



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0053819
(43) 공개일자 2024년04월25일

(51) 국제특허분류(Int. Cl.)

G06N 3/063 (2023.01) G06F 17/15 (2006.01)
G06F 7/499 (2022.01) G06N 3/08 (2023.01)
G06N 5/04 (2023.01) G06V 20/58 (2022.01)

(52) CPC특허분류

G06N 3/063 (2013.01)
G06F 17/153 (2013.01)

(21) 출원번호 10-2022-0133806

(22) 출원일자 2022년10월18일

심사청구일자 2022년10월18일

(71) 출원인

인하대학교 산학협력단

인천광역시 미추홀구 인하로 100(용현동, 인하대학교)

상명대학교 천안산학협력단

충청남도 천안시 동남구 상명대길 31, 상명대학교내 (안서동)

(72) 발명자

강진구

서울특별시 서초구 양재대로2길 109, 104동 1202호(우면동, 서초호반써밋)

김재명

경기도 고양시 덕양구 푸른마을로 55, 605동 907호(고양동, 푸른마을6단지아파트)

김용우

서울특별시 송파구 백제고분로19길 3-22, 3층(잠실동)

(74) 대리인

양성보

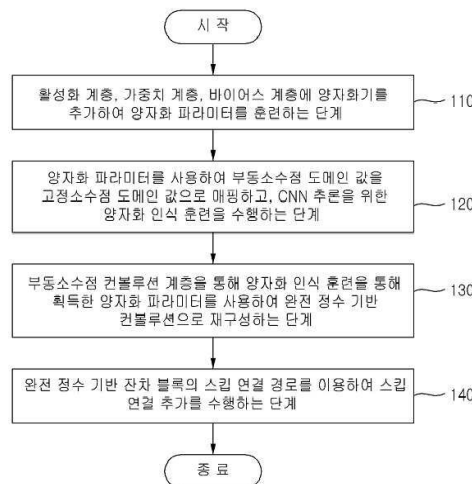
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기

(57) 요약

실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기 및 그 동작 방법이 제시된다. 일 측면에 있어서, 본 발명에서 제안하는 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 동작 방법은 활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련하는 단계, 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 단계, 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계 및 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계를 포함한다.

대표도 - 도1



(52) CPC특허분류

G06F 7/49915 (2013.01)

G06N 3/082 (2023.01)

G06N 5/04 (2023.01)

G06V 20/582 (2022.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711152938
과제번호	2021-0-02052-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	대학ITRC센터육성지원사업(교육운영수익)
연구과제명	[Ezbaro][정부] 스마트 모빌리티를 위한 인공지능 시스템반도체 핵심 기술 개발 및
인력 양성(2차년도)	
기 여 율	1/1
과제수행기관명	인하대학교 산학협력단
연구기간	2022.01.01 ~ 2022.12.31

명세서

청구범위

청구항 1

활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련하는 단계;

양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 단계;

부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계; 및

완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계

를 포함하는 CNN 가속기.

청구항 2

제1항에 있어서,

상기 활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련하는 단계는,

실제 범위 값을 정수 범위 값으로 변환하기 위한 스케일 계수 및 부동소수점 도메인 및 고정소수점 도메인에서 0을 정렬하기 위한 영점을 포함하는 양자화 파라미터를 활성화 계층, 가중치 계층, 바이어스 계층에 추가하여 훈련하는

CNN 가속기의 동작 방법.

청구항 3

제1항에 있어서,

상기 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련을 수행하는 단계는,

양자화기의 스케일 프로세스를 통해 스케일 계수와 영점을 사용하여 고정밀 부동소수점 도메인 값을 고정밀 고정소수점 도메인 값으로 매핑하고,

양자화기의 반올림 프로세스를 통해 가장 가까운 정수 값에 대한 실제 값을 근사하고,

양자화기의 클립 프로세스를 통해 양자화에 따른 비트폭 집합을 벗어나는 값을 감소시키고,

양자화기의 역양자 프로세스를 통해 스케일 계수와 영점을 사용하여 저정밀 고정 소수점 도메인 값을 고정밀 부동소수점 도메인 값으로 변환하는

CNN 가속기의 동작 방법.

청구항 4

제1항에 있어서,

상기 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계는,

정수 스케일러를 통해 스케일 계수를 사용하여 시프트 연산을 수행하고,

정수 클리퍼를 통해 양자화 비트 폭에 따라 텐서의 데이터 범위를 조정하며,

양자화에 따른 부동소수점 가중치 및 바이어스 파라미터를 정수 파라미터로 변환하는

CNN 가속기의 동작 방법.

청구항 5

제1항에 있어서,

상기 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계는,

제1 유형 잔차 블록을 통해 스킵 연결 경로와 ReLU 함수를 통과하는 주 경로 데이터를 추가하고,

제2 유형 잔차 블록을 통해 상기 제1 유형 잔차 블록의 스킵 연결 추가 후 메인 경로의 ReLU를 이동시키고,

제3 유형 잔차 블록을 통해 부동소수점 컨볼루션 계층을 사용하여 스킵 경로의 텐서 크기를 메인 경로와 동일하게 설정한 후 스킵 연결 추가를 수행하고,

제4 유형 잔차 블록을 통해 제3 유형 잔차 블록을 기준으로 스킵 경로의 부동소수점 컨볼루션 계층을 식별 계층으로 변경하는

CNN 가속기의 동작 방법.

청구항 6

활성화 계층, 가중치 계층, 바이어스 계층에 양자화를 추가하여 양자화 파라미터를 훈련하는 MPM(Main Processing Module);

양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 양자화기;

양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 부동소수점 컨볼루션 계층; 및

완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 PXM(Pixel Memory)

를 포함하는 CNN 가속기.

청구항 7

제6항에 있어서,

상기 MPM는,

실제 범위 값을 정수 범위 값으로 변환하기 위한 스케일 계수 및 부동소수점 도메인 및 고정소수점 도메인에서 0을 정렬하기 위한 영점을 포함하는 양자화 파라미터를 활성화 계층, 가중치 계층, 바이어스 계층에 추가하여 훈련하는

CNN 가속기.

청구항 8

제6항에 있어서,

상기 양자화기는,

스케일 프로세스, 반올림 프로세스, 클립 프로세스 및 역양자 프로세스를 포함하고,

스케일 프로세스를 통해 스케일 계수와 영점을 사용하여 고정밀 부동소수점 도메인 값을 고정밀 고정소수점 도메인 값으로 매핑하고,

반올림 프로세스를 통해 가장 가까운 정수 값에 대한 실제 값을 근사하고,

클립 프로세스를 통해 양자화에 따른 비트폭 집합을 벗어나는 값을 감소시키고,

역양자 프로세스를 통해 스케일 계수와 영점을 사용하여 저정밀 고정 소수점 도메인 값을 고정밀 부동소수점 도메인 값으로 변환하는

CNN 가속기.

청구항 9

제6항에 있어서,

상기 부동소수점 컨볼루션 계층은,

정수 스케일러 및 정수 클리퍼를 포함하고,

정수 스케일러를 통해 스케일 계수를 사용하여 시프트 연산을 수행하고,

정수 클리퍼를 통해 양자화 비트 폭에 따라 텐서의 데이터 범위를 조정하며,

양자화에 따른 부동소수점 가중치 및 바이어스 파라미터를 정수 파라미터로 변환하는

CNN 가속기.

청구항 10

제6항에 있어서,

상기 PXM는,

제1 유형 잔차 블록, 제2 유형 잔차 블록, 제3 유형 잔차 블록 및 제4 유형 잔차 블록을 포함하고,

제1 유형 잔차 블록을 통해 스킵 연결 경로와 ReLU 함수를 통과하는 주 경로 데이터를 추가하고,

제2 유형 잔차 블록을 통해 상기 제1 유형 잔차 블록의 스킵 연결 추가 후 메인 경로의 ReLU를 이동시키고,

제3 유형 잔차 블록을 통해 부동소수점 컨볼루션 계층을 사용하여 스킵 경로의 텐서 크기를 메인 경로와 동일하게 설정한 후 스킵 연결 추가를 수행하고,

제4 유형 잔차 블록을 통해 제3 유형 잔차 블록을 기준으로 스킵 경로의 부동소수점 컨볼루션 계층을 식별 계층으로 변경하는

CNN 가속기.

발명의 설명

기술 분야

[0001] 본 발명은 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기 및 그 동작 방법에 관한 것이다.

배경 기술

[0002] 교통 표지 인식(Traffic Sign Recognition; TSR) 기술은 차량이 카메라를 통해 도로표지를 인식해 주행에 활용할 수 있게 해준다. 교통안전을 위해 TSR은 고급 운전자 보조시스템(Advanced Driver Assistance Systems; ADAS)을 구성하는 핵심기술 중 하나이며, 여러 연구가 진행되고 있다. 컨볼루션 신경망(Convolutional Neural Networks; CNN)의 등장으로 자동차 환경, 특히 ADAS에서 새로운 가능성이 열렸다. 그러나 대부분의 CNN은 높은 연산 자원과 메모리 사용을 요구하기 때문에 자원이 제한된 ADAS에 실시간 TSR 애플리케이션을 배포하는 것은 어렵다. 이 문제를 해결하기 위해 임베디드 플랫폼의 최적화를 고려하기 위해 일부 작업이 연구되었지만, 기존 작업은 많은 하드웨어 자원을 사용했거나 낮은 연산 성능을 보였다.

[0003] 종래기술[1]은 임베디드 요건을 충족하면서 최적화된 정보 밀도로 CNN을 설계하기 위해 마이크로 아키텍처 및 매크로 아키텍처 레벨에서 통합 설계 전략 및 최적화 방법을 도입했다. 종래기술[2]은 CNN의 중단 간 교육, 양자화 및 배치를 위한 새로운 프레임워크를 제안했다. 제안된 프레임워크는 CNN을 정량화하고 마이크로컨트롤러에 배포한다. 종래기술[3]은 CNN의 파라미터 활용도를 높이는 방법을 제안하고 자원이 제한된 FPGA 플랫폼에 적합한 CNN 토폴로지를 만들기 위해 네 가지 최적화 전략을 채택했다. 종래기술[4]은 HLS(High-Level Synthesis)와 Xilinx PYNQ 플랫폼을 사용하여 빠르고 강력한 임베디드 애플리케이션을 구축하기 위해 맞춤 오버레이를 사용하여 하드웨어 가속기를 구현했다. 실험 결과는 실행 시간과 비용 측면에서 프로토타입 하드웨어의 효과를 입증한다. 종래기술[5]는 배치 정규화 및 지수 선형 유닛과 같은 복잡한 계산 대신 이진 가중치를 사용하여 효율적인 하드웨어 가속을 구현함으로써 포워드 패스에서 계산 비용을 줄이는 방법을 제안했다. 종래기술[6]은 네트

워크 각 부분의 기능 모듈을 합리적으로 분할하고 HLS를 이용하여 루프 설계를 최적화하여 처리량과 인식 속도를 향상시켰다. 일부 연구[3], [4], [6]는 하드웨어 자원이 많이 필요한 부동소수점 형식으로 구현되었다. 다른 연구[2]는 느린 추론 속도를 보였고, 이전 연구[5]는 낮은 인식 정확도를 보였다. 일부 연구[5], [6]는 TSR 작업에 10개의 클래스만 사용했다. 요약하면 임베디드 플랫폼에 구현된 기존 CNN 기반 TSR 작업은 실시간 추론이 필요하고 자원이 제한된 자동차 환경에 최적화되지 않았다.

선행기술문헌

특허문헌

- [0004] (특허문헌 0001) [1] A. Wong, M. J. Shafiee, and M. S. Jules, "MicronNet: A highly compact deep convolutional neural network architecture for real-time embedded traffic sign classification," IEEE Access, vol. 6, pp. 59803-59810, 2018.
- (특허문헌 0002) [2] P.-E. Novac, G. Boukli Hacene, A. Pegatoquet, B. Miramond, and V. Gripon, "Quantization and deployment of deep neural networks on microcontrollers," Sensors, vol. 21, no. 9, p. 2984, Apr. 2021.
- (특허문헌 0003) [3] Y. Yao, Z. Zhang, Z. Yang, J. Wang, and J. Lai, "FPGA-based convolution neural network for traffic sign recognition," in Proc. IEEE 12th Int. Conf. ASIC (ASICON), Oct. 2017, pp. 891-894.
- (특허문헌 0004) [4] A. Maraoui, S. Messaoud, S. Bouaafia, A. C. Ammari, L. Khriji, and M. Machhout, "PYNQ FPGA hardware implementation of LeNet-5-based traffic sign recognition application," in Proc. 18th Int. Multi-Conf. Syst., Signals Devices (SSD), Mar. 2021, pp. 1004-1009.
- (특허문헌 0005) [5] M. Lechner, A. Jantsch, and S. M. P. Dinakarrao, "ResCoNN: Resource-efficient FPGA-accelerated CNN for traffic sign classification," in Proc. 10th Int. Green Sustain. Comput. Conf. (IGSC), Oct. 2019, pp. 1-6.
- (특허문헌 0006) [6] J. Liu, Z. Zhang, L. Zheng, Y. Wen, F. Bin, and L. Tang, "Traffic sign recognition based on ZYNQ," in Proc. 9th Int. Symp. Next Gener. Electron. (ISNE), Jul. 2021, pp. 1-3.

발명의 내용

해결하려는 과제

- [0005] 본 발명이 이루고자 하는 기술적 과제는 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기 및 그 동작 방법을 제공하는데 있다. 연산 복잡성을 줄이기 위해 새로운 하드웨어 친화적인 양자화 방법을 확장하고, 제안된 저비용 CNN 기반 실시간 TSR 하드웨어 가속기는 실시간 추론과 자원 소비의 균형을 맞추기 위해 두 가지 병렬화 전략을 적용한다. 또한, 간차 블록의 스킵 연결 경로를 처리하는 간단하고 효과적인 하드웨어 설계 체계를 제안한다.

과제의 해결 수단

- [0006] 일 측면에 있어서, 본 발명에서 제안하는 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 동작 방법은 활성화 계층, 가중치 계층, 바이어스 계층에 양자화를 추가하여 양자화 파라미터를 훈련하는 단계, 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 단계, 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계 및 완전 정수 기반 간차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계를 포함한다.
- [0007] 상기 활성화 계층, 가중치 계층, 바이어스 계층에 양자화를 추가하여 양자화 파라미터를 훈련하는 단계는 실제 범위 값을 정수 범위 값으로 변환하기 위한 스케일 계수 및 부동소수점 도메인 및 고정소수점 도메인에서 0을 정렬하기 위한 영점을 포함하는 양자화 파라미터를 활성화 계층, 가중치 계층, 바이어스 계층에 추가하여 훈

련한다.

- [0008] 상기 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련을 수행하는 단계는 양자화기의 스케일 프로세스를 통해 스케일 계수와 영점을 사용하여 고정밀 부동소수점 도메인 값을 고정밀 고정소수점 도메인 값으로 매핑하고, 양자화기의 반올림 프로세스를 통해 가장 가까운 정수 값에 대한 실제 값을 근사하고, 양자화기의 클립 프로세스를 통해 양자화에 따른 비트폭 집합을 벗어나는 값을 감소시키고, 양자화기의 역양자 프로세스를 통해 스케일 계수와 영점을 사용하여 저정밀 고정소수점 도메인 값을 고정밀 부동소수점 도메인 값으로 변환한다.
- [0009] 상기 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계는 정수 스케일러를 통해 스케일 계수를 사용하여 시프트 연산을 수행하고, 정수 클리퍼를 통해 양자화 비트 폭에 따라 텐서의 데이터 범위를 조정하며, 양자화에 따른 부동소수점 가중치 및 바이어스 파라미터를 정수 파라미터로 변환한다.
- [0010] 상기 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계는 제1 유형 잔차 블록을 통해 스킵 연결 경로와 ReLU 함수를 통과하는 주 경로 데이터를 추가하고, 제2 유형 잔차 블록을 통해 상기 제1 유형 잔차 블록의 스킵 연결 추가 후 메인 경로의 ReLU를 이동시키고, 제3 유형 잔차 블록을 통해 부동소수점 컨볼루션 계층을 사용하여 스킵 경로의 텐서 크기를 메인 경로와 동일하게 설정한 후 스킵 연결 추가를 수행하고, 제4 유형 잔차 블록을 통해 제3 유형 잔차 블록을 기준으로 스킵 경로의 부동소수점 컨볼루션 계층을 식별 계층으로 변경한다.
- [0011] 또 다른 일 측면에 있어서, 본 발명에서 제안하는 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기는 활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련하는 MPM(Main Processing Module), 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 양자화기, 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 부동소수점 컨볼루션 계층 및 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 PXM(Pixel Memory)를 포함한다.

발명의 효과

- [0012] 본 발명의 실시예들에 따르면 연산 복잡성을 줄이기 위해 새로운 하드웨어 친화적인 양자화 방법을 확장하고, 제안하는 양자화 방법은 잔차 블록의 스킵 연결 경로를 포함한 모든 연산이 정수 산술만 사용하도록 CNN을 재구성할 수 있으며, 양자화 아핀(affine) 매핑 프로세스를 시프트 연산으로 대체하여 연산 오버헤드를 줄일 수 있다. 제안된 하드웨어 가속기는 실시간 추론과 자원 소비의 균형을 맞추기 위해 두 가지 병렬화 전략을 적용하고, 잔차 블록의 스킵 연결 경로를 처리하는 간단하고 효과적인 하드웨어 설계를 통해 스킵 연결 경로의 데이터 흐름을 최적화하고 추가 내부 메모리 사용량을 줄일 수 있다.

도면의 간단한 설명

- [0013] 도 1은 본 발명의 일 실시예에 따른 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 동작 방법을 설명하기 위한 흐름도이다.
- 도 2는 본 발명의 일 실시예에 따른 완전 정수 기반 컨볼루션 계층을 설명하기 위한 도면이다.
- 도 3은 본 발명의 일 실시예에 따른 잔여 블록의 유형을 설명하기 위한 도면이다.
- 도 4는 본 발명의 일 실시예에 따른 양자화기를 갖는 잔여 블록을 설명하기 위한 도면이다.
- 도 5는 본 발명의 일 실시예에 따른 완전 정수 기반 Type-B 잔여 블록을 설명하기 위한 도면이다.
- 도 6은 본 발명의 일 실시예에 따른 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 전체 아키텍처를 나타내는 도면이다.
- 도 7은 본 발명의 일 실시예에 따른 TSR 가속기의 하드웨어 구성 요소를 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0014] 본 발명에서는 저비용 CNN 기반 실시간 TSR 하드웨어 가속기를 제안한다. 첫째, 연산 복잡성을 줄이기 위해 새

로운 하드웨어 친화적인 양자화 방법을 확장한다. 양자화 방법은 잔차 블록의 스킵 연결 경로를 포함한 모든 연산이 정수 산술만 사용하도록 CNN을 재구성할 수 있으며, 양자화 아핀(affine) 매핑 프로세스를 시프트 연산으로 대체하여 연산 오버헤드를 줄일 수 있다. 둘째, 제안된 하드웨어 가속기는 실시간 추론과 자원 소비의 균형을 맞추기 위해 두 가지 병렬화 전략을 적용하였다. 또한, 잔차 블록의 스킵 연결 경로를 처리하는 간단하고 효과적인 하드웨어 설계 체계를 제시한다. 이 설계 방식은 스킵 연결 경로의 데이터 흐름을 최적화하고 추가 내부 메모리 사용량을 줄일 수 있다. 실험 결과에 따르면 재구성된 완전 정수 기반 CNN은 24M IOP(Integer Operations)만 필요하며 0.17MB의 모델 크기를 가지고 있다. 종래기술에 비해 제안된 CNN 모델 크기는 $\times 105$ 배, 작업 수는 $\times 58$ 배 감소했다. 또한 제안된 CNN은 임베디드 플랫폼에 구현된 CNN 기반 TSR 작업 중 가장 높은 정확도인 99.07%의 TSR 정확도를 달성할 수 있다. 제안된 하드웨어 가속기는 Xilinx ZC706 SoC에 구현될 때 960 MOPS의 연산 성능과 40 FPS의 프레임 속도를 달성한다. 따라서 본 발명에서 제안하는 기법은 종래기술에 비해 연산 성능과 프레임률이 $\times 11.87$ 배, $\times 36.7$ 배 향상된다. 이하, 본 발명의 실시 예를 첨부된 도면을 참조하여 상세하게 설명한다.

- [0015] 본 발명의 실시예에 따른 실시간 TSR을 위한 저비용 완전 정수 기반 CNN 가속기는 새로운 하드웨어 친화적인 양자화 방법을 확장하여 잔차 블록(Residual Blocks; RB)에 적용하였다. 하드웨어 친화적인 양자화 방법은 잔차 블록의 스킵 연결 경로를 포함한 모든 작업이 정수 산술만 사용하도록 CNN을 재구성할 수 있다. 또한 양자화 아핀 매핑 프로세스에 필요한 모든 작업을 시프트 작업으로 대체할 수 있다.
- [0016] 본 발명의 실시예에 따르면, 제안된 하드웨어 가속기는 실시간 추론과 자원 소비의 균형을 맞추기 위해 두 가지 병렬화 전략(커널 및 출력 채널 병렬화)을 적용한다. 또한, 잔차 블록의 스킵 연결 경로를 처리하는 간단하고 효과적인 하드웨어 설계 체계를 제시한다. 이러한 설계 방식, 즉 내부 메모리 스킵 연결 경로는 스킵 연결 경로의 데이터 흐름을 최적화하고 추가적인 내부 메모리 사용량을 줄일 수 있다.
- [0018] 도 1은 본 발명의 일 실시예에 따른 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 동작 방법을 설명하기 위한 흐름도이다.
- [0019] 제안하는 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 동작 방법은 활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련하는 단계(110), 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(Quantization-Aware Training; QAT)을 수행하는 단계(120), 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성하는 단계(130) 및 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행하는 단계(140)를 포함한다.
- [0020] 단계(110)에서, 활성화 계층, 가중치 계층, 바이어스 계층에 양자화기를 추가하여 양자화 파라미터를 훈련한다.
- [0021] 실제 범위 값을 정수 범위 값으로 변환하기 위한 스케일 계수 및 부동소수점 도메인 및 고정소수점 도메인에서 0을 정렬하기 위한 영점을 포함하는 양자화 파라미터를 활성화 계층, 가중치 계층, 바이어스 계층에 추가하여 훈련한다.
- [0022] 단계(120)에서, 양자화 파라미터를 사용하여 부동소수점 도메인 값을 고정소수점 도메인 값으로 매핑하고, CNN 추론을 위한 양자화 인식 훈련(QAT)을 수행한다.
- [0023] 상기 양자화기는 스케일 프로세스, 반올림 프로세스, 클립 프로세스 및 역양자 프로세스를 포함한다.
- [0024] 양자화기의 스케일 프로세스를 통해 스케일 계수와 영점을 사용하여 고정밀 부동소수점 도메인 값을 고정밀 고정소수점 도메인 값으로 매핑하고, 양자화기의 반올림 프로세스를 통해 가장 가까운 정수 값에 대한 실제 값을 근사한다.
- [0025] 그리고, 양자화기의 클립 프로세스를 통해 양자화에 따른 비트폭 집합을 벗어나는 값을 감소시키고, 양자화기의 역양자 프로세스를 통해 스케일 계수와 영점을 사용하여 저정밀 고정 소수점 도메인 값을 고정밀 부동소수점 도메인 값으로 변환한다.
- [0026] 단계(130)에서, 부동소수점 컨볼루션 계층을 통해 양자화 인식 훈련을 통해 획득한 양자화 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성한다.
- [0027] 상기 부동소수점 컨볼루션 계층은 정수 스케일러 및 정수 클리퍼를 포함한다.

- [0028] 부동소수점 컨볼루션 계층의 정수 스케일러를 통해 스케일 계수를 사용하여 시프트 연산을 수행하고, 부동소수점 컨볼루션 계층의 정수 클리퍼를 통해 양자화 비트 폭에 따라 텐서의 데이터 범위를 조정한다. 그리고, 양자화에 따른 부동소수점 가중치 및 바이어스 파라미터를 정수 파라미터로 변환한다.
- [0029] 단계(140)에서, 완전 정수 기반 잔차 블록의 스킵 연결 경로를 이용하여 스킵 연결 추가를 수행한다.
- [0030] 완전 정수 기반 잔차 블록은 제1 유형 잔차 블록(Type-A RB)을 통해 스킵 연결 경로와 ReLU 함수를 통과하는 주 경로 데이터를 추가하고, 제2 유형 잔차 블록(Type-B RB)을 통해 상기 제1 유형 잔차 블록의 스킵 연결 추가 후 메인 경로의 ReLU를 이동시킨다. 그리고, 제3 유형 잔차 블록(Type-C RB)을 통해 부동소수점 컨볼루션 계층을 사용하여 스킵 경로의 텐서 크기를 메인 경로와 동일하게 설정한 후 스킵 연결 추가를 수행하고, 제4 유형 잔차 블록(Type-D RB)을 통해 제3 유형 잔차 블록을 기준으로 스킵 경로의 부동소수점 컨볼루션 계층을 식별 계층으로 변경한다.
- [0032] 도 2는 본 발명의 일 실시예에 따른 완전 정수 기반 컨볼루션 계층을 설명하기 위한 도면이다.
- [0033] 양자화는 고정밀 부동소수점 도메인을 고정밀 고정소수점 도메인으로 변환하여 효율적인 CNN 추론을 가능하게 한다. 양자화는 양자화 인식 훈련(Quantization-Aware Training; QAT)과 훈련 후 양자화(Post-Training Quantization; PTQ)로 분류할 수 있다. QAT는 미세 조정 과정에서 양자화 파라미터를 업데이트하는 방법이다. PTQ는 추론 중 양자화 파라미터를 보정하여 부동소수점과 고정소수점 도메인 간의 양자화 오류를 줄이는 방법이다.
- [0034] 본 발명에서는 정확도의 손실을 최소화하기 위해 QAT 방법을 적용한다. 양자화를 위해 스케일 계수, 영점의 두 개의 파라미터가 추가된다. 스케일 계수는 실제 범위 값을 정수 범위 값으로 변환하는 데 필요한 파라미터이며, 영점은 두 도메인에서 0을 정렬하는 데 필요한 파라미터이다. 양자화 파라미터는 활성화 계층, 가중치 계층, 바이어스 계층에 양자화를 추가하여 훈련된다. 양자화는 주로 4개의 하위 프로세스(스케일, 반올림, 클립, 역양자화)로 구성된다. 우선, 스케일 프로세스는 스케일 계수와 영점을 사용하여 고정밀 부동소수점 도메인 값을 고정밀 고정소수점 도메인 값으로 매핑한다. 스케일 프로세스에 필요한 affine 매핑 방정식은 다음과 같다:
- [0035]
$$r = sf(q - zp) \quad (1)$$
- [0036] 여기서 r 은 고정밀 부동소수점 도메인 값, q 는 저정밀 고정소수점 도메인 값, sf 는 스케일 계수, zp 은 영점이다.
- [0037] 다음으로, 반올림 프로세스는 가까운 정수 값에 실제 값을 근사한다. 라운드가 과정에서 기울기 불연속성 문제 직렬 추정자(Straight-Through Estimator; STE)를 사용하여 해결될 수 있다.
- [0038] 클립 프로세스를 통해 양자화 단계에 있는 비트폭 집합을 벗어나는 값을 줄일 수 있다.
- [0039] 마지막으로 역양자 프로세스에서 스케일 계수와 영점을 활용하여 저정밀 고정 소수점 도메인 값을 다시 고정밀 부동소수점 도메인 값으로 변환한다. 양자화 아핀 매핑 프로세스에서 연산 오버헤드를 줄이기 위해서 본 발명에서는 LLTQ, 새로운 하드웨어 친화적 양자화 방법을 도입했다. LLTQ 방법으로 모든 zp 값을 0으로 설정하여 추가 연산을 제거할 수 있다. 또한 LLTQ는 sf 값을 2의 거듭제곱 값에 매핑함으로써 곱셈 연산을 시프트 연산으로 대체할 수 있다. 따라서 아핀 매핑 프로세스에는 시프트 연산만 필요하다. LLTQ 방법에서 스케일 계수를 2의 거듭제곱에 직접 매핑하는 방정식은 다음과 같다:
- [0040]
$$sf = \frac{2^{\lceil \log_2(\text{abs}(\alpha * q_level)) \rceil}}{q_level} \quad (2)$$
- [0041] 여기서 α 는 양자화를 위한 훈련 가능한 파라미터이고, q_level 은 양자화 동안 설정된 비트 폭으로 나타낼 수 있는 최대 정수 값이다(예를 들어, 부호 있는 8비트 데이터: $q_level = 128$, 부호 없는 8비트 데이터: $q_level = 255$).
- [0042] 부동소수점 컨볼루션 계층(CONV)은 양자화 인식 훈련(QAT) 프로세스에서 얻은 파라미터를 사용하여 완전 정수 기반 컨볼루션으로 재구성할 수 있다. 재구성된 CONV는 도 2와 같다. 도 2에서 SC는 정수 스케일러 및 정수 클리퍼를 갖는 블록이다. 정수 스케일러는 스케일 계수를 사용하여 시프트 연산을 수행한다. 정수 클리퍼는 양자

화 비트 폭에 따라 텐서의 데이터 범위를 조정한다. 예를 들어, 양자화 비트가 8비트이고 부호 있는 데이터의 값이 $-128 \sim 127$ 범위를 초과하는 경우, 데이터 범위의 마지막 값(-128 또는 127)으로 위치하도록 조정된다. ICONV와 SC는 하나의 계층으로 구성되어 있기 때문에 계층 간의 경계가 명확해지고 입출력도 동일한 비트폭이 된다. 이러한 방법은 보다 균일한 하드웨어 설계를 가능하게 하며 많은 하드웨어 모듈을 변경하지 않고도 다양한 유형의 CNN을 구현할 수 있게 한다. 양자화에 영향을 받는 부동소수점 가중치 및 바이어스 파라미터는 다음 방정식을 사용하여 정수 파라미터로 변환할 수 있다:

$$W_{int} = clip(round(W_{fp} \ll S_w, R)) \quad (3)$$

$$B_{int} = clip(round(B_{fp} \ll S_b, R)) \gg (S_{a1} + S_w - S_b) \quad (4)$$

여기서, S_{a1} 은 활성화 스케일 계수, S_b 는 바이어스 스케일 계수, S_w 는 가중치 스케일 계수, R 은 양자화 범위이다. 정수 가중치 및 바이어스 파라미터는 추론 프로세스 전에 변환된다. 재구성된 완전 정수 기반 CONV의 전체 연산 프로세스는 다음과 같다:

$$M_{int} = clip(C \gg (S_{a1} + S_w - S_{a2}), R) \quad (5)$$

도 3은 본 발명의 일 실시예에 따른 잔여 블록의 유형을 설명하기 위한 도면이다.

ResNet에 도입된 잔차 블록(RB)의 스킵 연결 경로는 바니싱(vanishing) 경사 문제를 완화하고 더 깊은 네트워크를 설계할 수 있음을 보여준다. 임베디드 TSR 환경에서 최고의 인식 정확도를 가진 RB를 찾기 위해 도 3에 표시된 4가지 유형의 RB를 구성했다. Type-A RB는 스킵 경로와 ReLU 함수(ReLU)를 통과하는 주 경로 데이터를 추가한다. Type-B RB는 Type-A RB에서 스킵 연결 추가 후 메인 경로의 ReLU가 이동하는 구조이다. Type-C RB는 CONV를 사용하여 스킵 경로의 텐서 크기를 메인 경로와 동일하게 설정한 후 스킵 연결 추가를 수행한다. Type-D RB는 타입-C RB를 기준으로 스킵 경로의 COV가 식별 계층(ID)[13]으로 변경되는 구조이다.

도 4는 본 발명의 일 실시예에 따른 양자화기를 갖는 잔여 블록을 설명하기 위한 도면이다.

LLTQ 방법을 확장하여 4개의 RB에 적용하였다. 양자화기를 포함한 RB는 도 4에 도시되어 있으며, CONV의 입력 포트와 각 RB의 스킵 연결 경로에 양자화기를 삽입하였다. 양자화 프로세스에서 사용되는 양자화기의 종류는 양자화기의 입력 부호에 따라 달라진다. 서로 다른 경로의 입력이 동일한 부호를 가질 경우 공유 양자화기를 사용할 수 있다. 도 4의 4개의 RB 모두에서 스킵 경로와 주 경로의 시작점은 모두 부호 없는 값이다. ReLU를 통과한 데이터가 분기되어 있기 때문이다. 따라서 이러한 데이터는 부호 없는 양자화기를 공유할 수 있으며 공유 양자화기는 두 번째 CONV 및 스킵 경로에 걸쳐 사용된다. 모든 CONV 입력이 부호가 없으므로 Type-A RB는 부호 없는 양자화기만 사용한다. 주 경로의 세 번째 CONV 출력은 부호 있는 데이터이기 때문에 Type-B RB는 부호 있는 양자화기를 사용한다. Type-C RB에서는 스킵 연결 추가를 위한 모든 데이터가 부호 있는 데이터이므로 부호 있는 양자화기를 공유하고 사용할 수 있다. 도 4(c)에는 주 경로의 스킵 경로 CONV 및 제2 CONV의 이후에 부호 있는 공유 양자화기가 도시되어 있다. 도 4(d)의 Type-D RB의 ID는 산술 연산이 없으므로 양자화기를 추가할 필요가 없다.

도 5는 본 발명의 일 실시예에 따른 완전 정수 기반 Type-B 잔여 블록을 설명하기 위한 도면이다.

설명하는 바와 같이, Type-B RB의 정확도가 가장 우수하기 때문에, 4개의 RB 중 Type B-RB의 정수 스킵 연결 경로만을 설명한다. 도 5에는 완전 정수 기반의 Type-B RB가 도시되어 있다. 부동소수점 버전에서, RB의 스킵 연결 추가는 단순히 두 계층의 출력을 추가한다. 그러나 정수 스킵 연결을 추가하려면 두 경로의 크기를 동일하게 설정해야 정확하게 작동한다. 이 과정을 경로 등가 스케일링이라고 하며 정수 스킵 연결 추가에 사용되는 두 경로에 대한 방정식은 (6)과 (7)과 같다:

$$F_{2,sk} = (F_2 \ll S_{a2}) \gg S_{as4} \quad (6)$$

[0057]
$$M_{3,sk} = (M_3 \ll S_{a4}) \gg S_{as4} \quad (7)$$

[0058] 여기서 S_{a2} 와 S_{a4} 는 다음 CONV의 활성화 스케일 계수이고 S_{as4} 는 주 경로의 세 번째 CONV의 출력 측에 위치한 양자화기에서 학습된 스케일 계수이다. 경로 증가 스케일링 프로세스는 다음 계층의 활성화 스케일 계수의 효과를 취소한 후 S_{as4} 를 이용하여 두 경로(F_2 , M_3)를 동일한 스케일로 조정한다. 동일한 스케일 데이터의 추가는 (8)에서 확인할 수 있다:

[0059]
$$M_{3,add} = F_{2,sk} + M_{3,sk} \quad (8)$$

[0060] 스킵 연결 추가 후 다음 계층의 입력 스케일에 따라 다운스케일링이 수행된다:

[0061]
$$M_{3,ds} = clip(M_{3,add} \ll S_{as4}) \gg S_{a4}, R) \quad (9)$$

[0062] 마지막으로 다운스케일링 후 정수 스킵 연결 추가가 종료된다.

[0064] 도 6은 본 발명의 일 실시예에 따른 실시간 교통 표지판 인식을 위한 FPGA 기반 저비용 정수 연산 CNN 가속기의 전체 아키텍처를 나타내는 도면이다.

[0065] 제안된 TSR 가속기는 이중 SoC에서 구현되며, 도 6에 도시되어 있다. 호스트 CPU는 AXI Lite 인터페이스를 사용하여 FPGA에 구현된 CNN 로직을 제어한다. AXI 인터페이스는 외부 메모리와 가속기 간의 데이터 교환에 사용된다. 각 계층의 입력 이미지, 파라미터, 출력 값은 외부 메모리에 저장된다. 제안된 가속기는 내부 메모리 모듈(Internal Memory Module; IMM)과 처리 모듈로 구분되며, 처리 모듈은 메인 프로세싱 모듈(Main Processing Module; MPM)과 서브 프로세싱 모듈(Sub Processing Module; SPM)로 더욱 구분되며, 각 하드웨어 구성 요소는 도 7을 참조하여 더욱 상세히 설명한다.

[0067] 도 7은 본 발명의 일 실시예에 따른 TSR 가속기의 하드웨어 구성 요소를 나타내는 도면이다.

[0068] IMM은 파라미터 메모리(Into Parameter Memory; PRM)와 픽셀 메모리(Pixel Memory; PXM)로 구분된다. PRM은 IMM에서 가장 큰 크기를 차지하며 작업이 수행되는 계층의 모든 파라미터를 저장한다. PRM은 효율적인 병렬 MAC 작업을 위해 분할되고 각 PE에 할당된다. 정수 스킵 연결 추가 구현에는 두 가지 문제가 있다. 첫째, 스킵 경로 데이터를 내부 메모리에 저장하면 내부 메모리 사용량이 증가한다. 둘째, 프로세싱 모듈에서 동작하는 동안 스킵 경로 데이터를 가져오기 위해 외부 메모리에 접근하는 것은 병목 현상이 될 수 있다. 이러한 문제점을 해결하기 위해 프로세싱 모듈에서 내부 메모리 모듈로 정수 스킵 연결 추가가 이동되었다. 도 7(a)에 도시된 PXM은 RB의 정수 스킵 연결 경로를 구현하기 위한 중요한 구성요소이다. 본 발명에서 도 5에 설명된 스킵 연결 가산기는 SPM이 아닌 PXM에서 구현된다. 즉, 스킵 연결 추가는 현재 계층의 출력 측인 SPM이 아니라 다음 계층의 입력 측인 PXM에서 수행된다. ctrl_skip 신호가 수신되면 Skip Path와 Main Path 모두 외부 메모리에 액세스하여 작업 결과(6)-(9)를 FIFO 메모리로 전송한다. 그렇지 않으면 외부 메모리 액세스는 기본 경로에서만 수행되고 이 데이터는 FIFO 메모리로 전송된다. SPM이 아닌 PXM에 스킵 연결 추가가 구현되므로 Skip Path 데이터를 저장하기 위해 추가 내부 메모리를 추가할 필요가 없다. 따라서, 내부 메모리 스킵 연결 경로 설계 방식은 프로세싱 모듈의 외부 메모리 액세스를 제거하여 스킵 연결 경로의 데이터 흐름을 최적화할 수 있으며, 보다 균일한 하드웨어 설계를 가능하게 한다.

[0069] 도 7(b)는 제안된 TSR 가속기에서 MAC 동작을 수행하는 MPM(Main Processing Module)이다. 하나의 PE는 9개의 곱셈기와 9개의 덧셈기로 구성되며, MPM은 PE의 집합이다. 실시간 TSR을 달성하면서 자원 사용을 최소화하기 위해 MPM에는 많은 병렬화 전략 중 2개만 적용되었다. 먼저 각 커널을 병렬로 계산하기 위해 커널별 병렬화 전략을 도입했다. 이 전략은 계산 루프를 완전히 풀어서 3×3 커널에 필요한 작업을 한 사이클로 수행한다. 도 7(b)에 도시된 바와 같이, 입력 피쳐 맵의 한 채널의 MAC 동작에는 9개의 곱셈기와 덧셈기 트리가 사용된다. MAC 연산 결과는 덧셈기 트리 뒤의 버퍼에 일시적으로 저장된다. CONV의 커널 크기가 1이면 9개의 MPM 곱셈기 중 하나만 작동한다.

[0070] 또한, 입력 피쳐 맵의 다른 채널에 대한 MAC 연산 결과는 버퍼 바로 앞에 위치한 덧셈기를 이용하여 이전 사이클에서 계산된 MAC 연산 결과로 누적된다. 둘째, 출력 채널 병렬화 전략을 적용하여 서로 다른 필터의 MAC 작업

을 동시에 수행하였다. 이 전략은 PE 수에 따라 CONV의 필터 루프를 부분적으로 전개하기 때문에 MPM에 여러 PE가 구현된다. 위에서 언급한 두 가지 병렬화 전략은 데이터 의존성이 없으며 간단하게 구현할 수 있다. 또한 다른 병렬화 전략 조합에 비해 연산 성능을 효과적으로 향상시키면서 하드웨어 자원을 적게 사용하므로 임베디드 환경에 적합한 실시간 추론이 가능하다. 도 7(c)는 MAC 동작을 제외한 나머지 동작을 수행하는 SPM이다. SPM은 바이어스 덧셈기, ReLU 함수(ReLU), 시프터, 클리퍼(CLP), 글로벌 평균 풀링(GAP) 모듈로 구성된다. MPM과 동기화된 병렬 후 처리를 수행하기 위해 PE 수에 따라 SPM도 부분적으로 전개하였다. 마지막 계층은 CLP 대신 ctrl_last 신호를 수신하고 GAP 모듈을 사용한다.

[0072] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 컨트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 컨트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0073] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치에 구체화(embodiment)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

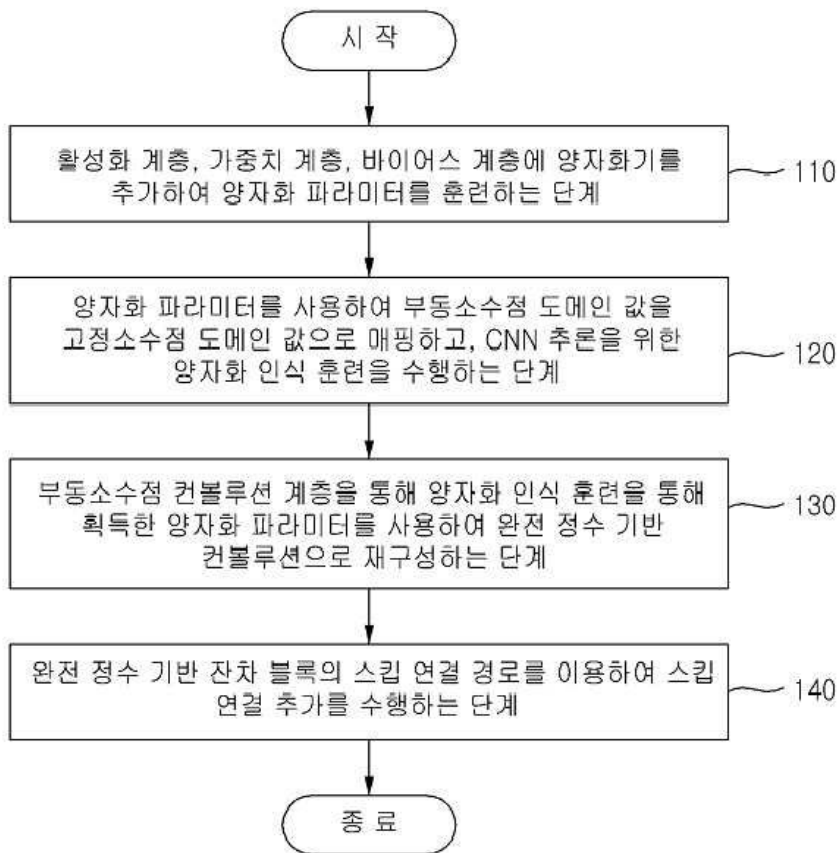
[0074] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다.

[0075] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

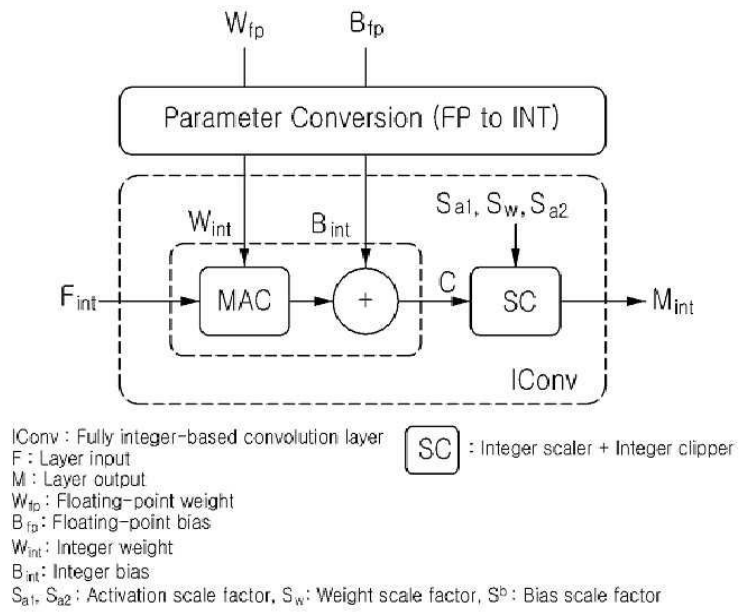
[0076] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

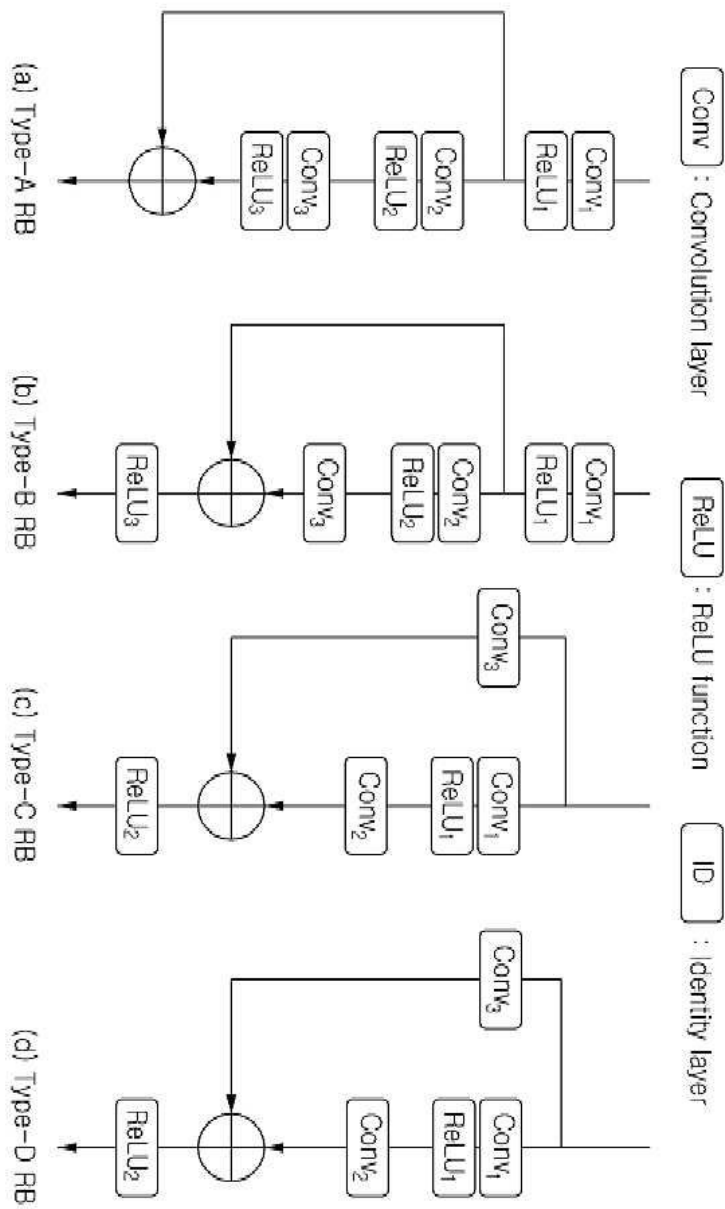
도면1



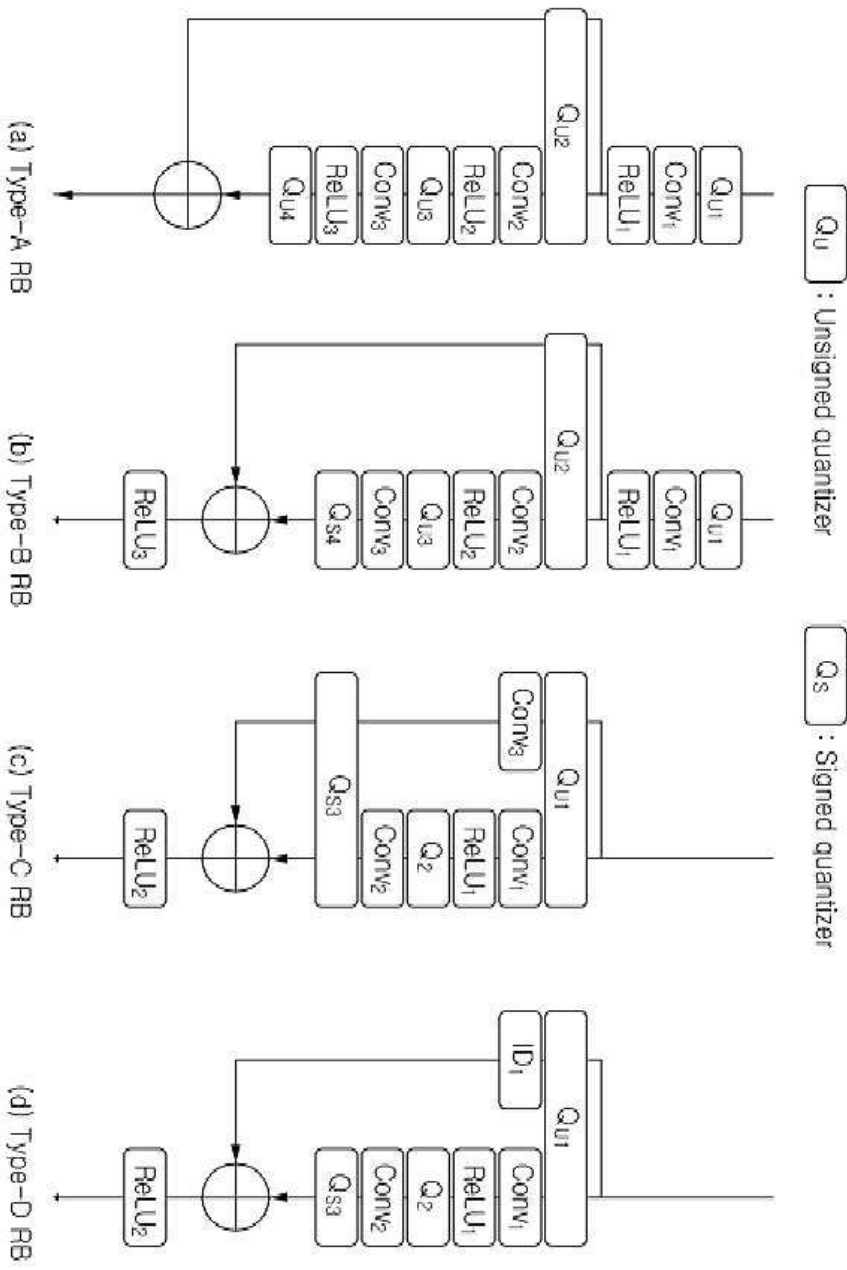
도면2



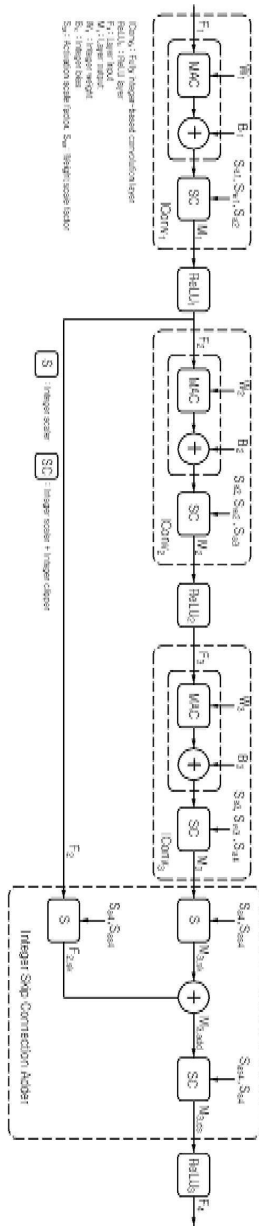
도면3



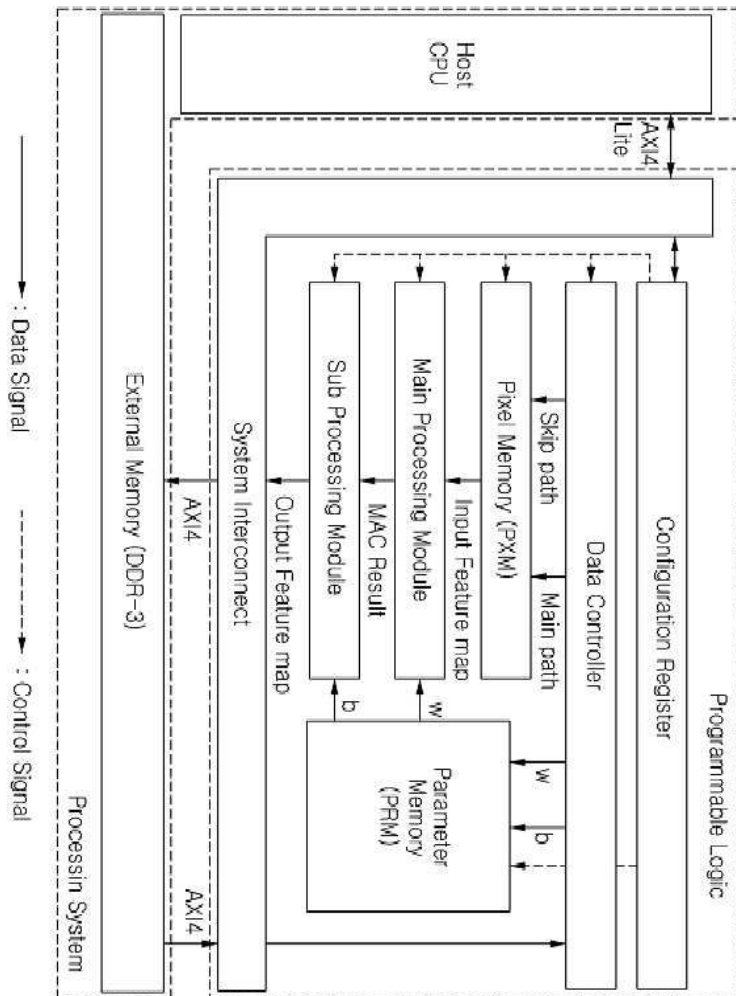
도면4



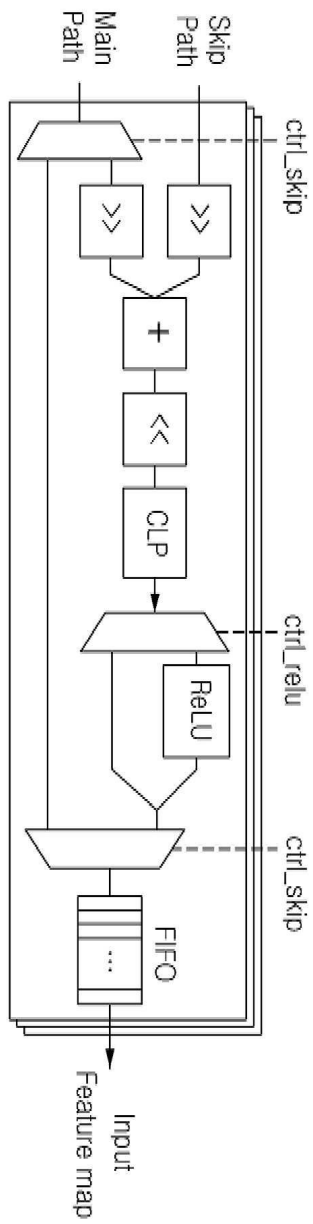
도면5



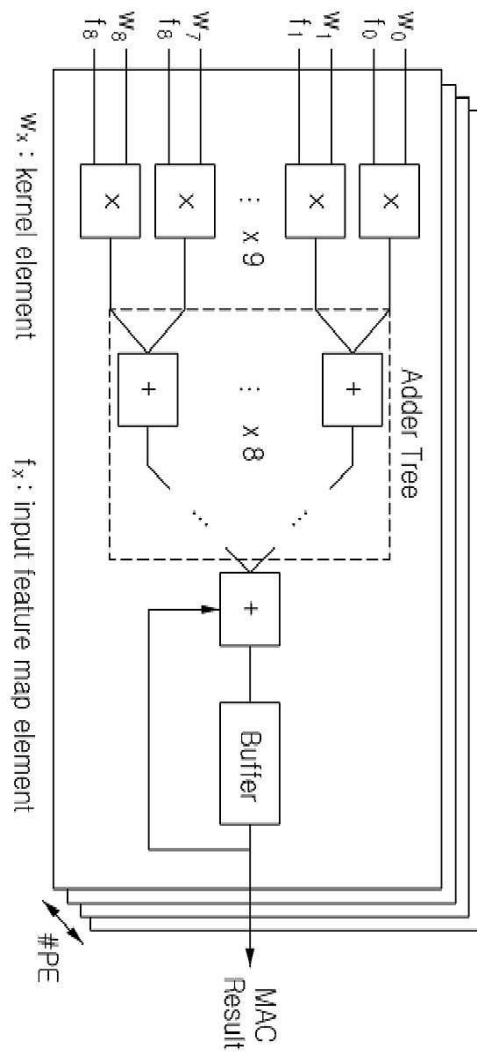
도면6



도면7a



도면7b



도면7c

