(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2024년07월01일
(11) 등록번호 10-2679563
(24) 등록일자 2024년06월25일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2023.01) *G06F 13/16* (2006.01)
G06F 17/15 (2006.01) *G06F 17/16* (2006.01)
G06F 7/544 (2017.01) *G06N 3/04* (2023.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06F 13/1668 (2013.01)
(21) 출원번호 10-2023-7014253(분할)
(22) 출원일자(국제) 2017년08월22일
심사청구일자 2023년04월26일
(85) 번역문제출일자 2023년04월26일
(65) 공개번호 10-2023-0061577
(43) 공개일자 2023년05월08일
(62) 원출원 특허 10-2022-7015590
원출원일자(국제) 2017년08월22일
심사청구일자 2022년05월09일
(86) 국제출원번호 PCT/US2017/047992
(87) 국제공개번호 WO 2018/080624
국제공개일자 2018년05월03일
(30) 우선권주장
15/336,066 2016년10월27일 미국(US)
15/465,774 2017년03월22일 미국(US)

(56) 선행기술조사문헌
Bert Moons et al., A 0.3-2.6 TOPS/W Precision-Scalable Processor for Real-Time Large-Scale ConvNets, 2016 IEEE Symposium on VLSI Circuits*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
구글 엘엘씨
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
(72) 발명자
우동혁
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
나라야나스와미 라비
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
(74) 대리인
박장원

전체 청구항 수 : 총 11 항 심사관 : 안지현

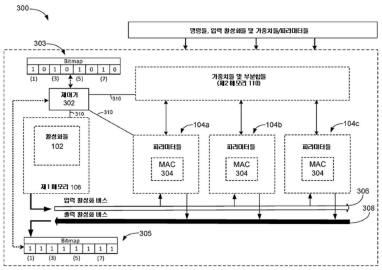
(54) 발명의 명칭 **신경망 계산 유닛에서 입력 데이터의 희소성 이용**

(57) 요 약

컴퓨터 구현 방법은 컴퓨팅 디바이스에 의해 입력 활성화들을 수신하는 단계와 상기 컴퓨팅 디바이스의 제어기에 의해 입력 활성화들 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계를 포함한다. 상기 방법은 컴퓨팅 디바이스의 메모리 뱅크에 입력 활성화들 중 적어도 하나를 저장하는 단계를 더 포함한다. 적어도

(뒷면에 계속)

대 표 도



하나의 입력 활성화를 저장하는 단계는 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계를 포함한다. 상기 방법은 제어기에 의해 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 더 포함한다. 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

(52) CPC특허분류

G06F 17/153 (2013.01)
G06F 17/16 (2013.01)
G06F 7/5443 (2013.01)
G06N 3/04 (2023.01)
Y02D 10/00 (2020.08)

명 세 서

청구범위

청구항 1

복수의 신경망 계층을 포함하는 신경망을 구현하도록 구성된 하드웨어 집적 회로에 있어서, 상기 집적 회로는,

제1 신경망 계층의 출력을 생성하도록 구성된 제1 계산 유닛을 포함하고, 상기 제1 계산 유닛은,

출력을 생성하기 위해 제1 입력 묶음(batch)에서 0이 아닌 입력(non-zero)을 대응 가중치와 곱셈하도록 구성된 제1 복수의 곱셈 누적 셀과;

복수의 곱셈 누적 셀 각각에 연결된 제1 입력 버스와; 그리고

제1 입력 버스에 연결된 제1 입력 메모리와, 상기 제1 입력 메모리는 제1 입력 묶음의 0이 아닌 입력만 저장하는데 사용되고;

제1 입력 버스가 제1 복수의 곱셈 누적 셀 각각에 제1 입력 묶음의 각각의 0이 아닌 입력을 제공하게 하는 제어 신호를 생성하도록 구성된 제1 컨트롤러를 포함하고, 상기 제어 신호는 제1 입력 묶음의 0이 아닌 입력을 저장하는 제1 입력 메모리의 메모리 위치만을 어드레싱하기 위한 것임을 특징으로 하는 하드웨어 집적 회로.

청구항 2

제1항에 있어서,

제1 입력 묶음에 상대적인 상이한 입력 묶음에서 0이 아닌 입력을 처리하는 것에 응답하여 상이한 제2 신경망 계층의 출력을 생성하도록 구성된 제2 계산 유닛을 더 포함하는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 3

제1항에 있어서,

제2 계산 유닛을 더 포함하고, 상기 제2 계산 유닛은,

제2 복수의 곱셈 누적 셀과;

제2 복수의 곱셈 누적 셀 각각에 연결된 제2 입력 버스와; 그리고

제2 입력 버스가 제2 복수의 곱셈 누적 셀 각각에 제1 입력 묶음의 각각의 0이 아닌 입력을 제공하게 하는 제어 신호를 생성하도록 구성된 제2 컨트롤러를 포함하는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 4

제3항에 있어서,

상기 제2 계산 유닛은,

제2 복수의 곱셈 누적 셀을 사용하여 제2 묶음의 입력에서 0이 아닌 입력을 처리하는 것에 응답하여 제2 신경망 계층의 출력을 생성하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 5

제1항에 있어서,

상기 제1 계산 유닛은,

제1 신경망 계층을 통해 처리될 제1 입력 묶음 및 제1 신경망 계층에 대한 대응하는 제1 가중치 세트를 수신하고; 그리고

i) 제1 가중치 세트의 대응하는 가중치 및 ii) 제1 입력 버스로부터 0이 아닌 입력을 수신하는 대응 곱셈 누적

셀을 사용하여 제1 입력 묶음에서 각각의 0이 아닌 입력을 처리하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 6

제5항에 있어서,

상기 제1 계산 유닛은,

제1 복수의 곱셈 누적 셀의 각각의 곱셈 누적 셀에 연결된 제1 가중치 메모리를 더 포함하고, 상기 제1 가중치 메모리는 제1 가중치 세트를 저장하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 7

제6항에 있어서,

상기 제1 복수의 곱셈 누적 셀의 각각의 곱셈 누적 셀은,

제1 입력 버스를 통해 제1 입력 메모리로부터 각각의 0이 아닌 입력을 수신하고;

제1 신경망 계층에 대해 제1 가중치 메모리로부터 대응 가중치를 수신하고; 그리고

제1 신경망 계층에 대한 출력을 생성하기 위해 각각의 0이 아닌 입력과 대응 가중치 간에 곱셈을 수행하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 8

제3항에 있어서,

상기 제2 계산 유닛은,

제2 입력 버스에 연결된 제2 입력 메모리와, 상기 제2 입력 메모리는 제2 입력 묶음에 0이 아닌 입력만을 저장하도록 구성되고; 그리고

제2 복수의 곱셈 누적 셀의 각각의 곱셈 누적 셀에 연결된 제2 가중치 메모리를 더 포함하고, 상기 제2 가중치 메모리는 제2 가중치 세트를 저장하도록 구성되는 것을 특징으로 하는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 9

제8항에 있어서,

상기 제2 복수의 곱셈 누적 셀의 각각의 곱셈 누적 셀은,

제2 입력 버스를 통해 제2 입력 메모리로부터 각각의 0이 아닌 입력을 수신하고;

제2 신경망 계층에 대해 제2 가중치 메모리로부터 대응 가중치를 수신하고; 그리고

제2 신경망 계층에 대한 출력을 생성하기 위해 각각의 0이 아닌 입력과 대응 가중치 간에 곱셈을 수행하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

청구항 10

복수의 신경망 계층을 포함하는 신경망을 구현하는 집적 회로를 사용하여 수행되는 방법으로서, 상기 방법은,

제어신호를 사용하여, 메모리 뱅크의 복수의 활성화 중에서 복수의 신경망 계의 신경망 계층에 의해 처리하기 위한 각각의 0이 아닌 활성화 값을 식별하는 단계와, 상기 제어 신호는 0이 아닌 입력 활성화 값을 저장하는 메모리 뱅크의 메모리 어드레스 위치만을 어드레싱하기 위한 것이고;

각각의 0이 아닌 활성화 값을 집적 회로의 입력 데이터 버스에 순차적으로 제공하는 단계와;

집적 회로의 복수의 MAC(multiply accumulate cells) 각각에 입력 데이터 버스로부터의 각각의 0이 아닌 활성화 값을 제공하는 단계와; 그리고

각각의 0이 아닌 활성화 값을 사용하여 복수의 MAC 각각에 의해 수행되는 행렬 곱셈에 기초하여 신경망 계층의 출력을 생성하는 단계를 포함하는 것을 특징으로 하는 집적 회로를 사용하여 수행되는 방법.

청구항 11

제10항에 있어서,

상기 복수의 활성화 중에서 각각의 0이 아닌 활성화 값을 식별하는 단계는,

집적 회로의 컴퓨팅 타일에서 입력 활성화의 데이터 스트림을 수신하는 단계; 및

입력 활성화의 데이터 스트림에서 각 입력 활성화의 분석에 기초하여 각각의 0이 아닌 활성화 값을 식별하는 단계를 포함하는 것을 특징으로 하는 집적 회로를 사용하여 수행되는 방법.

발명의 설명

기 술 분 야

[0001] 본 명세서는 특수 목적 계산 유닛을 사용하여 기계 학습 계산을 수행하는 것에 관한 것이다.

배 경 기 술

[0002] 신경망은 모델의 하나 이상의 계층을 사용하여 수신 입력에 대한 출력, 예를 들어 분류를 생성하는 기계 학습 모델이다. 일부 신경망은 출력 계층 이외에 하나 이상의 히든 계층를 포함한다. 각 히든 계층의 출력은 네트워크의 다음 계층, 즉 다음 히든 계층 또는 네트워크의 출력 계층에 대한 입력으로 사용된다. 네트워크의 각 계층은 각각의 파라미터 세트의 현재 값에 따라 수신 입력으로부터 출력을 생성한다.

[0003] 일부 신경망은 하나 이상의 컨볼루션 신경망 계층을 포함한다. 각 컨볼루션 신경망 계층은 관련 커널 세트를 가진다. 각 커널은 사용자가 만든 신경망 모델에 의해 설정된 값을 포함한다. 일부 구현예에서, 커널은 특정 이미지 윤곽, 모양 또는 색상을 식별한다. 커널은 가중치 입력의 행렬 구조로 나타낼 수 있다. 각 컨볼루션 계층은 또한 활성화 입력 세트를 처리할 수 있다. 활성화 입력 세트는 또한 행렬 구조로 표현될 수 있다.

발명의 내용

[0004] 컨볼루션 계산을 연산하는 한 가지 방법은 큰 차원 공간에서 다수의 행렬 곱셈을 필요로 한다. 계산 유닛의 프로세서 또는 제어기 디바이스는 무차별적 방법을 통해 행렬 곱셈을 계산할 수 있다. 예를 들어, 계산 집약적이고 시간 집약적이지만 프로세서는 컨볼루션 계산을 위해 개별 합산 및 곱(product)을 반복적으로 계산할 수 있다. 프로세서가 계산을 병렬화하는 정도는 그의 아키텍처 때문에 제한된다.

[0005] 본 명세서에 기술된 주제의 혁신적인 양태는 컴퓨터 구현 방법으로 구현될 수 있다. 상기 방법은 컴퓨팅 디바이스에 의해, 복수의 입력 활성화를 수신하는 단계와, 상기 입력 활성화는 상기 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되고, 상기 컴퓨팅 디바이스의 제어기에 의해, 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나인지 여부를 결정하는 단계를 포함한다. 상기 방법은 상기 컴퓨팅 디바이스의 메모리 뱅크에, 적어도 하나의 입력 활성화를 저장하는 단계를 더 포함하고, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계는 상기 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계를 포함한다. 상기 방법은 또한 상기 제어기에 의해 상기 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 더 포함하고, 상기 활성화들은 적어도 부분적으로 상기 인덱스와 관련된 메모리 어드레스 위치로부터 제공된다.

[0006] 일부 구현예에서, 상기 인덱스는 복수의 비트를 포함하는 비트 맵에 기초하여 생성되고, 상기 비트 맵의 각 비트는 비-제로 입력 활성화 값 또는 제로 입력 활성화 값 중 적어도 하나를 나타낸다. 일부 구현예에서, 상기 방법은 적어도 하나의 유닛에 의해, 상기 비-제로 값을 사용하는 계산을 수행하기 위해 비-제로 값을 갖는 제1 입력 활성화를 제공하고 후속하여 제로 값을 갖는 제2 입력 활성화를 제공하는 단계와, 그리고 적어도 하나의 유닛에서, 상기 제로 값을 사용하여 수행될 계산을 방지하는 단계를 더 포함한다.

[0007] 일부 구현예에서, 상기 방지하는 단계는 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는 것으로 결정하는 제어기에 응답하여 발생한다. 일부 구현예에서, 상기 방법은 제어기에 의해, 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는지를 검출하는 단계와, 상기 검출에 응답하여, 상기 제로 입력 활성화 값과 관련된 곱셈 연산을 방지하기 위해 계산 어레이의 적어도 하나의 유

닛에 제어 신호를 제공하는 단계를 더 포함한다.

[0008] 일부 구현예에서, 상기 방법은 제어기에 의해 제1 유닛에, 제1 입력 활성화를 사용하는 텐서 계산의 제1 부분을 매핑하고, 상기 제1 유닛과 다른 제2 유닛에, 상기 제1 입력 활성화를 또한 사용하는 상기 텐서 계산의 제2 부분을 매핑하는 단계를 더 포함한다. 일부 구현예에서, 상기 방법은 데이터 버스 상에 단일 입력 활성화를 순차적으로 제공하는 단계를 더 포함하고, 상기 단일 입력 활성화는 인덱스와 관련된 메모리 어드레스 위치들로부터 액세스되고 선택된다. 일부 구현예에서, 상기 제공 단계는 제로 값을 갖는 입력 활성화들을 제공하지 않는 단계를 더 포함한다.

[0009] 본 명세서에 기술된 주제의 다른 혁신적인 양태는, 동작들을 수행하도록 하나 이상의 처리 디바이스에 의해 실행 가능한 명령들을 저장하는 하나 이상의 기계-판독 가능 저장 디바이스에서 구현될 수 있으며, 상기 동작들은 컴퓨팅 디바이스에 의해, 상기 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되는 복수의 입력 활성화를 수신하는 단계와; 상기 컴퓨팅 디바이스의 제어기에 의해, 상기 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계를 포함한다. 상기 동작들은 컴퓨팅 디바이스의 메모리 뱅크에, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계를 포함하고, 상기 적어도 하나의 입력 활성화를 저장하는 단계는 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계를 포함한다. 상기 동작들은 제어기에 의해 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 더 포함하고, 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

[0010] 본 명세서에 기술된 주제의 또 다른 혁신적인 양태는 컴퓨팅 디바이스에 배치되어 하나 이상의 처리 디바이스를 포함하는 제어기와; 동작들을 실행하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능한 명령들을 저장하는 하나 이상의 기계-판독 가능 저장 디바이스를 포함하는 전자 시스템에서 구현될 수 있으며, 상기 동작들은 컴퓨팅 디바이스에 의해, 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되는 복수의 입력 활성화를 수신하는 단계와; 상기 컴퓨팅 디바이스의 제어기에 의해, 상기 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계를 포함한다. 상기 동작들은 상기 컴퓨팅 디바이스의 메모리 뱅크에, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계를 포함하고, 상기 적어도 하나의 입력 활성화를 저장하는 단계는 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계를 포함한다. 상기 동작들은 제어기에 의해 상기 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 더 포함하고, 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

[0011] 본 명세서에 기술된 주제는 다음의 이점들 중 하나 이상을 실현하도록 특정 실시예들에서 구현될 수 있다. 단일 컴퓨팅(compute) 시스템에서, 제1 메모리에서 액세스 가능한 활성화들 및 제2 메모리에서 액세스 가능한 가중치들은 레지스터들에서 검색된 메모리 어드레스 값들에 기초하여 트래버스할 수 있다. 컴퓨팅 시스템의 제어기는 비-제로 값들만을 제1 메모리에 저장하여 활성화 데이터를 압축할 수 있으므로 메모리 저장 공간과 해당 대역폭을 절약할 수 있다. 행렬 곱셈은 주로 비-제로 입력 활성화들을 제공하는 것에 부분적으로 기초하여 컴퓨팅 시스템에서 발생한다. 또한, 컴퓨팅 시스템이 주로 비-제로 활성화 값들을 포함하는 통신 기법을 사용할 때, 곱셈을 제로로 제거함으로써 계산 효율을 향상시키거나 가속화할 수 있다.

[0012] 이 양태 및 다른 양태의 다른 구현예들은 대응하는 시스템, 장치 및 컴퓨터 저장 디바이스 상에 인코딩된 상기 방법들의 동작들을 수행하도록 구성된 컴퓨터 프로그램을 포함한다. 하나 이상의 컴퓨터로 구성된 시스템은 소프트웨어, 펌웨어, 하드웨어 또는 이들이 조합된 시스템상에 설치되어, 동작시 시스템이 동작들을 수행하도록 구성될 수있다. 하나 이상의 컴퓨터 프로그램은 데이터 처리 장치에 의해 실행될 때 장치로 하여금 동작들을 수행하게 하는 명령들을 갖도록 구성될 수 있다.

[0013] 본 명세서에 기술된 주제는 또한 이미지 인식 및/또는 분류 방법/시스템에 관한 것이다. 시스템(들)은 추론 계산을 수행하도록 하드웨어 컴퓨팅 시스템의 컴퓨팅 유닛들이 신경망 계층에 대한 입력들을 처리할 때 입력 데이터 희소성을 이용하기 위해 개시된 기술을 사용하여 구현될 수 있다.

[0014] 본 명세서에서 설명된 주제의 하나 이상의 구현예의 세부 사항은 첨부된 도면 및 이하의 설명에서 설명된다. 주제의 다른 잠재적인 특징, 양상 및 장점은 상세한 설명, 도면 및 청구 범위로부터 명백해질 것이다.

도면의 간단한 설명

[0015] 도 1은 활성화 및 파라미터를 포함하는 예시적인 계산 구조를 도시한다.

도 2는 1보다 큰 출력 깊이에 대한 활성화 및 다중 파라미터 구조를 포함하는 예시적인 계산 구조를 도시한다.

도 3은 하나 이상의 파라미터에 입력 활성화를 공급하기 위한 예시적인 컴퓨팅 시스템을 도시한다.

도 4는 하나 이상의 곱셈 누적(MAC) 연산기들에 입력 활성화를 제공하는 메모리 유닛을 포함하는 예시적인 아키텍처를 도시한다.

도 5는 파라미터 계산을 감소시키고 입력 데이터 희소성을 이용하기 위한 프로세스의 예시적인 흐름도이다.

다양한 도면에서 유사한 참조 번호 및 명칭은 동일한 요소를 나타낸다.

발명을 실시하기 위한 구체적인 내용

[0016] 본 명세서에 기술된 주제는 예시적인 신경망 하드웨어 컴퓨팅 시스템의 계산 유닛 또는 타일(tile) 내에서 일어나는 계산을 감소시키는 것에 관한 것이다. 일반적으로, 신경망 추론의 일부로서, 입력 활성화(input activation)는 출력 활성화를 생성하도록 파라미터 또는 가중값과 승산된다. 여기서, 입력들 및 입력 활성화는 신경망에서 일반적으로 사용되는 텐서(tensor), 행렬 및/또는 데이터 어레이와 같은 다차원 데이터 구조에 포함된 데이터 엘리먼트를 지칭할 수 있다. 심층 신경망에 대한 컴퓨팅 추론의 알고리즘 특성들로 인해 입력 활성화의 대부분은 0이다. 환언하면, 현재의 계산 유닛은 제로(입력 활성화 값)에 하나의 수(예를 들어, 가중치)를 곱하는 것을 포함하는 많은 수의 불필요한 계산을 수행한다.

[0017] 본 명세서는 부분적으로, 특히 신경망의 컨볼루션 계층들을 처리하기 위한, 보다 효율적인 활성화 저장 및 통신 기법뿐만 아니라 심층 신경망 처리를 위한 커스텀 아키텍처 설계를 기술한다. 시간 경과에 따라 고밀도 행렬 곱셈을 수행하는 기존의 하드웨어 가속기들과는 달리, 본 명세서는 1) 제로 입력 값을 보았을 때 계산을 스킵하거나 바이패스하고, 2) 비-제로 값들만 포함하는 압축된 입력 활성화들을 저장함으로써 계산 유닛의 메모리 사용을 줄일 수 있는 아키텍처를 기술한다. 전반적으로, 본 명세서의 교시를 통해 신경망 추론 계산을 위한 계산 유닛 성능이 개선되고 불필요한 계산을 스킵함으로써 에너지 절감이 실현된다.

[0018] 도 1은 활성화 구조(102) 및 파라미터 구조(104)를 포함하는 예시적인 계산 구조(100)를 도시한다. 활성화 구조(102)는 (아래 첨자 0로 표시된) 제1 입력 깊이에 대응하는 다수의 데이터 요소(element)를 포함하는 제1 데이터 구조(102a)를 포함할 수 있다. 유사하게, 활성화 구조(102)는 (아래 첨자 1로 표시된) 제2 입력 깊이에 대응하는 다수의 데이터 요소를 포함하는 제2 데이터 구조(102b)를 또한 포함할 수 있다. 데이터 구조(102a 및 102b)에 도시된 다수의 데이터 요소는 각각 a_0, b_0, c_0, d_0 및 a_1, b_1, c_1, d_1으로 표시된다. 데이터 구조(102a/b)의 각 데이터 요소(a_0, a_1, b_0, b_1, c_0, d_0 등)는 입력 활성화 값이고, 각 입력 깊이는 신경망 계층에 대한 입력의 깊이(depth)에 대응한다. 일부 구현예에서, 신경망 계층은 하나의 입력 깊이를 가질 수 있는 반면, 다른 구현예에서 신경망 계층은 하나 이상의 입력 깊이를 가질 수 있다.

[0019] 파라미터 구조(104)는 활성화 구조(102)와 유사한 방식으로 기술될 수 있다. 파라미터 구조(104)는 제1 데이터 구조(104a) 및 제2 데이터 구조(104b)를 포함한다. 각 데이터 구조(104a/b)는 각각 커널 값을 포함하는 다수의 데이터 요소를 포함할 수 있다. 도 1에 도시된 바와 같이, 데이터 구조(104a 및 104b)에 대응하는 다수의 데이터 요소는 각각 x_0, y_0, z_0 및 x_1, y_1, z_1으로 표시된다.

[0020] 전술한 바와 같이, 신경망의 각 계층은 각각의 오퍼랜드 세트의 값에 따라 수신 입력으로부터 출력을 생성한다. 다른 신경망 계층들과 마찬가지로 각 컨볼루션 계층은 행렬 구조로 표현될 수 있는 활성화 입력들의 세트를 처리할 수 있다. 종래의 컨볼루션 신경망 계층은 또한 값들을 포함하는 관련 커널 세트를 가질 것이고, 커널들은 또한 가중치의 행렬 구조로 표현될 수 있다. 도 1에서, 활성화 구조(102)는 하나 이상의 활성화 입력을 갖는 행렬 구조에 대응할 수 있고, 파라미터 구조(104)는 하나 이상의 커널 또는 가중치 파라미터를 갖는 행렬 구조에 대응할 수 있다.

[0021] 아래에서 보다 상세하게 기술되는 바와 같이, 신경망의 다양한 계층은 행렬 곱셈을 포함하는 많은 양의 계산을 수행함으로써 기계 학습 추론을 처리한다. 신경망 계층(예컨대, 컨볼루션 계층) 내에서 수행되는 계산 프로세스들은 하나 이상의사이클서 입력 활성화(즉, 제1 오퍼랜드)와 가중치(즉, 제2 오퍼랜드)를 곱하는 단계 및 많은 사이클에 걸쳐 곱들의 누적을 수행하는 단계를 포함할 수 있다. 출력 활성화는 두 오퍼랜드에 대해 수행된 곱셈 및 누적 연산을 기반으로 생성된다.

[0022] 도시된 바와 같이, 방정식(106)은 활성화 구조(102)의 소정 데이터 요소와 관련된 입력 활성화가 파라미터 구조(104)의 소정 데이터 요소와 관련된 커널 값 및/또는 가중치/파라미터와 곱해질 때 수행될 수 있는 예시적인 일련의 시퀀스 기반의 수학적 연산을 제공한다. 예를 들어, 방정식(106)에서, 인덱스 "i"가 0일 때, 활성화 구조(102)의 데이터 요소(a_0)와 연관된 입력 활성화는 파라미터 구조(104)의 데이터 요소(x_0)과 관련된 가중치/파라미터와 곱해진다. 또한, 방정식(106)은 부분적으로는 직렬 기반 방정식이기 때문에, 활성화 구조(102) 및 파라미터 구조(104)의 다른 데이터 요소들에 대응하는 오버핸드 세트들간에 추가적인 곱셈 동작들이 발생할 것이다. 일부 구현예에서, 오퍼랜드 세트의 곱셈은 특정 출력 특성 (feature) 또는 활성화를 위한 부분합(106a/b)를 생성할 수 있다. 따라서, 방정식(106)에 도시된 바와 같이, 출력 특성을 생성하기 위해 부분합을 더할 수 있다.

[0023] 신경망은 네트워크 계층의 기계 학습 추론 작업 부하를 가속시키도록 구성된 다수의 컴퓨팅 유닛을 포함하는 하나 이상의 하드웨어 컴퓨팅 시스템에서 구현될 수 있다. 각 컴퓨팅 유닛은 주어진 계층에 대한 계산의 서브 세트를 처리할 수 있다. 일부 구현예에서, 구조(100)는 적어도 2개의 메모리 뱅크 및 집합적으로 곱셈 누적(multiply accumulate: MAC) 어레이(이하에서 설명됨)를 형성할 수 있는 하나 이상의 MAC 셀을 각각 포함하는 하나 이상의 컴퓨팅 유닛에서 구현될 수 있다.

[0024] 일 예에서, 예시적인 컴퓨팅 유닛의 제1 메모리 뱅크(108)는 활성화 구조(102)와 관련된 데이터를 저장하고, 메모리 뱅크(108)내의 메모리 어드레스 위치에 대한 입력 활성화 값을 수신 및 기록하도록 구성될 수 있다. 마찬가지로, 예시적인 컴퓨팅 유닛의 제2 메모리 뱅크(110)는 파라미터 구조(104)와 관련된 데이터를 저장하고, 메모리 뱅크(110) 내의 메모리 어드레스 위치에 대한 가중치 값를 수신 및 기록하도록 구성될 수 있다. 이 예에서, 데이터 요소(102a)의 각 요소(예컨대, a_0, b_0, c_0, d_0)은 제1 메모리 뱅크(108)의 각각의 메모리 어드레스에 저장될 수 있다. 마찬가지로, 데이터 요소(104a)의 각 요소(예를 들어, x_0, y_0, z_0)는 제2 메모리(110)의 각각의 메모리 어드레스에 저장될 수 있다.

[0025] 일부 구현예에서, 제1 메모리 뱅크(108) 및 제2 메모리 뱅크(110)는 각각 휘발성 메모리 유닛(들)이다. 일부 다른 구현예에서, 메모리 뱅크(108) 및 메모리 뱅크(110)는 각각 비-휘발성 메모리 유닛(들)이다. 메모리 뱅크(108, 110)는 또한 저장 영역 네트워크 또는 다른 구성의 디바이스들을 포함하여, 플로피 디스크 디바이스, 하드 디스크 디바이스, 광학 디스크 디바이스 또는 테이프 디바이스, 플래시 메모리 또는 다른 유사 고체 상태 메모리 디바이스, 또는 디바이스들의 어레이와 같은 컴퓨터 판독 가능 저장 매체의 다른 형태일 수 있다.

[0026] 일반적으로, 하드웨어 컴퓨팅 시스템의 컴퓨팅 유닛은 메모리 어드레스 값을 추적하기 위한 하나 이상의 레지스터를 포함할 수 있다. 활성화 구조(102)에 대응하는 행렬 구조의 데이터 요소들은 제1 메모리 뱅크(108)로부터 액세스될 수 있는 반면, 파라미터 구조(104)에 대응하는 행렬 구조의 데이터 요소들은 제2 메모리 뱅크(110)로부터 액세스될 수 있다. 컴퓨팅 타일/계산 유닛의 예시적인 제어 디바이스는 하나 이상의 레지스터로부터 액세스 가능한 어드레스 값들에 기초하여 행렬 구조들의 데이터 요소를 액세스 및/또는 트래버스(traverse)할 수 있다. 예시적인 제어 디바이스, 활성화 구조(102), 파라미터 구조(104), 제1 메모리 뱅크(108) 및 제2 메모리 뱅크(110)를 포함하는 예시적인 계산 유닛/타일은 도 3을 참조하여 아래에서 보다 상세하게 설명된다.

[0027] 또한, 신경망 추론 작업 부하를 위한 신경망 텐서 계산 및 행렬 곱셈을 가속화하기 위한 하드웨어 컴퓨팅 시스템에 관한 추가 세부 사항 및 설명은 2016년 10월 27일자로 출원된 "신경망 계산 타일(Newural Network Compute Tile)"이라는 제목의 미국 특허 출원 제15/335,769호에 기술되어 있다. 미국 특허 출원 제15/335,769호의 전체 개시 내용은 그 전체가 본 명세서에 참조로 포함된다.

[0028] 도 2는 1보다 큰 출력 특성 깊이에 대한 활성화 구조 및 다중 파라미터 구조를 포함하는 예시적인 계산 구조(200)를 도시한다. 일부 구현예에서, 신경망은 다수의 출력 특성 깊이를 갖는 출력을 생성하는 다수의 계층을 가질 수 있다. 일부 구현예에서, 각 파라미터 구조는 출력 깊이들 중 각각 하나를 담당할 수 있다. 따라서, 계산 구조(200)는 N개의 출력 깊이와 관련된 계산을 용이하게 하기 위해 추가 파라미터 구조(104a/b/c)가 추가되는 스케일러블 컴퓨팅 구조를 나타낸다. N은 변수이고, 컴퓨팅 시스템 설계자의 선호도 또는 요구에 따라, 예를 들어 1~5 또는 1~N의 범위의 정수값을 가질 수 있다.

[0029] 데이터 경로(105)에 의해 도시된 바와 같이, 데이터 구조(102a)와 관련된 요소들에 대한 개별 입력 활성화 값은 각각의 파라미터 구조(104)와 관련된 곱셈 연산기들에 의해 수행되는 계산에 사용하기 위해 각 파라미터 구조(104a/b/c)에 공급될 수 있다. 그 다음, 각각의 파라미터 구조(104)는 그의 좌측 이웃으로부터 수신된 활성와 값을 파이프 라인 방식으로 그의 우측 이웃으로 전달할 수 있다. 대안 적으로, 활성화는 각 파라미터 구조(104)에 의해 동시에 제공 및 소비될 수 있다.

[0030] 행렬(202)은 활성화들(102)에 대응하는 예시적인 행렬 구조를 나타낼 수 있다. 보다 구체적으로, 요소 행(202a)은 데이터 구조(102a)에 대응할 수 있고, 요소 행(202b)은 데이터 구조(102b)에 대응할 수 있다. 일반적으로 그리고 예를 들어, 제1 파라미터 구조(104(1))는 공간(206)과 관련된 계산들을 수행하기 위해 액세스되고, 제2 파라미터 구조(104(2))는 공간(208)과 관련된 계산들을 수행하기 위해 액세스된다. 도시되지는 않았지만, z-차원에 대응하여 추가 계산들이 또한 수행될 수 있다. 예를 들어, 요소 행(202a)은 RGB 이미지의 R 평면에 있을 수 있고, 요소 행(202b)은 동일한 RGB 이미지의 G 평면에 있을 수 있다. 신경망의 예시적인 컨볼루션 계층은 전형적으로 다중 출력 특성을 생성한다. 예시적인 출력 특성은 사과를 분류하기 위한 출력 특성 및 바나나를 분류하기 위한 다른 출력 특성을 포함될 수 있다. 데이터 구조(204)와 관련하여, 공간(들)(206 및 208)은 상이한 분류에 대해상이한 평면을 나타낼 수 있다.

[0031] 도 3은 하나 이상의 파라미터 구조에 입력 활성화들을 공급하는 예시적인 컴퓨팅(compute) 시스템(300)을 도시한다. 컴퓨팅 시스템(300)은 일반적으로 하나 이상의 제어 신호(310)를 제공하여, 활성화 구조(102)에 대한 입력 활성화들이 메모리 뱅크(108)의 메모리 어드레스에 저장되거나 메모리 뱅크(108)의 메모리 어드레스로부터 검색되도록 하는 제어기(302)를 포함한다. 마찬가지로, 제어기(302)는 또한 하나 이상의 제어 신호(310)를 제공하여, 파라미터 구조(104a/b/c)에 대한 가중치가 메모리 뱅크(110)의 메모리 어드레스에 저장되거나 메모리 어드레스로부터 검색되도록한다. 컴퓨팅 시스템(300)은 하나 이상의 MAC 셀/유닛(들)(304), 입력 활성화 버스(306) 및 출력 활성화 버스(308)를 더 포함한다. 제어 신호(310)는 예를 들어 메모리 뱅크(108)가 하나 이상의 입력 활성화를 입력 활성화 버스(306)에 제공하게 하고, 메모리 뱅크(110)가 하나 이상의 가중치를 파라미터 구조(104a/b/c)에 제공하게 하고 및/또는 MAC 유닛(304)이 출력 활성화 버스(308)에 제공되는 출력 활성화들을 생성하는 계산들을 수행하게 한다.

[0032] 제어기(302)는 하나 이상의 처리 유닛(들) 및 메모리를 포함할 수 있다. 일부 실시예에서, 제어기(302)의 처리 유닛(들)은 하나 이상의 프로세서(예컨대, 마이크로 프로세서 또는 중앙 처리 장치(CPU)), 그래픽 처리 장치(GPU), 주문형 집적 회로(ASIC) 또는 다른 프로세서의 조합을 포함할 수 있다. 대안적인 실시예에서, 제어기(302)는 본 명세서에 기술된 하나 이상의 결정 및 계산을 수행하기 위한 추가적인 처리 옵션을 제공하는 다른 저장 또는 컴퓨팅 자원/디바이스(예를 들어, 버퍼, 레지스터, 제어 회로 등)를 포함할 수 있다.

[0033] 일부 구현예에서, 제어기(302)의 처리 유닛(들)은 메모리에 저장된 프로그래밍된 명령들을 실행하여, 제어기(302) 및 컴퓨팅 시스템(300)으로 하여금 본 명세서에 기술된 하나 이상의 기능을 수행하게 한다. 제어기(302)의 메모리는 하나 이상의 비-일시적 기계 판독 가능 저장 매체를 포함할 수 있다. 비-일시적 기계 판독 가능 저장 매체는 고체 상태 메모리, 자기 디스크 및 광 디스크, 휴대용 컴퓨터 디스켓, 랜덤 액세스 메모리(RAM), 판독 전용 메모리(ROM), 소거 가능 프로그래머블 ROM(예컨대, EPROM, EEPROM 또는 플래시 메모리), 또는 정보를 저장할 수 있는 임의의 다른 유형의 매체를 포함할 수 있다.

[0034] 일반적으로, 컴퓨팅 시스템(300)은 예시적인 계산 유닛 또는 타일이고, 텐서, 행렬 및/또는 데이터 어레이와 같은 다차원 데이터 구조와 관련된 계산들을 수행하기 위한 추가적인 하드웨어 구조를 포함할 수 있다. 일부 구현예에서, 입력 활성화 값들은 활성화 구조(102)를 위해 메모리 뱅크(108)에 사전 로딩될 수 있고, 가중치 값들은 신경망 하드웨어 컴퓨팅 시스템과 관련된 외부 또는 상위 레벨 제어 디바이스로부터 컴퓨팅 시스템(300)에 도달하는 컴퓨팅 시스템(300)에 의해 수신된 데이터 값들을 사용하여 제2 메모리 뱅크(110)에 사전 로딩될 수 있다.

[0035] 명령들, 입력들 또는 입력 활성화들 및 가중치들은 외부 입/출력(I/O) 디바이스 또는 신경망 하드웨어 컴퓨팅 시스템과 관련된 상위 레벨 제어 디바이스와 같은 외부 소스로부터 시스템(300)에 제공될 수 있다. 일부 구현예에서, 하나 이상의 데이터 버스는 외부 소스(예컨대, 제어 장치)와 시스템(들)(300)간의 데이터 통신을 제공한다. 데이터 버스들은 예시적인 I/O 디바이스로부터 다수의 시스템(300) 각각으로 또는 신경망용 하드웨어 컴퓨팅 시스템에 포함된 다수의 계산 타일(예컨대, 다수의 시스템(300)) 사이에 명령, 입력 또는 활성화 및 가중치를 제공하는데 사용된다.

[0036] 시스템(300)은 시스템(300)에 의해 수행될 특정 계산 동작들을 정의하는 명령들을 수신할 수 있다. 또한, 제어기(302)는 예를 들어 수신된 입력 활성화와 관련된 데이터 스트림을 분석하기 위해 프로그래밍된 명령들을 실행할 수 있다. 입력 활성화 데이터 스트림을 분석하는 것은 제어기(302)가 각각의 입력 활성화와 관련된 값이 0 값 또는 비-제로 값인지 여부를 검출하거나 결정할 수 있게 한다. 일부 구현예에서, 제어기(302)는 예시적인 입력 활성화 데이터 스트림을 분석하여, 각각의 검출된 제로 활성화 값 및 각각의 검출된 비-제로 활성화 값을 비트 벡터 또는 비트 맵(303)에 매핑한다.

[0037] 도 3에 도시된 바와 같이, 비트 맵(303)은 이진값들을 사용하여, 검출된 제로 값 입력 활성화 및 검출된 비-

제로 값 입력 활성화를 매핑할 수 있다. 예를 들어, "0"의 이진 값은 검출된 제로 입력 활성화 값에 대응할 수 있고, "1"의 이진 값은 검출된 비-제로 입력 활성화 값에 대응할 수 있다. 예를 들어, 비트 맵(303)은 이진수 "1"을 포함하는 홀수 번째 비트 위치가 비-제로 활성화 값에 대응하고, 이진수 "0"을 포함하는 짝수 번째 비트 위치가 제로 활성화 값에 대응하는 8 비트 비트 맵일 수 있다.

[0038] 제어기(302)는 입력 활성화들이 메모리 뱅크(108)에 저장되도록 할 수 있다. 일반적으로, 메모리 뱅크(108)에 저장된 데이터 값들은 전형적으로 각각의 메모리 어드레스 위치에 기록된다. 그 다음 메모리 뱅크(108)의 어드레스 위치는 입력 활성화와 같은 데이터 값이 특정 계산 동작을 수행하기 위해 필요할 때 예시적인 제어 디바이스(예컨대, 제어기(302))에 의해 액세스될 수 있다. 제어기(302)는 비트 맵(303)을 사용하여 비-제로 입력 활성화 값들을 포함하는 메모리 어드레스 위치들의 인덱스를 생성할 수 있다.

[0039] 일부 구현예에서, 제어기(302)는 비트 맵(303)을 사용하여 어느 입력 활성화들이 메모리 뱅크(108)에 기록될지를 결정한다. 예를 들어, 비트 맵(303)의 분석은 비트 맵 위치 1, 3, 5, 7(비-제로 값))에 대응하는 활성화 값들만이 메모리 뱅크(108)의 어드레스 위치에 기록되어야 함을 나타낼 수 있다. 또한, 비트 맵 위치 2, 4, 6, 8(제로 값)와 관련된 데이터 값들은, 활성화 값들이 입력 버스(306)에 제공될 때 제어기(302)에 의해 액세스되거나 액세스될 수 없는 메모리 어드레스 위치로 폐기되거나 기록될 수 있다. 따라서, 비트 맵(303)은 제로 값 입력 활성화들이 메모리 어드레스 위치들에 기록되지 않을 때 압축이 발생하는 제로 활성화 값들을 압축하기 위한 기초로서 사용될 수 있으므로, 이에 의해 전체 메모리 사용을 감소시키고 다른 데이터 값들을 저장하기 위한 어드레스 위치를 자유롭게 해준다.

[0040] 제어기(302)는 메모리 뱅크(108)로부터 입력 활성화 버스(306)로 입력 활성화들을 로딩하고 그 값들을 MAC(304)를 포함하는 계산 유닛들의 어레이에 제공하기 위해 메모리 뱅크(108)에 하나 이상의 제어 신호(310)를 제공할 수 있다. 일부 구현예에서, 비트 맵(303) 또는 비트 맵(303)에 대응하는 비-제로 메모리 어드레스 인덱스는 제어기(302)에 의해 참조되어, 비-제로 활성 값들을 제공하기 위해 액세스되어야 하는 메모리 어드레스 값들을 결정할 수 있다. 활성화 값들은 제어기(302)에 의해 메모리 뱅크(108)로부터 데이터 버스(306)로 제공된다.

[0041] 일부 구현예에서, 입력 활성화들은 인덱스 또는 비트 맵(303)과 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다. 다른 구현예에서, 제어기(302)는 비트 맵(303) 또는 인덱스 중 하나에 기초하여, 제공된 입력 활성화가 제로 값을 갖는지 여부를 검출하거나 결정할 수 있다. 이러한 결정을 내리는 것에 응답하여, 제어기(302)는 계산 어레이 내의 유닛 또는 MAC(304)으로 제어 신호를 제공하여, 불필요한 곱셈 동작(예를 들어, 0으로 곱하는)의 발생을 방지, 정지 또는 억제할 수 있다. 컴퓨팅 시스템(300) 내에서, 에너지 절약은 제로 활성화 값을 제공하고 이어서 또는 동시에 해당 활성화와 관련된 계산 동작을 디스에이블시킴으로써 실현될 수 있다.

[0042] 전술한 바와 같이, 인덱스는 비-제로 값을 갖는 입력 활성화들을 갖는 모든 메모리 어드레스 위치를 포함한다. 데이터 버스(306)는 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능하다. 계산 어레이의 유닛들은 데이터 버스(306)로부터 하나 이상의 비-제로 활성화 값을 수신하여, 수신된 활성화 값들에 기초하여 행렬 곱셈에 관한 계산을 수행할 수 있다. 일부 구현예에서, 컴퓨팅 시스템(300)은 인덱싱된 어드레스에 대응하는 메모리 어드레스 위치로부터의 입력 활성화만을 제공할 것이다. 따라서, 제로 활성화들이 입력 버스(306)에 제공되지 않을 것이므로, 계산 동작들이 디스에이블되거나 그렇지 않으면 발생하는 것이 방지될 수 없다. 컴퓨팅 시스템(300)이 이 통신 기법을 사용할 때 계산 효율은 제로에 의한 곱셈을 제거함으로써 향상되거나 가속될 수 있다.

[0043] 주어진 계산 사이클에 대해, 컴퓨팅 시스템(300)은 신경망 계층에 대한 추론 계산들과 관련된 곱셈 동작들을 실행하기 위해 활성화 구조(102) 및 파라미터 구조(104)의 요소에 대한 액세스를 요구할 수 있다. 전술한 바와 같이, 메모리 뱅크(108) 및 메모리 뱅크(110)에 대한 소정 메모리 어드레스 값들은 각각 활성화 구조(102) 및 파라미터 구조(104)의 요소에 대응할 수 있다.

[0044] 계산들이 수행되는 사이클에서, 제어기(302)는 한 번에 하나의 입력 활성화 값을 제공할 것이고, MAC 셀(304)을 포함하는 계산 유닛 어레이는 주어진 입력 활성화에 대해 상이한 출력 활성화들을 생성하기 위해 가중치를 활성화에 곱할 것이다. 상기 계산 유닛 어레이의 각 요소(파라미터 구조로서 상술된) 또는 MAC 셀(304)은 신경망 계층의 상이한 출력 깊이를 담당할 수 있다. 일반적으로, 제어기(302)가 제로 활성 값을 검출할 때마다, 제어기(302)는 1) 메모리 뱅크(108)에 그 활성화 값을 저장하지 않거나; 2) 그 활성화 값을 제공하지 않거나, 또는 3) 그 값을 제공하고 특정 계산 유닛에 제어 신호를 제공하여 해당 유닛이 그 제로 활성화 값에 대응하는 곱셈 동작을 수행하지 않도록 할 수 있다.

[0045] 계산 유닛들의 어레이는 특정 계산을 스킵하거나 방지할 필요가 있을 때 제로 활성 값의 검출에 기초하여, 결정할 수 있는 제어기(302)에 의해 전적으로 제어된다. 따라서, 특정 계산을 스킵하기 위해 계산 유닛들의 어레이 내에 추가적인 복잡한 하드웨어 구조가 필요하지 않다. 게다가, 입력 활성화 값들은 메모리 뱅크(108)에 저장하기 위해 컴퓨팅 시스템(300)에 도달할 때 분석될 수 있다. 입력 활성화들을 분석하는 것에 응답하여, 제어기(302)는 메모리(108)에 비-제로 값들만 저장함으로써 활성화 데이터를 효율적으로 압축하기 위한 명령들을 실행할 수 있으므로, 이에 따라 메모리 저장 공간 및 대응 대역폭을 절약할 수 있다.

[0046] 컴퓨팅 시스템(300)이 입력 활성화들 및 가중치들을 수신할 때, 제어기(302)는 예를 들어 하나 이상의 직접 메모리 액세스 동작을 실행할 수 있다. 이러한 메모리 액세스 동작들의 실행은 메모리 뱅크(108)의 어드레스 위치들에 활성화 구조(102)의 차원 요소들에 대응하는 입력 활성화들을 저장하는 것을 포함한다. 마찬가지로, 제어기(302)는 또한 메모리 뱅크(110)의 어드레스 위치들에, 파라미터 구조(104)의 차원 요소들에 대응하는 파라미터들을 저장할 수 있다. 비트 맵(303) 이외에, 제어기(302)는 (예를 들어, 제로 값 또는 비제로 값을 갖는) 특정 입력 활성화가 인출(fetch)될 메모리 어드레스를 유지하는 하나 이상의 어드레스 레지스터를 더 포함할 수 있다. 더욱이, 하나 이상의 레지스터는 또한 대응하는 가중치가 인출되는 메모리 어드레스를 특정 입력 활성화와 곱하기 위해 저장할 것이다.

[0047] 전술한 바와 같이, 제어기(302)는 비트 맵(303)에 부분적으로 기초하여 비-제로 활성화 값들에 대한 메모리 어드레스를 식별한다. 일부 구현예에서, 제어기(302)는 비트 맵(303)을 판독하여, 예를 들어, 비-제로 활성화 값을 갖는 적어도 2개의 메모리 어드레스를 결정한다. 제어기(302)가 제로 활성화 값들을 제공하고, 후속하여 그 값들에 대해 계산을 스킵하거나 디스에이블하도록 구성되면, 제어기(302)는 또한 제로 활성화 값을 갖는 적어도 하나의 메모리 어드레스를 결정할 수 있다. 이 구현예에서, 제어기(302)는 위에서 언급한 레지스터들을 참조하여, 제1 입력 활성화에 대한 대응하는 가중치(및 메모리 어드레스)를 결정하고 제2 입력 활성화에 대한 대응하는 가중치(및 메모리 어드레스)를 결정할 수 있다.

[0048] 전술한 바와 같이, 제어기(302)는 메모리에 하나 이상의 어드레스 레지스터를 유지한다. 따라서, 제로 값 입력 활성화 검출시, 오퍼랜드들(입력 활성화 및 가중치)의 임의의 잠재적인 오정렬을 완화하거나 방지하기 위해, 제어기(302)는 대응하는 계산 유닛을 디스에이블하고, 특정 가중치를 로딩하는 것을 스킵하고, 다음의 비-제로 입력 활성화를 위한 적절한 대응하는 가중치(및 메모리 어드레스)를 검색하여, 주어진 신경망 계층에 대한 출력 활성화를 계산하는 것을 재개할 수 있다.

[0049] 일부 구현예에서, 제1 신경망 층에서 계산된 출력 활성화들은 네트워크의 다음 제2 층, 예를 들어 네트워크의 다음 히든 계층 또는 출력 계층에 대한 입력 활성화로서 사용된다. 일반적으로, 신경망의 각 계층은 각각의 파라미터 세트의 현재 값에 따라 수신 입력으로부터 출력을 생성한다. 일부 예에서, 제어기(302)는 프로그래밍된 명령들(즉, 출력 로직)를 실행하여, 출력 활성화 버스(308)로 제공된 출력 활성화들과 관련된 데이터 스트림을 분석할 수 있다. 출력 활성화 데이터 스트림을 분석하는 것은 제어기(302)가 각각의 출력 활성화와 관련된 값이 제로 값 또는 비-제로 값인지를 검출하거나 결정할 수 있게 한다. 제어기(302)는 예시적인 출력 활성화 데이터 스트림을 분석하여 각각의 검출된 비-제로 활성 값을 비트 맵(305)에 매핑할 수 있다. 비트 맵(305)에서 맵핑된 비-제로 활성 값들은 네트워크에서 다음의 제2 계층과 연관된 계산들을 담당하는 후속 컴퓨팅 시스템(300)에 입력 활성화들로서 비-제로 값만을 공급하는데 사용될 수 있다.

[0050] 대안적인 구현예에서, 단일 비-제로 입력 활성화가 파라미터 구조(104)의 주어진 차원 요소에 대한 다양한 가중치를 커버하는 몇몇 곱셈 연산(즉 "x" 또는 "y"차원을 반복하는)에 대한 오퍼랜드로서 사용되는 일부 계산 동작이 있을 수 있다. 예를 들어, 제어기(302)가 메모리 뱅크(108)로 하여금 제1 입력 활성화(예컨대, 비-제로 값)를 제공하게 하면, 파라미터 구조(104a)는 활성화를 수신하고, 주어진 어드레스에서의 대응하는 가중치는 또한 파라미터 구조(104a)에 로딩된다. 파라미터 구조(104a)는 제1 입력 활성화가 K개의 계산 사이클에 걸쳐 영향을 미치는 특정 수의 부분 합(예를 들어, 변수 "K"로 표시됨)을 업데이트하도록 진행할 것이다. 결과적으로, 이들 K 사이클 동안, 파라미터 구조(104a)는 추가적인 입력 활성화를 수신하지 않을 것이다. 그 다음 제어기(302)는 다음의 입력 활성화가 입력 활성화 버스(306)에 제공되도록 제어 신호를 메모리 뱅크(108)에 제공할 수 있다.

[0051] 도 4는 입력 버스(306)를 통해 하나 이상의 MAC 연산기(operator)들에게 입력 버스(306)를 통해 활성화(404)를 제공하는 메모리 뱅크(108)를 포함하는 예시적인 아키텍처를 도시한다. 시프트 레지스터(404)는 MAC 셀(304) 내의 하나 이상의 MAC 연산기에 의한 수신을 위해 활성화들(404)이 입력 버스(306) 상에 한번에 하나씩 전송되는 시프트 기능을 제공할 수 있다. 도시된 바와 같이, 일 구현예에서, 활성화(406)는 제로의 활성화 값을 가질 수 있고, 따라서, MAC 셀(304)에 의해 소비되지 않을 수 있다.

[0052] 일반적으로, MAC 연산기를 포함하는 MAC 셀(304)은 부분 합을 계산하고 일부 구현예에서는 부분 합 데이터를 출력 버스(308)에 기록하도록 구성된 계산 유닛들로 정의된다. 도시된 바와 같이, 셀들(304)은 하나 이상의 MAC 연산기로 구성될 수 있다. 일 구현예에서, MAC 셀(304)내의 MAC 연산기의 수는 셀의 발행 폭으로 지칭된다. 예로서, 이중 발행(dual issue) 셀은 (메모리 뱅크(108)로부터의) 2개의 활성화 값과 (메모리(110)로부터의) 2개의 파라미터와의 곱셈을 계산하고, 2개의 곱셈기의 결과와 현재의 부분 합간의 덧셈을 수행한다.

[0053] 전술한 바와 같이, 입력 버스(306)는 선형 유닛(즉, MAC 어레이(304))의 MAC 연산기에게 입력 활성화들을 제공하는 통신 버스이다. 일부 구현예에서, 동일한 입력은 모든 MAC 연산기들 사이에서 공유된다. 입력 버스(306)의 폭은 주어진 MAC 어레이(304)에 대한 대응 셀 수에 입력 활성화를 공급하기에 충분히 넓어야만 한다. 입력 버스(306)의 구조를 설명하기 위해 다음의 예를 고려한다. 선형 유닛 내의 셀의 수가 4와 같고 활성화 폭이 8 비트와 같을 때, 입력 버스(306)는 매 사이클마다 최대 4개의 입력 활성화를 제공하도록 구성될 수 있다. 이 예에서, MAC 어레이(304) 내의 모든 셀은 제공되는 4개의 활성화들 중 하나를 액세스할 뿐이다.

[0054] 일부 예에서, 명령 데이터(312)는 MAC 어레이(304)의 셀들이 동일한 입력 활성화를 사용하여 계산들을 수행할 필요가 있음을 나타낼 수 있다. 이는 MAC 어레이(304)의 셀 내에서 Zout 파티셔닝으로 지칭될 수 있다. 마찬가지로, 셀 내의 Zin 파티셔닝은 MAC 어레이(304)의 셀들이 계산을 수행하기 위해 상이한 활성화를 필요로할 때 발생한다. 전자의 경우, 단일 입력 활성화는 4번 복제되고, 메모리 뱅크(108)로부터 판독된 4개의 활성화는 4 사이클에 걸쳐 제공된다. 후자의 경우, 메모리(180)의 판독은 매 사이클마다 요구된다.

[0055] 도 5는 파라미터 계산을 감소시키고 입력 데이터 희소성을 이용하기 위한 프로세스의 예시적인 흐름도이다. 블록(502)에서, 컴퓨팅 시스템(300)은 제로 활성화 값 또는 비-제로 활성화 값을 갖는 입력 활성화들을 수신한다. 전술한 바와 같이, 일부 구현예에서, 컴퓨팅 시스템(300)은 예시적인 신경망 하드웨어 시스템의 호스트 인터페이스 디바이스 또는 상위 레벨 제어기로부터 입력 활성화들을 수신할 수 있다.

[0056] 블록(504)에서, 제어기(302)는 입력 활성화들 각각이 제로 값 또는 비-제로 값인지를 결정한다. 일부 구현예에서, 제어기(302)는 입력 활성화 데이터 스트림을 분석하고, 각각의 검출된 제로 값 및 비-제로 값을 제로 입력 활성 값("0") 및 비-제로 입력 활성화 값("1")에 대응하는 이진 값들을 포함하는 비트 맵에 매핑한다.

[0057] 블록(506)에서, 제어기(302)는 수신된 입력 활성화들을 메모리 뱅크(108)에 저장한다. 입력 활성화를 저장하는 것은 비-제로 값들을 포함하는 입력 활성화들을 갖는 하나 이상의 메모리 어드레스 위치들의 인덱스를 생성하는 제어기(302)를 포함할 수 있다. 일부 구현예에서, 인덱스는 비트 맵(303)에 기초하여 생성된다. 예를 들어, 비트 맵(303)의 각 비트는 비-제로 활성화 값 또는 제로 활성화 값을 나타내기 때문에, 비트 맵(303)은 입력 활성화들을 메모리 뱅크(108)에 기록할 때 비-제로 값을 갖는 메모리 어드레스 위치의 인덱스를 생성하기 위해 제어기(302)에 의해 참조될 수 있다.

[0058] 블록(508)에서, 제어기(302)는 메모리 뱅크(108)로부터 적어도 하나의 입력 활성화를 데이터 버스(306)에 제공한다. 일부 구현예에서, 입력 활성화들은 적어도 부분적으로 인덱스로 식별된 메모리 어드레스 위치로부터 제공된다. 전술한 바와 같이, 인덱스는 비-제로 값으로 입력 활성화들을 저장하는 모든 메모리 주소 위치를 식별한다. 데이터 버스(306)는 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능하다. 계산 어레이의 유닛들은 데이터 버스(306)로부터 하나 이상의 비-제로 활성화 값을 수신하여 행렬 곱셈에 관한 계산을 수행한다. 일부 구현예에서, 컴퓨팅 시스템(300)은 인덱싱된 어드레스에 대응하는 메모리 어드레스로부터의 입력 활성화만을 제공할 것이다. 컴퓨팅 시스템(300)이 이 통신 기법을 사용할 때, 계산 효율은 제로에 의한 곱셈을 제거함으로써 향상될 수 있다.

[0059] 블록(510)에서, 인덱싱된 어드레스로부터 입력 활성화만 아니라 모든 활성화 값이 제공되는 구현예에서, 제어기(302)는 입력 활성화가 비-제로 활성 값들을 포함하는 임의의 인덱싱된 어드레스와 관련되지 않은 메모리 어드레스로부터 제공됨을 검출한다. 이 검출 단계에 응답하여, 제어기(302)는 제로 입력과 관련된 곱셈 연산을 방지하기 위해 연산 어레이의 적어도 하나의 유닛에 제어 신호를 제공할 수 있다. 컴퓨팅 시스템(300)이 이 통신 기법을 사용하는 경우, 유용한 결과를 산출하지 않는 불필요하거나 낭비적인 계산을 방지함으로써 에너지 절약을 실현할 수 있다(예를 들어, 유용한 결과는 부분합 또는 출력 활성화의 계산을 포함한다).

[0060] 본 명세서에 기술된 주제 및 기능적 동작의 실시예는 본 명세서 및 그 구조적 등가물에 개시된 구조 또는 이들 중 하나 이상의 조합을 포함하여, 디지털 전자 회로, 유형으로 구현된 컴퓨터 소프트웨어 또는 펌웨어, 컴퓨터 하드웨어로 구현될 수 있다. 본 명세서에서 기술된 주제의 실시예는 하나 이상의 컴퓨터 프로그램, 즉 데이터 처리 장치에 의해 실행되거나 또는 데이터 처리 장치의 동작을 제어하기 위한 유형의 비일시적 저장 매체 상에

인코딩된 컴퓨터 프로그램 명령의 하나 이상의 모듈로 구현될 수 있다. 대안적으로 또는 부가적으로, 프로그램 명령들은 데이터 처리 장치에 의한 실행을 위해 적절한 수신기 장치로 전송하도록 정보를 인코딩하기 위해 생성되는 인위적으로 생성된 전파 신호, 예를 들어, 기계-생성의 전기, 광학 또는 전자기 신호상에서 인코딩될 수 있다. 컴퓨터 저장 매체는 기계 판독 가능 저장 디바이스, 기계 판독 가능 저장 기판, 랜덤 또는 시리얼 액세스 메모리 디바이스, 또는 이들 중 하나 이상의 조합일 수 있다

[0061] 본 명세서에 설명된 프로세스 및 논리 흐름은 입력 데이터를 조작하고 출력을 생성함으로써 기능들을 수행하기 위해 하나 이상의 컴퓨터 프로그램을 실행하는 하나 이상의 프로그램 가능 컴퓨터에 의해 수행될 수 있다. 프로세스 및 논리 흐름은 또한 FPGA(필드 프로그래머블 게이트 어레이), ASIC(주문형 집적 회로), GPGPU (범용 그래픽 처리 유닛) 또는 다른 처리 유닛과 같은 특수 목적의 논리 회로에 의해 수행될 수 있으며, 장치는 또한 특수 목적의 논리 회로로 구현될 수있다.

[0062] 컴퓨터 프로그램의 실행에 적합한 컴퓨터는 범용 또는 특수 목적 마이크로 프로세서 또는 둘 모두 또는 임의의 다른 종류의 중앙 처리 장치를 기반으로 할 수 있다. 일반적으로, 중앙 처리 장치는 판독 전용 메모리 또는 랜덤 액세스 메모리 또는 둘 모두로부터 명령 및 데이터를 수신할 것이다. 컴퓨터의 필수 구성 요소는 명령을 수행하거나 실행하기 위한 중앙 처리 장치 및 명령 및 데이터를 저장하기 위한 하나 이상의 메모리 장치이다. 일반적으로, 컴퓨터는 자기, 광 자기 디스크 또는 광 디스크와 같이 데이터를 저장하기 위한 하나 이상의 대용량 저장 디바이스로부터 데이터를 수신하거나 전송하기 위해 동작 가능하게 결합될 것이다. 그러나, 컴퓨터에는 이러한 디바이스가 있을 필요가 없다.

[0063] 컴퓨터 프로그램 명령 및 데이터를 저장하기에 적합한 컴퓨터 판독 가능 매체는, 예를 들어 반도체 메모리 디바이스(예를 들어, EPROM, EEPROM 및 플래시 메모리 디바이스), 자기 디스크(예를 들어, 내부 하드 디스크 또는 이동식 디스크), 광 자기 디스크, 및 CD ROM 및 DVD-ROM 디스크를 포함하여, 임의의 형태의 비 휘발성 메모리, 매체 및 메모리 디바이스를 포함한다. 프로세서 및 메모리는 특수 목적 논리 회로에 의해 보충되거나 또는 그 안에 포함될 수 있다.

[0064] 본 명세서는 다수의 특정 구현 세부 사항을 포함하지만, 이들은 임의의 발명의 범위 또는 청구 범위에 대한 제한으로서 해석되어서는 안되며, 오히려 특정 발명의 특정 실시예에 특정될 수 있는 특징에 대한 설명으로 해석되어야 한다. 별도의 실시예와 관련하여 본 명세서에서 기술되는 소정 특징들은 또한 단일 실시예에서 조합하여 구현될 수 있다. 반대로, 단일 실시예의 문맥에서 설명된 다양한 특징은 또한 다수의 실시예에서 개별적으로 또는 임의의 적합한 하위 조합으로 구현될 수 있다. 더욱이, 특징들은 소정 조합으로 작용하는 것으로 상술될 수 있고, 처음에는 그와 같이 청구될 지라도, 청구된 조합으로부터의 하나 이상의 특징이 어떤 경우 조합으로부터 제거될 수 있고, 청구된 조합은 부분 결합 또는 서브 조합 의 변형으로 유도될 수 있다.

[0065] 유사하게, 도면에서 동작들이 도시되고 특정 순서로 청구 범위에 인용되어 있지만, 이것은 바람직한 결과를 얻기 위해, 이들 동작이 도시된 특정 순서 또는 순차적 순서로 수행되거나 도시된 모든 동작이 수행될 것을 요구하는 것으로 이해되어서는 안된다. 특정 상황에서는 멀티 태스킹 및 병렬 처리가 유리할 수 있다. 또한, 상술한 실시예에서 다양한 시스템 모듈 및 구성 요소의 분리는 모든 실시예에서 그러한 분리를 요구하는 것으로 이해되어서는 안되며, 기술된 프로그램 구성 요소 및 시스템은 일반적으로 단일 소프트웨어 제품에 통합되거나 다수의 소프트웨어 제품으로 패키징 될 수 있음을 이해해야 한다.

[0066] 추가의 구현예가 다음의 예에 요약되어 있다.

[0067] 실시예 1: 컴퓨터 구현 방법은, 컴퓨팅 디바이스에 의해, 상기 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되는 복수의 입력 활성화를 수신하는 단계와; 상기 컴퓨팅 디바이스의 제어기에 의해, 상기 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계와; 상기 컴퓨팅 디바이스의 메모리 뱅크에, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계와; 상기 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계와; 그리고 상기 제어기에 의해 상기 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 포함하고, 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

[0068] 실시예 2: 실시예 1에 있어서, 상기 인덱스는 복수의 비트를 포함하는 비트 맵에 기초하여 생성되고, 상기 비트 맵의 각 비트는 비-제로 입력 활성화 값 또는 제로 입력 활성화 값 중 적어도 하나를 나타낸다.

[0069] 실시예 3: 실시예 1 또는 2에 있어서, 적어도 하나의 유닛에 의해, 상기 비-제로 값을 사용하는 계산을 수행하

기 위해 비-제로 값을 갖는 제1 입력 활성화를 제공하고 후속하여 제로 값을 갖는 제2 입력 활성화를 제공하는 단계와, 그리고 적어도 하나의 유닛에서, 상기 제로 값을 사용하여 수행될 계산을 방지하는 단계를 더 포함한다.

[0070] 실시예 4: 실시예 3에 있어서, 상기 방지하는 단계는 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는 것으로 결정하는 상기 제어기에 응답하여 발생한다.

[0071] 실시예 5: 실시예 4에 있어서, 상기 제어기에 의해, 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는지를 검출하는 단계와, 검출에 응답하여, 제로 입력 활성화 값과 관련된 곱셈 연산을 방지하기 위해 상기 계산 어레이의 적어도 하나의 유닛에 제어 신호를 제공하는 단계를 더 포함한다.

[0072] 실시예 6: 실시예 1 내지 5 중 어느 한 실시예에 있어서, 상기 방법은 상기 제어기에 의해 제1 유닛에, 제1 입력 활성화를 사용하는 텐서 계산의 제1 부분을 매핑하고, 상기 제1 유닛과 다른 제2 유닛에, 상기 제1 입력 활성화를 또한 사용하는 상기 텐서 계산의 제2 부분을 매핑하는 단계를 더 포함한다.

[0073] 실시예 7: 실시예 1 내지 6 중 어느 한 실시예에 있어서, 상기 데이터 버스 상에 단일 입력 활성화를 순차적으로 제공하는 단계를 더 포함하고, 상기 단일 입력 활성화는 인덱스와 관련된 메모리 어드레스 위치들로부터 액세스되고 선택된다.

[0074] 실시예 8: 실시예 1 내지 7 중 어느 한 실시예에 있어서, 상기 제공 단계는 제로 값을 갖는 입력 활성화들을 제공하지 않는 단계를 더 포함한다.

[0075] 실시예 9: 동작들을 수행하도록 하나 이상의 처리 디바이스에 의해 실행 가능한 명령들을 저장하는 하나 이상의 기계-판독 가능 저장 디바이스로서, 상기 동작들은, 컴퓨팅 디바이스에 의해, 상기 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되는 복수의 입력 활성화를 수신하는 단계와; 상기 컴퓨팅 디바이스의 제어기에 의해, 상기 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계와; 상기 컴퓨팅 디바이스의 메모리 뱅크에, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계와; 상기 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계와; 그리고 상기 제어기에 의해 상기 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 포함하고, 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

[0076] 실시예 10: 실시예 9에 있어서, 상기 인덱스는 복수의 비트를 포함하는 비트 맵에 기초하여 생성되고, 상기 비트 맵의 각 비트는 비-제로 입력 활성화 값 또는 제로 입력 활성화 값 중 적어도 하나를 나타낸다.

[0077] 실시예 11: 실시예 9에 있어서, 적어도 하나의 유닛에 의해, 상기 비-제로 값을 사용하는 계산을 수행하기 위해 비-제로 값을 갖는 제1 입력 활성화를 제공하고 후속하여 제로 값을 갖는 제2 입력 활성화를 제공하는 단계, 및 적어도 하나의 유닛에서, 상기 제로 값을 사용하여 수행될 계산을 방지하는 단계를 더 포함한다.

[0078] 실시예 12: 실시예 11에 있어서, 상기 방지하는 단계는 상기 입력 활성화가 상기 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는 것으로 결정하는 상기 제어기에 응답하여 발생한다.

[0079] 실시예 13: 실시예 12에 있어서, 상기 제어기에 의해, 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는지를 검출하는 단계와, 상기 검출에 응답하여, 제로 입력 활성화 값과 관련된 곱셈 연산을 방지하기 위해 상기 계산 어레이의 적어도 하나의 유닛에 제어 신호를 제공하는 단계를 더 포함한다.

[0080] 실시예 14: 실시예 9 내지 13 중 어느 한 실시예에 있어서, 상기 동작들은 상기 제어기에 의해 제1 유닛에, 제1 입력 활성화를 사용하는 텐서 계산의 제1 부분을 매핑하고, 상기 제1 유닛과 다른 제2 유닛에, 상기 제1 입력 활성화를 또한 사용하는 상기 텐서 계산의 제2 부분을 매핑하는 단계를 더 포함한다.

[0081] 실시예 15: 전자 시스템은, 컴퓨팅 디바이스에 배치되어 하나 이상의 처리 디바이스를 포함하는 제어기와; 그리고 동작들을 실행하도록 상기 하나 이상의 처리 디바이스에 의해 실행 가능한 명령들을 저장하는 하나 이상의 기계-판독 가능 저장 디바이스를 포함하고, 상기 동작들은 컴퓨팅 디바이스에 의해, 상기 컴퓨팅 디바이스 외부의 소스로부터 적어도 부분적으로 제공되는 복수의 입력 활성화를 수신하는 단계와; 상기 제어기에 의해, 상기 복수의 입력 활성화 각각이 제로 값 또는 비-제로 값 중 하나를 갖는지 여부를 결정하는 단계와; 상기 컴퓨팅 디바이스의 메모리 뱅크에, 상기 입력 활성화들 중 적어도 하나를 저장하는 단계와; 상기 제어기에 의해, 비-제로 값인 입력 활성화 값들을 갖는 하나 이상의 메모리 어드레스 위치를 포함하는 인덱스를 생성하는 단계와; 그리고 상기 제어기에 의해 상기 메모리 뱅크로부터, 계산 어레이의 하나 이상의 유닛에 의해 액세스 가능한 데

이터 버스상에 적어도 하나의 입력 활성화를 제공하는 단계를 포함하고, 상기 활성화들은 인덱스와 관련된 메모리 어드레스 위치로부터 적어도 부분적으로 제공된다.

[0082] 실시예 16: 실시예 15에 있어서, 상기 인덱스는 복수의 비트를 포함하는 비트 맵에 기초하여 생성되고, 상기 비트 맵의 각 비트는 비-제로 입력 활성화 값 또는 제로 입력 활성화 값 중 적어도 하나를 나타낸다.

[0083] 실시예 17: 실시예 15 또는 16에 있어서, 적어도 하나의 유닛에 의해, 상기 비-제로 값을 사용하는 계산을 수행하기 위해 비-제로 값을 갖는 제1 입력 활성화를 제공하고 후속하여 제로 값을 갖는 제2 입력 활성화를 제공하는 단계, 및 적어도 하나의 유닛에서, 상기 제로 값을 사용하여 수행될 계산을 방지하는 단계를 더 포함한다.

[0084] 실시예 18: 실시예 17에 있어서, 상기 방지하는 단계는 상기 입력 활성화가인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는 것으로 결정하는 상기 제어기에 응답하여 발생한다.

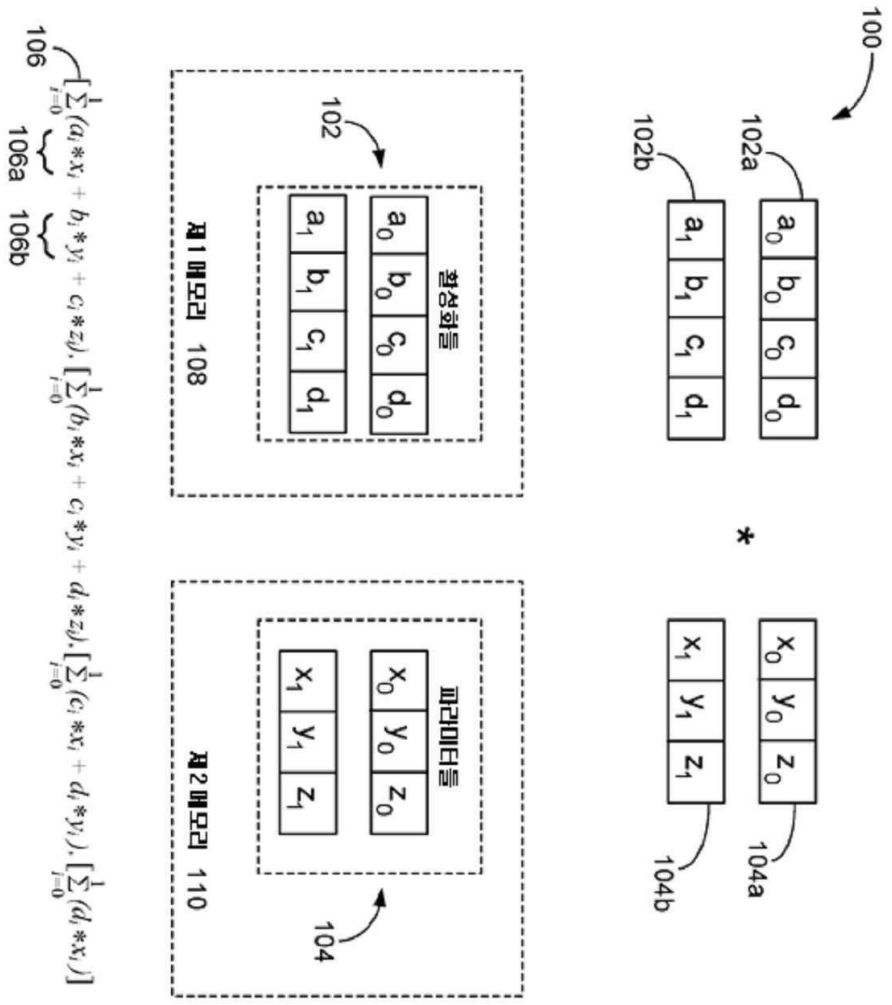
[0085] 실시예 19: 실시예 17 또는 18에 항에 있어서, 상기 제어기에 의해, 상기 입력 활성화가 인덱스와 관련되지 않은 메모리 어드레스 위치로부터 제공되는지를 검출하는 단계와, 상기 검출에 응답하여, 제로 입력 활성화 값과 관련된 곱셈 연산을 방지하기 위해 상기 계산 어레이의 적어도 하나의 유닛에 제어 신호를 제공하는 단계를 더 포함한다.

[0086] 실시예 20: 실시예 15 내지 19 중 어느 한 실시예에 있어서, 상기 동작들은 상기 제어기에 의해 제1 유닛에, 제1 입력 활성화를 사용하는 텐서 계산의 제1 부분을 매핑하고, 상기 제1 유닛과 다른 제2 유닛에, 상기 제1 입력 활성화를 또한 사용하는 상기 텐서 계산의 제2 부분을 매핑하는 단계를 더 포함한다.

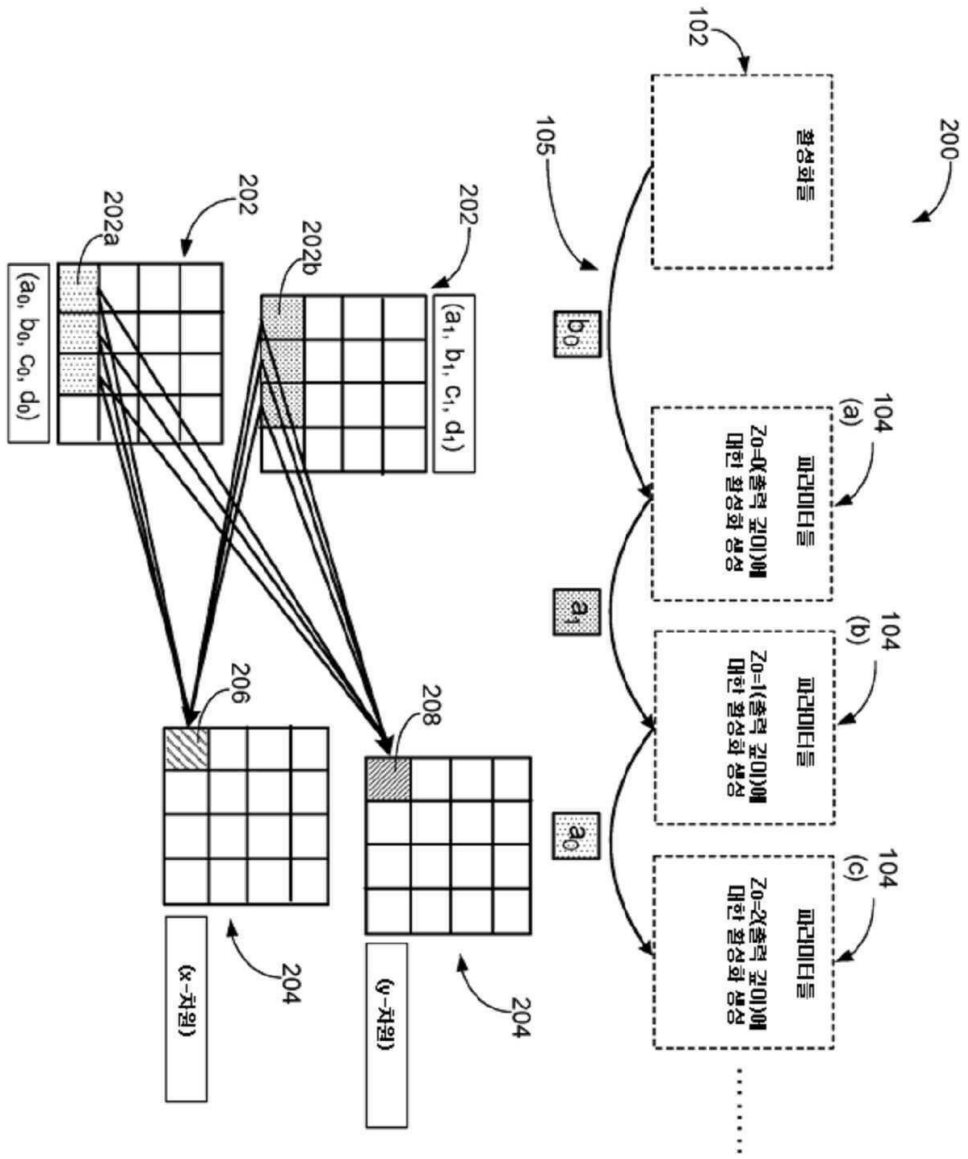
[0087] 주제의 특정 실시예들이 설명되었다. 다른 실시예들은 다음의 청구항들의 범위 내에 있다. 예를 들어, 청구항들에서 열거된 동작들은 상이한 순서로 수행될 수 있으며 여전히 바람직한 결과를 달성한다. 일례로서, 첨부된 도면에 도시된 프로세스는 바람직한 결과를 얻기 위해 도시된 특정 순서 또는 순차적 순서를 반드시 필요로 하지는 않는다. 특정 구현예에서는, 멀티 태스킹 및 병렬 처리가 유리할 수 있다.

도면

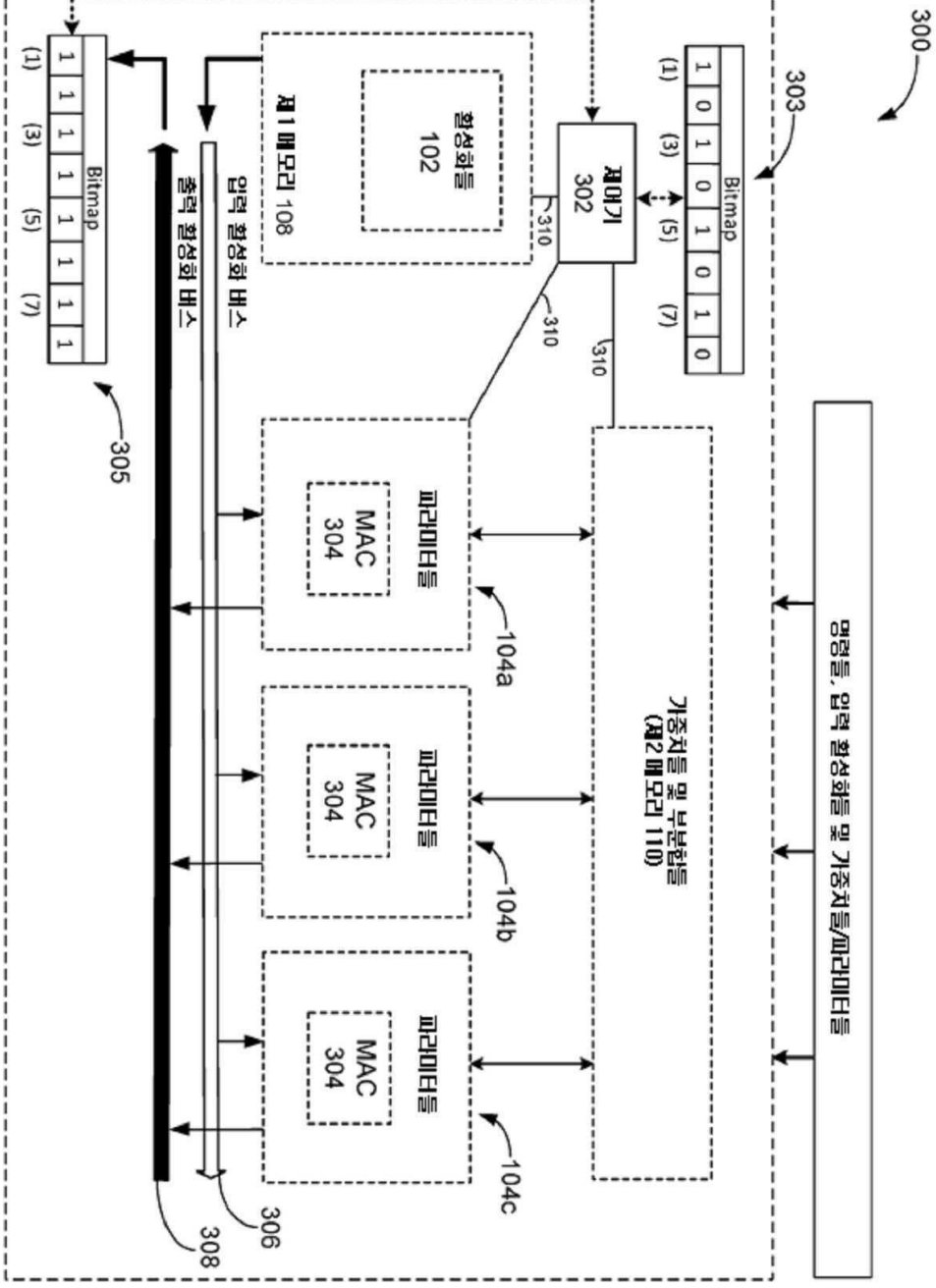
도면1


100
102a
102b
a_0 b_0 c_0 d_0
a_1 b_1 c_1 d_1
*
x_0 y_0 z_0
x_1 y_1 z_1
104a
104b
102
활성화들
a_0 b_0 c_0 d_0
a_1 b_1 c_1 d_1
제1메모리 108
104
파라미터들
x_0 y_0 z_0
x_1 y_1 z_1
제2메모리 110
106
106a
106b
$\left[\sum_{i=0}^{1}(a_i*x_i+b_i*y_i+c_i*z_i),\left[\sum_{i=0}^{1}(b_i*x_i+c_i*y_i+d_i*z_i),\left[\sum_{i=0}^{1}(c_i*x_i+d_i*y_i),\left[\sum_{i=0}^{1}(d_i*x_i)\right]\right.\right.\right.$

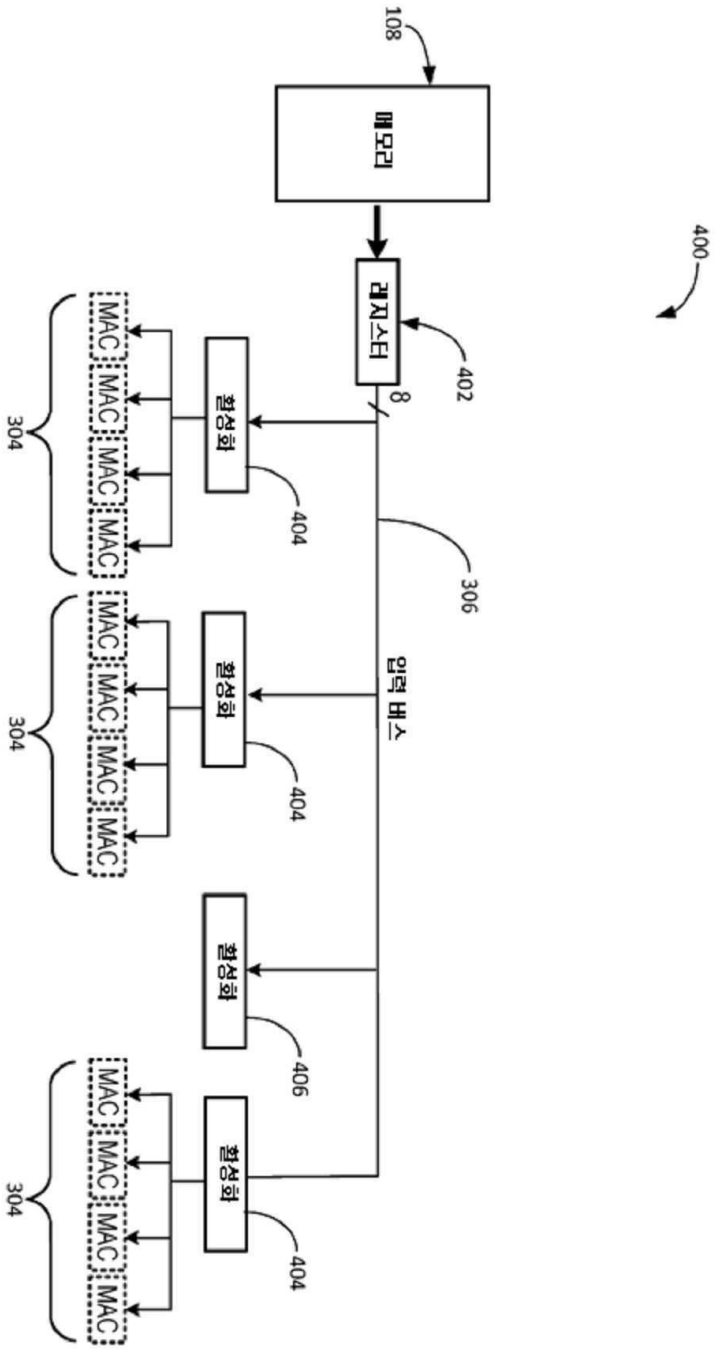
도면2


200
102
활성화들
105
b_0
a_1
a_0
104
(a)
파라미터들
Z_0=0(출력 깊이)에 대한 활성화 생성
104
(b)
파라미터들
Z_0=1(출력 깊이)에 대한 활성화 생성
104
(c)
파라미터들
Z_0=2(출력 깊이)에 대한 활성화 생성
202
(a_1, b_1, c_1, d_1)
202b
202
202a
(a_0, b_0, c_0, d_0)
208
(y-차원)
204
206
(x-차원)
204

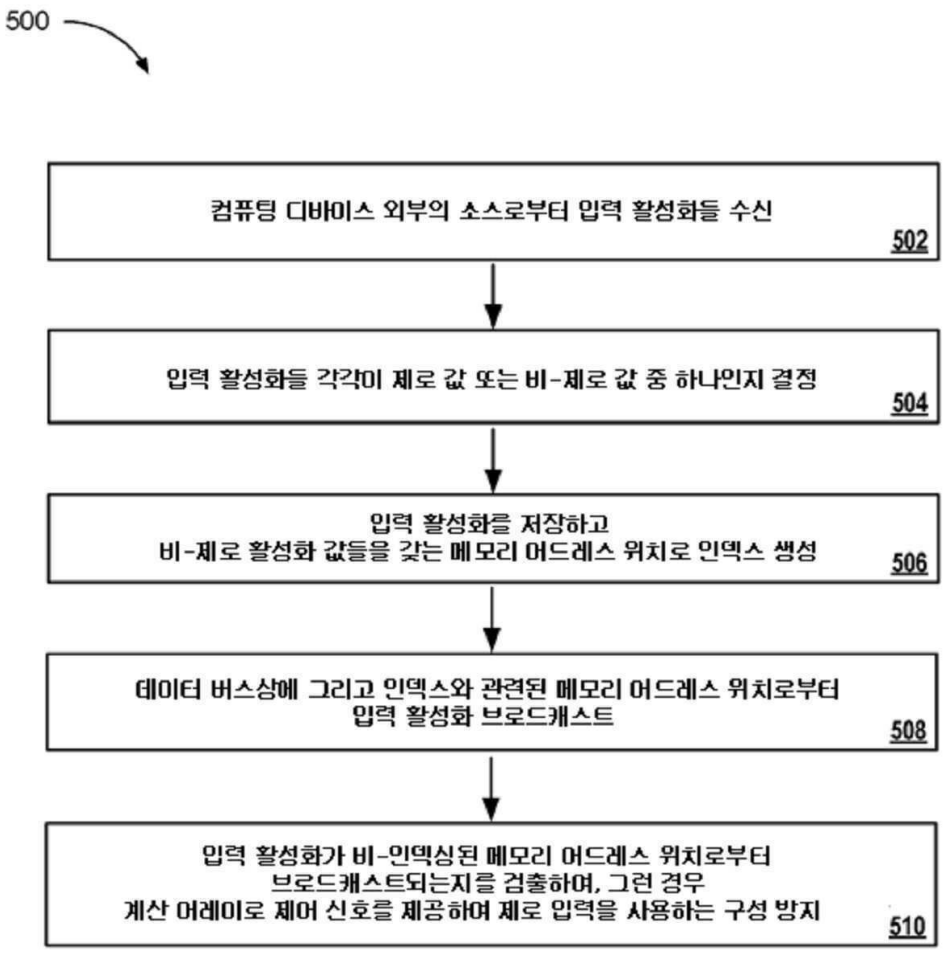
도면3


300
명령들, 입력 활성화들 및 가중치들/파라미터들
303
Bitmap
1 0 1 0 1 0 1 0
(1) (3) (5) (7)
제어기
302
310
310
310
가중치들 및 부분합들
(제2 메모리 110)
파라미터들
MAC
304
104a
104b
104c
306
308
활성화들
102
제1 메모리 108
입력 활성화 버스
출력 활성화 버스
305
Bitmap
1 1 1 1 1 1 1 1
(1) (3) (5) (7)

도면4



도면5



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

제6항에 있어서,

상기 제1 곱셈 누적 셀의 각각의 곱셈 누적 셀은,

제1 입력 버스를 통해 제1 입력 메모리로부터 각각의 0이 아닌 입력을 수신하고;

제1 신경망 계층에 대해 제1 가중치 메모리로부터 대응 가중치를 수신하고; 그리고

제1 신경망 계층에 대한 출력을 생성하기 위해 각각의 0이 아닌 입력과 대응 가중치 간에 곱셈을 수행하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.

【변경후】

제6항에 있어서,

상기 제1 복수의 곱셈 누적 셀의 각각의 곱셈 누적 셀은,

제1 입력 버스를 통해 제1 입력 메모리로부터 각각의 0이 아닌 입력을 수신하고;

제1 신경망 계층에 대해 제1 가중치 메모리로부터 대응 가중치를 수신하고; 그리고

제1 신경망 계층에 대한 출력을 생성하기 위해 각각의 0이 아닌 입력과 대응 가중치 간에 곱셈을 수행하도록 구성되는 것을 특징으로 하는 하드웨어 집적 회로.