(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0125115
(43) 공개일자 2022년09월14일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2006.01) *G06N 3/04* (2006.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06N 3/04 (2013.01)
(21) 출원번호 10-2021-0035050
(22) 출원일자 2021년03월18일
심사청구일자 2021년07월15일
(30) 우선권주장
1020210028664 2021년03월04일 대한민국(KR)

(71) 출원인
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
이세환
경기도 성남시 분당구 수내로 74, 113동 304호 (수내동, 양지마을)
심현욱
울산광역시 울주군 언양읍 유니스트길 50
이종은
울산광역시 울주군 언양읍 유니스트길 50
(74) 대리인
특허법인 무한

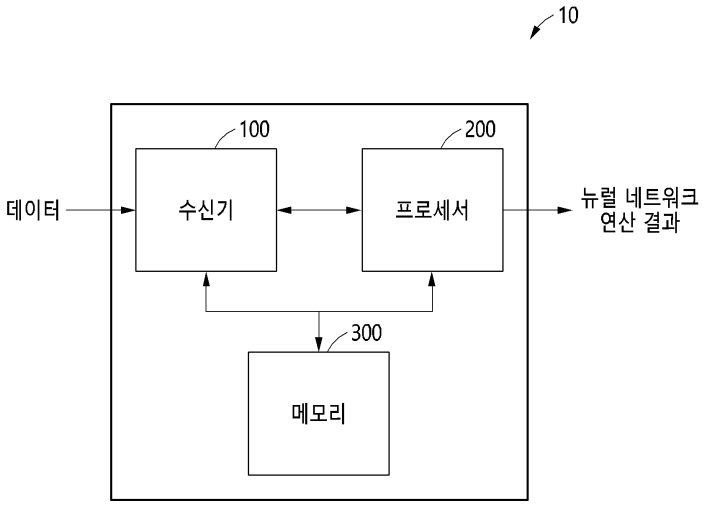
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 **희소화를 이용한 뉴럴 네트워크 연산 방법 및 장치**

(57) 요 약

희소화를 이용한 뉴럴 네트워크 연산 방법 및 장치가 개시된다. 일 실시예에 다른 뉴럴 네트워크 연산 방법은, 뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신하는 단계와, 상기 제1 임계값에 기초하여 상기 제1 활성화 기울기를 희소화(spasificate)하는 단계와, 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득하는 단계와, 상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 업데이트함으로써 제2 임계값을 계산하는 단계와, 상기 제2 활성화 기울기 및 상기 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행하는 단계를 포함한다.

대 표 도 - 도1



(52) CPC특허분류
G06N 3/0481 (2013.01)

명 세 서

청구범위

청구항 1

뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신하는 단계;

상기 제1 임계값에 기초하여 상기 제1 활성화 기울기를 희소화(spasificate)하는 단계;

희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득하는 단계;

상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 업데이트함으로써 제2 임계값을 계산하는 단계; 및

상기 제2 활성화 기울기 및 상기 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 2

제1항에 있어서,

상기 제1 활성화 기울기 및 상기 제2 활성화 기울기는,

입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함하는

뉴럴 네트워크 연산 방법.

청구항 3

제1항에 있어서,

상기 제2 임계값을 계산하는 단계는,

미리 결정된 이터레이션(iteration) 횟수만큼 반복하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 4

제1항에 있어서,

상기 제2 임계값을 계산하는 단계는,

타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 5

제4항에 있어서,

상기 제2 임계값을 계산하는 단계는,

상기 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 상기 제1 임계값에 곱함으로써 상기 제2 임계값을 계산하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 6

제1항에 있어서,

상기 제2 임계값을 계산하는 단계는,

상기 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단하는 단계; 및

상기 제한 범위를 초과하는 상기 제2 임계값을 상기 제한 범위 내의 값으로 보정하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 7

제1항에 있어서,

상기 제2 임계값을 계산하는 단계는,

상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 초기화함으로써 상기 제2 임계값을 계산하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 8

제1항에 있어서,

상기 뉴럴 네트워크를 수행하는 단계는,

상기 제2 임계값에 기초하여 상기 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성하는 단계; 및

상기 희소 데이터 및 밀집 데이터(dense data)를 이용하여 상기 뉴럴 네트워크 연산을 수행하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 9

제8항에 있어서,

상기 밀집 데이터는,

병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장되는

뉴럴 네트워크 연산 방법.

청구항 10

제1항에 있어서,

상기 뉴럴 네트워크를 수행하는 단계는,

상기 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행하는 단계

를 포함하는 뉴럴 네트워크 연산 방법.

청구항 11

뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신하는 수신기; 및

상기 제1 임계값에 기초하여 상기 제1 활성화 기울기를 희소화(spasificate)하고,

희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득하고,

상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 업데이트함으로써 제2 임계값을 계산하고,

상기 제2 활성화 기울기 및 상기 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행하는 프로세서

를 포함하는 뉴럴 네트워크 연산 장치.

청구항 12

제11항에 있어서,

상기 제1 활성화 기울기 및 상기 제2 활성화 기울기는,

입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함하는

뉴럴 네트워크 연산 장치.

청구항 13

제11항에 있어서,

상기 프로세서는,

미리 결정된 이터레이션 횟수만큼 반복하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는

뉴럴 네트워크 연산 장치.

청구항 14

제11항에 있어서,

상기 프로세서는,

타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는

뉴럴 네트워크 연산 장치.

청구항 15

제14항에 있어서,

상기 프로세서는,

상기 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 상기 제1 임계값에 곱함으로써 상

기 제2 임계값을 계산하는

뉴럴 네트워크 연산 장치.

청구항 16

제11항에 있어서,

상기 프로세서는,

상기 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단하고,

상기 제한 범위를 초과하는 상기 제2 임계값을 상기 제한 범위 내의 값으로 보정하는

뉴럴 네트워크 연산 장치.

청구항 17

제11항에 있어서,

상기 프로세서는,

상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 초기화함으로써 상기 제2 임계값을 계산하는

뉴럴 네트워크 연산 장치.

청구항 18

제11항에 있어서,

상기 프로세서는,

상기 제2 임계값에 기초하여 상기 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성하고,

상기 희소 데이터 및 밀집 데이터(dense data)를 이용하여 상기 뉴럴 네트워크 연산을 수행하는

뉴럴 네트워크 연산 장치.

청구항 19

제18항에 있어서,

상기 밀집 데이터는,

병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장되는

뉴럴 네트워크 연산 장치.

청구항 20

제11항에 있어서,

상기 프로세서는,

상기 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행하는

뉴럴 네트워크 연산 장치.

발명의 설명

기 술 분 야

[0001] 아래 실시예들은 희소화를 이용한 뉴럴 네트워크 연산 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] DNN(Deep Neural Network)의 많은 계산량을 빠르고 효율적으로 처리하기 위하여 GPU(Graphic Processing Unit), 멀티-GPU, FPGA(Field Programmable Gate Arrays), ASIC(Application-Specific Integrated Circuit) 및 이머징 장치(emerging device)에 이르기까지 많은 하드웨어에 대한 연구와, 모델 압축, 희소화(sparsification) 등의 소프트웨어에 대한 연구가 이루어지고 있다. 또한, 하드웨어 연구 및 소프트웨어 연구 모두를 고려한 공동설계(co-design)에 관한 연구도 많이 이루어지고 있다.

[0003] 하지만 종래 아키텍처 및 알고리즘들을 학습(training)에 적용시키는 데에는 문제가 있는데, 첫 번째 문제는 계산량과 메모리 요구량이 추론보다 매우 더 많다는 점이다. 또한 학습은 추론보다 높은 정밀도를 요구하는데, 추론의 정확도 하락은 재학습(re-training)으로 극복할 여지가 많지만, 학습 정확도가 낮아지는 것은 곧바로 학습 속도/인식률 하락의 문제로 이어진다.

[0004] DNN의 희소성(sparsity)은 매우 높고, 계산량과 메모리 요구량을 줄이는 데에 유용하여, 특히 추론에서 자주 활용되어 왔지만, 학습에서는 많이 활용되지 않았다.

[0005] 종래의 기울기(gradient) 희소화 기술(top-k)은 메모리 요구량을 줄이지 못하고 전처리(예: 소팅(sorting)) 시간을 추가로 요구한다. 또한 종래의 희소 행렬곱 아키텍처들(예: EIE(Efficient Inference Engine))은 원소들의 불규칙성 때문에 줄어든 계산량에 비해 컨트롤러와 같은 하드웨어에 오버헤드(overhead)가 크게 발생한다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0006] 일 실시예에 다른 뉴럴 네트워크 연산 방법은, 뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신하는 단계와, 상기 제1 임계값에 기초하여 상기 제1 활성화 기울기를 희소화(spasificate)하는 단계와, 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득하는 단계와, 상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 업데이트함으로써 제2 임계값을 계산하는 단계와, 상기 제2 활성화 기울기 및 상기 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행하는 단계를 포함한다.

[0007] 상기 제1 활성화 기울기 및 상기 제2 활성화 기울기는, 입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함할 수 있다.

[0008] 상기 제2 임계값을 계산하는 단계는, 미리 결정된 이터레이션(iteration) 횟수만큼 반복하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는 단계를 포함할 수 있다.

[0009] 상기 제2 임계값을 계산하는 단계는, 타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산하는 단계를 포함할 수 있다.

[0010] 상기 제2 임계값을 계산하는 단계는, 상기 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 상기 제1 임계값에 곱함으로써 상기 제2 임계값을 계산하는 단계를 포함할 수 있다.

[0011] 상기 제2 임계값을 계산하는 단계는, 상기 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단하는 단계와, 상기 제한 범위를 초과하는 상기 제2 임계값을 상기 제한 범위 내의 값으로 보정하는 단계를 포함할 수 있다.

[0012] 상기 제2 임계값을 계산하는 단계는, 상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 초기화함으로써 상기 제2 임계값을 계산하는 단계를 포함할 수 있다.

[0013] 상기 뉴럴 네트워크를 수행하는 단계는, 상기 제2 임계값에 기초하여 상기 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성하는 단계와, 상기 희소 데이터 및 밀집 데이터(dense data)를 이용하여 상기 뉴럴 네트워크 연산을 수행하는 단계를 포함할 수 있다.

[0014] 상기 밀집 데이터는, 병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장될 수 있다.

[0015] 상기 뉴럴 네트워크를 수행하는 단계는, 상기 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행하는 단계를 포함할 수 있다.

[0016] 일 실시예에 따른 뉴럴 네트워크 연산 장치는, 뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신하는 수신기와, 상기 제1 임계값에 기초하여 상기 제1 활성화 기울기를 희소화(spasificate)하고, 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득하고, 상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 업데이트함으로써 제2 임계값을 계산하고, 상기 제2 활성화 기울기 및 상기 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행하는 프로세서를 포함할 수 있다.

[0017] 상기 제1 활성화 기울기 및 상기 제2 활성화 기울기는, 입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함할 수 있다.

[0018] 상기 프로세서는, 미리 결정된 이터레이션 횟수만큼 반복하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산할 수 있다.

[0019] 상기 프로세서는, 타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 상기 제1 임계값을 업데이트함으로써 상기 제2 임계값을 계산할 수 있다.

[0020] 상기 프로세서는, 상기 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 상기 제1 임계값에 곱함으로써 상기 제2 임계값을 계산할 수 있다.

[0021] 상기 프로세서는, 상기 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단하고, 상기 제한 범위를 초과하는 상기 제2 임계값을 상기 제한 범위 내의 값으로 보정할 수 있다.

[0022] 상기 프로세서는, 상기 제2 활성화 기울기에 기초하여 상기 제1 임계값을 초기화함으로써 상기 제2 임계값을 계산할 수 있다.

[0023] 상기 프로세서는, 상기 제2 임계값에 기초하여 상기 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성하고, 상기 희소 데이터 및 밀집 데이터(dense data)를 이용하여 상기 뉴럴 네트워크 연산을 수행할 수 있다.

[0024] 상기 밀집 데이터는, 병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장될 수 있다.

[0025] 상기 프로세서는, 상기 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행할 수 있다.

도면의 간단한 설명

[0026] 도 1은 일 실시예에 따른 뉴럴 네트워크 연산 장치의 개략적인 블록도를 나타낸다.

도 2는 도 1에 도시된 뉴럴 네트워크 연산 장치의 구현의 일 예를 나타낸다.

도 3은 임계값을 업데이트하는 과정을 나타낸다.

도 4는 도 1에 도시된 뉴럴 네트워크 연산 장치의 구현의 다른 예를 나타낸다.

도 5는 도 1에 도시된 뉴럴 네트워크 연산 장치를 적용한 단말의 예를 나타낸다.

도 6은 도 1에 도시된 뉴럴 네트워크 연산 장치의 동작의 흐름도를 나타낸다.

발명을 실시하기 위한 구체적인 내용

[0027] 실시예들에 대한 특정한 구조적 또는 기능적 설명들은 단지 예시를 위한 목적으로 개시된 것으로서, 다양한 형태로 변경되어 구현될 수 있다. 따라서, 실제 구현되는 형태는 개시된 특정 실시예로만 한정되는 것이 아니며, 본 명세서의 범위는 실시예들로 설명한 기술적 사상에 포함되는 변경, 균등물, 또는 대체물을 포함한다.

[0028] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 이런 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 해석되어야 한다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.

[0029] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다.

[0030] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 설명된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함으로 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0031] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 해당 기술분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0032] 이하, 실시예들을 첨부된 도면들을 참조하여 상세하게 설명한다. 첨부 도면을 참조하여 설명함에 있어, 도면부호에 관계없이 동일한 구성 요소는 동일한 참조 부호를 부여하고, 이에 대한 중복되는 설명은 생략하기로 한다.

[0034] 도 1은 일 실시예에 따른 뉴럴 네트워크 연산 장치의 개략적인 블록도를 나타낸다.

[0035] 도 1을 참조하면, 뉴럴 네트워크 연산 장치(10)는 희소화를 이용하여 뉴럴 네트워크 연산을 수행할 수 있다. 뉴럴 네트워크 연산 장치(10)는 데이터를 뉴럴 네트워크를 이용하여 처리함으로써 뉴럴 네트워크 연산 결과를 출력할 수 있다.

[0036] 희소성(sparsity)은 모든 엘리먼트에 대한 0 값을 갖는 엘리먼트의 비율을 의미할 수 있다. 희소화는 뉴럴 네트워크 연산을 수행하는데 있어, 특정한 데이터들을 배제(skipping)하는 것을 의미할 수 있다. 예를 들어, 희소화는 0 또는 0이 아닌 값(non-zero element)들을 뉴럴 네트워크 연산으로부터 배제하는 것을 의미할 수 있다. 희소성은 학습 과정 중에 텐서(tensor)들에서 발견될 수 있다. 희소성은 뉴럴 네트워크에 포함된 레이어에 따라 상이할 수 있다.

[0037] 뉴럴 네트워크 연산 장치(10)는 뉴럴 네트워크에 연산에 사용되는 데이터의 일부를 희소화하여 처리함으로써 연산량을 줄이고, 부하(load)의 불균형을 해소할 수 있다.

[0038] 뉴럴 네트워크 연산 장치(10)는 뉴럴 네트워크를 학습시킬 수 있다. 뉴럴 네트워크 연산 장치(10)는 학습된 뉴럴 네트워크에 기초하여 추론을 수행할 수 있다.

[0039] 뉴럴 네트워크 연산 장치(10)는 가속기를 이용하여 뉴럴 네트워크 연산을 수행할 수 있다. 뉴럴 네트워크 연산 장치(10)는 가속기 내부 또는 외부에 구현될 수 있다.

[0040] 가속기는 GPU(Graphics Processing Unit), FPGA(Field Programmable Gate Array), ASIC(Application Specific Integrated Circuit) 또는 AP(Application Processor)를 포함할 수 있다. 또한, 가속기는 가상 머신(Virtual Machine)와 같이 소프트웨어 컴퓨팅 환경으로 구현될 수도 있다.

[0041] 뉴럴 네트워크(또는 인공 신경망)는 기계학습과 인지과학에서 생물학의 신경을 모방한 통계학적 학습 알고리즘을 포함할 수 있다. 뉴럴 네트워크는 시냅스의 결합으로 네트워크를 형성한 인공 뉴런(노드)이 학습을 통해 시냅스의 결합 세기를 변화시켜, 문제 해결 능력을 가지는 모델 전반을 의미할 수 있다.

[0042] 뉴럴 네트워크의 뉴런은 가중치 또는 바이어스의 조합을 포함할 수 있다. 뉴럴 네트워크는 하나 이상의 뉴런 또는 노드로 구성된 하나 이상의 레이어(layer)를 포함할 수 있다. 뉴럴 네트워크는 뉴런의 가중치를 학습을 통해 변화시킴으로써 임의의 입력으로부터 예측하고자 하는 결과를 추론할 수 있다.

[0043] 뉴럴 네트워크는 심층 뉴럴 네트워크 (Deep Neural Network)를 포함할 수 있다. 뉴럴 네트워크는 CNN(Convolutional Neural Network), RNN(Recurrent Neural Network), 퍼셉트론(perceptron), 다층 퍼셉트론

(multilayer perceptron), FF(Feed Forward), RBF(Radial Basis Network), DFF(Deep Feed Forward), LSTM(Long Short Term Memory), GRU(Gated Recurrent Unit), AE(Auto Encoder), VAE(Variational Auto Encoder), DAE(Denoising Auto Encoder), SAE(Sparse Auto Encoder), MC(Markov Chain), HN(Hopfield Network), BM(Boltzmann Machine), RBM(Restricted Boltzmann Machine), DBN(Depp Belief Network), DCN(Deep Convolutional Network), DN(Deconvolutional Network), DCIGN(Deep Convolutional Inverse Graphics Network), GAN(Generative Adversarial Network), LSM(Liquid State Machine), ELM(Extreme Learning Machine), ESN(Echo State Network), DRN(Deep Residual Network), DNC(Differentiable Neural Computer), NTM(Neural Turning Machine), CN(Capsule Network), KN(Kohonen Network) 및 AN(Attention Network)를 포함할 수 있다.

[0044] 뉴럴 네트워크 연산 장치(10)는 마더보드(motherboard)와 같은 인쇄 회로 기판(printed circuit board(PCB)), 집적 회로(integrated circuit(IC)), 또는 SoC(system on chip)로 구현될 수 있다. 예를 들어, 뉴럴 네트워크 연산 장치(10)는 애플리케이션 프로세서(application processor)로 구현될 수 있다.

[0045] 또한, 뉴럴 네트워크 연산 장치(10)는 PC(personal computer), 데이터 서버, 또는 휴대용 장치 내에 구현될 수 있다.

[0046] 휴대용 장치는 랩탑(laptop) 컴퓨터, 이동 전화기, 스마트 폰(smart phone), 태블릿(tablet) PC, 모바일 인터넷 디바이스(mobile internet device(MID)), PDA(personal digital assistant), EDA(enterprise digital assistant), 디지털 스틸 카메라(digital still camera), 디지털 비디오 카메라(digital video camera), PMP(portable multimedia player), PND(personal navigation device 또는 portable navigation device), 휴대용 게임 콘솔(handheld game console), e-북(e-book), 또는 스마트 디바이스(smart device)로 구현될 수 있다. 스마트 디바이스는 스마트 와치(smart watch), 스마트 밴드(smart band), 또는 스마트 링(smart ring)으로 구현될 수 있다.

[0047] 뉴럴 네트워크 연산 장치(10)는 수신기(100) 및 프로세서(200)를 포함한다. 뉴럴 네트워크 연산 장치(10)는 메모리(300)를 더 포함할 수 있다.

[0048] 수신기(100)는 수신 인터페이스를 포함할 수 있다. 수신기(100)는 뉴럴 네트워크 연산과 관련된 활성화 기울기 및 희소화를 위한 임계값을 수신할 수 있다. 예를 들어, 수신기(100)는 뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기(activation gradient) 및 제1 임계값(threshold)을 수신할 수 있다. 수신기(100)는 수신한 활성화 기울기 및 임계값을 프로세서(200)로 출력할 수 있다.

[0049] 프로세서(200)는 메모리(300)에 저장된 데이터를 처리할 수 있다. 프로세서(200)는 메모리(300)에 저장된 컴퓨터로 읽을 수 있는 코드(예를 들어, 소프트웨어) 및 프로세서(200)에 의해 유발된 인스트럭션(instruction)들을 실행할 수 있다.

[0050] "프로세서(200)"는 목적하는 동작들(desired operations)을 실행시키기 위한 물리적인 구조를 갖는 회로를 가지는 하드웨어로 구현된 데이터 처리 장치일 수 있다. 예를 들어, 목적하는 동작들은 프로그램에 포함된 코드(code) 또는 인스트럭션들(instructions)을 포함할 수 있다.

[0051] 예를 들어, 하드웨어로 구현된 데이터 처리 장치는 마이크로프로세서(microprocessor), 중앙 처리 장치(central processing unit), 프로세서 코어(processor core), 멀티-코어 프로세서(multi-core processor), 멀티프로세서(multiprocessor), ASIC(Application-Specific Integrated Circuit), FPGA(Field Programmable Gate Array)를 포함할 수 있다.

[0052] 프로세서(200)는 제1 임계값에 기초하여 제1 활성화 기울기를 희소화(spasificate)할 수 있다. 프로세서(200)는 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득할 수 있다.

[0053] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다. 프로세서(200)는 미리 결정된 이터레이션(iteration) 횟수만큼 반복하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다.

[0054] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 초기화함으로써 제2 임계값을 계산할 수 있다. 제1 임계값을 초기화하는 과정은 도 3을 참조하여 자세하게 설명한다.

[0055] 프로세서(200)는 타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여

제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다. 프로세서(200)는 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 제1 임계값에 곱함으로써 제2 임계값을 계산할 수 있다.

[0056] 프로세서(200)는 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단할 수 있다. 프로세서(200)는 제한 범위를 초과하는 제2 임계값을 제한 범위 내의 값으로 보정할 수 있다.

[0057] 프로세서(200)는 제2 활성화 기울기 및 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행할 수 있다. 프로세서(200)는 제2 임계값에 기초하여 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성할 수 있다.

[0058] 프로세서(200)는 희소 데이터 및 밀집 데이터(dense data)를 이용하여 뉴럴 네트워크 연산을 수행할 수 있다. 밀집 데이터는 병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장될 수 있다. 희소 데이터 및 밀집 데이터는 도 2를 참조하여 자세하게 설명한다.

[0059] 프로세서(200)는 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행할 수 있다.

[0060] 제1 활성화 기울기 및 상기 제2 활성화 기울기는 입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함할 수 있다.

[0061] 메모리(300)는 프로세서(200)에 의해 실행가능한 인스트럭션들(또는 프로그램)을 저장할 수 있다. 예를 들어, 인스트럭션들은 프로세서의 동작 및/또는 프로세서의 각 구성의 동작을 실행하기 위한 인스트럭션들을 포함할 수 있다.

[0062] 메모리(300)는 휘발성 메모리 장치 또는 불휘발성 메모리 장치로 구현될 수 있다.

[0063] 휘발성 메모리 장치는 DRAM(dynamic random access memory), SRAM(static random access memory), T-RAM(thyristor RAM), Z-RAM(zero capacitor RAM), 또는 TTRAM(Twin Transistor RAM)으로 구현될 수 있다.

[0064] 불휘발성 메모리 장치는 EEPROM(Electrically Erasable Programmable Read-Only Memory), 플래시(flash) 메모리, MRAM(Magnetic RAM), 스핀전달토크 MRAM(Spin-Transfer Torque(STT)-MRAM), Conductive Bridging RAM(CBRAM), FeRAM(Ferroelectric RAM), PRAM(Phase change RAM), 저항 메모리(Resistive RAM(RRAM)), 나노 튜브 RRAM(Nanotube RRAM), 폴리머 RAM(Polymer RAM(PoRAM)), 나노 부유 게이트 메모리(Nano Floating Gate Memory(NFGM)), 홀로그래픽 메모리(holographic memory), 분자 전자 메모리 소자(Molecular Electronic Memory Device), 또는 절연 저항 변화 메모리(Insulator Resistance Change Memory)로 구현될 수 있다.

[0066] 도 2는 도 1에 도시된 뉴럴 네트워크 연산 장치의 구현의 일 예를 나타낸다.

[0067] 도 2를 참조하면, 프로세서(200)는 뉴럴 네트워크와 관련된 데이터를 희소화할 수 있다. 프로세서(200)는 희소화된 데이터를 이용하여 뉴럴 네트워크를 학습시거나 뉴럴 네트워크 연산을 수행할 수 있다.

[0068] 뉴럴 네트워크의 학습 과정은 순전파(feedforward) 연산 및 역전파(backpropagation) 연산을 포함할 수 있다. 순전파 연산은 입력 레이어에서 히든 레이어를 거쳐 출력 레이어 방향으로 이동하면서 각 레이어의 입력과 가중치를 이용하여 연산을 수행함으로써 손실 함수의 값을 계산하는 과정을 의미할 수 있다.

[0069] 역전파 연산은 뉴럴 네트워크의 입력과 출력을 알고 있는 상태에서 손실 함수의 값이 최소가 되도록 뉴럴 네트워크의 가중치들을 조절(또는, 업데이트)하는 과정을 의미할 수 있다.

[0070] 프로세서(200)는 역전파 연산을 수행하는 동안에, 활성화 기울기를 임계값과 비교하여 희소화하며 뉴럴 네트워크를 학습시킬 수 있다. 프로세서(200)는 희소화를 이용하여 학습의 약 2/3를 차지하는 역전파 및 가중치 업데이트를 위한 계산량을 감소시킬 수 있다.

[0071] 프로세서(200)는 활성화 기울기를 실시간으로 임계값과 비교함으로써 전처리가 필요한 종래의 뉴럴 네트워크 학습 방식에 비해 실질적인 메모리 접근량 및 계산량을 감소시킬 수 있다.

[0072] 프로세서(200)는 희소성을 높게 유지하고, 정확도를 하락하지 않도록 임계값을 결정할 수 있다. 프로세서(200)는 임계값을 결정하는 이터레이션에서 이전의 이터레이션의 희소화 분포를 이용하여 동적으로(dynamically) 다음 이터레이션의 임계값을 조절할 수 있다.

[0073] 뉴럴 네트워크 연산 장치(10)는 희소 버퍼(sparse buffer), MAC 어레이(220), 출력 버퍼(230), 제1 밀집 버퍼(240), 제2 밀집 버퍼(250) 및 희소화기(sparsificator)(210)를 포함할 수 있다. 예를 들어, 희소 버퍼(210), 출력 버퍼(230), 제1 밀 집 버퍼(240) 및 제2 밀집 버퍼(250)는 메모리(예: 도 1의 메모리(300))에 포함될 수 있다.

[0074] MAC 어레이(220) 및 희소화기(260)는 프로세서(예: 도 1의 프로세서(200))에 포함될 수 있다. MAC 어레이(220)는 별도로, 뉴럴 네트워크 연산 장치(10)의 외부에 구현될 수도 있다.

[0075] MAC 어레이(220)는 희소 버퍼(210)로부터 수신한 희소 입력(sparse input)을 순차적(serial) 으로처리하고 제1 밀집 버퍼(240) 및 제2 밀집 버퍼(250)로부터 밀집 입력(dense input)을 병렬화하여, 부하 불균형의 의한 이용율(utilization) 감소를 없앨 수 있다.

[0076] 프로세서(200)는 활성화 기울기(예: 제2 활성화 기울기) 및 임계값(예: 제2 임계값)에 기초하여 뉴럴 네트워크 연산을 수행할 수 있다. 프로세서(200)는 임계값에 기초하여 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성할 수 있다. 희소 데이터는 희소 버퍼(210)에 저장될 수 있다.

[0077] 프로세서(200)는 희소 데이터 및 밀집 데이터(dense data)를 이용하여 뉴럴 네트워크 연산을 수행할 수 있다. 밀집 데이터는 병렬화된(parallelized) 복수의 밀집 버퍼(예: 제1 밀집 버퍼(240) 및 제2 밀집 버퍼(250))에 저장될 수 있다.

[0078] 도 2의 예시는 역전파 연산과 가중치의 업데이트를 처리하는 과정에서의 데이터 플로우(dataflow)를 나타낼 수 있다. 도 2의 예시에서, IA, W, GI, GO 및 GW는 각각 입력 활성화(input activation), 가중치(weight), 입력 활성화 기울기(gradient w.r.t. input activation), 출력 활성화 기울기(gradient w.r.t. output activation), 가중치 기울기(gradient w.r.t. weight)를 의미할 수 있다.

[0079] GO, W, GI 중에서 왼편에 기재된 데이터는 역전파시의 데이터 플로우를 나타내고, GO, IA, GW와 같이 오른편에 기재된 데이터는 가중치 업데이트 과정에서의 데이터 플로우를 의미할 수 있다. 희소화기(260)의 출력인 GI는 역전파 시의 데이터 플로우를 의미할 수 있다.

[0080] 도 2의 예시는 입력 활성화 기울기를 희소화하는 실시예에 대하여 표현되어 있지만, 실시예에 따라, 출력 활성화 기울기 또는 가중치 기울기에 대한 희소화도 가능하다.

[0081] 도 2에 도시된 데이터들은 DMA(Direct Memory Access)를 통해 통신할 수 있다. 활성화 기울기들은 희소 버퍼(210)에 로드(load)될 수 있고, 입력 활성화 및 가중치는 밀집 버퍼(예: 제1 밀집 버퍼(240) 및 제2 밀집 버퍼(250))에 로드될 수 있다.

[0082] MAC 어레이(220)는 희소 버퍼에서 순차적으로 기울기(예: 입력 활성화 기울기, 출력 활성화 기울기 또는 가중치 기울기)를 수신하여 병렬화된 밀집 데이터와 행렬곱을 수행할 수 있다.

[0083] MAC 어레이(220)는 T개의 곱셈기 및 덧셈기를 포함할 수 있다. T는 하드웨어 구현 예에 따라 조절될 수 있다.

[0084] 희소화기(260)는 행렬곱 결과인 다음 레이어의 활성화 기울기의 통계값을 계산할 수 있다. 활성화 기울기의 통계값은 활성화 기울기의 확률 분포를 포함할 수 있다. 예를 들어, 활성화 기울기의 통계값은 활성화 기울기의 평균, 분산, 표준 편차를 포함할 수 있다.

[0085] 희소화기(260)는 계산된 통계값에 기초하여 임계값(예: 제1 임계값)을 업데이트할 수 있다. 희소화기(260)는 활성화 기울기를 업데이트하여 메모리(300)(예: DRAM)으로 출력할 수 있다.

[0087] 도 3은 임계값을 업데이트하는 과정을 나타낸다.

[0088] 도 3을 참조하면, 프로세서(예: 도 1의 프로세서(200))는 초기 임계값을 설정할 수 있다(310). 예를 들어, 프로세서(200)는 Θ_1에 0을 대입함으로써 초기 임계값을 설정할 수 있다.

[0089] 프로세서(200)는 임계값에 기초하여 활성화 기울기(예: 입력 활성화 기울기)를 계산할 수 있다(320). 프로세서(200)는 도 2에서 설명한 것과 같이 뉴럴 네트워크 연산을 이용하여 활성화 기울기를 계산할 수 있다.

[0090] 프로세서(200)는 현재 이터레이션의 임계값에 기초하여 활성화 기울기를 희소화할 수 있다(330). 프로세서(200)는 제1 임계값에 기초하여 제1 활성화 기울기를 희소화(sparsificate)할 수 있다. 예를 들어, 프로세서

(200)는 제1 임계값 보다 작은 절대값을 갖는 활성화 기울기들을 희소화할 수 있다.

[0091] 임계값의 업데이트가 이루어지기 전에, 프로세서(200)는 초기 임계값(예: 0)에 기초하여 활성화 기울기를 희소화할 수 있다. 임계값의 업데이트가 이루어지는 경우, 프로세서(200)는 업데이트된 임계값에 기초하여 활성화 기울기를 희소화할 수 있다.

[0092] 프로세서(200)는 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득할 수 있다. 예를 들어, 프로세서(200)는 MAC 어레이(예: 도 2의 MAC 어레이(220))를 이용한 뉴럴 네트워크 연산을 통해 제2 활성화 기울기를 획득할 수 있다. 제1 활성화 기울기의 희소화는 이전 이터레이션에 수행된 희소화를 의미할 수 있다.

[0093] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다. 프로세서(200)는 미리 결정된 이터레이션(iteration) 횟수만큼 반복하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다.

[0094] 이터레이션의 시작을 위해, 프로세서(200)는 이터레이션 루프(loop)의 인덱스가 1인지 여부를 판단할 수 있다(340). i가 1인 경우 프로세서(200)는 제1 임계값을 초기화함으로써 제2 임계값을 계산할 수 있다(350).

[0095] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 초기화함으로써 제2 임계값을 계산할 수 있다. 예를 들어, 프로세서(200)는 수학식 1을 이용하여 제1 임계값을 초기화함으로써 제2 임계값을 계산할 수 있다.

수학식 1

[0097] $$\theta_{i+1} \leftarrow \frac{\max(abs(GI))}{100}$$

[0099] 여기서, θ_{i+1}는 다음 이터레이션을 위한 임계값(예: 제2 임계값)을 의미할 수 있다.

[0100] i가 1이 아닌 경우, 프로세서(200)는 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다(360). 프로세서(200)는 타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다.

[0101] 프로세서(200)는 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 제1 임계값에 곱함으로써 제2 임계값을 계산할 수 있다. 예를 들어, 프로세서(200)는 수학식 2를 이용하여 제2 임계값을 계산할 수 있다.

수학식 2

[0103] $$\theta_{i+1} \leftarrow \theta_i \cdot \frac{s^*}{s_i}$$

[0105] 여기서, s^*는 타겟 희소성을 의미하고, s_i는 현재 이터레이션의 희소성을 의미할 수 있다.

[0106] 프로세서(200)는 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단할 수 있다(370). 프로세서(200)는 제한 범위를 초과하는 제2 임계값을 제한 범위 내의 값으로 보정할 수 있다. 예를 들어, 제한 범위는 제1 임계값 보다 20% 작은 값으로부터 20% 큰 값까지일 수 있다. 제한 범위는 실시예에 따라 상이할 수 있다.

[0107] 프로세서(200)는 상술한 임계값 업데이트 과정을 통해, 활성화 기울기를 정렬(sort)하여 상위 k개(top-k)를 뽑

는 방식에 비해, 활성화 기울기들 사이의 의존도(dependency)를 줄임으로써 부분 활성화 기울기(partial activation gradient) 계산 결과를 바로 희소화함으로써 메모리(예: 도 1의 메모리(300))와의 통신량을 절약하고, 하드웨어 구현을 간소화시킬 수 있다.

[0108] 프로세서(200)는 i가 미리 결정된 이터레이션 값과 같은지 여부를 판단할 수 있다(380). i가 미리 결정된 이터레이션 값과 다른 경우, 프로세서(200)는 i에 1을 더할 수 있다(390). 그 후, 프로세서(200)는 업데이트된 임계값(제2 임계값)에 기초하여 단계 320을 다시 수행할 수 있다.

[0109] i가 미리 결정된 이터레이션 값과 같은 경우, 프로세서(200)는 임계값의 업데이트를 완료할 수 있다. 프로세서(200)는 업데이트된 임계값을 뉴럴 네트워크의 여러 구성에 공유할 수 있다. 예를 들어, 프로세서(200)는 업데이트된 임계값을 뉴럴 네트워크의 레이어들 중에서 일부 레이어에 공유할 수 있다.

[0111] 도 4는 도 1에 도시된 뉴럴 네트워크 연산 장치의 구현의 다른 예를 나타낸다.

[0112] 도 4를 참조하면, 뉴럴 네트워크 연산 장치(10)는 제1 희소 버퍼(410), 제2 희소 버퍼(420), 컨트롤러(430), 연산 타일(440) 및 희소화기(450)를 포함할 수 있다.

[0113] 제1 희소 버퍼(410)는 읽기(read) 및 쓰기(write) 포트(port)를 포함할 수 있다. 제1 희소 버퍼(410)는 로컬 버스(local bus)를 통해 희소화된 활성화 기울기를 연산 타일(440)로 출력할 수 있다.

[0114] 제2 희소 버퍼(410)는 읽기(read) 및 쓰기(write) 포트(port)를 포함할 수 있다. 제2 희소 버퍼(410)는 로컬 버스를 통해 희소화된 활성화 기울기를 연산 타일(440)로 출력할 수 있다.

[0115] 제1 희소 버퍼(410) 및 제2 희소 버퍼(420)의 동작은 도 2의 희소 버퍼(210)와 동일할 수 있다.

[0116] 컨트롤러(430)는 데이터 처리를 위한 신호들, DRAM 및 버퍼 주소를 분배할 수 있다. 컨트롤러(430)는 프로세서(예: 도 1의 프로세서(200))에 포함될 수 있다.

[0117] 연산 타일(440)은 복수의 병렬 MAC 연산기들을 포함할 수 있다. 연산 타일(440)에 포함된 복수의 병렬 MAC 연산기들의 동작은 도 2의 MAC 어레이(220)와 동일할 수 있다.

[0118] 희소화기(450)는 임계값을 업데이트하면서 활성화 기울기에 대한 희소화를 수행할 수 있다. 희소화기(450)는 프로세서(200)에 포함될 수 있다. 희소화기(450)의 동작은 도 2의 희소화기(260)와 동일할 수 있다.

[0119] 프로세서(200)는 행렬곱이 수행되는 하나의 입력 또는 복수의 입력에 대해서 희소화를 수행할 수 있다. 따라서, 프로세서(200)는 서로 다른 종류의 데이터들에 대하여 희소화를 수행하고, 희소화된 데이터들 간의 행렬곱을 수행할 수 있다.

[0120] 프로세서(200)는 희소화된 데이터들 간의 행렬 곱이 사용되는 뉴럴 네트워크 추론을 수행할 수 있다. 프로세서(200)는 병렬 곱셈 및 덧셈을 포함하는 다양한 뉴럴 네트워크 연산을 수행할 수 있다.

[0122] 도 5는 도 1에 도시된 뉴럴 네트워크 연산 장치를 적용한 단말의 예를 나타낸다.

[0123] 도 5를 참조하면, 뉴럴 네트워크 연산 장치(540)는 임의의 단말(500)에 구현될 수 있다. 단말(500)은 카메라(510), 프로세서(520), 이미지 분별기(530) 및 뉴럴 네트워크 연산 장치(540)를 포함할 수 있다. 프로세서(520)는 단말(500)의 동작을 제어할 수 있다.

[0124] 뉴럴 네트워크 연산 장치(540)는 희소화를 이용하여 경량 하드웨어 상에서 뉴럴 네트워크 학습 및 추론을 수행할 수 있다.

[0125] 일반적으로, 단말(500)(예: 스마트폰)은 사용할 수 있는 전력 에너지의 제약이 있기 때문에 추론에 비해 많은 학습 계산량을 감당하기 어렵지만, 뉴럴 네트워크 연산 장치(540)는 상술한 희소화 방식을 이용하여 계산량 및 메모리 접근량을 획기적으로 감소시킴으로써 모바일 기기에서 에너지를 절약할 수 있다.

[0126] 이미지 분별기(530)는 이미지 추론을 수행할 수 있다. 도 5의 예시에서 이미지 분별기(530)를 별도로 표현했지만, 뉴럴 네트워크 연산 장치(10)와 동일한 형태의 뉴럴 네트워크 연산을 이용하여 추론을 수행하는 경우(예: 희소 데이터와 밀집 데이터 간의 행렬곱)에는 별도의 이미지 분별기(530)가 필요하지 않을 수 있다.

[0128] 도 6은 도 1에 도시된 뉴럴 네트워크 연산 장치의 동작의 흐름도를 나타낸다.

[0129] 도 6을 참조하면, 수신기(100)는 뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기 및 제1 임계값을 수신할 수 있다(610).

[0130] 프로세서(200)는 제1 임계값에 기초하여 제1 활성화 기울기를 희소화(spasificate)할 수 있다(630).

[0131] 프로세서(200)는 희소화된 제1 활성화 기울기에 기초하여 뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득할 수 있다(650).

[0132] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다(670). 프로세서(200)는 미리 결정된 이터레이션(iteration) 횟수만큼 반복하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다.

[0133] 프로세서(200)는 제2 활성화 기울기에 기초하여 제1 임계값을 초기화함으로써 제2 임계값을 계산할 수 있다.

[0134] 프로세서(200)는 타겟 희소성(target sparsity) 및 현재 이터레이션(iteration)에 대응하는 희소성에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산할 수 있다. 프로세서(200)는 타겟 희소성을 상기 현재 이터레이션에 대응하는 희소성으로 나눈 값을 제1 임계값에 곱함으로써 제2 임계값을 계산할 수 있다.

[0135] 프로세서(200)는 제2 임계값이 미리 설정된 제한 범위를 초과하는지 여부를 판단할 수 있다. 프로세서(200)는 제한 범위를 초과하는 제2 임계값을 제한 범위 내의 값으로 보정할 수 있다.

[0136] 프로세서(200)는 제2 활성화 기울기 및 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행할 수 있다(690). 프로세서(200)는 제2 임계값에 기초하여 제2 활성화 기울기를 희소화함으로써 희소 데이터(sparse data)를 생성할 수 있다.

[0137] 프로세서(200)는 희소 데이터 및 밀집 데이터(dense data)를 이용하여 뉴럴 네트워크 연산을 수행할 수 있다. 밀집 데이터는 병렬화된(parallelized) 복수의 밀집 버퍼(dense buffer)에 저장될 수 있다.

[0138] 프로세서(200)는 제2 활성화 기울기에 기초하여 적어도 한 번의 MAC(Multiply Accumulate) 연산을 수행할 수 있다.

[0139] 제1 활성화 기울기 및 상기 제2 활성화 기울기는 입력 활성화에 대한 기울기, 가중치에 대한 기울기 또는 출력 활성화에 대한 기울기를 포함할 수 있다.

[0141] 이상에서 설명된 실시예들은 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치, 방법 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0142] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(signal wave)에 영구적으로, 또는 일시적으로 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서,

분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

[0143] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있으며 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

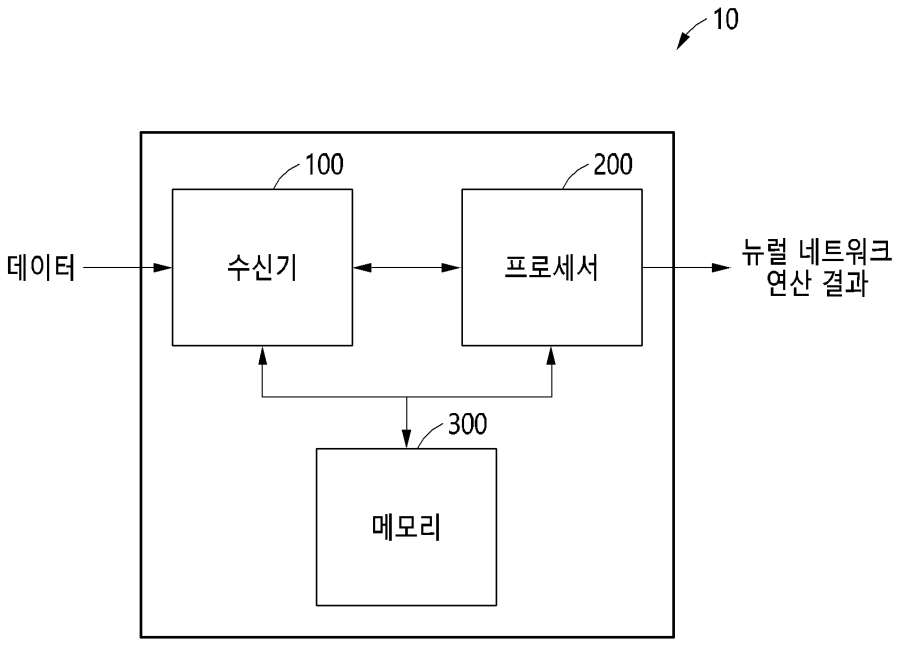
[0144] 위에서 설명한 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 또는 복수의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

[0145] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 이를 기초로 다양한 기술적 수정 및 변형을 적용할 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

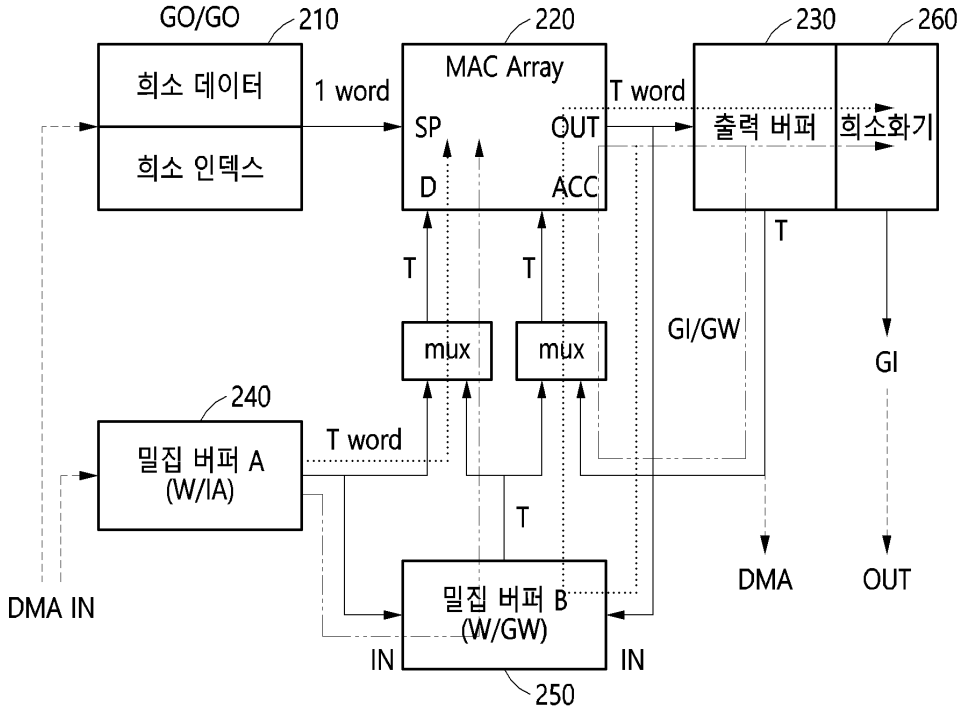
[0146] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

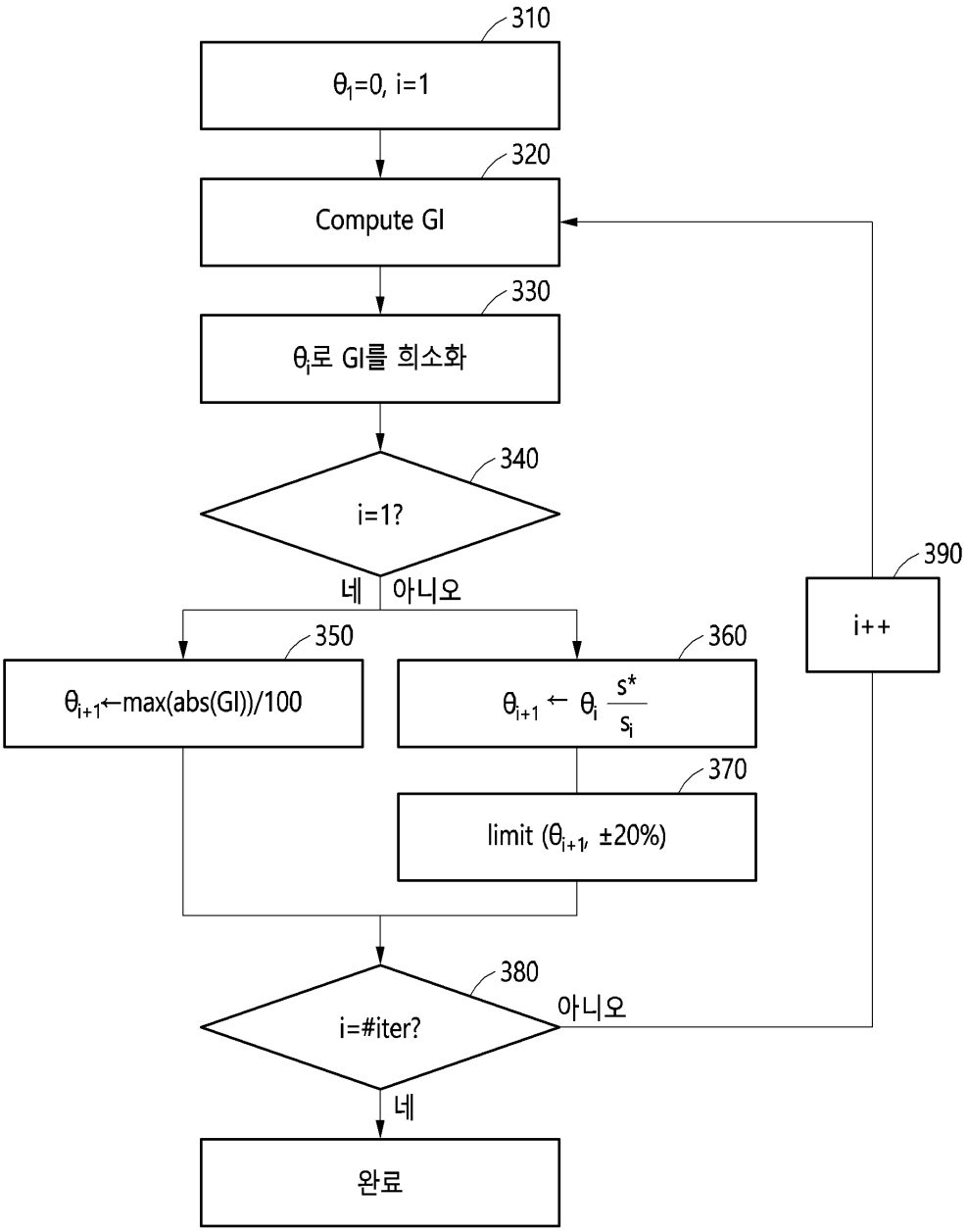
도면1



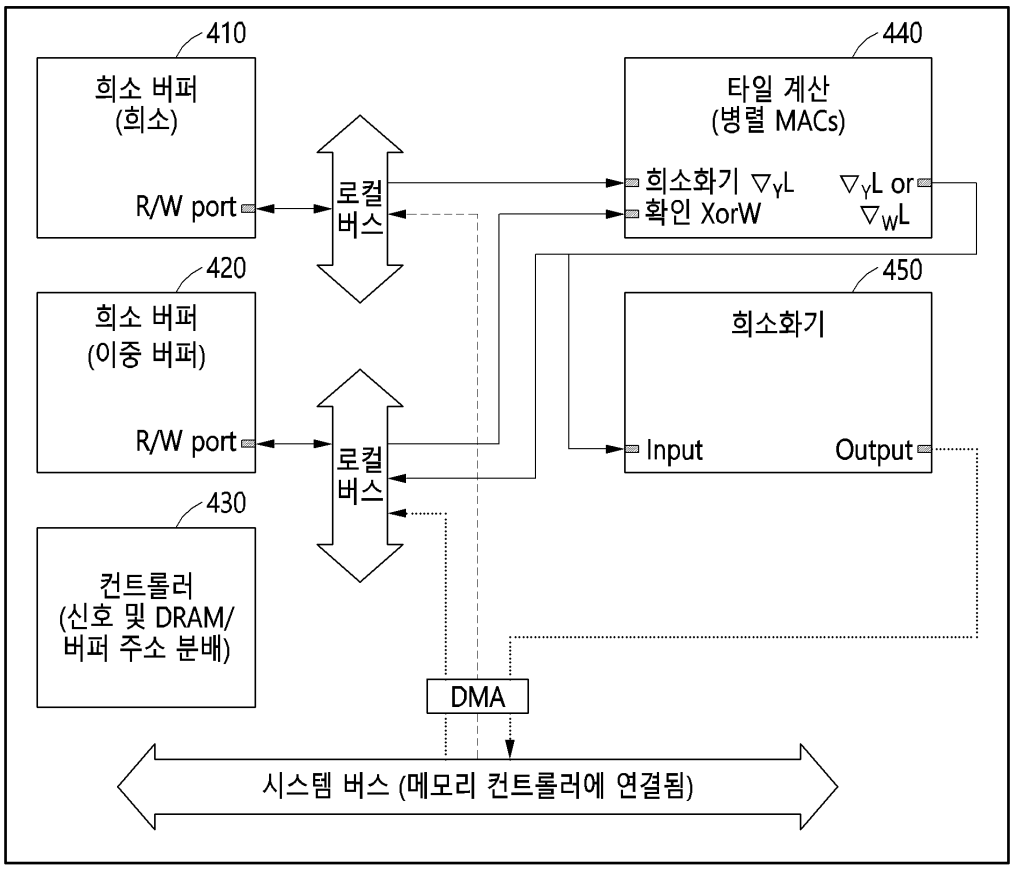
도면2


GO/GO
210
220
230
260
희소 데이터
희소 인덱스
1 word
MAC Array
SP
D
OUT
ACC
T word
출력 버퍼
희소화기
T
T
T
mux
mux
GI/GW
GI
240
밀집 버퍼 A
(W/IA)
T word
T
DMA IN
밀집 버퍼 B
(W/GW)
DMA
OUT
IN
IN
250

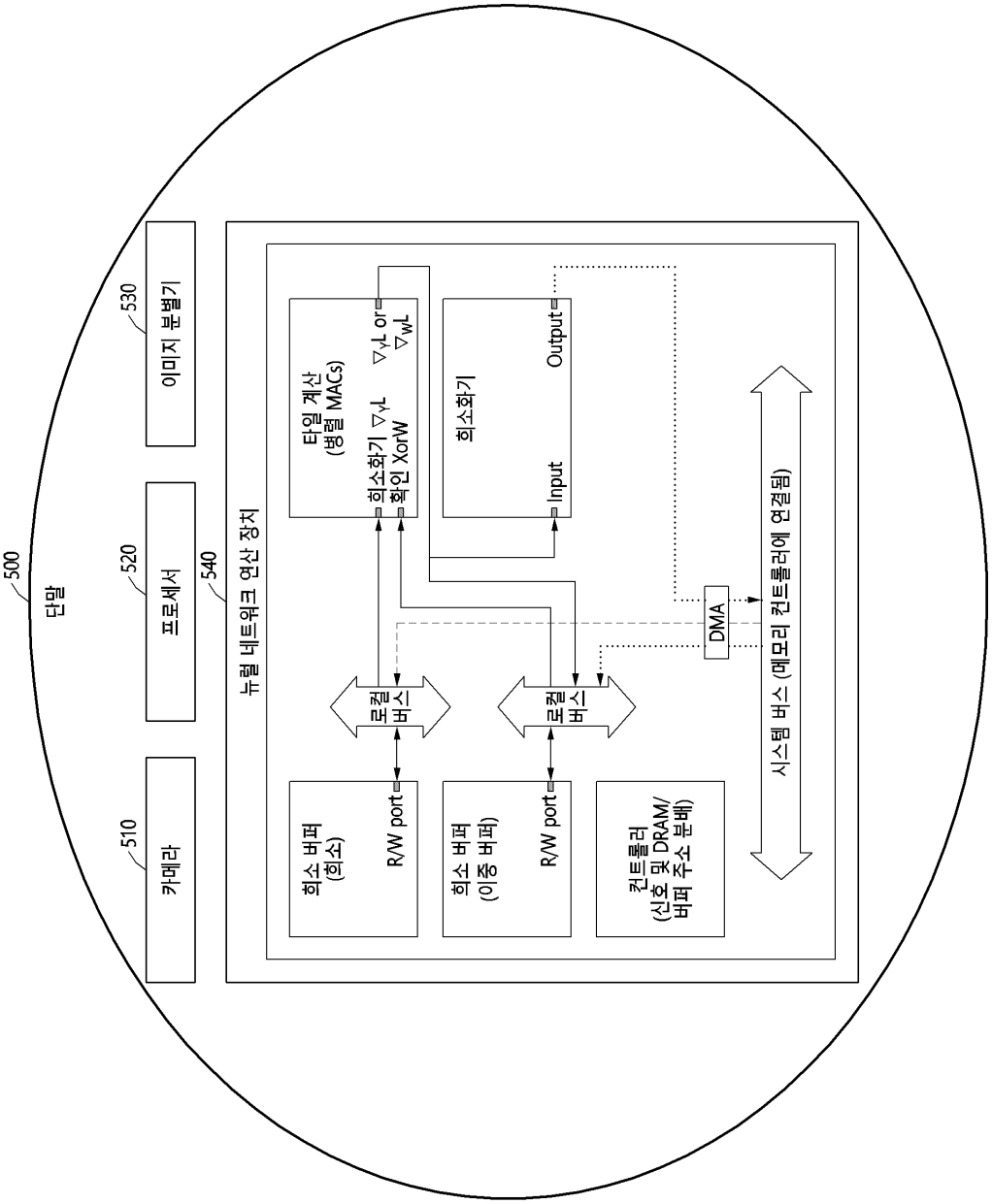
도면3


310
θ1=0, i=1
320
Compute GI
330
θi로 GI를 희소화
340
i=1?
네
아니오
350
θi+1←max(abs(GI))/100
360
θi+1 ← θi s*/si
370
limit (θi+1, ±20%)
380
i=#iter?
아니오
네
완료
390
i++

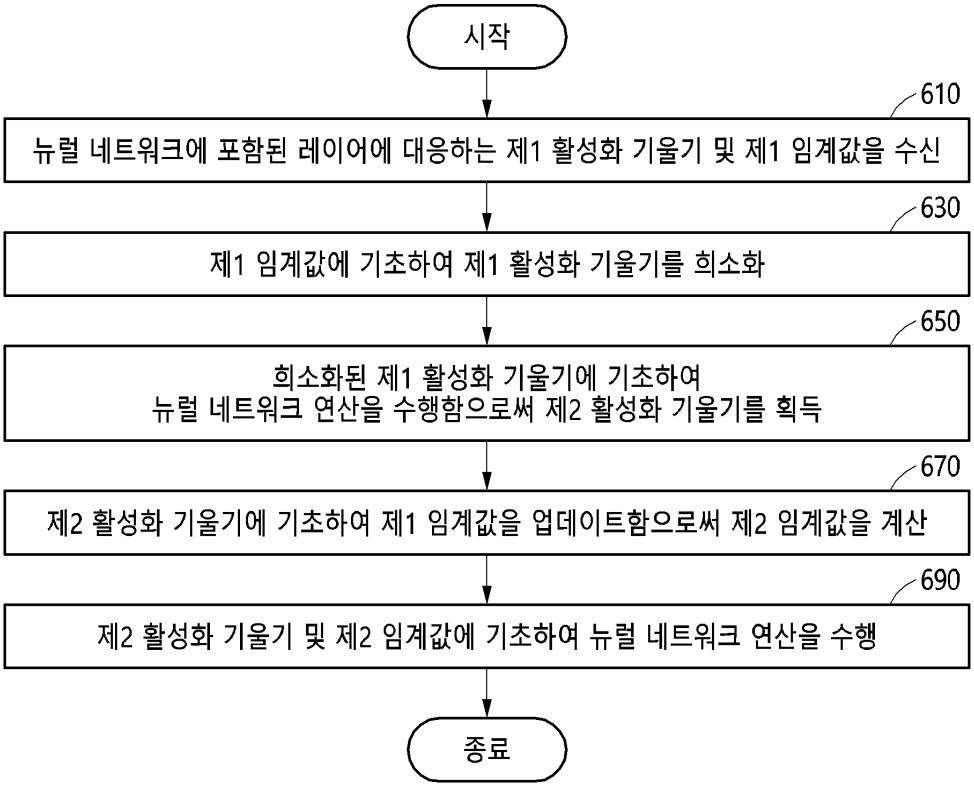
도면4


410
희소 버퍼
(희소)
R/W port
420
희소 버퍼
(이중 버퍼)
R/W port
430
컨트롤러
(신호 및 DRAM/
버퍼 주소 분배)
로컬
버스
로컬
버스
440
타일 계산
(병렬 MACs)
희소화기 ∇YL
확인 XorW
∇YL or
∇WL
450
희소화기
Input
Output
DMA
시스템 버스 (메모리 컨트롤러에 연결됨)

도면5


500
단말
510
카메라
520
프로세서
530
이미지 분별기
540
뉴럴 네트워크 연산 장치
타일 계산
(병렬 MACs)
희소화기 ▽yL
확인 XorW
▽yL or
▽wL
희소화기
Input
Output
로컬
버스
로컬
버스
DMA
희소 버퍼
(희소)
R/W port
희소 버퍼
(이중 버퍼)
R/W port
컨트롤러
(신호 및 DRAM/
버퍼 주소 분배)
시스템 버스 (메모리 컨트롤러에 연결됨)

도면6


시작
610
뉴럴 네트워크에 포함된 레이어에 대응하는 제1 활성화 기울기 및 제1 임계값을 수신
630
제1 임계값에 기초하여 제1 활성화 기울기를 희소화
650
희소화된 제1 활성화 기울기에 기초하여
뉴럴 네트워크 연산을 수행함으로써 제2 활성화 기울기를 획득
670
제2 활성화 기울기에 기초하여 제1 임계값을 업데이트함으로써 제2 임계값을 계산
690
제2 활성화 기울기 및 제2 임계값에 기초하여 뉴럴 네트워크 연산을 수행
종료