



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0053199
(43) 공개일자 2023년04월21일

(51) 국제특허분류(Int. Cl.)
G06N 3/08 (2023.01) G06N 3/04 (2023.01)
G06N 3/063 (2023.01)
(52) CPC특허분류
G06N 3/08 (2023.01)
G06N 3/04 (2023.01)
(21) 출원번호 10-2021-0136456
(22) 출원일자 2021년10월14일
심사청구일자 2021년10월14일

(71) 출원인
한국항공대학교산학협력단
경기도 고양시 덕양구 항공대학로 76 (화전동, 한국항공대학교)
(72) 발명자
김태환
경기도 고양시 덕양구 항공대학로 76 (화전동, 한국항공대학교)
박상준
서울특별시 금천구 범안로11길 4, 703호 (독산동)
(74) 대리인
안병규

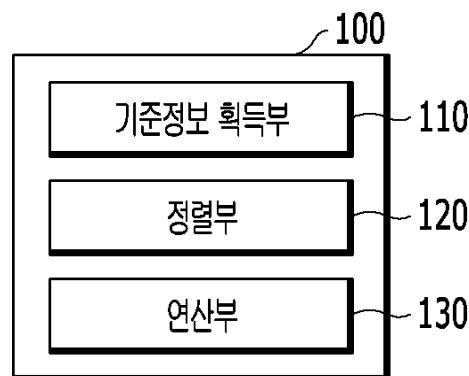
전체 청구항 수 : 총 15 항

(54) 발명의 명칭 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 연산 속도 향상 장치 및 방법

(57) 요약

이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치 및 방법이 개시되며, 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법은, (a) 상기 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득하는 단계, (b) 상기 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정하는 단계 및 (c) 상기 결정된 연산 순서에 기초하여 상기 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 상기 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 상기 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 상기 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정하는 단계를 포함할 수 있다.

대표도 - 도1



(52) CPC특허분류

G06N 3/063 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711126306
과제번호	2017-0-00528-005
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	ICT융합산업혁신기술개발
연구과제명	다중 모드 스마트 레이더용 지능형 반도체 개발 기초 연구실
기 여 율	1/1
과제수행기관명	한국항공대학교교산학협력단
연구기간	2021.01.01 ~ 2021.12.31

명세서

청구범위

청구항 1

이진화 컨볼루션 신경망의 공간적 인접성을 이용한 연산 속도 향상 방법에 있어서,

(a) 상기 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득하는 단계;

(b) 상기 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정하는 단계; 및

(c) 상기 결정된 연산 순서에 기초하여 상기 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 상기 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 상기 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 상기 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정하는 단계,

를 포함하는 것인, 연산 속도 향상 방법.

청구항 2

제1항에 있어서,

상기 (c) 단계는,

상기 참조 풀링 결과가 0이고, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 0으로 결정하는 것인, 연산 속도 향상 방법.

청구항 3

제1항에 있어서,

상기 (c) 단계는,

상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 1로 결정하는 것인, 연산 속도 향상 방법.

청구항 4

제1항에 있어서,

상기 (b) 단계는,

(b1) 상기 참조 풀링 결과가 0이면, 상기 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를 유지하는 단계; 및

(b2) 상기 참조 풀링 결과가 1이면, 상기 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 상기 이전 풀링 윈도우에서의 위치인 기준 위치에 기초하여 상기 연산 순서를 갱신하는 단계,

를 포함하는 것인, 연산 속도 향상 방법.

청구항 5

제4항에 있어서,

상기 (b2) 단계는,

상기 이전 풀링 윈도우와 상기 현재 풀링 윈도우의 윈도우 진행 방향 및 상기 기준 위치에 기초하여 상기 연산 순서의 시작점을 결정하는 것인, 연산 속도 향상 방법.

청구항 6

제5항에 있어서,

상기 시작점은 상기 결정된 연산 순서에 의할 때 상기 윈도우 진행 방향을 고려하여 상기 기준 위치로부터 근접한 뉴런부터 연산이 이루어지도록 결정되는 것인, 연산 속도 향상 방법.

청구항 7

제5항에 있어서,

상기 (b2) 단계는,

상기 윈도우 진행 방향에 기초하여 상기 연산 순서의 패턴을 결정하는 것인, 연산 속도 향상 방법.

청구항 8

제2항에 있어서,

상기 임계값은 풀링 윈도우에 포함되는 복수의 뉴런의 수에 기초하여 결정되는 것을 특징으로 하는, 연산 속도 향상 방법.

청구항 9

이진화 컨볼루션 신경망의 공간적 인접성을 이용한 연산 속도 향상 장치에 있어서,

상기 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득하는 기준정보 획득부;

상기 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정하는 정렬부; 및

상기 결정된 연산 순서에 기초하여 상기 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 상기 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 상기 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 상기 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정하는 연산부,

를 포함하는, 연산 속도 향상 장치.

청구항 10

제9항에 있어서,

상기 연산부는,

상기 참조 풀링 결과가 0이고, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 0으로 결정하는 것인, 연산 속도 향상 장치.

청구항 11

제9항에 있어서,

상기 연산부는,

상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 1로 결정하는 것인, 연산 속도 향상 장치.

청구항 12

제9항에 있어서,

상기 정렬부는,

상기 참조 풀링 결과가 0이면, 상기 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를 유지하고, 상기 참조 풀링 결과가 1이면, 상기 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 상기 이전 풀링 윈도우에서의 위치인 기준 위치에 기초하여 상기 연산 순서를 갱신하는 것인, 연산 속도 향상 장치.

청구항 13

제12항에 있어서,

상기 정렬부는,

상기 이전 풀링 윈도우와 상기 현재 풀링 윈도우의 윈도우 진행 방향 및 상기 기준 위치에 기초하여 상기 연산 순서의 시작점을 결정하는 것인, 연산 속도 향상 장치.

청구항 14

제13항에 있어서,

상기 정렬부는,

상기 윈도우 진행 방향에 기초하여 상기 연산 순서의 패턴을 결정하는 것인, 연산 속도 향상 장치.

청구항 15

제1항 내지 제8항 중 어느 한 항의 방법을 컴퓨터에서 실행하기 위한 프로그램을 기록한 컴퓨터에서 판독 가능한 기록매체.

발명의 설명

기술 분야

[0001] 본원은 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치 및 방법에 관한 것이다.

배경 기술

[0002] 이진화 컨볼루션 신경망(Binarized Convolution Neural Network, BCNN)은 하나의 비트로 각 뉴런과 가중치를 나타낸 컨볼루션 신경망이다. 기존의 컨볼루션 신경망(CNN)은 floating point 및 integer 연산을 하고 메모리 사용량이 상당히 많이 요구되는 문제점이 있다. 또한, 종래의 CNN은 상당히 많은 수의 MAC(Multiply-and-accumulate) 연산을 필요로 한다.

[0003] 이러한 기존 CNN의 문제점을 해결하기 위해 고안된 BCNN은 floating point 연산 및 integer 연산이 이루어지는 MAC 연산을 XNOR-bitcount 연산으로 대체하여 빠른 추론이 가능하며, 바이트 단위 뉴런과 가중치 대신에 하나의 비트로 뉴런, 가중치를 표현하기 때문에 메모리 사용량도 매우 적다. 또한, BCNN은 작은 규모의 데이터 셋인 CIFAR-10, SVHN 이미지 분류에 대해 CNN과 비슷한 정확도 성능을 보이며, 이러한 이유로 BCNN은 GPU나 전용 가속기가 없는 연산 제한장치에서 인공지능을 실현하기에 적합한 모델로 여겨진다.

[0004] 이러한 이진화 컨볼루션 신경망과 관련하여, 빠른 추론을 위한 기술들 또한 함께 연구되고 있는데, 추론 과정 중 정확도에 영향이 없는 불필요한 연산을 생략하거나 정확도 손실에 큰 영향을 미치지 않는 조건에서 연산을 생략하고 조기에 결과를 내는 기술들이 발표되고 있는데, 예를 들어 종래의 연구에서는 이진화 컨볼루션 신경망의 추론을 더 빠르게 하기 위해 이진화 컨볼루션 신경망의 추론 계층에서의 구조를 이진화 컨볼루션, 배치정규화, 이진화, 맥스풀링 순으로 변경하거나 배치정규화에서의 복잡한 플로팅 포인트 연산을 임계값과의 비교를 통한 이진화로 대체하는 방식을 적용하였다. 다른 예로, 맥스풀링 계층에서 맥스풀링 윈도우 안에 1을 가지는 뉴런이 하나라도 있으면 풀링 결과가 1이므로, 1을 가지는 뉴런을 발견하는 즉시 맥스풀링 윈도우에 남은 뉴런에 대한 연산을 생략하고 풀링 결과 1을 도출하는 방식으로 추론 속도를 높일 수도 있다.

[0005] 한편, 일반적으로 이미지 분류를 위한 이진화 컨볼루션 신경망은 중간 피쳐들에서 공간적 인접성이 존재한다. 즉, 임의의 구간의 뉴런에서 1 또는 0이 발생하면 1일 때 0보다 1이, 0일 때 1보다 0이 인접한 뉴런에서 재차 발생할 확률이 높을 수 있으며, 이러한 공간적 인접성을 이용하여 연산을 일부 생략하면, 기존의 신경망 대비 더 빠른 속도로 추론이 가능해질 수 있다.

[0006] 본원의 배경이 되는 기술은 한국등록특허공보 제10-2143928호에 개시되어 있다.

발명의 내용

해결하려는 과제

- [0007] 본원은 전술한 종래 기술의 문제점을 해결하기 위한 것으로서, 이진화 컨볼루션 신경망의 추론 과정에서 중간 피쳐들에서 존재하는 인접성을 이용하여 전체 연산과정 중 일부를 생략하고 풀링 결과를 조기에 추론할 수 있는 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치 및 방법을 제공하려는 것을 목적으로 한다.
- [0008] 다만, 본원의 실시예가 이루고자 하는 기술적 과제는 상기된 바와 같은 기술적 과제들로 한정되지 않으며, 또 다른 기술적 과제들이 존재할 수 있다.

과제의 해결 수단

- [0009] 상기한 기술적 과제를 달성하기 위한 기술적 수단으로서, 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법은, (a) 상기 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득하는 단계, (b) 상기 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정하는 단계 및 (c) 상기 결정된 연산 순서에 기초하여 상기 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 상기 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 상기 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 상기 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정하는 단계를 포함할 수 있다.
- [0010] 또한, 상기 (c) 단계는, 상기 참조 풀링 결과가 0이고, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 0으로 결정할 수 있다.
- [0011] 또한, 상기 (c) 단계는, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 1로 결정할 수 있다.
- [0012] 또한, 상기 (b) 단계는, (b1) 상기 참조 풀링 결과가 0이면, 상기 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를 유지하는 단계 및 (b2) 상기 참조 풀링 결과가 1이면, 상기 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 상기 이전 풀링 윈도우에서의 위치인 기준 위치에 기초하여 상기 연산 순서를 갱신하는 단계를 포함할 수 있다.
- [0013] 또한, 상기 (b2) 단계는, 상기 이전 풀링 윈도우와 상기 현재 풀링 윈도우의 윈도우 진행 방향 및 상기 기준 위치에 기초하여 상기 연산 순서의 시작점을 결정할 수 있다.
- [0014] 또한, 상기 시작점은 상기 결정된 연산 순서에 의할 때 상기 윈도우 진행 방향을 고려하여 상기 기준 위치로부터 근접한 뉴런부터 연산이 이루어지도록 결정될 수 있다.
- [0015] 또한, 상기 (b2) 단계는, 상기 윈도우 진행 방향에 기초하여 상기 연산 순서의 패턴을 결정할 수 있다.
- [0016] 또한, 상기 임계값은 풀링 윈도우에 포함되는 복수의 뉴런의 수에 기초하여 결정될 수 있다.
- [0017] 한편, 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 연산 속도 향상 장치는, 상기 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득하는 기준정보 획득부, 상기 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정하는 정렬부 및 상기 결정된 연산 순서에 기초하여 상기 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 상기 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 상기 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 상기 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정하는 연산부를 포함할 수 있다.
- [0018] 또한, 상기 연산부는, 상기 참조 풀링 결과가 0이고, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 0으로 결정할 수 있다.
- [0019] 또한, 상기 연산부는, 상기 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 상기 연산 생략 조건이 충족된 것으로 판단하여 상기 대상 풀링 결과를 1로 결정할 수 있다.
- [0020] 또한, 상기 정렬부는, 상기 참조 풀링 결과가 0이면, 상기 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를

유지하고, 상기 참조 폴링 결과가 1이면, 상기 이전 폴링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 상기 이전 폴링 윈도우에서의 위치인 기준 위치에 기초하여 상기 연산 순서를 갱신할 수 있다.

[0021] 또한, 상기 정렬부는, 상기 이전 폴링 윈도우와 상기 현재 폴링 윈도우의 윈도우 진행 방향 및 상기 기준 위치에 기초하여 상기 연산 순서의 시작점을 결정할 수 있다.

[0022] 또한, 상기 정렬부는, 상기 윈도우 진행 방향에 기초하여 상기 연산 순서의 패턴을 결정할 수 있다.

[0023] 상술한 과제 해결 수단은 단지 예시적인 것으로서, 본원을 제한하려는 의도로 해석되지 않아야 한다. 상술한 예시적인 실시예 외에도, 도면 및 발명의 상세한 설명에 추가적인 실시예가 존재할 수 있다.

발명의 효과

[0024] 전술한 본원의 과제 해결 수단에 의하면, 이진화 컨볼루션 신경망의 추론 과정에서 중간 피쳐들에서 존재하는 인접성을 이용하여 전체 연산과정 중 일부를 생략하고 폴링 결과를 조기에 추론할 수 있는 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치 및 방법을 제공할 수 있다.

[0025] 전술한 본원의 과제 해결 수단에 의하면, 이진화 컨볼루션 신경망의 추론 과정에서 중간 피쳐들에서 존재하는 인접성을 이용하여 연산 결과가 이미 나온 뉴런의 결과를 바탕으로 인접한 뉴런의 결과를 더 빠르게 추론할 수 있다.

[0026] 전술한 본원의 과제 해결 수단에 의하면, 종래 기술이 폴링 결과가 1인 경우에 한하여 연산을 생략 가능하였던 한계를 해결하여 반복적으로 연산 결과가 0인 뉴런이 탐색되는 경우에도 후속 연산을 생략할 수 있다.

[0027] 전술한 본원의 과제 해결 수단에 의하면, 인접한 폴링 윈도우에서 1이 탐색된 뉴런의 위치를 고려하여 연산 순서를 변경함으로써 폴링 윈도우 내에서 1인 뉴런이 상대적으로 조기에 발견되도록 하여 실질적으로 생략 가능한 연산량을 증가시킬 수 있다.

[0028] 다만, 본원에서 얻을 수 있는 효과는 상기된 바와 같은 효과들로 한정되지 않으며, 또 다른 효과들이 존재할 수 있다.

도면의 간단한 설명

[0029] 도 1은 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치의 개략적인 구성도이다.

도 2는 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치의 동작을 설명하기 위한 개념도이다.

도 3은 윈도우 진행 방향(Operation Orientation)을 설명하기 위한 개념도이다.

도 4는 참조 폴링 결과 및 윈도우 진행 방향에 기초하여 현재 폴링 윈도우에 대한 연산 순서의 시작점 및 패턴을 결정하는 여러 방식을 예시적으로 나타낸 도면이다.

도 5는 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법에 대한 동작 흐름도이다.




도 6은 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법의 세부 동작 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0030] 아래에서는 첨부한 도면을 참조하여 본원이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본원의 실시예를 상세히 설명한다. 그러나 본원은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본원을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0031] 본원 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결" 또는 "간접적으로 연결"되어 있는 경우도 포함한다.

- [0032] 본원 명세서 전체에서, 어떤 부재가 다른 부재 "상에", "상부에", "상단에", "하에", "하부에", "하단에" 위치하고 있다고 할 때, 이는 어떤 부재가 다른 부재에 접해 있는 경우뿐 아니라 두 부재 사이에 또 다른 부재가 존재하는 경우도 포함한다.
- [0033] 본원 명세서 전체에서, 어떤 부분이 어떤 구성 요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있는 것을 의미한다.
- [0034] 본원은 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치 및 방법에 관한 것이다.
- [0035] 도 1은 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치의 개략적인 구성도이다.
- [0036] 도 1을 참조하면, 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치(100)(이하, '추론속도 향상 장치(100)'라 한다.)는 기준정보 획득부(110), 정렬부(120) 및 연산부(130)를 포함할 수 있다.
- [0037] 본원의 일 실시예에 따르면, 추론속도 향상 장치(100)는 기학습된 이진화 신경망(BCNN)을 이용한 추론 프로세스(예를 들면, 입력 이미지에 대한 이미지 분류를 수행하는 추론 동작 등)를 수행하는 사용자 단말(미도시)과 연계되어 구동하거나 사용자 단말(미도시)에 대하여 탑재되는 형태로 마련될 수 있다.
- [0038] 사용자 단말(미도시)은 예를 들면, 스마트폰(Smartphone), 스마트패드(SmartPad), 태블릿 PC등과 PCS(Personal Communication System), GSM(Global System for Mobile communication), PDC(Personal Digital Cellular), PHS(Personal Handyphone System), PDA(Personal Digital Assistant), IMT(International Mobile Telecommunication)-2000, CDMA(Code Division Multiple Access)-2000, W-CDMA(W-Code Division Multiple Access), Wibro(Wireless Broadband Internet) 단말기 같은 모든 종류의 무선 통신 장치일 수 있다. 또한, 본원의 일 실시예에 따르면 사용자 단말(미도시)은 임베디드 디바이스일 수 있다.
- [0039] 추론속도 향상 장치(100) 및 사용자 단말(미도시) 상호간은 네트워크를 통해 통신할 수 있다. 네트워크는 단말들 및 서버들과 같은 각각의 노드 상호간에 정보 교환이 가능한 연결 구조를 의미하는 것으로, 이러한 네트워크의 일 예에는, 3GPP(3rd Generation Partnership Project) 네트워크, LTE(Long Term Evolution) 네트워크, 5G 네트워크, WIMAX(World Interoperability for Microwave Access) 네트워크, 인터넷(Internet), LAN(Local Area Network), Wireless LAN(Wireless Local Area Network), WAN(Wide Area Network), PAN(Personal Area Network), wifi 네트워크, 블루투스(Bluetooth) 네트워크, 위성 방송 네트워크, 아날로그 방송 네트워크, DMB(Digital Multimedia Broadcasting) 네트워크 등이 포함되나 이에 한정되지는 않는다.
- [0040] 도 2는 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 장치의 동작을 설명하기 위한 개념도이다.
- [0041] 도 2를 참조하면, 추론속도 향상 장치(100)는 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 기 설정된 윈도우 사이즈(예시적으로, 도 2를 참조하면, 2x2 크기 등)를 가지는 풀링 윈도우에 대하여 반복적으로 수행되는 최대값 풀링(Max Pooling)을 이전 풀링 윈도우의 출력값(Pooling Output)에 따라 이전 풀링 윈도우의 다음 차례로 연산되는 현재 풀링 윈도우의 출력값(Pooling Output)을 도출하기 위한 연산을 기 설정된 연산 생략 조건에 따라 선택적으로 생략(스킵)하는 방식으로 추론속도를 향상시킬 수 있다.
- [0042] 구체적으로, 도 2를 참조하면, 추론속도 향상 장치(100)는 이전 풀링 윈도우의 출력값인 참조 풀링 결과가 0이고, 현재 풀링 윈도우에 포함된 복수의 뉴런에 대한 연산 순서에 따라 연산 결과가 0인 뉴런이 반복적으로 탐색되면, 해당 풀링 윈도우(현재 풀링 윈도우)에 포함된 나머지 뉴런에 대한 연산은 생략하고, 현재 풀링 윈도우의 출력값(Pooling Output)인 대상 풀링 결과를 0으로 결정한 후 다음 풀링 윈도우로 진행하도록 동작하여 이진화 신경망의 추론 속도를 높일 수 있다.
- [0043] 또한, 이하에서 상세히 후술하는 바와 같이 추론속도 향상 장치(100)는 이전 풀링 윈도우의 출력값인 참조 풀링 결과가 1인 경우, 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 위치에 따라 현재 풀링 윈도우에서의 뉴런 간 연산 순서를 재배열할 수 있다.
- [0044] 이하에서는 추론속도 향상 장치(100)를 이루는 각 구성의 구체적인 기능 및 동작에 대하여 설명하도록 한다.
- [0045] 기준정보 획득부(110)는 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득할 수 있다.

- [0046] 예시적으로 전술한 도 2에 도시된 바와 같이 → 방향으로 맥스풀링 연산이 진행되며 2x2 크기의 풀링 윈도우에 대한 최대값 풀링을 통해 각 풀링 윈도우에 대한 출력값(Pooling Output)을 도출하는 연산이 진행되는 경우, 가장 좌측에 위치한 풀링 윈도우(가장 먼저 최대값 풀링 연산이 이루어진 풀링 윈도우)를 제1풀링 윈도우라고 하고, 순차적으로 우측으로 진행하며 이어지는 각각의 풀링 윈도우를 제2풀링 윈도우 내지 제5풀링 윈도우라 지칭하면, 제2풀링 윈도우의 연산 시에는 제1풀링 윈도우가 이전 풀링 윈도우이고, 제2풀링 윈도우가 현재 풀링 윈도우인 것으로 이해될 수 있다. 마찬가지로, 제3풀링 윈도우의 연산 시에는 제2풀링 윈도우가 이전 풀링 윈도우이고, 제4풀링 윈도우의 연산 시에는 제3풀링 윈도우가 이전 풀링 윈도우이고, 제5풀링 윈도우의 연산 시에는 제4풀링 윈도우가 이전 풀링 윈도우일 수 있다.
- [0047] 이와 관련하여, 기준정보 획득부(110)는 예시적으로 도 2를 참조하면, 제2풀링 윈도우를 현재 풀링 윈도우로 하는 맥스풀링 연산시, 이전 풀링 윈도우(제1풀링 윈도우)의 출력값인 참조 풀링 결과로서 '0'을 획득하고, 제3풀링 윈도우를 현재 풀링 윈도우로 하는 맥스풀링 연산시, 이전 풀링 윈도우(제2풀링 윈도우)의 출력값인 참조 풀링 결과로서 '0'을 획득하고, 제4풀링 윈도우를 현재 풀링 윈도우로 하는 맥스풀링 연산시, 이전 풀링 윈도우(제3풀링 윈도우)의 출력값인 참조 풀링 결과로서 '0'을 획득하고, 제5풀링 윈도우를 현재 풀링 윈도우로 하는 맥스풀링 연산시, 이전 풀링 윈도우(제4풀링 윈도우)의 출력값인 참조 풀링 결과로서 '1'을 획득할 수 있다.
- [0048] 정렬부(120)는 기준정보 획득부(110)가 획득한 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정할 수 있다.
- [0049] 구체적으로, 정렬부(120)는 참조 풀링 결과가 0이면, 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를 유지할 수 있다. 반대로, 획득한 참조 풀링 결과가 1이면, 정렬부(120)는 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 이전 풀링 윈도우에서의 위치인 기준 위치에 기초하여 현재 풀링 윈도우에 대하여 적용될 연산 순서를 갱신할 수 있다.
- [0050] 본원의 일 실시예에 따르면, 정렬부(120)는 이전 풀링 윈도우의 출력값(Pooling Output)이 1인 경우, 이전 풀링 윈도우와 현재 풀링 윈도우의 윈도우 진행 방향 및 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 풀링 윈도우 내의 위치인 기준 위치에 기초하여 연산 순서의 시작점을 결정할 수 있다.
- [0051] 즉, 정렬부(120)는 이전 풀링 윈도우에서 1이 탐색된 뉴런의 위치를 기초로 하여 현재 풀링 윈도우에 적용될 연산 순서의 시작점 및 패턴을 결정함으로써, 현재 풀링 윈도우에서 연산 순서상 가장 마지막 뉴런에서 1이 탐색되어 실질적으로 연산을 생략할 수 있는 뉴런이 부존재하는 경우가 발생하는 빈도를 줄일 수 있다.
- [0052] 도 3은 윈도우 진행 방향(Operation Orientation)을 설명하기 위한 개념도이다.
- [0053] 도 3을 참조하면, 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에 인가된 입력 피처에 대한 풀링 윈도우 기반의 최대값 풀링 연산은 풀링 윈도우의 진행 순서에 따라 '→'방향, '↓'방향 또는 '←'방향의 세 방향으로 구분될 수 있다. 달리 말해, 윈도우 진행 방향(Operation Orientation, A)은 세 가지 타입(A_1 , A_2 , A_3)으로 구분될 수 있다.
- [0054] 이와 관련하여, 정렬부(120)는 풀링 결과가 연산되어 나오는 방향이 풀링 윈도우 간의 경계에 다다랐을 때 가장 가까운 곳에 위치한 맥스풀링 윈도우의 연산을 연속하여 수행할 수 있도록 '→'방향, '↓'방향 또는 '←'방향의 세 방향으로 구분되는 윈도우 진행 방향을 적용할 수 있다.
- [0055] 도 4는 참조 풀링 결과 및 윈도우 진행 방향에 기초하여 현재 풀링 윈도우에 대한 연산 순서의 시작점 및 패턴을 결정하는 여러 방식을 예시적으로 나타낸 도면이다.
- [0056] 도 4를 참조하면, 2x2 크기의 풀링 윈도우를 기준으로 하여, 풀링 윈도우에 포함된 4개의 뉴런에 대한 연산 순서는 'Z'자 형상을 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 위치인 기준 위치 및 윈도우 진행 방향의 유형에 따라 회전 또는 플리핑한 형상에 부합하는 순서로 결정될 수 있다.
- [0057] 즉, 본원의 일 실시예에 따르면, 정렬부(120)는 이전 풀링 윈도우와 현재 풀링 윈도우의 최대값 풀링 연산 방향에 대응하는 윈도우 진행 방향에 기초하여 현재 풀링 윈도우에 대하여 적용될 연산 순서의 패턴을 결정할 수 있다.
- [0058] 보다 구체적으로, 윈도우 진행 방향이 '→'방향인 제1방향(A_1)인 경우, 현재 풀링 윈도우에 대하여 적용 가능한 연산 순서는  또는  패턴이되, 기준 위치가 상측 행(1행)인 경우, 정렬부(120)는  패턴으로 연산 순

서를 결정하여 이전 폴링 윈도우와 현재 폴링 윈도우에서의 공간적 인접성에 따라 연산 결과가 1인 뉴런이 현재 폴링 윈도우에서 탐색될 가능성이 높은 패턴으로 현재 폴링 윈도우에 적용되는 복수의 뉴런에 대한 연산 순서를 결정할 수 있다. 이와 달리, 기준 위치가 하측 행(2행)인 경우, 정렬부(120)는 \nearrow 패턴으로 연산 순서를 결정할 수 있다.

[0059] 또한, 윈도우 진행 방향이 ' \downarrow ' 방향인 제2방향(A_2)인 경우, 현재 폴링 윈도우에 대하여 적용 가능한 연산 순서는 \searrow 또는 \swarrow 이 되, 기준 위치가 왼쪽 열(1열)인 경우, 정렬부(120)는 \searrow 패턴으로 연산 순서를 결정하여 이전 폴링 윈도우와 현재 폴링 윈도우에서의 공간적 인접성에 따라 연산 결과가 1인 뉴런이 현재 폴링 윈도우에서 탐색될 가능성이 높은 패턴으로 현재 폴링 윈도우에 적용되는 복수의 뉴런에 대한 연산 순서를 결정할 수 있다. 이와 달리, 기준 위치가 오른쪽 열(2열)인 경우, 정렬부(120)는 \swarrow 패턴으로 연산 순서를 결정할 수 있다.

[0060] 또한, 윈도우 진행 방향이 ' \leftarrow ' 방향인 제3방향(A_3)인 경우, 현재 폴링 윈도우에 대하여 적용 가능한 연산 순서는 \nwarrow 또는 \nearrow 이 되, 기준 위치가 상측 행(1행)인 경우, 정렬부(120)는 \nwarrow 패턴으로 연산 순서를 결정하여 이전 폴링 윈도우와 현재 폴링 윈도우에서의 공간적 인접성에 따라 연산 결과가 1인 뉴런이 현재 폴링 윈도우에서 탐색될 가능성이 높은 패턴으로 현재 폴링 윈도우에 적용되는 복수의 뉴런에 대한 연산 순서를 결정할 수 있다. 이와 달리, 기준 위치가 하측 행(2행)인 경우, 정렬부(120)는 \nearrow 패턴으로 연산 순서를 결정할 수 있다.

[0061] 종합하면, 정렬부(120)는 결정된 연산 순서에 의할 때, 윈도우 진행 방향을 고려하여 기준 위치로부터 근접한 뉴런에서부터 현재 폴링 윈도우에서의 연산이 이루어지도록 연산 순서의 시작점 및 패턴을 결정할 수 있다. 한편, 전술한 설명에서는 폴링 윈도우의 크기가 2x2인 것을 가정하고 설명하여 연산 순서의 패턴이 'Z'자 형상을 회전 또는 플리핑한 형상으로 결정되는 것으로 예시하였으나, 폴링 윈도우의 크기가 3x3, 4x4 등으로 보다 확대되는 경우에도 정렬부(120)가 이전 폴링 윈도우에서 1이 탐색된 뉴런의 위치인 기준 위치와 근접한 위치의 뉴런에서 최대값 폴링 연산이 개시되도록 현재 폴링 윈도우의 연산 순서를 결정하며, 이전 폴링 윈도우와 현재 폴링 윈도우의 상대적 위치에 따른 윈도우 진행 방향에 부합하도록 연산 순서의 시작점 및 패턴을 결정할 수 있음은 물론이다.

[0062] 연산부(130)는 정렬부(120)에 의해 결정된 연산 순서에 기초하여 현재 폴링 윈도우에 대한 최대값 폴링을 수행하되, 현재 폴링 윈도우 내의 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 현재 폴링 윈도우에 대한 출력값인 대상 폴링 결과를 조기에 결정할 수 있다. 또한, 연산부(130)는 대상 폴링 결과가 결정되고 나면, 현재 폴링 윈도우에 대한 최대값 폴링 연산을 종료하고, 추론속도 향상 장치(100)는 윈도우 진행 방향에 따라 현재 폴링 윈도우에 후속하는 폴링 윈도우를 재차 현재 폴링 윈도우로 설정하여 전술한 과정을 반복할 수 있다.

[0063] 전술한 연산 생략 조건과 관련하여, 연산부(130)는 참조 폴링 결과가 0이고, 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 제1연산 생략 조건이 충족된 것으로 판단하여 대상 폴링 결과를 0으로 결정할 수 있다.

[0064] 이와 관련하여, 전술한 도 2를 참조하면, 폴링 윈도우의 크기가 2x2(즉, s=2)이고, 연산 생략을 위한 임계값이 3으로 설정(n=3)되고, 윈도우 진행 방향이 제1방향(A_1)이고, 현재 폴링 윈도우에 대하여 결정된 연산 순서가 \searrow 패턴인 경우, 연산부(130)는 현재 폴링 윈도우에 존재하는 4개의 뉴런 중 연산 순서에 따라 먼저 수행하는 세 개의 뉴런들의 값이 0일 때, 이전 폴링 윈도우의 폴링 결과인 참조 폴링 결과에 따라, 참조 폴링 결과가 0인 경우 마지막 뉴런에 대한 연산을 수행하지 않고 제1연산 생략 조건을 충족한 것으로 보아 대상 폴링 결과를 0으로 추론한 후 현재 폴링 윈도우에 대한 연산을 종료할 수 있다.

[0065] 도 2에 도시된 좌측 두 번째 폴링 윈도우와 좌측 세 번째 폴링 윈도우에서는 전술한 제1연산 생략 조건이 충족되어 일부 뉴런에 대한 연산이 생략(스킵)된 것을 확인할 수 있다. 즉, 좌측 두 번째 폴링 윈도우와 좌측 세 번째 폴링 윈도우에서는 연산 순서상 초기 3개의 뉴런 모두 0이고, 이전 폴링 윈도우에서의 참조 폴링 결과가 0이므로, 좌측 두 번째 폴링 윈도우와 좌측 세 번째 폴링 윈도우에서는 제1연산 생략 조건이 충족된 것으로 판단되어 마지막 뉴런에 대한 연산을 생략하고 대상 폴링 결과가 0으로 추론될 수 있다.

[0066] 한편, 도 2의 좌측 첫 번째 폴링 윈도우에서도 연산 순서상 초기 3개의 뉴런 모두 연산 결과가 0이지만 이전 폴링 윈도우에 대한 참조 폴링 결과에 대한 정보가 부존재하여 제1연산 생략 조건을 만족할 수 없으므로 모든 뉴

런에 대해 연산을 진행한 후 모든 뉴런의 연산 결과가 0에 해당함에 따라 대상 풀링 결과가 0으로 추론되게 된다.

[0067] 또한, 도 2의 좌측 네 번째 풀링 윈도우에서는 두 번째로 연산을 수행하는 뉴런이 1을 가지기 때문에 제1연산 생략 조건은 충족할 수 없으나, 대신에 해당 풀링 윈도우에서는 연산 결과가 1인 뉴런이 탐색되었으므로, 대상 풀링 결과를 1로 결정하고 나머지 뉴런에 대한 연산을 생략하는 제2연산 생략 조건이 충족될 수 있다. 즉, 연산 부(130)는 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 제2연산 생략 조건이 충족된 것으로 판단하여 현재 풀링 윈도우에 대한 대상 풀링 결과를 1로 결정하고, 나머지 뉴런에 대한 연산을 생략할 수 있다.

[0068] 또한, 도 2의 좌측 다섯 번째 풀링 윈도우에서는 참조 풀링 결과가 1이므로, 전술한 제1연산 생략 조건은 적용될 수 없고, 해당 풀링 윈도우에서의 연산 순서가 좌측 네 번째 풀링 윈도우에서 1인 뉴런이 탐색된 기준 위치에 따라 재배열될 수 있다.

[0069] 한편, 본원의 일 실시예에 따르면, 제1연산 생략 조건을 적용하기 위한 임계값은 풀링 윈도우에 포함되는 복수의 뉴런의 수에 기초하여 결정될 수 있다. 이와 관련하여, 임계값에 따른 이진화 신경망 모델의 정확도와 추론 수행 시간은 서로 반비례하므로, 정확도보다 추론 수행 시간의 단축에 초점을 둔다면 임계값을 상대적으로 작게 설정하여 생략되는 연산량을 늘릴 수 있고, 반대로 추론 수행 시간의 단축 보다는 정확도 수준을 높이거나 일정 수준 이상을 유지하는 것을 목표로 하는 경우에는 임계값을 상대적으로 크게 설정할 수 있다.

[0070] 또한, 본원의 일 실시예에 따르면, 복수 개의 최대값 풀링 계층이 존재하는 이진화 신경망 모델의 경우에는 전체 최대값 풀링 계층에 대하여 공통된 임계값을 적용하는 것이 아니라 계층별로 개별적인 임계값을 설정함으로써 전체 모델의 정확도 손실을 방지할 수 있다.

[0071] 한편, 본원에서 개시하는 추론속도 향상 기법은 이미지 분류를 위한 이진화 컨볼루션 신경망의 중간 피처에 일반적으로 존재하는 공간적 인접성을 이용한 기술이므로, 대부분 이진화 신경망 모델에서 성능 향상을 기대할 수 있다.

[0072] 특히, 제1연산 생략 조건에 따른 연산 생략 기법은 0인 뉴런이 반복되면 반복된 횟수와 인접한 뉴런의 값에 따라 다음 뉴런의 값을 연산 수행 없이 예측하는 기술이며 이 기술을 맥스풀링 계층에 사용하면, 1인 뉴런이 탐색되면 풀링 결과를 1로 결정하고 후속 연산을 생략하는 종래의 기법과 함께 적용되어 더 많은 뉴런의 출력을 조기에 결정하고 연산을 생략할 수 있다.

[0073] 즉, 이전에 연구된 기술은 맥스풀링 계층에서 풀링 결과가 1일 때에 한해 불필요한 연산을 생략 가능했으며, 맥스풀링 윈도우 상에서 뉴런의 값을 살펴볼 때, 1이라는 결과가 앞부분에서 발견되는 것이 아닌 뒷부분에서 발견되는 경우에 생략 가능한 연산이 없어지는 문제점이 있었으나, 본원에서 개시하는 제1연산 생략 조건에 따른 연산 생략 기법은 풀링 결과가 0일 때에 적용 가능한 기술로서 기존 기술의 한계를 해결하였다.

[0074] 또한, 이전 풀링 윈도우에서 1이 탐색된 뉴런의 위치인 기준 위치를 고려한 현재 풀링 윈도우에 대한 연산 순서 재배열 기법은 이전 풀링 윈도우에서의 풀링 결과가 1일 때 다음으로 연산되는 현재 풀링 윈도우에서 탐색하는 뉴런의 순서를 변경하여 상대적으로 조기에 1인 뉴런이 발견되도록 유도할 수 있다.

[0075] 또한, 본원에서 개시하는 제1연산 생략 조건에 따른 연산 생략 기법과 기준 위치를 고려한 연산 순서 재배열 기법은 맥스풀링 윈도우의 특정 크기에 의존하지 않고 모든 크기에 적용할 수 있다. 제1연산 생략 조건에 따른 연산 생략 기법의 경우에 맥스풀링 윈도우 크기가 큰 경우에 생략하는 뉴런 수가 크더라도 정확도 손실이 윈도우 크기가 작을 때에 비하여 작으므로 윈도우 크기가 큰 맥스풀링 계층을 가지는 모델의 추론을 빠르게 하는데 큰 효과를 볼 수 있다.

[0076] 또한, 윈도우가 큰 맥스풀링 계층에서 기준 위치를 고려한 연산 순서 재배열 기법을 적용하면 윈도우 크기가 작을 때에 비해 큰 효과를 볼 수 있다. 윈도우 크기가 큰 경우에 연산하는 뉴런의 수가 많으므로 1을 가진 뉴런이 뒷부분에 나온다면 윈도우가 작았을 때 비해 수행하는 연산이 커진다. 기준 위치를 고려한 연산 순서 재배열 기법을 적용하여 1을 가진 뉴런이 앞부분에 나오도록 유도하고 앞부분에서 1을 실제로 발견한다면 생략할 수 있는 뉴런의 개수가 윈도우가 작을 때에 비해 크므로 더 빠른 추론을 할 수 있는 여지가 커진다.

[0077] 한편, 기준 위치를 고려한 연산 순서 재배열 기법을 적용하여 조기에 뉴런의 출력을 결정하였을 때에는 오판의 가능성이 존재하지 않기 때문에 제1연산 생략 조건에 따른 연산 생략 기법과 다르게 정확도 손실이 발생하지 않는다. 제1연산 생략 조건에 따른 연산 생략 기법에서의 임계값이 맥스풀링 윈도우 크기에 비하여 작을수록 생략

하는 뉴런의 개수가 증가하고 그에 따라 뉴런의 출력을 오판하는 가능성이 증가하여 정확도 손실이 증가한다. 하지만, 임계값이 맥스폴링 윈도우의 크기를 고려하여 적절히 설정해준다면 낮은 정확도 손실과 함께 기술의 성능을 높일 수 있다.

- [0078] 이하에서는 상기에 자세히 설명된 내용을 기반으로, 본원의 동작 흐름을 간단히 살펴보기로 한다.
- [0079] 도 5는 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법에 대한 동작 흐름도이다.
- [0080] 도 5에 도시된 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법은 앞서 설명된 추론속도 향상 장치(100)에 의하여 수행될 수 있다. 따라서, 이하 생략된 내용이라고 하더라도 추론속도 향상 장치(100)에 대하여 설명된 내용은 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법에 대한 설명에도 동일하게 적용될 수 있다.
- [0081] 도 5를 참조하면, 단계 S11에서 기준정보 획득부(110)는 (a) 이진화 컨볼루션 신경망의 최대값 풀링 연산 계층에서 각각의 풀링 윈도우에 대하여 반복하여 수행되는 최대값 풀링의 출력값 중 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득할 수 있다.
- [0082] 다음으로, 단계 S12에서 정렬부(120)는 (b) 단계 S11에서 획득한 참조 풀링 결과에 기초하여 현재 풀링 윈도우의 복수의 뉴런에 대한 연산 순서를 결정할 수 있다.
- [0083] 구체적으로, 단계 S12에서 정렬부(120)는 (b1) 참조 풀링 결과가 0이면, 이전 풀링 윈도우에 대하여 적용된 이전 연산 순서를 유지할 수 있다.
- [0084] 이와 달리, 단계 S12에서 정렬부(120)는 (b2) 참조 풀링 결과가 1이면, 이전 풀링 윈도우에서 연산 결과가 1인 것으로 탐색된 뉴런의 이전 풀링 윈도우에서의 위치인 기준 위치에 기초하여 현재 풀링 윈도우에 대하여 적용될 연산 순서를 갱신할 수 있다.
- [0085] 구체적으로, 단계 S12에서 정렬부(120)는 이전 풀링 윈도우와 현재 풀링 윈도우의 윈도우 진행 방향 및 이전 풀링 윈도우에서의 기준 위치에 기초하여 현재 풀링 윈도우에 적용될 연산 순서의 시작점을 결정할 수 있다. 여기서, 연산 순서의 시작점은 결정된 연산 순서에 의할 때 윈도우 진행 방향을 고려하여 기준 위치로부터 근접한 뉴런부터 현재 풀링 윈도우에서 연산이 이루어지도록 결정될 수 있다.
- [0086] 다음으로, 단계 S13에서 연산부(130)는 (c) 단계 S12를 통해 결정된 연산 순서에 기초하여 현재 풀링 윈도우에 대한 최대값 풀링을 수행하되, 현재 풀링 윈도우에 포함된 복수의 뉴런 각각의 연산 결과가 기 설정된 연산 생략 조건을 충족하면 복수의 뉴런 중 적어도 일부를 제외한 뉴런의 연산 결과를 기초로 하여 현재 풀링 윈도우에 대한 출력값인 대상 풀링 결과를 결정할 수 있다.
- [0087] 구체적으로, 단계 S13에서 연산부(130)는 참조 풀링 결과가 0이고, 결정된 연산 순서를 따라 복수의 뉴런 중 연산 결과가 0인 뉴런이 기 설정된 임계값 이상 반복하여 존재하면, 제1연산 생략 조건이 충족된 것으로 판단하여 대상 풀링 결과를 0으로 결정할 수 있다.
- [0088] 또한, 단계 S13에서 연산부(130)는 단계 S12에서 결정된 연산 순서를 따라 복수의 뉴런 중 연산 결과가 1인 뉴런이 탐색되면, 제2연산 생략 조건이 충족된 것으로 판단하여 대상 풀링 결과를 1로 결정할 수 있다.
- [0089] 상술한 설명에서, 단계 S11 내지 S13은 본원의 구현예에 따라서, 추가적인 단계들로 더 분할되거나, 더 적은 단계들로 조합될 수 있다. 또한, 일부 단계는 필요에 따라 생략될 수도 있고, 단계 간의 순서가 변경될 수도 있다.
- [0090] 도 6은 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법의 세부 동작 흐름도이다.
- [0091] 도 6에 도시된 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 방법의 세부 동작 프로세스는 앞서 설명된 추론속도 향상 장치(100)에 의하여 수행될 수 있다. 따라서, 이하 생략된 내용이라고 하더라도 추론속도 향상 장치(100)에 대하여 설명된 내용은 도 6에 대한 설명에도 동일하게 적용될 수 있다.
- [0092] 도 6을 참조하면, 단계 S21에서 추론속도 향상 장치(100)는 현재 풀링 윈도우에 대한 연산을 개시할 수 있다.
- [0093] 다음으로, 단계 S22에서 기준정보 획득부(110)는 이전 풀링 윈도우의 출력값인 참조 풀링 결과를 획득할 수 있다.

- [0094] 단계 S22에서 획득된 참조 폴링 결과가 1이면, 단계 S23에서 정렬부(120)는 현재 폴링 윈도우에 대하여 적용될 연산 순서를 재배열할 수 있다. 반대로, 단계 S22에서 획득된 참조 폴링 결과가 0이면, 정렬부(120)는 이전 폴링 윈도우에서 적용된 이전 연산 순서를 유지할 수 있다.
- [0095] 또한, 단계 S24에서 연산부(130)는 현재 폴링 윈도우에 포함된 복수의 뉴런 각각의 연산 결과를 도출할 수 있다. 이 때, 단계 S24에서 연산부(130)가 복수의 뉴런 중 연산 결과가 1인 뉴런을 탐색하면 현재 폴링 윈도우에 대한 대상 폴링 결과를 1로 결정하고, 제2연산 생략 조건이 충족된 것으로 보아 나머지 뉴런에 대한 연산(탐색)을 종료할 수 있다.
- [0096] 반대로, 단계 S24에서 연산부(130)가 해당 뉴런의 연산 결과가 0인 것으로 도출되면, 남은 뉴런이 존재하는지 확인(S26) 후 만일 남은 뉴런이 존재하지 않으면(NO), 모든 뉴런의 출력이 0인 것이므로 폴링 결과를 0으로 결정할 수 있다(S29).
- [0097] 반대로, 단계 S26에서의 확인 결과 남은 뉴런이 존재하면, 연산부(130)는 연산 결과가 0인 뉴런이 반복된 횟수와 기 설정된 임계값을 비교할 수 있다(S27). 이때, 연산 결과가 0인 뉴런이 반복된 횟수가 기 설정된 임계값보다 작으면(YES), 연산부(130)는 연산 순서상 다음 뉴런의 연산을 수행할 수 있다. 반대로, 단계 S27의 판단 결과, 연산 결과가 0인 뉴런이 반복된 횟수가 기 설정된 임계값보다 크거나 같으면 이전 폴링 윈도우에서의 출력값인 참조 폴링 결과(S28)에 따라 참조 폴링 결과가 0이면 제1연산 생략 조건을 충족한 것으로 보아 나머지 뉴런에 대한 연산을 생략하고 대상 폴링 결과를 0으로 결정할 수 있다(S29). 반대로, 연산 결과가 0인 뉴런이 반복된 횟수가 기 설정된 임계값보다 크거나 같으나 이전 폴링 윈도우에서의 출력값인 참조 폴링 결과(S28)에 따라 참조 폴링 결과가 1이면, 제1연산 생략 조건이 충족될 수 없으므로, 연산부(130)는 연산 순서상 다음 뉴런의 연산을 이어서 수행할 수 있다.
- [0098] 상술한 설명에서, 단계 S21 내지 S29는 본원의 구현예에 따라서, 추가적인 단계들로 더 분할되거나, 더 적은 단계들로 조합될 수 있다. 또한, 일부 단계는 필요에 따라 생략될 수도 있고, 단계 간의 순서가 변경될 수도 있다.
- [0099] 본원의 일 실시예에 따른 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 추론속도 향상 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 본 발명을 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. 상기된 하드웨어 장치는 본 발명의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.
- [0100] 또한, 전술한 ~방법은 기록 매체에 저장되는 컴퓨터에 의해 실행되는 컴퓨터 프로그램 또는 애플리케이션의 형태로도 구현될 수 있다.
- [0101] 전술한 본원의 설명은 예시를 위한 것이며, 본원이 속하는 기술분야의 통상의 지식을 가진 자는 본원의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.
- [0102] 본원의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본원의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

- [0103] 100: 이진화 컨볼루션 신경망의 공간적 인접성을 이용한 연산 속도 향상 장치

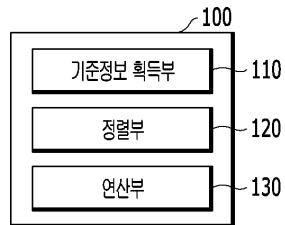
110: 기준정보 획득부

120: 정렬부

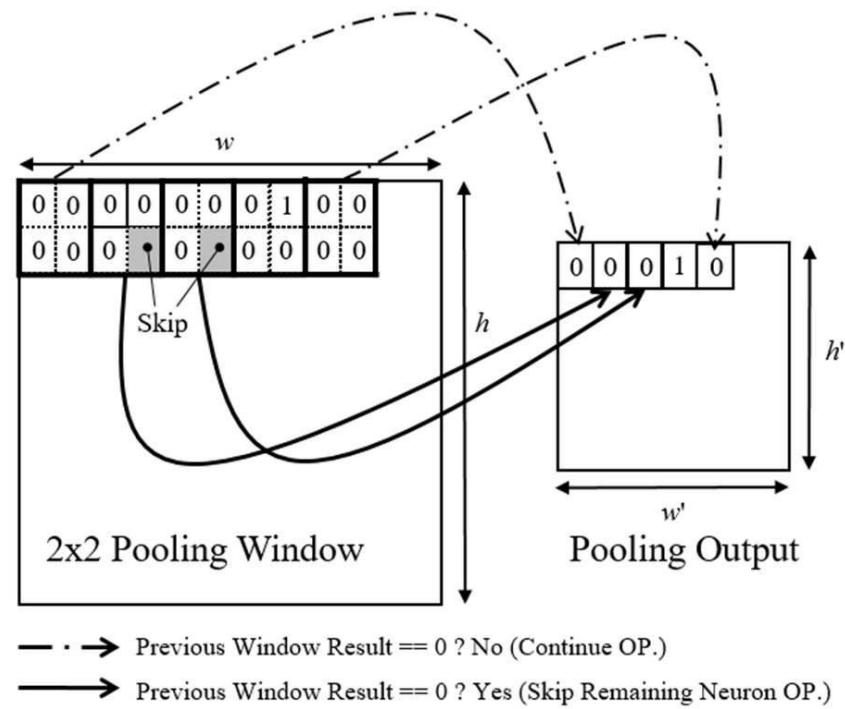
130: 연산부

도면

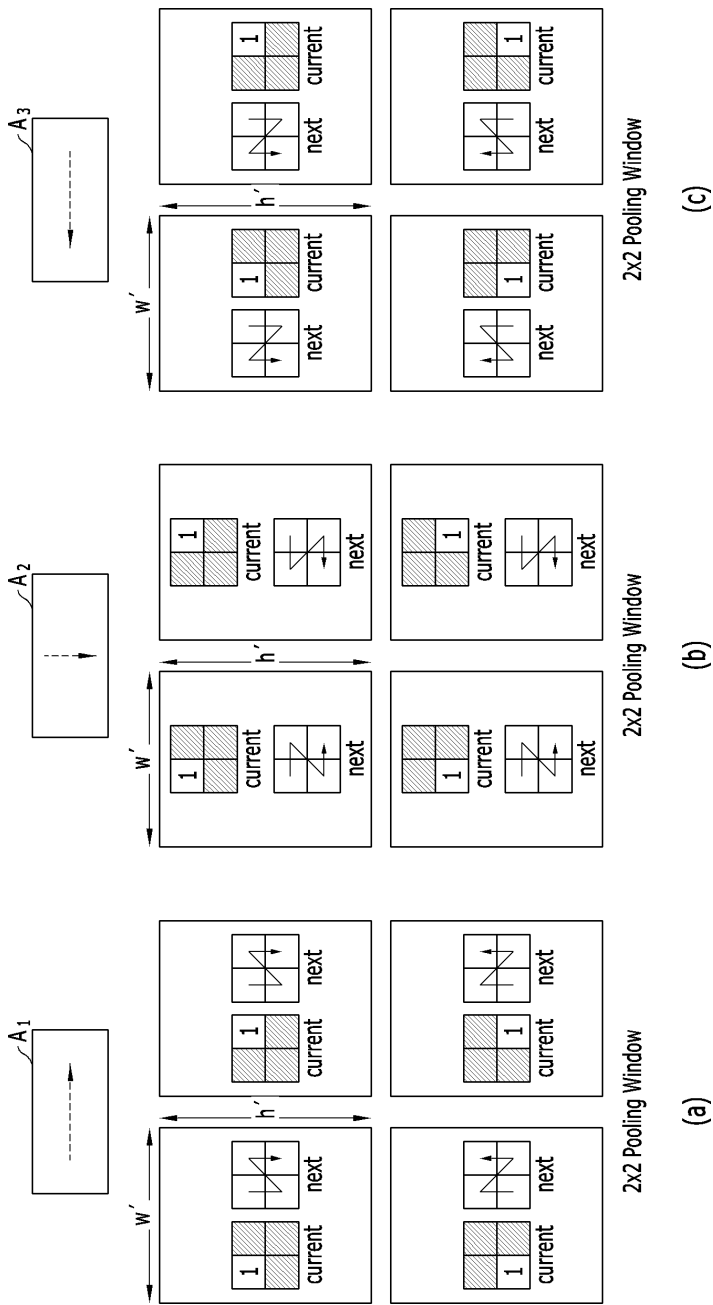
도면1



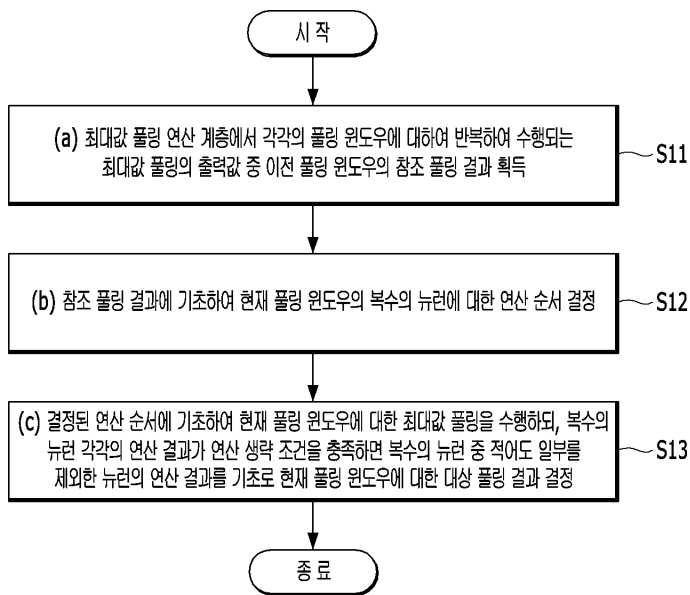
도면2



도면4



도면5



도면6

