



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2023년08월24일

(11) 등록번호 10-2570706

(24) 등록일자 2023년08월22일

(51) 국제특허분류(Int. Cl.)
G06F 18/00 (2023.01) G06N 99/00 (2019.01)
G06V 10/40 (2022.01)

(52) CPC특허분류
G06F 18/2413 (2023.01)
G06F 18/213 (2023.01)

(21) 출원번호 10-2018-7009144

(22) 출원일자(국제) 2016년08월04일

심사청구일자 2021년07월19일

(85) 번역문제출일자 2018년03월30일

(65) 공개번호 10-2018-0048930

(43) 공개일자 2018년05월10일

(86) 국제출원번호 PCT/US2016/045636

(87) 국제공개번호 WO 2017/039946

국제공개일자 2017년03월09일

(30) 우선권주장

62/213,591 2015년09월02일 미국(US)

15/077,873 2016년03월22일 미국(US)

(56) 선행기술조사문헌

US20150139559 A1*

Adam Coates et. al., Proceedings of the 28th
International Conference on International
Conference on Machine LearningJune, Pages
921-928, 2011*

US20030139929 A1

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

퀄컴 인코포레이티드

미국 92121-1714 캘리포니아주 샌 디에고 모어하
우스 드라이브 5775

(72) 발명자

마쭈다르 솜텝

미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775

토월 레전 블라이스

미국 92121-1714 캘리포니아주 샌디에고 모어하
우스 드라이브 5775

(74) 대리인

특허법인코리아나

전체 청구항 수 : 총 18 항

심사관 : 강현일

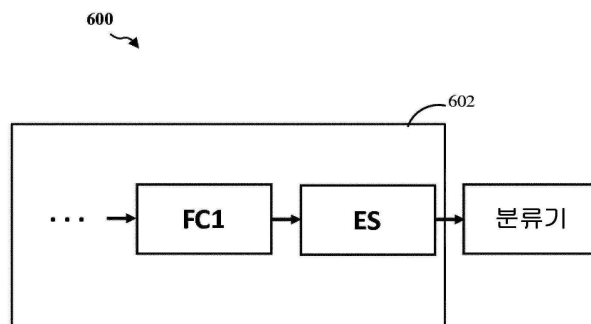
(54) 발명의 명칭 분류를 위한 강제된 회소성

(57) 요약

입력을 분류하기 위한 장치는 분류기 및 특징 추출기를 포함한다. 특징 추출기는 입력에 기초하여 특징 벡터를 생성하도록 구성된다. 특징 벡터는 또한 회소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하도록 구성된다. 회소한 특징 벡터는 특징 추출기에 의해 생성된 특징 벡터와 동일한 차원들

(뒷면에 계속)

대표도



을 갖는다. 하지만, 희소한 특징 벡터는 특징 추출기에 의해 생성된 특징 벡터보다 적은 년-제로 엘리먼트들을 포함한다. 특징 벡터는 또한 입력을 분류하기 위해 희소한 특징 벡터를 분류기로 포워딩하도록 구성된다.

(52) CPC특허분류

G06N 20/00 (2021.08)

G06V 10/454 (2023.08)

G06Y 10/50 (2018.01)

명세서

청구범위

청구항 1

장치로서,

메모리; 및

상기 메모리에 연결된 적어도 하나의 프로세서를 포함하고,

상기 적어도 하나의 프로세서는:

훈련된 인공 신경망에 포함되는 적어도 하나의 필터로 입력을 컨볼루션함으로써 상기 메모리에 저장된 상기 입력으로부터 특징들을 추출하고;

추출된 상기 특징들로부터 초기 특징 벡터를 생성하는 것으로서, 상기 초기 특징 벡터는 복수의 엘리먼트들을 포함하는, 상기 초기 특징 벡터를 생성하고;

상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하는 것으로서, 상기 유지되는 엘리먼트들의 세트는 엘리먼트들의 정량을 포함하고, 상기 정량은 성능 메트릭 및 상기 초기 특징 벡터에서의 상기 복수의 엘리먼트들 각각의 값에 기초하여 결정되고, 상기 유지되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값은 상기 제외되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값보다 큰, 상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하고;

상기 제외되는 엘리먼트들의 세트의 엘리먼트 각각의 값을 제로로 설정하고;

상기 유지되는 엘리먼트들의 세트로부터 희소한 (sparse) 특징 벡터를 생성하는 것으로서, 상기 희소한 특징 벡터에 의해 소비되는 상기 메모리의 양이 상기 초기 특징 벡터에 의해 소비되는 상기 메모리의 양보다 적도록, 상기 희소한 특징 벡터는 상기 초기 특징 벡터와 동일한 차원을 갖고, 상기 초기 특징 벡터보다 적은 нену엘 엘리먼트들을 포함하는, 상기 희소한 특징 벡터를 생성하고;

상기 메모리에 상기 희소한 특징 벡터를 저장하고; 및

상기 메모리에 저장된 상기 희소한 특징 벡터에 기초하여 상기 입력을 분류하도록 구성된, 장치.

청구항 2

제 1 항에 있어서,

상기 적어도 하나의 프로세서는 또한 상기 성능 메트릭을 온-디바이스로 컴퓨팅하도록 구성되는, 장치.

청구항 3

제 1 항에 있어서,

상기 적어도 하나의 프로세서는 또한 상기 성능 메트릭을 오프-디바이스로 컴퓨팅하도록 구성되는, 장치.

청구항 4

제 1 항에 있어서,

상기 적어도 하나의 프로세서는 유지되는 상기 엘리먼트들의 정량을 결정하도록 훈련되는, 장치.

청구항 5

제 4 항에 있어서,

상기 적어도 하나의 프로세서는 상기 특징 벡터의 낮은 희소성을 페널라이징하는 비용 함수를 이용하여 훈련되는, 장치.

청구항 6

제 5 항에 있어서,

상기 비용 함수는 L1 norm 또는 L0 norm을 포함하는, 장치.

청구항 7

제 1 항에 있어서,

상기 적어도 하나의 프로세서는 또한 상기 최소한 특징 벡터의 엘리먼트들을 이진화하도록 구성되는, 장치.

청구항 8

제 1 항에 있어서,

상기 적어도 하나의 프로세서는 또한 상기 최소한 특징 벡터의 엘리먼트들을 양자화하도록 구성되는, 장치.

청구항 9

방법으로서,

훈련된 인공 신경망에 포함되는 적어도 하나의 필터로 입력을 컨볼루션함으로써 상기 훈련된 인공 신경망을 실행하는 디바이스의 메모리에 저장된 상기 입력으로부터 특징들을 추출하는 단계;

추출된 상기 특징들로부터 초기 특징 벡터를 생성하는 단계로서, 상기 초기 특징 벡터는 복수의 엘리먼트들을 포함하는, 상기 초기 특징 벡터를 생성하는 단계;

상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하는 단계로서, 상기 유지되는 엘리먼트들의 세트는 성능 메트릭 및 상기 초기 특징 벡터에서의 상기 복수의 엘리먼트들 각각의 값에 기초하여 결정되는 엘리먼트들의 정량을 포함하고, 상기 유지되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값은 상기 제외되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값보다 큰, 상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하는 단계;

상기 제외되는 엘리먼트들의 세트의 엘리먼트 각각의 값을 제로로 설정하는 단계;

상기 유지되는 엘리먼트들의 세트로부터 최소한 (sparse) 특징 벡터를 생성하는 단계로서, 상기 최소한 특징 벡터에 의해 소비되는 상기 메모리의 양이 상기 초기 특징 벡터에 의해 소비되는 상기 메모리의 양보다 적도록, 상기 최소한 특징 벡터는 상기 초기 특징 벡터와 동일한 차원을 갖고, 상기 초기 특징 벡터보다 적은 non-zero 엘리먼트들을 포함하는, 상기 최소한 특징 벡터를 생성하는 단계;

상기 메모리에 상기 최소한 특징 벡터를 저장하는 단계; 및

상기 메모리에 저장된 상기 최소한 특징 벡터에 기초하여 상기 입력을 분류하는 단계를 포함하는, 방법.

청구항 10

제 9 항에 있어서,

상기 성능 메트릭을 온-디바이스로 컴퓨팅하는 단계를 더 포함하는, 방법.

청구항 11

제 9 항에 있어서,

상기 성능 메트릭을 오프-디바이스로 컴퓨팅하는 단계를 더 포함하는, 방법.

청구항 12

제 9 항에 있어서,

훈련에 기초하여 상기 엘리먼트들의 정량을 결정하는 단계를 더 포함하는, 방법.

청구항 13

제 12 항에 있어서,

상기 결정하는 단계는 상기 특징 벡터의 낮은 희소성을 페널라이징하는 비용 함수를 이용하여 특징 추출기를 훈련시키는 단계를 포함하는, 방법.

청구항 14

제 13 항에 있어서,

상기 비용 함수는 L1 norm 또는 L0 norm을 포함하는, 방법.

청구항 15

제 9 항에 있어서,

상기 희소한 특징 벡터의 엘리먼트들을 이진화하는 단계를 더 포함하는, 방법.

청구항 16

제 9 항에 있어서,

상기 희소한 특징 벡터의 엘리먼트들을 양자화하는 단계를 더 포함하는, 방법.

청구항 17

장치로서,

훈련된 인공 신경망에 포함되는 적어도 하나의 필터로 입력을 컨볼루션함으로써 상기 훈련된 인공 신경망을 실행하는 상기 장치의 메모리에 저장된 상기 입력으로부터 특징들을 추출하는 수단;

추출된 상기 특징들로부터 초기 특징 벡터를 생성하는 수단으로서, 상기 초기 특징 벡터는 복수의 엘리먼트들을 포함하는, 상기 초기 특징 벡터를 생성하는 수단;

상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하는 단계로서, 상기 유지되는 엘리먼트들의 세트는 성능 메트릭 및 상기 초기 특징 벡터에서의 상기 복수의 엘리먼트들 각각의 값에 기초하여 결정되는 엘리먼트들의 정량을 포함하고, 상기 유지되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값은 상기 제외되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값보다 큰, 상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하는 수단;

상기 제외되는 엘리먼트들의 세트의 엘리먼트 각각의 값을 제로로 설정하는 수단;

상기 유지되는 엘리먼트들의 세트로부터 희소한 (sparse) 특징 벡터를 생성하는 수단으로서, 상기 희소한 특징 벡터에 의해 소비되는 상기 메모리의 양이 상기 초기 특징 벡터에 의해 소비되는 상기 메모리의 양보다 적도록, 상기 희소한 특징 벡터는 상기 초기 특징 벡터와 동일한 차원을 갖고, 상기 초기 특징 벡터보다 적은 non-zero 엘리먼트들을 포함하는, 상기 희소한 특징 벡터를 생성하는 수단;

상기 메모리에 상기 희소한 특징 벡터를 저장하는 수단; 및

상기 메모리에 저장된 상기 희소한 특징 벡터에 기초하여 상기 입력을 분류하는 수단을 포함하는, 장치.

청구항 18

프로그램 코드가 인코딩되어 있는 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 프로그램 코드는 프로세서에 의해 실행되고: 그리고

상기 프로그램 코드는,

훈련된 인공 신경망에 포함되는 적어도 하나의 필터로 입력을 컨볼루션함으로써 상기 훈련된 인공 신경망을 실행하는 디바이스의 메모리에 저장된 상기 입력으로부터 특징들을 추출하기 위한 프로그램 코드;

추출된 상기 특징들로부터 초기 특징 벡터를 생성하기 위한 프로그램 코드로서, 상기 초기 특징 벡터는 복수의 엘리먼트들을 포함하는, 상기 초기 특징 벡터를 생성하기 위한 프로그램 코드;

상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하기 위한 프로그램 코드로서, 상기 유지되는 엘리먼트들의 세트는 성능 메트릭 및 상기 초기 특징 벡터에서의 상기 복수의 엘리먼트들 각각의 값에 기초하여 결정되는 엘리먼트들의 정량을 포함하고, 상기 유지되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값은 상기 제외되는 엘리먼트들의 세트에서의 엘리먼트 각각의 값보다 큰, 상기 복수의 엘리먼트들 중 유지되는 엘리먼트들의 세트 및 상기 복수의 엘리먼트들 중 제외되는 엘리먼트들의 대응하는 세트를 식별하기 위한 프로그램 코드;

상기 제외되는 엘리먼트들의 세트의 엘리먼트 각각의 값을 제로로 설정하기 위한 프로그램 코드;

상기 유지되는 엘리먼트들의 세트로부터 희소한 (sparse) 특징 벡터를 생성하기 위한 프로그램 코드로서, 상기 희소한 특징 벡터에 의해 소비되는 상기 메모리의 양이 상기 초기 특징 벡터에 의해 소비되는 상기 메모리의 양보다 적도록, 상기 희소한 특징 벡터는 상기 초기 특징 벡터와 동일한 차원을 갖고, 상기 초기 특징 벡터보다 적은 non-zero 엘리먼트들을 포함하는, 상기 희소한 특징 벡터를 생성하기 위한 프로그램 코드;

상기 메모리에 상기 희소한 특징 벡터를 저장하기 위한 프로그램 코드; 및

상기 메모리에 저장된 상기 희소한 특징 벡터에 기초하여 상기 입력을 분류하기 위한 프로그램 코드를 포함하는, 비일시적 컴퓨터 판독가능 저장 매체.

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

발명의 설명

기술 분야

[0001] 관련 출원에 대한 상호 참조

[0002] 본 출원은 2015 년 9 월 2 일에 출원되고 발명의 명칭이 "ENFORCED SPARSITY FOR CLASSIFICATION" 인 미국 특허 가출원 제62/213,591호의 이익을 주장하며, 그 개시는 그 전체가 참조로써 본원에 명시적으로 인용된다.

[0003] 기술분야

[0004] 본 개시의 소정의 양태들은 일반적으로 머신 학습에 관한 것이고, 더 구체적으로는 피쳐 추출 및 분류의 시스템 및 방법을 개선시키는 것에 관한 것이다.

배경 기술

[0005] 인공 뉴런들 (즉, 뉴런 모델) 의 상호접속된 그룹을 포함할 수도 있는 인공 신경망은 계산 디바이스이거나 또는 계산 디바이스에 의해 수행될 방법을 나타낸다.

[0006] 컨볼루션 신경망 (convolutional neural network) 은 일종의 피드 포워드 (feed-forward) 인공 신경망이다. 컨볼루션 신경망은, 각각 수용 영역을 갖고 입력 공간을 집합 적으로 타일링하는 뉴런 집합을 포함할 수도 있다. 컨볼루션 신경망 (CNN) 은 수많은 응용들을 갖는다. 특히, CNN은 패턴 인식 및 분류 분야에서 광범위하게 사용되어 왔다.

[0007] 심층 신뢰 망 (deep belief network) 및 심층 컨볼루션 신경망과 같은 심층 학습 아키텍처들은, 제 1 계층의 뉴런 출력이 제 2 계층의 뉴런에 대한 입력이 되고, 제 2 계층의 뉴런 출력이 제 3 계층의 뉴런에 대한 입력이 되는 등등의 계층형 신경망 아키텍처이다. 심층 신경망은 특징들의 체계 (hierarchy) 를 인식하도록 훈련될 수도 있으며, 따라서 객체 인식 응용들에서 점점 더 많이 사용되고 있다. 컨볼루션 신경망과 같이, 이러한 심층 학습 아키텍처에서의 계산은, 하나 이상의 계산 체인에서 구성될 수도 있는 처리 노드의 집단 상에 분산될 수도 있다. 이러한 다계층 아키텍처는 한 번에 하나의 계층씩 훈련될 수도 있으며 역 전파 (back propagation) 를 사용하여 미세 조정 (fine-tune) 될 수도 있다.

[0008] 다른 모델들이 또한 객체 인식을 위해 이용가능하다. 예를 들어 서포트 벡터 머신 (SVM) 은, 분류에 적용될 수 있는 학습 도구이다. 서포트 벡터 머신은, 데이터를 카테고리화하는 분리 초평면 (예 : 결정 경계) 를 포함한다. 초평면은 지도 학습 (supervised learning) 에 의해 정의된다. 원하는 초평면은 훈련 데이터의 마진 (margin) 을 증가시킨다. 즉, 초평면은 훈련 예들까지의 최소 거리가 가장 커야 한다.

[0009] 이러한 솔루션들은 여러 가지 분류 벤치마크에 대해 우수한 결과를 달성하지만, 계산상의 복잡성이 엄청나게 클 수 있다. 또한, 모델들의 훈련이 어려울 수도 있다.

발명의 내용

[0010] 본 개시의 일 양태에서, 입력을 분류하기 위한 장치가 개시된다. 장치는 분류기 및 특징 추출기를 포함한다. 특징 추출기는 입력으로부터 특징 벡터를 생성하도록 구성된다. 특징 벡터는 또한 희소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하도록 구성된다. 희소한 특징 벡터는 특징 추출기에 의해 생성된 특징 벡터와 동일한 차원을 갖는다. 희소한 특징 벡터는 특징 추출기에 의해 생성된 특징 벡터보다 적은 난-제로 엘리먼트들을 포함한다. 특징 벡터는 또한 입력을 분류하기 위해 희소한 특징 벡터를 분류기로 포워딩하도록 구성된다.

[0011] 본 개시의 다른 양태에서, 입력을 분류하기 위한 방법이 개시된다. 그 방법은 입력으로부터 특징 벡터를 생성하는 단계를 포함한다. 그 방법은 또한 희소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하는 단계를 포함한다. 희소한 특징 벡터는 생성된 특징 벡터와 동일한 차원을 갖는다. 희소한 특징 벡터는 또한 생성된 특징 벡터보다 적은 난-제로 엘리먼트들을 포함한다. 그 방법은 입력을 분류하기 위해 희소한 특징 벡터를 분류기로 포워딩하는 단계를 더 포함한다.

[0012] 본 개시의 또 다른 양태에서, 입력을 분류하기 위한 장치가 개시된다. 그 장치는 입력으로부터 특징 벡터를 생성하는 수단을 포함한다. 그 장치는 또한 희소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하는 수단을 포함한다. 희소한 특징 벡터는 생성된 특징 벡터와 동일한 차원을 갖는다. 희소한 특징 벡터는 또한 생성된 특징 벡터보다 적은 난-제로 엘리먼트들을 포함한다. 그 장치는 입력을 분류하기 위해 희소한 특징 벡터를 분류기로 포워딩하는 수단을 더 포함한다.

[0013] 본 개시의 또 다른 양태에서, 비일시적 컴퓨터 판독가능 매체가 제시된다. 비일시적 컴퓨터 판독가능 매체에는 분류를 위한 프로그램 코드가 인코딩되어 있다. 프로그램 코드는 프로세서에 의해 실행되며, 입력으로부터 특징 벡터를 생성하기 위한 프로그램 코드를 포함한다. 프로그램 코드는 또한 희소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하기 위한 프로그램 코드를 포함한다. 희소한 특징 벡터는 생성된 특징 벡터와 동일한 차원을 갖는다. 희소한 특징 벡터는 또한 생성된 특징 벡터보다 적은 난-제로 엘리먼트들을 포함한다. 프로그램 코드는 입력을 분류하기 위해 희소한 특징 벡터를 분류기로 포워딩하기 위한 프로그램 코드를 더 포함한다.

[0014] 본 개시의 추가 특징 및 이점들은 아래에서 설명될 것이다. 본 개시는 본 개시의 동일한 목적을 수행하기 위한 다른 구조들을 수정 및 설계하기 위한 기초로서 손쉽게 이용될 수도 있다는 것이 당업자에 의해 인식되어야 한다. 또한, 그러한 동등한 구성들은 첨부된 청구항에 제시된 본 개시의 교시로부터 벗어나지 않는다는 것이 당업자에 의해 인식되어야 한다. 다른 목적 및 이점들과 함께, 조직 및 동작 방법 양자 모두에 대하여, 본 개시의 특성인 것으로 생각되는 신규한 특징들은, 첨부 도면들과 관련하여 고려될 때 다음의 설명으로부터 보다 잘 이해될 것이다. 그러나, 도면들의 각각은 예시 및 설명의 목적만을 위해 제공되고 본 개시의 제한들의 정의로서 의도되지 않는다는 것이 명백히 이해되어야 한다.

도면의 간단한 설명

[0015] 본 개시의 특징, 성질 및 이점들이 도면들과 함께 취해질 때 아래에 제시된 상세한 설명으로부터 더 분명해질

것이고, 도면들에서 같은 참조 부호는 전체에 걸쳐 대응하여 동일시된다.

도 1은 본 개시의 특정 양태들에 따른 범용 프로세서를 포함하는, 시스템 온 칩 (SOC) 을 이용하여 신경망을 설계하는 예시적 구현을 나타낸다.

도 2는 본 개시의 양태들에 따른 시스템의 예시적 구현을 나타낸다.

도 3a는 본 개시의 양태들에 따른 신경망을 나타내는 도면이다.

도 3b는 본 개시의 양태들에 따른 예시적인 심층 컨볼루션망 (DCN) 을 나타내는 블록도이다.

도 4는 본 개시의 양태들에 따라 인공 지능 (AI) 기능을 모듈화할 수도 있는 예시적인 소프트웨어 아키텍처를 나타내는 블록도이다.

도 5는 본 개시의 양태들에 따라 스마트폰 상에서 AI 애플리케이션의 런타임 동작을 나타내는 블록도이다.

도 6 은 본 개시의 양태들에 따른 특정 추출기를 포함하는 예시적인 머신 학습 모델을 나타내는 블록도이다.

도 7 은 본 개시의 양태들에 따른 특정 추출 및 입력 분류를 위한 방법을 나타낸다.

발명을 실시하기 위한 구체적인 내용

- [0016] 첨부된 도면과 관련하여 아래에 제시되는 상세한 설명은 다양한 구성들의 설명으로서 의도된 것이며 본원에 설명된 개념들이 실시될 수도 있는 구성들만을 나타내도록 의도된 것은 아니다. 상세한 설명은 다양한 개념들의 완전한 이해를 제공하는 목적을 위해 특정 상세들을 포함한다. 하지만, 이들 개념들은 이들 특정 상세들 없이 실시될 수도 있음이 당업자에게 분명할 것이다. 일부 실례에서, 잘 알려진 구조 및 컴포넌트들은 그러한 개념들을 모호하게 하는 것을 피하기 위해서 블록도 형태로 도시된다.
- [0017] 교시들에 기초하여 당업자는, 본 개시의 범위가, 본 개시의 임의의 양태를, 본 개시의 임의의 다른 양태와 독립적으로 또는 조합되든지 간에, 커버하도록 의도된다는 것이 인식되어야 한다. 예를 들어, 제시된 임의의 수의 양태들을 이용하여 장치가 구현될 수도 있거나 또는 방법이 실시될 수도 있다. 또한, 본 개시의 범위는 제시된 본 개시의 다양한 양태들 외에 또는 추가하여 다른 구조, 기능, 또는 구조 및 기능을 이용하여 실시되는 그러한 장치 또는 방법을 커버하도록 의도된다. 개시된 본 개시의 임의의 양태는 청구항의 하나 이상의 요소들에 의해 구체화될 수도 있다는 것이 이해되어야 한다.
- [0018] "예시적" 이라는 용어는 "예, 실례, 또는 예시의 역할을 하는 것" 을 의미하는 것으로 여기에서 사용된다. "예시적" 으로서 여기에 설명된 임의의 양태는 반드시 다른 양태들보다 바람직하거나 또는 유리한 것으로 해석될 필요는 없다.
- [0019] 특정 양태들이 여기에서 설명되었지만, 이들 양태들의 많은 변형 및 치환이 본 개시의 범위내에 속한다. 바람직한 양태들의 일부 혜택 및 이점들이 언급되었지만, 본 개시의 범위는 특정 혜택, 용도 또는 목적에 한정되도록 의도되지 않았다. 오히려, 본 개시의 양태들은 상이한 기술들, 시스템 구성들, 망들 및 프로토콜들에 폭넓게 적용가능하도록 의도되고, 이들 중 일부는 예로써 도면에 그리고 다음의 바람직한 양태들의 설명에 예시되어 있다. 상세한 설명 및 도면들은 본 개시를 제한하는 것이 아니라 예시할뿐이고, 본 개시의 범위는 첨부된 청구항들 및 이의 균등물에 의해 정의된다.
- [0020] 분류를 위한 강제된 희소성
- [0021] 분류 작업에서, 특정 추출기를 통해 출력되는 특정 벡터들은 종종 조밀할 수 있다 (예를 들어, n -제로 엘리먼트가 많이 포함됨). 이러한 조밀한 특정 벡터들을 갖는 것은 메모리 요건들 및 분류기 레이턴시에 악영향을 미칠 수 있다. 또한, n -제로 피쳐 값들이 작은 다수의 특정 벡터 엘리먼트들을 갖는 것은 잡음이 있는 특정 벡터를 나타낼 수 있으며, 이는 차례로 분류 정확성을 감소시킬 수 있다.
- [0022] 본 개시의 양태들은 개선된 특정 추출 및 분류 정확성에 관한 것이다. 본 개시에서, 강제된 희소성 (enforced sparsity; ES) 프로세스는 주어진 특정 벡터의 특정 값들 또는 엘리먼트들의 상위 K 번호 또는 백분율만이 유지되도록 채택된다. 다른 값들은 제로 (0) 로 설정되어, 주어진 특정 벡터보다 적은 비-제로 값들을 갖는 희소한 특정 벡터를 생성할 수 있다. 하지만, 특정 벡터의 차원들 (예를 들어, 특정 벡터에서의 엘리먼트들의 수) 은 유지될 수 있다. 특정 벡터의 희소성을 증가시킴으로써, 예를 들어 재훈련을 위한 특징을 저장하는데 더 적은 메모리가 사용될 수 있다. 더욱이, 보다 높은 희소성은 또한 분류기 성능 (예를 들

어, 분류 속도 및 정확성)을 향상시킬 수 있다.

- [0023] 도 1은 본 개시의 특정 양태들에 따른 범용 프로세서 (CPU) 또는 멀티 코어 범용 프로세서 (CPU) (102)를 포함할 수도 있는 시스템 온 칩 (SOC) (100)을 사용하는 절충한 강제된 회소성 및 특징 추출의 예시적인 구현을 도시한다. 변수 (예를 들어, 신경 신호 및 시냅스 가중치), 계산 디바이스 (예를 들어, 가중치를 갖는 신경망)와 연관된 시스템 파라미터, 지연, 주파수 빈 (frequency bin) 정보 및 작업 정보는 신경 처리 유닛 (NPU) (108)과 연관된 메모리 블록에, CPU (102)와 연관된 메모리 블록에, 그래픽 처리 유닛 (GPU) (104)과 연관된 메모리 블록에, 디지털 신호 프로세서 (DSP) (106)와 연관된 메모리 블록에, 전용 메모리 블록 (118)에 저장되거나, 또는 다수의 블록들에 걸쳐 분산될 수 있다. 범용 프로세서 (102)에서 실행되는 명령들은 CPU (102)와 연관된 프로그램 메모리로부터 로딩될 수도 있거나 또는 전용 메모리 블록 (118)으로부터 로딩될 수도 있다.
- [0024] SOC (100)는 또한 그래픽 처리 유닛 (GPU) (104), 디지털 신호 프로세서 (DSP) (106), 접속 블록 (110) (제 4세대 롱 텀 에볼루션 (4G LTE) 접속성, 비허가 Wi-Fi 접속성, USB 접속성, 블루투스 접속성 등을 포함할 수도 있음), 및 예를 들어, 제스처를 검출하고 인식할 수도 있는 멀티미디어 프로세서 (112)와 같은 특정 기능에 맞추어진 추가 처리 블록을 포함할 수도 있다. SOC (100)는 또한 센서 프로세서 (114), 이미지 신호 프로세서 (ISP) 및/또는 네비게이션 (120) (위성 위치확인 시스템을 포함할 수도 있음)을 포함할 수도 있다. SOC는 ARM 명령 세트에 기초할 수도 있다.
- [0025] SOC (100)는 ARM 명령 세트에 기초할 수도 있다. 본 개시의 양태에서, 범용 프로세서 (102)로 로딩된 명령들은 특징 추출기로부터 특징 벡터를 수신하기 위한 코드를 포함할 수 있다. 범용 프로세서 (102)에 로딩된 명령들은 또한 회소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 백분율을 유지하기 위한 코드를 포함할 수 있다. 또한, 범용 프로세서 (102)에 로딩된 명령들은 또한 회소한 특징 벡터를 분류 기에 포워딩하기 위한 코드를 포함할 수 있다.
- [0026] 도 2은 본 개시의 특정 양태들에 따른 시스템 (200)의 예시적 구현을 나타낸다. 도 2에 나타낸 바와 같이, 시스템 (200)은 여기서 설명된 방법들의 다양한 동작들을 수행할 수도 있는 다수의 로컬 처리 유닛들 (202)을 가질 수도 있다. 각각의 로컬 처리 유닛 (202)은 신경망의 파라미터들을 저장할 수도 있는 로컬 파라미터 메모리 (206) 및 로컬 상태 메모리 (204)를 포함할 수도 있다. 또한, 로컬 처리 유닛 (202)은 로컬 모델 프로그램을 저장하기 위한 로컬 (뉴런) 모델 프로그램 (LMP) 메모리 (208), 로컬 학습 프로그램을 저장하기 위한 로컬 학습 프로그램 (LLP) 메모리 (210), 및 로컬 접속 메모리 (212)를 가질 수도 있다. 또한, 도 2에 나타낸 바와 같이, 각각의 로컬 처리 유닛 (202)은 로컬 처리 유닛의 로컬 메모리들을 위한 구성을 제공하기 위한 구성 프로세서 유닛 (214)과, 그리고 로컬 처리 유닛들 (202) 사이에 라우팅을 제공하는 라우팅 접속 처리 유닛 (216)과 인터페이스 접속될 수도 있다.
- [0027] 심층 학습 아키텍처는 각 계층에서 연속적으로 더 높은 수준의 추상화에서 입력을 표현하기 위해 학습에 의해 객체 인식 작업을 수행함으로써, 입력 데이터의 유용한 특징 표현을 구축할 수도 있다. 이런 식으로, 심층 학습은 전통적인 머신 학습의 주요 병목을 해결한다. 심층 학습의 출현 전에, 객체 인식 문제에 대한 머신 학습 접근법은, 아마도 얇은 분류기 (shallow classifier)와 조합한, 인간 공학적 특징들에 크게 의존했을 수도 있다. 얇은 분류기는, 예를 들어, 특징 벡터 성분들의 가중 합을 임계치와 비교하여 입력이 어느 부류에 속하는지를 예측할 수도 있는 2-부류 선형 분류기일 수도 있다. 인간 공학적 특징들은, 도메인 전문 기술을 가진 기술자에 의해 특정 문제 도메인에 맞추어진 템플릿 또는 커널일 수도 있다. 대조적으로, 심층 학습 아키텍처는, 인간 기술자가 설계할 수도 있는 것과 유사한 특징들을 표현하는 것을, 그러나 훈련을 통해, 학습할 수도 있다. 또한, 심층 망은 인간이 고려하지 않았을 수도 있는 새로운 유형의 특징들을 표현하고 인식하는 것을 학습할 수도 있다.
- [0028] 심층 학습 아키텍처는 특징들의 체계를 학습할 수도 있다. 예를 들어, 시각적 데이터가 제시되면, 제 1 계층은 입력 스트림에서 에지들과 같은 비교적 단순한 특징들을 인식하는 것을 학습할 수도 있다. 또 다른 예에서, 청각 데이터가 제시되면, 제 1 계층은 특정 주파수에서 스펙트럼 파워를 인식하는 것을 학습할 수도 있다. 제 1 계층의 출력을 입력으로서 취하는 제 2 계층은 시각적 데이터에 대한 단순한 형상 또는 청각적 데이터에 대한 소리의 조합과 같은 특징들의 조합을 인식하는 것을 학습할 수도 있다. 가령, 상위 계층들은 시각 데이터에서의 복잡한 형상들 또는 청각 데이터에서의 단어들을 나타내는 것을 학습할 수도 있다. 더욱 상위 계층들은 일반적인 시각적 객체 또는 구어를 인식하는 것을 학습할 수도 있다.
- [0029] 심층 학습 아키텍처는 자연적 체계 구조를 갖는 문제에 적용될 때 특히 잘 수행될 수도 있다. 예를 들어,

자동차의 분류는 바퀴, 앞 유리 및 다른 특징들을 인식하는 것을 먼저 학습하는 것으로부터 이익을 얻을 수도 있다. 이러한 특징들은 승용차, 트럭 및 비행기를 인식하기 위해 상이한 방식으로 상위 계층에서 결합될 수도 있다.

[0030] 신경망은 다양한 접속 패턴들로 설계될 수도 있다. 피드 포워드 망에서, 정보는 하위 계층으로부터 상위 계층으로 보내지며, 주어진 계층에 있는 각 뉴런은 상위 계층에 있는 뉴런들과 통신한다. 계층적 표현은 전술한 바와 같이 피드 포워드 망의 연속적인 계층들에 구축될 수도 있다. 신경망은 또한, 회귀성 (recurrent) 또는 피드백 (feedback) (하향식이라고도 함) 접속들을 가질 수도 있다. 회귀성 접속에서, 주어진 층에 있는 뉴런으로부터의 출력은 동일한 계층에 있는 다른 뉴런에 전달될 수도 있다. 회귀성 아키텍처는 시퀀스에서 신경망에 전달되는 입력 데이터 청크 중 하나를 초과하여 걸쳐있는 패턴들을 인식하는 데 도움이 될 수도 있다. 주어진 계층에 있는 뉴런으로부터 하위 계층에 있는 뉴런으로의 접속을 피드백 (또는 하향식) 접속이라고 한다. 높은 수준의 개념의 인식이 입력의 특정 저수준 특징들을 식별하는 데 도움이 될 수도 있을 때 많은 피드백 접속을 가진 망이 도움이 될 수도 있다.

[0031] 도 3a 를 참조하면, 신경망의 계층들 사이의 접속은 완전히 접속 (302) 되거나 또는 국부적으로 접속 (304) 될 수도 있다. 완전히 접속된 망 (302) 에서, 제 1 계층에 있는 뉴런은 그의 출력을 제 2 계층에 있는 모든 뉴런에 전달하여, 제 2 계층에 있는 각 뉴런은 제 1 계층에 있는 모든 뉴런으로부터 입력을 수신할 것이다. 대안적으로, 국부적으로 접속된 망 (304) 에서, 제 1 계층에 있는 뉴런은 제 2 계층에 있는 제한된 수의 뉴런에 접속될 수도 있다. 컨볼루셔널 망 (306) 은 국부적으로 접속될 수도 있고, 또한, 제 2 계층에 있는 각 뉴런에 대한 입력과 연관된 접속 강도가 공유되도록 구성된다 (예 : 308). 보다 일반적으로, 망의 국부적으로 접속된 계층은, 계층에 있는 각 뉴런이 동일하거나 또는 유사한 접속 패턴을 갖지만, 상이한 값 (예를 들어, 310, 312, 314 및 316) 을 가질 수도 있는 접속 강도를 갖도록 구성될 수도 있다. 주어진 영역에 있는 상위 계층 뉴런이, 망에 대한 전체 입력의 제한된 부분의 특성에 맞게 훈련을 통해 조정된 입력을 수신할 수도 있기 때문에, 국부적으로 접속된 접속 패턴은 상위 계층에서 공간적으로 구분되는 수용 영역들을 발생시킬 수도 있다.

[0032] 국부적으로 접속된 신경망은 입력의 공간적 위치가 의미있는 문제에 잘 맞을 수도 있다. 가령, 차재 카메라로부터 시각적 특징들을 인식하도록 설계된 망 (300) 은 이미지의 하부 대 상부와의 연관성에 따라 상이한 특성을 갖는 상위 계층 뉴런을 개발할 수도 있다. 이미지의 하부와 연관된 뉴런은 예를 들어, 차선 표시를 인식하는 것을 학습할 수도 있는 반면, 이미지의 상부와 연관된 뉴런은 신호등, 교통 표지 등을 인식하는 것을 학습할 수도 있다.

[0033] DCN (deep convolutional network) 은 지도 학습으로 훈련될 수도 있다. 훈련 동안, DCN 에는 속도 제한 표지 (326) 의 크롭된 이미지와 같은 이미지가 제시될 수도 있고, 다음으로 "포워드 패스" 가 계산되어 출력 (322) 을 생성할 수도 있다. 출력 (322) 은 "표지", "60" 및 "100"과 같은 특징들에 대응하는 값의 벡터일 수도 있다. 망 설계자는, DCN 이, 훈련되었던 망 (300) 에 대한 출력 (322) 에 보여진 바와 같이, 예를 들어 "표지" 및 "60" 에 대응하는 것들인, 출력 특징 벡터에 있는 일부 뉴런들에 대해 높은 점수를 출력하기를 원할 수도 있다. 훈련 전에 DCN 에 의해 생성된 출력이 부정확할 가능성이 있으므로, 실제 출력과 타겟 출력 사이에 에러가 산출될 수도 있다. 다음으로, DCN 의 가중치는 DCN의 출력 점수가 타겟과 보다 긴밀히 정렬 되도록 조정될 수도 있다.

[0034] 가중치를 조정하기 위해, 학습 알고리즘은 가중치들에 대한 그래디언트 벡터를 계산할 수도 있다. 그래디언트는 가중치가 약간 조정되면 에러가 증가하거나 또는 감소하는 양을 나타낼 수도 있다. 최상위 계층에서, 그래디언트는, 끝에서 두 번째 계층에 있는 활성화된 뉴런과 출력 계층에 있는 뉴런을 접속하는 가중치의 값에 직접적으로 대응할 수도 있다. 하위 계층들에서, 그래디언트는 가중치의 값과 상위 계층들의 계산된 에러 기울기에 의존할 수도 있다. 그런 다음, 에러를 줄이기 위해 가중치를 조정할 수도 있다. 가중치를 조정하는 이러한 방식은 신경망을 통한 "백워드 패스" 를 수반하므로 "역 전파" 라고 할 수도 있다.

[0035] 실제로, 가중치들의 에러 그래디언트는, 산출된 그래디언트가 참 에러 그래디언트에 근사하도록 소수의 예들에 대해 산출될 수도 있다. 이 근사법은 확률론적 그래디언트 하강 (stochastic gradient descent) 이라고 할 수도 있다. 확률론적 그래디언트 하강은, 전체 시스템의 달성 가능한 에러율이 감소하는 것을 중단할때까지 또는 에러율이 타겟 수준에 도달할 때까지 반복될 수도 있다.

[0036] 학습 후에, DCN 에는 새로운 이미지들 (326) 이 제시될 수도 있고, 망을 통한 포워드 패스는 DCN의 추론 또는 예측으로 간주될 수도 있는 출력 (322) 을 낳을 수도 있다.

- [0037] 심층 신뢰 망 (DBN) 은 숨겨진 노드의 다수의 계층들을 포함하는 확률 모델이다. DBN 은 훈련 데이터 세트의 체계적 표현을 추출하는데 사용될 수도 있다. DBN은 제한 볼츠만 머신 (RBM) 의 계층들을 적층하여 획득될 수도 있다. RBM은 입력 세트에 대한 확률 분포를 학습할 수있는 일종의 인공 신경망이다. RBM은 각 입력이 카테고리화되어야 하는 부류에 대한 정보가 없는 경우 확률 분포를 학습할 수 있기 때문에, RBM은 비지도 학습 (unsupervised learning) 에서 종종 사용된다. 하이브리드 비지도 및 지도 패러다임을 사용하여, DBN의 최하위 RBM 은 비지도 방식으로 훈련될 수도 있으며 특징 추출기의 역할을 할 수도 있으며, 최상위 RBM은 (이전 계층 및 타겟 부류로부터의 입력들의 결합 분포 (joint distribution) 상에서) 지도 방식으로 훈련될 수도 있으며 분류기의 역할을 할 수도 있다.
- [0038] 심층 컨볼루션 망 (DCN) 은, 추가적 풀링 (pooling) 및 정규화 계층들로 구성된, 컨볼루션 망의 망이다. DCN은 많은 작업에 대한 최첨단 기술 성능을 달성했다. DCN은, 입력 및 출력 타겟들 양자 모두가 많은 전형 (exemplar) 들에 대해 알려져 있고 그래디언트 하강 법을 사용하여 망의 가중치를 수정하는데 사용되는 지도 학습을 사용하여 훈련될 수 있다.
- [0039] DCN은 피드 포워드 망일 수도 있다. 또한, 전술한 바와 같이, DCN의 제 1 계층에 있는 뉴런으로부터 다층 상위 계층에 있는 뉴런들의 그룹으로의 접속은 제 1 계층에 있는 뉴런들에 걸쳐 공유된다. DCN의 피드 포워드 및 공유 접속은 빠른 처리를 위해 활용될 수도 있다. DCN의 계산 부담은, 예를 들어, 회귀성 또는 피드백 접속을 포함하는 유사한 크기의 신경망의 계산 부담보다 훨씬 적을 수도 있다.
- [0040] 컨볼루션 망의 각 계층의 처리는 공간적으로 불변의 템플릿 또는 기초 투영으로 고려될 수도 있다. 입력이 컬러 이미지의 적색, 녹색 및 청색 채널과 같은 다수의 채널들로 먼저 분해되면, 그 입력에 대해 훈련된 컨볼루션 망은 이미지의 축을 따라 2개의 공간 차원 및 색 정보를 캡처하는 제 3 차원을 갖는, 3차원으로 고려될 수도 있다. 컨볼루션 접속들의 출력들은, 후속 계층 (318 및 320) 에서 특징 맵을 형성하는 것으로 고려될 수도 있으며, 특징 맵 (예를 들어, 320) 의 각 요소는 이전 계층 (예를 들어, 318) 에 있는 뉴런들의 범위로부터 그리고 다수의 채널들 각각으로부터 입력을 수신한다. 특징 맵에 있는 값들은 비선형성, 이를테면 교정 (rectification), $\max(0, x)$ 으로 더 처리될 수도 있다. 인접한 뉴런들로부터의 값들은 추가로 풀링될 수도 있는데, 이것은 다운 샘플링에 대응하고, 추가적인 국부적 불변성 및 차원성 감소를 제공할 수도 있다. 백색화 (whitening) 에 대응하는 정규화는 또한, 특징 맵에서의 뉴런들 사이의 측방억제 (lateral inhibition) 를 통해 적용될 수도 있다.
- [0041] 심층 학습 아키텍처들의 성능은 더 많은 라벨링된 데이터 포인트들이 이용가능해짐에 따라, 또는 계산력이 증가함에 따라 증가할 수도 있다. 현대의 심층 신경 망들은 단지 15 년 전에 통상적인 연구자에게 이용가능하였던 것보다 수천 배 더 큰 계산 자원들로 일상적으로 훈련된다. 새로운 아키텍처들 및 훈련 패러다임들은 심층 학습의 성능을 더 상승시킬 수도 있다. 교정된 선형 유닛들은 소실 그래디언트 (vanishing gradient) 들로서 알려진 훈련 문제를 감소시킬 수도 있다. 새로운 훈련 기법들은 오버-피팅 (overfitting) 을 감소시킬 수도 있고, 이에 따라, 더 큰 모델들이 더욱 양호한 일반화를 달성하는 것을 가능하게 할 수도 있다. 캡슐화 (encapsulation) 기법들은 주어진 수용 영역에서 데이터를 추상화 (abstract) 할 수도 있고, 전체적인 성능을 추가로 상승시킬 수도 있다.
- [0042] 도 3b 는 예시적인 심층 컨볼루션 망 (350) 을 예시하는 블록도이다. 심층 컨볼루션 망 (350) 은 접속성 및 가중치 공유에 기초한 다수의 상이한 유형들의 계층들을 포함할 수도 있다. 도 3b 에서 도시된 바와 같이, 예시적인 심층 컨볼루션 망 (350) 은 다수의 컨볼루션 블록들 (예컨대, C1 및 C2) 을 포함한다. 컨볼루션 블록들의 각각은 컨볼루션 계층, 정규화 계층 (LNorm), 및 풀링 계층으로 구성될 수도 있다. 컨볼루션 계층들은 특징 맵을 생성하기 위하여 입력 데이터에 적용될 수도 있는 하나 이상의 컨볼루션 필터들을 포함할 수도 있다. 2개의 컨볼루션 블록들만이 도시되어 있지만, 본 개시는 그렇게 제한되지 않고, 그 대신에, 임의의 수의 컨볼루션 블록들이 설계 선호도에 따라 심층 컨볼루션 망 (350) 에 포함될 수도 있다. 정규화 계층은 컨볼루션 필터들의 출력을 정규화하기 위하여 이용될 수도 있다. 예를 들어, 정규화 계층은 백색화 또는 측방억제를 제공할 수도 있다. 풀링 계층은 국부적 불변성 및 차원성 감소를 위하여 공간에 대한 다운 샘플링 어그리게이션 (down sampling aggregation) 을 제공할 수도 있다.
- [0043] 예를 들어, 심층 컨볼루션 망의 병렬 필터 뱅크들은 높은 성능 및 낮은 전력 소비를 달성하기 위하여, 선택적으로 ARM 명령 세트에 기초하여, SOC (100) 의 CPU (102) 또는 GPU (104) 상에 로딩될 수도 있다. 대안적인 실시형태들에서, 병렬 필터 뱅크들은 SOC (100) 의 DSP (106) 또는 ISP (116) 상에 로딩될 수도 있다. 게다가, DCN 은 센서들 (114) 및 내비게이션 (120) 에 전용된 처리 블록들과 같은, SOC 상에 존재할 수도 있는

다른 처리 블록들에 액세스할 수도 있다.

- [0044] 심층 컨볼루션 망 (350) 은 또한, 하나 이상의 완전히 접속된 계층들 (예컨대, FC1 및 FC2) 을 포함할 수도 있다. 심층 컨볼루션 망 (350) 은 로지스틱 회귀 (logistic regression; LR) 계층을 더 포함할 수도 있다. 심층 컨볼루션 망 (350) 의 각각의 계층 사이에는, 업데이트될 가중치들 (도시되지 않음) 이 있다. 각각의 계층의 출력은 제 1 컨볼루션 블록 (C1) 에서 공급된 입력 데이터 (예컨대, 이미지들, 오디오, 비디오, 센서 데이터, 및/또는 다른 입력 데이터) 로부터의 체계적 특징 표현들을 학습하기 위하여 심층 컨볼루션 망 (350) 에서의 뒤이은 계층의 입력의 역할을 할 수도 있다.
- [0045] 도 4는 인공 지능 (AI) 기능을 모듈화할 수도 있는 예시적인 소프트웨어 아키텍처 (400) 를 나타내는 블록도이다. 아키텍처를 이용하여, SOC (420) 의 다양한 처리 블록들 (예를 들어, CPU (422), DSP (424), GPU (426), 및/또는 NPU (428)) 로 하여금, 애플리케이션 (402) 의 런타임 동작 동안의 계산들을 지원하는 것을 수행하게 할 수도 있는 애플리케이션들 (402) 이 설계될 수도 있다.
- [0046] AI 애플리케이션 (402) 은 예를 들어, 디바이스가 현재 동작하는 위치를 표시하는 장면의 검출 및 인식을 제공할 수도 있는 사용자 공간 (404) 에서 정의된 함수들을 호출하도록 구성될 수도 있다. AI 애플리케이션 (402) 은 예를 들어, 인식된 장면이 사무실, 강당, 식당, 또는 호수와 같은 실외 세팅인지 여부에 따라 상이하게, 마이크로폰 및 카메라를 구성할 수도 있다. AI 애플리케이션 (402) 은 현재의 장면의 추정을 제공하기 위하여 SceneDetect 애플리케이션 프로그래밍 인터페이스 (application programming interface; API) (406) 에서 정의된 라이브러리와 연관된 컴파일링된 프로그램 코드에 대한 요청을 행할 수도 있다. 이 요청은 예를 들어, 비디오 및 위치결정 데이터에 기초하여 장면 추정치들을 제공하도록 구성된 심층 신경 망의 출력에 궁극적으로 의존할 수도 있다.
- [0047] 런타임 프레임워크 (Runtime Framework) 의 컴파일링된 코드일 수도 있는 런타임 엔진 (408) 은 AI 애플리케이션 (402) 에 의해 추가로 액세스가능할 수도 있다. AI 애플리케이션 (402) 은 런타임 엔진으로 하여금, 예를 들어, 특정한 시간 간격으로, 또는 애플리케이션의 사용자 인터페이스에 의해 검출된 이벤트에 의해 트리거링된 장면 추정을 요청하게 할 수도 있다. 장면을 추정하게 될 때, 런타임 엔진은 이어서 신호를, SOC (420) 상에서 실행되는 리눅스 커널 (Linux Kernel) (412) 과 같은 운영 시스템 (410) 으로 전송할 수도 있다. 운영 시스템 (410) 은 이어서, 계산이 CPU (422), DSP (424), GPU (426), NPU (428), 또는 그 일부 조합 상에서 수행되게 할 수도 있다. CPU (422) 는 운영 시스템에 의해 직접적으로 액세스될 수도 있고, 다른 처리 블록들은 DSP (424), GPU (426), 또는 NPU (428) 를 위한 드라이버 (414 내지 418) 와 같은 드라이버를 통해 액세스될 수도 있다. 예시적인 예에서, 심층 신경 망은 CPU (422) 및 GPU (426) 와 같은, 처리 블록들의 조합 상에서 실행되도록 구성될 수도 있거나, 존재할 경우, NPU (428) 상에서 실행될 수도 있다.
- [0048] 도 5 는 스마트폰 (502) 상에서의 AI 애플리케이션의 런타임 동작 (500) 을 나타내는 블록도이다. AI 애플리케이션은, 이미지 (506) 의 포맷을 변환하고 그 다음으로, 이미지 (508) 를 크롭하거나 및/또는 리사이즈 (resize) 하도록 (예를 들어, JAVA 프로그래밍 언어를 이용하여) 구성될 수도 있는 전처리 (preprocess) 모듈 (504) 을 포함할 수도 있다. 그 다음으로, 전처리된 이미지는, 시각적 입력에 기초하여 장면들을 검출하고 분류하도록 (예를 들어, C 프로그래밍 언어를 이용하여) 구성될 수도 있는 SceneDetect 백엔드 엔진 (Backend Engine) (512) 을 포함하는 분류 애플리케이션 (510) 으로 전달될 수도 있다. SceneDetect 백엔드 엔진 (512) 은 스케일링 (516) 및 크롭핑 (518) 에 의해 이미지를 더 전처리 (514) 하도록 구성될 수도 있다. 예를 들어, 이미지는 결과적인 이미지가 224 픽셀들 x 224 픽셀들이 되도록 스케일링 및 크롭될 수도 있다. 이 차원들은 신경망의 입력 차원들로 맵핑될 수도 있다. 신경망은 SOC (100) 의 다양한 처리 블록들로 하여금, 심층 신경 망으로 이미지 픽셀들을 더 처리하게 하기 위하여 심층 신경 망 블록 (520) 에 의해 구성될 수도 있다. 그 다음으로, 심층 신경 망의 결과들은 임계화 (522) 될 수도 있고 분류 애플리케이션 (510) 에 있는 지수 평활화 블록 (524) 을 통과하게 될 수도 있다. 그 다음으로, 평활화된 결과들은 스마트폰 (502) 의 세팅 및/또는 디스플레이의 변경을 야기할 수도 있다.
- [0049] 하나의 구성에서, 머신 학습 모델은 입력으로부터 특징 벡터를 생성하기 위해 구성된다. 모델은 또한 최소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정하도록 구성된다. 머신 학습 모델은 또한 최소한 특징 벡터를 분류기로 포워딩하도록 구성된다. 머신 학습 모델은 생성 수단, 설정 수단, 및/또는 포워딩 수단을 포함한다. 하나의 양태에서, 생성 수단, 설정 수단, 및/또는 포워딩 수단은 범용 프로세서 (102), 범용 프로세서 (102) 와 연관된 프로그램 메모리, 메모리 블록 (118), 로컬 처리 유닛들 (202), 및 또는 언급된 기능들을 수행하도록 구성된 라우팅 연결 처리 유닛들 (216) 일 수도 있다. 또 다른 구성에

서, 상기 언급된 수단은 상기 언급된 수단에 의해 언급된 기능들을 수행하도록 구성된 임의의 모듈 또는 임의의 장치일 수도 있다.

- [0050] 본 개시물의 특징의 양태들에 따르면, 각각의 로컬 처리 유닛 (202) 은 신경망의 소망하는 하나 이상의 기능적 특징들에 기초하여 신경망의 파라미터들을 결정하고, 결정된 파라미터들이 또한 적응, 조정, 및 업데이트됨에 따라 소망하는 기능적 특징들을 향한 하나 이상의 기능적 특징들을 개발하도록 구성될 수도 있다.
- [0051] 도 6 은 본 개시의 양태들에 따른 특징 추출기 (602) 를 포함하는 예시적인 머신 학습 모델 (600) 을 나타내는 블록도이다. 도 6 을 참조하면, 특징 추출기 (602) 의 전체 연결된 계층 (FC1) 및 강제된 희소성 유니티가 도시되어 있다. 몇몇 양태들에서, 전체 연결된 계층은 예를 들어, 심층 컨벌루션 신경 네트워크 (DCN) 또는 다른 네트워크의 계층 (예를 들어, 마지막 계층) 을 포함할 수 있다.
- [0052] 전체 연결된 계층 (FC1) 은 강제된 희소성 (ES) 유닛에 특징 벡터를 공급할 수 있다. 이러한 예시적인 구성에서, 입력 (예를 들어, 이미지 픽셀, 스피치 등) 은 DCN의 다중 계층들을 통과하여 특정 특징들을 추출하고 전체 접속된 계층 (FC1) 을 통해 특징 벡터를 출력할 수 있다.
- [0053] 일반적으로, DCN들은 데이터를 정류하기 위해 정류기 리니어 유닛들 (ReLU) 또는 파라미터정 정류기 리니어 유닛들 (PReLU) 을 채용할 수 있다. ReLU들은 네거티브 특징 벡터 값들을 제로로 설정하고, 포지티브 값들을 유지함으로써 데이터를 정류한다. 반면, PReLU들은 포지티브 특징 벡터 값들을 유지하고 네거티브 값들을 선형적으로 스케일한다. 하지만, 양자 모두가 강제된 희소성의 제안된 방법들과 비교하여 교육 및 추론에 있어서 증가된 메모리 소비와 강화된 레이턴시를 갖는 특징 벡터들을 생성한다.
- [0054] 본 개시의 양태들에 따라, 강제된 희소성이 ES 유닛을 통해 적용될 수 있다. 즉, 특징 벡터에 포함된 데이터는 데이터를 희소화하거나 또는 데이터를 희소하게 만들기 위해 강제된 희소성 (ES) 유닛으로 공급될 수 있다. ES 유닛을 사용하여, 데이터 엘리먼트들의 상위 K%가 유지될 수 있고, 여기서 K는 정수이다. 즉, 최상위 값을 갖는 엘리먼트들의 K% 또는 엘리먼트들의 K 수가 유지될 수 있다. 특징 벡터의 나머지 엘리먼트들은 제로로 설정될 수 있다. 이와 같이, 공급된 특징 벡터와 엘리먼트들의 차원 또는 수가 동일한 희소한 특징 벡터는, 단지 엘리먼트들의 K% 또는 엘리먼트들의 K 숫자에 대해 n -제로 값들을 포함하는 것으로 생성될 수도 있다.
- [0055] 일부 양태들에서, K 값은 오프라인으로 계산되거나 또는 결정될 수 있다. 예를 들어, K 값은 0%와 100% 사이의 다양한 K 값들에 걸친 파라미터 스위프에 기초하여 결정될 수 있다.
- [0056] 반대로, K 값은 온라인으로도 결정될 수 있다. 예를 들어, K 값은 사용자로부터 일 세트의 훈련 및 유효성 샘플들을 유지하고 K에 걸친 스위프를 수행함으로써 결정될 수 있다.
- [0057] 도 6에서는 ES 유닛이 DCN의 외부로서 도시되지만, ES 유닛은 DCN 또는 다른 네트워크 내에 통합될 수 있다. 일 예에서, 희소성은, 최소 절대 오차 (L1) 비용 함수를 DCN 또는 다른 특징 추출기의 훈련 과정의 일부로서 적용함으로써 DCN에 통합될 수 있다. 몇몇 양태들에서, 비용 함수는 특징 벡터에 포함된 n -제로 엘리먼트들의 수를 페널티화하도록 구성될 수 있다. 또한, 오차의 L1 norm을 최소화하여 n -제로 특징 값들의 수를 보다 작은 수로 강제할 수 있다. 이와 같이, DCN은 희소성 인자, 및 경우에 따라, 특징 벡터의 가장 양호하거나 바람직한 희소성 인자를 학습할 수 있다.
- [0058] DCN에서, 훈련은 예측된 라벨과 실제 라벨 사이의 오차의 함수로서 가중치를 업데이트함으로써 진행된다. 이러한 오차는 페널티 항이며 한 가지 목표는 오차를 제로로 감소시킨다는 것이다. 본 개시의 양태들에 따르면, 제 2 페널티 항이 추가될 수 있다. 제 2 페널티 항은 희소성이 요구되는 계층의 활성화의 norm을 포함할 수 있다. 목표가 특징 벡터에서 n -제로 엘리먼트들의 수를 최소화하는 것이기 때문에, 일부 양태들에서 이러한 제 2 페널티 항은 그 계층에서 n -제로 항의 수의 카운트를 포함할 수 있다. n -제로 항의 카운트는 L0 norm 일 수 있다.
- [0059] 하지만, 이 정량은 미분가능하지 않을 것이고 (예를 들어, 가중치 업데이트를 유도하기 위한 비용 함수의 기울기를 사용하는 후방 전파를 사용하여) 망을 훈련하는데 어려움으로 이어질 수 있다. 이에 따라서, 몇몇 양태들에서, L1 norm (항의 카운트 대신에 항의 절대값들의 합) 이 제 2 페널티 항으로 사용될 수 있다. 절대값의 작은 합을 강제함으로써, 항의 수는 제로 (또는 제로로 임계화될 수 있는 매우 작은 수) 로 향하도록 간접적으로 강제될 수 있다.
- [0060] 이러한 패러다임에서, n -제로 항의 "최적의" 수 (예를 들어, K) 가 비용 함수의 최소화의 일부로서 결정될 수

있다. 이 경우, 년-제로 항의 수가 오차를 최소화하고 희소성을 최대화하기 때문에, 그것은 최적인 것으로 간주될 수 있다.

[0061] 일 예에서, 엘리먼트들을 포함하는 특징 벡터 $F[-1 \ 0 \ 2 \ -3 \ 5 \ 7 \ 9 \ 4 \ -1 \ 2]$ 가 ES 유닛으로 공급될 수 있다. ES 유닛은, 예를 들어, 특징 벡터 값들 또는 엘리먼트들의 상위 20%를 유지하도록 구성될 수 있다. 이에 따라서, ES 유닛은 특징 벡터의 열개 (10)의 엘리먼트들의 특징 값들 또는 상위 2개 (2)의 엘리먼트들을 결정할 수 있다. 따라서, ES 유닛은 년-제로 특징 값들 또는 엘리먼트들의 희소성으로 변경된 특징 벡터 $F'[0 \ 0 \ 0 \ 0 \ 7 \ 9 \ 0 \ 0 \ 0]$ 를 출력할 수 있다. 이에 따라서, 희소한 특징은 ES 유닛을 통해 출력되고 분류기에 공급될 수 있다.

[0062] 일부 양태들에서, ES 유닛은 최대 K%의 특징 벡터를 유지할 수 있다. 대안으로, ES 유닛은 특징 벡터 엘리먼트들의 절대값들, 양의 값들, 또는 음의 값들의 상위 K%를 유지할 수 있다. 부가하여, 상위 K%를 선택하는 것은 특징 벡터의 년-제로 엘리먼트들 또는 서바이빙 특징 값들을 선택하기 위한 실증이다.

[0063] 희소한 특징 벡터는, 재훈련 특징들을 저장하는 메모리 소비를 감소시킬 수 있기 때문에 특히 유용할 수 있다. 증가된 희소성은 또한, 보다 적은 계산들이 수행되는 보다 빠른 분류기 훈련 및 추론을 가능하도록 도울 수 있다. 또한, 희소한 특징 벡터는 분류기 정확도를 개선할 수 있다. 더욱이, 희소한 특징 벡터들은 또한 2개 이상의 특징 벡터들 사이의 특징 공간에서의 거리 산출을 수반하는 작업을 수행하고, 예를 들어 "유사한" 특징들의 클러스터를 형성하고, 또는 "최근접 이웃 분류기들" 등의 보다 간단한 분류기를 구축하는 것을 돕는다.

[0064] 몇몇 양태에서, 수정된 특징 벡터의 엘리먼트 값은 이진화 또는 양자화될 수 있다. 예를 들어, 위의 예에서 F의 이진화된 버전은 $F'[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0]$ 일 수 있다. 희소한 특징 벡터는 차례로 분류기로 제출될 수 있다.

[0065] 제 2 예에서, 엘리먼트 값들은 양자화될 수 있다. 이 예에서, 모든 "서바이빙 정량" 또는 K 최고 값들은 1로 인코딩될 수 있고 다른 모든 것은 0으로 인코딩될 수 있다. 예를 들어, 원하는 희소성이 80%이고 벡터 크기가 10인 경우라면, 최소 정량들중 8은 0으로 설정될 수 있고 2개의 서바이빙 정량들 (예를 들어, 최고의 엘리먼트 값들)은 1로 설정될 수 있다.

[0066] 도 7은 특징 추출 및 입력 분류를 위한 방법 (700)을 나타낸다. 블록 (702)에서, 공정은 입력으로부터 특징 벡터를 생성한다. 입력은 이미지, 보이스, 스피치, 또는 다른 입력 데이터일 수 있다. 블록 704에서, 프로세스는 희소한 특징 벡터를 생성하기 위해 특징 벡터의 엘리먼트들의 수를 제로로 설정한다. 희소한 특징 벡터는 생성된 특징 벡터와 동일한 차원들을 가지며, 생성된 특징 벡터보다 적은 년-제로 엘리먼트들을 포함한다.

[0067] 일부 양태들에서, 엘리먼트들의 수는, 예를 들어, 분류기 레이턴시, 분류기 정확성, 분류기 속도 및/또는 메모리 이용과 같은 성능 메트릭에 기초하여 결정될 수 있다. 성능 메트릭은 온-디바이스로 (예를 들어, 분류 작업을 수행하는 디바이스 상에서) 또는 오프-디바이스로 결정될 수 있다.

[0068] 더욱이, 블록 706에서, 프로세스는 희소한 특징 벡터를 분류기로 포워드한다. 몇몇 양태들에서, 이 프로세스는 메모리 소비를 더욱 감소시키기 위해 희소한 피쳐 벡터의 엘리먼트를 더욱 양자화할 수 있다.

[0069] 일부 양태에서, 이 프로세스는 유지되는 특징 벡터의 엘리먼트들의 수를 결정하기 위해 특징 추출기를 더욱 훈련시킬 수 있다. 훈련은 특징 벡터의 낮은 희소성을 페널라이징하는 비용 함수의 적용을 포함할 수 있다. 몇몇 양태들에서, 비용 함수는 적어도 절대 오차들 (L1-norm) 또는 L0 norm 정규화를 포함할 수 있다.

[0070] 몇몇 양태들에서, 방법 (700)은 SoC (100) (도 1) 또는 시스템 (200) (도 2)에 의해 수행될 수 있다. 즉, 방법 (700)의 엘리먼트들의 각각은, 예를 들어, 제한되지는 않지만, SoC (100) 또는 시스템 (200) 또는 하나 이상의 프로세서들 (예를 들어, CPU (102) 또는 로컬 처리 유닛 (202)) 및/또는 내부에 포함된 다른 컴포넌트들에 의해 수행될 수 있다.

[0071] 위에 설명된 방법들의 다양한 동작들은 대응하는 기능들을 수행할 수 있는 임의의 적합한 수단에 의해 수행될 수도 있다. 그 수단은, 회로, 주문형 집적 회로 (ASIC) 또는 프로세서를 포함하지만 이에 한정되지 않는 다양한 하드웨어 및/또는 소프트웨어 컴포넌트(들) 및/또는 모듈(들)을 포함할 수도 있다. 일반적으로, 도면에 예시된 동작들이 있는 경우에, 그러한 동작들은 유사한 넘버링을 갖는 대응하는 상대의 기능식 컴포넌트들을 가질 수도 있다.

- [0072] 본원에서 사용된, 용어 "결정"은 광범위하게 다양한 활동들을 포함한다. 예를 들어, "결정"은 산출, 계산, 처리, 도출, 조사, 록업 (예를 들면, 테이블, 데이터베이스 또는 다른 데이터 구조에서의 록업), 확인 등을 포함할 수도 있다. 또한, "결정"은 수신하는 것 (예를 들면, 정보를 수신하는 것), 액세스하는 것 (예컨대, 메모리에서 데이터에 액세스하는 것) 등을 포함할 수도 있다. 게다가, "결정"은 해결하는 것, 선택하는 것, 선정하는 것, 확립하는 것 등을 포함할 수도 있다.
- [0073] 본원에 사용된, 항목들의 리스트 "중 적어도 하나"를 나타내는 어구는, 단일 멤버들을 포함한 그러한 아이템들의 임의의 조합을 나타낸다. 일 예로서, "a, b, 또는 c 중 적어도 하나"는 a, b, c, a-b, a-c, b-c, 및 a-b-c를 포함하도록 의도된다.
- [0074] 본 개시와 관련하여 설명된 다양한 예시적인 논리 블록, 모듈, 및 회로는 범용 프로세서, 디지털 신호 프로세서 (DSP), 주문형 집적 회로 (ASIC), 필드 프로그래밍가능 게이트 어레이 신호 (FPGA) 또는 다른 프로그램가능 로직 디바이스 (PLD), 이산 게이트 또는 트랜지스터 로직, 이산 하드웨어 컴포넌트 또는 여기에 설명된 기능을 수행하도록 설계된 이들의 임의의 조합으로 구현 또는 수행될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있지만, 다르게는, 프로세서는 임의의 상용 프로세서, 제어기, 마이크로제어기 또는 상태 머신일 수도 있다. 또한, 프로세서는 계산 디바이스들의 조합, 예를 들어, DSP와 마이크로프로세서의 조합, 복수의 마이크로프로세서, DSP 코어와 결합한 하나 이상의 마이크로프로세서, 또는 임의의 다른 이러한 구성으로서 구현될 수도 있다.
- [0075] 본 개시와 관련하여 설명된 방법 또는 알고리즘의 단계는, 직접적으로 하드웨어로, 프로세서에 의해 실행되는 소프트웨어 모듈로, 또는 양자의 조합으로 구현될 수도 있다. 소프트웨어 모듈은 당업계에 공지된 임의의 형태의 저장 매체에 상주할 수도 있다. 이용될 수도 저장 매체들의 일부 예들은, 랜덤 액세스 메모리 (RAM), 판독 전용 메모리 (ROM), 플래시 메모리, EPROM (erasable programmable read-only memory), EEPROM (electrically erasable programmable read-only memory), 레지스터들, 하드 디스크, 이동식 디스크, CD-ROM 등을 포함한다. 소프트웨어 모듈은 단일 명령 또는 많은 명령들을 포함할 수도 있고, 여러 상이한 코드 세그먼트들 상에, 상이한 프로그램들 사이에서, 그리고 다수의 저장 매체들에 걸쳐 분포될 수도 있다. 저장 매체는 프로세서가 저장 매체로부터 정보를 판독할 수 있고 저장 매체에 정보를 기입할 수 있도록 프로세서에 연결될 수도 있다. 다르게는, 저장 매체는 프로세서에 통합될 수도 있다.
- [0076] 본원에 개시된 방법들은 설명된 방법을 달성하기 위한 하나 이상의 단계들 또는 행동들을 포함한다. 방법 단계들 및/또는 행동들은 청구항들의 범위로부터 이탈함이 없이 서로 상호교환될 수도 있다. 즉, 단계들 또는 행동들의 특정 순서가 명시되지 않으면, 특정 단계들 및/또는 행동들의 순서 및/또는 사용은 청구항들의 범위로부터 이탈함이 없이 수정될 수도 있다.
- [0077] 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어 또는 이들의 임의의 조합으로 구현될 수도 있다. 하드웨어에서 구현되면, 예시적인 하드웨어 구성은 디바이스에 처리 시스템을 포함할 수도 있다. 처리 시스템은 버스 아키텍처로 구현될 수도 있다. 버스는 처리 시스템의 특정 응용 및 전체 설계 제약들에 따라 임의의 수의 상호접속 버스 및 브리지들을 포함할 수도 있다. 버스는 프로세서, 머신 판독가능 매체들, 및 버스 인터페이스를 포함한 다양한 회로들을 함께 링크할 수도 있다. 버스 인터페이스는 다른 것들 중에서 네트워크 어댑터를 버스를 통해 처리 시스템에 접속시키는데 이용될 수도 있다. 네트워크 어댑터는 신호 처리 기능들을 구현하는데 이용될 수도 있다. 특정 양태들에서, 사용자 인터페이스 (예를 들어, 키패드, 디스플레이, 마우스, 조이스틱 등)가 또한 버스에 접속될 수도 있다. 버스는 또한, 타이밍 소스, 주변기기, 전압 레귤레이터, 전력 관리 회로 등과 같은 다양한 다른 회로들을 링크할 수도 있는데, 이들은 업계에 잘 알려져 있으므로, 더 이상 설명되지 않을 것이다.
- [0078] 프로세서는, 버스를 관리하는 것 및 머신 판독가능 매체에 저장된 소프트웨어의 실행을 포함한, 일반적인 처리를 담당할 수도 있다. 프로세서는 하나 이상의 범용 및/또는 특수-목적 프로세서들로 구현될 수도 있다. 예들은 마이크로프로세서들, 마이크로제어기들, DSP 프로세서들, 및 소프트웨어를 실행할 수 있는 다른 회로들을 포함한다. 소프트웨어는 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 하드웨어 기술 언어, 또는 달리 지칭되는지 간에 명령들, 데이터, 또는 이의 임의의 조합을 의미하는 것으로 넓게 해석되어야 할 것이다. 머신 판독가능 매체는, 예로서, RAM (random access memory), 플래시 메모리, ROM (read only memory), PROM (programmable read-only memory), EPROM (erasable programmable read-only memory), EEPROM (electrically erasable programmable Read-only memory), 레지스터들, 자기 디스크들, 광학 디스크들, 하드 드라이브들, 또는 임의의 적합한 저장 매체, 또는 이의 임의의 조합을 포함할 수도 있다. 머신 판독가능 매체는 컴퓨터 프

로그래밍 제품에 수록될 수도 있다. 컴퓨터 프로그램 제품은 패키징 재료들을 포함할 수도 있다.

[0079] 하드웨어 구현에서, 머신 판독가능 매체들은 프로세서와 별개인 처리 시스템의 일부분일 수도 있다. 그러나, 당업자라면 쉽게 이해하는 바와 같이, 머신 판독가능 매체들 또는 이의 임의의 부분은 처리 시스템의 외부에 있을 수도 있다. 예로서, 머신 판독가능 매체들은 송신 라인, 데이터에 의해변조된 반송파, 및/또는 디바이스와 별개인 컴퓨터 제품을 포함할 수도 있으며, 이들 모두는 버스 인터페이스를 통해서 프로세서에 의해 액세스될 수도 있다. 대안적으로 또는 추가적으로, 머신 판독가능 매체들 또는 이의 임의의 부분은 캐시 및/또는 일반 레지스터 파일들의 경우처럼 프로세서에 통합될 수도 있다. 설명된 다양한 컴포넌트들은 로컬 컴포넌트와 같은 특정 위치를 갖는 것으로 설명될 수도 있지만, 분산형 컴퓨팅 시스템의 일부로서 구성되는 특정 컴포넌트와 같이 다양한 방식으로 또한 구성될 수도 있다.

[0080] 처리 시스템은 프로세서 기능성을 제공하는 하나 이상의 마이크로프로세서들 및 적어도 머신 판독가능 매체의 일부를 제공하는 외부 메모리를 갖는 범용 처리 시스템으로서 구성될 수도 있으며, 이들 모두는 외부 버스 아키텍처를 통해서 다른 지원 회로부와 함께 링크된다. 대안적으로, 처리 시스템은 여기에 설명된 뉴런 모델 및 신경 시스템의 모델을 구현하기 위한 하나 이상의 뉴로모픽(neuromorphic) 프로세서를 포함할 수도 있다. 다른 대안으로서, 처리 시스템은 단일 칩으로 통합된 프로세서, 버스 인터페이스, 사용자 인터페이스, 지원하는 회로부, 및 머신 판독가능 매체들의 적어도 일부를 갖는 ASIC(application specific integrated circuit) 로, 또는 하나 이상의 FPGA(field programmable gate array) 들, PLD(programmable logic device) 들, 제어기들, 상태 머신들, 게이팅된 로직, 이산 하드웨어 컴포넌트들, 또는 임의의 다른 적합한 회로부, 또는 본 개시물 전체에 걸쳐 설명된 다양한 기능성을 수행할 수 있는 회로들의 임의의 조합으로 구현될 수도 있다. 당업자라면, 전체 시스템에 부과되는 설계 제약 및 특정 응용들에 따라 처리 시스템을 위한 설명된 기능성을 구현하기 위한 최선의 방법을 인식할 것이다.

[0081] 머신 판독가능 매체들은 다수의 소프트웨어 모듈들을 포함할 수도 있다. 소프트웨어 모듈들은, 프로세서에 의해 실행되는 경우, 처리 시스템으로 하여금, 다양한 기능들을 수행하게 하는 명령들을 포함한다. 소프트웨어 모듈들은 송신 모듈 및 수신 모듈을 포함할 수도 있다. 각각의 소프트웨어 모듈은 단일 저장 디바이스에 상주하거나 또는 다수의 저장 디바이스들에 걸쳐 분산될 수도 있다. 예로서, 트리거링 이벤트가 일어나는 경우 소프트웨어 모듈은 하드웨어 드라이브로부터 RAM 으로 로딩될 수도 있다. 소프트웨어 모듈의 실행 중에, 프로세서는 액세스 속도를 증가시키기 위해 캐시 내로 명령들 중 일부를 로딩할 수도 있다. 다음으로, 하나 이상의 캐시 라인들이 프로세서에 의한 실행을 위해 일반 레지스터 파일 내로 로딩될 수도 있다. 하기의 소프트웨어 모듈의 기능성을 언급할 때, 해당 소프트웨어 모듈로부터 명령들을 실행하는 경우, 그러한 기능성이 프로세서에 의해 구현된다는 것이 이해될 것이다. 더욱이, 본 개시의 양태들이 프로세서, 컴퓨터, 머신, 또는 이러한 양태들을 구현하는 다른 시스템의 기능을 개선시킨다는 것을 이해해야 한다.

[0082] 소프트웨어로 구현되면, 그 기능들은 컴퓨터 판독가능 매체 상에 하나 이상의 명령 또는 코드로서 저장되거나 또는 송신될 수도 있다. 컴퓨터 판독가능 매체는 일 장소로부터 다른 장소로의 컴퓨터 프로그램의 전송을 가능하게 하는 임의의 매체를 포함하는 통신 매체 및 컴퓨터 저장 매체 양자 모두를 포함한다. 저장 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 이용가능한 매체일 수도 있다. 비한정적 예로서, 이러한 컴퓨터 판독가능 매체는 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 저장, 자기 디스크 저장 또는 다른 자기 저장 디바이스들, 또는 명령 또는 데이터 구조의 형태로 원하는 프로그램 코드를 반송 또는 저장하는데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 매체를 포함할 수 있다. 또한, 임의의 접속이 컴퓨터 판독가능 매체로 적절히 칭해진다. 예를 들어, 소프트웨어가 동축 케이블, 광섬유 케이블, 연선(twisted pair), 디지털 가입자 라인("DSL"), 또는 적외선(IR), 전파(radio), 및 마이크로파와 같은 무선 기술을 사용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 송신되는 경우, 그 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 전파, 및 마이크로파와 같은 무선 기술은 매체의 정의 내에 포함된다. 여기에 사용된 바와 같이, 디스크(disk) 및 디스크(disc)는 콤팩트 디스크(compact disc; CD), 레이저 디스크(laser disc), 광 디스크(optical disc), DVD(digital versatile disc), 플로피 디스크(floppy disk) 및 블루레이 디스크(Blu-ray® disc)를 포함하며, 여기서, 디스크(disk)는 보통 데이터를 자기적으로 재생하지만, 디스크(disc)는 레이저를 이용하여 광학적으로 데이터를 재생한다. 따라서, 일부 양태들에서 컴퓨터 판독가능 매체들은 비일시적 컴퓨터 판독가능 매체들(예를 들어, 유형의 매체들)을 포함할 수도 있다. 추가적으로, 다른 양태들에 있어서, 컴퓨터 판독가능 매체들은 일시적 컴퓨터 판독가능 매체들(예를 들어, 신호)을 포함할 수도 있다. 또한, 상기의 조합은 컴퓨터 판독가능 매체의 범위 내에 포함되어야 한다.

[0083] 따라서, 소정의 양태들은 본원에 제시된 동작들을 수행하기 위한 컴퓨터 프로그램 제품을 포함할 수도 있다.

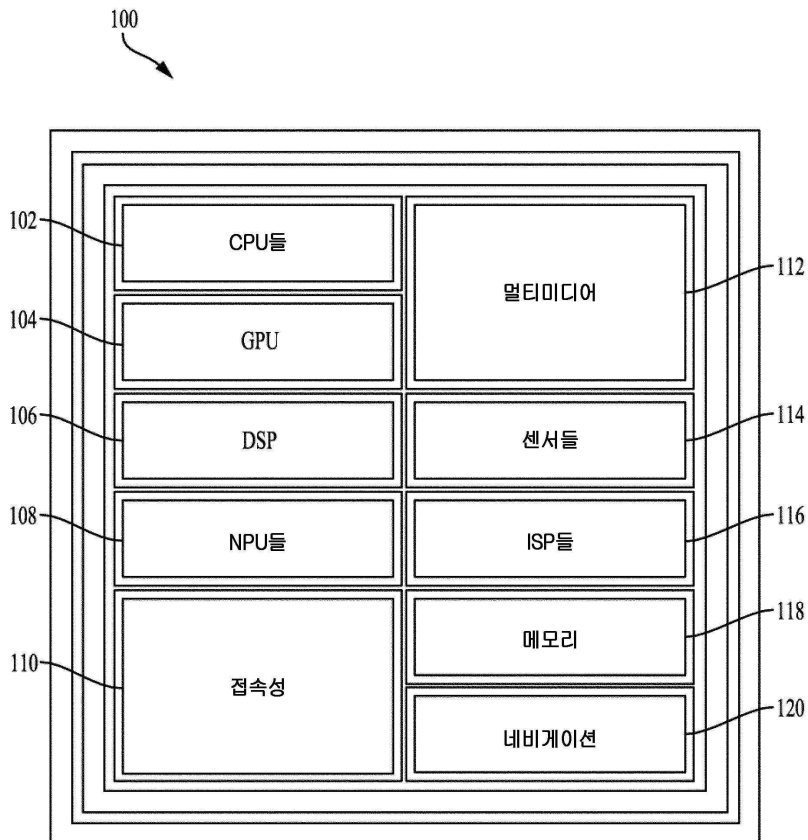
예를 들어, 이러한 컴퓨터 프로그램 제품은 저장된 (및/또는 인코딩된) 명령들을 갖는 컴퓨터 판독가능 매체를 포함할 수도있으며, 그 명령들은 본원에 설명된 동작들을 수행하기 위해 하나 이상의 프로세서들에 의해 실행가능할 수도 있다. 소정의 양태들에 있어서, 컴퓨터 프로그램 제품은 패키징 재료를 포함할 수도 있다.

[0084] 또한, 본원에 설명된 방법들 및 기법들을 수행하는 모듈들 및/또는다른 적절한 수단이 적용가능한 경우 다운로드될 수도 있거나 및/또는 그렇지 않으면 사용자 단말기 및/또는 기지국에 의해 획득될 수도 있다는 것이 인식되어야 한다. 예를 들어, 그러한 디바이스는 본원에 설명된 방법들을 수행하는 수단의 전달을 가능하게 하기 위해 서버에 연결될 수 있다. 다르게는, 본원에 기재된 다양한 방법들은 저장 수단 (예를 들어, RAM, ROM, 물리적 저장 매체, 이블테면 콤팩트 디스크 (CD) 또는 플로피 디스크 등) 을 통해 제공되어, 사용자 단말기 및/또는 기지국은, 디바이스에 저장 수단을 연결 또는 제공할 시에 그 다양한 방법들을 획득할 수 있다. 더욱이, 여기에 기재된 방법들 및 기법들을 제공하기 위한 임의의 다른 적합한 기법이 이용될 수 있다.

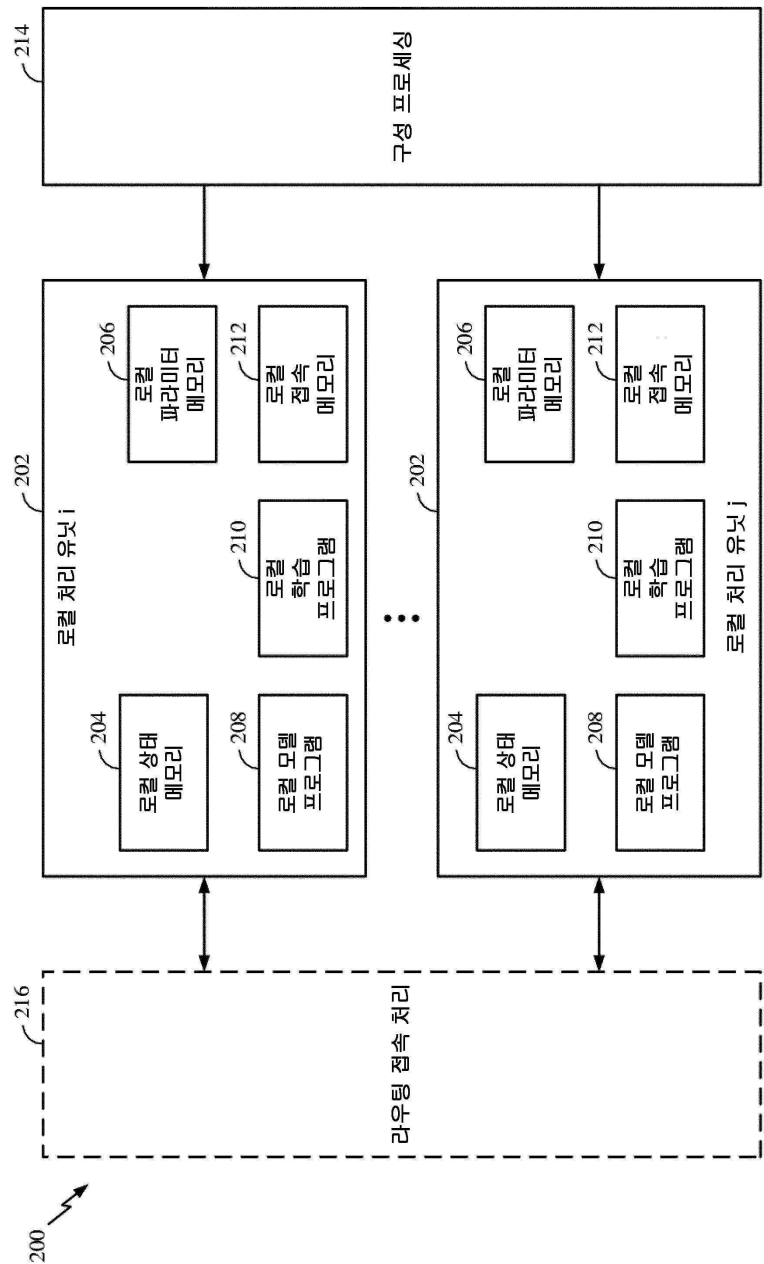
[0085] 청구항들은 위에 예시된 바로 그 구성 및 컴포넌트들에 한정되지 않는다는 것이 이해되어야 한다. 청구항들의 범위로부터 이탈함이 없이 위에서 설명된, 방법 및 장치의 배열, 동작 및 상세들에서 다양한 수정, 변경 및 변형들이 이루어질 수도 있다.

도면

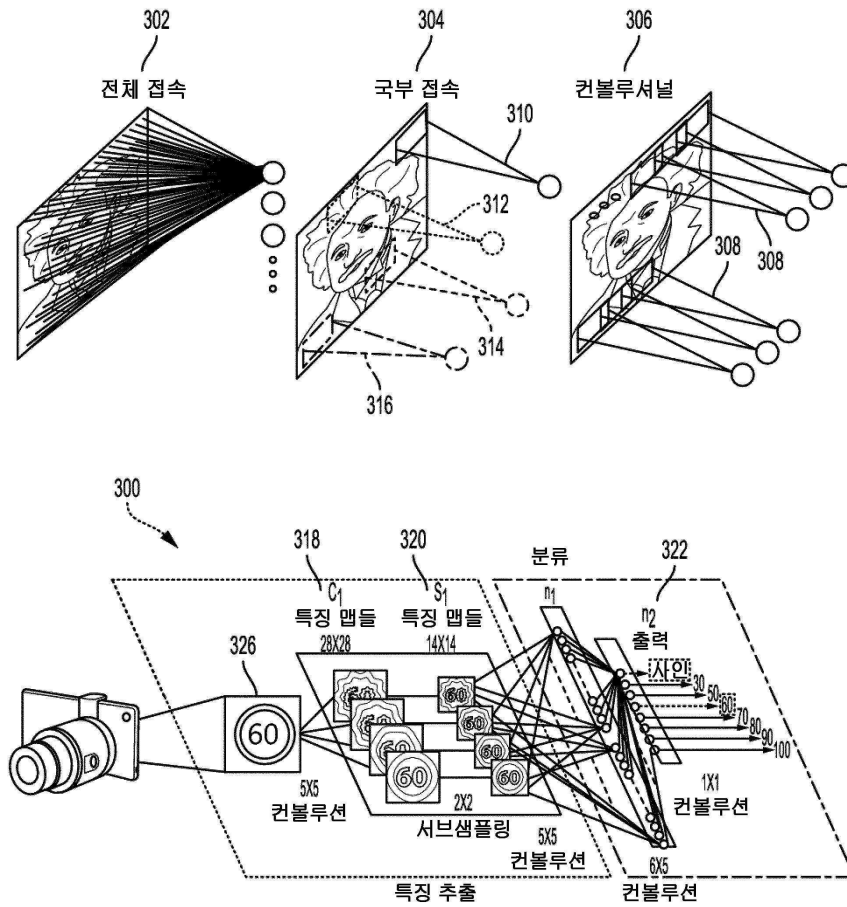
도면1



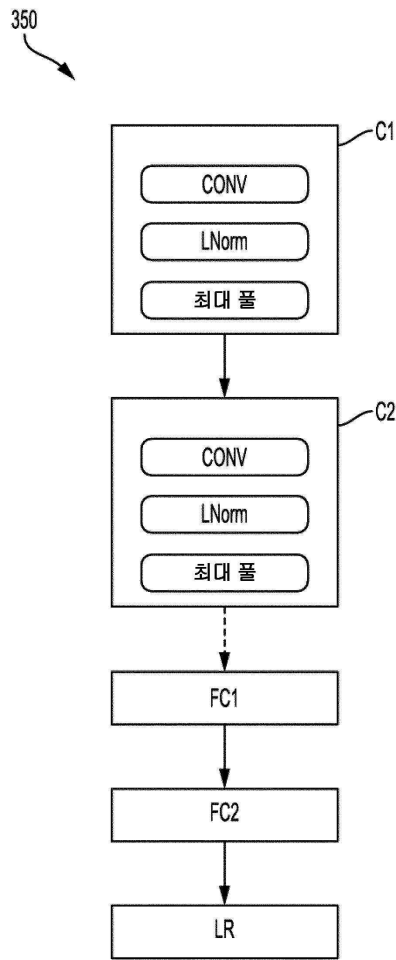
도면2



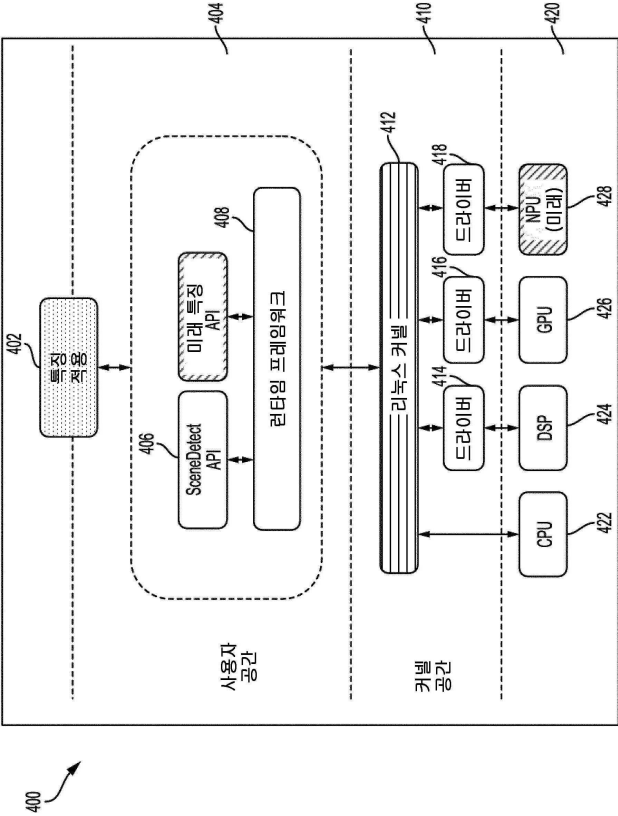
도면3a



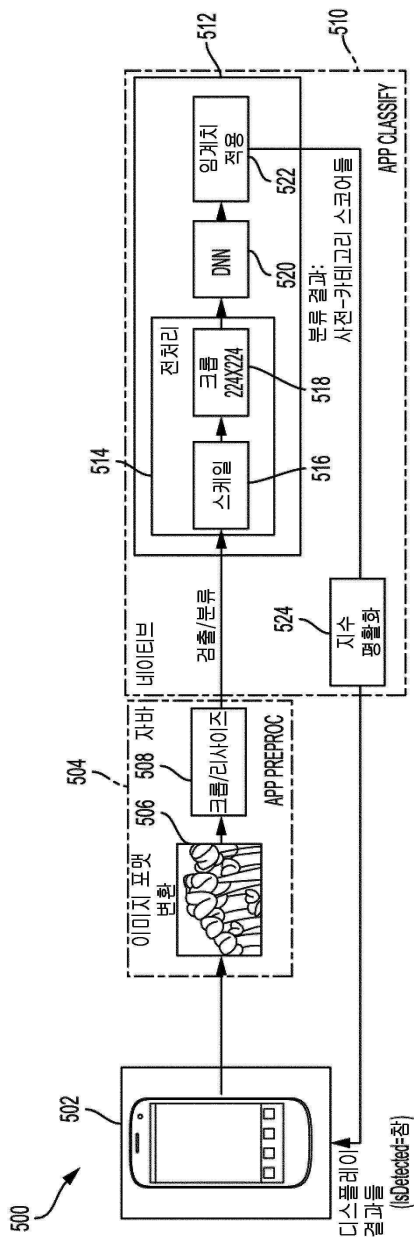
도면 3b



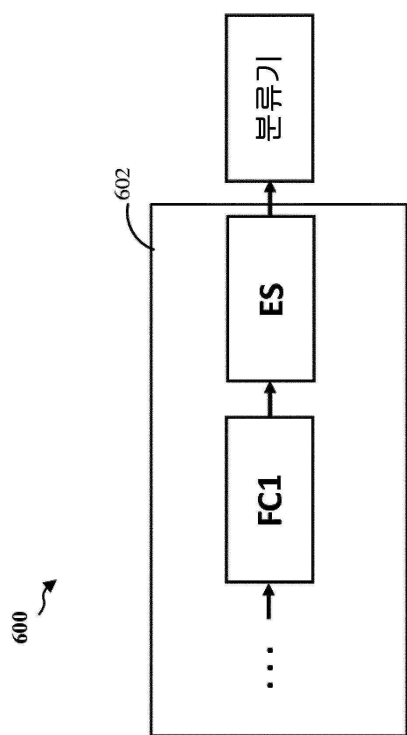
도면4



도면5



도면6



도면7

