(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년11월27일
(11) 등록번호 10-2606825
(24) 등록일자 2023년11월22일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2023.01) *G06N 3/04* (2023.01)
G06N 3/08 (2023.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06N 3/04 (2023.01)
(21) 출원번호 10-2017-0117236
(22) 출원일자 2017년09월13일
심사청구일자 2020년07월27일
(65) 공개번호 10-2019-0030034
(43) 공개일자 2019년03월21일
(56) 선행기술조사문헌
KR1020170005562 A
US20160085690 A1
Jiecao Yu 외 5인, " Scalpel:Customizing DNN pruning to the underlying hardware parallelism",2017 ACM/IEEE 44th Annual International Symposium on Computer Architecture (ISCA) (2017.06.24.)*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
양승수
경기도 화성시 동탄반석로 16, 635동 403호 (반송동, 동탄나루마을 월드메르디앙 반도유보라)
(74) 대리인
리앤목특허법인

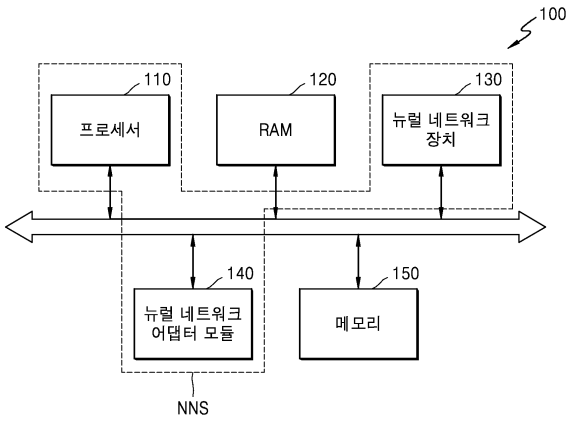
전체 청구항 수 : 총 9 항 심사관 : 이준상

(54) 발명의 명칭 **뉴럴 네트워크 모델을 변형하는 뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법**

(57) 요 약

뉴럴 네트워크 모델을 변형하는 뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법이 개시된다. 본 개시의 기술적 사상의 일측면에 따른 뉴럴 네트워크 시스템의 동작방법은, 프로세서의 제어에 기반하여, 뉴럴 네트워크 모델로부터 뉴럴 네트워크 연산에 관련된 하나 이상의 정보들을 파싱하는 단계와, 상기 프로세서의 제어에 기반하여, 상기 뉴럴 네트워크 모델을 실행할 적어도 하나의 전용 하드웨어의 정보를 판단하는 단계와, 상기 프로세서의 제어에 기반하여, 상기 전용 하드웨어의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하는 단계 및 상기 전용 하드웨어에 의해 상기 변형된 뉴럴 네트워크 모델을 실행하는 단계를 구비하는 것을 특징으로 한다.

대 표 도 - 도1



(52) CPC특허분류
G06N 3/08 (2023.01)

명 세 서

청구범위

청구항 1

프로세서의 제어에 기반하여, 뉴럴 네트워크 모델로부터 뉴럴 네트워크 연산에 관련된 하나 이상의 정보들을 파싱하는 단계;

상기 프로세서의 제어에 기반하여, 상기 뉴럴 네트워크 모델을 실행할 적어도 하나의 전용 하드웨어의 정보들을 판단하는 단계;

상기 프로세서의 제어에 기반하여, 상기 전용 하드웨어의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하는 단계; 및

상기 전용 하드웨어에 의해 상기 변형된 뉴럴 네트워크 모델을 실행하는 단계를 구비하고,

상기 전용 하드웨어의 정보들은, 상기 뉴럴 네트워크 모델이 실행되기 전의 스태틱 정보로서 컴퓨팅 리소스(Computing resource) 정보 및 상기 뉴럴 네트워크 모델의 런타임 도중에 발생되는 다이나믹 정보로서 컴퓨팅 컨텍스트(Computing context) 정보 중 적어도 하나를 포함하는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 2

제1항에 있어서,

상기 파싱된 정보들은 상기 뉴럴 네트워크 모델에 적용된 커널의 사이즈에 관련된 정보를 포함하고,

상기 변형된 뉴럴 네트워크 모델은, 상기 뉴럴 네트워크 모델의 상기 커널의 사이즈를 변경함에 의해 생성되는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 3

삭제

청구항 4

제1항에 있어서,

상기 컴퓨팅 리소스 정보는, 하나 이상의 하드웨어들의 연산 방식 정보, 커널 구조 정보, 데이터 플로우 정보 및 데이터 재사용 정보들 중 적어도 하나를 포함하는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 5

제1항에 있어서,

상기 컴퓨팅 컨텍스트 정보는, 상기 런타임 도중 뉴럴 네트워크 모델의 실행과 관련된 정보와, 상기 런타임 도중 하나 이상의 하드웨어들에 대한 컴퓨팅 리소스 상태 변동 정보와 및 실행되는 어플리케이션 정보를 포함하는 그룹에서 선택된 적어도 하나의 정보를 포함하는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 6

뉴럴 네트워크 시스템의 동작방법에 있어서,

프로세서의 제어에 기반하여, 뉴럴 네트워크 모델로부터 뉴럴 네트워크 연산에 관련된 하나 이상의 정보들을 파싱하는 단계;

상기 프로세서의 제어에 기반하여, 상기 뉴럴 네트워크 시스템이 채용된 전자 시스템의 사용자에게 사용자 인터페이스를 제공하는 단계;

상기 사용자 인터페이스를 통해 하나 이상의 사용자 선택 정보를 수신하는 단계;

상기 프로세서의 제어에 기반하여, 상기 뉴럴 네트워크 모델을 실행할 적어도 하나의 전용 하드웨어의 정보들을 판단하는 단계;

상기 프로세서의 제어에 기반하여, 상기 전용 하드웨어의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하는 단계; 및

상기 전용 하드웨어에 의해 상기 변형된 뉴럴 네트워크 모델을 실행하는 단계를 구비하고,

상기 변형된 뉴럴 네트워크 모델을 생성하는 단계는, 상기 사용자 선택 정보를 더 이용하여 상기 뉴럴 네트워크 모델의 정보를 변경하는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 7

제6항에 있어서,

상기 사용자 선택 정보는, 상기 뉴럴 네트워크 모델이 실행될 전용 하드웨어의 종류를 나타내는 정보를 포함하고,

상기 프로세서의 제어에 기반하여, 상기 변형된 뉴럴 네트워크 모델을 상기 사용자에 의해 선택된 전용 하드웨어로 할당하는 단계를 더 구비하는 것을 특징으로 하는 뉴럴 네트워크 시스템의 동작방법.

청구항 8

프로그램들을 저장하는 메모리;

상기 메모리에 저장된 프로그램들을 실행하는 프로세서; 및

상기 메모리에 로딩 가능한 프로그램들을 포함하는 뉴럴 네트워크 어댑터 모듈을 구비하고,

상기 프로세서는, 상기 뉴럴 네트워크 어댑터 모듈을 실행함으로써, 뉴럴 네트워크 모델을 실행할 하나 이상의 전용 하드웨어들의 정보들을 판단하고, 상기 전용 하드웨어들의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하며,

상기 하나 이상의 전용 하드웨어들의 정보들은, 상기 뉴럴 네트워크 모델이 실행되기 전의 스태틱 정보로서 컴퓨팅 리소스(Computing resource) 정보 및 상기 뉴럴 네트워크 모델의 런타임 도중에 발생되는 다이나믹 정보로서 컴퓨팅 컨텍스트(Computing context) 정보 중 적어도 하나를 포함하는 것을 특징으로 하는 어플리케이션 프로세서.

청구항 9

제8항에 있어서,

상기 프로세서는, 상기 뉴럴 네트워크 어댑터 모듈을 실행함으로써, 상기 뉴럴 네트워크 모델로부터 뉴럴 네트워크 연산에 관련된 하나 이상의 정보들을 파싱하고, 상기 파싱된 정보들 중 적어도 하나의 정보를 변경함으로써 상기 변형된 뉴럴 네트워크 모델을 생성하는 것을 특징으로 하는 어플리케이션 프로세서.

청구항 10

제9항에 있어서,

상기 파싱된 정보들은 상기 뉴럴 네트워크 모델의 컨벌루션 연산에 관련된 커널 사이즈 정보를 포함하고,

상기 프로세서는, 상기 뉴럴 네트워크 어댑터 모듈을 실행함으로써, 상기 뉴럴 네트워크 모델이 할당될 전용 하드웨어에 따라 상기 커널 사이즈를 변경함으로써 상기 변형된 뉴럴 네트워크 모델을 생성하는 것을 특징으로 하는 어플리케이션 프로세서.

발명의 설명

기 술 분 야

[0001] 본 개시의 기술적 사상은 뉴럴 네트워크 시스템에 관한 것으로서, 상세하게는 뉴럴 네트워크 모델을 변형하는

뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법에 관한 것이다.

배 경 기 술

[0002] 뉴럴 네트워크(neural network)는 생물학적 뇌를 모델링한 컴퓨터 과학적 아키텍쳐(computational architecture)를 참조한다. 최근 뉴럴 네트워크(neural network) 기술이 발전함에 따라, 다양한 종류의 전자 시스템에서 하나 이상의 뉴럴 네트워크 모델을 이용한 뉴럴 네트워크 장치를 사용하여 입력 데이터를 분석하고 유효한 정보를 추출하는 연구가 활발히 진행되고 있다.

[0003] 딥 러닝 알고리즘을 포함하는 다양한 종류의 뉴럴 네트워크 모델들이 개발되고, 이러한 모델들은 다양한 종류의 하드웨어들에 의해 실행될 수 있다. 그러나, 뉴럴 네트워크 모델의 특성(예컨대, 웨이트를 이용하는 연산 특성)이 하드웨어에 최적화되지 않는 경우에는 하드웨어 리소스를 낭비하거나 연산 속도가 낮아지는 문제가 발생될 수 있다.

발명의 내용

해결하려는 과제

[0004] 본 발명의 기술적 사상이 해결하려는 과제는, 뉴럴 네트워크 모델의 연산 효율을 향상할 수 있는 뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법을 제공하는 데 있다.

과제의 해결 수단

[0005] 상기와 같은 목적을 달성하기 위하여, 본 개시의 기술적 사상의 일측면에 따른 뉴럴 네트워크 시스템의 동작방법은, 프로세서의 제어에 기반하여, 뉴럴 네트워크 모델로부터 뉴럴 네트워크 연산에 관련된 하나 이상의 정보들을 파싱하는 단계와, 상기 프로세서의 제어에 기반하여, 상기 뉴럴 네트워크 모델을 실행할 적어도 하나의 전용 하드웨어의 정보를 판단하는 단계와, 상기 프로세서의 제어에 기반하여, 상기 전용 하드웨어의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하는 단계 및 상기 전용 하드웨어에 의해 상기 변형된 뉴럴 네트워크 모델을 실행하는 단계를 구비하는 것을 특징으로 한다.

[0006] 한편, 본 개시의 기술적 사상의 일측면에 따른 어플리케이션 프로세서는, 프로그램들을 저장하는 메모리와, 상기 메모리에 저장된 프로그램들을 실행하는 프로세서 및 상기 메모리에 로딩 가능한 프로그램들을 포함하는 뉴럴 네트워크 어댑터 모듈을 구비하고, 상기 프로세서는, 상기 뉴럴 네트워크 어댑터 모듈을 실행함으로써, 뉴럴 네트워크 모델을 실행할 하나 이상의 전용 하드웨어들의 정보들을 판단하고, 상기 전용 하드웨어들의 정보들을 판단한 결과에 따라 상기 뉴럴 네트워크 모델의 정보를 변경함으로써 변형된(Reshaped) 뉴럴 네트워크 모델을 생성하는 것을 특징으로 한다.

[0007] 한편, 본 개시의 기술적 사상의 일측면에 따른 뉴럴 네트워크 시스템은, 제1 뉴럴 네트워크 모델을 수신하고, 상기 제1 뉴럴 네트워크 모델로부터 파싱된 정보 및 제1 전용 하드웨어의 정보를 판단함에 기반하여 상기 제1 뉴럴 네트워크 모델을 변형함으로써, 상기 제1 전용 하드웨어로 할당될 제2 뉴럴 네트워크 모델을 생성하는 뉴럴 네트워크 어댑터 모듈 및 상기 제1 전용 하드웨어를 포함하고, 입력 데이터에 대해 상기 제2 뉴럴 네트워크 모델에 따른 연산을 수행하여 정보 신호를 생성하는 뉴럴 네트워트 장치를 구비하는 것을 특징으로 한다.

발명의 효과

[0008] 본 발명의 기술적 사상의 뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법에 따르면, 다양한 종류의 뉴럴 네트워크 모델을 특정한 전용 하드웨어에 최적화시킬 수 있으므로 뉴럴 네트워크 모델의 연산 효율을 향상할 수 있는 효과가 있다.

[0009] 또한, 본 발명의 기술적 사상의 뉴럴 네트워크 시스템, 이를 포함하는 어플리케이션 프로세서 및 뉴럴 네트워크 시스템의 동작방법에 따르면, 특정한 뉴럴 네트워크 모델을 다양한 종류의 전용 하드웨어들에 최적화시킬 수 있으므로 뉴럴 네트워크 모델의 연산 효율을 향상할 수 있는 효과가 있다.

도면의 간단한 설명

[0010] 도 1은 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈을 포함하는 전자 시스템을 나타내는 블록도이다.

도 2는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈의 일 동작 예를 나타내는 블록도이다.

도 3은 컨벌루션 뉴럴 네트워크 모델의 일 예를 나타내는 도면이다.

도 4 및 도 5는 본 발명의 예시적인 실시예들에 따른 뉴럴 네트워크 어댑터 모듈의 일 구현 예를 나타내는 블록도이다.

도 6a,b,c는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈의 동작 개념을 나타내는 블록도이다.

도 7 내지 도 10은 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 모델 어댑터의 동작방법을 나타내는 플로우차트이다.

도 11은 커널 사이즈를 조절함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다.

도 12는 재사용 방식을 변경함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다.

도 13은 성능을 조절함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다.

도 14a,b,c는 하드웨어의 연산 특성에 따라 뉴럴 네트워크 모델이 변형되는 예를 나타내는 도면이다.

도 15는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈이 소프트웨어적으로 구현되는 예를 나타내는 블록도이다.

도 16은 뉴럴 네트워크 어댑터 모듈이 자율 운행 모듈 내에 구현되는 예를 나타내는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0011] 이하, 첨부한 도면을 참조하여 본 발명의 실시 예에 대해 상세히 설명한다.

[0012] 도 1은 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈을 포함하는 전자 시스템을 나타내는 블록도이다. 일 실시예에 따라, 도 1의 전자 시스템(100)은 뉴럴 네트워크를 기초로 입력 데이터를 실시간으로 분석하여 유효한 정보를 추출하고, 추출된 정보를 기초로 상황 판단을 하거나 또는 전자 시스템(100)에 탑재되는 전자 장치의 구성들을 제어할 수 있다.

[0013] 도 1의 전자 시스템(100)은 모바일 장치에 채용되는 어플리케이션 프로세서(Application Processor, AP)일 수 있다. 또는, 도 1의 전자 시스템(100)은 컴퓨팅 시스템에 해당하거나, 드론(drone), 첨단 운전자 보조 시스템(Advanced Drivers Assistance System; ADAS) 등과 같은 로봇 장치, 스마트 TV, 스마트 폰, 의료 장치, 모바일 장치, 영상 표시 장치, 계측 장치, IoT(Internet of Things) 장치 등에 해당될 수 있으며, 이 외에도 본 발명의 뉴럴 네트워크 어댑터 모듈은 다양한 시스템들에 적용될 수 있다. 이하에서는, 도 1의 전자 시스템(100)이 어플리케이션 프로세서(AP)에 해당하는 것으로 가정된다.

[0014] 도 1을 참조하면, 전자 시스템(100)은 프로세서(110), RAM(Random Access memory, 120), 뉴럴 네트워크 장치(130), 뉴럴 네트워크 어댑터 모듈(140) 및 메모리(150)를 포함할 수 있다. 일 실시예에 있어서, 전자 시스템(100)의 구성들 중 적어도 일부는 하나의 반도체 칩에 탑재될 수 있다.

[0015] 전자 시스템(100)이 뉴럴 네트워크 연산 기능을 수행하는 점에서, 전자 시스템(100)은 뉴럴 네트워크 시스템(NNS)을 포함하는 것으로 정의될 수 있다. 뉴럴 네트워크 시스템(NNS)은 뉴럴 네트워크 동작과 관련하여, 전자 시스템(100)에 구비되는 구성들 중 적어도 일부를 포함할 수 있다. 일 예로서, 도 1에는 뉴럴 네트워크 시스템(NNS)이 프로세서(110), 뉴럴 네트워크 장치(130) 및 뉴럴 네트워크 어댑터 모듈(140)을 포함하는 것으로 예시되었으나 본 발명의 실시예는 이에 국한될 필요가 없다. 예컨대, 뉴럴 네트워크 동작에 관여되는 다른 다양한 종류의 구성들이 뉴럴 네트워크 시스템(NNS)에 구비되는 것으로 정의되어도 무방할 것이다.

[0016] 프로세서(110)는 전자 시스템(100)의 전반적인 동작을 제어한다. 프로세서(110)는 하나의 프로세서 코어(Single Core)를 포함하거나, 복수의 프로세서 코어들(Multi-Core)을 포함할 수 있다. 프로세서(110)는 메모리(140)에 저장된 프로그램들 및/또는 데이터를 처리 또는 실행할 수 있다. 일 실시예에 있어서, CPU(110)는 메모리(140)에 저장된 프로그램들을 실행함으로써, 뉴럴 네트워크 장치(130)의 기능을 제어할 수 있다.

[0017] RAM(120)은 프로그램들, 데이터, 또는 명령들(instructions)을 일시적으로 저장할 수 있다. 예컨대 메모리(140)에 저장된 프로그램들 및/또는 데이터는 프로세서(110)의 제어 또는 부팅 코드에 따라 RAM(120)에 일시적으로 저장될 수 있다. RAM(120)은 DRAM(Dynamic RAM) 또는 SRAM(Static RAM) 등의 메모리로 구현될 수 있다.

[0018] 뉴럴 네트워크 장치(130)는 수신되는 입력 데이터를 기초로 뉴럴 네트워크 의 연산을 수행하고, 수행 결과를 기초로 정보 신호를 생성할 수 있다. 뉴럴 네트워크의 모델들은 Convolutional Neural Networks(CNN), Recurrent Neural Networks(RNN), Deep Belief Networks, Restricted Boltzman Machines 등 다양한 종류의 모델들을 포함할 수 있으나 이에 제한되지 않는다. 뉴럴 네트워크 장치(130)는 뉴럴 네트워크의 모델들에 따른 연산을 수행하기 위한 하나 이상의 프로세서(예컨대, 전용 프로세서)를 포함할 수 있다. 또한, 뉴럴 네트워크 장치(130)는 뉴럴 네트워크의 모델들에 대응하는 프로그램들을 저장하기 위한 별도의 메모리(미도시)를 포함할 수도 있다.

[0019] 정보 신호는 음성 인식 신호, 사물 인식 신호, 영상 인식 신호, 생체 정보 인식 신호 등과 같은 다양한 종류의 인식 신호 중 하나를 포함할 수 있다. 예를 들어, 뉴럴 네트워크 장치(130)는 비디오 스트림에 포함되는 프레임 데이터를 입력 데이터로서 수신하고, 프레임 데이터로부터 프레임 데이터가 나타내는 이미지에 포함된 사물에 대한 인식 신호를 생성할 수 있다. 그러나, 이에 제한되는 것은 아니며, 전자 시스템(100)이 탑재된 전자 장치의 종류 또는 기능에 따라 뉴럴 네트워크 장치(130)는 다양한 종류의 입력 데이터를 수신할 수 있고, 입력 데이터에 따른 정보 신호를 생성할 수 있다.

[0020] 뉴럴 네트워크 모델은 일반(general) 하드웨어나 특정 소프트웨어에 최적화된 전용 하드웨어 등 다양한 종류의 하드웨어에 의해 실행될 수 있다. 일 예로, CPU나 GPU 등의 일반 하드웨어에 의해 다양한 종류의 뉴럴 네트워크 모델들이 실행될 수 있으나, 전용 하드웨어에 대비하여 성능이나 전력 소모에서 단점이 발생될 수 있다. 또한, 전용 하드웨어(예컨대, ASIC, NPU(neural processing unit), TPU(Tensor Processing Unit), Neural Engine 등)에 의해 특정 뉴럴 네트워크 모델이 실행되면 성능 및 전력 소모 측면에서 유리하나, 이는 다양한 종류의 뉴럴 네트워크 모델을 채용하는 데에 단점이 발생될 수 있다.

[0021] 일 실시예에 따라, 뉴럴 네트워크 어댑터 모듈(140)은 입력 뉴럴 네트워크 모델을 이용하여 변형된(Reshaped) 뉴럴 네트워크 모델을 생성할 수 있다. 예컨대, 뉴럴 네트워크 어댑터 모듈(140)은 다양한 종류의 입력 뉴럴 네트워크 모델들을 수신하고, 이에 대해 특정한 전용 하드웨어에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 또한, 네트워크 어댑터 모듈(140)은 어느 하나의 입력 뉴럴 네트워크 모델을 수신하고, 이에 대해 다양한 종류의 전용 하드웨어들에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 즉, 네트워크 어댑터 모듈(140)은 다양한 종류의 뉴럴 네트워크 모델들이 전용 하드웨어에 최적화되어 실행될 수 있도록 뉴럴 네트워크 모델에 대한 변형(Reshape) 동작을 수행할 수 있다.

[0022] 일 동작 예로서, 뉴럴 네트워크 동작은 웨이트(weight)를 이용한 컨벌루션 연산 등 다양한 종류의 연산들을 포함할 수 있다. 본 발명의 예시적인 실시예에 따라, 변형된 뉴럴 네트워크 모델은 입력 뉴럴 네트워크 모델과 비교하여 모델 컨텐츠(예컨대, 웨이트 값)은 동일하거나 유사한 값을 가질 수 있으며, 이에 따라 입력 뉴럴 네트워크 모델을 이용한 연산 결과와 변형된 뉴럴 네트워크 모델을 이용한 연산 결과는 실질적으로 동일하거나 유사할 수 있다. 반면에, 변형된 뉴럴 네트워크 모델에서는 상기 컨텐츠를 적용하는 방식 또는 컨텐츠를 이용한 연산 방식 등에서 입력 뉴럴 네트워크 모델과 차별될 수 있다.

[0023] 상기와 같은 본 발명의 예시적인 실시예에 따르면, 뉴럴 네트워크 연산을 수행하는 전용 하드웨어가 다양한 종류의 뉴럴 네트워크 모델들에 최적화되도록 할 수 있다. 일 예로서, 전자 시스템(100) 내에서 다양한 종류의 어플리케이션들이 실행될 수 있으며, 각각의 어플리케이션의 실행에는 다양한 종류의 뉴럴 네트워크 모델이 이용될 수 있다. 이 때, 상기 다양한 종류의 뉴럴 네트워크 모델들에 대해 뉴럴 네트워크 어댑터 모듈(140)을 통해 전용 하드웨어에 최적화된 변형된 뉴럴 네트워크 모델이 생성될 수 있으며, 이에 따라 전용 하드웨어를 통한 향상된 성능 및 전력 소모를 최소화하여 뉴럴 네트워크를 실행할 수 있다. 또한, 다양한 종류의 어플리케이션들에 대해 서로 다른 종류의 뉴럴 네트워크 모델이 이용될 수 있으며, 이와 같은 경우에도 전용 하드웨어를 통한 향상된 성능 및 전력 소모를 최소화하여 뉴럴 네트워크를 실행할 수 있다.

[0024] 한편, 뉴럴 네트워크 어댑터 모듈(140)은 다양한 형태로 구현될 수 있으며, 일 실시예에 따라 뉴럴 네트워크 어댑터 모듈(140)은 하드웨어 형태로 구현되거나 또는 소프트웨어 형태로 구현될 수 있다. 뉴럴 네트워크 어댑터 모듈(140)이 하드웨어 형태로 구현되는 경우, 뉴럴 네트워크 어댑터 모듈(140)은 입력 뉴럴 네트워크 모델로부터 파싱된 하나 이상의 정보에 대응하는 입력 신호를 수신하고, 수신된 입력 신호를 하나 이상의 회로를 포함하는 하드웨어를 이용하여 변경하며, 이를 통해 변형된(Reshaped) 뉴럴 네트워크 모델을 생성할 수 있다. 또는, 뉴럴 네트워크 어댑터 모듈(140)은 운영 체제(OS)나 그 하위 단에 소프트웨어 형태로 구현될 수 있으며,

RAM(120)에 로딩되어 프로세서(110)에 의해 실행됨으로써 상기 입력 뉴럴 네트워크 모델의 하나 이상의 정보가 소프트웨어적으로 변경될 수 있다. 즉, 전술한 모델 변형 동작은 프로세서(110)가 뉴럴 네트워크 어댑터 모듈(140)을 실행함으로써 수행되는 것으로 설명될 수 있을 것이다.

[0025] 한편, 도 1에 도시된 전자 시스템(100)이 어플리케이션 프로세서에 해당하는 경우, 전자 시스템(100)이 채용된 시스템은 뉴럴 네트워크 모델을 실행하기 위한 추가의 전용 하드웨어를 더 포함할 수도 있다. 이 때, 전자 시스템(100)에서 생성된 변형된 뉴럴 네트워크 모델은, 그 내부의 전용 하드웨어에 의해 실행될 수도 있을 것이며, 또는 전자 시스템(100)의 외부에 배치되는 전용 하드웨어에 의해 실행될 수도 있을 것이다.

[0026] 이하, 본 발명의 예시적인 실시예들에 따른 뉴럴 네트워크 모델의 변형 예들을 설명함에 있어서, 하드웨어로 지칭되는 구성은 일반 하드웨어 및 전용 하드웨어를 포함하는 개념일 수 있다. 바람직하게는, 변형된 뉴럴 네트워크 모델을 실행하는 하드웨어는 전용 하드웨어에 해당할 수 있을 것이다.

[0027] 도 2는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈의 일 동작 예를 나타내는 블록도이다.

[0028] 도 1 및 도 2를 참조하면, 뉴럴 네트워크 어댑터 모듈(140)은 입력 뉴럴 네트워크 모델(DL Model)을 수신하고 이에 대한 변형 동작을 통해 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 도 2에서는 하나의 동일한 입력 뉴럴 네트워크 모델(DL Model)에 대해 N 개의 뉴럴 네트워크 모델(N variable DL Model)이 생성되는 예가 도시된다.

[0029] 뉴럴 네트워크 어댑터 모듈(140)은 변형 모듈(또는, 변형기(141))을 포함할 수 있다. 변형 모듈(141)은 입력 뉴럴 네트워크 모델(DL Model) 및 각종 정보들을 이용하여 변형 동작을 수행하고, 그 결과에 기반하여 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 실시예에 따라, 뉴럴 네트워크 어댑터 모듈(140)은 스태틱 정보(Static Info.) 및 다이나믹 정보(Dynamic Info.) 중 적어도 하나를 수신하고, 변형 모듈(141)은 수신된 스태틱 정보(Static Info.) 및 다이나믹 정보(Dynamic Info.)를 이용하여 입력 뉴럴 네트워크 모델(DL Model)을 변형할 수 있다. 즉, 입력 뉴럴 네트워크 모델(DL Model)이 하드웨어의 용량, 최적화된 연산 방식 등을 고려하여 변형될 수 있다.

[0030] 스태틱 정보(Static Info.) 및 다이나믹 정보(Dynamic Info.) 각각은 다양한 종류의 정보들을 포함할 수 있다. 일 예로서, 스태틱 정보(Static Info.)는 전자 시스템(100) 내의 각종 구성 요소들의 기본 정보들을 포함할 수 있으며, 일 예로 뉴럴 네트워크 모델(또는, 뉴럴 네트워크 알고리즘)을 실행하는 하드웨어의 성능 및 특성 등의 컴퓨팅 리소스(Computing resource) 정보를 포함할 수 있다. 또한, 다이나믹 정보(Dynamic Info.)는 뉴럴 네트워크 모델을 실행하는 도중에 발생될 수 있는 각종 정보들을 포함하며, 일 예로서 런타임(runtime) 과정에서의 컴퓨팅 컨텍스트(Computing context) 정보를 포함할 수 있다.

[0031] 스태틱 정보(Static Info.)를 이용한 뉴럴 네트워크 모델의 변형은 오프라인 모드에서 수행될 수 있다. 오프라인 모드에서의 변형은 실제 뉴럴 네트워크 모델이 실행되기 전의 임의의 시점들에서의 변형에 해당할 수 있다. 일 예로서, 전자 시스템(100)의 동작 전, 또는 뉴럴 네트워크 동작이 수행되기 전에 뉴럴 네트워크 어댑터 모듈(140)은 하나 이상의 입력 뉴럴 네트워크 모델을 수신하고, 상기 스태틱 정보(Static Info.)에 기반하여 변형 뉴럴 네트워크 모델을 생성하여 이를 전자 시스템(100) 내부에 저장할 수 있다.

[0032] 또는, 뉴럴 네트워크 모델의 변형은 온라인 모드에서 수행될 수 있으며, 이 때 런타임(runtime) 도중 동적으로 발생되는 다이나믹 정보(Dynamic Info.)를 이용하여 뉴럴 네트워크 모델이 실시간으로 변형될 수 있다. 일 예로서, 전자 시스템(100) 내에서 어플리케이션이 구동됨에 따라 하나 이상의 뉴럴 네트워크 모델이 실행될 수 있으며, 뉴럴 네트워크 모델의 실행 과정에서 발생되는 다이나믹 정보(Dynamic Info.)에 기반하여 뉴럴 네트워크 모델의 처리 방식을 변경하기 위한 변형된 뉴럴 네트워크 모델이 생성될 수 있다.

[0033] 도 3은 본 발명에 적용 가능한 뉴럴 네트워크의 모델의 일 예로서, 컨벌루션 뉴럴 네트워크(Convolutional Neural Networks, CNN) 모델을 나타내는 도면이다. 일 실시예에 따라, 도 3에서는 컨벌루션 뉴럴 네트워크 모델의 다양한 레이어들 중 컨벌루션 레이어가 도시되었으나, 컨벌루션 뉴럴 네트워크 모델은 풀링 레이어(pooling layer) 및 풀리 커넥티드(fully connected) 레이어 등을 더 포함할 수 있을 것이다.

[0034] 컨벌루션 레이어(CL)에서, 제 1 피처 맵(FM1)은 입력 피처 맵이 될 수 있고, 제 2 피처 맵(FM2)는 출력 피처 맵이 될 수 있다. 피처 맵은 입력 데이터의 다양한 특징이 표현된 데이터를 의미한다. 피처 맵들(FM1, FM2)은 2차원 매트릭스 또는 3차원 매트릭스 형태를 가질 수 있다. 이러한 다차원 매트릭스 형태를 가지는 피처 맵들(FM1, FM2)은 피처 텐서(tensor)로 지칭될 수 있다. 또한, 입력 피처 맵은 액티베이션(activation)으로 지칭될 수 있다. 피처 맵들(FM1, FM2)은 너비(W)(또는 칼럼이라고 함), 높이(H)(또는 로우라고 함) 및 깊이(D)를 가지며, 이

는 좌표상의 x축, y축 및 z축에 각각 대응할 수 있다. 이때, 깊이(D)는 채널 수로 지칭될 수 있다.

[0035] 컨벌루션 레이어(CL)에서, 제1 피처 맵(FM1) 및 웨이트 맵(WM)에 대한 컨벌루션 연산이 수행될 수 있고, 그 결과 제2 피처 맵(FM2)이 생성될 수 있다. 웨이트 맵(WM)은 제1 피처 맵(FM1)을 필터링할 수 있으며, 필터 또는 커널로 지칭될 수 있다. 웨이트 맵(WM)의 깊이, 즉 채널 개수는 제1 피처 맵(FM1)의 깊이, 즉 채널 개수와 동일하며, 웨이트 맵(WM)과 제1 피처 맵(FM1)의 동일한 채널끼리 컨벌루션될 수 있다. 웨이트 맵(WM)이 제1 입력 피처 맵(FM1)을 슬라이딩 윈도로 하여 횡단하는 방식으로 시프트된다. 시프트되는 양은 스트라이드(stride) 길이 또는 스트라이드로 지칭될 수 있다. 각 시프트동안, 웨이트 맵(WM)에 포함되는 웨이트 각각이 제1 피처 맵(FM1)과 중첩된 영역에서의 모든 피처값과 곱해지고 더해질 수 있다.

[0036] 제1 피처 맵(FM1)과 웨이트 맵(WM)이 컨벌루션 됨에 따라, 제2 피처 맵(FM2)의 하나의 채널이 생성될 수 있다. 도 1에는 하나의 웨이트 맵(WM)이 표시되었으나, 실질적으로는 복수개의 웨이트 맵이 제1 피처 맵(FM1)과 컨벌루션 되어, 제2 피처 맵(FM2)의 복수개의 채널이 생성될 수 있다. 다시 말해, 제2 피처 맵(FM2)의 채널의 수는 웨이트 맵의 개수에 대응할 수 있다.

[0037] 또한, 컨벌루션 레이어(10)의 제 2 피처 맵(FM2)은 다른 레이어의 입력 피처 맵이 될 수 있다. 예를 들어, 제 2 피처 맵(FM2)는 풀링(pooling) 레이어의 입력 피처 맵이 될 수 있다.

[0038] 본 발명의 예시적인 실시예에 따라, 입력 뉴럴 네트워크 모델을 변형함에 있어서 상기와 같은 웨이트 맵(WM)이 변형될 수 있다. 일 예로, 입력 뉴럴 네트워크 모델에서는 N*N 사이즈의 웨이트 맵(WM)을 기초로 컨벌루션 연산이 수행되나, 변형된 뉴럴 네트워크 모델에서는 변형된 사이즈를 갖는 웨이트 맵(WM)을 기초로 컨벌루션 연산이 수행될 수 있다. 이 때, N*N 사이즈의 웨이트 맵(WM)은 하나 또는 두 개 이상의 웨이트 맵(WM)으로 변경될 수 있으며, 일 예로 웨이트 맵(WM)은 M*M 사이즈, N*1, 1*N 사이즈, M*1, 1*M 사이즈 등 다양한 형태로 변경될 수 있다. 또한, 전술한 바와 같이, 변형되기 전과 변형된 후의 웨이트 맵(WM)은 그 컨텐츠 자체는 동일 또는 유사할 수 있다.

[0039] 도 4 및 도 5는 본 발명의 예시적인 실시예들에 따른 뉴럴 네트워크 어댑터 모듈의 일 구현 예를 나타내는 블록도이다. 도 4에서는 오프라인 모드에서의 뉴럴 네트워크 모델 변형을 위한 구현 예가 도시되고, 도 5에서는 온라인 모드에서의 뉴럴 네트워크 모델 변형을 위한 구현 예가 도시된다.

[0040] 도 4를 참조하면, 뉴럴 네트워크 어댑터 모듈(200)은 모델 파서(210), 변형 모듈(220) 및 컴퓨팅 리소스 모듈(230)을 포함할 수 있다. 일 실시예에 따라, 상기 뉴럴 네트워크 어댑터 모듈(200)은 컴파일러(compiler)에 포함되는 구성일 수 있으며, 뉴럴 네트워크 어댑터 모듈(200)이 전용 하드웨어(예컨대, NPU)에 최적화된 변형된 뉴럴 네트워크 모델을 생성하는 경우, 뉴럴 네트워크 어댑터 모듈(200)은 NPU 컴파일러에 구비되는 구성일 수 있다. NPU 컴파일러는 NPU에 적합하도록 명령어를 컴파일링할 수 있으며, 이 때 뉴럴 네트워크 어댑터 모듈(200)에 의해 NPU에 최적화된 변형된 뉴럴 네트워크 모델이 생성될 수 있다.

[0041] 모델 파서(210)는 입력 뉴럴 네트워크 모델로부터 각종 정보를 파싱할 수 있다. 일 예로서, 모델 파서(210)는 입력 뉴럴 네트워크 모델로부터 뎁스(depth) 및 브랜치(branch) 등의 레이어 토폴로지(Layer topology), 압축 방법에 관련된 정보, 각각의 레이어에서의 연산 타입에 관련된 정보, 포맷(format), 보안(security) 및 사이즈 등의 데이터 특성(Data property) 정보, 입력, 커널/필터, 출력 등의 피연산자(operand)를 위한 메모리 레이아웃(memory layout) 정보, 데이터 압축 방법 정보 등의 다양한 정보들을 파싱할 수 있다. 상기 커널/필터는 전술한 웨이트에 해당할 수 있으며, 메모리 레이아웃 정보는 패딩(padding) 및 스트라이드(stride) 등의 정보를 포함할 수 있다.

[0042] 컴퓨팅 리소스 모듈(230)은 전술한 스태틱 정보로서 다양한 종류의 정보들을 판단할 수 있다. 일 예로서, 컴퓨팅 리소스 모듈(230)은 더 나은 가속(acceleration)을 위한 하드웨어(예컨대, 전용 하드웨어)에 대한 다양한 정보들로서, 컨벌루션/가산/최대값 등의 연산 방식 정보, 3*3, 5*5 등의 커널 구조 정보, 데이터 플로우(data flow) 정보 및 데이터 재사용(reuse) 방식 정보 등 다양한 종류의 정보들을 판단할 수 있다. 또한, 컴퓨팅 리소스 모듈(230)은 하드웨어에 대한 다른 정보들로서, 하드웨어의 성능 및 파워 소모 등의 용량(Capacity)에 관련된 정보나, 지지되지 않는 데이터 타입(unsupported data type)이나 데이터 레이아웃, 압축, 양자화 알고리즘(quantization algorithm) 등의 하드웨어의 한계(Limitation) 정보를 판단할 수 있다.

[0043] 변형 모듈(220)은 상기와 같은 정보들에 기반하여 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 실시예에 따라, 변형 모듈(220)은 모델 파서(210)로부터의 정보에 기반하여 입력 뉴럴 네트워크 모델의 특성(예컨대, 연산 특성)을 판단하고, 컴퓨팅 리소스 모듈(230)로부터의 정보에 기반하여 모델 파서(210)로부터 파싱된 정보들

중 적어도 하나를 변경하는 방식에 따라 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 실시예에 따라, 변형 모듈(220)은 하드웨어가 더 나은 가속으로서 뉴럴 네트워크 모델을 실행할 수 있도록, 입력 뉴럴 네트워크 모델의 적어도 하나의 정보를 변경할 수 있다. 일 예로서, 하드웨어가 특정한 사이즈의 커널을 이용함으로써 최적화된 연산 처리를 수행하는 경우, 변형 모듈(220)은 모델 파서(210)로부터 파싱된 커널 구조를 변경함으로써 전술한 변형된 뉴럴 네트워크 모델을 생성할 수 있다.

[0044] 한편, 도 5를 참조하면, 뉴럴 네트워크 어댑터 모듈(300)은 모델 파서(210), 유저 API(320), 변형 모듈(330), 컨텍스트 매니저(340), 컴퓨팅 리소스 모듈(350) 및 컴퓨팅 AL(abstract layer, 360)을 포함할 수 있다. 도 5에는 뉴럴 네트워크 어댑터 모듈(300)과 함께, 다양한 종류의 뉴럴 네트워크 모델들을 실행하는 각종 하드웨어들을 포함하는 하드웨어 블록(301)이 더 도시된다. 상기 하드웨어 블록(301)은 CPU, GPU, DSP(Digital Signal Processor), FPGA(Field Programmable Gate Array) 및 NPU 등 다양한 종류의 하드웨어들을 포함할 수 있다. 이 때, 상기 하드웨어 블록(301)에서 일부의 하드웨어들은 일반 하드웨어이고, 다른 일부의 하드웨어들은 특정 뉴럴 네트워크 모델에 최적화된 전용 하드웨어일 수 있다.

[0045] 일 실시예에 따라, 상기 뉴럴 네트워크 어댑터 모듈(300)은 다양한 형태로 구현될 수 있다. 예컨대, 전자 시스템 내에 NPU 및 GPU 등 다양한 하드웨어들은 장치 드라이버를 통해 뉴럴 네트워크 프레임워크(Framework)에 접근하여 소프트웨어를 실행할 수 있으며, 뉴럴 네트워크 어댑터 모듈(300)은 뉴럴 네트워크 프레임워크(Framework) 내에 구비되는 구성일 수 있다. 또한, 뉴럴 네트워크 어댑터 모듈(300)은 변형된 뉴럴 네트워크 모델을 장치 드라이버 및/또는 펌웨어를 통해서 하나 이상의 하드웨어로 제공할 수 있다.

[0046] 모델 파서(310)는 전술한 실시예에서와 동일 또는 유사하게 입력 뉴럴 네트워크 모델로부터 각종 정보를 파싱할 수 있다. 일 예로서, 뉴럴 네트워크 어댑터 모듈(300)이 온라인 모드에서 동작함에 따라, 모델 파서(310)는 현재 실행중인 뉴럴 네트워크 모델로부터 각종 정보들을 파싱할 수 있다.

[0047] 일 실시예에 따라, 유저 API(320)는 변형된 뉴럴 네트워크 모델의 생성과 관련하여 사용자로부터의 선택 정보를 관리할 수 있다. 일 실시예에 따라, 유저 API(320)는 전자 시스템의 사용자에게 뉴럴 네트워크 모델의 변형에 관련된 선택 정보를 입력하기 위한 유저 인터페이스를 제공할 수 있으며, 사용자는 유저 인터페이스를 통해 각종 선택 정보를 입력할 수 있다. 일 예로서, 사용자가 뉴럴 네트워크 동작을 포함하는 어플리케이션을 실행할 때, 유저 API(320)의 제어에 기반하여 유저 인터페이스가 사용자에게 제공될 수 있다.

[0048] 뉴럴 네트워크 동작에 관련된 각종 정보가 유저 API(320)를 통해 뉴럴 네트워크 어댑터 모듈(300)로 제공될 수 있으며, 일 예로서 어플리케이션 구동시에 실행될 뉴럴 네트워크 모델의 종류(예컨대, CNN, RNN 등)가 선택 정보로서 입력될 수 있다. 또한, 사용자가 원하는 각종 선호(preference) 정보들로서, 커널 사이즈에 관련된 정보, 뉴럴 네트워크 처리 속도 및 파워 소모에 관련된 정보, 데이터 사이즈 정보 등이 선택 정보로서 입력될 수 있다.

[0049] 일 실시예에 따라, 유저 API(320)를 통해 입력되는 선택 정보에 따라 뉴럴 네트워크를 실행할 하드웨어를 설정하는 방식은 명시적 하드웨어 맞춤(fitting) 및 암묵적 하드웨어 맞춤 방식으로 분류될 수 있다. 예컨대, 명시적 하드웨어 맞춤 방식의 경우, 전자 시스템 내에 구비되는 다수의 하드웨어들 중 뉴럴 네트워크를 실행할 하드웨어와, 상기 하드웨어의 처리 속도 및 파워 소모 정도가 사용자에 의해 제공되는 선택 정보에 의해 설정될 수 있다. 반면에, 암묵적 하드웨어 맞춤 방식의 경우, 뉴럴 네트워크 동작과 관련된 다양한 종류의 선호 정보가 사용자에 의해 제공될 수 있는 반면에, 뉴럴 네트워크 모델을 실행할 하드웨어에 대한 설정은 사용자의 선택에 무관하게 전자 시스템 내에서 임의적으로 수행될 수 있다. 일 예로서, 암묵적 하드웨어 맞춤 방식에서, 뉴럴 네트워크를 실행할 하드웨어, 상기 하드웨어의 처리 속도 및 파워 소모 정도는 운영 체제(OS)에 의해 설정될 수 있다.

[0050] 한편, 컨텍스트 매니저(340)는 전술한 실시예에 따라 뉴럴 네트워크를 실행하는 과정에서 다이나믹 정보를 관리하고 이를 변형 모듈(330)로 제공할 수 있다. 런타임 도중 뉴럴 네트워크 실행에 관련된 각종 상태나 정보들이 컨텍스트 매니저(340)에 의해 관리될 수 있으며, 일 예로서 출력 정확도(Output accuracy), 레이턴시(latency) 및 영상 처리 속도(Frame Per Second (FPS)) 등에 관련된 정보나, 어플리케이션 실행시에 허용 가능한 정확도 손실(Loss)에 관련된 정보가 컨텍스트 매니저(340)를 통해 변형 모듈(330)로 제공될 수 있다. 상기와 같은 런타임 관련 다이나믹 정보와 함께, 리소스에 관련된 다이나믹 정보로서 컴퓨팅 리소스 상태의 변동, 파워/온도 정보, 버스(BUS)/메모리/스토리지 상태, 어플리케이션 종류 및 어플리케이션의 라이프 사이클(Lifecycle) 등의 각종 정보들이 컨텍스트 매니저(340)를 통해 변형 모듈(330)로 제공될 수 있다. 또한, 컴퓨팅 리소스 모듈(350)은 전술한 도 4의 실시예에서와 같이 스태틱 정보를 관리하고 이를 변형 모듈(330)로 제공할 수 있다. 즉, 본 발명

의 예시적인 실시예에서, 온라인 모드 방식의 뉴럴 네트워크 모델 변형 동작에서 컴퓨팅 리소스 모듈(350)에 의해 제공되는 스태틱 정보가 더 이용될 수도 있을 것이다.

[0051] 한편, 컴퓨팅 AL(abstract layer, 360)은 컴퓨팅 리소스를 관리하는 기능을 수행할 수 있으며, 변형 모듈(330)은 뉴럴 네트워크 어댑터 모듈(300) 내의 각종 구성 요소들로부터 제공되는 정보들에 기반하여 입력 뉴럴 네트워크 모델을 변경할 수 있다.

[0052] 입력 뉴럴 네트워크 모델 및 변형된 뉴럴 네트워크 모델은 하드웨어 블록(301)의 각종 하드웨어들로 제공될 수 있다. 일 예로서, 입력 뉴럴 네트워크 모델이 일반 하드웨어에 의해 실행될 때, 입력 뉴럴 네트워크 모델이 변형 없이 일반 하드웨어에 의해 실행될 수 있다. 또한, 런타임 도중 입력 뉴럴 네트워크 모델을 실행하는 하드웨어가 FPGA, NPU 등의 전용 하드웨어로 변경되는 경우, 뉴럴 네트워크 어댑터 모듈(300)에 의해 변형된 뉴럴 네트워크 모델이 전용 하드웨어에 의해 실행될 수 있다.

[0053] 전술한 바와 같이, 도 4 및 도 5의 뉴럴 네트워크 어댑터 모듈들(200, 300)은 온라인 및 오프라인 모드에서 변형된 뉴럴 네트워크 모델을 생성할 수 있으며, 상기와 같은 변형 동작은 함수로서 모델링될 수 있다. 일 예로, 상기 변형 동작은 입력 뉴럴 네트워크 모델, 하드웨어의 종류 및 하드웨어의 스태틱/다이나믹 정보, 사용자 API를 통한 선택 정보를 함수 입력으로서 수신하고, 변형된 뉴럴 네트워크 모델을 함수 결과로서 발생하는 동작으로 정의될 수 있다.

[0054] 도 6a,b,c는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈의 동작 개념을 나타내는 블록도이다. 도 6a,b,c에서는 입력 및 변형된 뉴럴 네트워크 모델로서 딥 뉴럴 네트워크(Deep Neural Network, DNN)가 예시된다.

[0055] 도 6a를 참조하면, 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 전용 하드웨어(Dedicated HW)로서 NPU에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 예로서, 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 다양한 종류의 입력 뉴럴 네트워크 모델(segmentation, NLU등)을 수신하고, 이에 대한 변형 과정을 거쳐 전용 하드웨어(NPU)에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 예로서, 입력 뉴럴 네트워크 모델(segmentation)은 화면 인식을 위한 뉴럴 네트워크 모델이고, 입력 뉴럴 네트워크 모델(NLU)은 음성 인식을 위한 뉴럴 네트워크 모델일 수 있다.

[0056] 한편, 도 6b를 참조하면, 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 다양한 종류의 전용 하드웨어(Dedicated HW)로서 NPU, FPGA, ASIC 등에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 일 예로, 특정한 입력 뉴럴 네트워크 모델(예컨대, segmentation등)이 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)로 제공되면, 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 실제 뉴럴 네트워크 처리가 진행될 전용 하드웨어의 종류에 따라 이에 적응적으로 모델 변형 동작을 수행할 수 있다. 일 예로서, 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 특정한 입력 뉴럴 네트워크 모델(segmentation)에 대한 변형 과정을 통해 NPU에 최적화된 변형된 뉴럴 네트워크 모델을 생성하거나, 또는 FPGA나 ASIC에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있다.

[0057] 한편, 도 6c는 뉴럴 네트워크 모델을 실행할 전용 하드웨어(NPU)가 두 배 이상의 연산을 처리할 수 있는 고성능 하드웨어인 경우에서의 동작 예를 나타낸다. 일 예로, 입력 뉴럴 네트워크 모델은 전용 하드웨어(NPU)에 최적화된 정보들을 포함하고, 이에 따라 모델 변형 과정에서 커널 사이즈 등의 연산 방식 변경은 적용되지 않을 수 있다. 뉴럴 네트워크 어댑터 모듈(DNN Adaptor)은 전용 하드웨어(NPU)의 성능을 고려하여 두 개 이상의 뉴럴 네트워크 모델이 실행되도록 변형된 뉴럴 네트워크 모델을 생성할 수 있으며, 일 예로서 인스턴스(instance) 개수를 변경함으로써 변형된 뉴럴 네트워크 모델을 생성할 수 있다.

[0058] 도 6a,b,c의 예에서는 뉴럴 네트워크 모델이 도형화되어 도시되었으며, 상기 도형의 형태는 다양한 정보들에 의해 결정될 수 있으며, 일 예로서 커널의 사이즈나 웨이트 또는 데이터의 비트 수 등이 상기 도형의 형태를 결정하는 정보에 해당할 수 있다.

[0059] 도 7 내지 도 10은 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 모델 어댑터의 동작방법을 나타내는 플로우차트이다.

[0060] 도 7을 참조하면, 오프라인 모드에서 뉴럴 네트워크 모델 어댑터는 실제 뉴럴 네트워크가 수행되기 전에 하드웨어에 관련된 스태틱 정보에 기반하여 전술한 뉴럴 네트워크 변경 기능을 수행할 수 있다. 뉴럴 네트워크가 수행되기 이전은 다양한 시기를 지칭할 수 있으며, 일 예로서 어플리케이션 개발 시에 기존의 뉴럴 네트워크 모델 및 하드웨어에 관련된 정보를 이용하여 뉴럴 네트워크 변형 기능이 수행될 수 있으며, 또는 전자 시스템의 부팅 도중, 또는 부팅 이후 실제 뉴럴 네트워크 모델이 수행되기 전에 스태틱 정보에 기반하는 뉴럴 네트워크 모델의

변형 기능이 수행될 수 있다.

[0061] 먼저, 기존의 뉴럴 네트워크 모델에서 각종 정보들이 파싱되고(S11), 스태틱 정보로서 컴퓨팅 리소스 정보가 분석될 수 있다(S12). 또한, 기존의 다양한 뉴럴 네트워크 모델들 중 이를 실행할 하드웨어(예컨대, 컴퓨팅 하드웨어)에 의해 지원되지 않거나 또는 더 나은 가속으로서 실행되지 않는 모델들이 존재하는 지가 판단될 수 있다(S13). 만약, 컴퓨팅 하드웨어에 의해 지원되지 않는 뉴럴 네트워크 모델이 존재하는 경우에는, 전술한 실시예들에 따른 뉴럴 네트워크 모델에 대한 변형 동작을 통해 상기 컴퓨팅 하드웨어에 적합한 변형된 뉴럴 네트워크 모델을 생성할 수 있다(S14).

[0062] 한편, 컴퓨팅 하드웨어에 의해 지원 가능한 뉴럴 네트워크 모델이나 앞선 단계에서의 변형된 뉴럴 네트워크 모델에 대해, 각각의 뉴럴 네트워크 모델의 특성들 중 상기 컴퓨팅 하드웨어에 더 적합한 또는 조화되는 부분이 존재하는 지가 판단될 수 있다(S15). 만약, 더 적합한 부분이 존재하지 않는 경우에는 상기 뉴럴 네트워크 모델이 상기 컴퓨팅 하드웨어에 의해 실행될 뉴럴 네트워크 모델로 결정될 수 있다. 반면에, 더 적합한 부분이 존재하는 경우에는 전술한 실시예들에 따라 기존 뉴럴 네트워크 모델의 특성들 중 일부를 변경함으로써 변형된 뉴럴 네트워크 모델이 생성되며, 즉, 변형된 특성이 상기 컴퓨팅 하드웨어에 링크될 수 있다(S16).

[0063] 한편, 도 8은 온라인 모드에서의 뉴럴 네트워크 모델의 변형 예를 나타낸다. 도 8을 참조하면, 뉴럴 네트워크의 실행을 대기하는 단계가 존재하고(S21), 상기 실행 대기중의 뉴럴 네트워크 모델은 기존의 뉴럴 네트워크 모델이거나 또는 앞선 도 7에서의 오프라인 모드에서 생성된 변형된 뉴럴 네트워크 모델일 수 있다. 뉴럴 네트워크 모델의 실행이 시작되면, 런타임 도중 컨텍스트 매니저로부터의 컨텍스트 정보가 뉴럴 네트워크 어댑터 모듈에 의해 체크될 수 있으며(S22), 체크된 컨텍스트 정보에 따라 현재 실행중인 뉴럴 네트워크 모델의 일부 정보를 변경할 필요성이 있는 지 판단될 수 있다(S23).

[0064] 뉴럴 네트워크 모델의 정보는 다양하게 변경될 수 있으며, 일 예로 도 8에서는 판단 결과에 따라 뉴럴 네트워크의 성능(예컨대, 정확도)을 낮추는 예가 설명된다. 예컨대, 현재 실행되는 어플리케이션의 허용 가능한 정확도 수준을 고려하여 어느 정도로 성능을 저하시킬 지가 산출될 수 있으며(S24), 산출 결과에 따라 필요한 정도의 성능을 충족시키기 위한 변형 동작이 수행될 수 있다(S25).

[0065] 도 9는 변형된 뉴럴 네트워크 모델을 통해 런타임 도중 커널 사이즈가 변경되는 예를 나타낸다.

[0066] 도 9를 참조하면, 뉴럴 네트워크 연산을 수행할 하드웨어에 의해 뉴럴 네트워크 모델이 수신되고(S31), 하드웨어는 수신된 뉴럴 네트워크 모델에 따라 제1 사이즈의 커널에 따른 연산을 수행할 수 있다(S32). 또한, 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈은 런타임 도중 전술한 예에 따른 각종 다이나믹 정보를 수신하고(S33), 상기 뉴럴 네트워크 모델을 실행하는 하드웨어가 변경되는 지 여부를 판단할 수 있다(S34).

[0067] 판단 결과, 하드웨어가 변경되지 않는 경우에는 현재 뉴럴 네트워크 모델을 실행하는 하드웨어에 의해 연산이 수행될 수 있다. 반면에, 상기 뉴럴 네트워크 모델이 실행되는 하드웨어가 다른 전용 하드웨어로 변경되는 경우, 뉴럴 네트워크 어댑터 모듈은 상기 뉴럴 네트워크 모델 및 다이나믹 정보를 기반으로 하여 커널 사이즈를 변경함으로써 변형된 뉴럴 네트워크 모델을 생성할 수 있다(S35). 일 예로, 상기 커널 사이즈는 제1 사이즈에서 제2 사이즈로 변경될 수 있으며, 전용 하드웨어는 제2 사이즈의 커널에 따른 연산을 수행할 수 있다(S36).

[0068] 도 10은 유저 인터페이스를 통한 하드웨어 선택 정보를 이용한 동작 예를 나타낸다.

[0069] 도 10을 참조하면, 뉴럴 네트워크 어댑터 모듈은 입력 뉴럴 네트워크 모델을 수신하고(S41), 이와 함께 유저 인터페이스를 통해 제공된 유저 API 정보를 수신할 수 있다(S42). 또한, 뉴럴 네트워크 어댑터 모듈은 유저 API 정보를 통해 전용 하드웨어의 선택 정보가 포함되어 있는 지를 판단할 수 있다(S43).

[0070] 만약, 하드웨어 선택 정보가 포함되어 있지 않은 경우, 뉴럴 네트워크 어댑터 모듈의 제어에 기반하여 입력 뉴럴 네트워크 모델은 임의의 하드웨어에 할당될 수 있다(S44). 만약, CPU 등의 일반 하드웨어에 모델이 할당되는 경우에는 전술한 예에서의 모델 변형 동작 없이 입력 뉴럴 네트워크 모델이 일반 하드웨어에 할당될 수 있다. 또는, NPU 등의 전용 하드웨어에 모델이 할당되는 경우에는 전술한 예에 따라 변형된 뉴럴 네트워크 모델이 생성되고, 변형된 뉴럴 네트워크 모델이 전용 하드웨어에 할당될 수 있다.

[0071] 한편, 하드웨어 선택 정보가 포함된 경우, 뉴럴 네트워크 어댑터 모듈의 제어에 기반하여 사용자에 의해 선택된 전용 하드웨어의 정보가 판단될 수 있다(S45). 뉴럴 네트워크 어댑터 모듈은 선택된 전용 하드웨어의 스태틱 및/또는 다이나믹 정보를 이용하여 변형된 뉴럴 네트워크 모델을 생성할 수 있다(S46). 또한, 생성된 뉴럴 네트워크 모델이 전용 하드웨어에 할당될 수 있다(S47).

[0072] 이하에서는, 입력 뉴럴 네트워크 모델에서 파싱되는 다양한 정보들을 변경함으로써 변형된 뉴럴 네트워크 모델을 생성하는 예들이 설명된다.

[0073] 도 11은 커널 사이즈를 조절함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다.

[0074] 전용 프로세서로서 NPU는 특정 사이즈의 커널 구조에서 최적의 성능으로 동작될 수 있으며, 일 예로 N*N 커널 구조에서 NPU는 N이 특정 값을 가질 때 최적의 성능으로 동작될 수 있다. 일 예로서, NPU는 N*N 커널 구조에서 N이 4의 배수 값을 가지거나, 또는 4의 배수 값 보다 다소 작은 값을 가질 때 최적의 성능으로 동작될 수 있다. 반면에, 입력 뉴럴 네트워크 모델이 5*5 사이즈의 커널에 따른 연산을 수행할 때, 상기 입력 뉴럴 네트워크 모델이 NPU에 의해 실행되는 경우에는 그 성능이 저하될 수 있다.

[0075] 본 발명의 실시예에 따른 뉴럴 네트워크 어댑터 모듈은 상기 입력 뉴럴 네트워크 모델로부터 N*N 커널 구조에 관련된 정보를 파싱할 수 있으며, 또한 NPU의 스태틱 및/또는 다이나믹 정보를 통해 NPU에 최적화된 커널 사이즈 정보를 판단할 수 있다. 일 예로서, 뉴럴 네트워크 어댑터 모듈은 입력 뉴럴 네트워크 모델에서 5*5 사이즈의 커널을 두 개의 3*3 사이즈의 커널들로 대략화(approximation)할 수 있으며, 커널이 두 개의 3*3 사이즈의 커널들로 변경된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 또한, 상기 변형된 뉴럴 네트워크 모델은 NPU에 의해 실행될 수 있다.

[0076] 도 12는 재사용(reuse) 방식을 변경함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다.

[0077] 도 12를 참조하면, 뉴럴 네트워크를 실행하는 프로세서는 다수의 프로세싱 엘레먼트(PE)를 포함할 수 있으며, 프로세싱 엘레먼트(PE)의 연산 특성에 따라 상기 프로세서의 데이터 플로우(data flow)는 다수의 종류들로 분류될 수 있다. 일 예로서, NPU 등의 하드웨어는 웨이트 고정(Weight Stationary) 형태의 데이터 플로우를 가질 수 있으며, GPU 등의 하드웨어는 출력 고정(Output Stationary) 형태의 데이터 플로우를 가질 수 있다. 한편, 다른 일부의 하드웨어는 재사용(reuse)을 이용하지 않는 형태의 데이터 플로우를 가질 수 있다.

[0078] 본 발명의 실시예에 따른 뉴럴 네트워크 어댑터 모듈은 상기 입력 뉴럴 네트워크 모델로부터 데이터 재사용에 관련된 정보를 파싱할 수 있으며, 재사용에 관련된 종류로서 컨벌루션 재사용, 피쳐 맵 재사용 및 필터 재사용 등의 정보가 파싱될 수 있다. 이 때, 하드웨어의 연산 방식 특성을 고려하여, 웨이트 고정(Weight Stationary) 형태의 하드웨어는 필터 재사용 방식에 최적화될 수 있으며, 출력 고정(Output Stationary) 형태의 하드웨어는 피쳐 맵 재사용 방식에 최적화될 수 있다. 반면에, 재사용(reuse)을 이용하지 않는 하드웨어는 뉴럴 네트워크 모델의 재사용 방식에 돈 케어될 수 있다.

[0079] 뉴럴 네트워크 어댑터 모듈의 변형 모듈은 입력 뉴럴 네트워크 모델이 실행될 하드웨어의 종류에 관련된 정보를 스태틱 정보 또는 다이나믹 정보로서 수신할 수 있으며, 하드웨어의 데이터 플로우 형태를 고려하여 입력 뉴럴 네트워크 모델의 재사용 정보를 변경할 수 있다. 일 예로서, 뉴럴 네트워크 어댑터 모듈은 웨이트 고정(Weight Stationary) 형태의 하드웨어에 의해 실행될 뉴럴 네트워크 모델에 대해 상기 필터 재사용 방식이 적용된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 또는, 뉴럴 네트워크 어댑터 모듈은 출력 고정(Output Stationary) 형태의 하드웨어에 의해 실행될 뉴럴 네트워크 모델에 대해 상기 피쳐 맵 재사용 방식이 적용된 변형된 뉴럴 네트워크 모델을 생성할 수 있다. 또는, 재사용(reuse)을 이용하지 않는 하드웨어에 의해 뉴럴 네트워크 모델이 실행되는 경우, 뉴럴 네트워크 어댑터 모듈은 임의의 재사용 방식을 변형된 뉴럴 네트워크 모델에 적용할 수 있다.

[0080] 도 13은 성능을 조절함에 의해 변형된 뉴럴 네트워크 모델을 생성하는 예를 나타내는 블록도이다. 실행되는 어플리케이션의 종류에 따라 뉴럴 네트워크 모델의 정확도 등의 성능이 조절될 수 있으며, 일 예로서 영상 처리를 위한 뉴럴 네트워크 연산을 수행함에 있어서 차량 자율 주행의 경우에는 그 성능이 향상될 필요가 있으나, 스마트 폰 등의 단순 카메라 어플리케이션이 실행되는 경우에는 그 성능이 허용되는 범위 내에서는 다소 저하되어도 무방하다.

[0081] 도 13을 참조하면, 입력 뉴럴 네트워크 모델은 2D 형태의 d*d 필터(또는, 커널)를 이용한 컨벌루션 연산이 적용되나, 전술한 유저 API 정보(Info_API), 스태틱 정보(Info_Static) 및 다이나믹 정보(Info_Dynamic) 중 적어도 하나에 기반하여 그 성능을 낮춘 변형된 뉴럴 네트워크 모델이 생성될 수 있다. 뉴럴 네트워크 모델의 성능 조절은 다양한 정보에 기반하여 수행될 수 있으며, 일 예로서 다이나믹 정보(Info_Dynamic)에 의해 뉴럴 네트워크 동작을 요구하는 어플리케이션의 종류가 판단될 수 있으며, 판단된 어플리케이션에 따라 뉴럴 네트워크 모델의 성능이 조절될 수 있다.

[0082] 일 실시예에 따라, 성능 조절은 필터의 사이즈를 조절함에 의해 수행될 수 있다. 예컨대, 입력 뉴럴 네트워크 모델에 적용된 2D 형태의 d*d 필터는 하나 이상의 필터들로 대략화(approximation)될 수 있으며, 도 13에서는 2D 형태의 d*d 필터가 d*1 X-필터와 1*d Y-필터로 대략화(approximation)되는 예가 도시된다.

[0083] 한편, 도 13에 도시된 실시예에서, 어플리케이션 종류 이외에도 다양한 다른 정보들에 기반하여 뉴럴 네트워크 모델의 성능이 조절될 수도 있다. 일 예로서, 특정 뉴럴 네트워크 모델에서 요구하는 데이터 처리 속도가 30 fps에 해당하고, 런타임 중에 상기 특정 뉴럴 네트워크 모델이 60 fps에 해당하는 속도로 실행되고 있는 경우, 상기 런타임 중의 속도에 관련된 정보가 다이나믹 정보(Info_Dynamic)로서 제공될 수 있으며, 이 때 상기 특정 뉴럴 네트워크 모델의 커널 사이즈를 조절함으로써 그 처리 속도가 감소될 수 있다. 반대로, 런타임 중의 속도가 요구되는 속도보다 느릴 경우에는, 속도가 높아지도록 뉴럴 네트워크 모델이 변형될 수 있을 것이다.

[0084] 도 14a,b,c는 하드웨어의 연산 특성에 따라 뉴럴 네트워크 모델이 변형되는 예를 나타내는 도면이다.

[0085] 도 14a를 참조하면, 하드웨어는 다양한 구조로서 연산을 수행할 수 있으며, 일 예로서 하드웨어는 다수의 연산 유닛(ALU)들을 포함할 수 있으며, 하드웨어가 시간적인 구조(temporal architecture)를 갖는 경우 다수의 연산 유닛(ALU)들은 병렬한 방식으로 연산을 수행할 수 있다. 반면에, 하드웨어가 공간적인 구조(spatial architecture)를 갖는 경우 다수의 연산 유닛(ALU)들은 순차적인 방식으로 연산을 수행할 수 있다.

[0086] 상기와 같은 프로세서의 구조에 따라 서로 다른 모델의 뉴럴 네트워크 모델이 각각의 프로세서에 최적화될 수 있다. 예컨대, 시간적인 구조의 프로세서는 곱셈 연산의 횟수를 감소시킴으로써 그 연산이 최적화될 수 있으며, 도 14b에 도시된 바와 같이 본 발명의 일 실시예에 따라 상대적으로 큰 커널 사이즈(예컨대, 5*5)의 입력 뉴럴 네트워크 모델은 하나 이상의 작은 커널 사이즈(예컨대, 3*3)를 갖는 뉴럴 네트워크 모델로 변형되어 시간적인 구조의 프로세서로 제공될 수 있다. 반면에, 공간적인 구조의 프로세서는 데이터 재사용(reuse)을 증가시킴으로써 그 연산이 최적화될 수 있으며, 도 14c에 도시된 바와 같이 본 발명의 일 실시예에 따라 2 차원 구조의 커널 사이즈(예컨대, 5*5)의 입력 뉴럴 네트워크 모델은 하나 이상의 1 차원 구조의 커널 사이즈(예컨대, 1*5, 5*1)를 갖는 뉴럴 네트워크 모델로 변형되어 공간적인 구조의 프로세서로 제공될 수 있다.

[0087] 한편, 변형 가능한 실시예로서, 동일한 구조의 프로세서이더라도 프로세서에 의해 실행되는 커널 사이즈의 선호 정보는 서로 다를 수 있다. 일 예로, 시간적인 구조(temporal architecture)의 전용 하드웨어로서 FPGA는 퓨리에 트랜스폼(FFT)에 기반하는 연산을 수행하고 5*5 사이즈의 커널에서 최적화된 연산을 수행할 수 있다. 반면에, 시간적인 구조(temporal architecture)의 전용 하드웨어로서 NPU는 위노가드(Winograd) 방식에 기반하는 연산을 수행하고 3*3 사이즈의 커널에서 최적화된 연산을 수행할 수 있다. 이 때, 도 14b에 도시된 실시예에서, 만약 입력 뉴럴 네트워크가 7*7 사이즈의 커널을 포함하고, 상기 입력 뉴럴 네트워크가 전용 하드웨어로서 FPGA에 의해 실행되는 경우, 상기 7*7 사이즈의 커널은 두 개의 5*5 사이즈의 커널들로 변경될 수 있다. 또한, 상기 입력 뉴럴 네트워크가 전용 하드웨어로서 NPU에 의해 실행되는 경우, 상기 7*7 사이즈의 커널은 세 개의 3*3 사이즈의 커널들로 변경될 수 있다.

[0088] 또한, 변형 가능한 실시예로서, 입력 뉴럴 네트워크 모델의 종류에 따라 뉴럴 네트워크 어댑터 모듈의 제어에 기반하여 특정 전용 하드웨어에 할당하는 방식으로써 뉴럴 네트워크 모델의 실행이 최적화될 수 있다. 일 예로, 뉴럴 네트워크 모델이 CNN에 해당하는 경우, CNN은 빈번한 데이터 재사용이 수행되는 반면에 피쳐 맵이나 커널은 재사용되지 않을 수 있다. 이 때, CNN에 해당하는 뉴럴 네트워크 모델은 전술한 공간적인 구조(spatial architecture)의 전용 하드웨어에 할당될 수 있다. 반면에, 뉴럴 네트워크 모델이 RNN/LSTM(Long Short Term Memory)에 해당하는 경우, RNN/LSTM는 데이터 재사용을 수행하지 않거나 드문 빈도로서 수행할 수 있으며, 이 때 RNN/LSTM에 해당하는 뉴럴 네트워크 모델은 압축 가속(compression-accelerator) 임베디드 하드웨어에 의해 최적으로 실행될 수 있으며, 일 예로 RNN/LSTM에 해당하는 뉴럴 네트워크 모델은 전술한 시간적인 구조(temporal architecture)의 전용 하드웨어에 할당될 수 있다.

[0089] 이외에도, 뉴럴 네트워크 모델에 의해 수행되는 다양한 태스크(task)의 종류에 따라 전술한 전용 하드웨어의 할당 동작이 제어될 수 있다. 일 예로 뉴럴 네트워크 모델은 다양한 종류의 레이어들로서 컨벌루션 레이어와 풀-커텍티드(Fully-connected) 레이어를 포함할 수 있으며, 컨벌루션 레이어는 높은 연산을 요구하는 레이어이고 풀-커텍티드(Fully-connected) 레이어는 높은 메모리 접근을 요구하는 레이어일 수 있다. 이 때, 뉴럴 네트워크 모델의 실행에서 컨벌루션 레이어는 Transition-based Object-oriented Programming System(TOPS) 급의 하드웨어로서 공간적인 구조(spatial architecture)의 전용 하드웨어에 할당될 수 있다. 반면에, 풀-커텍티드(Fully-connected) 레이어는 높은 메모리 용량의 하드웨어로서 시간적인 구조(temporal architecture)의 전용 하드웨어

에 할당될 수 있다.

[0090] 도 15는 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈이 소프트웨어적으로 구현되는 예를 나타내는 블록도이다. 도 15에 도시된 시스템은 어플리케이션 프로세서(400)일 수 있으며, 어플리케이션 프로세서(400)는 반도체 칩으로서 시스템 온 칩(SoC)으로 구현될 수 있다.

[0091] 어플리케이션 프로세서(400)는 프로세서(410) 및 동작 메모리(420)를 포함할 수 있다. 또한, 도 15에는 도시되지 않았으나, 어플리케이션 프로세서(400)는 시스템 버스에 연결되는 하나 이상의 IP(Intellectual Property) 모듈들을 더 포함할 수도 있다. 동작 메모리(420)는 어플리케이션 프로세서(400)가 채용되는 시스템의 동작과 관련된 각종 프로그램 및 명령어 등의 소프트웨어를 저장할 수 있으며, 일 예로서 운영 체제(421), 뉴럴 네트워크 모듈(422) 및 어댑터 모듈(423)을 포함할 수 있다. 어댑터 모듈(423)은 전술한 실시예들에 따른 뉴럴 네트워크 어댑터 모듈의 기능을 수행할 수 있다.

[0092] 뉴럴 네트워크 모듈(422)은 기존의 뉴럴 네트워크 모듈 또는 본 발명의 실시예들에 따른 변형된 뉴럴 네트워크 모듈의 연산을 실행할 수 있다. 또한, 어댑터 모듈(423)은 입력 뉴럴 네트워크 모델을 수신하고, 전술한 실시예들에 따른 모델 변형 동작을 수행할 수 있다. 예시적인 실시예에 따라, 뉴럴 네트워크 모듈(422)은 운영 체제(421) 내에 구현될 수도 있을 것이다.

[0093] 한편, 도 15에는 하나의 프로세서(410)가 도시되었으나, 어플리케이션 프로세서(400)는 다수의 프로세서들을 포함할 수도 있다. 이 때, 다수의 프로세서들 중 일부는 일반 프로세서에 해당하고, 다른 일부는 뉴럴 네트워크 모델의 실행을 위한 전용 프로세서일 수 있다. 어댑터 모듈(423)은 전용 프로세서에 관련하여 스태틱 정보 및 다이나믹 정보를 이용한 모델 변형 동작을 수행하고, 상기 전용 프로세서에 최적화된 변형된 뉴럴 네트워크 모델을 생성할 수 있을 것이다.

[0094] 도 16은 본 발명의 예시적인 실시예에 따른 뉴럴 네트워크 어댑터 모듈이 자동차에 채용되는 자율 운행 모듈 내에 구현되는 예를 나타내는 블록도이다. 도 16에 도시된 시스템은 자율 운행 시스템(500)에 해당할 수 있으며, 자율 운행 시스템(500)은 센서 정보 수집부(510), 네비게이션 정보 수집부(520), 자율 운행 모듈(530) 및 중앙처리 장치(540)를 포함할 수 있다. 또한, 자율 운행 모듈(530)은 뉴럴 네트워크 장치(531) 및 뉴럴 네트워크 어댑터 모듈(532)을 포함할 수 있다.

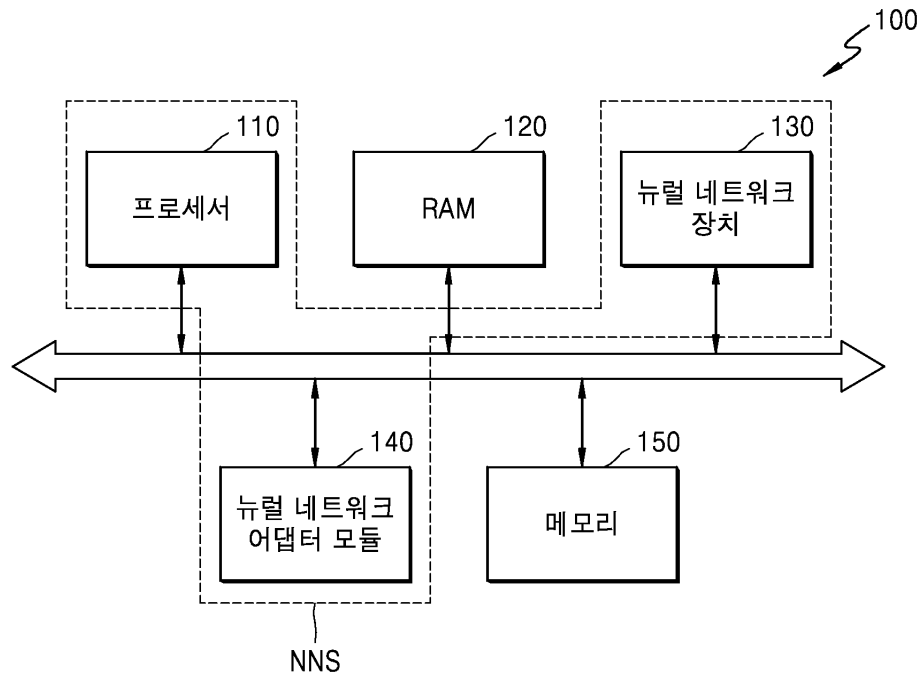
[0095] 뉴럴 네트워크 장치(531)는 각종 영상 정보 및 음성 정보를 이용한 뉴럴 네트워크 동작을 수행하고, 수행 결과를 기초로 영상 인식 결과 및 음성 인식 결과 등의 정보 신호를 생성할 수 있다. 일 예로서, 센서 정보 수집부(510)는 카메라나 마이크 등의 각종 영상 정보 및 음성 정보를 수집할 수 있는 장치들을 포함하고, 이를 자율 운행 모듈(530)로 제공할 수 있다. 또한, 네비게이션 정보 수집부(520)는 자동차 운행과 관련된 각종 정보(예컨대, 위치 정보 등)를 자율 운행 모듈(530)로 제공할 수 있다. 뉴럴 네트워크 장치(531)는 센서 정보 수집부(510) 및/또는 네비게이션 정보 수집부(520)로부터의 정보를 입력으로 하여, 다양한 종류의 뉴럴 네트워크 모델을 실행함으로써 상기 정보 신호를 생성할 수 있다.

[0096] 뉴럴 네트워크 어댑터 모듈(532)은 전술한 실시예들에 따라 뉴럴 네트워크 모델을 변형하는 동작을 수행할 수 있다. 뉴럴 네트워크 어댑터 모듈(532)은 자율 운행 모듈(530) 내부 또는 외부에 구비될 수 있는 하나 이상의 전용 프로세서(미도시)의 스태틱 정보 및/또는 다이나믹 정보에 기반하여 변형된 뉴럴 네트워크 모델을 생성하고, 뉴럴 네트워크 장치(531)는 변형된 뉴럴 네트워크 모델을 실행할 수 있다.

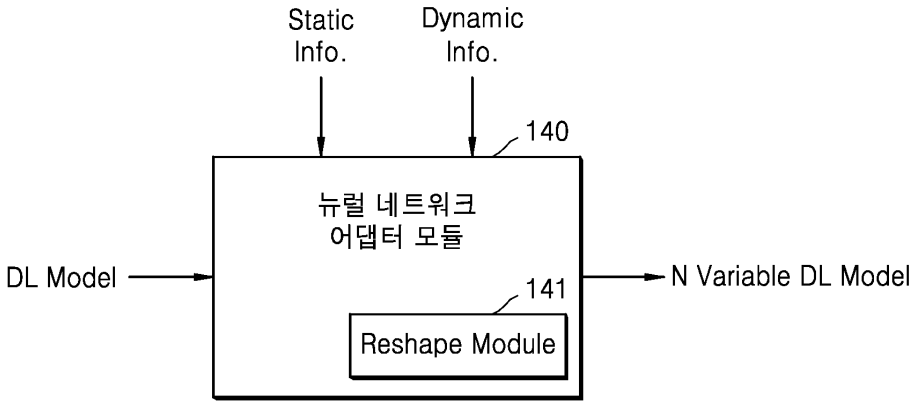
[0097] 이상에서와 같이 도면과 명세서에서 예시적인 실시예들이 개시되었다. 본 명세서에서 특정한 용어를 사용하여 실시예들을 설명되었으나, 이는 단지 본 개시의 기술적 사상을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 개시의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 기술분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 개시의 진정한 기술적 보호범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

도면

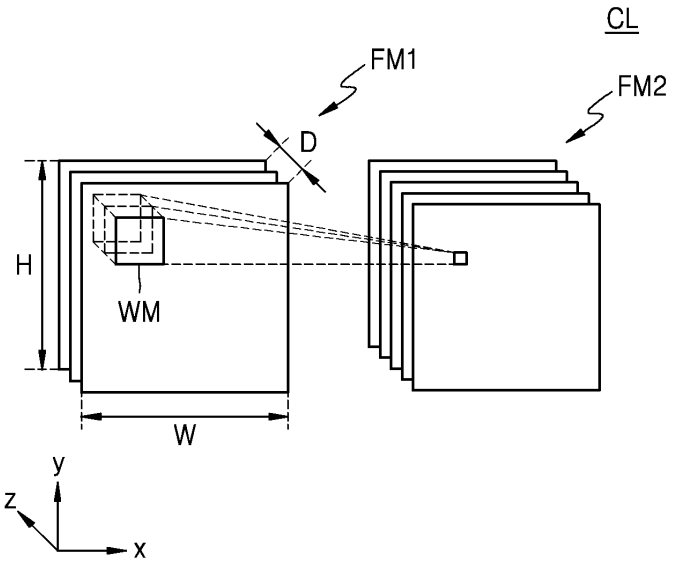
도면1


100
110
프로세서
120
RAM
130
뉴럴 네트워크
장치
140
뉴럴 네트워크
어댑터 모듈
150
메모리
NNS

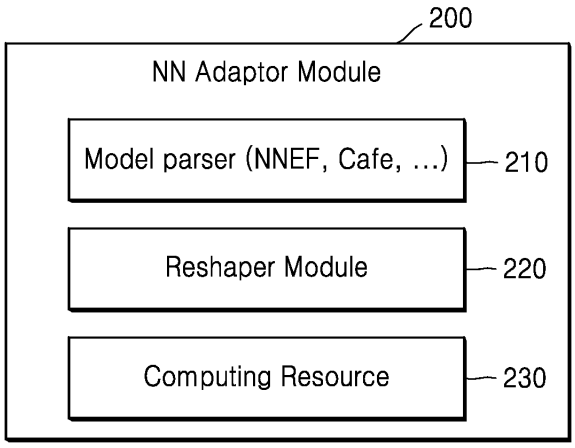
도면2


Static
Info.
Dynamic
Info.
140
뉴럴 네트워크
어댑터 모듈
DL Model
141
Reshape Module
N Variable DL Model

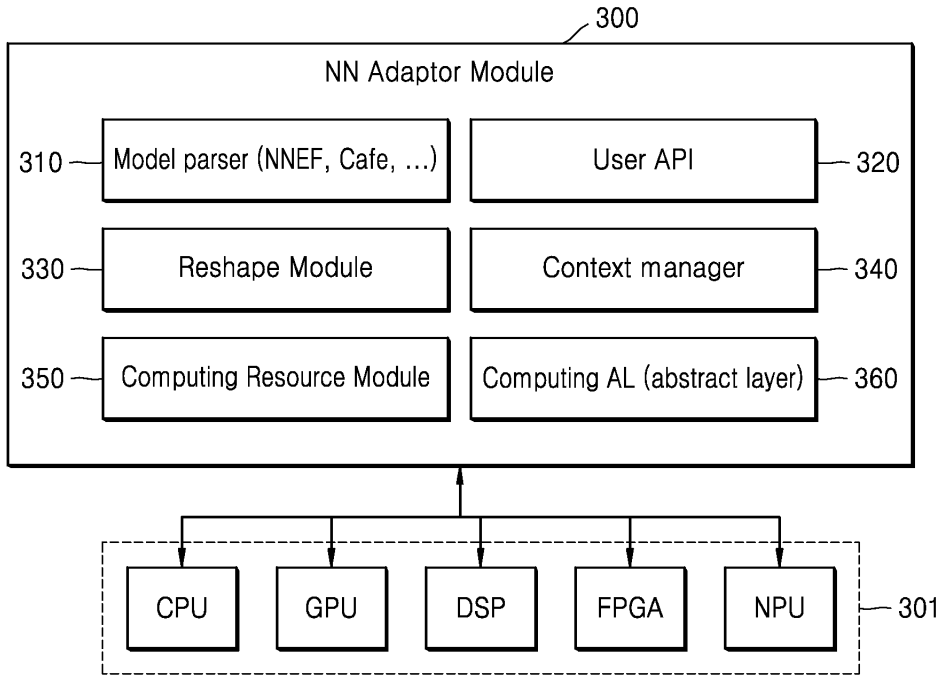
도면3


CL
FM1
FM2
D
H
WM
W
y
z
x

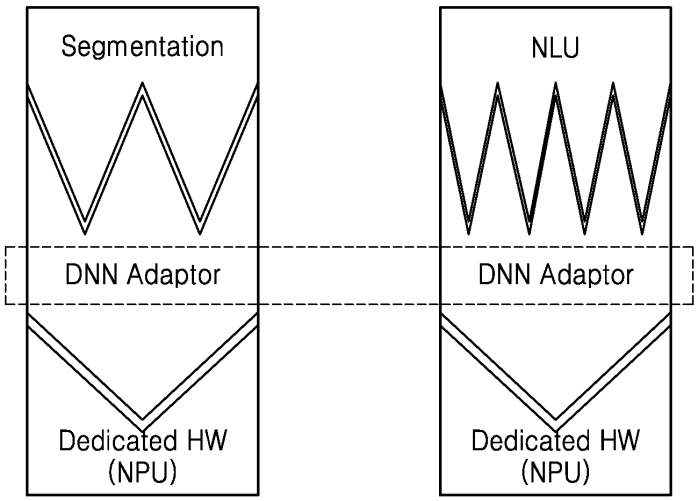
도면4


200
NN Adaptor Module
Model parser (NNEF, Cafe, ...)
210
Reshaper Module
220
Computing Resource
230

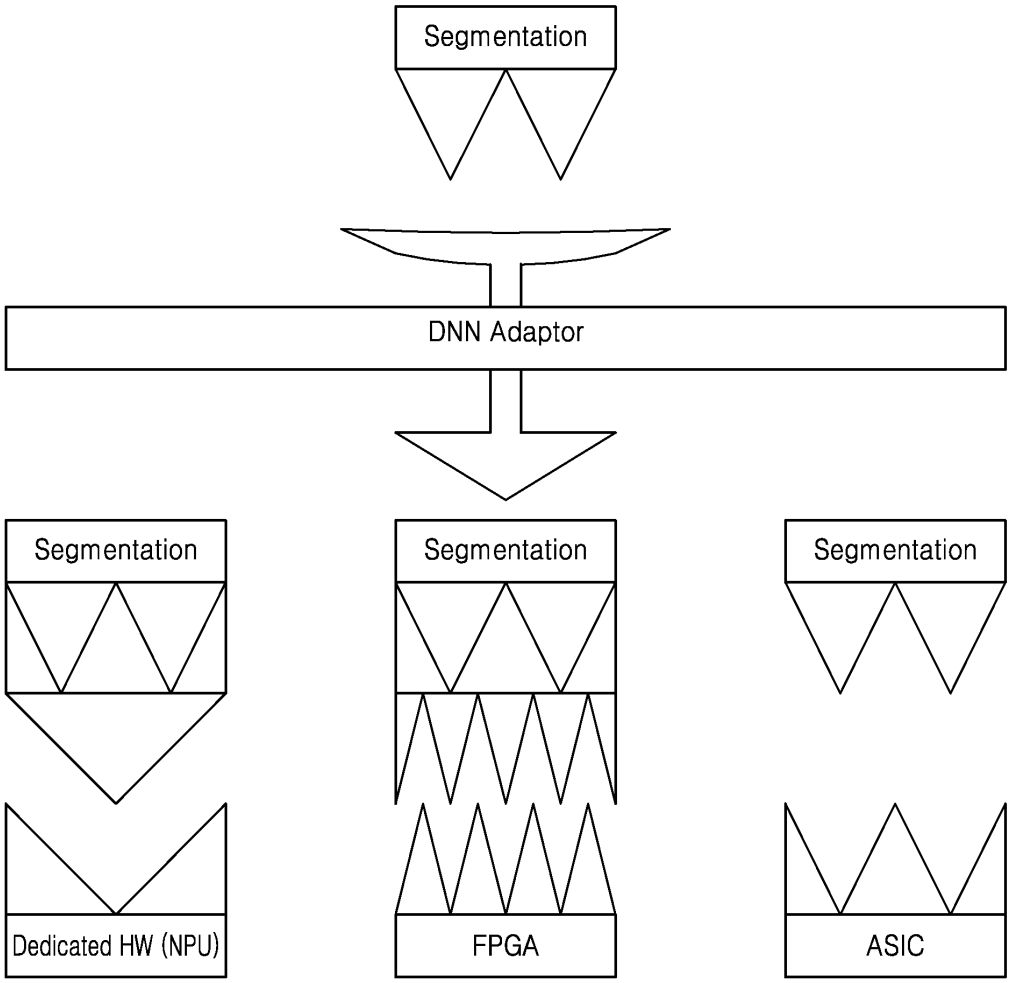
도면5


300
NN Adaptor Module
310
Model parser (NNEF, Cafe, ...)
User API
320
330
Reshape Module
Context manager
340
350
Computing Resource Module
Computing AL (abstract layer)
360
CPU
GPU
DSP
FPGA
NPU
301

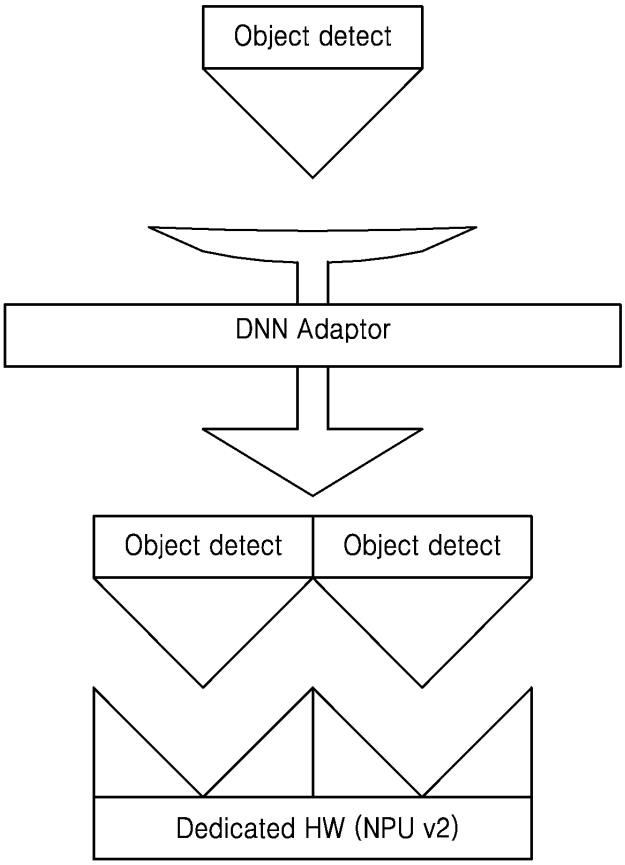
도면6a


Segmentation
NLU
DNN Adaptor
DNN Adaptor
Dedicated HW
(NPU)
Dedicated HW
(NPU)

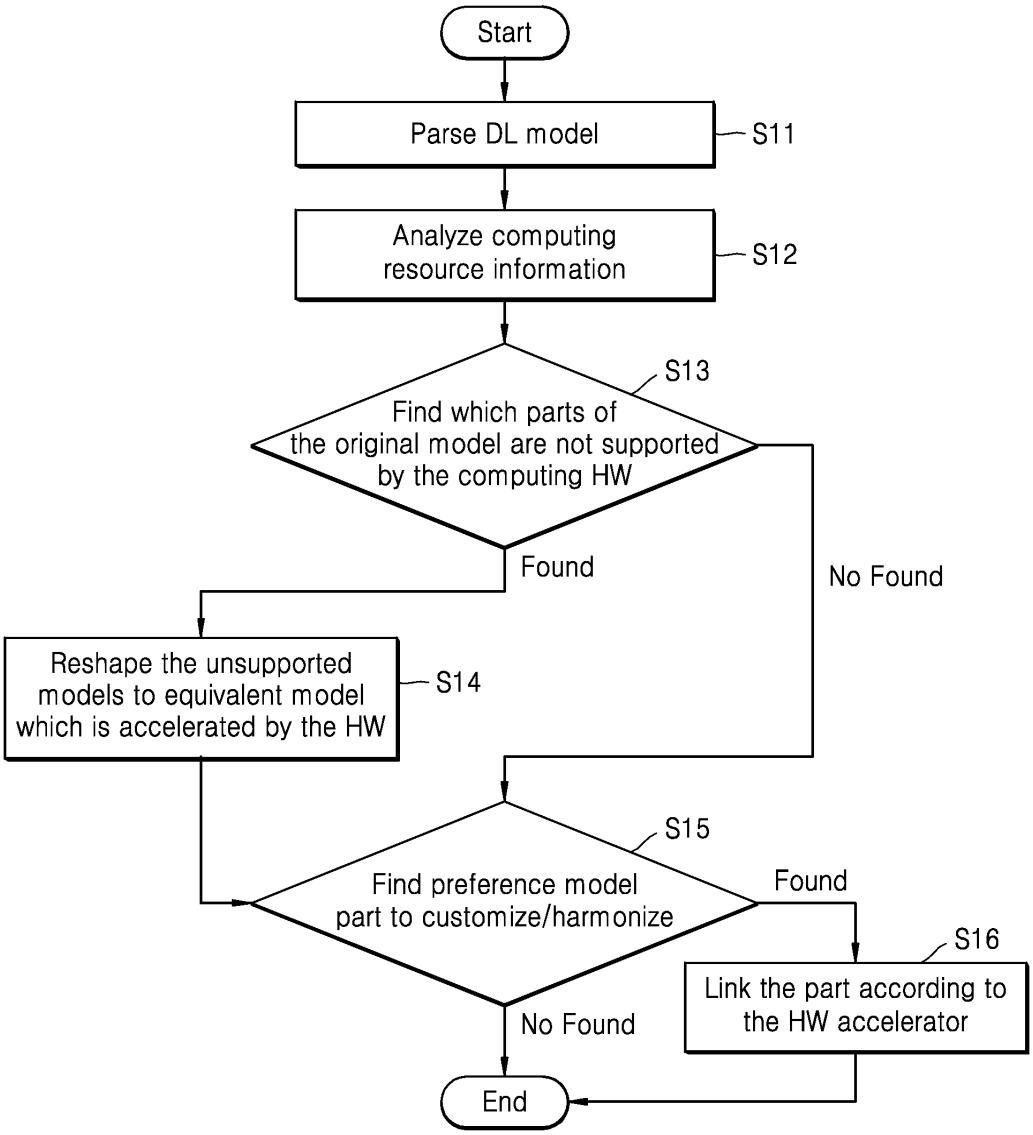
도면6b


Segmentation
DNN Adaptor
Segmentation
Segmentation
Segmentation
Dedicated HW (NPU)
FPGA
ASIC

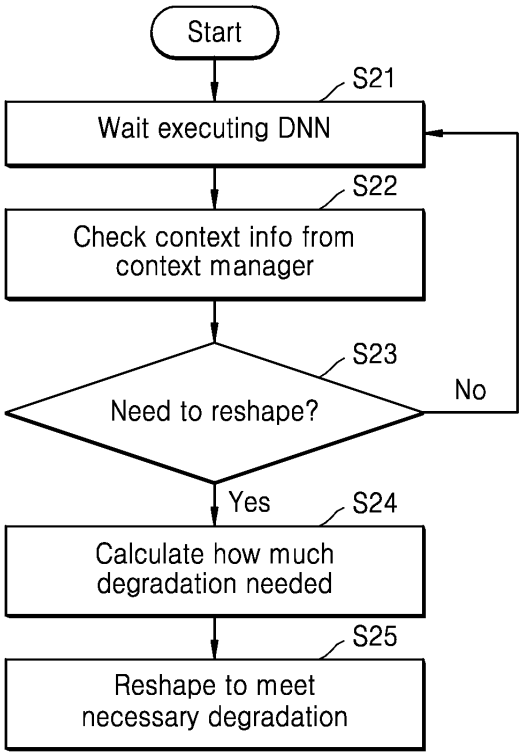
도면6c


Object detect
DNN Adaptor
Object detect
Object detect
Dedicated HW (NPU v2)

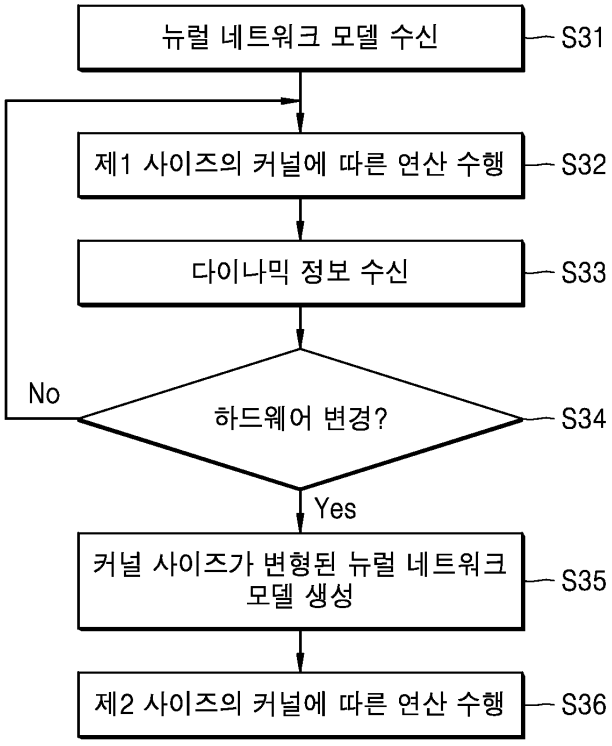
도면7


Start
Parse DL model
S11
Analyze computing resource information
S12
S13
Find which parts of the original model are not supported by the computing HW
Found
No Found
Reshape the unsupported models to equivalent model which is accelerated by the HW
S14
S15
Find preference model part to customize/harmonize
Found
S16
Link the part according to the HW accelerator
No Found
End

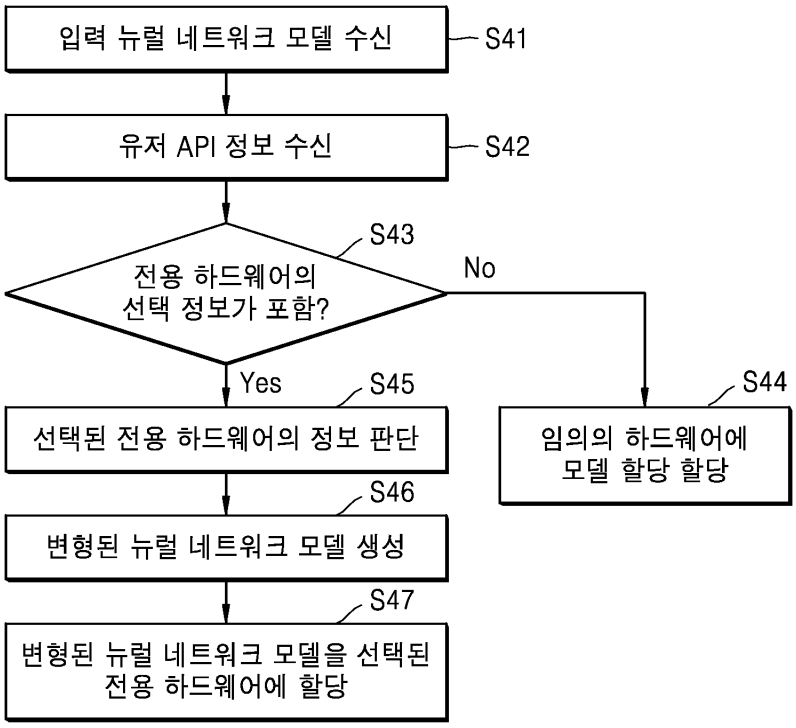
도면8


Start
S21
Wait executing DNN
S22
Check context info from context manager
S23
Need to reshape?
No
Yes
S24
Calculate how much degradation needed
S25
Reshape to meet necessary degradation

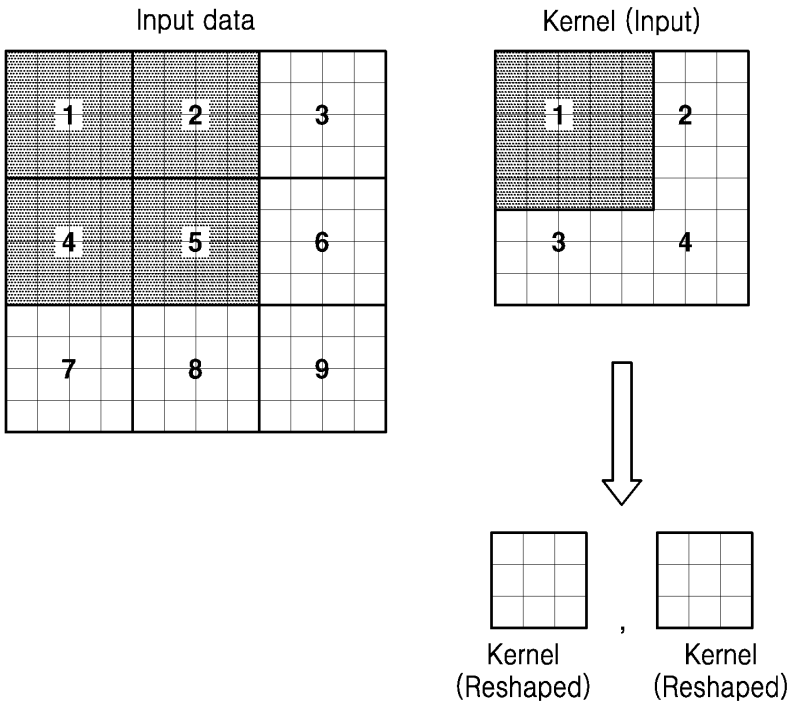
도면9


뉴럴 네트워크 모델 수신
S31
제1 사이즈의 커널에 따른 연산 수행
S32
다이나믹 정보 수신
S33
No
하드웨어 변경?
S34
Yes
커널 사이즈가 변형된 뉴럴 네트워크 모델 생성
S35
제2 사이즈의 커널에 따른 연산 수행
S36

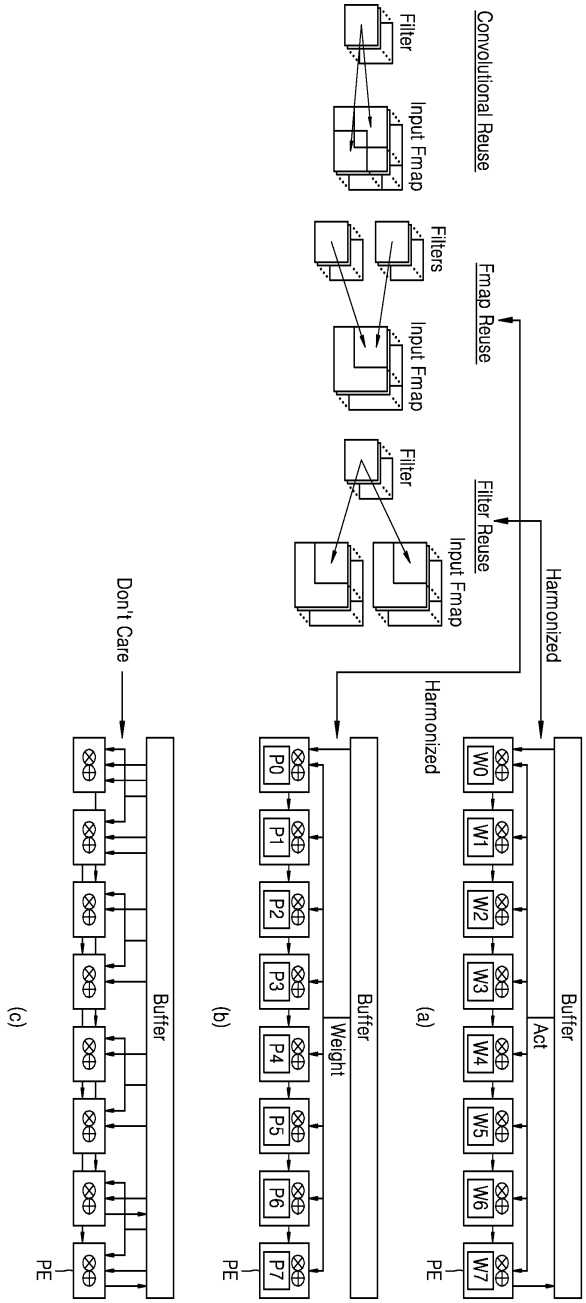
도면10


입력 뉴럴 네트워크 모델 수신
S41
유저 API 정보 수신
S42
S43
전용 하드웨어의
선택 정보가 포함?
No
Yes
S45
선택된 전용 하드웨어의 정보 판단
S44
임의의 하드웨어에
모델 할당 할당
S46
변형된 뉴럴 네트워크 모델 생성
S47
변형된 뉴럴 네트워크 모델을 선택된
전용 하드웨어에 할당

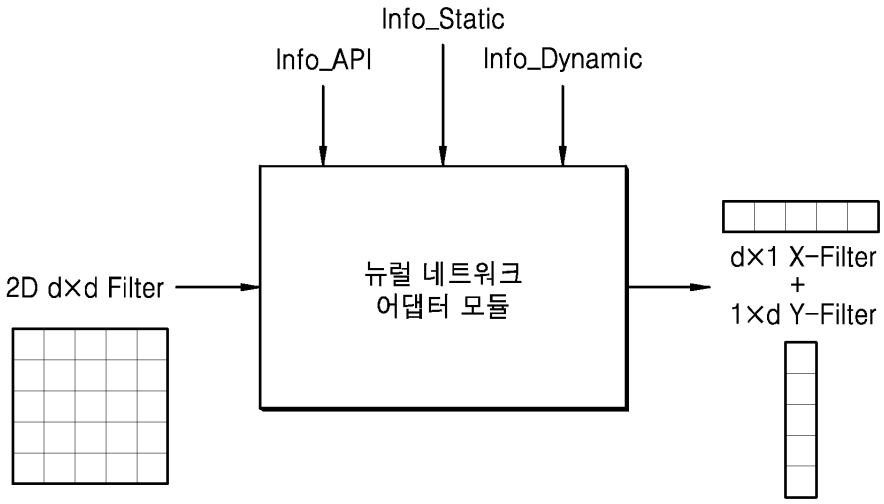
도면11


Input data
1
2
3
4
5
6
7
8
9
Kernel (Input)
1
2
3
4
Kernel
(Reshaped)
,
Kernel
(Reshaped)

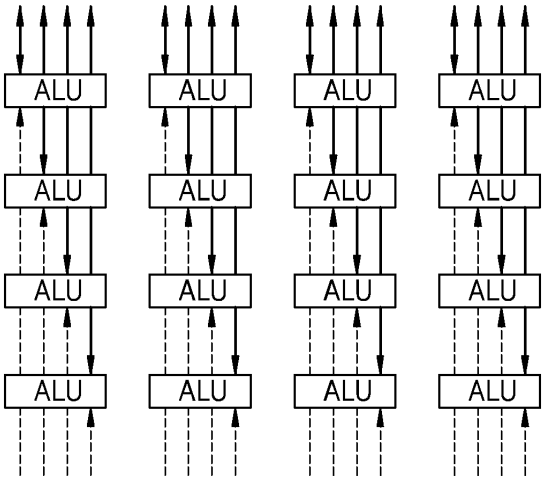
도면12


Convolutional Reuse
Filter
Input Fmap
Fmap Reuse
Filters
Input Fmap
Filter Reuse
Filter
Input Fmap
Harmonized
Harmonized
Don't Care
Buffer
Act
Buffer
Weight
Buffer
PE
PE
PE
(a)
(b)
(c)
W0
W1
W2
W3
W4
W5
W6
W7
P0
P1
P2
P3
P4
P5
P6
P7

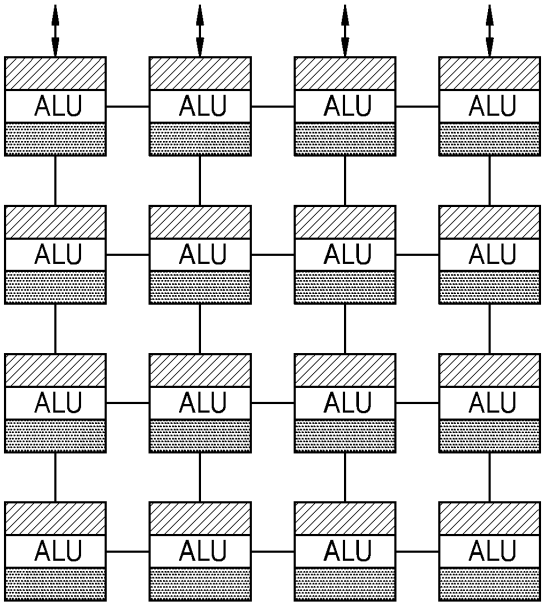
도면13


Info_Static
Info_API
Info_Dynamic
2D d×d Filter
뉴럴 네트워크
어댑터 모듈
d×1 X-Filter
+
1×d Y-Filter

도면14a

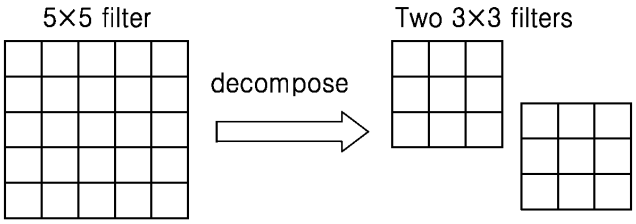


(a)

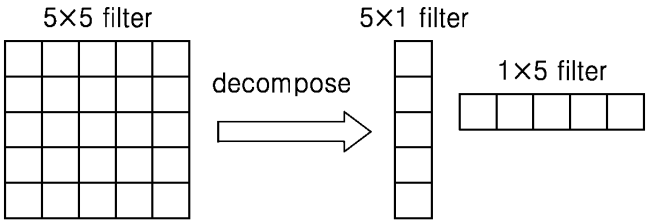


(b)

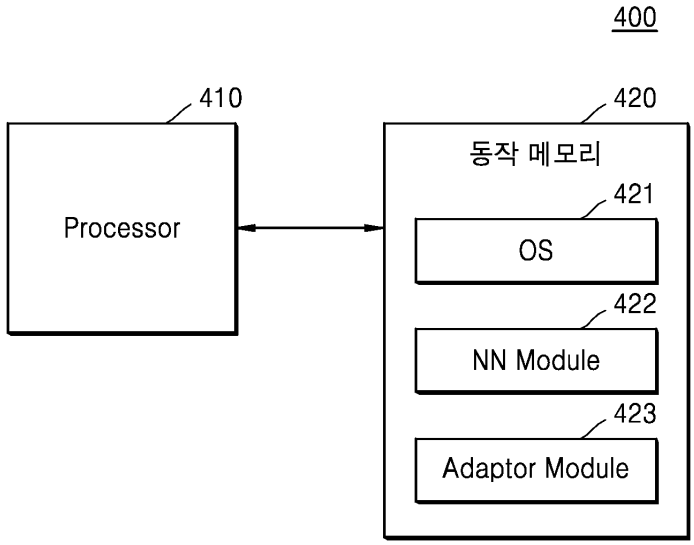
도면14b



도면14c



도면15



도면16