(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0089228
(43) 공개일자 2025년06월18일

(51) 국제특허분류(Int. Cl.)
G06N 3/04 (2023.01) *G06N 3/08* (2023.01)
(52) CPC특허분류
G06N 3/04 (2023.01)
G06N 3/084 (2023.01)
(21) 출원번호 10-2023-0178909
(22) 출원일자 2023년12월11일
심사청구일자 2023년12월11일

(71) 출원인
이화여자대학교 산학협력단
서울특별시 서대문구 이화여대길 52 (대현동, 이화여자대학교)
(72) 발명자
심재형
서울특별시 마포구 만리재로 74, 207동 801호(신공덕동, 신공덕2차삼성래미안)
임하영
서울특별시 서대문구 증가로 150, 111동 1101호(남가좌동, DMC센트럴아이파크)
(뒷면에 계속)
(74) 대리인
김연권

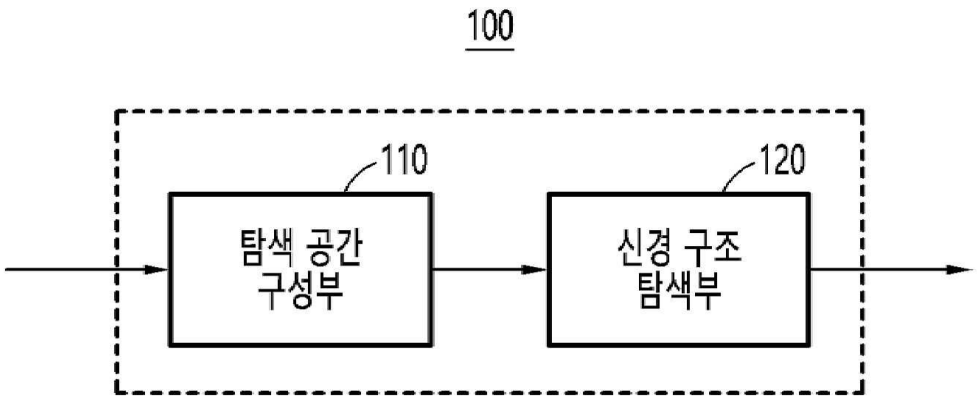
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **템플릿에 기반하는 신경 구조 탐색장치 및 그 방법**

(57) 요 약

본 발명은 신경 구조 탐색장치 및 그 방법에 관한 것으로서, 일실시예에 따른 신경 구조 탐색장치는 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성하는 탐색 공간 구성부 및 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출하는 신경 구조 탐색부를 포함할 수 있다.

대 표 도 - 도1



(72) 발명자

김주연

경기도 부천시 역곡로504번길 52, 오성빌라 라동 401호

장예서

서울특별시 강남구 삼성로 212, 19동 111호(대치동, 은마아파트)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711193622
과제번호	2021-0-02068-003
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	인공지능 혁신 허브 연구 개발
기 여 율	34/100
과제수행기관명	이화여자대학교산학협력단
연구기간	2021.07.01 ~ 2025.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1711179344
과제번호	00155966
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	인공지능융합혁신인재양성
연구과제명	인공지능융합혁신인재양성(이화여자대학교)
기 여 율	33/100
과제수행기관명	이화여자대학교산학협력단
연구기간	2022.07.01 ~ 2025.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1711194804
과제번호	00213548
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	과학기술분야 기초연구사업(IRIS)-개인연구사업-생애첫연구
연구과제명	FPGA 기반 인공지능 반도체를 위한 종단간 설계 자동화 프레임워크
기 여 율	33/100
과제수행기관명	이화여자대학교산학협력단
연구기간	2023.03.01 ~ 2026.02.28

명 세 서

청구범위

청구항 1

복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 상기 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성하는 탐색 공간 구성부 및

상기 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출하는 신경 구조 탐색부

를 포함하는 신경 구조 탐색장치.

청구항 2

제1항에 있어서,

상기 복수의 연산자는,

스킵 연결(skip connection) 연산자, 평균 풀링(average pooling) 연산자, 최대 풀링(max pooling) 연산자, 표준 컨볼루션(standard convolution) 연산자 및 깊이별 컨볼루션(depthwise convolution) 연산자 중 적어도 하나를 포함하는

신경 구조 탐색장치.

청구항 3

제1항에 있어서,

상기 탐색 공간 구성부는,

상기 복수의 연산자에 대응되는 하드웨어 블록을 100%의 활용도(utilization)를 갖도록 설계하여 상기 복수의 연산자 템플릿을 구성하는

신경 구조 탐색장치.

청구항 4

제1항에 있어서,

상기 신경 구조 탐색부는,

셀 기반의 탐색 및 경사 하강법에 기초하는 상기 신경 탐색 과정을 통해 상기 타겟 신경망을 도출하는

신경 구조 탐색장치.

청구항 5

제1항에 있어서,

상기 신경 구조 탐색부는,

상기 타겟 신경망을 구성하는 복수의 노드와, 상기 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿으로 구성되며 상기 복수의 노드를 서로 연결하는 타겟 연산자 템플릿을 각각 포함하는 복수의 셀을 추출하고, 상기 적어도 하나의 연산자 템플릿 각각의 가중치를 설정하며, 상기 가중치를 업데이트하는 학습 과정의 결과로 상기 타겟 신경망에 대응되는 적어도 하나의 후보 신경망을 도출하는

신경 구조 탐색장치.

청구항 6

제5항에 있어서,

상기 신경 구조 탐색부는,

상기 복수의 셀에 적어도 하나의 완전 연결 레이어(fully connected layer) 및 소프트맥스 활성화 함수(softmax activation function)를 적용하여 상기 후보 신경망을 도출하는

신경 구조 탐색장치.

청구항 7

제5항에 있어서,

상기 신경 구조 탐색부는,

상기 후보 신경망에 대한 평가 과정에 기초하여 상기 타겟 신경망을 도출하는

신경 구조 탐색장치.

청구항 8

제7항에 있어서,

상기 신경 구조 탐색부는,

상기 후보 신경망에 대한 모델 정확도 및 하드웨어 비용(hardware cost)을 평가하는 상기 평가 과정에 기초하여 상기 타겟 신경망을 도출하는

신경 구조 탐색장치.

청구항 9

제7항에 있어서,

상기 신경 구조 탐색부는,

교차 엔트로피 손실(cross-entropy loss) 및 하드웨어 비용에 기초하는 손실 함수에 대응하여 상기 타겟 신경망을 도출하는

신경 구조 탐색장치.

청구항 10

탐색 공간 구성부에서, 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 상기 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성하는 단계 및

신경 구조 탐색부에서, 상기 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출하는 단계

를 포함하는 신경 구조 탐색방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 신경 구조 탐색장치 및 그 방법에 관한 것으로, 보다 상세하게는 템플릿 구조로 구성된 탐색 공간을 이용하여 신경망 구조를 탐색하는 기술적 사상에 관한 것이다.

배 경 기 술

[0002] NAS(Neural Architecture Search)는 딥러닝 모델의 구조를 자동으로 찾아주는 AutoML(Automated Machine Learning)의 한 방식이다.

[0003] 기존 NAS는 모델의 정확도만을 고려하여 딥러닝 모델의 구조를 찾았기 때문에 하드웨어에서는 느리게 동작하는 경우가 많았으며, 하드웨어를 고려하더라도 특정 하드웨어에 맞춰서 동작하는 NAS 기법이 많았다.

[0004] 이에, 딥러닝 모델과 그 모델에 최적화된 하드웨어를 찾는 NAS 기법 (Hardware co-design NAS)에 대한 관심이 증가하고 있으나, 기존 하드웨어와 딥러닝 모델의 구조를 함께 찾는 NAS는 탐색 공간(Search Space)의 크기가

크다는 문제가 있다.

[0005] 구체적으로, 기존 기술들은 딥러닝 모델의 탐색 공간(일례로, 채널의 수, 층의 수 등)에 하드웨어 검색 공간(일례로, 프로세싱 요소(Processing Element)의 가로, 세로 크기, 버퍼 크기 등)이 추가되면서 탐색해야 하는 범위가 크게 증가했다.

[0006] 또한, 기존 기술들은 탐색 중에 후보 모델, 하드웨어 구조(즉, 하드웨어 블록)의 성능을 실시간으로 계산을 하는데, 이러한 방식은 시간이 오래 걸리고 부정확한 측정값을 가질 수 있으며, 이로 인해 하드웨어 후보들 중 최적의 구조가 아닌 구조를 도출하는 결과를 가져올 수 있다는 문제가 있다.

선행기술문헌

특허문헌

[0007] (특허문헌 0001) 한국공개특허 제10-2022-0150703호, "동적 탐험을 이용한 인공 신경망 구조 탐색 방법과 기록 매체에 저장한 그 컴퓨터 프로그램".

(특허문헌 0002) 한국등록특허 제10-2535636호, "뉴럴 네트워크 구조 탐색 장치 및 방법".

비특허문헌

[0008] (비특허문헌 0001) Kanghyun Choi et al., "DANCE: Differentiable Accelerator/Network Co-Exploration", in DAC. 2021.

(비특허문헌 0002) Yuhong Li et al., "EDD: Efficient Differentiable DNN Architecture and Implementation Co-Search for Embedded AI Solutions", in DAC. 2020.

발명의 내용

해결하려는 과제

[0009] 본 발명은 각 연산에 대해 최적의 하드웨어 구조를 먼저 찾고 각 연산과 최적의 하드웨어 블록을 짝지은 연산자-하드웨어 쌍을 새로운 기본 요소를 정의하여 탐색 공간을 구성하는 신경 구조 탐색장치 및 그 방법을 제공하고자 한다.

[0010] 또한, 본 발명은 연산자-하드웨어 쌍에 기초하여 탐색 공간의 크기와, 신경망 탐색에 소요되는 시간 및 자원의 수를 최소화할 수 있는 신경 구조 탐색장치 및 그 방법을 제공하고자 한다.

과제의 해결 수단

[0011] 본 발명의 일실시예에 따른 신경 구조 탐색장치는 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성하는 탐색 공간 구성부 및 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출하는 신경 구조 탐색부를 포함할 수 있다.

[0012] 일측에 따르면, 복수의 연산자는 스킵 연결(skip connection) 연산자, 평균 풀링(average pooling) 연산자, 최대 풀링(max pooling) 연산자, 표준 컨볼루션(standard convolution) 연산자 및 깊이별 컨볼루션(depthwise convolution) 연산자 중 적어도 하나를 포함할 수 있다.

[0013] 일측에 따르면, 탐색 공간 구성부는 복수의 연산자에 대응되는 하드웨어 블록을 100%의 활용도(utilization)를 갖도록 설계하여 복수의 연산자 템플릿을 구성할 수 있다.

[0014] 일측에 따르면, 신경 구조 탐색부는 셀 기반의 탐색 및 경사 하강법에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출할 수 있다.

[0015] 일측에 따르면, 신경 구조 탐색부는 타겟 신경망을 구성하는 복수의 노드와, 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿으로 구성되며 복수의 노드를 서로 연결하는 타겟 연산자 템플릿을 각각 포함하는 복수의 셀을 추출하고, 적어도 하나의 연산자 템플릿 각각의 가중치를 설정하며, 가중치를 업데이트하는 학습 과정의

결과로 타겟 신경망에 대응되는 적어도 하나의 후보 신경망을 도출할 수 있다.

[0016] 일측에 따르면, 신경 구조 탐색부는 복수의 셀에 적어도 하나의 완전 연결 레이어(fully connected layer) 및 소프트맥스 활성화 함수(softmax activation function)를 적용하여 후보 신경망을 도출할 수 있다.

[0017] 일측에 따르면, 신경 구조 탐색부는 후보 신경망에 대한 평가 과정에 기초하여 타겟 신경망을 도출할 수 있다.

[0018] 일측에 따르면, 신경 구조 탐색부는 후보 신경망에 대한 모델 정확도 및 하드웨어 비용(hardware cost)을 평가하는 평가 과정에 기초하여 타겟 신경망을 도출할 수 있다.

[0019] 일측에 따르면, 신경 구조 탐색부는 교차 엔트로피 손실(cross-entropy loss) 및 하드웨어 비용에 기초하는 손실 함수에 대응하여 타겟 신경망을 도출할 수 있다.

[0020] 본 발명의 일실시예에 따른 신경 구조 탐색방법은 탐색 공간 구성부에서, 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성하는 단계 및 신경 구조 탐색부에서, 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출하는 단계를 포함할 수 있다.

발명의 효과

[0021] 일실시예에 따르면, 본 발명은 각 연산에 대해 최적의 하드웨어 구조를 먼저 찾고 각 연산과 최적의 하드웨어 블록을 짝지은 연산자-하드웨어 쌍을 새로운 기본 요소로 정의하여 탐색 공간을 구성할 수 있다.

[0022] 일실시예에 따르면, 본 발명은 연산자-하드웨어 쌍에 기초하여 탐색 공간의 크기와, 신경망 탐색에 소요되는 시간 및 자원의 수를 최소화할 수 있다.

도면의 간단한 설명

[0023] 도 1은 일실시예에 따른 신경 구조 탐색장치를 설명하기 위한 도면이다.

도 2a 내지 도 2f는 일실시예에 따른 신경 구조 탐색장치에서 하드웨어 블록을 설계하는 예시를 구체적으로 설명하기 위한 도면이다.

도 3은 일실시예에 따른 신경 구조 탐색방법을 설명하는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0024] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.

[0025] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 변경, 균등물, 또는 대체물을 포함한다.

[0026] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 구성요소들은 용어들에 의해 한정되어서는 안 된다. 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만, 예를 들면 본 발명의 개념에 따른 권리 범위로부터 이탈되지 않은 채, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.

[0027] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성요소들 간의 관계를 설명하는 표현들, 예를 들면 "~사이에"와 "바로~사이에" 또는 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0028] 본 명세서에서 사용한 용어는 단지 특정한 실시예들을 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서,

"포함하다" 또는 "가지다" 등의 용어는 설시된 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.

[0029] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 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.

[0031] 이하, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0033] 도 1은 일실시예에 따른 신경 구조 탐색장치를 설명하기 위한 도면이다.

[0034] 도 1을 참조하면, 신경 구조 탐색장치(100)는 각 연산에 대해 최적의 하드웨어 구조를 먼저 찾고 각 연산과 최적의 하드웨어 블록을 짝지은 연산자-하드웨어 쌍을 새로운 기본 요소로 정의하여 탐색 공간을 구성할 수 있다.

[0035] 또한, 신경 구조 탐색장치(100)는 연산자-하드웨어 쌍에 기초하여 탐색 공간의 크기와, 신경망 탐색에 소요되는 시간 및 자원의 수를 최소화할 수 있다.

[0036] 이를 위해, 신경 구조 탐색장치(100)는 탐색 공간 구성부(110) 및 신경 구조 탐색부(120)를 포함할 수 있다.

[0037] 일실시예에 따른 탐색 공간 구성부(110)는 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿(즉, 연산자-하드웨어 쌍)을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성할 수 있다.

[0038] 다시 말해, 탐색 공간 구성부(110)는 사전에 복수의 연산자에 대응하여 최적의 구조를 갖는 하드웨어 블록 각각을 도출하고, 도출된 하드웨어 블록 각각을 대응되는 복수의 연산자 각각에 매칭하여 복수의 연산자 템플릿을 구성할 수 있다.

[0039] 구체적으로, 탐색 공간 구성부(110)는 하기 수식1과 같은 복수의 연산자 템플릿($\mathcal{T}_i$)에 기초하여 하기 수식2와 같은 탐색 공간($\mathcal{S}$)을 구성할 수 있다.

[0040] [수식1]

[0041] $$\mathcal{T}_i = (o_i, h_i)$$

[0042] [수식2]

[0043] $$\mathcal{S} = \{\mathcal{T}_0, \mathcal{T}_1, \ldots, \mathcal{T}_N\}$$

[0044] 여기서, o_i는 i번째(여기서, i는 양의 정수) 연산자, h_i는 연산자 o_i를 구현하는 하드웨어 블록, N은 연산자 수를 의미한다.

[0045] 즉, 기존 기술은 연산을 먼저 쌓고 그 이후에 하드웨어를 탐색하는 방식을 사용하여 전체 탐색 공간이 연산의 탐색 공간과 하드웨어의 탐색공간의 곱으로 나타나 크기가 매우 컸으나, 본 발명의 복수의 연산자 템플릿에 기반하는 탐색 공간은 최적의 연산자-하드웨어 쌍을 템플릿으로 구성하여 연산의 수를 현저히 감소시킬 수 있으며, 이에 따라 신경 구조 탐색 프로세스를 연산을 쌓는 방식만 결정하면 되는 문제로 간소화하여, 탐색 시간과 필요한 자원의 수를 최소화할 수 있다.

[0046] 또한, 복수의 연산자 템플릿에 기반하는 탐색 공간은 신경 구조 탐색 전에 연산에 최적인 하드웨어의 구조를 찾아 놓기 때문에 정답 후보의 성능 측정을 계산을 통해 측정할 수 있으며, 이러한 계산을 사용하면 빠른 시간 내에 비교적 정확하게 하드웨어의 성능을 측정할 수 있어 가장 최적의 신경망(즉, 딥러닝 모델)과 이에 대응되는

하드웨어 구조를 찾을 수 있다.

[0047] 예를 들면, 복수의 연산자는 스킵 연결(skip connection) 연산자, 평균 풀링(average pooling) 연산자, 최대 풀링(max pooling) 연산자, 표준 컨볼루션(standard convolution) 연산자 및 깊이별 컨볼루션(depthwise convolution) 연산자 중 적어도 하나를 포함할 수 있다.

[0048] 보다 구체적인 예를 들면, 복수의 연산자는 하기 표1의 연산자를 포함할 수 있으나, 이에 한정되는 것은 아니다.

표 1

[0049]

연산자	설명
nop	No operation
skip_connect	스킵(잔차) 연결
avg_pool_3x3	3x3 평균 풀링
max_pool_3x3	3x3 최대 풀링
conv_1x1	1x1 표준 컨볼루션
conv_3x3	3x3 표준 컨볼루션
conv_5x5	5x5 표준 컨볼루션
dw_conv_3x3	3x3 깊이별 컨볼루션
dw_conv_5x5	5x5 깊이별 컨볼루션

[0050] 일측에 따르면, 탐색 공간 구성부(110)는 복수의 연산자에 대응되는 하드웨어 블록을 100%의 활용도(utilization)를 갖도록 설계하여 복수의 연산자 템플릿을 구성할 수 있다.

[0051] 다시 말해, 탐색 공간 구성부(110)는 복수의 연산자 각각에 대응되는 하드웨어 블록의 각 구성요소들이 모두 활용될 수 있도록, 하드웨어 블록을 최적화 설계하여 효율을 극대화할 수 있다.

[0052] 일실시예에 따른 탐색 공간 구성부(110)에서 하드웨어 블록을 100%의 활용도를 갖도록 설계하는 방법은 이후 실시예 도 2a 내지 도 2f를 통해 보다 구체적으로 설명하기로 한다.

[0053] 일실시예에 따른 신경 구조 탐색부(120)는 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망(즉, 딥러닝 모델)을 도출할 수 있다.

[0054] 일측에 따르면, 신경 구조 탐색부(120)는 셀 기반의 탐색 및 경사 하강법(Gradient Descent)에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출할 수 있으며, 이를 통해 탐색 비용을 절감하고 최적화 문제를 해결할 수 있다.

[0055] 여기서, 경사 하강법은 기계 학습과 최적화 문제에서 매개 변수를 조정하여 모델을 학습시키는데 사용되는 최적화 알고리즘 중 하나로, 주어진 함수의 기울기(경사)를 활용하여 함수의 최소값을 탐색하되, 기울기가 감소하는 방향으로 반복적으로 이동하여 최적화된 값을 도출할 수 있다.

[0056] 일측에 따르면, 신경 구조 탐색부(120)는 셀 추출 과정, 연산자 가중치 설정 과정, 가중치 갱신 과정, 연산자 선택 과정 및 연산자 평가 과정으로 구성되는 신경 탐색 과정을 통해 타겟 신경망을 도출할 수 있다.

[0057] 구체적으로, 신경 구조 탐색부(120)는 타겟 신경망을 구성하는 복수의 노드와, 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿으로 구성되며 복수의 노드를 서로 연결하는 타겟 연산자 템플릿을 각각 포함하는 복수의 셀을 추출할 수 있다.

[0058] 예를 들면, 신경 구조 탐색부(120)는 노멀 셀(Normal Cell)과 리덕션 셀(Reduction Cell)을 포함하는 복수의 셀을 추출할 수 있으며, 여기서, 노멀 셀과 리덕션 셀은 기 설정된 순서대로 배치될 수 있다.

[0059] 보다 구체적인 예를 들면, 하나의 노멀 셀은 다른 2개의 노멀 셀 사이, 또는 다른 하나의 노멀 셀과 하나의 리덕션 셀 사이에 위치할 수 있고, 하나의 리덕션 셀은 2개의 노멀 셀 사이, 또는 M개(여기서, M은 양의 정수)의 노멀 셀 마다 위치할 수도 있다.

[0060] 다음으로, 신경 구조 탐색부(120)는 적어도 하나의 연산자 템플릿 각각의 가중치를 설정할 수 있다.

[0061] 다시 말해, 신경 구조 탐색부(120)는 타겟 연산자 템플릿에 대한 후보 연산자 템플릿(즉, 탐색 공간에 구비된 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿) 각각에 대한 가중치를 설정할 수 있다.

[0062] 예를 들면, 연산자 템플릿의 가중치는 너무 많은 후보 연산자가 사용되면 각 후보 연산자에 대응하는 가중치가 너무 작아서 학습이 제대로 수행되지 않으며, 검색 공간이 커져서 학습 시간이 과도하게 오래 걸리는 문제가 발생될 수 있으므로 최적화된 값으로 설정될 수 있으며, 여기서 가중치의 최적화된 값은 관리자의 입력에 따라 설정될 수 있다.

[0063] 다음으로, 신경 구조 탐색부(120)는 가중치를 업데이트하는 학습 과정(즉, 신경망 학습 과정)의 결과로 타겟 신경망에 대응되는 후보 신경망을 도출하고, 후보 신경망에 대한 평가 과정에 기초하여 타겟 신경망을 도출할 수 있다.

[0064] 일측에 따르면, 신경 구조 탐색부(120)는 복수의 셀에 적어도 하나의 완전 연결 레이어(Fully Connected Layer) 및 소프트맥스 활성화 함수(Softmax Activation Function)를 적용하여 후보 신경망을 도출할 수 있다.

[0065] 다시 말해, 신경 구조 탐색부(120)는 복수의 셀을 여러 번 쌓고 완전 연결 레이어와 소프트맥스 활성화 함수를 적용하여 완전한 네트워크를 구축할 수 있다.

[0066] 예를 들면, 신경 구조 탐색부(120)는 타겟 신경망에 대응되는 학습 데이터를 수신하고, 수신한 학습 데이터에 기초하는 학습 과정을 통해 가중치를 업데이트할 수 있다.

[0067] 또한, 신경 구조 탐색부(120)는 후보 신경망을 평가 세트(evaluation set)에 적용하는 평가 과정에 기초하여 타겟 신경망을 도출할 수 있으며, 여기서 평가 세트는 후보 신경망에 대한 모델 정확도 및 하드웨어 비용을 평가하는 수단을 포함할 수 있다.

[0068] 보다 구체적인 예를 들면, 신경 구조 탐색부(120)는 교차 엔트로피 손실(cross-entropy loss) 및 하드웨어 비용에 기초하는 손실 함수에 대응하여 타겟 신경망을 도출할 수 있다.

[0069] 구체적으로, 신경 구조 탐색부(120)는 하기 수식3에 기초하는 평가 과정을 통해 타겟 신경망을 도출할 수 있다.

[0070] [수식3]

[0071]
$$\min_{\alpha} \mathcal{L}_{val}(w^*(\alpha), \alpha)$$
$$\text{s.t. } w^*(\alpha) = \text{argmin}_w \mathcal{L}_{train}(w, \alpha)$$

[0072] 여기서, $\alpha \subset \mathcal{S}$는 후보 연산자 템플릿의 선택에 따른 연산자 가중치, w는 네트워크 가중치, w^*는 최적의 가중치를 의미한다.

[0073] 즉, 신경 구조 탐색부(120)는 최적의 가중치(w^*)를 수식3에 개시된 바와 같이 훈련 손실($\mathcal{L}_{train}$)를 최소화하여 도출할 수 있으며, 이때 훈련 손실($\mathcal{L}_{train}$)에 대응되는 손실 함수($\mathcal{L}$)는 하기 수식4를 통해 도출될 수 있다.

[0074] [수식4]

[0075]
$$\mathcal{L} = \mathcal{L}_{CE} + \lambda \mathcal{C}_{HW}$$

[0076] 여기서, $\mathcal{L}_{CE}$는 교차 엔트로피 손실, $\mathcal{C}_{HW}$는 하드웨어 비용(일례로, 대기 시간, 면적, 에너지 소비), λ는 하이퍼 파라미터를 의미한다.

[0077] 한편, 일실시예에 따른 신경 구조 탐색장치(즉, TD-NAAS)(100)는 하기 표2에 개시된 바와 같이, 기존 기술(즉, EDD, DANCE) 대비 정확도는 0.8% 및 0.2% 증가하고, 탐색 속도는 2.75배 빨라진 것을 확인할 수 있다.

[0078] 여기서, 표3의 실험 결과는 학습 데이터로 CIFAR-10 데이터셋을 사용하고, 연산자에 최적인 하드웨어 구조는 Synopsys Design Compiler를 사용하였으며, 합성 후 RTL 시뮬레이션 결과를 하드웨어 성능 평가 값으로 사용하고, 딥러닝 모델은 파이토치를 사용했고, NVIDIA RTX A6000 GPU 1개로 훈련한 결과를 나타낸다.

표 3

[0079]

Algorithm	Accuracy(%)	Latency(ms)
EDD	94.4	N/A

DANCE	95	11
TD-NAAS	95.2	4

[0081] 도 2a 내지 도 2f는 일실시예에 따른 신경 구조 탐색장치에서 하드웨어 블록을 설계하는 예시를 구체적으로 설명하기 위한 도면이다.

[0082] 도 2a 내지 도 2f를 참조하면, 도면부호 210은 일실시예에 따른 신경 구조 탐색장치에서 3x3 표준 컨볼루션 연산자에 대응하여 하드웨어 블록을 최적화 설계하는 방법을 설명하기 위한 도면이고, 도면부호 220 내지 240은 2x2 표준 컨볼루션 연산자에 대응하여 하드웨어 블록을 최적화 설계하는 방법을 설명하기 위한 도면이다.

[0083] 또한, 도면부호 250은 신경 구조 탐색장치에서 3x3 평균 풀링 연산자에 대응하여 하드웨어 블록을 최적화 설계하는 방법을 설명하기 위한 도면이며, 도면부호 260은 신경 구조 탐색장치에서 3x3 최대 풀링 연산자에 대응하여 하드웨어 블록을 최적화 설계하는 방법을 설명하기 위한 도면이다.

[0084] 구체적으로, 일실시예에 따른 신경 구조 탐색장치는 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성할 수 있다.

[0085] 다시 말해, 신경 구조 탐색장치는 연산자와 이에 대응되는 하드웨어 블록을 하나의 템플릿으로 묶어서 새로운 기본 요소로 정의하고, 이러한 기본 요소로 탐색 공간을 구성함으로써, 기존 기술 대비 탐색 공간의 크기를 줄여 효율적인 탐색이 가능해지며, 이로 인해 보다 좋은 성능의 신경망 구조가 도출되도록 지원할 수 있다.

[0086] 또한, 신경 구조 탐색장치는 연산자에 대응되는 하드웨어 블록이 항상 100% 활용도를 유지하도록 최적화 설계함으로써, 신경망 구조를 보다 효율적으로 처리하도록 지원할 수 있다.

[0087] 예를 들면, 신경 구조 탐색장치는 도면부호 210에 도시된 바와 같이, 연산자가 3x3 표준 컨볼루션 연산자이면, 9개의 활성값(I0 내지 I8)과 9개의 가중치(W0 내지 W8)에 대응되는 9개의 곱셈기와 이를 모두 더하는 덧셈-트리를 포함하는 하드웨어 블록을 구성하고, 3x3 컨볼루션 윈도우(convolution window)를 일렬로 펼쳐 하드웨어 블록에 공급할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0088] 보다 구체적으로, 도면부호 220 내지 240에 도시된 2x2 컨볼루션 연산자로 예시하면, 신경 구조 탐색장치는 4개의 곱셈기를 만들고 4:1 덧셈-트리를 구성하면(도면부호 220), 곱셈기(즉, 4개의 곱셈기)가 모두 동작하므로 하드웨어 블록의 활용도가 항상 100%가 되도록 설계될 수 있다.

[0089] 이 경우, 컨볼루션 윈도우가 (I0, I1, I5, I6)에서 (I1, I2, I6, I7)으로 슬라이딩(sliding) 되더라도(도면부호 230), 하드웨어 블록의 활용도가 여전히 100%를 유지하는 것을 확인할 수 있다.

[0090] 반면, 일실시예에 따른 신경 구조 탐색장치와 같이 2x2 컨볼루션 연산자에 대응되는 하드웨어 블록을 4개의 곱셈기로 최적화 설계하지 않고 5개의 곱셈기로 설계하여 5:1 덧셈-트리를 구성하면(도면부호 240), 80%의 곱셈기(즉, 5개 중 4개의 곱셈기)만 동작 하므로 하드웨어 블록의 활용도가 80%가 되어 비효율이 발생되는 것을 확인할 수 있다.

[0091] 예를 들면, 신경 구조 탐색장치는 도면부호 250에 도시된 바와 같이, 연산자가 3x3 평균 풀링 연산자이면, 9개의 활성값(I0 내지 I8)에 대응되는 덧셈-트리 및 최종 곱셈기를 포함하는 하드웨어 블록을 구성하고, 3x3 풀링 윈도우(pooling window)를 일렬로 펼쳐 하드웨어 블록에 공급할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0092] 또한, 신경 구조 탐색장치는 도면부호 250에 도시된 바와 같이, 연산자가 3x3 최대 풀링 연산자이면, 9개의 활성값(I0 내지 I8)에 대응되는 9개의 비교기-트리를 포함하는 하드웨어 블록을 구성하고, 3x3 풀링 윈도우를 일렬로 펼쳐 하드웨어에 공급할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0093] 한편, 도 2a 내지 도 2f에서는 설명의 편의 상 복수의 연산자를 3x3 표준 컨볼루션 연산자, 2x2 표준 컨볼루션 연산자, 3x3 평균 풀링 연산자 및 3x3 최대 풀링 연산자로 예시하였으나, 일실시예에 따른 복수의 연산자는 이에 한정되지 않고 NxN(여기서, N은 양의 정수) 표준 컨볼루션 연산자, NxN 평균 풀링 연산자, NxN 최대 풀링 연산자를 포함할 수 있다.

[0094] 구체적으로, 신경 구조 탐색장치는 연산자가 NxN 표준 컨볼루션 연산자이면, N^2 개의 곱셈기와, 곱셈기의 출력을 모두 더하는 덧셈-트리를 포함하는 하드웨어 블록을 구성하고, NxN 컨볼루션 윈도우를 일렬로 펼쳐 하드웨어 블록에 공급할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0095] 또한, 신경 구조 탐색장치는 연산자가 NxN 평균 풀링 연산자이면, N^2 개의 활성값 덧셈-트리와, 최종 곱셈기를 포함하는 하드웨어 블록을 구성하고, NxN 풀링 윈도우를 일렬로 펼쳐 하드웨어 블록에 공급(곱셈기에는 $1/N^2$을 공급)하고 할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0096] 또한, 신경 구조 탐색장치는 연산자가 NxN 최대 풀링 연산자이면, N^2 개의 비교기-트리를 포함하는 하드웨어 블록을 구성하고, NxN 풀링 윈도우를 일렬로 펼쳐 하드웨어 블록에 공급할 수 있으며, 이를 통해 하드웨어 블록은 활용도가 항상 100%가 되도록 설계될 수 있다.

[0098] 도 3은 일실시예에 따른 신경 구조 탐색방법을 설명하는 도면이다.

[0099] 다시 말해, 도 3은 도 1 내지 도 2f를 통해 설명한 일실시예에 따른 신경 구조 탐색장치의 동작방법을 설명하는 도면일 수 있다.

[0100] 310 단계에서 신경 구조 탐색방법은 탐색 공간 구성부에서, 복수의 연산자에 대응되는 하드웨어 블록을 각각 매칭하여 복수의 연산자 템플릿을 구성하고, 복수의 연산자 템플릿에 기초하여 탐색 공간을 구성할 수 있다.

[0101] 다시 말해, 신경 구조 탐색방법은 탐색 공간 구성부에서, 복수의 연산자에 대응하여 최적의 구조를 갖는 하드웨어 블록 각각을 사전에 매칭하여 복수의 연산자 템플릿을 구성할 수 있다.

[0102] 예를 들면, 복수의 연산자는 스킵 연결(skip connection) 연산자, 평균 풀링(average pooling) 연산자, 최대 풀링(max pooling) 연산자, 표준 컨볼루션(standard convolution) 연산자 및 깊이별 컨볼루션(depthwise convolution) 연산자 중 적어도 하나를 포함할 수 있다.

[0103] 일측에 따르면, 310 단계에서 신경 구조 탐색방법은 탐색 공간 구성부에서, 복수의 연산자에 대응되는 하드웨어 블록을 100%의 활용도(utilization)를 갖도록 설계하여 복수의 연산자 템플릿을 구성할 수 있다.

[0104] 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 탐색 공간에 기초하는 신경 탐색 과정을 통해 타겟 신경망(즉, 딥러닝 모델)을 도출할 수 있다.

[0105] 일측에 따르면, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 셀 기반의 탐색 및 경사 하강법(Gradient Descent)에 기초하는 신경 탐색 과정을 통해 타겟 신경망을 도출할 수 있으며, 이를 통해 탐색 비용을 절감하고 최적화 문제를 해결할 수 있다.

[0106] 예를 들면, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 셀 추출 과정, 연산자 가중치 설정 과정, 가중치 갱신 과정, 연산자 선택 과정 및 연산자 평가 과정으로 구성되는 신경 탐색 과정을 통해 타겟 신경망을 도출할 수 있다.

[0107] 구체적으로, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 타겟 신경망을 구성하는 복수의 노드와, 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿으로 구성되며 복수의 노드를 서로 연결하는 타겟 연산자 템플릿을 각각 포함하는 복수의 셀을 추출할 수 있다.

[0108] 예를 들면, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 노멀 셀(Normal Cell)과 리덕션 셀(Reduction Cell)을 포함하는 복수의 셀을 추출할 수 있다.

[0109] 다음으로, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 적어도 하나의 연산자 템플릿 각각의 가중치를 설정할 수 있다.

[0110] 다시 말해, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 타겟 연산자 템플릿에 대한 후보 연산자 템플릿(즉, 탐색 공간에 구비된 복수의 연산자 템플릿 중 적어도 하나의 연산자 템플릿) 각각에 대한 가중치를 설정할 수 있다.

[0111] 다음으로, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 가중치를 업데이트하는 학습 과정(즉, 신경망 학습 과정)의 결과로 타겟 신경망에 대응되는 후보 신경망을 도출하고, 후보 신경망에 대한 평가 과정에

기초하여 타겟 신경망을 도출할 수 있다.

[0112] 예를 들면, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 복수의 셀에 적어도 하나의 완전 연결 레이어(Fully Connected Layer) 및 소프트맥스 활성화 함수(Softmax Activation Function)를 적용하여 후보 신경망을 도출할 수 있다.

[0113] 또한, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 타겟 신경망에 대응되는 학습 데이터를 수신하고, 수신한 학습 데이터에 기초하는 학습 과정을 통해 가중치를 업데이트할 수 있다.

[0114] 또한, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 후보 신경망을 평가 세트(evaluation set)에 적용하는 평가 과정에 기초하여 타겟 신경망을 도출할 수 있으며, 여기서 평가 세트는 후보 신경망에 대한 모델 정확도 및 하드웨어 비용을 평가하는 수단을 포함할 수 있다.

[0115] 보다 구체적인 예를 들면, 320 단계에서 신경 구조 탐색방법은 신경 구조 탐색부에서, 교차 엔트로피 손실(cross-entropy loss) 및 하드웨어 비용에 기초하는 손실 함수에 대응하여 타겟 신경망을 도출할 수 있다.

[0116]

[0117] 결국, 본 발명을 이용하면, 각 연산에 대해 최적의 하드웨어 구조를 먼저 찾고 각 연산과 최적의 하드웨어 블록을 짝지은 연산자-하드웨어 쌍을 새로운 기본 요소로 정의하여 탐색 공간을 구성할 수 있다.

[0118] 또한, 본 발명을 이용하면, 연산자-하드웨어 쌍에 기초하여 탐색 공간의 크기와, 신경망 탐색에 소요되는 시간 및 자원의 수를 최소화할 수 있다.

[0120] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들면, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 장치, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

[0121] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

부호의 설명

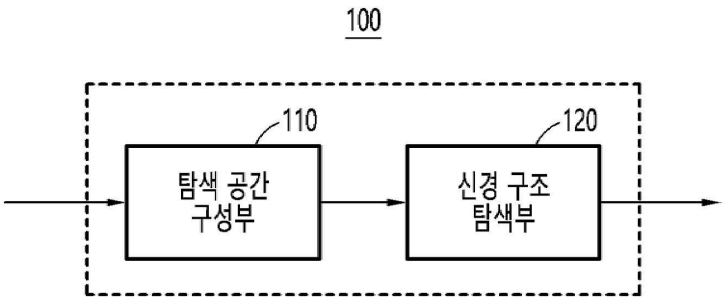
[0122] 100: 신경 구조 탐색장치

110: 탐색 공간 구성부

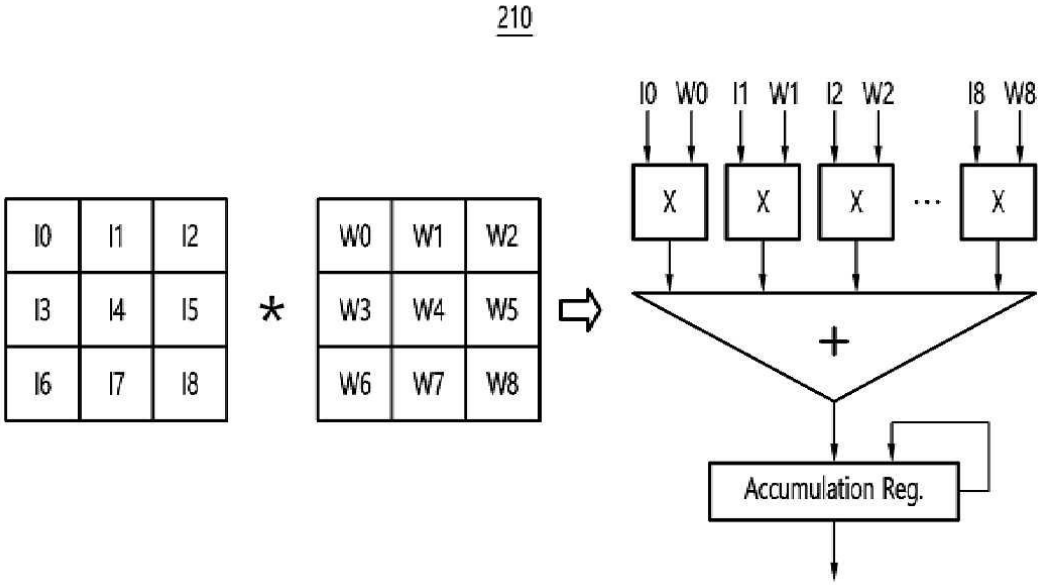
120: 신경 구조 탐색부

도면

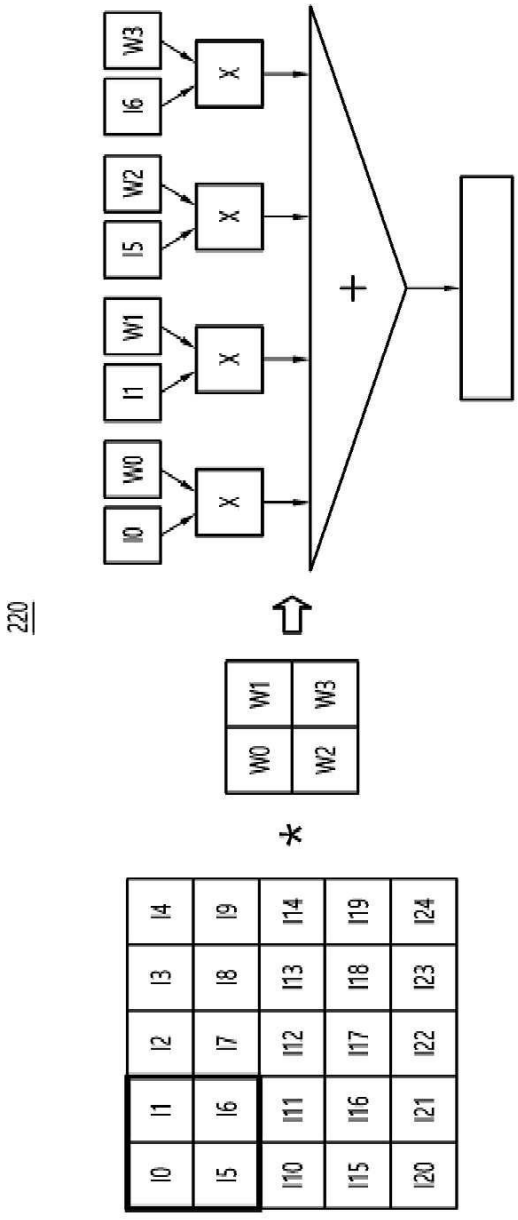
도면1



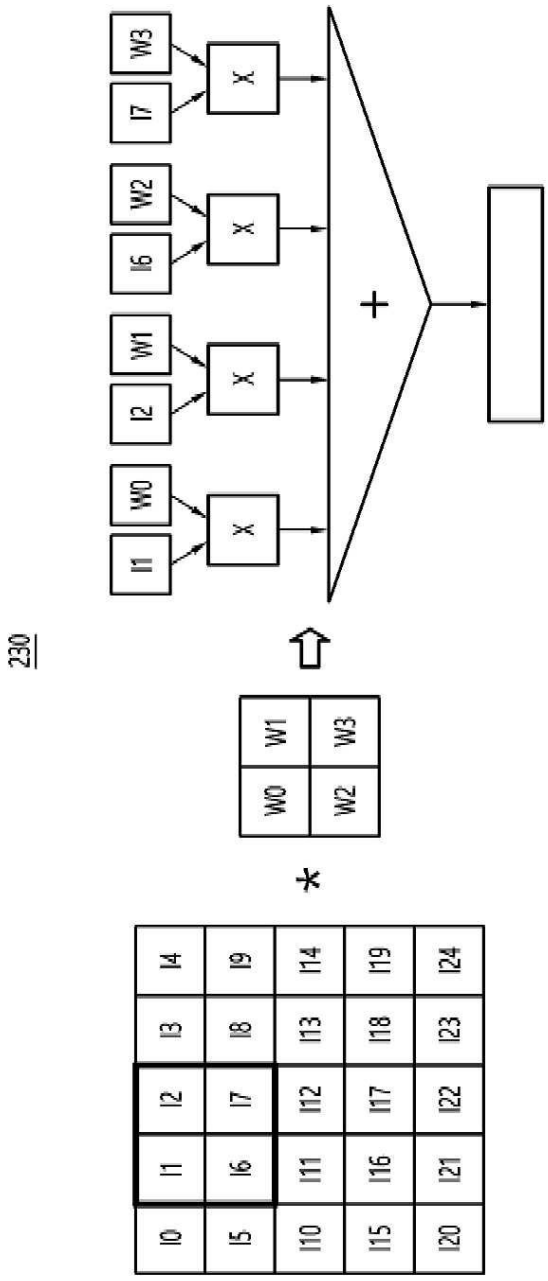
도면2a


210
I0
I1
I2
I3
I4
I5
I6
I7
I8
*
W0
W1
W2
W3
W4
W5
W6
W7
W8
I0 W0 I1 W1 I2 W2 I8 W8
X
X
X
X
+
Accumulation Reg.

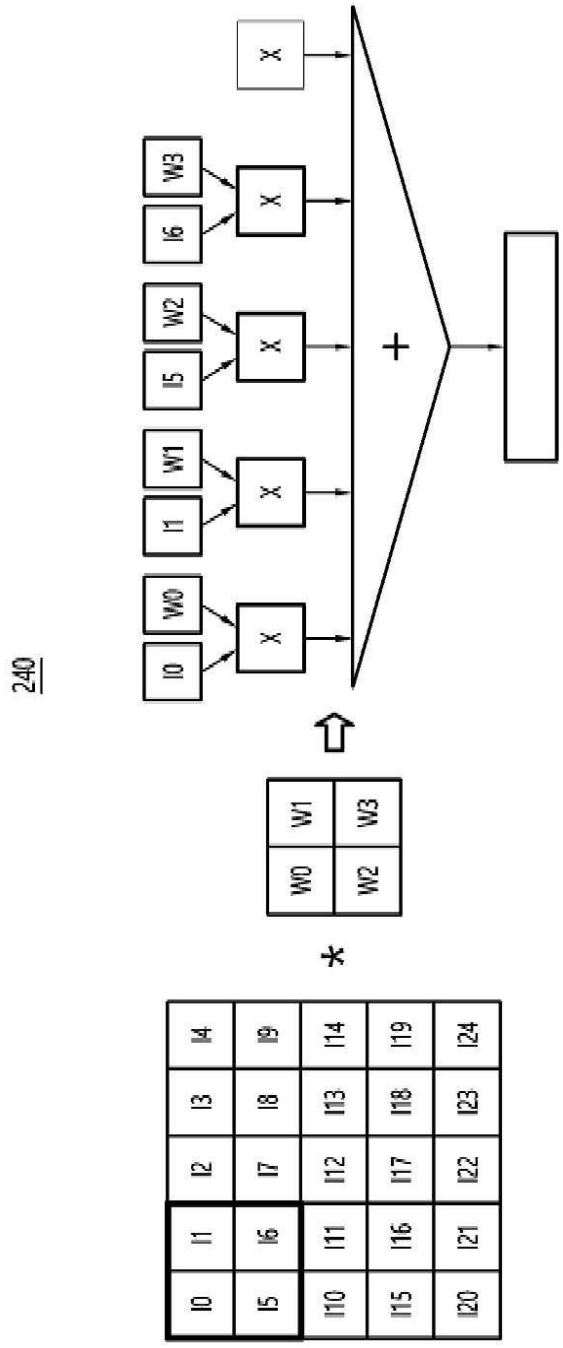
도면2b


220
W3
I6
W2
I5
W1
I1
W0
I0
X
X
X
X
+
W1
W3
W0
W2
*
I4
I9
I14
I19
I24
I3
I8
I13
I18
I23
I2
I7
I12
I17
I22
I1
I6
I11
I16
I21
I0
I5
I10
I15
I20

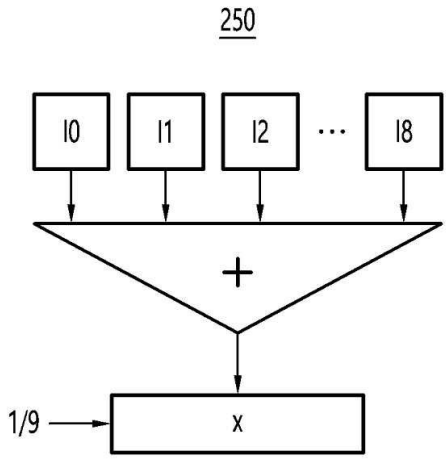
도면2c


230
W3
I7
W2
I6
W1
I2
W0
I1
X
X
X
X
+
W1
W3
W0
W2
*
I4
I9
I14
I19
I24
I3
I8
I13
I18
I23
I2
I7
I12
I17
I22
I1
I6
I11
I16
I21
I0
I5
I10
I15
I20

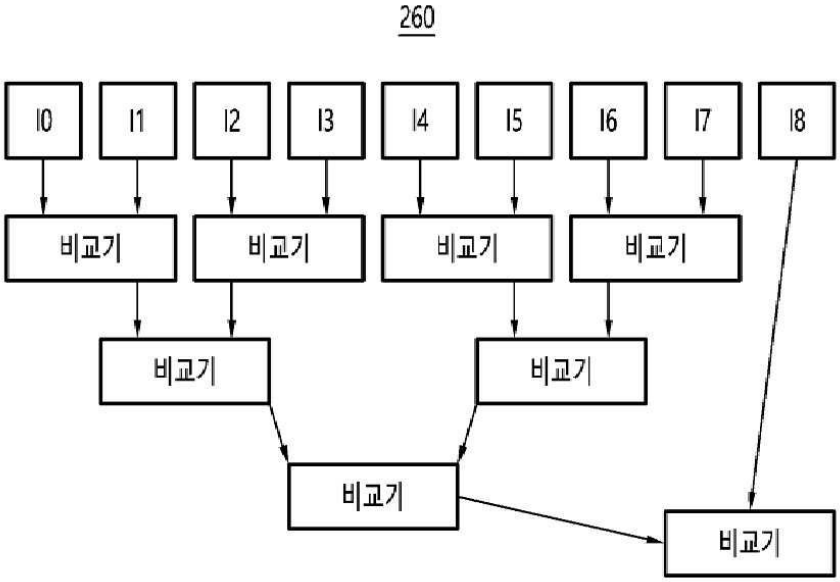
도면2d


240
I0
W0
I1
W1
I5
W2
I6
W3
X
+
W0
W1
W2
W3
*
I0
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
I12
I13
I14
I15
I16
I17
I18
I19
I20
I21
I22
I23
I24

도면2e



도면2f



도면3

