 연결, 4세대 롱 텀 에볼루션(4G LTE) 연결, Wi-Fi 연결, USB 연결, 블루투스 연결 등을 포함할 수도 있는 연결 블록(110); 및/또는 예를 들어 피처들을 검출 및 인식할 수도 있는 멀티미디어 프로세서(112)와 같은 특정 기능들에 맞춤화된 추가적인 프로세싱 블록들을 포함할 수도 있다. 일 구현에서, NPU(108)는 CPU(102), DSP(106) 및/또는 GPU(104)에서 구현된다. 이미지 프로세싱 시스템(100)은 또한 센서 프로세서(114), 하나 이상의 이미지 신호 프로세서(ISP)들(116) 및/또는 스토리지(120)를 포함할 수도 있다. 일부 예들에서, 이미지 프로세싱 시스템(100)은 ARM 명령어 세트에 기초할 수도 있다.

[0050] 이미지 프로세싱 시스템(100)은 컴퓨팅 디바이스 또는 다수의 컴퓨팅 디바이스의 일부일 수 있다. 일부 예들에서, 이미지 프로세싱 시스템(100)은 카메라 시스템(예를 들어, 디지털 카메라, IP 카메라, 비디오 카메라, 보안 카메라 등), 전화 시스템(예를 들어, 스마트폰, 셀룰러 전화, 회의 시스템 등), 데스크톱 컴퓨터, XR 디바이스(예를 들어, 헤드 마운티드 디스플레이 등), 스마트 웨어러블 디바이스(예를 들어, 스마트 워치, 스마트 안경 등), 랩톱 또는 노트북 컴퓨터, 태블릿 컴퓨터, 셋톱 박스, 텔레비전, 디스플레이 디바이스, 디지털 미디어 플레이어, 게이밍 콘솔, 비디오 스트리밍 디바이스, 드론, 자동차 내의 컴퓨터, 시스템-온-칩(SOC), 사물 인터넷(IoT) 디바이스, 또는 임의의 다른 적절한 전자 디바이스(들)와 같은 전자 디바이스(또는 디바이스들)의 일부일 수 있다.

[0051] 이미지 프로세싱 시스템(100)이 특정 컴포넌트들을 포함하는 것으로 도시되지만, 당업자는 이미지 프로세싱 시스템(100)이 도 1에 도시된 것들보다 더 많거나 더 적은 컴포넌트들을 포함할 수 있다는 것을 인식할 것이다. 예를 들어, 확장 현실 시스템 (100) 은 또한, 일부 사례들에서, 하나 이상의 메모리 디바이스들 (예컨대, RAM, ROM, 캐시 등), 하나 이상의 네트워킹 인터페이스들 (예컨대, 유선 및/또는 무선 통신 인터페이스들 등), 하나 이상의 디스플레이 디바이스들, 및/또는 도 1 에 도시되지 않은 다른 하드웨어 또는 프로세싱 디바이스들을 포함할 수 있다. 이미지 프로세싱 시스템(100)으로 구현될 수 있는 컴퓨팅 디바이스 및 하드웨어 컴포넌트들의 예시적인 예가 도 14와 관련하여 아래에서 설명된다.

[0052] 이미지 프로세싱 시스템(100) 및/또는 그 컴포넌트들은 본 명세서에 설명된 머신 러닝 시스템들 및 기술들을 사용하여 압축 및/또는 압축해제(인코딩 및/또는 디코딩으로도 지칭되고, 집합적으로 이미지 코딩으로도 지칭됨)를 수행하도록 구성될 수 있다. 일부 경우들에서, 이미지 프로세싱 시스템(100) 및/또는 그것의 컴포넌트들은 본 명세서에 설명된 기술들을 사용하여 이미지 또는 비디오 압축 및/또는 압축해제를 수행하도록 구성될 수 있다. 일부 예들에서, 머신 러닝 시스템들은 이미지, 비디오, 및/또는 오디오 데이터의 압축 및/또는 압축해제를 수행하기 위해 딥 러닝 신경망 아키텍처들을 이용할 수 있다. 딥 러닝 신경망 아키텍처들을 사용함으로써, 머신 러닝 시스템들은 디바이스 상의 콘텐츠의 압축 및/또는 압축해제의 효율 및 속도를 증가시킬 수 있다. 예를 들어, 설명된 압축 및/또는 압축해제 기술들을 사용하는 디바이스는 머신 러닝 기반 기술들을 사용하여 효율적으로 하나 이상의 이미지들을 압축할 수 있고, 압축된 하나 이상의 이미지들을 수신 디바이스로 송신할 수 있으며, 수신 디바이스는 본 명세서에 설명된 머신 러닝 기반 기술들을 사용하여 효율적으로 하나 이상의 압축된 이미지들을 압축해제할 수 있다. 본 명세서에서 사용되는 바와 같이, 이미지는 프레임들의 시퀀스(예를 들어, 비디오)와 연관된 정지 이미지 및/또는 비디오 프레임을 지칭할 수 있다.

[0053] 상기 언급된 바와 같이, 신경망은 머신 러닝 시스템의 예이다. 신경망은 입력 계층, 하나 이상의 은닉 계층들, 및 출력 계층을 포함할 수 있다. 데이터는 입력 계층의 입력 노드들로부터 제공되고, 프로세싱은 하나 이상의 은닉 계층들의 은닉 노드들에 의해 수행되고, 출력은 출력 계층의 출력 노드들을 통해 생성된다. 딥 러닝 네트워크들은 통상적으로 다수의 은닉 계층들을 포함한다. 신경망의 각각의 계층은 인공 뉴런들(또는 노드들)을 포함할 수 있는 피처 맵들 또는 활성화 맵들을 포함할 수 있다. 피처 맵은 필터, 커널 등을 포함할 수 있다. 노드들은 계층들 중 하나 이상의 계층들의 노드들의 중요도를 표시하는 데 사용되는 하나 이상의 가중치들을 포함할 수 있다. 일부 경우들에서, 딥 러닝 네트워크는 일련의 많은 은닉 계층들을 가질 수 있으며, 초기 계층들은 입력의 단순하고 낮은 레벨 특성들을 결정하는 데 사용되고, 나중의 계층들은 더 복잡하고 추상적인 특성들의 계층을 구축한다.

[0054] 딥 러닝 아키텍처는 피처들의 계위를 학습할 수도 있다. 예를 들어, 시각적 데이터로 제시되면, 제 1 계층

은 입력 스트림에서, 에지들과 같은 비교적 간단한 피처들을 인식하는 것을 학습할 수도 있다. 다른 예에서, 청각적 데이터로 제시되면, 제 1 계층은 특정 주파수들에서의 스펙트럼 전력을 인식하는 것을 학습할 수도 있다. 제 1 계층의 출력을 입력으로서 취하는 제 2 계층은, 시각 데이터에 대한 간단한 형상들 또는 청각 데이터에 대한 사운드들의 조합들과 같은 피처들의 조합들을 인식하는 것을 학습할 수도 있다. 예를 들어, 상위 계층들은 시각적 데이터에서의 복잡한 형상들 또는 청각적 데이터에서의 단어들을 나타내는 것을 학습할 수도 있다. 여전히 상위 계층들은 공통 시각적 객체들 또는 구어체들을 인식하는 것을 학습할 수도 있다.

[0055] 딥 러닝 아키텍처들은 자연스러운 계위 구조를 갖는 문제들에 적용될 때 특히 잘 수행할 수도 있다. 예를 들어, 모터구동 차량들 (motorized vehicle) 의 분류는 휠들, 윈드쉴드들 및 다른 피처들을 인식하는 것을 먼저 학습하는 것으로 이익을 얻을 수도 있다. 이러한 피처들은 자동차, 트럭, 및 비행기를 인식하기 위해 상이한 방식들로 상위 계층에서 조합될 수도 있다.

[0056] 신경망들은 다양한 접속성 패턴들로 설계될 수도 있다. 피드-포워드 네트워크들에서, 정보는 하위 계층에서 상위 계층으로 전달되고, 주어진 계층에서의 각각의 뉴런은 상위 계층들에서의 뉴런들에 통신한다. 계위적 표현은 상술한 바와 같이, 피드-포워드 네트워크의 연속적인 계층들에 구축될 수도 있다. 신경망들은 또한 순환(recurrent) 또는 피드백(feedback) (또한 하향식이라 함) 연결들을 가질 수도 있다. 순환 연결에서, 주어진 계층의 뉴런으로부터의 출력은 동일한 계층의 다른 뉴런으로 통신될 수도 있다. 순환 아키텍처는 시퀀스로 신경망에 전달되는 입력 데이터 청크들 중 하나보다 많은 청크들에 걸쳐 있는 패턴들을 인식하는데 도움이 될 수도 있다. 주어진 계층의 뉴런에서 하위 계층의 뉴런으로의 연결은 피드백 (또는 하향식) 연결이라고 한다. 많은 피드백 연결들을 갖는 네트워크는 하이-레벨 개념의 인식이 입력의 특정 로우-레벨 피처들을 식별하는 것을 보조할 수도 있을 때 도움이 될 수도 있다.

[0057] 신경망의 계층들 사이의 연결들은 완전히(fully) 연결되거나 국부적으로(locally) 연결될 수도 있다. 도 2a는 완전히 연결된 신경망(202)의 예를 도시한다. 완전히 연결된 신경망 (202) 에서, 제 1 계층에서의 뉴런은 제 2 계층에서의 모든 뉴런에 그의 출력을 통신할 수도 있으므로, 제 2 계층에서의 각각의 뉴런이 제 1 계층에서의 모든 뉴런으로부터 입력을 수신할 것이다. 도 2b는 국부적으로 연결된 신경망(204)의 예를 도시한다. 국부적으로 연결된 신경망 (204) 에서, 제 1 계층에서의 뉴런은 제 2 계층에서의 제한된 수의 뉴런들에 연결될 수도 있다. 보다 일반적으로, 국부적으로 연결된 신경망 (204) 의 국부적으로 연결된 계층은 계층에서의 각각의 뉴런이 동일하거나 유사한 접속성 패턴을 가질 것이지만, 상이한 값들 (예를 들어, 210, 212, 214, 및 216) 을 가질 수도 있는 연결 강도들을 갖도록 구성될 수도 있다. 국부적으로 연결된 접속성 패턴은 상위 계층에서 공간적으로 별개의 수용 필드들을 발생할 수도 있는데, 이는 주어진 영역에서 상위 계층 뉴런들이 네트워크에 대한 총 입력의 제한된 부분의 특성들에 대한 트레이닝을 통해 튜닝되는 입력들을 수신할 수도 있기 때문이다.

[0058] 국부적으로 연결된 신경망의 일 예는 콘볼루션 신경망이다. 도 2c는 콘볼루션 신경망(206)의 예를 예시한다. 콘볼루션 신경망 (206) 은 제 2 계층에서의 각각의 뉴런에 대한 입력들과 연관된 연결 강도들이 공유되도록 (예를 들어, 208) 구성될 수도 있다. 콘볼루션 신경망들은 입력들의 공간적 위치가 의미있는 문제들에 매우 적합할 수도 있다. 콘볼루션 신경망(206)은 본 개시의 양상들에 따라, 비디오 압축 및/또는 압축해제의 하나 이상의 양상들을 수행하는 데 사용될 수도 있다.

[0059] 콘볼루션 신경망의 하나의 타입은 딥 콘볼루션 네트워크 (deep convolutional network; DCN) 이다. 도 2d는 자동차-탑재형 (car-mounted) 카메라와 같은 이미지 캡처 디바이스 (230) 로부터 입력된 이미지 (226) 로부터 시각적 피처들을 인식하도록 설계된 DCN (200) 의 상세한 예를 도시한다. 본 예의 DCN (200) 은 교통 표지판 및 교통 표지판 상에 제공된 번호를 식별하도록 트레이닝될 수도 있다. 물론, DCN (200) 은 차선 마킹들을 식별하거나 신호등들을 식별하는 것과 같은 다른 태스크들을 위해 트레이닝될 수도 있다.

[0060] DCN (200) 은 지도 학습으로 트레이닝될 수도 있다. 트레이닝 동안, DCN (200) 은 속도 제한 표지판의 이미지 (226) 와 같은 이미지로 제시될 수도 있고, 그 후 순방향 패스가 출력 (222) 을 생성하기 위해 계산될 수도 있다. DCN (200) 은 피처 추출 섹션 및 분류 섹션을 포함할 수도 있다. 이미지 (226) 를 수신하면, 콘볼루션 계층 (232) 은 이미지 (226) 에 콘볼루션 커널들 (미도시) 을 적용하여 피처 맵들 (218) 의 제 1 세트를 생성할 수도 있다. 예로서, 콘볼루션 계층 (232) 에 대한 콘볼루션 커널은 28x28 피처 맵들을 생성하는 5x5 커널일 수도 있다. 본 예에서, 4개의 상이한 피처 맵이 피처 맵들의 제 1 세트 (218) 에서 생성되기 때문에, 4개의 상이한 콘볼루션 커널이 콘볼루션 계층 (232) 에서 이미지 (226) 에 적용되었다. 콘볼루션

커널들은 또한 필터들 또는 콘볼루션 필터들로 지칭될 수도 있다.

[0061] 피처 맵들의 제 1 세트 (218) 는 피처 맵들의 제 2 세트 (220) 를 생성하기 위해 최대 풀링 계층 (미도시) 에 의해 서브샘플링될 수도 있다. 최대 풀링 계층(max pooling layer)은 피처 맵들 (218) 의 제 1 세트의 사이즈를 감소시킨다. 즉, 14x14 와 같은 피처 맵들의 제 2 세트 (220) 의 사이즈는 28x28 과 같은 피처 맵들의 제 1 세트 (218) 의 사이즈보다 작다. 감소된 사이즈는 메모리 소비를 감소시키면서 후속 계층에 유사한 정보를 제공한다. 피처 맵들의 제 2 세트 (220) 는 추가로, 피처 맵들의 하나 이상의 후속 세트 (미도시) 를 생성하기 위해 하나 이상의 후속 콘볼루션 계층 (미도시) 을 통해 콘볼루션될 수도 있다.

[0062] 도 2d 의 예에서, 피처 맵들의 제 2 세트 (220) 는 제 1 피처 벡터 (224) 를 생성하도록 콘볼루션된다. 또한, 제 1 피처 벡터 (224) 는 제 2 피처 벡터 (228) 를 생성하도록 추가로 콘볼루션된다. 제 2 피처 벡터 (228) 의 각각의 피처는 "표지판", "60" 및 "100" 과 같은 이미지 (226) 의 가능한 피처에 대응하는 수를 포함할 수도 있다. 소프트맥스 함수 (softmax function)(미도시) 는 제 2 피처 벡터 (228) 에서의 수들을 확률로 변환할 수도 있다. 이와 같이, DCN (200) 의 출력 (222) 은 하나 이상의 피처들을 포함하는 이미지 (226) 의 확률이다.

[0063] 본 예에서, "표지판" 및 "60" 에 대한 출력 (222) 에서의 확률들은 "30", "40", "50", "70", "80", "90" 및 "100" 과 같은 출력 (222) 의 다른 것들의 확률들보다 높다. 트레이닝 전에, DCN (200) 에 의해 생성된 출력 (222) 은 부정확할 가능성이 있다. 따라서, 출력 (222) 과 타겟 출력 사이에 에러가 계산될 수도 있다. 타겟 출력은 이미지 (226) 의 실측 자료(ground truth)(예를 들어, "표지판" 및 "60") 이다. DCN (200) 의 가중치들은 그 후 DCN (200) 의 출력 (222) 이 타겟 출력과 더 밀접하게 정렬되도록 조정될 수도 있다.

[0064] 가중치들를 조정하기 위해, 러닝 알고리즘은 가중치들에 대한 그래디언트 벡터를 계산할 수도 있다. 그래디언트는 가중치가 조정되었으면 에러가 증가 또는 감소할 양을 표시할 수도 있다. 최상위 계층에서, 그래디언트는 끝에서 두번째 계층에서의 활성화된 뉴런 및 출력 계층에서의 뉴런을 연결하는 가중치의 값에 직접 대응할 수도 있다. 하위 계층들에서, 그래디언트는 가중치들의 값 및 상위 계층들의 계산된 에러 그래디언트들에 의존할 수도 있다. 가중치들은 그 후 에러를 감소시키기 위해 조정될 수도 있다. 가중치를 조정하는 이러한 방식은 신경망을 통한 "역방향 패스" 를 수반하기 때문에 "역 전파" 로 지칭될 수도 있다.

[0065] 실제로, 가중치들의 에러 그래디언트는 작은 수의 예들에 걸쳐 계산될 수도 있어서, 계산된 그래디언트는 실제 에러 그래디언트에 근사한다. 이러한 근사화 방법은 확률적 그래디언트 하강법 (stochastic gradient descent) 으로 지칭될 수도 있다. 확률적 그래디언트 하강법은 전체 시스템의 달성가능한 에러율이 감소하는 것을 멈출 때까지 또는 에러율이 목표 레벨에 도달할 때까지 반복될 수도 있다. 학습 후에, DCN 은 새로운 이미지들을 제시받을 수도 있고, 네트워크를 통한 포워드 패스는 DCN 의 추론 또는 예측으로 고려될 수도 있는 출력 (222) 을 산출할 수도 있다.

[0066] DBN (deep belief network) 은 은닉된 노드들의 다중 계층들을 포함하는 확률 모델이다. DBN 은 트레이닝 데이터 세트의 계위적 표현을 추출하는데 사용될 수도 있다. DBN 은 제한된 볼츠만 머신 (Restricted Boltzmann Machines)(RBM) 의 계층들을 적층하여 획득될 수도 있다. RBM 은 입력들의 세트에 걸친 확률 분포를 학습할 수 있는 인공 신경망의 타입이다. RBM들은 각각의 입력이 카테고리화되어야 하는 클래스에 관한 정보의 부재 시 확률 분포를 학습할 수 있기 때문에, RBM들은 종종 비지도 학습에 사용된다. 하이브리드 비지도 및 지도 패러다임을 사용하여, DBN 의 최하위 RBM들은 비지도 방식으로 트레이닝될 수도 있고 피처 추출기들로서 작용할 수도 있으며, 최상위 RBM 은 (이전 계층 및 타겟 클래스들로부터의 입력들의 공동 분포에 대해) 지도 방식으로 트레이닝될 수도 있고 분류기로서 작용할 수도 있다.

[0067] 딥 콘볼루션 네트워크 (DCN) 는 추가적인 풀링 및 정규화 계층들로 구성된, 콘볼루션 네트워크들의 네트워크들이다. DCN들은 많은 태스크들에 대해 최첨단 성능을 달성하였다. DCN들은 입력 및 출력 타겟들 양자 모두가 많은 예시들에 대해 알려져 있고 그래디언트 하강 방법들의 사용에 의해 네트워크의 가중치들을 수정하는데 사용되는 지도 학습을 사용하여 트레이닝될 수 있다.

[0068] DCN 은 피드-포워드 네트워크일 수도 있다. 또한, 상술한 바와 같이, DCN 의 제 1 계층에서의 뉴런으로부터 다음 상위 계층에서의 뉴런들의 그룹으로의 연결들은 제 1 계층에서의 뉴런들에 걸쳐 공유된다. DCN들의 피드-포워드 및 공유 연결들은 빠른 프로세싱을 위해 이용될 수도 있다. DCN 의 계산 부담은 예를 들어, 순환 또는 피드백 연결들을 포함하는 유사하게 사이징된 신경망의 것보다 훨씬 적을 수도 있다.

[0069] 콘볼루션 네트워크의 각각의 계층의 프로세싱은 공간적으로 불변 템플릿 또는 기저 투영으로 간주될 수도 있다.

입력이 컬러 이미지의 적색, 녹색 및 청색 채널들과 같은 다중 채널들로 먼저 분해되면, 그 입력에 대해 트레이닝된 콘볼루션 네트워크는 이미지의 축들을 따라 2개의 공간 차원 및 컬러 정보를 캡처하는 제 3 차원을 갖는, 3 차원으로 간주될 수도 있다. 콘볼루션 연결들의 출력들은 후속 계층에서 피처 맵을 형성하는 것으로 간주될 수도 있고, 피처 맵의 각각의 엘리먼트 (예를 들어, 220) 는 이전 계층에서의 뉴런들의 범위 (예를 들어, 피처 맵들 (218)) 로부터 그리고 다중 채널들 각각으로부터 입력을 수신한다. 피처 맵에서의 값들은 교정 (rectification) 과 같은 비-선형성, max(0,x) 으로 추가로 프로세싱될 수도 있다. 인접한 뉴런들로부터의 값들은 추가로 풀링될 수도 있으며, 이는 다운 샘플링에 대응하고, 부가적인 로컬 불변 및 차원성 감소를 제공할 수도 있다.

[0070] 도 3 은 딥 콘볼루션 네트워크 (350) 의 일 예를 나타내는 블록도이다. 딥 콘볼루션 네트워크 (350) 는 접속성 및 가중치 공유에 기초한 다수의 상이한 타입들의 계층들을 포함할 수도 있다. 도 3 에 나타낸 바와 같이, 딥 콘볼루션 네트워크 (350) 는 콘볼루션 블록들 (354A, 354B) 을 포함한다. 콘볼루션 블록들 (354A, 354B) 의 각각은 콘볼루션 계층 (CONV)(356), 정규화 계층 (LNorm)(358), 및 최대 풀링 계층 (MAX POOL)(360) 으로 구성될 수도 있다.

[0071] 콘볼루션 계층들 (356) 은 피처 맵을 생성하기 위해 입력 데이터 (352) 에 적용될 수도 있는 하나 이상의 콘볼루션 필터들을 포함할 수도 있다. 단지 2개의 콘볼루션 블록들(354A, 354B) 만이 도시되지만, 본 개시는 그렇게 제한되지 않으며, 대신에, 설계 선호도에 따라 임의의 수의 콘볼루션 블록들(예를 들어, 블록들(354A, 354B))이 딥 콘볼루션 네트워크(350)에 포함될 수도 있다. 정규화 계층 (358) 은 콘볼루션 필터들의 출력을 정규화할 수도 있다. 예를 들어, 정규화 계층 (358) 은 화이트닝 또는 측면 억제를 제공할 수도 있다. 최대 풀링 계층 (360) 은 로컬 불변 및 차원성 감소를 위해 공간에 걸쳐 다운 샘플링 집성을 제공할 수도 있다.

[0072] 예를 들어, 딥 콘볼루션 네트워크의 병렬 필터 뱅크들은 높은 성능 및 낮은 전력 소비를 달성하기 위해 이미지 프로세싱 시스템(100)의 CPU(102) 또는 GPU(104) 상에 로딩될 수도 있다. 대안적인 실시양태들에서, 병렬 필터 뱅크들은 이미지 프로세싱 시스템(100)의 DSP(106) 또는 ISP(116) 상에 로딩될 수도 있다. 또한, 딥 콘볼루션 네트워크(350)는 센서 프로세서(114)와 같은 이미지 프로세싱 시스템(100) 상에 존재할 수 있는 다른 프로세싱 블록들에 액세스할 수도 있다.

[0073] 딥 콘볼루션 네트워크(350)는 또한 계층(362A)("FC1"로 라벨링됨) 및 계층(362B)("FC2"로 라벨링됨)과 같은 하나 이상의 완전히 연결된 계층들을 포함할 수도 있다. 딥 콘볼루션 네트워크 (350) 는 로지스틱 회귀 (logistic regression; LR) 계층 (364) 을 더 포함할 수도 있다. 딥 콘볼루션 네트워크 (350) 의 각각의 계층 (356, 358, 360, 362, 364) 사이에는 업데이트될 가중치들 (미도시) 이 있다. 계층들 (예를 들어, 356, 358, 360, 362, 364) 각각의 출력은 콘볼루션 블록들 (354A) 중 첫번째에 공급된 입력 데이터 (352)(예를 들어, 이미지들, 오디오, 비디오, 센서 데이터 및/또는 다른 입력 데이터) 로부터 계위적 피처 표현들을 학습하기 위해 딥 콘볼루션 네트워크 (350) 에서 계층들 (예를 들어, 356, 358, 360, 362, 364) 중 후속하는 하나의 입력으로서 작용할 수도 있다. 딥 콘볼루션 네트워크 (350) 의 출력은 입력 데이터 (352) 에 대한 분류 스코어(classification score) (366) 이다. 분류 스코어 (366) 는 확률들의 세트일 수도 있으며, 여기서 각각의 확률은 피처들의 세트로부터의 피처를 포함하는, 입력 데이터의 확률이다.

[0074] 이미지 및 비디오 콘텐츠는 저장될 수 있고/있거나 디바이스들 간에 공유될 수도 있다. 예를 들어, 이미지 및 비디오 콘텐츠는 미디어 호스팅 서비스들 및 공유 플랫폼들에 업로드될 수 있고, 다양한 디바이스들에 전송될 수 있다. 압축되지 않은 이미지 및 비디오 콘텐츠를 기록하는 것은 일반적으로 이미지 및 비디오 콘텐츠의 해상도가 증가함에 따라 크게 증가하는 큰 파일 사이즈를 초래한다. 예를 들어, 1080p/24에 기록된 채널당 압축되지 않은 16-비트 비디오(예를 들어, 초당 24개의 프레임이 캡처되는 폭 1920 픽셀 및 높이 1080 픽셀의 해상도)는 초당 12.4 메가바이트, 또는 초당 297.6 메가바이트를 차지할 수도 있다. 초당 24개의 프레임으로 4K 해상도로 기록된 채널당 압축되지 않은 16비트 비디오는 프레임당 49.8메가바이트, 즉 초당 1195.2메가바이트를 차지할 수 있다.

[0075] 압축되지 않은 이미지 및 비디오 콘텐츠는 물리적 저장을 위한 상당한 메모리 및 송신을 위한 상당한 대역폭을 수반할 수도 있는 큰 파일들을 초래할 수 있기 때문에, 이러한 비디오 콘텐츠를 압축하기 위한 기법들이 이용될 수 있다. 예를 들어, 이미지 콘텐츠의 사이즈 - 따라서 이미지 콘텐츠를 저장하는 데 수반되는 스토리지의 양 및 비디오 콘텐츠를 전달하는 데 수반되는 대역폭의 양을 감소시키기 위해, 다양한 압축 알고리즘들이 이미지 및 비디오 콘텐츠에 적용될 수도 있다.

[0076] 일부 경우들에서, 이미지 콘텐츠는 특히 JPEG(Joint Photographic Experts Group), BPG(Better Portable

Graphics)와 같은 선험적으로(*a priori*) 정의된 압축 알고리즘을 사용하여 압축될 수 있다. JPEG는 이산 코사인 변환(Discrete Cosine Transform; DCT)에 기반한 압축의 손실 형태이다. 예를 들어, 이미지의 JPEG 압축을 수행하는 디바이스는 이미지를 최적의 컬러 공간 (예를 들어, 휘도 (Y), 크로미넌스-블루 (Cb), 크로미넌스-레드 (Cr) 를 포함하는 YCbCr 컬러 공간) 으로 변환할 수 있고, 픽셀들의 그룹들을 함께 평균화함으로써 크로미넌스 성분들을 다운샘플링할 수 있고, DCT 함수를 픽셀들의 블록들에 적용하여 리던던트 이미지 데이터를 제거하고 따라서 이미지 데이터를 압축할 수 있다. 압축은 이미지 내부의 유사한 영역들의 식별에 기초하고, 그 영역들을 (DCT 함수에 기초하여) 동일한 컬러 코드로 변환한다. 비디오 콘텐츠는 또한 MPEG(Motion Picture Experts Group) 알고리즘, H.264, 또는 고효율 비디오 코딩 알고리즘과 같은 선험적으로 정의된 압축 알고리즘을 사용하여 압축될 수 있다.

[0077] 이러한 선험적으로 정의된 압축 알고리즘들은 원시 이미지 및 비디오 콘텐츠에서의 정보의 대부분을 유지할 수 있을 수 있고, 신호 프로세싱 및 정보 이론 아이디어에 기초하여 선험적으로 정의될 수도 있다. 그러나, 이들 미리 정의된 압축 알고리즘들이 일반적으로 (예를 들어, 임의의 타입의 이미지/비디오 콘텐츠에) 적용가능할 수도 있지만, 압축 알고리즘들은 콘텐츠, 비디오 캡처 및 전달을 위한 새로운 해상도들 또는 프레임 레이트들, 비-자연 이미지(예를 들어, 레이더 이미지 또는 다양한 센서들을 통해 캡처된 다른 이미지) 등에서의 유사성들을 고려하지 않을 수도 있다.

[0078] 선험적으로 정의된 압축 알고리즘들은 손실성 압축 알고리즘들로 간주된다. 입력 영상(또는 비디오 프레임)의 손실성 압축(lossy compression)에서, 입력 이미지는 정확한 입력 이미지가 재구성되도록 코딩될 수 없고 디코딩/재구성될 수 없다. 오히려, 손실성 압축에서, 압축된 입력 이미지의 디코딩/재구성 후에 입력 이미지의 근사 버전이 생성된다. 손실성 압축은 재구성된 이미지에 존재하는 아티팩트들을 초래하는 왜곡을 감수하면서 비트레이트의 감소를 초래한다. 따라서, 손실성 압축 시스템들에서는 레이트-왜곡 트레이드 오프(rate-distortion trade-off)가 존재한다. 특정 압축 방법들(예를 들어, 다른 것들 중에서도, JPEG, BPG)의 경우, 왜곡 기반 아티팩트들은 차단 또는 다른 아티팩트들의 형태를 취할 수 있다. 일부 경우들에서, 신경망 기반 압축이 사용될 수 있고 이미지 데이터 및 비디오 데이터의 고품질 압축을 초래할 수 있다. 일부 경우들에서, 블러링(blurring) 및 컬러 시프트(color shift)가 아티팩트들의 예들이다.

[0079] 비트레이트가 입력 데이터의 진정한 엔트로피(true entropy)보다 낮아질 때마다, 정확한 입력 데이터를 재구성하는 것이 어렵거나 불가능할 수 있다. 그러나, 데이터의 압축/압축해제로부터 실현되는 왜곡/손실이 있다는 사실이 재구성된 이미지 또는 프레임이 아티팩트들을 가질 필요가 없다는 것을 의미하지는 않는다. 실제로, 압축된 이미지를 높은 시각적 품질을 갖는 다른 유사하지만 상이한 이미지로 재구성하는 것이 가능할 수 있다.

[0080] 이전에 언급된 바와 같이, 본 명세서에 설명된 시스템들 및 기술들은 하나 이상의 머신 러닝(ML) 시스템들을 사용하여 압축 및 압축해제를 수행할 수 있다. 일부 예들에서, 머신 러닝 기술들은 고품질의 시각적 출력들을 생성하는 이미지 및/또는 비디오 압축을 제공할 수 있다. 일부 예들에서, 본 명세서에 설명된 시스템들 및 기법들은 레이트-왜곡 오토인코더(RD-AE)와 같은 심층 신경망(들)을 사용하여 콘텐츠(예를 들어, 이미지 콘텐츠, 비디오 콘텐츠, 오디오 콘텐츠 등)의 압축 및 압축해제를 수행할 수 있다. 심층 신경망(deep neural network)은 이미지들을 잠재 코드 공간(latent code space)(예를 들어, 코드들 z의 세트를 포함함)으로 맵핑하는 오토인코더(autoencoder; AE)를 포함할 수 있다. 잠재 코드 공간은 인코더 및 디코더에 의해 사용되고 콘텐츠가 코드들 z로 인코딩된 코드 공간을 포함할 수 있다. 코드들(예를 들어, 코드들 z)은 또한 잠재들, 잠재 변수들 또는 잠재 표현들로 지칭될 수 있다. 심층 신경망은 잠재 코드 공간으로부터 코드들 z를 손실없이 압축할 수 있는 확률적 모델(프리어 또는 코드 모델이라고도 함)을 포함할 수 있다. 확률 모델은 입력 데이터에 기초하여 인코딩된 데이터를 나타낼 수 있는 코드들 z의 세트에 대한 확률 분포를 생성할 수 있다. 일부 경우들에서, 확률 분포는 (P(z))로 표시될 수 있다.

[0081] 일부 예들에서, 심층 신경망은 확률 분포 P(z) 및/또는 코드들의 세트 z 에 기초하여 출력될 압축된 데이터를 포함하는 비트스트림을 생성하는 산술 코더를 포함할 수도 있다. 압축된 데이터를 포함하는 비트스트림은 저장될 수 있고 및/또는 수신 디바이스로 송신될 수 있다. 수신 디바이스는 예를 들어, 산술 디코더, 확률적(또는 코드) 모델, 및 AE의 디코더를 사용하여 비트스트림을 디코딩 또는 압축해제하기 위한 역 프로세스를 수행할 수 있다. 압축된 데이터를 포함하는 비트스트림을 생성한 디바이스는 또한 스토리지로부터 압축된 데이터를 취출할 때 유사한 디코딩/압축해제 프로세스를 수행할 수 있다. 업데이트된 모델 파라미터들을 압축/인코딩 및 압축 해제/디코딩하기 위해 유사한 기술이 수행될 수 있다.

[0082] 일부 예들에서, RD-AE는 (하이-레이트 및 로우-레이트 동작들을 포함하는) 멀티-레이트 AE로서 수행하도록 트레이닝되고 동작될 수 있다. 예를 들어, 멀티-레이트 AE의 인코더에 의해 생성된 잠재 코드 공간은 2개 이상의 청크들(예를 들어, 청크들 z_1 및 z_2 로 분할된 코드들 z)로 분할될 수 있다. 고속 동작에서, 멀티-레이트 AE는 RD-AE에 대해 위에서 설명된 동작들과 유사하게, 데이터를 압축해제하기 위해 수신 디바이스에 의해 사용될 수 있는 전체 잠재 공간(예를 들어, z_1, z_2 등을 포함하는 코드들 z)에 기초하는 비트스트림을 전송할 수 있다. 로우-레이트 동작에서, 수신 디바이스로 전송되는 비트스트림은 잠재 공간의 서브세트(예를 들어, z_2가 아닌 청크 z_1)에 기초한다. 수신 디바이스는 전송된 서브세트에 기초하여 잠재 공간의 나머지 부분을 추론할 수 있고, 잠재 공간의 서브세트 및 잠재 공간의 추론된 나머지 부분을 사용하여 재구성된 데이터를 생성할 수 있다.

[0083] RD-AE 또는 멀티-레이트 AE를 사용하여 콘텐츠를 압축(및 압축 해제)함으로써, 인코딩 및 디코딩 메커니즘은 다양한 사용 사례들에 적응가능할 수 있다. 머신 러닝 기반 압축 기술들은 높은 품질 및/또는 감소된 비트레이트를 갖는 압축된 콘텐츠를 생성할 수 있다. 일부 예들에서, RD-AE는 이미지 및/또는 비디오 데이터포인트들과 같은 데이터포인트들의 데이터세트에 걸쳐 평균 레이트-왜곡 손실을 최소화하도록 트레이닝될 수 있다.

일부 경우들에서, RD-AE는 또한 수신기에 전송되고 수신기에 의해 디코딩될 특정 데이터포인트에 대해 미세-튜닝될 수 있다. 일부 예들에서, 데이터 포인트 상에서 RD-AE 를 미세-튜닝함으로써, RD-AE 는 높은 압축(레이트/왜곡) 성능을 획득할 수 있다. RD-AE 와 연관된 인코더는 비트스트림을 디코딩하기 위해 수신기(예를 들어, 디코더)에 AE 모델 또는 AE 모델의 일부를 전송할 수 있다.

[0084] 일부 경우들에서, 신경망 압축 시스템은 (양자화된) 잠재 표현으로부터 입력 인스턴스(예를 들어, 입력 이미지, 비디오, 오디오 등)를 재구성할 수 있다. 신경망 압축 시스템은 또한 잠재 표현을 무손실 압축하기 전에 프리어(prior)를 사용할 수 있다. 일부 경우들에서, 신경망 압축 시스템은 테스트 시간 데이터 분포가 알려져 있고 비교적 낮은 엔트로피(예를 들어, 정적 장면을 보는 카메라, 자율 주행 차량에서의 대시 캠 등)를 결정할 수 있고, 이러한 분포에 미세-튜닝되거나 적응될 수 있다. 미세-튜닝 또는 적응은 개선된 레이트/왜곡 (RD) 성능을 야기할 수 있다. 일부 예들에서, 신경망 압축 시스템의 모델은 압축될 단일 입력 인스턴스에 적응될 수 있다. 신경망 압축 시스템은 모델 업데이트들을 제공할 수 있으며, 이는 일부 예들에서 잠재 표현과 함께, 파라미터-공간 프리어를 사용하여 양자화되고 압축될 수 있다.

[0085] 미세-튜닝은 모델 양자화의 효과 및 모델 업데이트들을 전송함으로써 발생된 부가적인 비용들을 고려할 수 있다. 일부 예들에서, 신경망 압축 시스템은 RD 손실뿐만 아니라, 모델 프리어 하에서 모델 업데이트들을 전송하는데 필요한 비트들의 수를 측정하는 추가적인 모델 레이트 항(M)을 사용하여 미세-튜닝될 수 있으며, 이는 결합된 RDM 손실을 초래한다.

[0086] 도 4 는 본 개시의 일부 예들에 따른, 송신 디바이스 (410) 및 수신 디바이스 (420) 를 포함하는 시스템 (400) 을 나타내는 다이어그램이다. 송신 디바이스(410) 및 수신 디바이스(420)는 일부 경우들에서 RD-AE로 각각 지칭될 수 있다. 송신 디바이스(410)는 이미지 콘텐츠를 압축할 수 있고, 압축된 이미지 콘텐츠를 저장할 수 있고 그리고/또는 압축 해제를 위해 압축된 이미지 콘텐츠를 수신 디바이스(420)에 송신할 수 있다. 수신 디바이스(420)는 압축된 이미지 콘텐츠를 압축 해제할 수 있고, 압축 해제된 이미지 콘텐츠를 (예를 들어, 디스플레이, 편집 등을 위해) 수신 디바이스(420) 상에 출력할 수 있고 그리고/또는 압축 해제된 이미지 콘텐츠를 수신 디바이스(420)에 연결된 다른 디바이스들(예를 들어, 텔레비전, 모바일 디바이스, 또는 다른 디바이스)에 출력할 수 있다. 일부 경우들에서, 수신 디바이스(420)는 (인코더(422)를 사용하여) 이미지 콘텐츠를 압축하고 압축된 이미지 콘텐츠를 저장 및/또는 송신 디바이스(410)와 같은 다른 디바이스에 송신함으로써 송신 디바이스가 될 수 있다(이 경우 송신 디바이스(410)는 수신 디바이스가 될 것이다). 시스템(400)이 이미지 압축 및 압축해제와 관련하여 본 명세서에서 설명되지만, 당업자는 시스템(400)이 비디오 콘텐츠를 압축 및 압축해제하기 위해 본 명세서에서 설명된 기술들을 사용할 수 있다는 것을 인식할 것이다.

[0087] 도 4에 도시된 바와 같이, 송신 디바이스(410)는 이미지 압축 파이프라인을 포함하고, 수신 디바이스(420)는 이미지 비트스트림 압축해제 파이프라인을 포함한다. 송신 디바이스(410)의 이미지 압축 파이프라인 및 수신 디바이스(420)의 비트스트림 압축해제 파이프라인은 일반적으로 본 개시의 양상들에 따라, 이미지 콘텐츠를 압축하고 그리고/또는 수신된 비트스트림을 이미지 콘텐츠로 압축해제하기 위해 하나 이상의 인공 신경망들을 사용한다. 송신 디바이스(410)에서의 이미지 압축 파이프라인은 오토인코더(401), 코드 모델(404), 및 산술 코더(406)를 포함한다. 일부 구현들에서, 산술 코더(406)는 선택적이며, 일부 경우들에서 생략될 수 있다.

수신 디바이스(420)의 이미지 압축해제 파이프라인은 오토인코더(421), 코드 모델(424), 및 산술 디코더(426)를 포함한다. 일부 구현들에서, 산술 디코더(426)는 선택적이며, 일부 경우들에서 생략될 수 있다. 송신 디바이스(410)의 오토인코더(401) 및 코드 모델(404)은 이전에 트레이닝되고 따라서 트레이닝된 머신 러닝 시스템의 추론 또는 동작 동안 동작들을 수행하도록 구성된 머신 러닝 시스템으로서 도 4에 예시된다. 오토인코더(421), 코드 모델(424), 및 완료 모델(425)은 또한 이전에 트레이닝된 머신 러닝 시스템으로서 예시된다.

[0088] 오토인코더(401)는 인코더(402) 및 디코더(403)를 포함한다. 인코더(402)는 비압축 이미지 콘텐츠의 하나 이상의 이미지 내의 픽셀들을 잠재 코드 공간(코드들 z 을 포함함)에 맵핑함으로써 수신된 비압축 이미지 콘텐츠에 대해 손실성 압축을 수행할 수 있다. 일반적으로, 인코더(402)는 압축된(또는 인코딩된) 이미지를 나타내는 코드들 z 이 이산 또는 바이너리이도록 구성될 수도 있다. 이들 코드들은 확률적 섭동 기법들, 소프트 벡터 양자화, 또는 별개의 코드들을 생성할 수 있는 다른 기법들에 기초하여 생성될 수도 있다. 일부 양태들에서, 오토인코더 (401) 는 압축되지 않은 이미지들을 압축가능한 (낮은 엔트로피) 분포를 갖는 코드들에 맵핑할 수도 있다. 이러한 코드들은 사전 정의되거나 학습된 프리어 분포에 교차 엔트로피로 가까울 수도 있다.

[0089] 일부 예들에서, 오토인코더 (401) 는 콘볼루션 아키텍처를 사용하여 구현될 수 있다. 예를 들어, 일부 경우들에서, 오토인코더(401)는 오토인코더(401)가 잠재 코드 공간에 이미지 콘텐츠를 맵핑하기 위한 공간적 필터들을 학습하도록 2차원 콘볼루션 신경망(CNN)으로서 구성될 수 있다. 시스템 (400) 이 비디오 데이터를 코딩하기 위해 사용되는 예들에서, 오토인코더 (401) 는 오토인코더 (401) 가 잠재적 코드 공간에 비디오를 맵핑하기 위한 공간-시간적 필터들을 학습하도록 3 차원 CNN 으로서 구성될 수 있다. 이러한 네트워크에서, 오토인코더(401)는 키 프레임(예를 들어, 시퀀스 내의 후속 프레임들이 시퀀스 내의 초기 프레임에 대한 차이로서 기술되는 프레임들의 시퀀스의 시작을 마킹하는 초기 프레임), 키 프레임과 비디오 내의 다른 프레임들 사이의 워핑(warping)(또는 차이들), 및 잔차 인자의 관점에서 비디오를 인코딩할 수 있다. 다른 양태들에서, 오토인코더(401)는 이전 프레임들, 프레임들 사이의 잔차 인자(residual factor)에 대해 컨디셔닝되고, 적층 채널들을 통해 컨디셔닝하거나 순환 계층들을 포함하는 2차원 신경망으로서 구현될 수도 있다.

[0090] 오토인코더(401)의 인코더(402)는 입력으로서 제 1 이미지(도 4에서 이미지 x 로서 지정됨)를 수신할 수 있고, 그 제 1 이미지 x 를 잠재 코드 공간에서의 코드 z 에 맵핑할 수 있다. 전술한 바와 같이, 인코더(402)는 잠재 코드 공간이 각각의 (x, y) 포지션에서 그 포지션에 중심을 둔 이미지 x 의 블록을 기술하는 벡터를 갖도록 2차원 콘볼루션 네트워크로서 구현될 수 있다. x-좌표는 이미지 x 의 블록 내의 수평 픽셀 위치를 나타낼 수 있고, y-좌표는 이미지 x 의 블록 내의 수직 픽셀 위치를 나타낼 수 있다. 비디오 데이터를 코딩할 때, 잠재 코드 공간은 t 변수 또는 포지션을 가질 수 있으며, t 변수는 (공간 x- 및 y-좌표들에 더하여) 비디오 데이터의 블록에서의 타임스탬프를 나타낸다. 수평 및 수직 픽셀 포지션들의 2개의 차원들을 사용함으로써, 벡터는 이미지 x 에서 이미지 패치를 기술할 수 있다.

[0091] 그 후 오토인코더(401)의 디코더(403)는 코드 z 를 압축 해제하여 제 1 이미지 x 의 재구성 $\hat{x}$ 을 획득할 수 있다. 일반적으로, 재구성 $\hat{x}$ 은 압축되지 않은 제 1 이미지 x 의 근사일 수 있고 제 1 이미지 x 의 정확한 사본일 필요는 없다. 일부 경우들에서, 재구성된 이미지 $\hat{x}$ 는 송신 디바이스에 저장하기 위한 압축된 이미지 파일로서 출력될 수 있다.

[0092] 코드 모델(404)은 인코딩된 이미지 또는 그 일부를 나타내는 코드 z 를 수신하고, 그 코드 z 를 나타내는데 사용될 수 있는 압축된 코드워드들의 세트에 걸쳐 확률 분포 $P(z)$를 생성한다. 일부 예들에서, 코드 모델(404)은 확률적 자동-회귀적 생성 모델을 포함할 수 있다. 일부 경우들에서, 확률 분포가 생성될 수도 있는 코드

들은 산술 코더(406)에 기초하여 비트 할당을 제어하는 학습된 분포를 포함한다. 예를 들어, 산술 코더(406)를 사용하여, 제 1 코드 z에 대한 압축 코드가 개별적으로 예측될 수 있고; 제 2 코드 z에 대한 압축 코드가 제 1 코드 z에 대한 압축 코드에 기초하여 예측될 수 있고; 제 3 코드 z에 대한 압축 코드가 제 1 코드 z 및 제 2 코드 z에 대한 압축 코드들에 기초하여 예측될 수 있는 것 등이 가능하다. 압축 코드들은 일반적으로 압축될 주어진 이미지의 상이한 공간-시간 청크들을 나타낸다.

[0093] 일부 양상들에서, z는 3차원 텐서로서 표현될 수도 있다. 텐서의 3개의 차원들은 피처 채널 차원, 및 높이 및 폭 공간 차원들(예를 들어, 코드 $z_{c,w,h}$로 표시됨)을 포함할 수도 있다. 각각의 코드 $z_{c,w,h}$(채널 및 수평 및 수직 포지션에 의해 인덱싱된 코드를 나타냄)는 코드들의 고정적이고 이론적으로 임의적인 순서일 수 있는 이전 코드(previous code)에 기초하여 예측될 수 있다. 일부 예들에서, 코드들은 주어진 이미지 파일을 시작부터 종료까지 분석하고 래스터 스캔 순서로 이미지의 각각의 블록을 분석함으로써 생성될 수 있다.

[0094] 코드 모델(404)은 확률적 자동 회귀 모델을 사용하여 입력 코드 z에 대한 확률 분포를 학습할 수 있다. 확률 분포는 (전술한 바와 같이) 그것의 이전 값들에 대해 컨디셔닝될 수 있다. 일부 예들에서, 확률 분포는 다음의 식으로 표현될 수 있다:

[0095] $$P(z) = \prod_{c=0}^{C} \prod_{w=0}^{W} \prod_{h=0}^{H} p(z_{c,w,h} | z_{0:c,0:w,0:h})$$ 식 (1),

[0096] 여기서, c는 모든 이미지 채널들 C (예를 들어, R, G, 및 B 채널들, Y, Cb, 및 Cr 채널들, 또는 다른 채널들)에 대한 채널 인덱스이고, w는 총 이미지 프레임 폭 W에 대한 폭 인덱스이며, h 는 총 이미지 높이 H 에 대한 높이 인덱스이다.

[0097] 일부 예들에서, 확률 분포 $P(z)$는 인과적 콘볼루션들의 완전 콘볼루션 신경망에 의해 예측될 수 있다. 일부 양태들에서, 콘볼루션 신경망의 각각의 계층의 커널들은, 콘볼루션 네트워크가 이전 값들 $z_{0:c,0:w,0:h}$을 알고 있고 확률 분포를 계산함에 있어서 다른 값들을 알지 못할 수도 있도록 마스킹될 수 있다. 일부 양태들에서, 콘볼루션 네트워크의 최종 계층은 잠재 공간 내의 코드가 입력 값에 대해 적용 가능할 확률 (예를 들어, 주어진 코드가 주어진 입력을 압축하는데 사용될 수 있는 가능성) 을 결정하는 소프트맥스 함수(softmax function)를 포함할 수도 있다.

[0098] 산술 코더(406)는 코드 모델(404)에 의해 생성된 확률 분포 $P(z)$를 사용하여 코드 z의 예측에 대응하는 비트스트림(415)(" 0010011..."으로 도 4에 도시됨)을 생성한다. 코드 z의 예측은 가능한 코드들의 세트에 걸쳐 생성된 확률 분포 $P(z)$에서 가장 높은 확률 스코어를 갖는 코드로서 표현될 수 있다. 일부 양태들에서, 산술 코더 (406) 는 코드 z의 예측 및 오토인코더 (401)에 의해 생성된 실제 코드 z의 정확도에 기초하여 가변 길이의 비트스트림을 출력할 수 있다. 예를 들어, 비트스트림(415)은 예측이 정확하면 짧은 코드워드에 대응할 수 있는 반면, 비트스트림(415)은 코드 z 와 코드 z 의 예측 사이의 차이의 크기가 증가함에 따라 더 긴 코드워드들에 대응할 수도 있다.

[0099] 일부 경우들에서, 비트스트림(415)은 압축된 이미지 파일에의 저장을 위해 산술 코더(406)에 의해 출력될 수 있다. 비트스트림(415)은 또한 요청 디바이스(예를 들어, 도 4에 예시된 바와 같은 수신 디바이스(420))로의 송신을 위해 출력될 수 있다. 일반적으로, 산술 코더(406)에 의해 출력된 비트스트림(415)은 z 이 압축 이미지 파일에 적용된 압축해제 프로세스들 동안 정확하게 복원될 수도 있도록 z 를 무손실 인코딩할 수도 있다.

[0100] 산술 코더(406)에 의해 생성되고 송신 디바이스(410)로부터 송신된 비트스트림(415)은 수신 디바이스(420)에 의해 수신될 수 있다. 송신 디바이스(410)와 수신 디바이스(420) 사이의 송신은 다양한 적합한 유선 또는 무선 통신 기술들 중 임의의 것을 사용하여 발생할 수 있다. 송신 디바이스(410)와 수신 디바이스(420) 사이의 통신은 직접적일 수도 있거나, 또는 하나 이상의 네트워크 인프라스트럭처 컴포넌트들(예를 들어, 기지국들, 중계국들, 이동국들, 네트워크 허브들, 라우터들, 및/또는 다른 네트워크 인프라스트럭처 컴포넌트들)을 통해 수행될 수도 있다.

[0101] 예시된 바와 같이, 수신 디바이스 (420) 는 산술 디코더 (426), 코드 모델 (424), 및 오토인코더 (421) 를 포함할 수 있다. 오토인코더(421)는 인코더(422) 및 디코더(423)를 포함한다. 디코더(423)는, 주어진 입력에 대해, 디코더(403)와 동일하거나 유사한 출력을 생성할 수 있다. 오토인코더 (421) 가 인코더 (422) 를 포함하는 것으로 예시되지만, 인코더 (422) 는 송신 디바이스 (410) 로부터 수신된 코드 z 로부터 $\hat{x}$ (예를 들어, 송신 디바이스 (410)에서 압축된 원래의 이미지 x 의 근사치) 를 획득하기 위해 디코딩 프로세스 동안 사용될 필요는 없다.

[0102] 수신된 비트스트림(415)은 비트스트림으로부터 하나 이상의 코드들 z 을 획득하기 위해 산술 디코더(426)에 입력될 수 있다. 산술 디코더(426)는 가능한 코드들의 세트에 걸쳐 코드 모델(424)에 의해 생성된 확률 분포 $P(z)$ 및 각각의 생성된 코드 z 를 비트스트림과 연관시키는 정보에 기초하여 압축해제된 코드 z 를 추출할 수도 있다. 비트스트림의 수신된 부분 및 다음 코드 z 의 확률적 예측이 주어지면, 산술 디코더(426)는 송신 디바이스 (410)에서 산술 코더(406)에 의해 인코딩되었던 새로운 코드 z 를 생성할 수 있다. 새로운 코드 z 를 사용하여, 산술 디코더(426)는 연속적인 코드 z 에 대해 확률적 예측을 수행하고, 비트스트림의 추가적인 부분을 판독하고, 전체 수신된 비트스트림이 디코딩될 때까지 연속적인 코드 z 를 디코딩할 수 있다. 압축 해제된 코드 z 는 오토 인코더(421) 내의 디코더(423)로 제공될 수도 있다. 디코더(423)는 코드 z 를 압축해제하여 이미지 콘텐츠 x 의 근사치 $\hat{x}$ (재구성된 또는 디코딩된 이미지라고 지칭될 수 있음)를 출력한다. 일부 경우들에서, 콘텐츠 x 의 근사 $\hat{x}$ 는 추후 취출(retrieval)을 위해 저장될 수 있다. 일부 경우들에서, 콘텐츠 x 의 근사 $\hat{x}$ 는 수신 디바이스(420)에 의해 복원되고 수신 디바이스(420)에 통신가능하게 커플링되거나 그와 통합된 스크린 상에 디스플레이될 수도 있다.

[0103] 전술한 바와 같이, 송신 디바이스 (410) 의 오토인코더 (401) 및 코드 모델 (404) 은 이전에 트레이닝된 머신 러닝 시스템으로서 도 4에 예시된다. 일부 양태들에서, 오토인코더(401) 및 코드 모델(404)은 이미지 데이터를 사용하여 함께 트레이닝될 수 있다. 예를 들어, 오토인코더(401)의 인코더(402)는 입력으로서 제 1 트레이닝 이미지 n 를 수신할 수 있고, 제 1 트레이닝 이미지 n 를 잠재 코드 공간 내의 코드 z 에 맵핑할 수 있다.

코드 모델(404)은 (전술한 기법들과 유사하게) 확률적 자동 회귀 모델을 사용하여 코드 z 에 대한 확률 분포 $P(z)$ 를 학습할 수 있다. 산술 코더(406)는 이미지 비트스트림을 생성하기 위해 코드 모델(404)에 의해 생성된 확률 분포 $P(z)$ 를 사용할 수 있다. 코드 모델(404)로부터의 비트스트림 및 확률 분포 $P(z)$ 를 사용하여, 산술 코더(406)는 코드 z 를 생성할 수 있고, 그 코드 z 를 오토인코더(401)의 디코더(403)로 출력할 수 있다. 그 후, 디코더(403)는 코드 z 를 압축 해제하여 제 1 트레이닝 이미지 n 의 재구성 $\hat{n}$ (여기서, 재구성

$\hat{n}$ 은 압축되지 않은 제 1 트레이닝 이미지 n 의 근사치)을 획득할 수 있다.

[0104] 일부 경우들에서, 송신 디바이스(410)의 트레이닝 동안 사용되는 역전파 엔진은 하나 이상의 손실 함수들에 기초하여 오토인코더(401)의 신경망 및 코드 모델(404)의 파라미터들(예를 들어, 가중치들, 바이어스들 등)을 튜닝하기 위해 역전파 프로세스를 수행할 수 있다. 일부 경우들에서, 역전파 프로세스는 확률적 경사 하강 기법들에 기초할 수 있다. 역전파는 순방향 패스, 하나 이상의 손실 함수들, 역방향 패스, 및 가중치(및/또는 다른 파라미터(들)) 업데이트를 포함할 수 있다. 순방향 패스, 손실 함수, 역방향 패스, 및 파라미터 업데이트는 하나의 트레이닝 반복에 대해 수행될 수 있다. 프로세스는 신경망의 가중치들 및/또는 다른 파라미터들이 정확하게 튜닝될 때까지 트레이닝 데이터의 각각의 세트에 대해 특정 횟수의 반복들 동안 반복될 수 있다.

[0105] 예를 들어, 오토인코더(401)는 n 와 $\hat{n}$ 를 비교하여, 제 1 트레이닝 이미지 n 와 재구성된 제 1 트레이닝 이미지 $\hat{n}$ 사이의 손실(예를 들어, 거리 벡터 또는 다른 차이 값으로 표현됨)을 결정할 수 있다. 손실 함수는 출력에서의 에러를 분석하는 데 사용될 수 있다. 일부 예들에서, 손실은 최대 가능성에 기초할 수 있다. 비압축된 이미지 n 를 입력으로서 사용하고 재구성된 이미지 $\hat{n}$ 를 출력으로서 사용하는 하나의 예시적인 예에서, 손실 함수 $\mathrm{Loss} = \mathrm{D} + \mathrm{beta} * \mathrm{R}$ 는 오토인코더(401) 및 코드 모델(404)의 신경망 시스템을 트레이닝하기 위해 사용될 수 있으며, 여기서, R 는 레이트(rate)이고, D 는 왜곡(distortion)이고, *는 곱셈 함수를 나타내고, beta 는 비트레이트를 정의하는 값으로 설정되는 트레이드오프 파라미터(tradeoff parameter)이다.

다른 예에서, 손실 함수 $Loss0 = \sum_t distortion(n, \hat{n})$ 는 오토인코더(401) 및 코드 모델(404)의 신경망 시스템을 트레이닝하는데 사용될 수 있다. 다른 트레이닝 데이터가 사용될 때와 같은 일부 경우들에서 다른 손실 함수들이 사용될 수 있다. 다른 손실 함수의 일 예는 $E_{total} = \sum \frac{1}{2}(target - output)^2$ 로서 정의된 평균 제곱 오차(mean squared error; MSE)를 포함한다. MSE는 실제 해답의 1/2배에서 예측 (출력) 해답을 제곱한 합계를 계산한다.

[0106] 결정된 손실(예를 들어, 거리 벡터 또는 다른 차이 값)에 기초하여 그리고 역전파 프로세스를 사용하여, 오토인코더(401)의 신경망 시스템 및 코드 모델(404)의 파라미터들(예를 들어, 가중치들, 바이어스들 등)은 (수신된 이미지 콘텐츠와 잠재 코드 공간 사이의 맵핑들을 효과적으로 조정하여) 입력 비압축된 이미지들과 오토인코더(401)에 의한 출력으로서 생성된 압축된 이미지 콘텐츠 사이의 손실을 감소시키도록 조정될 수 있다.

[0107] 실제 출력 값들(재구성된 이미지)이 입력 이미지와 많이 다를 수도 있기 때문에, 제 1 트레이닝 이미지들에 대한 손실(또는 오차)이 높을 수도 있다. 트레이닝의 목표는 예측된 출력에 대한 손실량을 최소화하는 것이다. 신경망은 (대응하는 가중치들을 갖는) 신경망의 어느 노드들이 신경망의 손실에 가장 기여했는지를 결정함으로써 역방향 패스를 수행할 수 있고, 손실이 감소하고 궁극적으로 최소화되도록 가중치들(및/또는 다른 파라미터들)을 조정할 수 있다. 가중치들에 대한 손실의 도함수(dL/dW 로 표시됨, 여기서 W 는 특정 계층에서의 가중치들)는 신경망의 손실에 가장 기여한 가중치들을 결정하기 위해 계산될 수 있다. 예를 들어, 가중치들은 그들이 그래디언트의 반대 방향으로 변경되도록 업데이트될 수 있다. 가중치 업데이트는 $w = w_i - \eta \frac{dL}{dW}$ 로서 표기될 수 있으며, 여기서 w 는 가중치를 표기하고, w_i 는 초기 가중치를 표기하며, η 는 러닝 레이트를 표기한다. 러닝 레이트는 임의의 적절한 값으로 설정될 수 있으며, 높은 러닝 레이트는 더 큰 가중치 업데이트들을 포함하고, 더 낮은 값은 더 작은 가중치 업데이트들을 표시한다.

[0108] 오토인코더(401) 및 코드 모델(404)의 신경망 시스템은 원하는 출력이 달성될 때까지 이러한 방식으로 계속 트레이닝될 수 있다. 예를 들어, 오토인코더(401) 및 코드 모델(404)은 생성된 코드 z 의 압축해제로부터 초래되는 재구성된 이미지 $\hat{n}$ 와 입력 이미지 n 사이의 차이들을 최소화하거나 그렇지 않으면 감소시키기 위해 역전

파 프로세스를 반복할 수 있다.

[0109] 오토인코더 (421) 및 코드 모델 (424) 은 송신 디바이스 (410) 의 오토인코더 (401) 및 코드 모델 (404) 을 트레이닝하기 위해 상기 설명된 것과 유사한 기술들을 사용하여 트레이닝될 수 있다. 일부 경우들에서, 오토인코더 (421) 및 코드 모델 (424) 은 송신 디바이스 (410) 의 오토인코더 (401) 및 코드 모델 (404) 을 트레이닝하는데 사용되는 동일하거나 상이한 트레이닝 데이터세트를 사용하여 트레이닝될 수 있다.

[0110] 도 4에 도시된 예에서, 레이트-왜곡 오토인코더들(송신 디바이스(410) 및 수신 디바이스(420))은 비트레이트에 따라 추론에서 트레이닝되고 실행된다. 일부 구현들에서, 레이트-왜곡 오토인코더는, 가변하는 양들의 정보가 잠재 코드들 $\mathbf{z}$ 에서 제공될 때 (예를 들어, 입력 이미지에 대한 왜곡으로 인한 제한된 아티팩트들 없이 또는 제한된 아티팩트들을 갖는) 고품질 재구성된 이미지들 또는 비디오 프레임들의 생성 및 출력을 허용하기 위해 다수의 비트레이트들에서 트레이닝될 수 있다.

[0111] 일부 구현들에서, 잠재 코드들 z는 적어도 2개의 청크들 z_1 및 z_2 로 분할될 수 있다. RD-AE 모델이 높은-레이트 설정에서 사용될 때, 청크들 둘 모두는 디코딩을 위해 디바이스로 송신된다. 레이트-왜곡 오토인코더 모델이 로우-레이트 설정에서 사용될 때, 청크 z_1 만이 전송되고 청크 z_2 는 디코더 측에서 z_1 로부터 추론된다. z_1 으로부터 z_2 의 추론은 아래에서 더 상세히 설명되는 바와 같이 다양한 기법들을 사용하여 수행될 수 있다.

[0112] 일부 구현들에서, (예를 들어, 많은 양의 정보를 전달할 수 있는) 연속적인 잠재들(latents)의 세트 및 (예를 들어, 더 적은 정보를 포함하는) 대응하는 양자화된 이산 잠재들이 사용될 수 있다. RD-AE 모델을 트레이닝한 후, 보조 역양자화 모델이 트레이닝될 수 있다. 일부 경우들에서, RD-AE를 사용할 때, 이산 잠재들만이 송신되고, 보조 역양자화 모델은 이산 잠재들로부터 연속 잠재들을 추론하기 위해 디코더 측에서 사용된다.

[0113] 시스템 (400) 이 특정 컴포넌트들을 포함하는 것으로 도시되지만, 당업자는 시스템(400) 이 도 4 에 도시된 컴포넌트들보다 더 많거나 더 적은 컴포넌트들을 포함할 수 있다는 것을 이해할 것이다. 예를 들어, 시스템(400)의 송신 디바이스(410) 및/또는 수신 디바이스(420)는 또한, 일부 경우들에서, 하나 이상의 메모리 디바이스들(예를 들어, RAM, ROM, 캐시 등), 하나 이상의 네트워킹 인터페이스들(예를 들어, 유선 및/또는 무선 통신 인터페이스들 등), 하나 이상의 디스플레이 디바이스들, 및/또는 도 4에 도시되지 않은 다른 하드웨어 또는 프로세싱 디바이스들을 포함할 수 있다. 도 4에 도시된 컴포넌트들, 및/또는 시스템(400)의 다른 컴포넌트들은 하나 이상의 컴퓨팅 또는 프로세싱 컴포넌트들을 사용하여 구현될 수 있다. 하나 이상의 컴퓨트 컴포넌트들은 중앙 프로세싱 유닛 (CPU), 그래픽 프로세싱 유닛 (GPU), 디지털 신호 프로세서 (DSP), 및/또는 이미지 신호 프로세서 (ISP) 를 포함할 수 있다. 시스템(1400)으로 구현될 수 있는 컴퓨팅 디바이스 및 하드웨어 컴포넌트들의 예시적인 예가 도 14와 관련하여 아래에서 설명된다.

[0114] 시스템(400)은 단일 컴퓨팅 디바이스 또는 다수의 컴퓨팅 디바이스들의 일부이거나 이에 의해 구현될 수 있다. 일부 예들에서, 송신 디바이스(410)는 제 1 디바이스의 일부일 수 있고, 수신 디바이스(420)는 제 2 컴퓨팅 디바이스의 일부일 수 있다. 일부 예들에서, 송신 디바이스(410) 및/또는 수신 디바이스(420)는 전화 시스템(예를 들어, 스마트폰, 셀룰러 전화, 회의 시스템 등), 데스크톱 컴퓨터, 랩톱 또는 노트북 컴퓨터, 태블릿 컴퓨터, 셋톱 박스, 스마트 텔레비전, 디스플레이 디바이스, 게이밍 콘솔, 비디오 스트리밍 디바이스, SOC, IoT(Internet-of-Things) 디바이스, 스마트 웨어러블 디바이스(예를 들어, 헤드-마운티드 디스플레이(HMD), 스마트 안경 등), 카메라 시스템(예를 들어, 디지털 카메라, IP 카메라, 비디오 카메라, 보안 카메라 등), 또는 임의의 다른 적절한 전자 디바이스(들)와 같은 전자 디바이스(또는 디바이스들)의 일부로서 포함될 수 있다. 일부 경우들에서, 시스템(400)은 도 1에 도시된 이미지 프로세싱 시스템(100)에 의해 구현될 수 있다. 다른 경우들에서, 시스템(400)은 하나 이상의 다른 시스템들 또는 디바이스들에 의해 구현될 수 있다.

[0115] 도 5a는 예시적인 신경망 압축 시스템(500)을 나타내는 도면이다. 일부 예들에서, 신경망 압축 시스템(500)은 RD-AE 시스템을 포함할 수 있다. 도 5a에서, 신경망 압축 시스템(500)은 인코더(502), 산술 인코더(508), 산술 디코더(512), 및 디코더(514)를 포함한다. 일부 경우들에서, 인코더(502) 및/또는 디코더(514)는 각각 인코더(402) 및/또는 디코더(403)와 동일할 수 있다. 다른 경우에, 인코더(502) 및/또는 디코더(514)는 각각 인코더(402) 및/또는 디코더(403)와 상이할 수 있다.

[0116] 인코더(502)는 입력으로서 이미지(501)(이미지 x_i)를 수신할 수 있고, 이미지(501)(이미지 x_i)를 잠재 코드 공간

내의 잠재 코드(504)(잠재 z_i)에 맵핑 및/또는 변환할 수 있다. 이미지(501)는 프레임들의 시퀀스(예를 들어, 비디오)와 연관된 정지 이미지 및/또는 비디오 프레임을 나타낼 수 있다. 일부 경우들에서, 인코더(502)는 잠재 코드(504)를 생성하기 위해 순방향 패스를 수행할 수 있다. 일부 예들에서, 인코더(502)는 학습가능한 함수를 구현할 수 있다. 일부 경우들에서, 인코더(502)는 φ 에 의해 파라미터화된 학습가능한 함수를 구현할 수 있다. 예를 들어, 인코더(502)는 함수 $q_\varphi(z \mid x)$ 를 구현할 수 있다 . 일부 예들에서, 학습가능한 함수는 디코더(514)와 공유되거나 디코더(514)에 의해 알려질 필요가 없다.

[0117] 산술 인코더(508)는 잠재 코드(504)(잠재 z_i) 및 잠재 프리어(506)에 기초하여 비트스트림(510)을 생성할 수 있다. 일부 예들에서, 잠재 프리어(506)는 학습 가능한 함수를 구현할 수 있다. 일부 경우들에서, 잠재 프리어(506)는 ψ 에 의해 파라미터화된 학습 가능한 함수를 구현할 수 있다. 예를 들어, 잠재 프리어(506)는 함수 $p_\psi(z)$ 를 구현할 수 있다. 잠재 프리어(506)는 무손실 압축을 사용하여 잠재 코드(504)(잠재 z_i)를 비트스트림(510)으로 변환하는 데 사용될 수 있다. 잠재 프리어(506)는 전송자 측(예를 들어, 인코더(502) 및/또는 산술 인코더(508)) 및 수신기 측(예를 들어, 산술 디코더(512) 및/또는 디코더(514)) 양자 모두에서 공유되고 및/또는 이용가능하게 될 수 있다.

[0118] 디코더(514)는 산술 인코더(508)로부터 인코딩된 비트스트림(510)을 수신할 수 있고, 인코딩된 비트스트림(510) 내의 잠재 코드(504)(잠재 z_i)를 디코딩하기 위해 잠재 프리어(506)를 사용할 수 있다. 디코더(514)는 잠재 코드(504)(잠재 z_i)를 근사 재구성 이미지(516)(재구성 $\hat{x}_i$)로 디코딩할 수 있다. 일부 경우들에서, 디코더(514)는 θ 에 의해 파라미터화된 학습 가능한 함수를 구현할 수 있다. 예를 들어, 디코더(514)는 함수 $p_\theta(x \mid z)$ 를 구현할 수 있다. 디코더(514)에 의해 구현되는 학습 가능한 함수는 전송자 측(예를 들어, 인코더(502) 및/또는 산술 인코더(508)) 및 수신기 측(예를 들어, 산술 디코더(512) 및/또는 디코더(514)) 양자 모두에서 공유 및/또는 이용 가능하게 될 수 있다.

[0119] 신경망 압축 시스템(500)은 레이트-왜곡을 최소화하도록 트레이닝될 수 있다. 일부 예들에서, 레이트는 비트스트림(510)(비트스트림 b)의 길이를 반영하고, 왜곡은 이미지(501)(이미지 x_i)와 재구성 이미지(516)(재구성 $\hat{x}_i$) 사이의 왜곡을 반영한다. 파라미터 β 는 특정 레이트-왜곡 비율에 대한 모델을 트레이닝시키는 데 사용될 수 있다. 일부 예들에서, 파라미터 β 는 레이트와 왜곡 사이의 특정 트레이드오프를 정의 및/또는 구현하는 데 사용될 수 있다.

[0120] 일부 예들에서, 손실은 $L_{RD}(\mathrm{x};\ \varphi,\psi,\theta) = E_{q_\varphi(z\,|x)}[-\log p_\theta(x|z) - \beta\ log p_\psi(\mathrm{z})],$ 로서 표시될 수 있고, 여기서 함수 E 는 기대치이다. 왜곡 $(x|z;\theta)$ 은 예를 들어, 평균 제곱 오차(mean squared error; MSE)와 같은 손실 함수에 기초하여 결정될 수 있다. 일부 예들에서, 항들 $-\log p_\theta(x|z)$ 은 왜곡 $D(x|z;\theta)$ 을 표시 및/또는 표현할 수 있다.

[0121] 잠재들을 전송하기 위한 레이트는 $R_z(z;\psi)$ 로서 표시될 수 있다. 일부 예들에서, 항들 $log p_\psi(z)$ 은 레이트 $R_z(z;\psi)$ 를 표시 및/또는 표현할 수 있다. 일부 경우들에서, 손실은 $\varphi_D^*,\psi_D^*,\theta_D^* = argmin\ E_{x\sim D}[L_{RD}(\mathrm{x};\varphi,\psi,\theta)]$ 로서 전체 데이터세트 D에 걸쳐 최소화될 수 있다:

[0122] 도 5b는 신경망 압축 시스템(500)에 의해 수행되는 추론 프로세스(530)를 도시한 도면이다. 도시된 바와 같이, 인코더(502)는 이미지(501)를 잠재 코드(504)로 변환할 수 있다. 일부 예들에서, 이미지(501)는 프레임들의 시퀀스(예를 들어, 비디오)와 연관된 정지 이미지 및/또는 비디오 프레임을 나타낼 수 있다.

[0123] 일부 예들에서, 인코더(502)는 단일 순방향 패스 $\mathbf{z} = \mathrm{q}_{\varphi_D^*}(\mathbf{z}|\mathbf{x})$ 를 사용하여 이미지(501)를 인코딩할 수 있다.

산술 인코더(508)는 그 후 잠재 프리어(506) 하의 잠재 코드(504)(잠재 z_i)의 산술 코딩을 수행하여 비트스트림(520)(b_z^i)을 생성할 수 있다. 일부 예들에서, 산술 인코더(508)는 $b_z^i = \mathrm{AE}\left(z|p_{\psi_D^*}(z)\right)$ 와 같이 비트스트림(520)을 생성할 수 있다:

[0124] 산술 디코더(512)는 산술 인코더(508)로부터 비트스트림(520)을 수신하고 잠재 프리어(506) 하의 잠재 코드(504)(잠재 z_i)의 산술 디코딩을 수행할 수 있다. 일부 예들에서, 산술 디코더(512)는 $z = \mathrm{AD}\left(b_z^i|p_{\psi_D^*}(z)\right)$ 와 같이 비트스트림(520)으로부터 잠재 코드(504)를 디코딩할 수 있다: 디코더(514)는 잠재 코드(504)(잠재 z_i)를 디코딩하고 재구성 이미지(516)(재구성 $\hat{x}_i$)를 생성할 수 있다. 일부 예들에서, 디코더(514)는 $\hat{\mathbf{x}} = \mathrm{p}_{\theta_D^*}(\mathbf{x}|\mathbf{z})$ 와 같이 단일 순방향 패스를 사용하여 잠재 코드(504)(잠재 z_i)를 디코딩할 수 있다:

[0125] 일부 예에서, RD-AE 시스템은 트레이닝 데이터의 세트를 사용하여 트레이닝될 수 있고, 수신기(예를 들어, 디코더)에 의해 전송되고 디코딩될 데이터포인트(예를 들어, 이미지 데이터, 비디오 데이터, 오디오 데이터)에 대해 추가로 미세 튜닝될 수 있다. 예를 들어, 추론 시간에, RD-AE 시스템은 수신기로 전송되는 이미지 데이터에 대해 미세 튜닝될 수 있다. 압축 모델들이 일반적으로 크기 때문에, 모델과 연관된 파라미터들을 수신기에 전송하는 것은 네트워크(예를 들어, 대역폭 등), 저장 및 계산 자원과 같은 자원의 관점에서 매우 비용이 많이 들 수 있다. 일부 경우들에서, RD-AE 시스템은 압축되고 압축해제를 위해 수신기에 전송되는 단일 데이터포인트 상에서 미세-튜닝될 수 있다. 이는 압축/압축해제 효율, 성능, 및/또는 품질을 유지 및/또는 증가시키면서, 수신기로 전송되는 정보의 양(및 연관된 비용)을 제한할 수 있다.

[0126] 도 6은 인스턴스 적응적 데이터 압축(instance-adaptive data compression)을 위한 예시적인 신경망 압축 시스템(600)을 나타내는 도면이다. 신경망 압축 시스템(600)은 압축되고 있는 데이터에 적응된/미세-튜닝된 압축을 제공하기 위해 압축되고 있는 데이터에 대해 트레이닝되고 추가로 미세-튜닝될 수 있다(예를 들어, 인스턴스 적응됨). 이 예에서, 신경망 압축 시스템(600)은 가변 오토인코더(VAE) 프레임워크를 사용하여 평균-스케일 하이퍼프리어 모델 아키텍처(mean-scale hyperprior model architecture)를 구현하는 것으로 도시된다. 일부 경우들에서, 공유 하이퍼디코더는 평균-스케일 하이퍼프리어 모델에 대한 평균 및 스케일 파라미터들을 예측하는데 사용될 수 있다.

[0127] 도 6에 도시된 바와 같이, 신경망 압축 시스템(600)은 트레이닝 데이터세트(602)를 사용하여 트레이닝될 수 있다. 트레이닝 데이터세트(602)는 트레이닝 데이터세트(602)의 잠재 공간 표현(608)(z_2)을 생성하기 위해 코덱(604)의 인코더(606)에 의해 프로세싱될 수 있다. 인코더(606)는 잠재 공간 표현(608)(z_2)을 코덱(604)의 디코더(610) 및 하이퍼코덱(612)의 하이퍼인코더(614)에 제공할 수 있다.

[0128] 하이퍼인코더(614)는 트레이닝 데이터세트(602)의 하이퍼잠재(hyperlatent) 공간 표현(616)(z_1)을 생성하기 위해 하이퍼프리어(620)의 잠재 공간 표현(608)(z_2) 및 잠재 프리어(622)를 사용할 수 있다. 일부 예들에서, 하이퍼잠재 공간 표현(616) (z_1) 및 하이퍼잠재 공간 표현(616) (z_1) 은 잠재 공간 $z = \{z_1, z_2\}$ 에 대한 계층적 잠재 변수 모델(hierarchical latent variable model)을 제공할 수 있다.

[0129] 하이퍼코덱(612)의 하이퍼디코더(618)는 하이퍼잠재 공간 표현(616)(z_1)을 사용하여 하이퍼프리어 모델(624)을 생성할 수 있다. 하이퍼디코더(618)는 하이퍼프리어 모델(624)에 대한 평균 및 스케일 파라미터들을 예측할 수 있다. 일부 예들에서, 하이퍼프리어 모델(624)은 잠재 공간 표현(608)(z_2) 및 하이퍼잠재 공간 표현(616)(z_1)의 파라미터들에 대한 확률 분포를 포함할 수 있다. 일부 예들에서, 하이퍼프리어 모델(624)은 잠재 공간 표현(608)(z_2), 하이퍼잠재 공간 표현(616)(z_1), 및 하이퍼디코더(618)의 파라미터들에 걸친 확률 분포

를 포함할 수 있다.

[0130] 디코더(610)는 트레이닝 데이터세트(602)에 대한 재구성된 데이터(626)(재구성 $\hat{x}$)를 생성하기 위해 하이퍼프리어 모델(624) 및 잠재 공간 표현(608)(z_2)을 사용할 수 있다.

[0131] 일부 예들에서, 트레이닝 동안, 신경망 압축 시스템(600)은 혼합된 양자화 전략을 구현할 수 있으며, 여기서 양자화된 잠재 공간 표현(608)(z_2)은 왜곡 손실을 계산하기 위해 사용되고, 잠재 공간 표현(608)(z_2) 및 하이퍼잠재 공간 표현(616)(z_1)에 대한 노이즈성 샘플들은 레이트 손실을 계산할 때 사용된다.

[0132] 도 7은 모델 프리어를 사용하여 미세 튜닝된(예를 들어, 인스턴스 적응된) 신경망 압축 시스템(700)의 예시적인 아키텍처를 나타내는 도면이다. 일부 예들에서, 신경망 압축 시스템(700)은 RDM-AE 모델 프리어를 사용하여 미세 튜닝된 RD-AE 시스템을 포함할 수 있다. 신경망 압축 시스템(700)은 인코더(702), 산술 인코더(706), 산술 디코더(716), 및 디코더(718)를 포함할 수 있다. 일부 경우들에서, 인코더(702)는 인코더(402), 인코더(502) 또는 인코더(606)와 동일하거나 상이할 수 있고, 디코더(718)는 디코더(403), 디코더(514) 또는 디코더(610)와 동일하거나 상이할 수 있다. 산술 인코더(706)는 산술 코더(406) 또는 산술 인코더(508)와 동일하거나 상이할 수 있고, 산술 디코더(716)는 산술 디코더(426 또는 508)와 동일하거나 상이할 수 있다.

[0133] 인코더(702)는 입력으로서 이미지(701)(이미지 x_i)를 수신할 수 있고, 이미지(701)(이미지 x_i)를 잠재 코드 공간 내의 잠재 코드(704)(잠재 z_i)에 맵핑 및/또는 변환할 수 있다. 일부 예들에서, 이미지(701)는 프레임들의 시퀀스(예를 들어, 비디오)와 연관된 정지 이미지 및/또는 비디오 프레임을 나타낼 수 있다. 일부 예들에서, 이미지(701)를 프로세싱하기 이전에, 인코더(702)는 트레이닝 데이터세트(예를 들어, 도 6의 트레이닝 데이터세트(602)) 상에서 트레이닝될 수 있다. 또한, 인코더(702)는 이미지(701)에 대해 추가로 트레이닝되거나 미세 튜닝(예를 들어, 인스턴스 적응)될 수 있다. 예를 들어, 이미지(701)는 이미지(701)에 대해 인코더(702)를 미세-튜닝하는데 사용될 수 있다. 이미지(701)에 대한 인코더(702)의 미세-튜닝은 이미지(701)에 대한 높은 압축 성능을 초래할 수 있다. 예를 들어, 인코더(702)의 미세-튜닝은 신경망 압축 시스템(700)이 압축된 이미지(701)의 레이트-왜곡을 개선하게 할 수 있다.

[0134] 일부 경우들에서, 인코더(702)는 단일 순방향 패스를 사용하여 잠재 코드(704)를 생성할 수 있다. 일부 예들에서, 인코더(702)는 학습가능한 함수를 구현할 수 있다. 일부 경우들에서, 학습 가능한 함수는 φ 에 의해 파라미터화될 수 있다. 예를 들어, 인코더(702)는 함수 $q_\varphi(z \mid x)$ 를 구현할 수 있다 . 일부 예들에서, 학습가능한 함수는 산술 디코더(716) 및/또는 디코더(718)와 공유되거나 그에 의해 알려질 필요가 없다.

[0135] 산술 인코더(706)는 잠재 코드(704)를 엔트로피 코딩하고 산술 디코더(716)에 송신될 비트스트림(710) (예를 들어, 비트스트림 b_z^i) 을 생성하기 위해 잠재 프리어(708)를 사용할 수 있다. 비트스트림(710)은 잠재 코드(704)를 나타내는 압축된 데이터를 포함할 수 있다. 잠재 프리어(708)는 무손실 압축을 사용하여 잠재 코드(704)(잠재 z_i)를 비트스트림(710)으로 변환하는 데 사용될 수 있다. 잠재 프리어(708)는 전송자 측(예를 들어, 인코더(702) 및/또는 산술 인코더(706)) 및 수신기 측(예를 들어, 산술 디코더(716) 및/또는 디코더(718)) 양자 모두에서 공유되고 및/또는 이용가능하게 될 수 있다. 일부 예들에서, 잠재 프리어(708)는 학습 가능한 함수를 구현할 수 있다. 일부 경우들에서, 학습 가능한 함수는 ψ 에 의해 파라미터화된다. 예를 들어, 잠재 프리어(708)는 함수 $p_\psi(z)$ 를 구현할 수 있다.

[0136] 신경망 압축 시스템(700)은 또한 모델 프리어(714)를 포함할 수 있다. 모델 프리어(712)는 잠재 프리어(708) 및 디코더(718)의 파라미터들에 대한 확률 분포를 포함할 수 있다. 일부 예들에서, 모델 프리어(714)는 학습 가능한 함수를 구현할 수 있다. 일부 경우들에서, 학습 가능한 함수는 ω 에 의해 파라미터화된다. 예를 들어, 모델 프리어(712)는 함수 $p_\omega(\Psi \mid \Theta)$ 를 구현할 수 있다 . 신경망 압축 시

스템(700)은 인코더(702) 및 잠재 프리어(708)의 미세-튜닝된 파라미터들을 산술 디코더(716)에 송신될 비트스트림(712)(예를 들어, 비트스트림 $b^{i}_{\psi,\theta}$)으로 변환하기 위해 모델 프리어(714)를 사용할 수 있다.

[0137] 모델 프리어(712)는 전송자 측(예를 들어, 인코더(702) 및/또는 산술 인코더(706)) 및 수신기 측(예를 들어, 산술 디코더(716) 및/또는 디코더(718)) 양자 모두에서 공유되고 및/또는 이용가능하게 될 수 있다. 예를 들어, 산술 인코더(706)는 잠재 코드(704)로부터 생성된 비트스트림(710)을 디코딩하는데 사용하기 위해 비트스트림(712)을 산술 디코더(716)에 송신할 수 있다. 비트스트림(712)은 인코더(702) 및 잠재 프리어(708)의 미세-튜닝된 파라미터들을 나타내는 압축된 데이터를 포함할 수 있으며, 산술 디코더(716)는 인코더(702) 및 잠재 프리어(708)의 미세-튜닝된 파라미터들을 획득하기 위해 사용할 수 있다. 산술 디코더(716) 및 디코더(718)는 비트스트림(710)으로부터 획득된 잠재 코드(704) 및 비트스트림(712)으로부터 획득된 잠재 프리어(708) 및 인코더(702)의 미세-튜닝된 파라미터들에 기초하여 이미지(701)를 재구성할 수 있다.

[0138] 산술 디코더(716)는 잠재 프리어(708) 및 모델 프리어(714)를 사용하여 비트스트림(710)을 잠재 코드(704)(잠재 z_i)로 변환할 수 있다. 예를 들어, 산술 디코더(716)는 비트스트림(710)을 디코딩하기 위해 잠재 프리어(708) 및 모델 프리어(714)를 사용할 수 있다. 디코더(718)는 산술 디코더(716)에 의해 디코딩된 잠재 코드(704)(잠재 z_i)를 사용하여 재구성된 이미지(720)(재구성 $\hat{x}_i$)를 생성할 수 있다. 예를 들어, 디코더(718)는 잠재 코드(704)(잠재 z_i)를 근사한 재구성 이미지(720)(재구성 $\hat{x}_i$)로 디코딩할 수 있다. 일부 예들에서, 디코더(718)는 잠재 코드(704), 잠재 프리어(708) 및/또는 모델 프리어(714)를 사용하여 재구성된 이미지(720)를 생성할 수 있다.

[0139] 일부 경우들에서, 디코더(718)는 θ 에 의해 파라미터화된 학습 가능한 함수를 구현할 수 있다. 예를 들어, 디코더(718)는 학습 가능한 함수 $p_\theta(x \mid z)$ 를 구현할 수 있다. 디코더(718)에 의해 구현되는 학습 가능한 함수는 전송자 측(예를 들어, 인코더(702) 및/또는 산술 인코더(706)) 및 수신기 측(예를 들어, 산술 디코더(716) 및/또는 디코더(718)) 양자 모두에서 공유 및/또는 이용 가능하게 될 수 있다.

[0140] 일부 예들에서, 인코더 측에서, 모델은 $\varphi_D^*, \psi_D^*, \theta_D^* = argmin\ L_{RD}(\mathrm{x};\varphi,\psi,\theta,\omega)$ 와 같이 레이트-왜곡-모델(RDM) 손실을 사용하여 이미지 (701) (이미지 x_i)에 대해 미세-튜닝될 수 있다: . 일부 예들에서, 이미지(701)(이미지 x_i)는 $\mathrm{z} = \mathrm{q}_{\varphi_x^*}(\mathrm{z}|\mathrm{x})$ 와 같이 미세-튜닝된 인코더(702)의 단일 순방향 패스를 사용하여 인코딩될 수 있다. 일부 경우들에서, 미세-튜닝된 잠재 프리어(708)는 $b^i_\psi = \mathrm{AE}(\psi_x^*|p_\omega(\psi)),$ 와 같이 엔트로피-코딩될 수 있고, 미세-튜닝된 디코더(718) 및/또는 산술 인코더(706)는 $b^i_\theta = \mathrm{AE}(\theta_x^*|p_\omega(\theta))$ 와 같이 엔트로피-코딩될 수 있다. 일부 경우들에서, 잠재 코드(704)(잠재 z_i)는 $b^i_z = \mathrm{AE}\left(z|p_{\psi_x^*}(z)\right)$ 와 같이 엔트로피 코딩될 수 있다.

[0141] 디코더 측에서, 일부 예들에서, 미세-튜닝된 잠재 프리어(708)는 $\psi_x^* = \mathrm{AD}\left(b^i_\psi|p_\omega(\psi)\right)$ 와 같이 엔트로피-코딩될 수 있고, 미세-튜닝된 디코더(718) 및/또는 산술 디코더(716)는 $\theta_x^* = \mathrm{AD}\left(b^i_\theta|p_\omega(\theta)\ \right)$ 와 같이 엔트로피-코딩될 수 있다. 일부 경우들에서, 잠재 코드(704)(잠재 z_i)는 또한 $z = \mathrm{AD}\left(b^i_z|p_{\psi_x^*}(z)\right)$ 와 같이 엔트로피 코딩될 수 있다.

[0142] 일부 예들에서, 디코더(718)는

[0143] $\hat{x} = p_{\theta_x^*}(x|z)$ 와 같이 미세 튜닝된 디코더(예를 들어, 디코더(718))의 단일 순방향 패스를 사용하여 잠재 코드

(704)(잠재 z_i)를 근사 재구성 이미지(720)(재구성 $\hat{x}_i$)로 디코딩할 수 있다.

[0144] 도 8은 모델 프리어를 사용하여 미세 튜닝된 예시적인 신경망 압축 시스템(800)에 의해 구현되는 예시적인 추론 프로세스를 예시하는 도면이다. 일부 예들에서, 신경망 압축 시스템(800)은 RDM-AE 모델 프리어를 사용하여 미세 튜닝된 RD-AE 시스템을 포함할 수 있다. 일부 경우들에서, 신경망 압축 시스템(800)은 모델 프리어를 사용하여 미세 튜닝된 AE 모델을 포함할 수 있다.

[0145] 이 예시적인 예에서, 신경망 압축 시스템(800)은 인코더(802), 산술 인코더(808), 산술 디코더(812), 디코더(814), 모델 프리어(816) 및 잠재 프리어(806)를 포함한다. 일부 경우들에서, 인코더(802)는 인코더(402), 인코더(502), 인코더(606) 또는 인코더(702)와 동일하거나 상이할 수 있고, 디코더(814)는 디코더(403), 디코더(514), 디코더(610) 또는 디코더(718)와 동일하거나 상이할 수 있다. 산술 인코더(808)는 산술 코더(406), 산술 인코더(508), 또는 산술 인코더(706)와 동일하거나 상이할 수 있고, 산술 디코더(812)는 산술 디코더(426), 산술 디코더(508), 또는 산술 디코더(716)와 동일하거나 상이할 수 있다.

[0146] 신경망 압축 시스템(800)은 이미지(801)에 대한 잠재 코드(804)(잠재 z_i)를 생성할 수 있다. 신경망 압축 시스템(800)은 잠재 코드(804) 및 잠재 프리어(806)를 사용하여 이미지(801)(이미지 x_i)를 인코딩하고, 수신기에 의해 사용되어 재구성 이미지(820)(재구성 $\hat{x}_i$)를 생성할 수 있는 비트스트림(810)을 생성할 수 있다. 일부 예들에서, 이미지(801)는 프레임들의 시퀀스(예를 들어, 비디오)와 연관된 정지 이미지 및/또는 비디오 프레임을 나타낼 수 있다.

[0147] 일부 예들에서, 신경망 압축 시스템(800)은 RDM-AE 손실을 사용하여 미세 튜닝될 수 있다. 신경망 압축 시스템(800)은 레이트-왜곡-모델 레이트(rate-distortion-model rate; RDM) 손실을 최소화함으로써 트레이닝될 수 있다. 일부 예들에서, 인코더 측에서, AE 모델은 $\varphi_D^*, \psi_D^*, \theta_D^* = argmin\ L_{RD}(\mathrm{x};\varphi,\psi,\theta,\omega)$ 와 같이 RDM 손실을 사용하여 이미지 (801) (이미지 x_i)에 대해 미세-튜닝될 수 있다:

[0148] 미세-튜닝된 인코더(802)는 잠재 코드(804)를 생성하기 위해 이미지(801)(이미지 x_i)를 인코딩할 수 있다. 일부 경우들에서, 미세-튜닝된 인코더(802)는 $\mathrm{z} = \mathrm{q}_{\varphi_x^*}(\mathrm{z}|\mathrm{x})$ 와 같이 단일 순방향 패스를 사용하여 이미지(801)(이미지 x_i)를 인코딩할 수 있다. 산술 인코더(808)는 잠재 프리어(806)를 사용하여 잠재 코드(804)를 산술 디코더(812)에 대한 비트스트림(810)으로 변환할 수 있다. 산술 인코더(808)는 모델 프리어(816) 하에서 미세-튜닝된 디코더(814) 및 미세-튜닝된 잠재 프리어(806)의 파라미터들을 엔트로피-코딩할 수 있고, 미세-튜닝된 디코더(814) 및 미세-튜닝된 잠재 프리어(806)의 압축된 파라미터들을 포함하는 비트스트림(811)을 생성할 수 있다. 일부 예들에서, 비트스트림(811)은 미세-튜닝된 디코더(814) 및 미세-튜닝된 잠재 프리어(806)의 업데이트된 파라미터들을 포함할 수 있다. 업데이트된 파라미터들은, 예를 들어, 미세-튜닝 이전의 디코더(814) 및 잠재 프리어(806)와 같은, 베이스라인 디코더 및 잠재 프리어에 대한 파라미터 업데이트들을 포함할 수 있다.

[0149] 일부 경우들에서, 미세-튜닝된 잠재 프리어(806)는 $b_\psi^i = \mathrm{AE}(\psi_x^*|p_\omega(\psi))$ 와 같이 모델 프리어(816) 하에서 엔트로피-코딩될 수 있고, 미세-튜닝된 디코더(814)는 $b_\theta^i = \mathrm{AE}(\theta_x^*|p_\omega(\theta))$ 와 같이 모델 프리어(816) 하에서 엔트로피-코딩될 수 있으며, 잠재 코드(804)(잠재 z_i)는 $b_z^i = \mathrm{AE}\left(z|p_{\psi_x^*}(z)\right)$ 와 같이 미세-튜닝된 잠재 프리어(806) 하에서 엔트로피-코딩될 수 있다. 일부 경우들에서, 디코더 측에서, 미세-튜닝된 잠재 프리어(806)는 $\psi_x^* = \mathrm{AD}\left(b_\psi^i|p_\omega(\psi)\right)$ 와 같이 모델 프리어(816) 하에서 엔트로피-코딩될 수 있고, 미세-튜닝된 디코더(814)는 $\theta_x^* = \mathrm{AD}\left(b_\theta^i|p_\omega(\theta)\ \right)$ 와 같이 모델 프리어(816) 하에서 엔트로피-코딩될 수 있으며, 잠재 코드(804)(잠재 z_i)

는 $z = \mathrm{AD}(b_z^i | p_{\psi_x^*}(z))$ 와 같이 미세-튜닝된 잠재 프리어(806) 하에서 엔트로피-코딩될 수 있다.

[0150] 디코더(814)는 잠재 코드(804)(잠재 z_i)를 근사 재구성 이미지(820)(재구성 $\hat{x}_i$)로 디코딩할 수 있다. 일부 예들에서, 디코더(814)는 $\hat{x} = p_{\theta_x^*}(x|z)$ 와 같이 미세-튜닝된 디코더의 단일 순방향 패스를 사용하여 잠재 코드(804)를 디코딩할 수 있다.

[0151] 이전에 설명된 바와 같이, 신경망 압축 시스템(800)은 RDM 손실을 최소화함으로써 트레이닝될 수 있다. 일부 경우들에서, 레이트는 비트스트림 b(예를 들어, 비트스트림(810 및/또는 811))의 길이를 반영할 수 있고, 왜곡은 입력 이미지(801)(이미지 x_i)와 재구성 이미지(820)(재구성 $\hat{x}_i$) 사이의 왜곡을 반영할 수 있고, 모델 레이트는 모델 업데이트들(예를 들어, 업데이트된 파라미터들)을 수신기에(예를 들어, 디코더(814)에) 전송하는 데 사용되고 그리고/또는 필요한 비트스트림의 길이를 반영할 수 있다. 파라미터 β 는 특정 레이트-왜곡 비율에 대한 모델을 트레이닝시키는 데 사용될 수 있다.

[0152] 일부 예들에서, 데이터포인트 x 에 대한 손실은 $\varphi_x^*, \psi_x^*, \theta_x^* = \operatorname{argmin} \mathcal{L}_{\mathrm{RDM}}(x; \varphi, \psi, \theta, \omega)$ 와 같이 추론 시간에서 최소화될 수 있다: 일부 예들에서, RDM 손실은 $L\_RDM\ (\mathrm{x};\ \varphi, \psi, \theta, \omega) = E_{q_\varphi} q\_(z\ |x)\ [-\log p_\theta\ (x\ |z) - \beta \log p_\psi\ (\mathrm{z}) - \beta \log p_\omega\ (\psi, \theta)]$ 와 같이 표시될 수 있다: 일부 경우들에서, 왜곡 $D(x|z;\theta)$ 은 예를 들어, 평균 제곱 오차(mean squared error; MSE)와 같은 손실 함수에 기초하여 결정될 수 있다.

[0153] 용어들 $-\log p_\theta(x|z)$ 은 왜곡 $D(x|z;\theta)$ 을 표시 및/또는 표현할 수 있다. 용어들 $\beta \log p_\psi\ (\mathrm{z})$ 은 잠재들 $R_z(z;\psi)$ 을 전송하기 위한 레이트를 표시 및/또는 표현할 수 있고, 용어들 $\beta \log p_\omega\ (\psi, \theta)$ 은 미세-튜닝된 모델 업데이트들 $R_{\psi,\theta}(\psi, \theta; \omega)$ 을 전송하기 위한 레이트를 표시 및/또는 표현할 수 있다.

[0154] 일부 경우들에서, 모델 프리어(816)는 모델 업데이트들을 전송하기 위한 비트레이트 오버헤드의 길이를 반영할 수 있다. 일부 예들에서, 모델 업데이트들을 전송하기 위한 비트레이트는 $|b_{\psi,\theta}^i| = R_{\psi,\theta}(\psi, \theta; \omega) = -\log p_\omega(\psi, \theta)$ 와 같이 설명될 수 있다. 일부 경우들에서, 모델 프리어는 업데이트들 없이 모델을 전송하는 것이 저렴하도록, 즉, 비트길이(모델-레이트-손실)가 작도록 선택될 수 있다: $R_{\psi,\theta}(\psi_D^*, \theta_D^*; \omega)$.

[0155] 일부 경우들에서, RDM 손실 함수를 사용하여, 신경망 압축 시스템 (800) 은 잠재 레이트 또는 왜곡이 적어도 많은 비트들로 감소한다면 모델 업데이트들 $b_{\psi,\theta}^i$ 에 대해 비트스트림에 비트들을 오직 추가할 수도 있다. 이것은 부스트 대 레이트-왜곡(R/D) 성능을 제공할 수도 있다. 예를 들어, 신경망 압축 시스템(800)은 모델 업데이트들을 전송하기 위해 비트스트림(811) 내의 비트들의 수를 증가시킬 수도 있는데, 이는 또한 적어도 동일한 수의 비트들로 레이트 또는 왜곡을 감소시킬 수 있는 경우에 그러하다. 다른 경우들에서, 신경망 압축 시스템(800)은 잠재 레이트 또는 왜곡이 적어도 많은 비트들로 감소하지 않더라도 모델 업데이트들 $b_{\psi,\theta}^i$ 을 위해 비트스트림에 비트들을 추가할 수도 있다.

[0156] 신경망 압축 시스템(800)은 엔드-투-엔드 (end-to-end) 트레이닝될 수 있다. 일부 경우들에서, RDM 손실은 추론 시간 엔드-투-엔드(inference time end-to-end)에서 최소화될 수 있다. 일부 예들에서, 특정 양의 계산이 한번 소비될 수 있고(예를 들어, 모델을 미세-튜닝함), 높은 압축비들이 후속적으로 수신기 측에 대한 가외의 비용 없이 획득될 수 있다. 예를 들어, 콘텐츠 제공자는 많은 수의 수신기들에 제공될 비디오에 대해 신경망 압축 시스템(800)을 더 광범위하게 트레이닝하고 미세-튜닝하기 위해 많은 양의 계산을 소비할 수도 있

다. 고도로 트레이닝되고 미세 튜닝된 신경망 압축 시스템(800)은 그 비디오에 대한 높은 압축 성능을 제공할 수 있다. 높은 양의 계산을 소비한 경우, 비디오 제공자는 모델 프리어의 업데이트된 파라미터들을 저장하고, 비디오를 압축해제하도록 압축된 비디오의 각각의 수신기에 효율적으로 제공할 수 있다. 비디오 제공자는 모델을 트레이닝하고 미세-튜닝하는 초기 컴퓨팅 비용을 상당히 능가할 수 있는 비디오의 각각의 전송으로 압축 (및 네트워크 및 컴퓨팅 자원의 감소) 에서 큰 이점을 달성할 수 있다.

[0157] 비디오 및 고해상도 이미지들에서의 많은 수의 픽셀들로 인해, 본 명세서에서의 학습 및 미세-튜닝 접근법들은 비디오 압축 및/또는 고해상도 이미지들에 매우 유리할 수 있다. 일부 경우들에서, 복잡도 및/또는 디코더 컴퓨트는 전체 시스템 설계 및/또는 구현에 대한 추가된 고려사항들로서 사용될 수 있다. 예를 들어, 추론을 행하는데 빠른 매우 작은 네트워크들이 미세 튜닝될 수 있다. 다른 예로서, 모델이 하나 이상의 계층들을 제거하도록 강제하고 그리고/또는 야기할 수 있는, 수신기 복잡도에 대한 비용 항이 추가될 수 있다. 일부 예들에서, 훨씬 더 큰 이득들을 달성하기 위해 머신 러닝을 사용하여 더 복잡한 모델 프리어들이 학습될 수 있다.

[0158] 모델 프리어 설계는 다양한 속성들을 포함할 수 있다. 일부 예들에서, 구현된 모델 프리어는 임의의 업데이트들 없이 모델을 전송하기 위한 높은 확률 $p_{\omega}(\psi_D^*, \theta_D^*)$ 을 할당하는 모델 프리어, 및 따라서 낮은 비트레이트 $R_{\psi,\theta}(\psi_D^*, \theta_D^*; \omega)$ 를 포함할 수 있다. 일부 경우들에서, 모델 프리어는, 미세-튜닝된 모델들의 상이한 인스턴스들이 실제로 인코딩될 수 있도록 ψ_D^*, θ_D^* 주변의 값들에 0이 아닌 확률을 할당하는 모델 프리어를 포함할 수 있다. 일부 경우들에서, 모델 프리어는 추론 시간에서 양자화되고 엔트로피 코딩을 수행하는 데 사용될 수 있는 모델 프리어를 포함할 수 있다.

[0159] 도 9는 모델 프리어를 사용하여 미세 튜닝된 예시적인 신경망 압축 시스템(900)에 의해 수행되는 인코딩 및 디코딩 태스크들을 예시하는 도면이다. 이 예에서, 인코더(904)는 이미지(902)를 수신하고 이미지(902)를 잠재 공간(906)(z)으로 압축한다. 산술 인코더(908)는 양자화된 모델 파라미터들($\bar{\theta}$)을 계산하기 위해, 잠재 프리어(910), 디코더(922)의 파라미터들, 및 트레이닝 세트(D)에 미리 트레이닝된 글로벌 모델 파라미터들(914)(θ_D)을 사용할 수 있다.

[0160] 일부 예들에서, 양자화된 모델 파라미터들($\bar{\theta}$)은 트레이닝 세트 D에 대해 미리 트레이닝된 글로벌 모델 파라미터들(914)(θ_D) 및 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)의 합을 나타낼 수 있다. 일부 예들에서, 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)은 트레이닝 세트 D에 대해 미리 트레이닝된 글로벌 모델 파라미터들(914)(θ_D)과 모델 파라미터 업데이트들(θ) 사이의 차이에 기초하여 양자화기(Q_t)에 의해 생성될 수 있다.

[0161] 산술 인코더(908)는 모델 파라미터 업데이트들을 수신기(예를 들어, 산술 디코더(920))에 시그널링하기 위한 비트스트림(918)을 생성하기 위해 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)을 엔트로피 인코딩할 수 있다. 산술 인코더(908)는 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)을 엔트로피 인코딩하기 위해 양자화된 모델 프리어(912)($p[\bar{\delta}]$)를 사용할 수 있다. 일부 예들에서, 산술 인코더(908)는 모델 파라미터 업데이트들 (δ)을 정규화하기 위해 연속 모델 프리어($p(\delta)$)를 구현할 수 있고, 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)을 엔트로피 인코딩하기 위해 양자화된 모델 프리어(912)($p[\bar{\delta}]$)를 사용할 수 있다.

[0162] 산술 인코더(908)는 또한 잠재 공간(906)(z)을 수신기(예를 들어, 산술 디코더(920))에 시그널링하기 위한 비트스트림(916)을 생성하기 위해 잠재 공간(906)(z)을 엔트로피 인코딩할 수 있다. 일부 예들에서, 산술 인코더(908)는 비트스트림(916) 및 비트스트림(918)을 연접(concatenate)시키고 연접된 비트스트림을

수신기(예컨대, 산술 디코더(920))에 송신할 수 있다.

[0163] 산술 디코더(920)는 비트스트림(918)을 엔트로피 디코딩하고 양자화된 모델 파라미터 업데이트들($\bar{\delta}$)을 획득하기 위해 양자화된 모델 프리어(912)($p[\bar{\delta}]$)를 사용할 수 있다. 산술 디코더(920)는 비트스트림(916)으로부터 잠재 공간(906)(z)을 엔트로피 디코딩하기 위해 미세-튜닝된 잠재 프리어(910)를 사용할 수 있다. 디코더(922)는 잠재 공간(906) 및 모델 파라미터 업데이트들을 사용하여 재구성된 이미지(924)를 생성할 수 있다.

[0164] 일부 예들에서, 신경망 압축 시스템(900)은 인코딩 프로세스 동안 아래의 알고리즘 1, 및 디코딩 프로세스 동안 아래의 알고리즘 2를 구현할 수 있다.

표 1

[0165]

알고리즘 1 x의 인코딩

Input(입력): 트레이닝 세트 D, 모델 파라미터 양자화기 Q_t, 모델 프리어 $p[\bar{\delta}]$, 및 압축될 데이터포인트 x 에 대해 미리 트레이닝된 글로벌 모델 파라미터들 $\{\emptyset_D, \theta_D\}$.

Output(출력): 압축된 비트스트림 b.

1: 모델 파라미터들 초기화: $\emptyset = \emptyset_D$ 및 $\theta = \theta_D$

2: **for** step in MAX STEPS **do** (최대 단계들에서의 단계 동안 행함)

3: x_f ~ x 에서 하나 이상의 프레임들의 선택을 샘플링

4: 전송가능한 파라미터들 양자화: $\bar{\theta} \leftarrow Q_t(\delta) + \theta_D$; $\delta = \theta - \theta_D$

5: 순방향 패스: z ~ $q_\emptyset(z \mid x_f)$ 및 $p_\emptyset(x_f \mid z)$ 및 $p_\emptyset(z)$ 평가

6. 아래의 식 (2) 에 따라 x_f 에 대해 손실 $L_{RDM}(\emptyset, \delta)$ 을 계산,

7: Q_t 에 대해 STE를 사용하여 역전파한 다음 그래디언트들 $\frac{\partial L_{RDM}}{\partial \emptyset}$ 및 $\frac{\partial L_{RDM}}{\partial \theta}$ 을 사용하여 $\emptyset, \theta$ 업데이트

8: **end for (for 구문 종료)**

9: x 를 z ~ $q_\emptyset(x)$ 로 압축

10: 양자화된 모델 파라미터들 계산: $\bar{\theta} = \theta_D + \bar{\delta}$, 여기서, $\bar{\delta} = Q_t(\bar{\theta} - \theta_D)$

11: 엔트로피 인코딩: $b_{\bar{\delta}} = enc(\bar{\delta}; p[\bar{\delta}])$ 및 $b_z = enc(z; p_{\bar{\theta}}(z))$

12: 비트스트림들 연접: $b = (b_{\bar{\delta}}, b_z)$

알고리즘 2 x의 *디코딩*

Input(입력): 모델 프리어 $p[\bar{\delta}]$, 비트스트림 $b = (b_{\bar{\delta}}, b_z)$, 글로벌 모델 파라미터들 θ_D

Output(출력): 디코딩된 데이터 포인트 $\hat{x}$

1: 엔트로피 디코딩: $\bar{\delta} = dec(b_{\bar{\delta}}; p[\bar{\delta}])$

2: 업데이트된 모델 파라미터들 계산: $\bar{\theta} = \theta_D + \bar{\delta}$

3: 미세 튜닝된 프리어 하의 잠재를 엔트로피 디코딩: $z = dec(b_z; p_{\bar{\theta}}(z))$

4: 인스턴스를 미세 튜닝된 디코더의 평균으로서 디코딩: $p_{\bar{\theta}}(x \mid z)$

[0166] 일부 예들에서, 위에서 참조된 식 (2) 는 다음과 같이 RDM 손실을 계산하는데 사용될 수 있다:

[0167] $L_{RDM}(\emptyset, \delta) = L_{RD}(\emptyset, \theta_D + \delta) - \beta \log p(\delta)$ 식 (2)

[0168] 여기서, 용어 $\log p(\delta)$ 는 모델 업데이트 M을 나타내며, β 는 트레이드오프 파라미터이다.

[0169] 일부 예들에서, 모델 프리어(912)는 확률 밀도 함수(probability density function; PDF)로서 설계될 수 있다. 일부 경우들에서, 파라미터 업데이트 $\delta = \theta - \theta_D$ 는 제로-업데이트를 중심으로 하는 가우시안 분포를 사용하여 모델링될 수 있다. 일부 경우들에서, 모델 프리어는 제로 공분산을 갖는 다변량 제로-중심 가우시안, 및 $p(\theta) = \mathcal{N}(\theta | \theta_D, \sigma I)$ 에 의한 θ 모델링과 동등할 수 있는 표준 편차 $p(\delta) = \mathcal{N}(\delta | \mathbf{0}, \sigma I)$ 를 나타내는 단일 공유(하이퍼파라미터) σ로서 업데이트들에 대해 정의될 수 있다.

[0170] 일부 경우들에서, p[$\bar{\delta}$] 하에서 양자화된 업데이트들 $\bar{\delta}$ 을 엔트로피 코딩할 때, 심지어 제로 업데이트들이 자유롭지 않을 수도 있다. 일부 경우들에서, 이러한 초기 정적 비용은 $\overline{M_0} = -\log p[\delta = \mathbf{0}].$ 와 같이 정의될 수 있다. 정의된 모델 프리어의 모드가 제로이기 때문에, 이들 초기 비용들 $\overline{M_0}$ 은 최소 비용들과 동일할 수도 있다. 상기 식 (2)의 최소화는 이들 정적 비용들을 극복한 후, 모델 업데이트들에 소비되는 임의의 가외의 비트가 *RD* 성능의 상응하는 개선을 수반할 수 있음을 보장할 수 있다.

[0171] 일부 경우들에서, 모델 프리어 설계는 파라미터들 사이의 독립성이 $p_\omega(\psi, \theta) = \prod_{\theta^{(i)} \in \theta} p_\omega(\theta^{(i)}) \prod_{\psi^{(i)} \in \psi} p_\omega(\psi^{(i)})$ 와 같이 가정될 수 있는 독립 가우시안 네트워크 프리어에 기초할 수 있다. 본 명세서의 다양한 예들에서, 단일 파라미터 $p_\omega(\theta^{(i)})$ 에 대한 식들 및 예들이 제공된다. 그러나, 다른 예들은 다수의 파라미터들에 대한 식들을 수반할 수도 있다.

[0172] 일부 예들에서, 가우시안 모델 프리어는 $p_\omega(\theta^{(i)}) = \mathcal{N}(\theta_D^{*(i)}, \sigma)$ 와 같이 가변 표준 편차 σ 를 갖는 글로벌 모델 RDAE 모델을 중심으로 하는 가우시안을 포함할 수 있다:

[0173] 일부 경우들에서, 더 높은 σ (예를 들어, 튜닝되지 않은 모델을 전송하기 위한 더 높은 비용(초기 비용))과 더 낮은 σ (예를 들어, 미세-튜닝 모델들에 대한 더 제한된 확률 질량) 사이의 트레이드오프가 있을 수 있다.

[0174] 일부 예들에서, 모델 프리어 설계는 독립 라플라스 모델 프리어(Laplace model prior)에 기초할 수 있다. 일부 예들에서, 라플라스 모델 프리어는 $p_\omega(\theta^{(i)}) = L(\theta_D^{*(i)}, \sigma)$ 와 같이 가변 표준 편차 σ 를 갖는 글로벌 모델 RDAE 모델을 중심으로 하는 가우시안을 포함할 수 있다:

[0175] 일부 경우들에서, 더 높은 σ (예를 들어, 튜닝되지 않은 모델을 전송하기 위한 더 높은 비용(초기 비용))과 더 낮은 σ (예를 들어, 미세-튜닝 모델들에 대한 더 제한된 확률 질량) 사이의 트레이드오프가 있을 수 있다. 일부 예들에서, 모델은 파라미터 업데이트들의 희소성(sparsity)을 강제할 수 있다.

[0176] 일부 경우들에서, 모델 프리어 설계는 독립 스파이크 앤 슬래브 모델 프리어(Spike and Slab model prior)에 기초할 수 있다. 일부 예들에서, 스파이크 앤 슬래브 모델 프리어는 $p_\omega(\theta^{(i)}) = \frac{1}{c}\left(\mathcal{N}(\theta_D^{*(i)}, \sigma_{slab}) + \alpha \mathcal{N}(\theta_D^{*(i)}, \sigma_{spike})\right)$ 와 같이 슬래브 컴포넌트에 대한 가변 표준 편차 σ_{slab}, 스파이크 컴포넌트 $\sigma_{spike} \ll \sigma_{slab}$, 에 대한 가변 표준 편차 및 스파이크/슬래브 비 α 를 갖는 글로벌 모델 RDAE 모델 주위에 중심을 둔 가우시안을 포함할 수 있다.

[0177] 일부 예들에서, α 이 큰 경우 튜닝되지 않은 모델을 전송하기 위한 낮은 비용 (초기 비용), 및 미세-튜닝 모델들에 대한 광범위한 지원이 있을 수 있다.

[0178] 일부 예들에서, 모델 프리어 및 글로벌 AE 모델은 공동으로 트레이닝될 수 있다. 일부 경우들에서, 파라미터들 ω 을 수동으로 설정하는 대신에, 모델 프리어는 인스턴스 RDM-AE 모델들과 함께 엔드-투-엔드 트레이닝될 수 있다.

[0179] 전술한 바와 같이, 수신기로 전송되고 수신기에 의해 디코딩되는 데이터포인트에 대해 신경 압축 시스템을 미세-튜닝하는 것은 레이트 및/또는 왜곡 이점들 및 이익들을 제공할 수 있다. 도 10은 수신기로 송신될 데이터포인트(예를 들어, 이미지 또는 프레임, 비디오, 오디오 데이터 등)에 대해 미세-튜닝되는 예시적인 RD-AE 모델 및 수신기로 송신될 데이터포인트에 대해 미세-튜닝되지 않는 RD-AE 모델의 예시적인 레이트-왜곡들을 나타내는 그래프(1000)이다.

[0180] 이 예에서, 그래프(1000)는 본 명세서에 설명된 바와 같이 미세-튜닝되지 않은 RD-AE 모델의 레이트-왜곡(1002), 및 미세-튜닝된 RD-AE 모델의 레이트-왜곡(1004)을 예시한다. 레이트-왜곡 (1002) 및 레이트-왜곡(1004)에 의해 도시된 바와 같이, 미세-튜닝된 RD-AE 모델은 다른 RD-AE 모델에 비해 상당히 더 높은 압축 (레이트-왜곡) 성능을 갖는다.

[0181] 레이트-왜곡(1004)과 연관된 RD-AE 모델은 압축되어 수신기로 송신될 데이터포인트를 사용하여 미세 튜닝될 수 있다. 일부 예들에서, 압축 AE는 $\varphi_x^*, \psi_x^*, \theta_x^* = \operatorname{argmin} \mathcal{L}_{\mathrm{RD}}(x; \varphi, \psi, \theta)$ 와 같이 데이터포인트 x 에 대해 미세-튜닝될 수 있다: 일부 경우들에서, 이것은 데이터포인트 x에 대해 높은 압축(레이트-왜곡 또는 R/D) 성능을 허용할 수 있다. 미세-튜닝된 프리어 (예를 들어, ψ_x^*) 의 파라미터들 및 미세-튜닝된 디코더 (예를 들어, ψ_x^*, θ_x^*) 의 파라미터들은 비트스트림을 디코딩하는데 사용하기 위해 수신기와 공유될 수 있다. 수신기는 미세-튜닝된 RD-AE 시스템에 의해 생성된 비트스트림을 디코딩하기 위해 미세-튜닝된 프리어(예를 들어, ψ_x^*) 및 미세-튜닝된 디코더(예를 들어, ψ_x^*, θ_x^*)의 파라미터들을 사용할 수 있다.

[0182] 아래 테이블 1은 예시적인 튜닝 세부사항들을 나타낸다:

표 2

[0183] **테이블 1.**

		파라미터들의 수	비트 수(16비트)
모델 사이즈	프리어	800만	16MB
모델 사이즈	디코더	290만	5.8 MB
모델 사이즈	전송 사이즈	1090만	21.8 MB
이미지 사이즈		1600x1216픽셀	
미세 튜닝된 모델 파라미터들을 송신하는 레이트에서의 오버헤드		89 bpp(픽셀당 비트)	

[0184] 테이블 1에서의 튜닝 세부사항들은 단지 예시적인 예들이다. 당업자는 다른 예들이 더 많은, 더 적은, 동일한 및/또는 상이한 튜닝 세부사항들을 포함할 수 있다는 것을 인식할 것이다.

[0185] 도 11은 압축되고 있는 데이터를 입력하도록 적응된 신경망 압축 시스템(예를 들어, 신경망 압축 시스템(500), 신경망 압축 시스템(600), 신경망 압축 시스템(700), 신경망 압축 시스템(800), 신경망 압축 시스템(900))을 사용하는 인스턴스 적응적 압축을 위한 예시적인 프로세스(1100)를 나타내는 흐름도이다.

[0186] 블록(1102)에서, 프로세스(1100)는 신경망 압축 시스템에 의해, 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하는 단계를 포함할 수 있다. 신경망 압축 시스템은 트레이닝 데이터세트(예를 들어, 도 6의 트레이닝 데이터세트(602))에 대해 트레이닝될 수 있다. 일부 예들에서, 신경망 압축 시스템은 트레이닝 세

트(D)(예를 들어, 트레이닝 데이터 세트)에 대해 미리 트레이닝된 글로벌 모델 파라미터들 $\{\emptyset_D, \theta_D\}$을 포함할 수 있다. 입력 데이터는 이미지 데이터, 비디오 데이터, 오디오 데이터, 및/또는 임의의 다른 데이터를 포함할 수 있다.

[0187] 블록(1104)에서, 프로세스(1100)는 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 단계를 포함할 수 있다. 일부 예들에서, 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들(예를 들어, 모델 파라미터 업데이트들(θ) 및/또는 양자화된 모델 파라미터 업데이트들($\bar{\delta}$))을 포함할 수 있다.

[0188] 일부 예들에서, 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것은 신경망 압축 시스템에서 입력 데이터를 프로세싱(예를 들어, 코딩)하는 것; 프로세싱된 입력 데이터에 기초하여 신경망 압축 시스템에 대한 하나 이상의 손실들(예를 들어, RDM 손실)을 결정하는 것; 및 하나 이상의 손실들에 기초하여 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 것을 포함할 수 있다. 튜닝된 모델 파라미터들은 신경망 압축 시스템에 대한 업데이트들의 세트를 포함할 수 있다.

[0189] 일부 예들에서, 하나 이상의 손실들은 제 1 비트스트림의 사이즈에 기초하여 입력 데이터의 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 제 2 비트스트림의 사이즈에 기초하여 업데이트된 모델 파라미터들의 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함할 수 있다. 일부 경우들에서, 하나 이상의 손실들은 상술된 식 2 를 사용하여 계산될 수 있다.

[0190] 블록(1106)에서, 프로세스(1100)는 잠재 프리어(예를 들어, 잠재 프리어(506), 잠재 프리어(622), 잠재 프리어(708), 잠재 프리어(806), 잠재 프리어(910))를 사용하여 신경망 압축 시스템에 의해, 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림(예를 들어, 비트스트림(510), 비트스트림(520), 비트스트림(710), 비트스트림(810), 비트스트림(916))을 생성하는 단계를 포함할 수 있다. 일부 예들에서, 제 1 비트스트림은 입력 데이터를 잠재 공간 표현으로 인코딩하고 잠재 공간 표현을 엔트로피 인코딩함으로써 생성될 수 있다.

[0191] 블록(1108)에서, 프로세스(1100)는 잠재 프리어 및 모델 프리어(예를 들어, 모델 프리어(714), 모델 프리어(816), 모델 프리어(912), 모델 프리어 $p[\delta])$를 사용하여 신경망 압축 시스템에 의해, 업데이트된 모델 파라미터들(예를 들어, 양자화된 모델 파라미터 업데이트들($\bar{\delta}$))의 압축된 버전을 포함하는 제 2 비트스트림(예를 들어, 비트스트림(510), 비트스트림(520), 비트스트림(712), 비트스트림(811), 비트스트림(918))을 생성하는 단계를 포함할 수 있다.

[0192] 일부 예들에서, 모델 프리어는 독립 가우시안 네트워크 프리어, 독립 라플라스 네트워크 프리어, 및/또는 독립 스파이크 앤 슬래브 네트워크 프리어를 포함할 수 있다.

[0193] 일부 경우들에서, 제 2 비트스트림을 생성하는 것은, 신경망 압축 시스템에 의해, 모델 프리어를 사용하여 잠재 프리어를 엔트로피 인코딩하는 것; 및 신경망 압축 시스템에 의해, 모델 프리어를 사용하여 업데이트된 모델 파라미터들을 엔트로피 인코딩하는 것을 포함할 수 있다.

[0194] 일부 경우들에서, 업데이트된 모델 파라미터들은, 예를 들어, 가중치들, 바이어스들 등과 같은 신경망 파라미터들을 포함할 수 있다. 일부 예들에서, 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함할 수 있다. 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝될 수 있다(예를 들어, 입력 데이터에 대한 하나 이상의 손실들을 감소시키도록 튜닝/조정됨).

[0195] 일부 예들에서, 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함할 수 있다. 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝될 수 있다. 일부 경우들에서, 제 1 비트스트림은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해 생성될 수 있다.

[0196] 일부 예들에서, 제 2 비트스트림을 생성하는 것은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해, 입력 데이터를 입력 데이터의 잠재 공간 표현으로 인코딩하는 것; 및 잠재 프리어를 사용하여 신경망 압축 시스템에 의해, 잠재 공간 표현을 제 1 비트스트림으로 엔트로피 인코딩하는 것을 포함할 수 있다.

[0197] 블록(1110)에서, 프로세스(1100)는 수신기로의 전송을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력하는 것

을 포함할 수 있다. 일부 예들에서, 수신기는 디코더(예를 들어, 디코더(514), 디코더(610), 디코더(718), 디코더(814), 디코더(922))를 포함할 수 있다. 일부 예들에서, 제 2 비트스트림은 또한 잠재 프리어의 압축된 버전 및 모델 프리어의 압축된 버전을 포함할 수 있다.

[0198] 일부 경우들에서, 프로세스 (1100) 는 제 1 비트스트림 및 제 2 비트스트림을 포함하는 연접된 비트스트림을 생성하는 것, 및 연접된 비트스트림을 수신기로 전송하는 것을 더 포함할 수 있다. 예를 들어, 신경망 압축 시스템은 제 1 비트스트림 및 제 2 비트스트림을 연접하고 연접된 비트스트림을 수신기로 전송할 수 있다. 다른 경우들에서, 프로세스 (1100) 는 제 1 비트스트림 및 제 2 비트스트림을 개별적으로 수신기로 전송하는 것을 포함할 수 있다.

[0199] 일부 경우들에서, 프로세스(1100)는 트레이닝 데이터세트에 기초하여 신경망 압축 시스템의 모델 파라미터들을 생성하는 것, 입력 데이터를 사용하여 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 것, 및 모델 파라미터들과 튜닝된 모델 파라미터들 사이의 차이에 기초하여 업데이트들의 세트를 결정하는 것을 포함할 수 있다. 일부 예들에서, 모델 파라미터들은 입력 데이터, 입력 데이터의 압축된 버전의 비트 사이즈, 업데이트들의 세트의 비트 사이즈, 및 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에 기초하여 튜닝될 수 있다.

[0200] 일부 경우들에서, 모델 파라미터들은 입력 데이터 및 업데이트들의 세트를 전송하는 비용과, 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡 사이의 비율(예를 들어, 레이트/왜곡 비율)에 기초하여 튜닝될 수 있다. 일부 예들에서, 비용은 업데이트들의 세트의 비트 사이즈에 기초할 수 있다. 일부 예들에서, 모델 파라미터들을 튜닝하는 것은 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 추가/포함하는 것이 입력 데이터의 압축된 버전의 비트 사이즈 및/또는 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에서의 감소를 수반한다는 결정에 기초하여 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 추가/포함시키는 것을 포함할 수 있다.

[0201] 일부 예들에서, 수신기는 디코더(예를 들어, 디코더(514), 디코더(610), 디코더(718), 디코더(814), 디코더(922))를 포함할 수 있다. 일부 경우들에서, 프로세스 (1100) 는 인코더에 의해, 제 1 비트스트림 및 제 2 비트스트림을 포함하는 데이터를 수신하는 단계를 포함할 수 있다. 일부 경우들에서, 프로세스 (1100) 는 디코더에 의해, 제 2 비트스트림에 기초하여 업데이트된 파라미터들의 세트의 압축된 버전을 디코딩하는 것, 및 업데이트된 파라미터들의 세트를 사용하여 디코더에 의해, 제 1 비트스트림에서의 입력 데이터의 압축된 버전에 기초하여 입력 데이터의 재구성된 버전을 생성하는 것을 더 포함할 수 있다.

[0202] 일부 예들에서, 프로세스(1100)는 레이트-왜곡 및 모델-레이트 손실을 감소시킴으로써 신경망 압축 시스템을 (예를 들어, 트레이닝 데이터세트에 기초하여) 트레이닝하는 것을 포함할 수 있다. 일부 예들에서, 모델-레이트는 모델 업데이트들을 전송하기 위한 비트스트림의 길이를 반영한다.

[0203] 일부 예들에서, 프로세스(1100)는 이전에 설명된 알고리즘 1 및/또는 알고리즘 2 를 구현할 수 있다.

[0204] 도 12는 하나 이상의 이미지들을 압축하기 위한 프로세스(1200)의 예를 나타내는 흐름도이다. 블록(1202)에서, 프로세스(1200)는 압축을 위해 이미지 콘텐츠를 수신하는 단계를 포함할 수 있다. 블록(1204)에서, 프로세스(1200)는 압축할 이미지에 대한 압축 성능을 개선하기 위해 신경망 압축 시스템을 트레이닝함으로써 신경망 압축 시스템을 업데이트하는 것을 포함할 수 있다. 일부 예들에서, 신경망 압축 시스템을 트레이닝하는 것은 신경망 압축 시스템과 연관된 오토인코더 시스템을 트레이닝하는 것을 포함할 수 있다.

[0205] 블록(1206)에서, 프로세스(1200)는 업데이트된 신경망 압축 시스템을 사용하여, 수신된 이미지 콘텐츠를 잠재 공간 표현으로 인코딩하는 단계를 포함할 수 있다.

[0206] 블록(1208)에서, 프로세스(1200)는 확률적 모델 및 코드들의 제 1 서브세트에 기초하여, 업데이트된 신경망 압축 시스템을 사용하여 인코딩된 이미지 콘텐츠의 압축된 버전을 생성하는 단계를 포함할 수 있다. 코드들의 제 1 서브세트는 잠재 공간 표현의 부분을 포함할 수 있다. 예를 들어, 잠재 공간 표현은 코드들의 제 1 서브세트 및 코드들의 하나 이상의 추가적인 서브세트들로 분할될 수 있다.

[0207] 블록(1210)에서, 프로세스(1200)는 확률적 모델 및 업데이트된 신경망 압축 시스템을 사용하여, 업데이트된 신경망 압축 시스템의 압축된 버전을 생성하는 단계를 포함할 수 있다. 일부 경우들에서, 업데이트된 신경망 압축 시스템의 압축된 버전은 신경망 압축 시스템의 양자화된 파라미터 업데이트들을 포함할 수 있고, 신경망 압축 시스템의 업데이트되지 않은 파라미터들을 배제할 수도 있다.

[0208] 블록(1212)에서, 프로세스(1200)는 송신을 위해 업데이트된 신경망 압축 시스템의 압축된 버전 및 인코딩된 이미지 콘텐츠의 압축된 버전을 출력하는 단계를 포함할 수 있다. 업데이트된 신경망 압축 시스템의 압축된 버전 및 인코딩된 이미지 콘텐츠의 압축된 버전은 디코딩을 위해 수신기(예를 들어, 디코더)로 전송될 수 있다. 신경망 압축 시스템의 압축된 버전은 신경망 압축 시스템의 전체 모델 또는 신경망 압축 시스템의 업데이트된 모델 파라미터들을 포함할 수 있다.

[0209] 일부 예들에서, 프로세스(1200)는 이전에 설명된 알고리즘 1 을 구현할 수 있다.

[0210] 도 13은 본 명세서에 설명된 기술들을 사용하여 하나 이상의 이미지들을 압축해제하기 위한 프로세스(1300)의 예를 나타내는 흐름도이다. 블록(1302)에서, 프로세스(1300)는 업데이트된 신경망 압축 시스템의 압축된 버전 (및/또는 신경망 압축 시스템의 하나 이상의 파라미터들) 및 인코딩된 이미지 콘텐츠의 압축된 버전을 수신하는 단계를 포함한다. 업데이트된 신경망 압축 시스템의 압축된 버전 및 인코딩된 이미지 콘텐츠의 압축된 버전은 도 11 또는 도 12에서 이전에 설명된 바와 같이 생성되고 송신될 수 있다. 일부 경우들에서, 인코딩된 이미지 콘텐츠는 인코딩된 이미지 콘텐츠가 생성되는 이미지 콘텐츠의 잠재 공간 표현의 코드들의 제 1 서브세트를 포함할 수 있다.

[0211] 일부 경우들에서, 업데이트된 신경망 압축 시스템의 압축된 버전은 모델 파라미터들을 포함할 수 있다. 일부 예들에서, 모델 파라미터들은 업데이트된 모델 파라미터들을 포함하고 하나 이상의 다른 모델 파라미터들을 배제할 수 있다.

[0212] 블록(1304)에서, 프로세스(1300)는 공유된 확률론적 모델을 사용하여, 업데이트된 신경망 압축 시스템의 압축된 버전을 업데이트된 신경망 압축 시스템 모델로 압축해제하는 단계를 포함할 수 있다.

[0213] 블록(1306)에서, 프로세스(1300)는, 업데이트된 확률적 모델 및 업데이트된 신경망 압축 시스템 모델을 사용하여, 인코딩된 이미지 콘텐츠의 압축된 버전을 잠재 공간 표현으로 압축해제하는 단계를 포함할 수 있다.

[0214] 블록(1308)에서, 프로세스(1300)는 업데이트된 신경망 압축 시스템 모델 및 잠재 공간 표현을 사용하여 재구성된 이미지 콘텐츠를 생성하는 단계를 포함할 수 있다.

[0215] 블록(1310)에서, 프로세스(1300)는 재구성된 이미지 콘텐츠를 출력하는 것을 포함할 수 있다.

[0216] 일부 예들에서, 프로세스(1300)는 위에서 설명된 알고리즘 2 를 구현할 수 있다.

[0217] 일부 예들에서, 본 명세서에 설명된 프로세스들(예를 들어, 프로세스(1100), 프로세스(1200), 프로세스(1300), 및/또는 본 명세서에 설명된 다른 프로세스)은 컴퓨팅 디바이스 또는 장치에 의해 수행될 수도 있다. 일 예에서, 프로세스(1100 및/또는 1200)는 도 4에 예시된 시스템(400)의 송신 디바이스(410)에 의해 수행될 수 있다. 다른 예에서, 프로세스(1100, 1200 및/또는 1300)는 도 4에 도시된 시스템(400) 또는 도 14에 도시된 컴퓨팅 시스템(1400)에 따른 컴퓨팅 디바이스에 의해 수행될 수 있다.

[0218] 컴퓨팅 디바이스는 모바일 디바이스(예를 들어, 모바일 폰), 데스크톱 컴퓨팅 디바이스, 태블릿 컴퓨팅 디바이스, 웨어러블 디바이스(예를 들어, VR 헤드셋, AR 헤드셋, AR 안경, 네트워크 연결 시계 또는 스마트워치, 또는 기타 웨어러블 디바이스), 서버 컴퓨터, 자율 차량 또는 자율 차량의 컴퓨팅 디바이스, 로봇 디바이스, 텔레비전 및/또는 프로세스들 (1100, 1200, 1500, 1300) 을 포함하여 여기에 설명되어 있는 프로세스를 수행하기 위한 리소스 능력들을 갖는 임의의 다른 컴퓨팅 디바이스와 같은 임의의 적합한 디바이스를 포함할 수 있다. 일부 경우들에서, 컴퓨팅 디바이스 또는 장치는 본원에 설명된 프로세스들의 단계들을 수행하도록 구성되는 하나 이상의 입력 디바이스들, 하나 이상의 출력 디바이스들, 하나 이상의 프로세서들, 하나 이상의 마이크로프로세서들, 하나 이상의 마이크로컴퓨터들, 하나 이상의 카메라들, 하나 이상의 센서들, 및/또는 다른 컴포넌트(들)과 같은 여러 컴포넌트들을 포함할 수도 있다. 일부 예들에 있어서, 컴퓨팅 디바이스는 디스플레이, 데이터를 통신 및/또는 수신하도록 구성된 네트워크 인터페이스, 이들의 임의의 조합, 및/또는 다른 컴포넌트(들)를 포함할 수도 있다. 네트워크 인터페이스는 인터넷 프로토콜 (IP) 기반 데이터 또는 다른 타입의 데이터를 통신 및/또는 수신하도록 구성될 수도 있다.

[0219] 컴퓨팅 디바이스의 컴포넌트들은 회로부에서 구현될 수 있다. 예를 들어, 컴포넌트들은 본 명세서에서 설명된 다양한 동작들을 수행하기 위해, 하나 이상의 프로그래밍가능 전자 회로들 (예컨대, 마이크로프로세서들, 그래픽스 프로세싱 유닛들 (GPU들), 디지털 신호 프로세서들 (DSP들), 중앙 프로세싱 유닛들 (CPU들), 및/또는 다른 적합한 전자 회로들) 을 포함할 수 있는 전자 회로들 또는 다른 전자 하드웨어를 포함할 수 있고/있거나 이들을 사용하여 구현될 수 있고, 및/또는 컴퓨터 소프트웨어, 펌웨어, 또는 이들의 임의의 조합을 포함할 수 있

고/있거나 이들을 사용하여 구현될 수 있다.

[0220] 프로세스들 (1100, 1200, 및 1300) 은 논리 흐름도들로서 예시되고, 그 동작은 하드웨어, 컴퓨터 명령들, 또는 이들의 조합으로 구현될 수 있는 동작들의 시퀀스를 나타낸다. 컴퓨터 명령들의 맥락에서, 그 동작들은, 하나 이상의 프로세서들에 의해 실행될 때, 열거된 동작들을 수행하는 하나 이상의 컴퓨터 판독가능 저장 매체들 상에 저장된 컴퓨터 실행가능 명령들을 나타낸다. 일반적으로, 컴퓨터 실행가능 명령들은 특정의 기능들을 수행하거나 또는 특정의 데이터 타입들을 구현하는 루틴들, 프로그램들, 오브젝트들, 컴포넌트들, 데이터 구조들 등을 포함한다. 동작들이 기술되는 순서는 제한으로서 해석되도록 의도되지 않으며, 임의의 수의 기술된 동작들은 프로세스들을 구현하기 위해 임의의 순서로 및/또는 병렬로 결합될 수 있다.

[0221] 추가적으로, 본원에 설명된 프로세스들 (1100, 1200, 및 1300) 및/또는 다른 프로세스들은 실행가능 명령들로 구성된 하나 이상의 컴퓨터 시스템들의 제어 하에서 수행될 수도 있고, 집합적으로 하나 이상의 프로세서 상에서 실행하는 코드 (예를 들어, 실행가능 명령들, 하나 이상의 컴퓨터 프로그램들, 또는 하나 이상의 애플리케이션들) 로서, 하드웨어에 의해, 또는 이들의 조합으로 구현될 수도 있다. 상기 언급된 바와 같이, 코드는 컴퓨터 판독가능 또는 머신 판독가능 저장 매체 상에, 예를 들어, 하나 이상의 프로세서들에 의해 실행가능한 복수의 명령들을 포함하는 컴퓨터 프로그램의 형태로 저장될 수도 있다. 컴퓨터 판독가능 또는 머신 판독가능 저장 매체는 비일시적(non-transitory)일 수도 있다.

[0222] 도 14 는 본 기술의 특정 양태들을 구현하기 위한 시스템의 일 예를 예시한 다이어그램이다. 특히, 도 14는 예를 들어 내부 컴퓨팅 시스템, 원격 컴퓨팅 시스템, 카메라 또는 이들의 임의의 컴포넌트를 구성하는 임의의 컴퓨팅 디바이스일 수 있는 컴퓨팅 시스템(1400)의 예를 예시하고, 여기서 시스템의 컴포넌트는 연결(1405)을 사용하여 서로 통신한다. 연결(1405)은 버스를 사용한 물리적 연결이거나, 또는 칩셋 아키텍처에서와 같이 프로세서(1410)로의 직접 연결일 수 있다. 연결(1405)은 가상 연결, 네트워크형 연결 또는 논리적 연결일 수도 있다.

[0223] 일부 실시양태들에서, 컴퓨팅 시스템(1400)은 본 개시에서 설명된 기능이 데이터 센터, 다중 데이터 센터, 피어 네트워크 등 내에서 분산될 수 있는 분산 시스템이다. 일부 실시양태들에서, 설명된 시스템 컴포넌트들 중 하나 이상은 컴포넌트가 설명된 기능의 일부 또는 전체를 각각 수행하는 다수의 그러한 컴포넌트들을 나타낸다. 일부 실시양태들에서, 컴포넌트는 물리적 또는 가상 디바이스들일 수 있다.

[0224] 예시적인 시스템 (1400) 은 적어도 하나의 프로세싱 유닛 (CPU 또는 프로세서) (1410), 및 판독 전용 메모리 (ROM) (1420) 및 랜덤 액세스 메모리 (RAM) (1425) 와 같은 시스템 메모리 (1415) 를 포함하는 다양한 시스템 컴포넌트들을 프로세서 (1410) 에 커플링시키는 연결 (1405) 을 포함한다. 컴퓨팅 시스템 (1400) 은, 프로세서 (1410) 와 직접 연결되거나 그에 매우 근접하거나 또는 그의 부분으로서 통합된 고속 메모리의 캐시 (1412) 를 포함할 수 있다.

[0225] 프로세서 (1410) 는 임의의 범용 프로세서 및 프로세서 (1410) 를 제어하도록 구성된 스토리지 디바이스 (1430) 에 저장된 서비스들 (1432, 1434 및 1436) 과 같은 하드웨어 서비스 또는 소프트웨어 서비스 그리고 소프트웨어 명령들이 실제 프로세서 설계에 통합되는 특수 목적 프로세서를 포함할 수 있다. 프로세서 (1410) 는 본질적으로 다중 코어 또는 프로세서, 버스, 메모리 컨트롤러, 캐시 등을 포함하는 완전히 독립형 컴퓨팅 시스템일 수도 있다. 다중 코어 프로세서는 대칭 또는 비대칭일 수도 있다.

[0226] 사용자 상호작용을 가능하게 하기 위해, 컴퓨팅 시스템(1400)은 음성용 마이크, 제스처 또는 그래픽 입력용 터치 감지 스크린, 키보드, 마우스, 모션 입력, 음성 등과 같은 임의의 다수의 입력 메커니즘을 나타낼 수 있는 입력 디바이스 (1445) 를 포함한다. 컴퓨팅 시스템(1400)은 또한 다수의 출력 메커니즘 중 하나 이상일 수 있는 출력 디바이스 (1435) 를 포함할 수 있다. 일부 사례들에서, 멀티모달 시스템들이 사용자로 하여금 컴퓨팅 시스템 (1400) 과 통신하기 위해 다중의 타입들의 입력/출력을 제공할 수 있게 할 수 있다. 컴퓨팅 시스템 (1400) 은, 사용자 입력 및 시스템 출력을 일반적으로 통제하고 관리할 수 있는 통신 인터페이스 (1440) 를 포함할 수 있다.

[0227] 통신 인터페이스는, 오디오 잭/플러그, 마이크로폰 잭/플러그, 범용 직렬 버스 (USB) 포트/플러그, Apple® Lightning® 포트/플러그, 이더넷 포트/플러그, 광섬유 포트/플러그, 독점적 유선 포트/플러그, BLUETOOTH® 무선 신호 전송, BLUETOOTH® 저에너지 (BLE) 무선 신호 전송, IBEACON® 무선 신호 전송, 무선 주파수 식별 (RFID) 무선 신호 전송, 근접장 통신 (NFC) 무선 신호 전송, 전용 단거리 통신 (DSRC) 무선 신호 전송, 802.11 Wi-Fi 무선 신호 전송, 무선 로컬 영역 네트워크 (WLAN) 신호 전송, 가시광 통신 (VLC), WiMAX (Worldwide

Interoperability for Microwave Access), 적외선 (IR) 통신 무선 신호 전송, 공중 교환 전화 네트워크 (PSTN) 신호 전송, 통합 서비스 디지털 네트워크 (ISDN) 신호 전송, 3G/4G/5G/LTE 셀룰러 데이터 네트워크 무선 신호 전송, 애드혹 네트워크 신호 전송, 라디오파 신호 전송, 마이크로파 신호 전송, 적외선 신호 전송, 가시광 신호 전송, 자외선 광 신호 전송, 전자기 스펙트럼을 따른 무선 신호 전송, 또는 이들의 일부 조합을 이용하는 것들을 포함하는, 유선 및/또는 무선 트랜시버들을 사용하여 유선 또는 무선 통신들의 수신 및/또는 송신을 수행하거나 용이하게 할 수도 있다.

[0228] 통신 인터페이스 (1440) 는 또한, 하나 이상의 GNSS(Global Navigation Satellite System) 시스템들과 연관된 하나 이상의 위성들로부터의 하나 이상의 신호들의 수신에 기초하여 컴퓨팅 시스템 (1400) 의 위치를 결정하는 데 사용되는 하나 이상의 GNSS 수신기들 또는 트랜시버들을 포함할 수도 있다. GNSS 시스템들은 미국 기반 글로벌 포지셔닝 시스템 (GPS), 러시아 기반 글로벌 내비게이션 위성 시스템 (GLONASS), 중국 기반 베이더우 내비게이션 위성 시스템 (BDS) 및 유럽 기반 Galileo GNSS 를 포함하지만 이에 한정되지 않는다. 임의의 특정 하드웨어 배열에 대해 동작하는 것에 제한이 없으며, 따라서, 여기에서의 기본 특징들은 이들이 개발됨에 따라 개선된 하드웨어 또는 펌웨어 배열들로 쉽게 대체될 수도 있다.

[0229] 스토리지 디바이스 (1430) 는 비휘발성 및/또는 비일시적 및/또는 컴퓨터 판독가능 메모리 디바이스일 수 있고, 다음과 같은 컴퓨터에 의해 액세스가능한 데이터를 저장할 수 있는 하드 디스크 또는 다른 타입들의 컴퓨터 판독가능 매체들일 수 있다: 자기 카세트들, 플래시 메모리 카드들, 솔리드 스테이트 메모리 디바이스들, 디지털 다기능 디스크들, 카트리지들, 플로피 디스크, 플렉시블 디스크, 하드 디스크, 자기 테이프, 자기 스트립/스트라이프, 임의의 다른 자기 저장 매체, 플래시 메모리, 메모리 스토리지, 임의의 다른 솔리드-스테이트 메모리, 컴팩트 디스크 판독 전용 메모리 (CD-ROM) 광 디스크, 재기록가능 컴팩트 디스크 (CD) 광 디스크, 디지털 비디오 디스크 (DVD) 광 디스크, 블루-레이 디스크 (BDD) 광 디스크, 홀로그래픽 광 디스크, 다른 광학 매체, 보안 디지털 (SD) 카드, 마이크로 보안 디지털 (microSD) 카드, Memory Stick® 카드, 스마트카드 칩, EMV 칩, 가입자 아이덴티티 모듈 (SIM) 카드, 미니/마이크로/나노/피코 SIM 카드, 다른 집적 회로 (IC) 칩/카드, 랜덤 액세스 메모리 (RAM), 정적 RAM (SRAM), 동적 RAM (DRAM), 판독 전용 메모리 (ROM), 프로그래밍가능 판독 전용 메모리 (PROM), 소거가능한 프로그래밍가능 판독 전용 메모리 (EPROM), 전기적으로 소거가능한 프로그래밍가능 판독 전용 메모리 (EEPROM), 플래시 EPROM (FLASHEPROM), 캐시 메모리 (L1/L2/L3/L4/L5/L#), 저항성 랜덤 액세스 메모리 (RRAM/ReRAM), 상 변화 메모리 (PCM), 스핀 전달 토크 RAM (STT-RAM), 다른 메모리 칩 또는 카트리지, 및/또는 이들의 조합.

[0230] 스토리지 디바이스 (1430) 는, 그러한 소프트웨어를 정의하는 코드가 프로세서 (1410) 에 의해 실행될 경우 시스템으로 하여금 기능을 수행하게 하는 소프트웨어 서비스들, 서버들, 서비스들 등을 포함할 수 있다. 일부 실시양태들에서, 특정 기능을 수행하는 하드웨어 서비스는, 기능을 수행하기 위해 프로세서 (1410), 연결 (1405), 출력 디바이스 (1435) 등과 같은 필요한 하드웨어 컴포넌트들과 관련하여 컴퓨터 판독가능 매체에 저장된 소프트웨어 컴포넌트를 포함할 수 있다. 용어 "컴퓨터 판독가능 매체" 는 휴대용 또는 비-휴대용 저장 디바이스들, 광학 저장 디바이스들, 및 명령(들) 및/또는 데이터를 저장, 포함, 또는 나를 수 있는 다양한 다른 매체들을 포함하지만 이에 한정되지 않는다. 컴퓨터 판독가능 매체는 데이터가 저장될 수 있고 반송파 및/또는 무선 또는 유선 연결을 통해 전파되는 일시적인 전자 신호를 포함하지 않는 비일시적인 매체를 포함할 수도 있다. 비일시적인 매체의 예들은 자기 디스크 또는 테이프, 콤팩트 디스크 (CD) 또는 디지털 다용도 디스크 (DVD) 와 같은 광학 저장 매체, 플래시 메모리, 메모리 또는 메모리 장치를 포함하나, 이에 한정되는 것은 아니다. 컴퓨터 판독가능 매체는 프로시저, 함수, 서브프로그램, 프로그램, 루틴, 서브루틴, 모듈, 소프트웨어 패키지, 클래스, 또는 명령들, 데이터 구조들, 또는 프로그램 스테이트먼트들의 임의의 조합을 나타낼 수도 있는 코드 및/또는 머신 실행가능 명령들을 저장할 수도 있다. 코드 세그먼트는, 정보, 데이터, 인수들 (arguments), 파라미터들, 또는 메모리 콘텐츠를 전달 및/또는 수신함으로써 다른 코드 세그먼트 또는 하드웨어 회로에 커플링될 수도 있다. 정보, 인수들, 파라미터들, 데이터 등은 메모리 공유, 메시지 전달, 토큰 전달, 네트워크 전송 등을 포함한 임의의 적합한 수단을 통해 전달, 포워딩, 또는 전송될 수도 있다.

[0231] 일부 실시양태들에서, 컴퓨터 판독가능 저장 디바이스들, 매체들, 및 메모리들은 비트 스트림 등을 포함하는 무선 신호 또는 케이블을 포함할 수 있다. 하지만, 언급될 때, 비일시적 컴퓨터 판독가능 저장 매체들은 에너지, 캐리어 신호들, 전자기 파들, 및 신호들 그 자체와 같은 매체들을 명시적으로 배제한다.

[0232] 구체적 상세들은 본원에 제공된 실시양태들 및 예들의 철저한 이해를 제공하기 위하여 상기 설명에서 제공되었다. 하지만, 실시양태들은 이들 특정 상세들 없이 실시될 수도 있음이 당업자에 의해 이해될 것이다. 설명의 명료화를 위해, 일부 사례들에 있어서, 본 기술은 디바이스들, 디바이스 컴포넌트들, 소프트웨어에서 구

현된 방법에서의 단계들 또는 루틴들, 또는 하드웨어와 소프트웨어의 조합들을 포함하는 개별 기능 블록들을 포함하는 것으로서 제시될 수도 있다. 도면들에서 도시되고/되거나 본 명세서에서 설명된 것들 이외의 추가적인 컴포넌트들이 사용될 수도 있다. 예를 들어, 회로들, 시스템들, 네트워크들, 프로세스들, 및 다른 컴포넌트들은 그 실시양태들을 불필요한 상세로 불명료하게 하지 않기 위해 블록도 형태의 컴포넌트들로서 도시될 수도 있다. 다른 예들에서, 잘 알려진 회로들, 프로세스들, 알고리즘들, 구조들, 및 기법들은, 실시양태들을 불명료하게 하는 것을 회피하기 위해 불필요한 상세 없이 도시될 수도 있다.

[0233] 개별 실시양태들은, 플로우차트, 흐름도, 데이터 흐름도, 구조도, 또는 블록도로서 도시되는 프로세스 또는 방법으로서 위에서 설명될 수도 있다. 비록 플로우차트가 동작들을 순차적인 프로세스로서 기술할 수도 있지만, 동작들 중 다수는 병렬로 또는 동시에 수행될 수 있다. 부가적으로, 동작들의 순서는 재배열될 수도 있다. 프로세스는, 그의 동작들이 완료될 때 종료되지만, 도면에 포함되지 않은 추가적인 단계들을 가질 수 있을 것이다. 프로세스는 방법, 함수, 프로시저, 서브루틴, 서브프로그램 등에 대응할 수도 있다. 프로세스가 함수에 대응할 경우, 그 종료는 그 함수의 호출 함수 또는 메인 함수로의 복귀에 대응할 수 있다.

[0234] 상술된 예들에 따른 프로세스들 및 방법들은 컴퓨터 판독가능 매체들에 저장되거나 그 외에 컴퓨터 판독가능 매체들로부터 이용가능한 컴퓨터 실행가능 명령들을 이용하여 구현될 수 있다. 이러한 명령들은, 예를 들어, 범용 컴퓨터, 특수 목적 컴퓨터, 또는 프로세싱 디바이스가 특정 기능 또는 기능들의 그룹을 수행하게 하거나 그 외에 수행하도록 구성하는 명령들 및 데이터를 포함할 수 있다. 사용되는 컴퓨터 리소스들의 부분들은 네트워크를 통해 액세스가능할 수 있다. 컴퓨터 실행가능 명령들은, 예를 들어, 어셈블리 언어, 펌웨어, 소스 코드와 같은 바이너리들, 중간 포맷 명령들일 수도 있다. 설명된 예들에 따른 방법들 동안 명령들, 사용된 정보, 및/또는 생성된 정보를 저장하기 위해 사용될 수도 있는 컴퓨터 판독가능 매체들의 예들은 자기 또는 광학 디스크들, 플래시 메모리, 비휘발성 메모리가 제공된 USB 디바이스들, 네트워킹된 저장 디바이스들 등을 포함한다.

[0235] 이들 개시들에 따른 프로세스들 및 방법들을 구현하는 디바이스들은 하드웨어, 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 하드웨어 디스크립션 언어들, 또는 이들의 임의의 조합을 포함할 수 있으며, 다양한 폼 팩터들 중 임의의 것을 취할 수 있다. 소프트웨어, 펌웨어, 미들웨어, 또는 마이크로코드로 구현될 경우, 필요한 태스크들을 수행하기 위한 프로그램 코드 또는 코드 세그먼트들 (예를 들어, 컴퓨터 프로그램 제품) 은 컴퓨터 판독가능 또는 머신 판독가능 매체에 저장될 수도 있다. 프로세서(들)는 필요한 태스크들을 수행할 수도 있다. 폼 팩터들의 통상적인 예들은 랩톱들, 스마트 폰들, 모바일 폰들, 태블릿 디바이스들 또는 다른 소형 퍼스널 컴퓨터들, 퍼스널 디지털 어시스턴트들, 랙마운트 디바이스들, 독립형 디바이스들 등을 포함한다. 본원에 기술된 기능성은 또한, 주변장치들 또는 추가 카드들에서 구현될 수 있다. 이러한 기능성은 또한, 추가적인 예에 의해, 단일 디바이스에서 실행되는 상이한 프로세스들 또는 상이한 칩들 중에서 회로 기판 상에서 구현될 수 있다.

[0236] 명령들, 이러한 명령들을 운반하기 위한 매체들, 그것들을 시행하기 위한 컴퓨팅 리소스들, 및 이러한 컴퓨팅 리소스들을 지원하기 위한 다른 구조들은 본 개시물에서 설명될 기능들을 제공하기 위한 예시적인 수단들이다.

[0237] 전술한 설명에서, 본 출원의 양태들은 그것들의 특정 실시양태들을 참조하여 설명되었지만, 당업자는 본원이 이에 제한되지 않는다는 것을 인식할 것이다. 따라서, 본 출원의 예시적인 실시양태들이 본원에 상세히 설명되었지만, 본 발명의 개념은 달리 다양하게 구체화되고 채택될 수도 있으며, 첨부된 청구범위는 선행 기술에 의해 제한되는 것을 제외하고는 그러한 변형을 포함하는 것으로 해석되도록 의도된다. 전술한 애플리케이션의 다양한 특징들 및 양태들은 개별적으로 또는 공동으로 사용될 수도 있다. 또한, 실시양태들은 본 명세서의 더 넓은 사상 및 범위를 벗어나지 않으면서 본 명세서에 기재된 것 이외의 임의의 수의 환경들 및 애플리케이션들에서 이용될 수 있다. 이에 따라, 명세서 및 도면들은 한정적 의미보다는 예시적 의미로 간주되어야 한다. 예시의 목적 상, 방법들은 특정 순서로 기술되었다. 대안적인 실시양태들에서, 상기 방법들은 설명된 것과 다른 순서로 수행될 수도 있다는 것을 이해해야 한다.

[0238] 당업자는 본 명세서에서 사용된 미만 ("<") 및 초과 (">") 기호들 또는 용어가 본 개시의 범위를 벗어나지 않으면서 이하 ("≤") 및 이상 ("≥") 기호들로 각각 대체될 수 있음을 알 것이다.

[0239] 컴포넌트들이 특정 동작들을 수행하도록 "구성된" 것으로서 설명되는 경우, 그러한 구성은 예를 들어, 전자 회로들 또는 다른 하드웨어를 설계하여 그 동작을 수행하는 것에 의해, 프로그래밍가능 전자 회로들 (예컨대, 마이크로프로세서들 또는 다른 적합한 전자 회로들) 을 프로그래밍하여 그 동작을 수행하는 것에 의해, 또는 이들의 임의의 조합에 의해, 달성될 수 있다.

[0240] 문구 “~ 에 커플링된 (coupled to)” 은 다른 컴포넌트에 직접적으로 또는 간접적으로 물리적으로 접속된 임의의 컴포넌트, 및/또는, 다른 컴포넌트와 직접적으로 또는 간접적으로 통신하는 (예컨대, 유선 또는 무선 접속, 및/또는 다른 적합한 통신 인터페이스를 통해 다른 컴포넌트에 접속된) 임의의 컴포넌트를 지칭한다.

[0241] 세트 "중의 적어도 하나" 또는 세트 "중의 하나 이상" 을 인용하는 청구항 언어 또는 기타 언어는 그 세트 중의 하나의 멤버 또는 그 세트의 다수의 멤버들이 청구항을 만족하는 것을 나타낸다. 예를 들어, "A 및 B 중 적어도 하나" 또는 "A 또는 B 중 적어도 하나"를 인용하는 청구항 언어는 A, B, 또는 A 및 B 를 의미한다. 다른 예에서, "A, B, 및 C 중 적어도 하나" 또는 "A, B, 또는 C 중 적어도 하나"를 인용하는 청구항 언어는 A, B, C, 또는 A 및 B, 또는 A 및 C, 또는 B 및 C, 또는 A 및 B 및 C 를 의미한다. 언어 세트 “중 적어도 하나” 및/또는 세트 중 “하나 이상” 은 세트를 그 세트에 열거된 항목들로 제한하지 않는다. 예를 들어, "A 및 B 중 적어도 하나” 또는 "A 또는 B 중 적어도 하나” 를 인용하는 청구항 언어는 A, B, 또는 A 및 B 를 의미할 수 있으며, A 및 B 의 세트에 열거되지 않은 항목들을 추가적으로 포함할 수 있다.

[0242] 본 명세서에 개시된 예들과 관련하여 설명된 다양한 예시적인 논리 블록들, 모듈들, 회로들, 및 알고리즘 단계들은 전자 하드웨어, 컴퓨터 소프트웨어, 펌웨어, 또는 이들의 조합들로서 구현될 수도 있다. 하드웨어와 소프트웨어의 이러한 상호교환가능성을 분명히 예시하기 위해, 다양한 예시적인 컴포넌트들, 블록들, 모듈들, 회로들 및 단계들이 일반적으로 그들의 기능성의 관점에서 상기에서 설명되었다. 그러한 기능성이 하드웨어로서 구현되는지 또는 소프트웨어로서 구현되는지는 전체 시스템에 부과된 설계 제약들 및 특정 애플리케이션에 의존한다. 당업자는 설명된 기능성을 각각의 특정 애플리케이션에 대하여 다양한 방식들로 구현할 수도 있지만, 그러한 구현의 결정들이 본 출원의 범위로부터의 일탈을 야기하는 것으로서 해석되지는 않아야 한다.

[0243] 본 명세서에서 설명되는 기법들은 전자 하드웨어, 컴퓨터 소프트웨어, 펌웨어, 또는 그것들의 임의의 조합으로 또한 구현될 수도 있다. 이러한 기법들은 범용 컴퓨터들, 무선 통신 디바이스 핸드셋들, 또는 무선 통신 디바이스 핸드셋들 및 다른 디바이스들에서의 애플리케이션을 포함한 다수의 용도들을 갖는 집적회로 디바이스들과 같은 다양한 디바이스들 중 임의의 것으로 구현될 수도 있다. 모듈들 또는 컴포넌트들로서 설명되는 임의의 피처들은 통합형 로직 디바이스에 함께 또는 개별적이지만 상호작용하는 로직 디바이스들로서 따로따로 구현될 수도 있다. 소프트웨어에서 구현되는 경우, 그 기법들은, 실행될 경우 상기 설명된 방법들, 알고리즘들, 및/또는 동작들 중 하나 이상을 수행하는 명령들을 포함하는 프로그램 코드를 포함하는 컴퓨터 판독가능 데이터 저장 매체에 의해 적어도 부분적으로 실현될 수도 있다. 컴퓨터 판독가능 데이터 저장 매체는 컴퓨터 프로그램 제품의 부분을 형성할 수도 있으며, 이는 패키징 재료들을 포함할 수도 있다. 컴퓨터 판독가능 매체는 동기식 동적 랜덤 액세스 메모리 (SDRAM) 와 같은 랜덤 액세스 메모리 (RAM), 판독 전용 메모리 (ROM), 비휘발성 랜덤 액세스 메모리 (NVRAM), 전기적으로 소거가능한 프로그래밍가능 판독 전용 메모리 (EEPROM), 플래시 메모리, 자기 또는 광학 데이터 저장 매체들 등과 같은 메모리 또는 데이터 저장 매체들을 포함할 수도 있다. 그 기법들은, 부가적으로 또는 대안적으로, 전파된 신호들 또는 파들과 같이, 명령들 또는 데이터 구조들의 형태로 프로그램 코드를 수록하거나 통신하고 그리고 컴퓨터에 의해 액세스, 판독 및/또는 실행될 수 있는 컴퓨터 판독가능 통신 매체에 의해 적어도 부분적으로 실현될 수도 있다.

[0244] 프로그램 코드는, 하나 이상의 디지털 신호 프로세서들 (DSP들), 범용 마이크로 프로세서들, 주문형 집적 회로들 (ASIC들), 필드 프로그래밍가능 로직 어레이들 (FPGA들), 또는 다른 균등한 집적된 또는 별개의 로직 회로부와 같은 하나 이상의 프로세서들을 포함할 수도 있는 프로세서에 의해 실행될 수도 있다. 그러한 프로세서는 본 개시에서 설명된 기법들 중 임의의 기법을 수행하도록 구성될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 대안적으로, 그 프로세서는 임의의 종래의 프로세서, 제어기, 마이크로제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한, 컴퓨팅 디바이스들의 조합, 예를 들어, DSP 와 마이크로프로세서의 조합, 복수의 마이크로프로세서들, DSP 코어와 결합된 하나 이상의 마이크로프로세서들, 또는 임의의 다른 그러한 구성으로서 구현될 수도 있다. 이에 따라, 본 명세서에서 사용된 바와 같은 용어 "프로세서" 는 전술한 구조, 전술한 구조의 임의의 조합, 또는 본 명세서에서 설명된 기법들의 구현에 적합한 임의의 다른 구조 또는 장치 중 임의의 것을 지칭할 수도 있다.

[0245] 본 개시의 예시적인 예들은 다음을 포함한다:

[0246] 양태 1: 메모리; 및 상기 메모리에 커플링된 하나 이상의 프로세서들을 포함하는 장치로서, 상기 하나 이상의 프로세서들은: 신경망 압축 시스템에 의해, 상기 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하고; 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 것으로서, 상기 업데이트들의 세트는 상기 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함하는, 상기 업데이트들의 세트를 결정하고;

잠재 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 생성하고; 상기 잠재 프리어 및 모델 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 생성하고; 그리고 수신기로의 전송을 위해 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 출력하도록 구성되는, 장치.

[0247] 양태 2: 양태 1 의 장치에 있어서, 상기 제 2 비트스트림은 잠재 프리어의 압축된 버전 및 모델 프리어의 압축된 버전을 더 포함하는, 장치.

[0248] 양태 3: 양태 1 내지 양태 2 중 어느 것의 장치에 있어서, 상기 하나 이상의 프로세서들은, 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 연접된 비트스트림을 생성하고; 그리고 상기 연접된 비트스트림을 상기 수신기에 전송하도록 구성되는, 장치.

[0249] 양태 4: 양태들 1 내지 양태 3 중 어느 것의 장치에 있어서, 상기 제 2 비트스트림을 생성하기 위해, 상기 하나 이상의 프로세서들은, 상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 잠재 프리어를 엔트로피 인코딩하고; 상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 업데이트된 모델 파라미터들을 엔트로피 인코딩하도록 구성되는, 장치.

[0250] 양태 5: 양태 1 내지 양태 4 중 어느 것의 장치에 있어서, 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝되는, 장치.

[0251] 양태 6: 양태 1 내지 양태 5 중 어느 것의 장치에 있어서, 상기 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝되고, 제 1 비트스트림은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해 생성되는, 장치.

[0252] 양태 7: 양태 6 의 장치에 있어서, 상기 제 2 비트스트림을 생성하기 위해, 상기 하나 이상의 프로세서들은, 상기 하나 이상의 업데이트된 파라미터들을 사용하여 상기 신경망 압축 시스템에 의해, 상기 입력 데이터를 상기 입력 데이터의 잠재 공간 표현으로 인코딩하고; 상기 잠재 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 잠재 공간 표현을 상기 제 1 비트스트림으로 엔트로피 인코딩하도록 구성되는, 장치.

[0253] 양태 8: 양태 1 내지 양태 7 중 어느 것의 장치에 있어서, 상기 하나 이상의 프로세서들은, 상기 신경망 압축 시스템을 트레이닝시키는 데 사용되는 트레이닝 데이터세트에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 생성하고; 상기 입력 데이터를 사용하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하고; 그리고 상기 모델 파라미터들과 상기 튜닝된 모델 파라미터들 사이의 차이에 기초하여 상기 업데이트들의 세트를 결정하도록 구성되는, 장치.

[0254] 양태 9: 양태 8 의 장치에 있어서, 상기 모델 파라미터들은 입력 데이터, 입력 데이터의 압축된 버전의 비트 사이즈, 업데이트들의 세트의 비트 사이즈, 및 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에 기초하여 튜닝되는, 장치.

[0255] 양태 10: 양태 8 의 장치에 있어서, 상기 모델 파라미터들은 상기 입력 데이터 및 상기 업데이트들의 세트를 전송하는 비용 및 상기 입력 데이터와 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡의 비율에 기초하여 튜닝되고, 상기 비용은 상기 업데이트들의 세트의 비트 사이즈에 기초하는, 장치.

[0256] 양태 11: 양태 8 의 장치에 있어서, 상기 모델 파라미터들을 튜닝하기 위해, 상기 하나 이상의 프로세서들은, 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함하는 것이 상기 입력 데이터의 압축된 버전의 비트 사이즈, 및 상기 입력 데이터와 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡 중 적어도 하나에서의 감소를 수반한다는 결정에 기초하여 상기 튜닝된 모델 파라미터들에 상기 하나 이상의 파라미터들을 포함하도록 구성되는, 장치.

[0257] 양태 12: 양태 1 내지 양태 11 중 어느 것의 장치에 있어서, 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하기 위해, 상기 하나 이상의 프로세서들은, 상기 신경망 압축 시스템에서 상기 입력 데이터를 프로세싱하고; 상기 프로세싱된 입력 데이터에 기초하여 상기 신경망 압축 시스템에 대한 하나 이상의 손실들을 결정하고; 그리고 상기 하나 이상의 손실들에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하도록 구성되고, 상기 튜닝된 모델 파라미터들은 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 포함하는, 장치.

[0258] 양태 13: 양태 12 의 장치에 있어서, 상기 하나 이상의 손실들은, 상기 제 1 비트스트림의 사이즈에 기초하여 상기 입력 데이터의 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 상기 입력 데이터와 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 상기 제 2 비트스트림의 사이즈에 기초하여 상기 업데이트된 모델 파라미터들의 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함하는, 장치.

[0259] 양태 14: 양태 1 내지 양태 13 중 어느 것의 장치에 있어서, 상기 수신기는 인코더를 포함하고, 상기 하나 이상의 프로세서들은, 상기 인코더에 의해, 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 데이터를 수신하고; 상기 디코더에 의해, 상기 제 2 비트스트림에 기초하여 상기 업데이트된 모델 파라미터들의 압축된 버전을 디코딩하고; 그리고 상기 디코더에 의해, 상기 업데이트된 파라미터들의 세트를 사용하여, 상기 제 1 비트스트림에서의 상기 입력 데이터의 압축된 버전에 기초하여 상기 입력 데이터의 재구성된 버전을 생성하도록 구성되는, 장치.

[0260] 양태 15: 양태 1 내지 양태 14 중 어느 것의 장치에 있어서, 상기 하나 이상의 프로세서들은, 레이트-왜곡 및 모델-레이트 손실을 감소시킴으로써 상기 신경망 압축 시스템을 트레이닝하도록 구성되고, 모델-레이트는 모델 업데이트들을 전송하기 위한 비트스트림의 길이를 반영하는, 장치.

[0261] 양태 16: 양태 1 내지 양태 15 중 어느 것의 장치에 있어서, 상기 모델 프리어는 독립 가우시안 네트워크 프리어, 독립 라플라스 네트워크 프리어, 및 독립 스파이크 앤 슬래브 네트워크 프리어 중 적어도 하나를 포함하는, 장치.

[0262] 양태 17: 양태 1 내지 양태 16 중 어느 것의 장치에 있어서, 상기 장치는 모바일 디바이스를 포함하는, 장치.

[0263] 양태 18: 양태 1 내지 양태 17 중 어느 것의 장치에 있어서, 입력 데이터를 캡처하도록 구성된 카메라를 더 포함하는, 장치.

[0264] 양태 19: 방법은: 신경망 압축 시스템에 의해, 상기 신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 수신하는 단계; 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 단계 - 상기 업데이트들의 세트는 상기 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함함 -; 잠재 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 입력 데이터의 압축된 버전을 포함하는 제1 비트스트림을 생성하는 단계; 상기 잠재 프리어 및 모델 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제2 비트스트림을 생성하는 단계; 및 수신기로의 전송을 위해 상기 제1 비트스트림 및 상기 제2 비트스트림을 출력하는 단계를 포함하는, 방법.

[0265] 양태 20: 양태 19 의 방법에 있어서, 상기 제 2 비트스트림은 잠재 프리어의 압축된 버전 및 모델 프리어의 압축된 버전을 더 포함하는, 방법.

[0266] 양태 21: 양태 19 내지 양태 20 중 어느 것의 방법에 있어서, 상기 하나 이상의 프로세서들은, 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 연접된 비트스트림을 생성하고; 그리고 상기 연접된 비트스트림을 상기 수신기에 전송하도록 구성되는, 방법.

[0267] 양태 22: 양태들 19 내지 양태 21 중 어느 것의 방법에 있어서, 상기 제 2 비트스트림을 생성하는 단계는: 상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 잠재 프리어를 엔트로피 인코딩하는 단계; 및 상기 신경망 압축 시스템에 의해, 상기 모델 프리어를 사용하여 상기 업데이트된 모델 파라미터들을 엔트로피 인코딩하는 단계를 포함하는, 방법.

[0268] 양태 23: 양태 19 내지 양태 22 중 어느 것의 방법에 있어서, 업데이트된 모델 파라미터들은 디코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝되는, 방법.

[0269] 양태 24: 양태 19 내지 양태 23 중 어느 것의 방법에 있어서, 상기 업데이트된 모델 파라미터들은 인코더 모델의 하나 이상의 업데이트된 파라미터들을 포함하고, 상기 하나 이상의 업데이트된 파라미터들은 입력 데이터를 사용하여 튜닝되고, 제 1 비트스트림은 하나 이상의 업데이트된 파라미터들을 사용하여 신경망 압축 시스템에 의해 생성되는, 방법.

[0270] 양태 25: 양태 24 의 방법에 있어서, 상기 제 2 비트스트림을 생성하는 단계는, 상기 하나 이상의 업데이트된 파라미터를 사용하여 상기 신경망 압축 시스템에 의해, 상기 입력 데이터를 상기 입력 데이터의 잠재 공간 표현으로 인코딩하는 단계; 및 상기 잠재 프리어를 사용하여 상기 신경망 압축 시스템에 의해, 상기 잠재 공간 표현

을 상기 제 1 비트스트림으로 엔트로피 인코딩하는 단계를 포함하는, 방법.

[0271] 양태 26: 양태 19 내지 양태 25 중 어느 것의 방법에 있어서, 상기 하나 이상의 프로세서들은, 상기 신경망 압축 시스템을 트레이닝시키는 데 사용되는 트레이닝 데이터세트에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 생성하고; 상기 입력 데이터를 사용하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하고; 그리고 상기 모델 파라미터들과 상기 튜닝된 모델 파라미터들 사이의 차이에 기초하여 상기 업데이트들의 세트를 결정하도록 구성되는, 방법.

[0272] 양태 27: 양태 26 의 방법에 있어서, 상기 모델 파라미터들은 입력 데이터, 입력 데이터의 압축된 버전의 비트 사이즈, 업데이트들의 세트의 비트 사이즈, 및 입력 데이터와 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡에 기초하여 튜닝되는, 방법.

[0273] 양태 28: 양태 26 의 방법에 있어서, 상기 모델 파라미터들은 상기 입력 데이터 및 상기 업데이트들의 세트를 전송하는 비용 및 상기 입력 데이터와 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡의 비율에 기초하여 튜닝되고, 상기 비용은 상기 업데이트들의 세트의 비트 사이즈에 기초하는, 방법.

[0274] 양태 29: 양태 26 의 방법에 있어서, 상기 모델 파라미터들을 튜닝하는 단계는, 상기 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함하는 것이 상기 입력 데이터의 압축된 버전의 비트 사이즈, 및 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터와 상기 입력 데이터 사이의 왜곡 중 적어도 하나에서의 감소를 수반한다는 결정에 기초하여 상기 튜닝된 모델 파라미터들에 하나 이상의 파라미터들을 포함시키는 단계를 포함하는, 방법.

[0275] 양태 30: 양태 19 내지 양태 29 중 어느 것의 방법에 있어서, 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 결정하는 단계는: 상기 신경망 압축 시스템에서 상기 입력 데이터를 프로세싱하는 단계; 상기 프로세싱된 입력 데이터에 기초하여 상기 신경망 압축 시스템에 대한 하나 이상의 손실들을 결정하는 단계; 및 상기 하나 이상의 손실들에 기초하여 상기 신경망 압축 시스템의 모델 파라미터들을 튜닝하는 단계를 포함하고, 상기 튜닝된 모델 파라미터들은 상기 신경망 압축 시스템에 대한 업데이트들의 세트를 포함하는, 방법.

[0276] 양태 31: 양태 30 의 방법에 있어서, 상기 하나 이상의 손실들은, 상기 제 1 비트스트림의 사이즈에 기초하여 상기 입력 데이터의 압축된 버전을 전송하기 위한 레이트와 연관된 레이트 손실, 상기 입력 데이터와 상기 입력 데이터의 압축된 버전으로부터 생성된 재구성된 데이터 사이의 왜곡과 연관된 왜곡 손실, 및 상기 제 2 비트스트림의 사이즈에 기초하여 상기 업데이트된 모델 파라미터들의 압축된 버전을 전송하기 위한 레이트와 연관된 모델 레이트 손실을 포함하는, 방법.

[0277] 양태 32: 양태 19 내지 양태 31 중 어느 것의 방법에 있어서, 상기 수신기는 인코더를 포함하고, 상기 하나 이상의 프로세서들은, 상기 인코더에 의해, 상기 제 1 비트스트림 및 상기 제 2 비트스트림을 포함하는 데이터를 수신하고; 상기 디코더에 의해, 상기 제 2 비트스트림에 기초하여 상기 업데이트된 모델 파라미터들의 압축된 버전을 디코딩하고; 그리고 상기 디코더에 의해, 상기 업데이트된 파라미터들의 세트를 사용하여, 상기 제 1 비트스트림에서의 상기 입력 데이터의 압축된 버전에 기초하여 상기 입력 데이터의 재구성된 버전을 생성하도록 구성되는, 방법.

[0278] 양태 33: 양태 19 내지 양태 32 중 어느 것의 방법에 있어서, 상기 하나 이상의 프로세서들은, 레이트-왜곡 및 모델-레이트 손실을 감소시킴으로써 상기 신경망 압축 시스템을 트레이닝하도록 구성되고, 모델-레이트는 모델 업데이트들을 전송하기 위한 비트스트림의 길이를 반영하는, 방법.

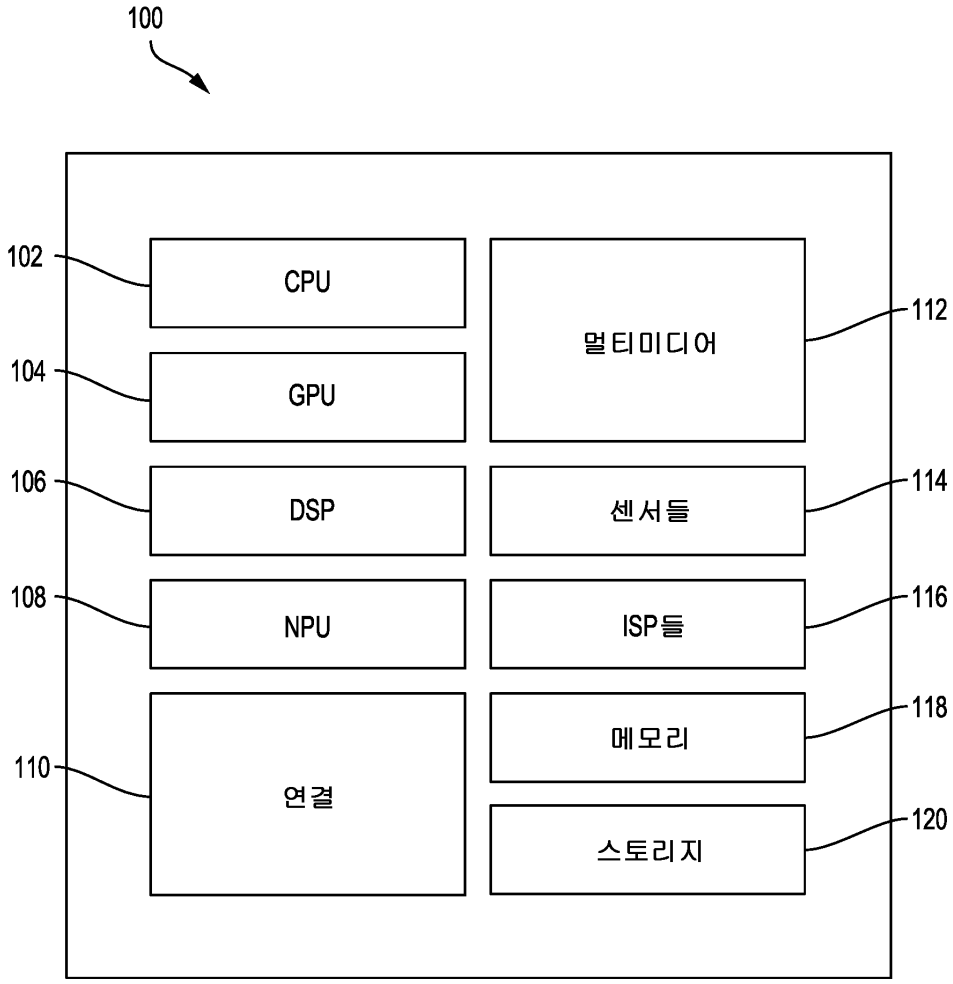
[0279] 양태 34: 양태 19 내지 양태 33 중 어느 것의 방법에 있어서, 상기 모델 프리어는 독립 가우시안 네트워크 프리어, 독립 라플라스 네트워크 프리어, 및 독립 스파이크 앤 슬래브 네트워크 프리어 중 적어도 하나를 포함하는, 방법.

[0280] 양태 35: 하나 이상의 프로세서들에 의해 실행될 때, 상기 하나 이상의 프로세서들로 하여금, 양태 19 내지 33 중 어느 것에 따른 방법을 수행하게 하는 명령들을 저장한 비일시적 컴퓨터 판독가능 매체.

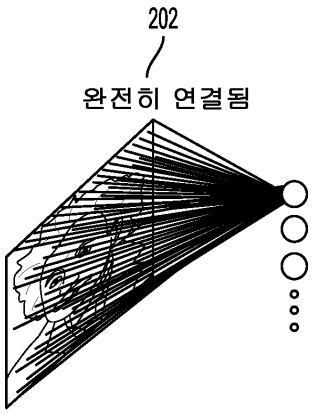
[0281] 양태 36: 양태 19 내지 33 중 어느 것에 따른 방법을 수행하기 위한 수단을 포함하는 장치.

도면

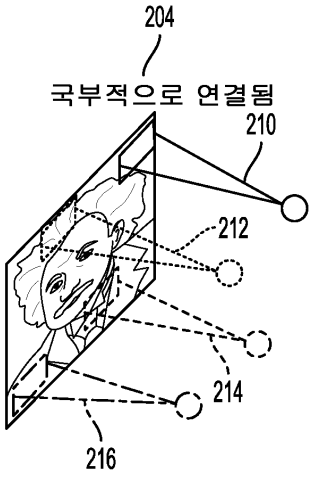
도면1


100
102
CPU
104
GPU
106
DSP
108
NPU
110
연결
112
멀티미디어
114
센서들
116
ISP들
118
메모리
120
스토리지

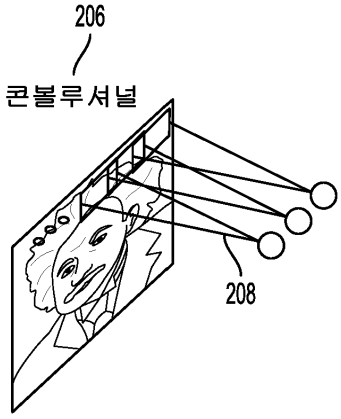
도면2a


202
완전히 연결됨

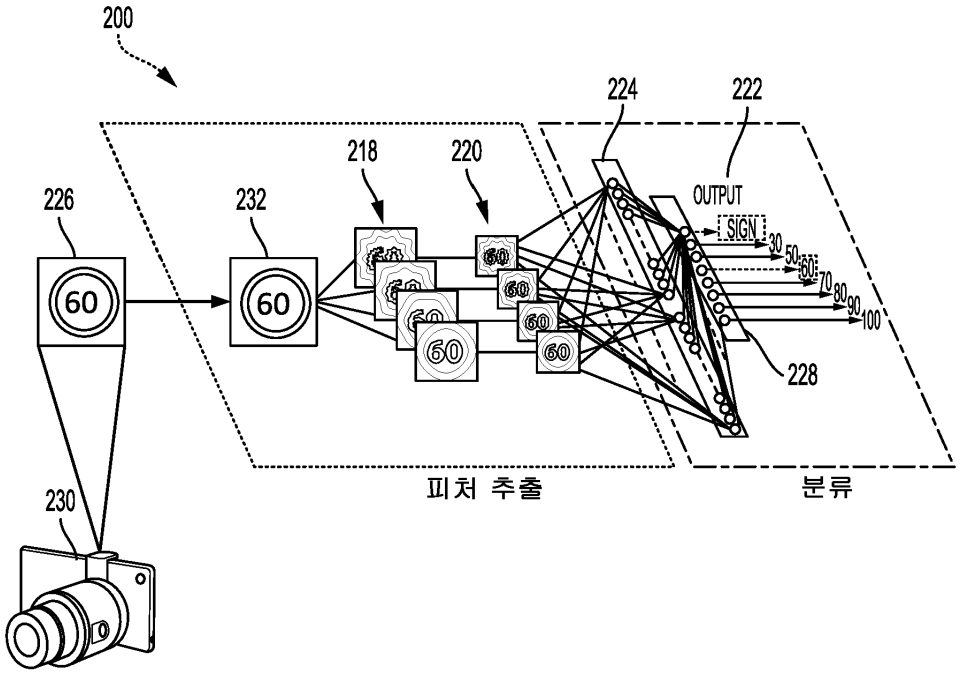
도면2b



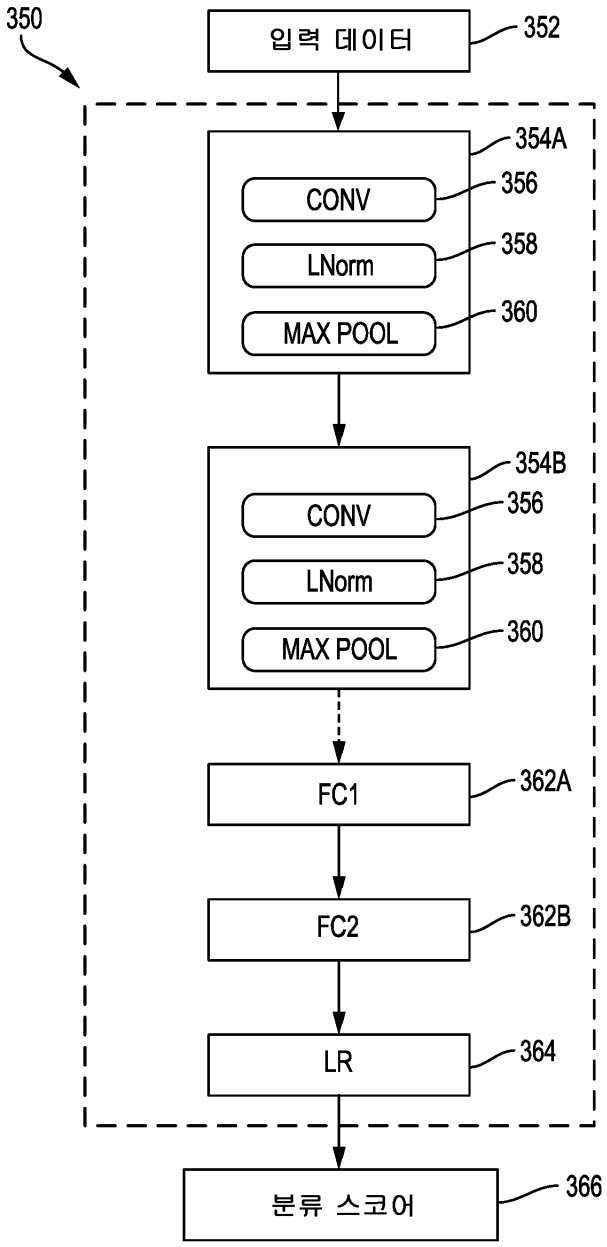
도면2c



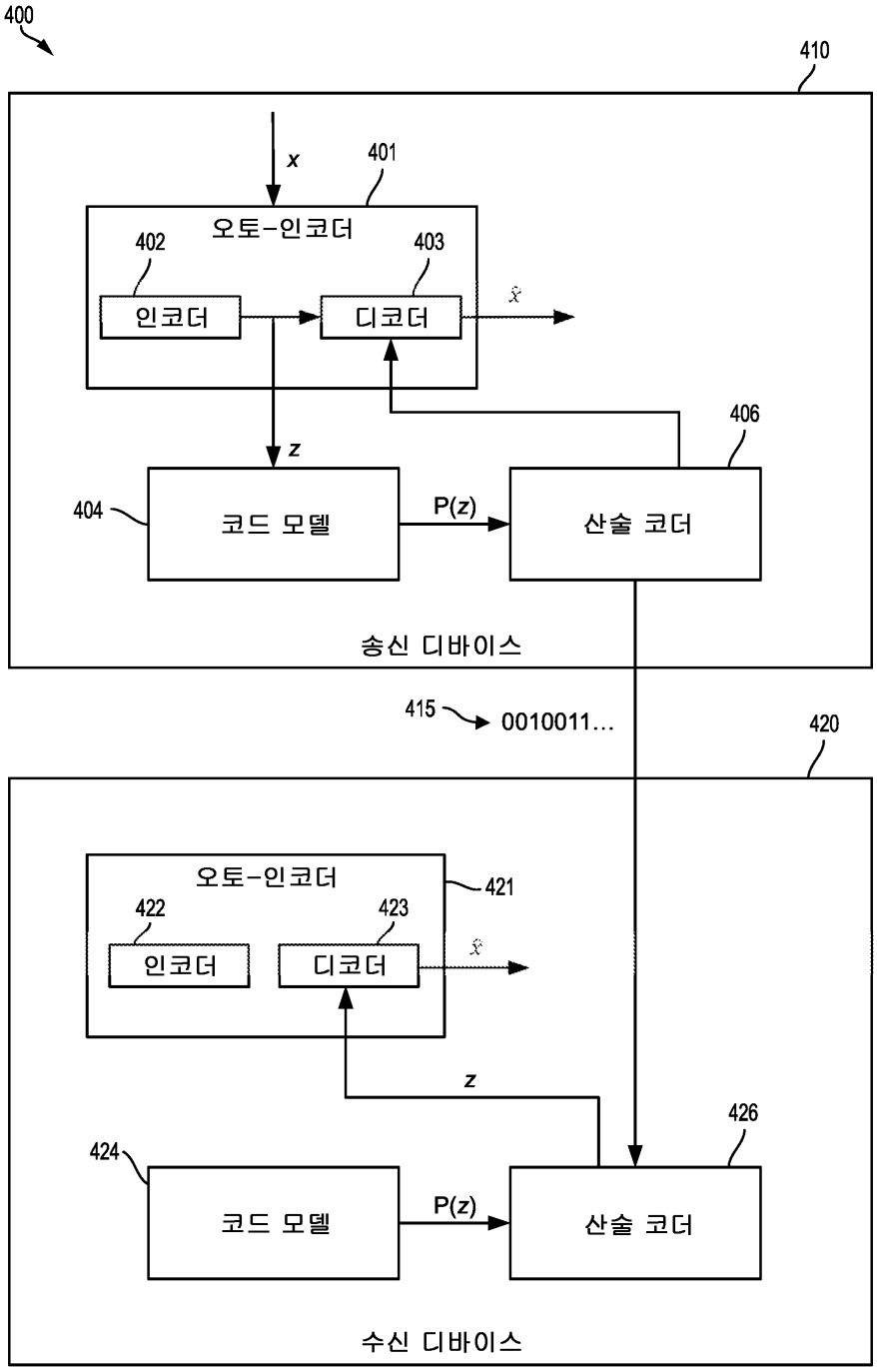
도면2d



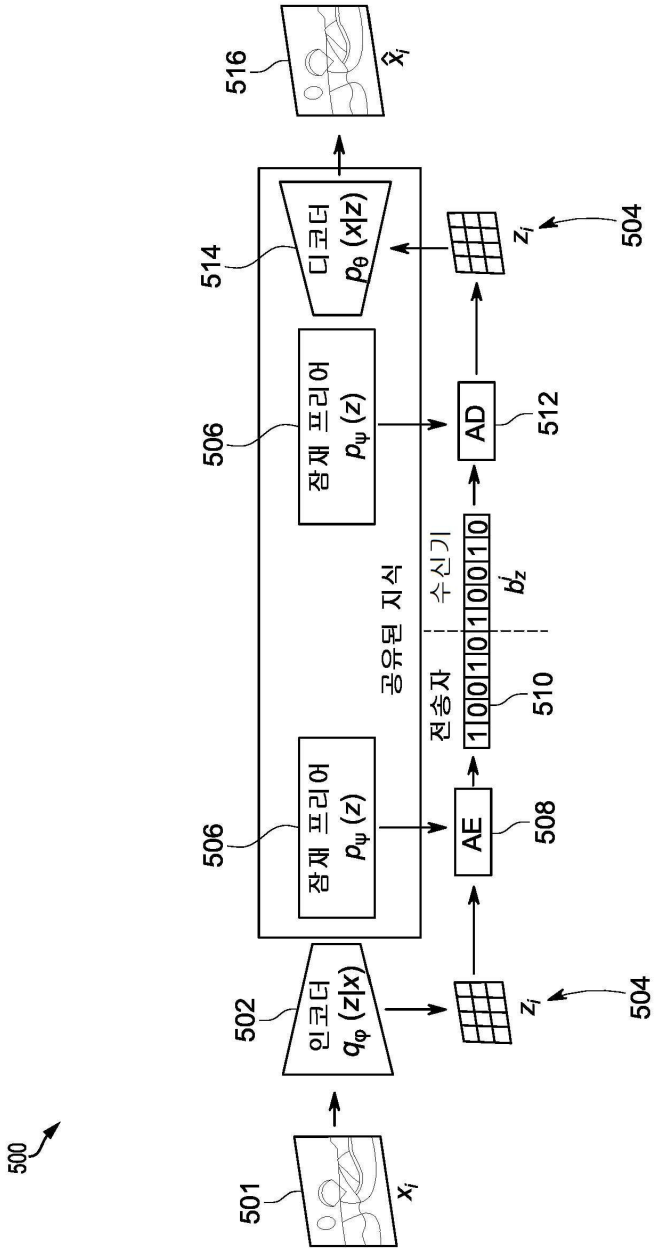
도면3


350
입력 데이터
352
354A
CONV
356
LNorm
358
MAX POOL
360
354B
CONV
356
LNorm
358
MAX POOL
360
FC1
362A
FC2
362B
LR
364
분류 스코어
366

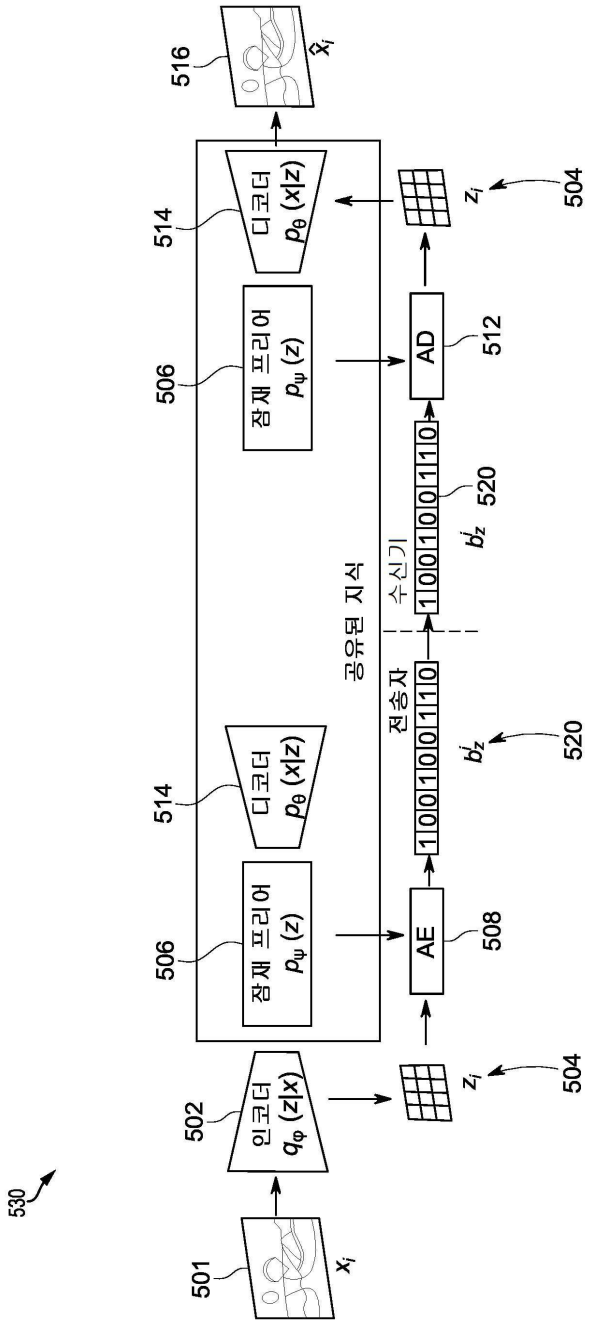
도면4


400
410
x
401
오토-인코더
402
403
인코더
디코더
$\hat{x}$
406
z
404
코드 모델
P(z)
산술 코더
송신 디바이스
415
0010011...
420
오토-인코더
421
422
423
인코더
디코더
$\hat{x}$
z
426
424
코드 모델
P(z)
산술 코더
수신 디바이스

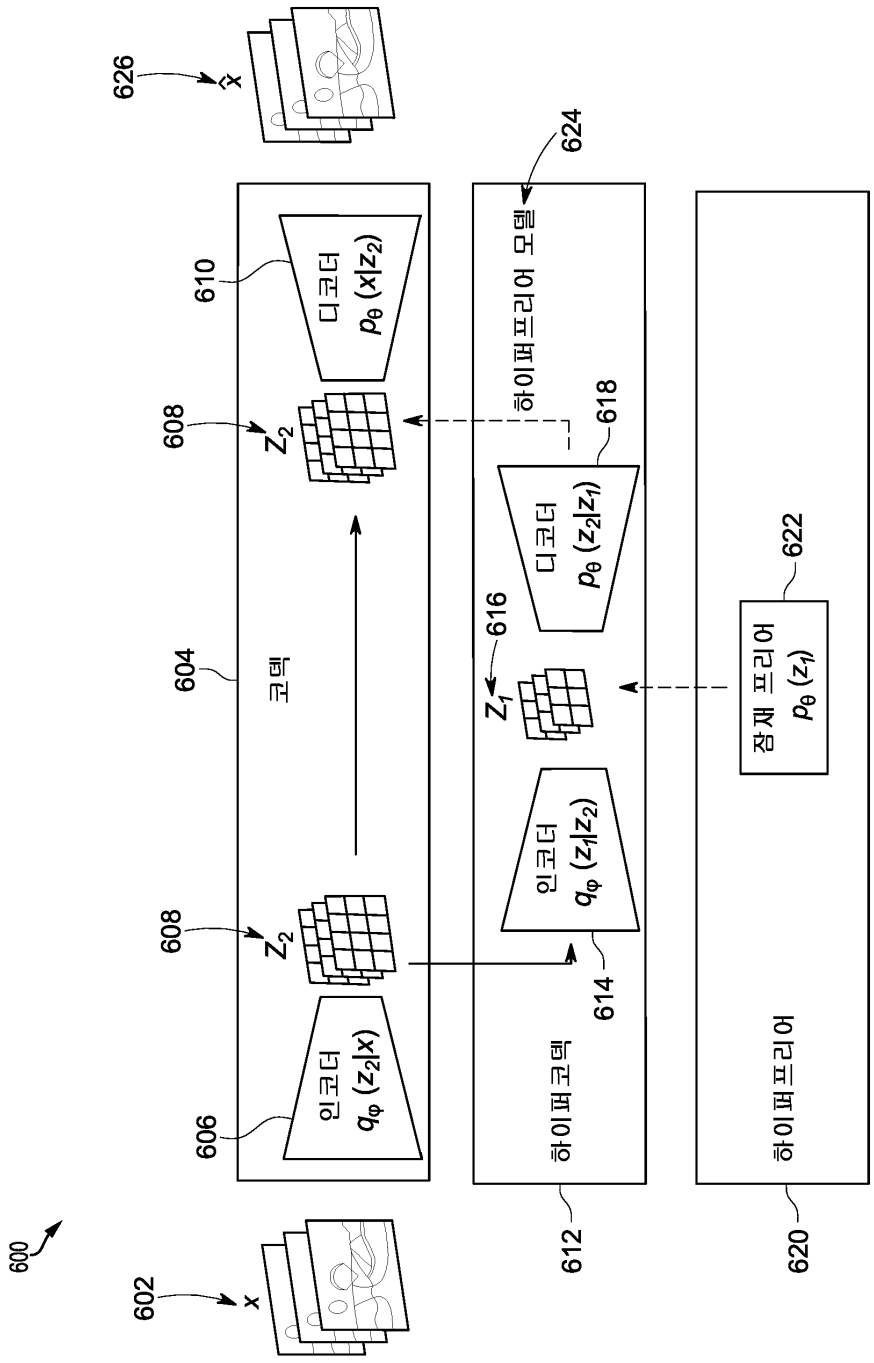
도면5a


500
501
x_i
502
인코더
$q_\phi(z|x)$
z_i
504
506
잠재 프리어
$p_\psi(z)$
508
AE
510
1|0|0|1|0
전송자
수신기
1|0|0|1|0
b_z'
512
AD
506
잠재 프리어
$p_\psi(z)$
514
디코더
$p_\theta(x|z)$
z_i
504
516
$\hat{x}_i$
공유된 지식

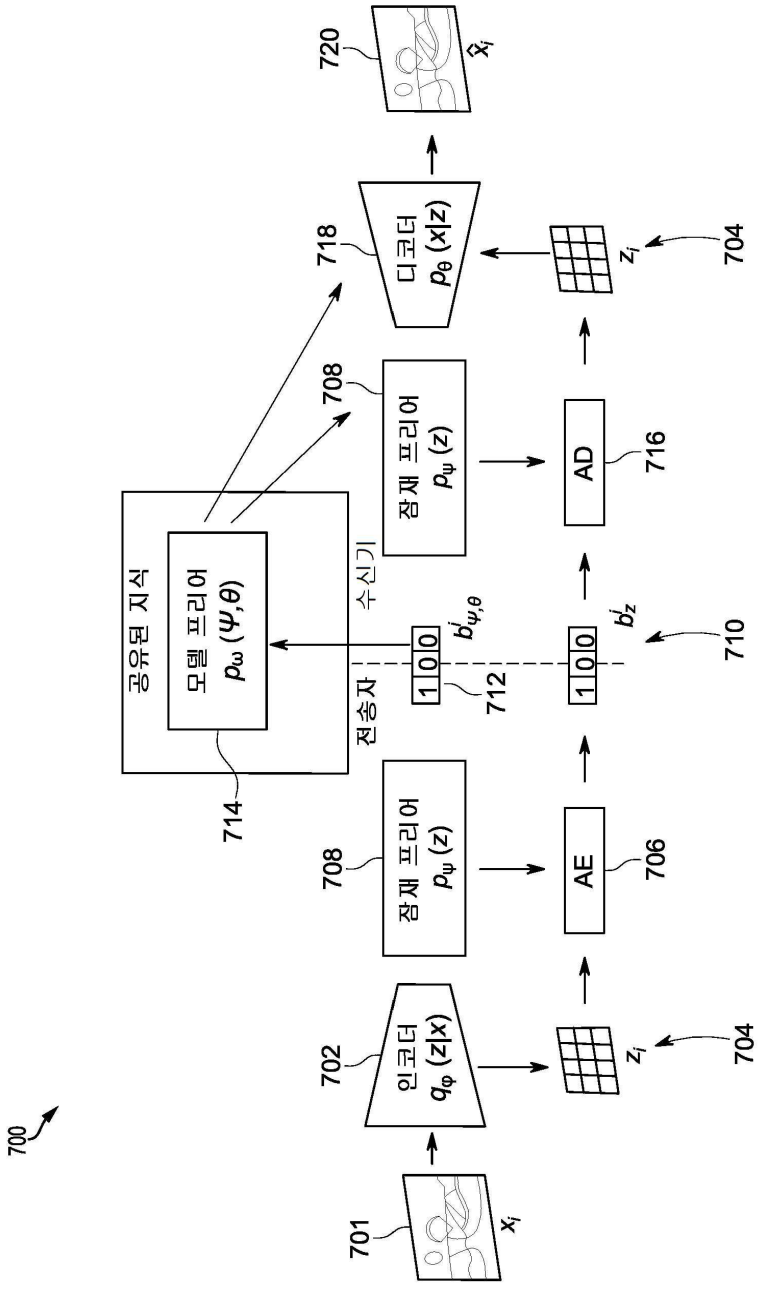
도면5b


530
501
x_i
502
인코더 $q_\phi(z|x)$
z_i
504
506
잠재 프리어 $p_\psi(z)$
514
디코더 $p_\theta(x|z)$
516
$\hat{x}_i$
공유된 지식
AE
508
AD
512
전송자
수신기
1|0|0|1|0|0|1|1|1|0
b_z
520

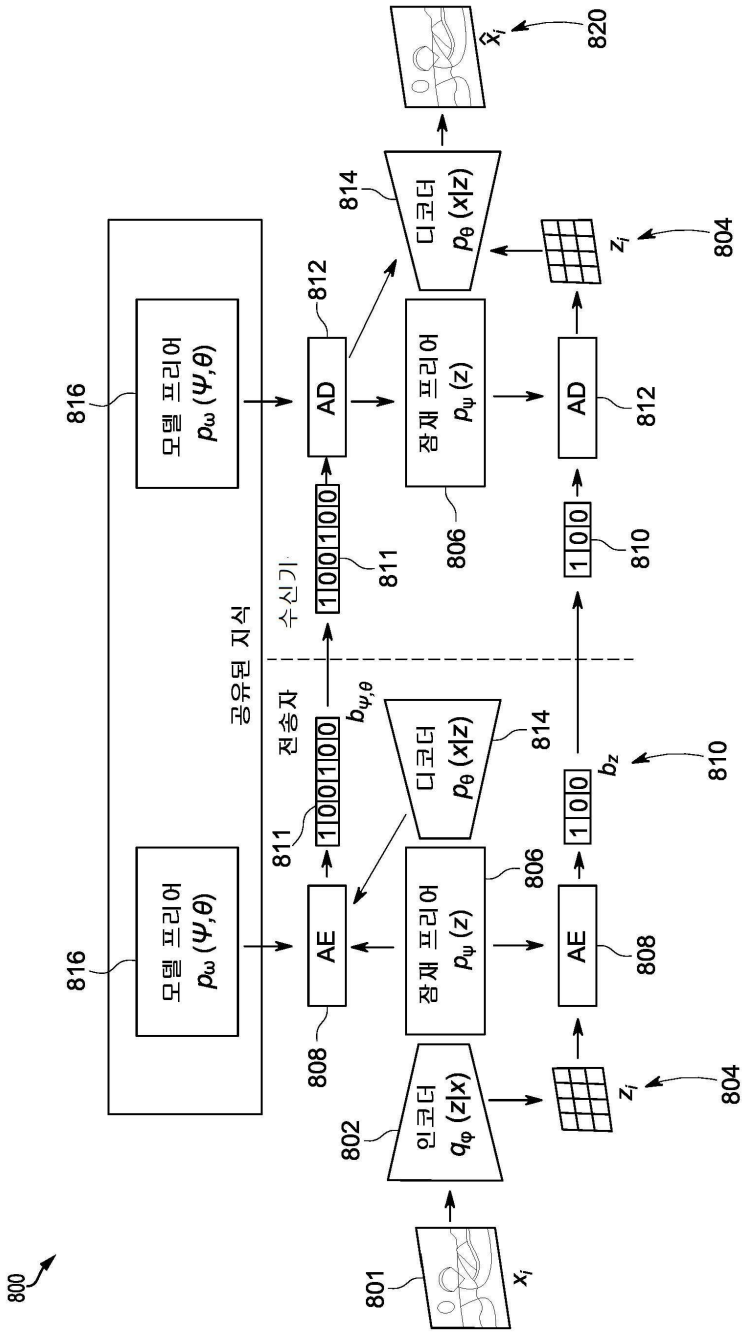
도면6


600
602
x
606
인코더
$q_\phi(z_2|x)$
608
z_2
604
코덱
610
디코더
$p_\theta(x|z_2)$
626
$\hat{x}$
612
하이퍼코덱
614
인코더
$q_\phi(z_1|z_2)$
616
z_1
618
디코더
$p_\theta(z_2|z_1)$
624
하이퍼프리어 모델
620
하이퍼프리어
622
잠재 프리어
$p_\theta(z_1)$

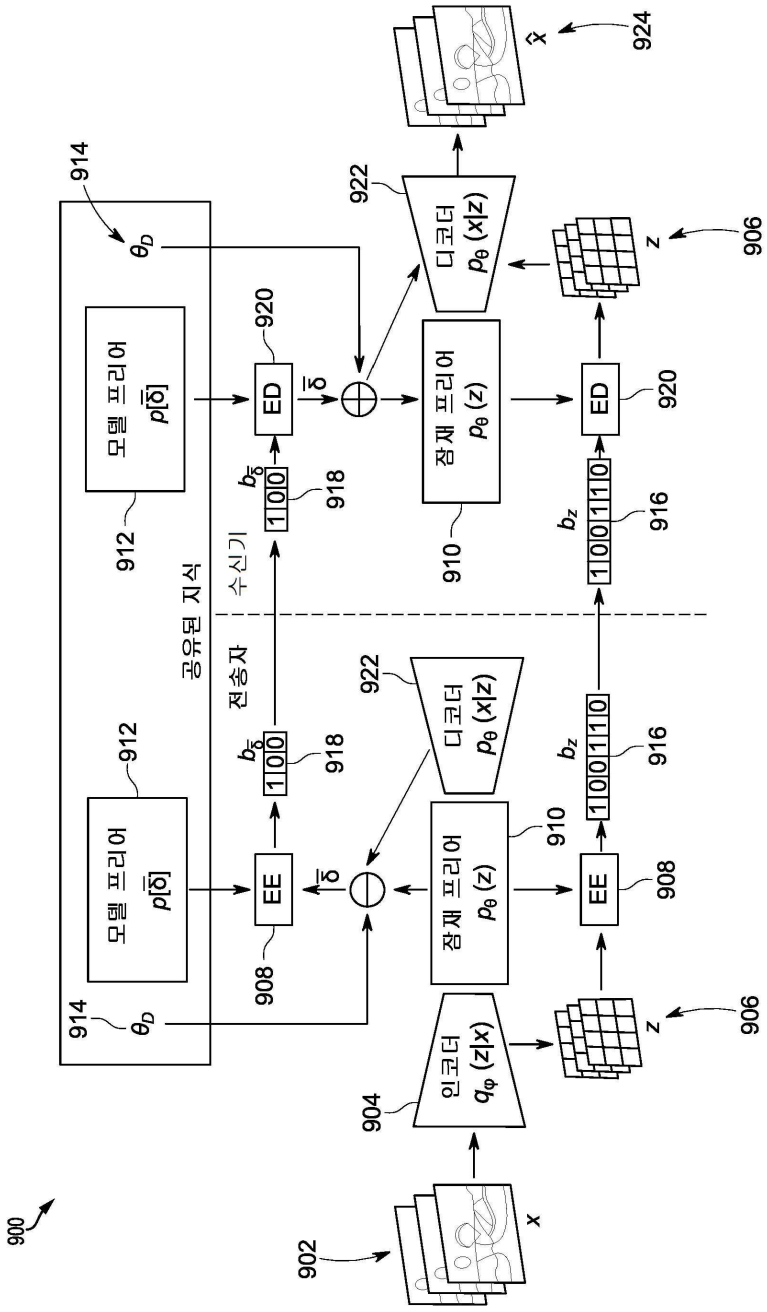
도면7


700
701
x_i
인코더 $q_\phi(z|x)$
702
z_i
704
잠재 프리어 $p_\psi(z)$
708
AE
706
100
712
$b^i_{\psi,\theta}$
b^i_z
710
AD
716
잠재 프리어 $p_\psi(z)$
708
디코더 $p_\theta(x|z)$
718
z_i
704
$\hat{x}_i$
720
공유된 지식
모델 프리어 $p_\omega(\Psi,\theta)$
714
전송자
수신기

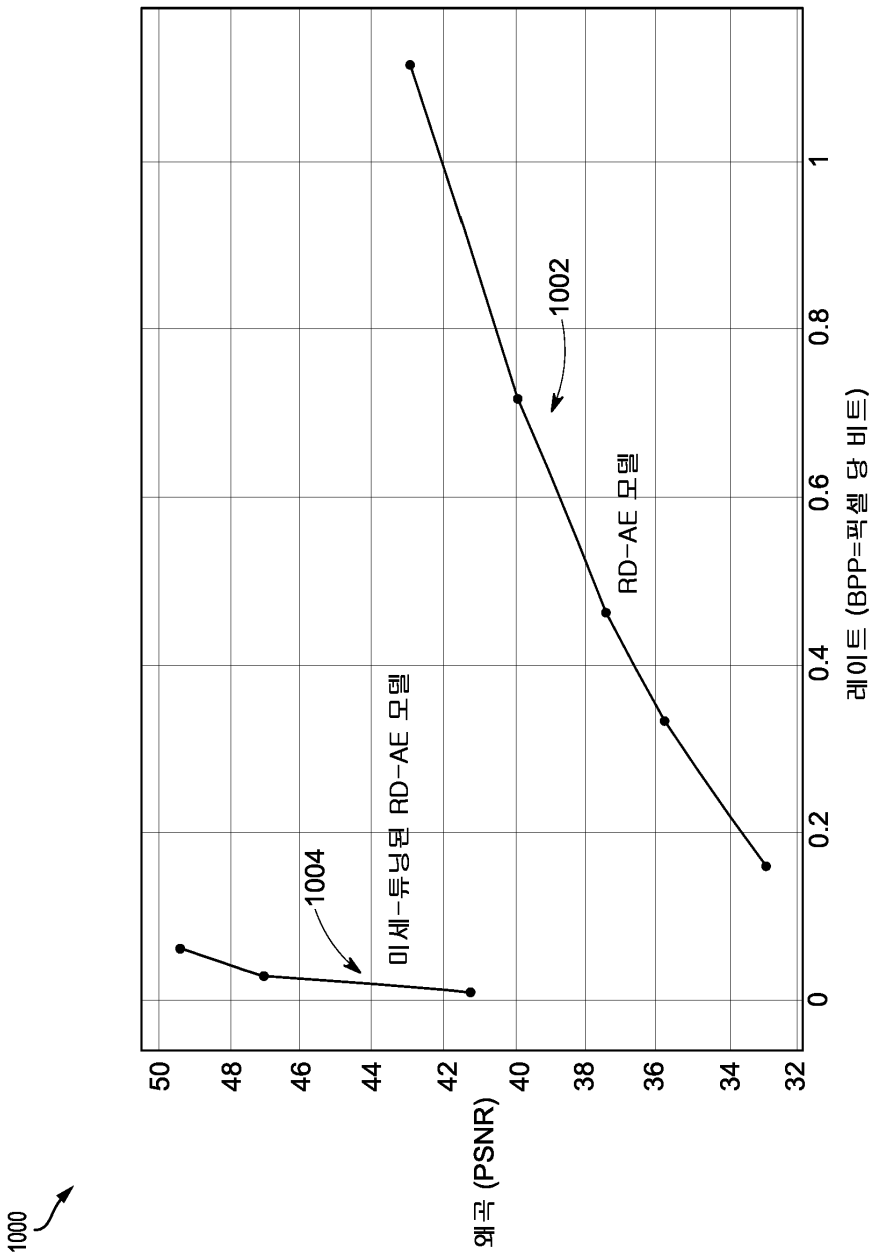
도면8


800
801
x_i
인코더 $q_\phi(z|x)$
802
808
AE
804
z_i
잠재 프리어 $p_\psi(z)$
806
디코더 $p_\theta(x|z)$
814
$\hat{x}_i$
820
모델 프리어 $p_\omega(\psi,\theta)$
816
공유된 지식
전송자
수신기
1 1 0 0 1 0 0
811
$b_{\psi,\theta}$
1 0 0
b_z
810
AD
812

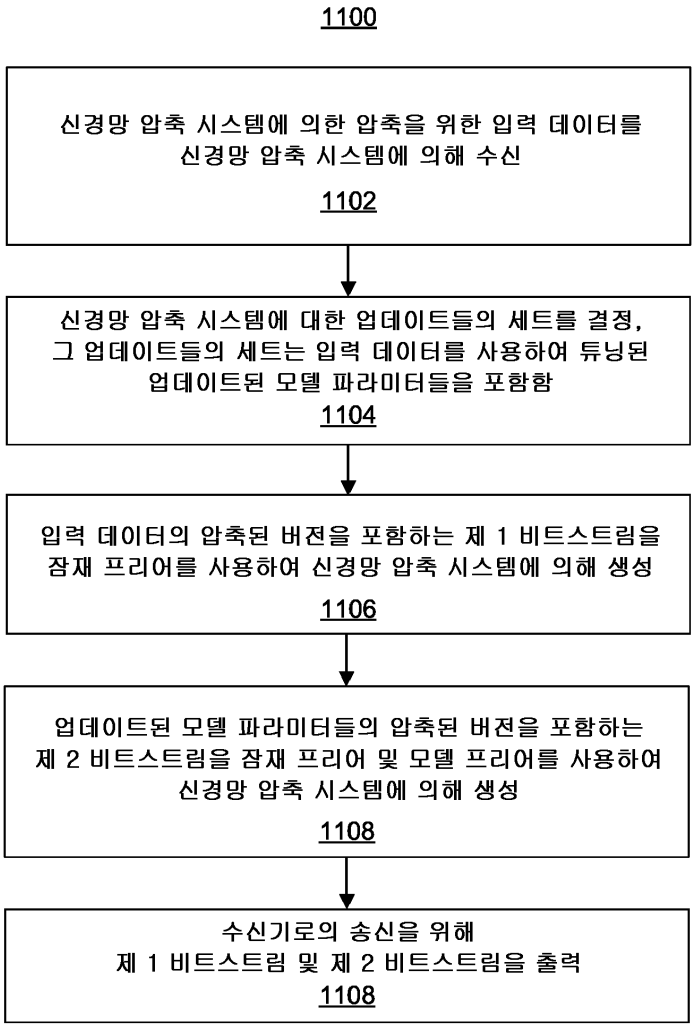
도면9


900
902
x
904
인코더 $q_\phi(z|x)$
906
z
908
EE
910
잠재 프리어 $p_\theta(z)$
912
모델 프리어 $p[\delta]$
914
θ_D
공유된 지식
전송자
수신기
916
b_z
1100|1110
918
b_δ
1100
920
ED
922
디코더 $p_\theta(x|z)$
924
$\hat{x}$

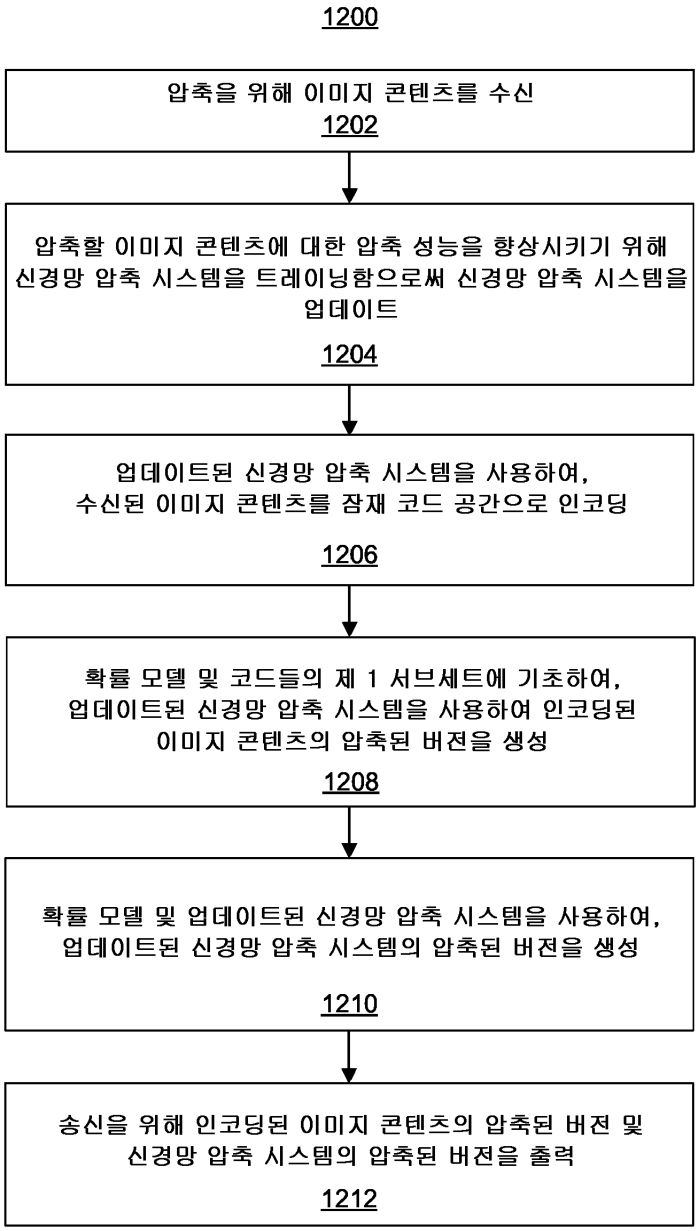
도면10


1000
왜곡 (PSNR)
50
48
46
44
42
40
38
36
34
32
0
0.2
0.4
0.6
0.8
1
레이트 (BPP=픽셀 당 비트)
1002
RD-AE 모델
1004
미세-튜닝된 RD-AE 모델

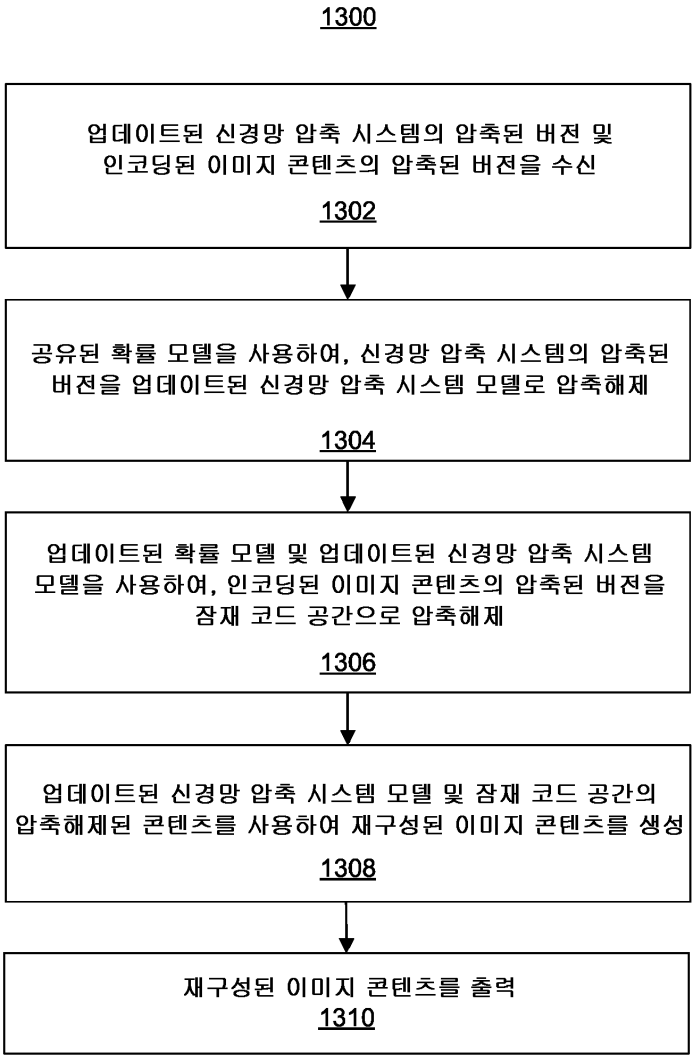
도면11


1100
신경망 압축 시스템에 의한 압축을 위한 입력 데이터를 신경망 압축 시스템에 의해 수신
1102
신경망 압축 시스템에 대한 업데이트들의 세트를 결정, 그 업데이트들의 세트는 입력 데이터를 사용하여 튜닝된 업데이트된 모델 파라미터들을 포함함
1104
입력 데이터의 압축된 버전을 포함하는 제 1 비트스트림을 잠재 프리어를 사용하여 신경망 압축 시스템에 의해 생성
1106
업데이트된 모델 파라미터들의 압축된 버전을 포함하는 제 2 비트스트림을 잠재 프리어 및 모델 프리어를 사용하여 신경망 압축 시스템에 의해 생성
1108
수신기로의 송신을 위해 제 1 비트스트림 및 제 2 비트스트림을 출력
1108

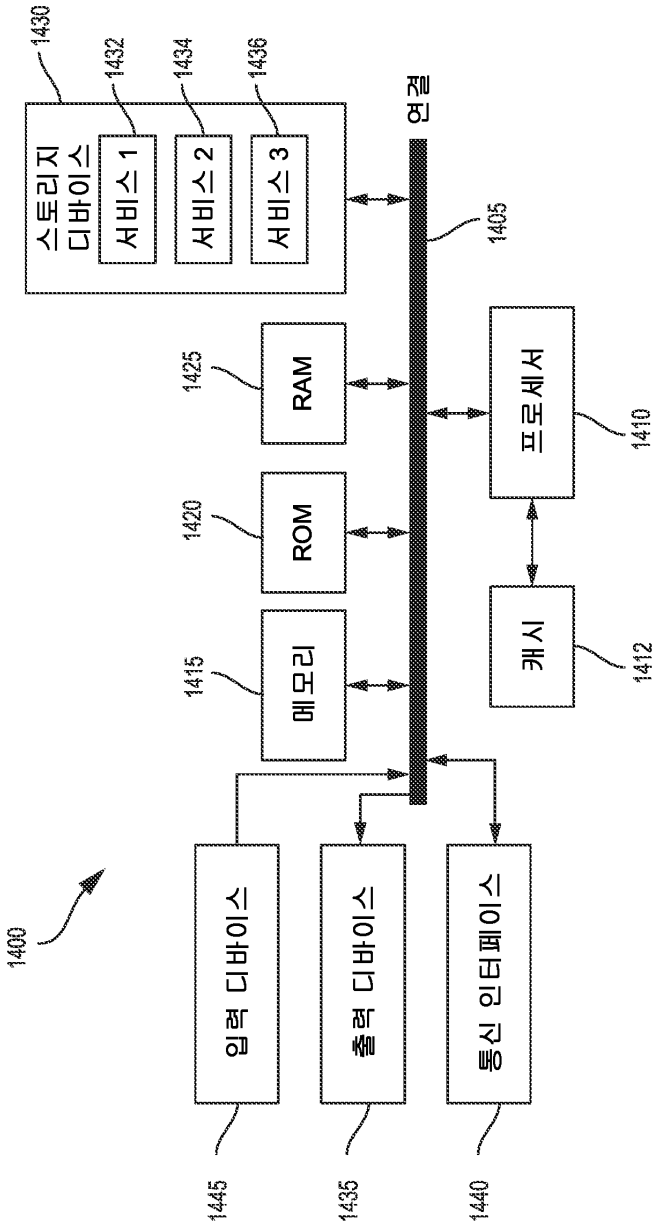
도면12


1200
압축을 위해 이미지 콘텐츠를 수신
1202
압축할 이미지 콘텐츠에 대한 압축 성능을 향상시키기 위해 신경망 압축 시스템을 트레이닝함으로써 신경망 압축 시스템을 업데이트
1204
업데이트된 신경망 압축 시스템을 사용하여, 수신된 이미지 콘텐츠를 잠재 코드 공간으로 인코딩
1206
확률 모델 및 코드들의 제 1 서브세트에 기초하여, 업데이트된 신경망 압축 시스템을 사용하여 인코딩된 이미지 콘텐츠의 압축된 버전을 생성
1208
확률 모델 및 업데이트된 신경망 압축 시스템을 사용하여, 업데이트된 신경망 압축 시스템의 압축된 버전을 생성
1210
송신을 위해 인코딩된 이미지 콘텐츠의 압축된 버전 및 신경망 압축 시스템의 압축된 버전을 출력
1212

도면13


1300
업데이트된 신경망 압축 시스템의 압축된 버전 및 인코딩된 이미지 콘텐츠의 압축된 버전을 수신
1302
공유된 확률 모델을 사용하여, 신경망 압축 시스템의 압축된 버전을 업데이트된 신경망 압축 시스템 모델로 압축해제
1304
업데이트된 확률 모델 및 업데이트된 신경망 압축 시스템 모델을 사용하여, 인코딩된 이미지 콘텐츠의 압축된 버전을 잠재 코드 공간으로 압축해제
1306
업데이트된 신경망 압축 시스템 모델 및 잠재 코드 공간의 압축해제된 콘텐츠를 사용하여 재구성된 이미지 콘텐츠를 생성
1308
재구성된 이미지 콘텐츠를 출력
1310

도면14


1400
1430
스토리지 디바이스
1432
서비스 1
1434
서비스 2
1436
서비스 3
버스
1405
1425
RAM
1420
ROM
1415
메모리
프로세서
1410
캐시
1412
입력 디바이스
출력 디바이스
통신 인터페이스
1445
1435
1440