



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2022-0055827
(43) 공개일자 2022년05월04일

(51) 국제특허분류(Int. Cl.)
G06N 3/08 (2006.01) G06N 3/04 (2006.01)
G06T 5/20 (2006.01)
(52) CPC특허분류
G06N 3/08 (2013.01)
G06N 3/04 (2013.01)
(21) 출원번호 10-2020-0140315
(22) 출원일자 2020년10월27일
심사청구일자 2020년10월27일

(71) 출원인
한림대학교 산학협력단
강원도 춘천시 한림대학길 1, 한림대학교(옥천동)
(72) 발명자
이정근
강원도 춘천시 서부대성로 34, 103동 1301호(요선동, 한신희플러스아파트)
정재민
서울특별시 강북구 도봉로68길 26, 103동 405호(미아동, 현대아파트)
(74) 대리인
김남혁

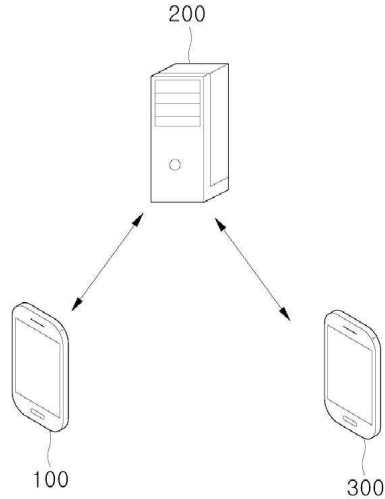
전체 청구항 수 : 총 9 항

(54) 발명의 명칭 인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법, 장치 및 프로그램

(57) 요약

인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법이 제공된다. 상기 제공 방법은, 서버가, 입력 필터, 상기 입력 필터의 선형 조합을 위한 조합 계수 및 인공지능 모델에 대응되는 네트워크 구조를 전자 장치로 전송하는 단계; 상기 전자 장치가, 상기 입력 필터, 상기 조합 계수 및 상기 네트워크 구조를 바탕으로 합성곱 필터를 획득하는 단계; 및 상기 전자 장치가, 상기 합성곱 필터를 바탕으로 상기 인공지능 모델의 특징맵을 획득하는 단계;를 포함한다.

대표도 - 도1



(52) CPC특허분류
G06T 5/20 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1345316268
과제번호	2018R1D1A1B07043399
부처명	과학기술정보통신부
과제관리(전문)기관명	한국연구재단
연구사업명	이공학개인지초연구지원사업(기초연구)
연구과제명	고신뢰 시스템을 위한 저전자파/결함-감내 디지털 회로 설계
기 여 율	1/1
과제수행기관명	한림대학교 산학협력단
연구기간	2020.03.01 ~ 2021.02.28

명세서

청구범위

청구항 1

인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법에 있어서,

서버가, 입력 필터, 상기 입력 필터의 선형 조합을 위한 조합 계수 및 인공지능 모델에 대응되는 네트워크 구조를 전자 장치로 전송하는 단계;

상기 전자 장치가, 상기 입력 필터, 상기 조합 계수 및 상기 네트워크 구조를 바탕으로 합성곱 필터를 획득하는 단계; 및

상기 전자 장치가, 상기 합성곱 필터를 바탕으로 상기 인공지능 모델의 특징맵을 획득하는 단계; 를 포함하는 제어 방법.

청구항 2

제1항에 있어서,

상기 합성곱 필터를 획득하는 단계는,

상기 전자 장치가, 하기 수학적 1을 바탕으로 상기 합성곱 필터를 획득하는 단계; 를 포함하고,

[수학적 1]

$$F_i^j = \left\{ \sum_{k=0}^n I_k \times a_{i,j}^{c,k} \right\} \text{ where } 0 < j \leq C_i, 0 < c \leq C_{i-1}$$

이때, F_i^j 는 i번째 계층의 j번째 필터, I_k 는 $k \times k \times k$ 크기를 가지는 k번째 입력 필터, $a_{i,j}^{c,k}$ 는 i번째 계층의 j번째 필터에 대한 c번째 채널과 연결되는 k번째 입력 필터의 계수, C_i 는 i번째 계층의 채널의 수, n은 입력 필터의 개수인 것을 특징으로 하는 제어 방법.

청구항 3

제2항에 있어서,

상기 특징맵을 획득하는 단계는,

상기 전자 장치가, 하기 수학적 2를 바탕으로 상기 특징맵을 획득하는 단계; 를 포함하고,

[수학적 2]

$$X_i = \{X_{i-1} * F_i^j\} \text{ where } 0 < j \leq C_i$$

이때, X_i 는 합성곱 연산을 통해서 출력으로 나오는 I번째 계층의 특징 맵인 것을 특징으로 하는 제어 방법.

청구항 4

제2항에 있어서,

상기 입력 필터의 개수 n은,

상기 인공지능 모델의 합성곱 계층의 커널이 3X3인 경우 상기 n은 3이고,

상기 인공지능 모델의 합성곱 계층의 커널이 5X5인 경우 상기 n은 8인 것을 특징으로 하는 제어 방법.

청구항 5

제1항에 있어서,

상기 제어 방법은,

상기 전자 장치의 메모리가 기 설정된 크기 이상인 경우, 상기 합성곱 필터를 바탕으로 획득된 상기 파라미터를 상기 전자 장치의 메모리에 저장하는 단계;를 더 포함하는 것을 특징으로 하는 제어 방법.

청구항 6

제2항에 있어서,

상기 합성곱 필터를 획득하는 단계는,

상기 전자 장치의 메모리가 기 설정된 크기 미만인 경우, 상기 전자 장치의 가용 메모리를 판단하는 단계; 및
상기 가용 메모리를 바탕으로 상기 입력 필터의 개수 n 을 결정하는 단계;를 포함하는 제어 방법.

청구항 7

하나 이상의 인스트럭션을 저장하는 메모리; 및

상기 메모리에 저장된 상기 하나 이상의 인스트럭션을 실행하는 프로세서를 포함하고,

상기 프로세서는 상기 하나 이상의 인스트럭션을 실행함으로써,

제1 항의 방법을 수행하는, 장치.

청구항 8

하드웨어인 컴퓨터와 결합되어, 제1 항의 방법을 수행할 수 있도록 컴퓨터에서 독출가능한 기록매체에 저장된 컴퓨터프로그램.

청구항 9

전자 장치의 제어 방법에 있어서,

상기 전자 장치가, 인공지능 모델의 특징맵 획득을 위한 필터 F 를 획득하는 단계;

상기 전자 장치가, 상기 필터 F 를 바탕으로 입력 필터 및 상기 입력 필터의 선형 조합을 위한 조합 계수를 획득하는 단계; 및

상기 전자 장치가, 상기 획득된 입력 필터 및 상기 획득된 조합 계수를 서버로 전송하는 단계;를 포함하는 제어 방법.

발명의 설명

기술 분야

[0001] 본 발명은 인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법, 장치 및 프로그램에 관한 것이다.

배경 기술

[0002] 최근 실생활에서 많은 문제를 해결하기 위한 딥러닝, 인공지능 기술의 적용 분야가 증가하고 있다. 특히, 딥러닝 기법 중 하나인 CNN(Convolutional Neural Network)은 이미지 분류, 객체 검출, 객체 분할 등 컴퓨터 비전 분야에서 다양하게 활용되고 있다. 그러나 이러한 딥러닝 모델은 수백 수천만 파라미터에 의존한다. 이때, 파라미터가 증가할수록 연산량이 늘어나고 요구되는 메모리의 크기가 증가하게 되므로 딥러닝 기술은 고성능의 GPU 및 메모리를 가지는 장치에서 동작할 수 있다. 즉, 메모리가 작고 컴퓨팅 파워가 낮은 엡지 디바이스의 경우, 딥러닝 모델을 실행하기 어렵거나, 정확도가 낮아지는 문제점이 존재한다.

[0003] 상술한 문제점을 해결하기 위하여, 프루닝, 양자화, 이진화, 지식 증류, 적은 파라미터를 가지는 모델 설계 등 다양한 인공지능 모델 압축방법이 제시되고 있으나, 최적의 압축 방법을 찾기 위한 연구들이 계속하여 진행되고 있는 추세이다.

선행기술문헌

특허문헌

[0004] (특허문헌 0001) 공개특허공보 제10-2020-0046188호, 2020.05.07

발명의 내용

해결하려는 과제

[0005] 본 발명이 해결하고자 하는 과제는 인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법, 장치 및 프로그램을 제공하는 것이다.

[0006] 본 발명이 해결하고자 하는 과제들은 이상에서 언급된 과제로 제한되지 않으며, 언급되지 않은 또 다른 과제들은 아래의 기재로부터 통상의 기술자에게 명확하게 이해될 수 있을 것이다.

과제의 해결 수단

[0007] 상술한 과제를 해결하기 위한 본 발명의 일 면에 따른 인공지능 모델의 파라미터 저장을 위한 필터 조합 학습 네트워크 시스템의 제어 방법은, 서버가, 입력 필터, 상기 입력 필터의 선형 조합을 위한 조합 계수 및 인공지능 모델에 대응되는 네트워크 구조를 전자 장치로 전송하는 단계; 상기 전자 장치가, 상기 입력 필터, 상기 조합 계수 및 상기 네트워크 구조를 바탕으로 합성곱 필터를 획득하는 단계; 및 상기 전자 장치가, 상기 합성곱 필터를 바탕으로 상기 인공지능 모델의 특징맵을 획득하는 단계; 를 포함한다.

[0008] 이때, 상기 합성곱 필터를 획득하는 단계는, 상기 전자 장치가, 하기 수학식 1을 바탕으로 상기 합성곱 필터를 획득하는 단계; 를 포함하고,

[0009] [수학식 1]

$$F_i^j = \left\{ \sum_{k=0}^n I_k \times a_{i,j}^{c,k} \right\} \text{ where } 0 < j \leq C_i, 0 < c \leq C_{i-1}$$

[0011] 이때, F_i^j 는 i번째 계층의 j번째 필터, I_k 는 $k \times k$ 크기를 가지는 k번째 입력 필터, $a_{i,j}^{c,k}$ 는 i번째 계층의 j번째 필터에 대한 c번째 채널과 연결되는 k번째 입력 필터의 계수, C_i 는 i번째 계층의 채널의 수, n은 입력 필터의 개수일 수 있다.

[0012] 이때, 상기 특징맵을 획득하는 단계는, 상기 전자 장치가, 하기 수학식 2를 바탕으로 상기 특징맵을 획득하는 단계; 를 포함하고,

[0013] [수학식2]

$$X_i = \{X_{i-1} * F_i^j\} \text{ where } 0 < j \leq C_i$$

[0015] 이때, X_i 는 합성곱 연산을 통해서 출력으로 나오는 I번째 계층의 특징 맵일 수 있다.

[0016] 이때, 상기 입력 필터의 개수 n은, 상기 인공지능 모델의 합성곱 계층의 커널이 3X3인 경우 상기 n은 3이고, 상기 인공지능 모델의 합성곱 계층의 커널이 5X5인 경우 상기 n은 8일 수 있다.

[0017] 이때, 상기 제어 방법은, 상기 전자 장치의 메모리가 기 설정된 크기 이상인 경우, 상기 합성곱 필터를 바탕으로 획득된 상기 파라미터를 상기 전자 장치의 메모리에 저장하는 단계;를 더 포함할 수 있다.

[0018] 이때, 상기 합성곱 필터를 획득하는 단계는, 상기 전자 장치의 메모리가 기 설정된 크기 미만인 경우, 상기 전자 장치의 가용 메모리를 판단하는 단계; 및 상기 가용 메모리를 바탕으로 상기 입력 필터의 개수 n 을 결정하는 단계;를 포함할 수 있다.

[0019] 한편, 본 발명의 또 다른 일면에 따른 전자 장치의 제어 방법은, 상기 전자 장치가, 인공지능 모델의 특징맵 획득을 위한 필터 F 를 획득하는 단계; 상기 전자 장치가, 상기 필터 F 를 바탕으로 입력 필터 및 상기 입력 필터의 선형 조합을 위한 조합 계수를 획득하는 단계; 및 상기 전자 장치가, 상기 획득된 입력 필터 및 상기 획득된 조합 계수를 서버로 전송하는 단계;를 포함한다.

[0020] 본 발명의 기타 구체적인 사항들은 상세한 설명 및 도면들에 포함되어 있다.

발명의 효과

[0021] 상술한 본 발명의 다양한 실시예에 따라, 메모리량이 적은 엣지 디바이스에 인공지능 모델을 탑재시킬 수 있는 효과가 발생한다.

[0022] 본 발명의 효과들은 이상에서 언급된 효과로 제한되지 않으며, 언급되지 않은 또 다른 효과들은 아래의 기재로부터 통상의 기술자에게 명확하게 이해될 수 있을 것이다.

도면의 간단한 설명

[0023] 도 1은 본 발명의 일 실시예에 따른 시스템의 구성도이다.

도 2는 본 발명의 일 실시예에 따른 필터 조합 학습 네트워크 시스템의 동작을 설명하기 위한 흐름도이다.

도 3 내지 도 6은 본 발명의 일 실시예에 따른 입력 필터 생성 방법을 설명하기 위한 예시도이다.

도 7 내지 도 11은 본 발명의 일 실시예에 따른 필터 조합 학습 네트워크 시스템에서의 파라미터 감소 효과를 설명하기 위한 예시도이다.

도 12는 본 발명의 일 실시예에 따른 장치의 구성도이다.

발명을 실시하기 위한 구체적인 내용

[0024] 본 발명의 이점 및 특징, 그리고 그것들을 달성하는 방법은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 그러나, 본 발명은 이하에서 개시되는 실시예들에 제한되는 것이 아니라 서로 다른 다양한 형태로 구현될 수 있으며, 단지 본 실시예들은 본 발명의 개시가 완전하도록 하고, 본 발명이 속하는 기술 분야의 통상의 기술자에게 본 발명의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 발명은 청구항의 범주에 의해 정의될 뿐이다.

[0025] 본 명세서에서 사용된 용어는 실시예들을 설명하기 위한 것이며 본 발명을 제한하고자 하는 것은 아니다. 본 명세서에서, 단수형은 문구에서 특별히 언급하지 않는 한 복수형도 포함한다. 명세서에서 사용되는 "포함한다(comprises)" 및/또는 "포함하는(comprising)"은 언급된 구성요소 외에 하나 이상의 다른 구성요소의 존재 또는 추가를 배제하지 않는다. 명세서 전체에 걸쳐 동일한 도면 부호는 동일한 구성 요소를 지칭하며, "및/또는"은 언급된 구성요소들의 각각 및 하나 이상의 모든 조합을 포함한다. 비록 "제1", "제2" 등이 다양한 구성요소들을 서술하기 위해서 사용되나, 이들 구성요소들은 이들 용어에 의해 제한되지 않음은 물론이다. 이들 용어들은 단지 하나의 구성요소를 다른 구성요소와 구별하기 위하여 사용하는 것이다. 따라서, 이하에서 언급되는 제1 구성요소는 본 발명의 기술적 사상 내에서 제2 구성요소일 수도 있음은 물론이다.

[0026] 다른 정의가 없다면, 본 명세서에서 사용되는 모든 용어(기술 및 과학적 용어를 포함)는 본 발명이 속하는 기술 분야의 통상의 기술자에게 공통적으로 이해될 수 있는 의미로 사용될 수 있을 것이다. 또한, 일반적으로 사용되는 사전에 정의되어 있는 용어들은 명백하게 특별히 정의되어 있지 않는 한 이상적으로 또는 과도하게 해석되지 않는다.

[0027] 명세서에서 사용되는 "부" 또는 "모듈"이라는 용어는 소프트웨어, FPGA 또는 ASIC과 같은 하드웨어 구성요소를 의미하며, "부" 또는 "모듈"은 어떤 역할들을 수행한다. 그렇지만 "부" 또는 "모듈"은 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. "부" 또는 "모듈"은 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다. 따라서, 일 예로서 "부" 또는 "모듈"은 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성요소들과

같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로 코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들 및 변수들을 포함한다. 구성요소들과 "부" 또는 "모듈" 들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 "부" 또는 "모듈" 들로 결합되거나 추가적인 구성요소들과 "부" 또는 "모듈" 들로 더 분리될 수 있다.

- [0028] 공간적으로 상대적인 용어인 "아래(below)", "아래(beneath)", "하부(lower)", "위(above)", "상부(upper)" 등은 도면에 도시되어 있는 바와 같이 하나의 구성요소와 다른 구성요소들과의 상관관계를 용이하게 기술하기 위해 사용될 수 있다. 공간적으로 상대적인 용어는 도면에 도시되어 있는 방향에 더하여 사용시 또는 동작시 구성요소들의 서로 다른 방향을 포함하는 용어로 이해되어야 한다. 예를 들어, 도면에 도시되어 있는 구성요소를 뒤집을 경우, 다른 구성요소의 "아래(below)"또는 "아래(beneath)"로 기술된 구성요소는 다른 구성요소의 "위(above)"에 놓여질 수 있다. 따라서, 예시적인 용어인 "아래"는 아래와 위의 방향을 모두 포함할 수 있다. 구성요소는 다른 방향으로도 배향될 수 있으며, 이에 따라 공간적으로 상대적인 용어들은 배향에 따라 해석될 수 있다.
- [0029] 본 명세서에서, 컴퓨터는 적어도 하나의 프로세서를 포함하는 모든 종류의 하드웨어 장치를 의미하는 것이고, 실시 예에 따라 해당 하드웨어 장치에서 동작하는 소프트웨어적 구성도 포괄하는 의미로서 이해될 수 있다. 예를 들어, 컴퓨터는 스마트폰, 태블릿 PC, 데스크톱, 노트북 및 각 장치에서 구동되는 사용자 클라이언트 및 애플리케이션을 모두 포함하는 의미로서 이해될 수 있으며, 또한 이에 제한되는 것은 아니다.
- [0030] 이하, 첨부된 도면을 참조하여 본 발명의 실시예를 상세하게 설명한다.
- [0031] 본 명세서에서 설명되는 각 단계들은 컴퓨터에 의하여 수행되는 것으로 설명되나, 각 단계의 주체는 이에 제한되는 것은 아니며, 실시 예에 따라 각 단계들의 적어도 일부가 서로 다른 장치에서 수행될 수도 있다.
- [0032] 도 1은 본 발명의 일 실시예에 따른 시스템의 구성도이다.
- [0033] 도 1에 도시된 바와 같이, 본 발명의 일 실시예에 따른 필터 조합 학습 네트워크 시스템은 제1 전자 장치(100), 서버(200) 및 제2 전자 장치(300)를 포함할 수 있다.
- [0034] 제1 전자 장치(100)는 서버(200)로부터 입력 필터, 입력 필터의 선형 조합을 위한 조합 계수 및 인공지능 모델에 대응되는 네트워크 구조를 수신하여 인공지능 모델의 특징맵을 획득하기 위한 구성이다.
- [0035] 제1 전자 장치(100)는 엣지 디바이스 또는 온-디맨드 디바이스로 구현될 수 있다.
- [0036] 제2 전자 장치(300)는 훈련된 인공지능 모델에 대한 데이터를 서버(200)로 전송하기 위한 구성이다. 구체적으로, 제2 전자 장치(300)는 학습된 인공지능 모델로부터 입력 필터, 조합 계수 및 네트워크 구조를 추출하여 서버(200)로 전송하는 역할을 수행할 수 있다. 제2 전자 장치(300)는 고성능의 인공지능 모델을 구현하기 위한 서버로 구성될 수 있다.
- [0037] 구체적으로, 제2 전자 장치(300)는, 인공지능 모델의 특징맵 획득을 위한 필터 F를 획득하고, 필터 F를 바탕으로 입력 필터 및 입력 필터의 선형 조합을 위한 조합 계수를 획득하고, 획득된 입력 필터 및 상기 획득된 조합 계수를 서버로 전송할 수 있다.
- [0038] 서버(200)는, 제2 전자 장치(300)로부터 수신된 데이터를, 제1 전자 장치(100)로 전송하는 역할을 수행하기 위한 구조이다. 즉, 서버(200)는, 입력 필터, 조합 계수 및 네트워크 구조를 제1 전자 장치(100)로 전송할 수 있다.
- [0039] 구체적으로, 도 3에 도시된 바와 같이, 제2 전자 장치(300)는 서버(200)를 통해 제1 전자 장치(100)로 인공지능 모델의 파라미터를 획득하기 위한 데이터를 전송하며, 제1 전자 장치(100)는 학습된 인공지능 파라미터를 그대로 저장하는 것이 아니라, 적은 데이터량을 가지는 입력 필터, 조합 계수 및 네트워크 구조만을 수신하여 인공지능 모델의 파라미터를 획득할 수 있다.
- [0040] 도 2는 본 발명의 일 실시예에 따른 필터 조합 학습 네트워크 시스템의 동작을 설명하기 위한 흐름도이다.
- [0041] 단계 S110에서, 서버(200)는, 입력 필터, 입력 필터의 선형 조합을 위한 조합 계수 및 인공지능 모델에 대응되는 네트워크 구조를 제1 전자 장치(100)로 전송할 수 있다.
- [0042] 이후, 단계 S120에서, 제1 전자 장치(100)는, 입력 필터, 조합 계수 및 네트워크 구조를 바탕으로 합성곱 필터를 획득할 수 있다.

- [0043] 이후, 단계 S130에서, 제1 전자 장치(100)는, 합성곱 필터를 바탕으로 인공지능 모델의 특징맵을 획득할 수 있다.
- [0044] 구체적으로, 도 3에 도시된 바와 같이, 제2 전자 장치(300)는 선형 조합을 위한 계수를 학습하고, 제1 전자 장치(100)는 특징맵 생성을 위한 합성곱 필터를 생성 또는 재구성할 수 있다.
- [0045] 이하 본 발명의 실시예를 설명하기에 앞서, 본 발명에서 사용되는 용어에 대해 정의한다.
- [0046] 먼저, F_i^j 는 i번째 계층의 j번째 필터, I_k 는 $k_w \times k_h$ 크기를 가지는 k번째 입력 필터, $a_{i,j}^{c,k}$ 는 i번째 계층의 j번째 필터에 대한 c번째 채널과 연결되는 k번째 입력 필터의 계수, C_i 는 i번째 계층의 채널의 수, n은 입력 필터의 개수, X_i 는 합성곱 연산을 통해서 출력으로 나오는 I번째 계층의 특징 맵, W_i , H_i 는 i번째 특징 맵의 가로와 세로 길이, k_w 및 k_h 는 합성곱 계층의 필터 커널 사이즈의 가로와 세로 길이를 의미한다.
- [0047] 이 경우, 본 발명의 일 실시예에 따른 입력 필터는, 도 4에 도시된 바와 같이, 입력 필터 집합(I)는 조합 계수 ($a_{i,j}^{c,k}$) 및 입력 필터 (I_0 내지 I_n)의 합성곱 연산으로 획득될 수 있다.
- [0048] 한편, 도 5에 도시된 바와 같이, 합성곱 필터 F는 입력 필터 및 조합 계수를 바탕으로 생성될 수 있다. 이에 대하여는 후술한다.
- [0049] 즉, 제1 전자 장치(100)는, CNN 모델의 필터를 학습하는 대신 입력 필터의 선형 조합을 사용해 합성곱 필터를 생성하도록 계수를 학습할 수 있다. 이때, 입력 필터는 고성능의 제2 전자 장치(300)로부터 획득될 수 있다.
- [0050] 일 실시예로, 제1 전자 장치(100)의 메모리가 기 설정된 값 이하인 경우, 제1 전자 장치(100)는 필터 조합 계층을 그대로 사용할 수 있다. 또 다른 실시예로, 제1 전자 장치(100)의 메모리가 기 설정된 값 초과인 경우, 필터 조합 계층으로 만들어진 합성곱 계층을 재구성할 수 있다.
- [0051] 후술하는 바와 같이, 인공지능 모델의 파라미터의 크기는 입력 필터의 수에 의해 결정된다. 따라서, 본 발명에서는 성능 또는 정확도에 큰 차이가 없는 범위 내에서 입력 필터의 개수를 획득하고, 획득된 입력 필터를 바탕으로 인공지능 모델의 파라미터를 생성할 수 있다.
- [0052] 이하에서는, 본 발명에 따른 합성곱 필터 획득 방법, 및 특징맵 획득 방법에 대하여 구체적으로 살펴본다.
- [0053] 일 실시예에 따라, 제1 전자 장치(100)는 하기 수학적 1을 바탕으로 합성곱 필터를 획득할 수 있다.

수학적 1

$$F_i^j = \left\{ \sum_{k=0}^n I_k \times a_{i,j}^{c,k} \right\} \text{ where } 0 < j \leq C_i, 0 < c \leq C_{i-1}$$

- [0055] 이후, 제1 전자 장치(100)는 획득된 합성곱 필터를 바탕으로 특징맵을 획득할 수 있다. 구체적으로, 제1 전자 장치(100)는 하기 수학적 2를 바탕으로 특징맵을 획득할 수 있다.

수학적 2

$$X_i = \{X_{i-1} * F_i^j\} \text{ where } 0 < j \leq C_i$$

- [0057] 즉, 도 6에 도시된 바와 같이, X_i 는 X_{i-1} 과 필터 F의 합성곱 연산을 바탕으로 획득된다.
- [0058] 상술한 실시예에 의한 경우, 제1 전자 장치(100)는 필터 F에 대한 모든 파라미터를 저장할 필요 없이, 입력 필터 및 조합 계수만을 저장하고, 필요한 경우, 상술한 수학적 1 및 수학적 2를 바탕으로 필터 F에 필요한 파라미터를 직접 생성하여 사용할 수 있다.

[0059] 즉, 필터 F에 대한 파라미터 생성 과정에서 제1 전자 장치(100)의 연산량은 다소 증가하나, 메모리가 절약될 수 있는 효과가 있으며, 이러한 본 발명의 실시예에 따라, 기존의 옛지 디바이스에서 메모리 문제로 딥러닝 적용이 어려웠던 문제점을 해결할 수 있게 된다.

[0060] 한편, 본 발명의 실시예에 따른 필터 F에 대한 파라미터의 크기보다 입력 필터 및 조합 계수를 이용한 선형 조합의 학습 파라미터의 크기가 더 작기 위해서는 입력 필터의 개수가 합성곱 계층의 필터 커널 사이즈(즉, $k_w \times k_h$)보다 작아야 한다. 즉, 입력 필터의 개수 n 은 하기 수학식 3의 조건을 만족해야 한다.

수학식 3

[0061]
$$n \leq k_w \times k_h$$

[0062] 이때, 본 발명의 일 실시예에 따라, 입력 필터의 개수 n 은, 인공지능 모델의 합성곱 계층의 커널이 3X3인 경우 n 은 3이고, 인공지능 모델의 합성곱 계층의 커널이 5X5인 경우 n 은 8일 때 후술하는 바와 같이 최적의 파라미터 감소율을 얻을 수 있다.

[0063] 구체적으로, i 번째 특징맵에서 합성곱 필터연산을 통해 $i+1$ 번째 특징 맵이 만들어 질 때 필요한 파라미터의 수는 $k_w \times k_h \times C_i \times C_{i+1}$ 이다. 이때, 본 발명의 실시예에 따르면, 각 필터는 n 개의 입력필터의 선형 조합으로 만들어지기 때문에 $k_w \times k_h$ 를 n 으로 대체할 수 있다. 즉 본 발명의 실시예에 따르면, 선형 조합을 위해서 필요한 학습 파라미터는 $n \times C_i \times C_{i+1}$ 이게 된다.

[0064] 결론적으로, 본 발명의 실시예에 따른 입력 필터를 이용한 파라미터 감소율 rr_{param} 는 하기 수학식 4와 같이 도출된다.

수학식 4

[0065]
$$rr_{param} = \frac{(k_w \times k_h \times C_i) \times C_{i+1}}{(n \times C_i) \times C_{i+1}} = \frac{k_w \times k_h}{n}$$

[0066] 즉, 상기 수학식 3의 조건을 만족하는 경우, 파라미터 감소율은 1 이상이며, 제1 전자 장치(100)의 메모리가 절약될 수 있다.

[0067] 한편, 본 발명의 일 실시예에 따라, 제1 전자 장치(100)의 메모리가 기 설정된 크기 이상인 경우, 합성곱 필터를 바탕으로 획득된 파라미터를 제1 전자 장치(100)의 메모리에 저장할 수 있다.

[0068] 즉, 제1 전자 장치(100)의 메모리가 충분한 경우, 제1 전자 장치(100)는 합성곱 연산을 수행할 필요 없이 파라미터를 저장하여 빠른 연산이 가능하게 할 수 있다.

[0069] 상술한 경우, 제1 전자 장치(100)는 필터 F에 대한 파라미터를 메모리에 저장하게 되지만, 서버(200)로부터 필터 F에 대한 파라미터를 수신하는 것이 아닌 입력 필터 및 조합 계수만을 수신함으로써, 데이터 다운로드량을 절감할 수 있는 효과가 발생한다.

[0070] 한편, 본 발명의 또 다른 실시예로, 제1 전자 장치(100)의 메모리가 기 설정된 크기 미만인 경우, 제1 전자 장치(100)는 가용 메모리를 판단하고, 가용 메모리를 바탕으로 입력 필터의 개수 n 을 결정할 수 있다.

[0071] 일 실시예에 따라, 입력 필터의 수는 사용자가 직접 설정할 수 있으며, 나아가, 인공지능 모델 학습을 위한 학습 파라미터로 활용될 수 있음은 물론이다.

[0072] 즉, 제1 전자 장치(100)는 메모리의 크기에 맞게 입력 필터의 개수를 조절할 수 있으며, 이 경우, 입력 필터의 개수에 따라 정확도가 다소 감소하는 경우라고 하더라도, 메모리 문제로 인해 인공지능 모델의 동작이 불가능했던 기존 옛지 디바이스 대비 인공지능 모델을 동작시킬 수 있는 새로운 효과가 발생한다.

[0073] 도 7 내지 도 11은 본 발명의 일 실시예에 따른 필터 조합 학습 네트워크 시스템에서의 파라미터 감소 효과를

설명하기 위한 예시도이다.

- [0074] 구체적으로 도 7 내지 도 11의 실험 그래프는 하기 조건 하에서 수행되었다.
- [0075] 먼저 실험 그래프는 VGG-16과 ResNet-18 모델을 사용하여 도출되었다. 데이터셋으로는 CIFAR10을 사용하며 VGG-16은 마지막 완전 연결 계층을 한 개로 사용하였다.
- [0076] 또한 Pytorch 프레임워크를 사용하여 구현하였으며 이미지 보강을 하기위해 랜덤한 확률로 이미지를 수평으로 뒤집고 패딩을 4만큼 추가해 32 X 32크기에 맞게 랜덤하게 잘라주었다.
- [0077] 이후, 정규화를 적용한 합성곱의 커널 크기는 3(또는 5)이며 Intel Core(TM) i7-3770K CPU와 RTX 2080Ti GPU를 사용하였다. 학습률은 스케줄링을 적용해 150, 250에폭에 0.1씩 급해지며 에폭은 250 배치 크기는 256을 사용하였다.
- [0078] 도 7은 커널의 크기가 3 x 3일 경우와 5 x 5일 경우 입력 필터의 개수(IBF)에 따른 정확도를 나타낸다.
- [0079] 정확도는 커널의 크기가 3 x 3일 경우에는 입력 필터의 개수가 3개 이상부터 유사하게 나타나며, 커널의 크기가 5 x 5일 경우 입력 필터의 개수가 8개 이상부터 유사하게 나타나는 것을 확인할 수 있다.
- [0080] 상술한 수학적 식 4에 의할 때, 입력 필터의 개수가 적을수록 메모리 절감 효과가 발생하므로, 커널의 크기가 3X3인 경우, 입력 필터는 3개($rr_{param}=3$), 커널의 크기가 5X5인 경우, 입력 필터는 8개($rr_{param}=3.125$)인 경우 가장 높은 효율을 보일 것이다.
- [0081] 구체적으로, VGG16의 경우 파라미터는 14.738M을 가지며 기존 ResNet18의 경우 11.174M을 가진다. 필터 조합 학습 방식을 사용하면 VGG16은 4.948M을 가지고 ResNet18은 3.854M을 가지게 된다.
- [0082] 즉, 제1 전자 장치(100)는 상술한 입력 필터의 설정에 따라, 기존 대비 메모리를 3배 이상 절감할 수 있다.
- [0083] 도 8은 입력 필터의 수에 따른 인공지능 모델의 학습 정확도의 수렴 변화도를 나타낸 그래프이다.
- [0084] 도 8에 도시된 바와 같이, 입력 필터의 개수가 적은 경우, 학습 정확도가 수렴하지 않는 것을 확인할 수 있다($n=1$ 및 2의 경우). 그러나, 필터의 개수가 3이상인 경우부터는, 입력 필터의 개수가 증가한다고 해서 학습 정확도가 크게 향상되는 것이 아님을 확인할 수 있다(커널 크기 3X3의 경우).
- [0085] 도 9는 VGG16에서의 편향 파라미터 (bias weight parameter)의 유무에 따른 예측 정확도를 도시한 그래프이다.
- [0086] 전체적으로 편향이 있어야 성능이 안정적인 것으로 해석할 수 있으며, 도 9에 도시된 바와 같이, 입력 필터의 개수가 3개 이상인 경우부터는, 안정적인 정확도를 가지는 것을 확인할 수 있다(커널 크기 3X3의 경우).
- [0087] 도 10은 VGG16에서 입력 필터의 개수에 따른 정확도를 나타낸 그래프이다.
- [0088] 마찬가지로, 입력 필터의 개수가 3개 이상인 경우, 정확도가 유사한 것을 확인할 수 있다.
- [0089] 도 11은 VGG16에서 입력 필터를 사용한 FCL 모델과 기본 모델에서 모델 압축을 위해 프루닝(pruning)을 적용할 경우 성능차이를 나타내는 그래프이다.
- [0090] 구체적으로, 도 11은 모든 파라미터의 절대값을 정렬하고 값의 크기가 가장 작은 값들을 0으로 만드는 프루닝(pruning)을 할 때 감소하는 비율이 어느 정도인지를 그래프에 나타낸 것이며 양자의 성능이 70%까지는 유사한 것을 확인할 수 있다.
- [0091] 결론적으로, 본 발명의 실시예에 따르면, 입력 필터를 이용하는 경우, 인공지능 모델의 경량화 모델링이 가능하며, 이때 입력 필터의 개수가 적을수록 더 클 경량화를 수행할 수 있게 된다.
- [0092] 다만, 인공지능 모델을 경량화 하게 되면 인공지능 모델의 정확도가 떨어지는 단점이 존재하게 된다. 그러나, 상술한 실험예를 살펴보면, 효율적인 경량화를 위한 입력 필터의 개수를 찾을 수 있음을 확인할 수 있다.
- [0093] 한편, 본 발명에 따른 인공지능 기술은 기계학습(딥러닝) 및 기계학습을 활용한 요소 기술들로 구성된다.
- [0094] 기계학습은 입력 데이터들의 특징을 스스로 분류/학습하는 알고리즘 기술이며, 요소기술은 딥러닝 등의 기계학습 알고리즘을 활용하여 인간 두뇌의 인지, 판단 등의 기능을 모사하는 기술로서, 언어적 이해, 시각적 이해, 추론/예측, 지식 표현, 동작 제어 등의 기술 분야로 구성된다.
- [0095] 구체적으로, 딥 러닝은 여러 비선형 변환기법의 조합을 통해 높은 수준의 추상화(abstractions, 다량의 데이터

나 복잡한 자료들 속에서 핵심적인 내용 또는 기능을 요약하는 작업)를 시도하는 기계학습(machine learning) 알고리즘의 집합으로 정의된다. 딥 러닝은 큰 틀에서 사람의 사고방식을 컴퓨터에게 가르치는 기계학습의 한 분야로 볼 수 있다.

[0096] 어떠한 데이터가 있을 때 이를 컴퓨터가 알아들을 수 있는 형태(예를 들어 영상의 경우는 픽셀정보를 열벡터로 표현하는 등)로 표현(representation)하고 이를 학습에 적용하기 위해 많은 연구(어떻게 하면 더 좋은 표현기법을 만들고 또 어떻게 이것들을 학습할 모델을 만들지에 대한)가 진행되고 있다. 이러한 노력의 결과로 다양한 딥 러닝 기법들이 개발되었다. 딥 러닝 기법들로는 심층 신경망(Deep Neural Networks, DNN), 합성곱 신경망(Convolutional deep Neural Networks, CNN), 순환 신경망(Reccurent Neural Network, RNN) 및 심층 신뢰 신경망(Deep Belief Networks, DBN)을 예로 들 수 있다.

[0097] 심층 신경망(Deep Neural Networks, DNN)은 입력 계층(input layer)과 출력 계층(output layer) 사이에 복수개의 은닉 계층(hidden layer)들로 이뤄진 인공신경망(Artificial Neural Network, ANN)이다.

[0098] 이때, 심층 신경망의 구조는 퍼셉트론(perceptron)으로 구성될 수 있다. 퍼셉트론은 여러 개의 입력 값(input)과 하나의 프로세서(prosessor), 하나의 출력 값으로 구성된다. 프로세서는 여러 개의 입력 값에 각각 가중치(또는 파라미터)를 곱한 후, 가중치(또는 파라미터)가 곱해진 입력 값들을 모두 합한다. 그 다음 프로세서는 합해진 값을 활성화함수에 대입하여 하나의 출력 값을 출력한다. 만약 활성화함수의 출력 값으로 특정한 값이 나오기를 원하는 경우, 각 입력 값에 곱해지는 가중치(또는 파라미터)를 수정하고, 수정된 가중치(또는 파라미터)를 이용하여 출력 값을 다시 계산할 수 있다. 이때, 각각의 퍼셉트론은 서로 다른 활성화함수를 사용할 수 있다. 또한 각각의 퍼셉트론은 이전 계층에서 전달된 출력들을 입력으로 받아들이고 다음, 활성화 함수를 이용해서 출력을 구한다. 구해진 출력은 다음 계층의 입력으로 전달된다. 상술한 바와 같은 과정을 거치면 최종적으로 몇 개의 출력 값을 얻을 수 있다.

[0099] 합성곱 신경망(Convolutional deep Neural Networks, CNN)은 최소한의 전처리(preprocess)를 사용하도록 설계된 다계층 퍼셉트론(multilayer perceptrons)의 한 종류이다. 합성곱 신경망은 하나 또는 여러개의 합성곱 계층(convolutional layer)과 그 위에 올려진 일반적인 인공신경망 계층들로 이루어져 있으며, 가중치(또는 파라미터)와 통합 계층(pooling layer)들을 추가로 활용한다. 이러한 구조 덕분에 합성곱 신경망은 2차원 구조의 입력 데이터를 충분히 활용할 수 있다. 또한, 합성곱 신경망은 표준 역전달을 통해 훈련될 수 있다. 합성곱 신경망은 다른 피드포워드 인공신경망 기법들보다 쉽게 훈련되는 편이고 적은 수의 매개변수를 사용한다는 이점이 있다. 합성곱 신경망은 입력 영상에 대하여 합성곱과 서브샘플링을 번갈아 수행함으로써 입력 영상으로부터 특징을 추출한다.

[0100] 합성곱 신경망은 여러 개의 합성곱 계층(Convolution layer), 여러 개의 서브샘플링 계층(Subsampling layer, Lactal pooling layer, Max-Pooling layer), 완전 연결 층(Fully Connected layer)을 포함한다. 합성곱 계층은 입력 영상(Input Image)에 대해 합성곱을 수행하는 계층이다. 그리고 서브샘플링 계층은 입력 영상에 대해 지역적으로 최대값을 추출하여 2차원 영상으로 매핑하는 계층으로, 국소적인 영역을 더 크게 하고, 서브샘플링을 수행한다.

[0101] 합성곱 계층에서는 커널의 크기(kernel size), 사용할 커널의 개수(즉, 생성할 맵의 개수), 및 합성곱 연산 시에 적용할 가중치(또는 파라미터) 테이블 등의 정보가 필요하다. 예를 들어, 입력 영상의 크기가 32×32 이고, 커널의 크기가 5×5 이고, 사용할 커널의 개수가 20개인 경우를 예로 들자. 이 경우, 32×32 크기의 입력 영상에 5×5 크기의 커널을 적용하면, 입력 영상의 위, 아래, 왼쪽, 오른쪽에서 각각 2개의 픽셀(pixel)에는 커널을 적용하는 것이 불가능하다. 입력 영상의 위에 커널을 배치한 후 합성곱을 수행하면, 그 결과 값인 '-8'은 커널에 포함된 입력 영상의 픽셀들 중에서 커널의 중심요소(center element)에 대응하는 픽셀의 값으로 결정되기 때문이다. 따라서, 32×32 크기의 입력 영상에 5×5 크기의 커널을 적용하여 합성곱을 수행하면 28×28 크기의 맵(map)이 생성된다. 앞서, 사용할 커널의 개수가 총 20개인 경우를 가정하였으므로, 첫 번째 합성곱 계층에서는 총 20개의 28×28 크기의 맵이 생성된다.

[0102] 서브샘플링 계층에서는 서브샘플링할 커널의 크기에 대한 정보, 커널 영역 내의 값들 중 최대값을 선택할 것인지 최소값을 선택할 것인지에 대한 정보가 필요하다.

[0103] 딥 러닝 기법에 대한 설명으로 다시 돌아가면, 순환 신경망(Reccurent Neural Network, RNN)은 인공신경망을 구성하는 유닛 사이의 연결이 Directed cycle을 구성하는 신경망을 말한다. 순환 신경망은 앞먹임 신경망과 달리, 임의의 입력을 처리하기 위해 신경망 내부의 메모리를 활용할 수 있다.

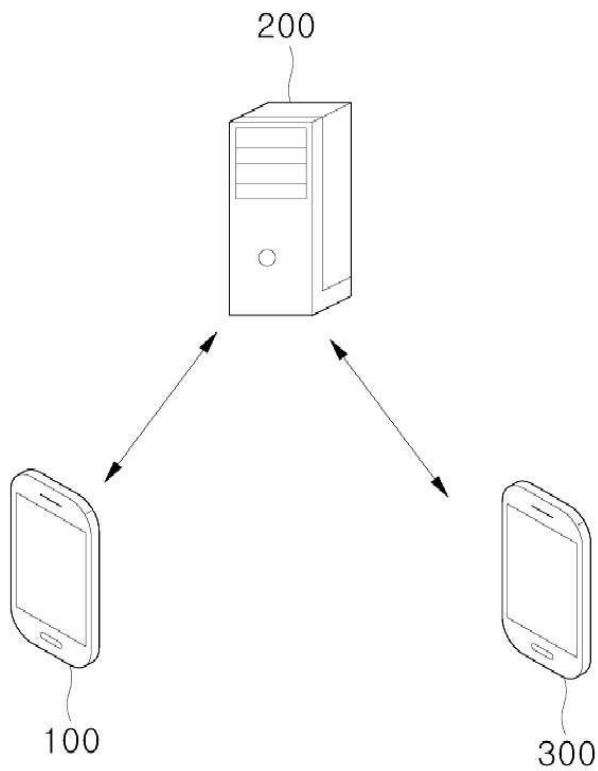
- [0104] 심층 신뢰 신경망(Deep Belief Networks, DBN)이란 기계학습에서 사용되는 그래프 생성 모형(generative graphical model)으로, 딥 러닝에서는 잠재변수(latent variable)의 다중계층으로 이루어진 심층 신경망을 의미한다. 계층 간에는 연결이 있지만 계층 내의 유닛 간에는 연결이 없다는 특징이 있다.
- [0105] 심층 신뢰 신경망은 생성 모형이라는 특성상 선행학습에 사용될 수 있고, 선행학습을 통해 초기 가중치(또는 파라미터)를 학습한 후 역전과 혹은 다른 판별 알고리즘을 통해 가중치(또는 파라미터)의 미조정을 할 수 있다. 이러한 특성은 훈련용 데이터가 적을 때 굉장히 유용한데, 이는 훈련용 데이터가 적을수록 가중치(또는 파라미터)의 초기값이 결과적인 모델에 끼치는 영향이 세지기 때문이다. 선행학습된 가중치(또는 파라미터) 초기값은 임의로 설정된 가중치(또는 파라미터) 초기값에 비해 최적의 가중치(또는 파라미터)에 가깝게 되고 이는 미조정 단계의 성능과 속도향상을 가능케 한다.
- [0106] 상술한 인공지능 및 그 학습방법에 관한 내용은 예시를 위하여 서술된 것이며, 이하에서 설명되는 실시 예들에서 이용되는 인공지능 및 그 학습방법은 제한되지 않는다. 예를 들어, 당 업계의 통상의 기술자가 동일한 과제 해결을 위하여 적용할 수 있는 모든 종류의 인공지능 기술 및 그 학습방법이 개시된 실시 예에 따른 시스템을 구현하는 데 활용될 수 있다.
- [0107] 도 12는 본 발명의 일 실시예에 따른 장치의 구성도이다.
- [0108] 프로세서(102)는 하나 이상의 코어(core, 미도시) 및 그래픽 처리부(미도시) 및/또는 다른 구성 요소와 신호를 송수신하는 연결 통로(예를 들어, 버스(bus) 등)를 포함할 수 있다.
- [0109] 일 실시예에 따른 프로세서(102)는 메모리(104)에 저장된 하나 이상의 인스트럭션을 실행함으로써, 도 2와 관련하여 설명된 방법을 수행한다.
- [0110] 한편, 프로세서(102)는 프로세서(102) 내부에서 처리되는 신호(또는, 데이터)를 일시적 및/또는 영구적으로 저장하는 램(RAM: Random Access Memory, 미도시) 및 롬(ROM: Read-Only Memory, 미도시)을 더 포함할 수 있다. 또한, 프로세서(102)는 그래픽 처리부, 램 및 롬 중 적어도 하나를 포함하는 시스템온칩(SoC: system on chip) 형태로 구현될 수 있다.
- [0111] 메모리(104)에는 프로세서(102)의 처리 및 제어를 위한 프로그램들(하나 이상의 인스트럭션들)을 저장할 수 있다. 메모리(104)에 저장된 프로그램들은 기능에 따라 복수 개의 모듈들로 구분될 수 있다.
- [0112] 본 발명의 실시예와 관련하여 설명된 방법 또는 알고리즘의 단계들은 하드웨어로 직접 구현되거나, 하드웨어에 의해 실행되는 소프트웨어 모듈로 구현되거나, 또는 이들의 결합에 의해 구현될 수 있다. 소프트웨어 모듈은 RAM(Random Access Memory), ROM(Read Only Memory), EPROM(Erasable Programmable ROM), EEPROM(Electrically Erasable Programmable ROM), 플래시 메모리(Flash Memory), 하드 디스크, 착탈형 디스크, CD-ROM, 또는 본 발명이 속하는 기술 분야에서 잘 알려진 임의의 형태의 컴퓨터 판독가능 기록매체에 상주할 수도 있다.
- [0113] 본 발명의 구성 요소들은 하드웨어인 컴퓨터와 결합되어 실행되기 위해 프로그램(또는 애플리케이션)으로 구현되어 매체에 저장될 수 있다. 본 발명의 구성 요소들은 소프트웨어 프로그래밍 또는 소프트웨어 요소들로 실행될 수 있으며, 이와 유사하게, 실시 예는 데이터 구조, 프로세스들, 루틴들 또는 다른 프로그래밍 구성들의 조합으로 구현되는 다양한 알고리즘을 포함하여, C, C++, 자바(Java), 어셈블리(assembler) 등과 같은 프로그래밍 또는 스크립팅 언어로 구현될 수 있다. 기능적인 측면들은 하나 이상의 프로세서들에서 실행되는 알고리즘으로 구현될 수 있다.
- [0114] 이상, 첨부된 도면을 참조로 하여 본 발명의 실시예를 설명하였지만, 본 발명이 속하는 기술분야의 통상의 기술자는 본 발명이 그 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 실시될 수 있다는 것을 이해할 수 있을 것이다. 그러므로, 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며, 제한적이지 않은 것으로 이해해야만 한다.

부호의 설명

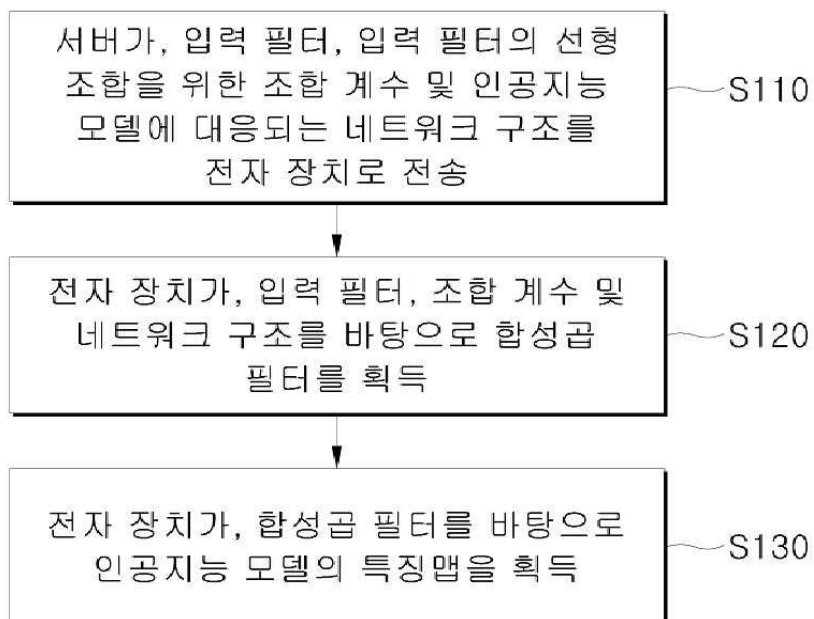
- [0115] 100 : 제1 전자 장치
200 : 서버
300 : 제2 전자 장치

도면

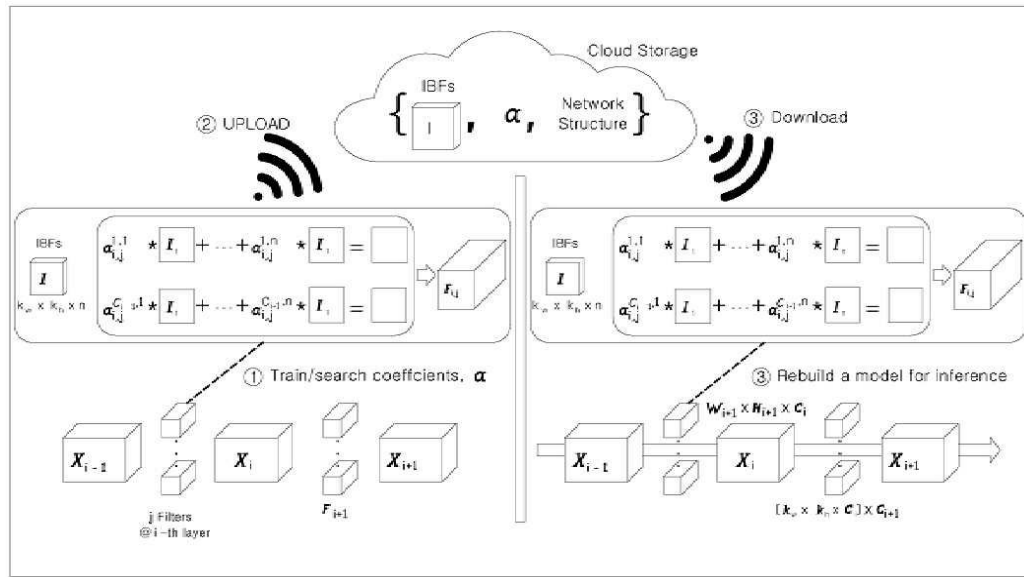
도면1



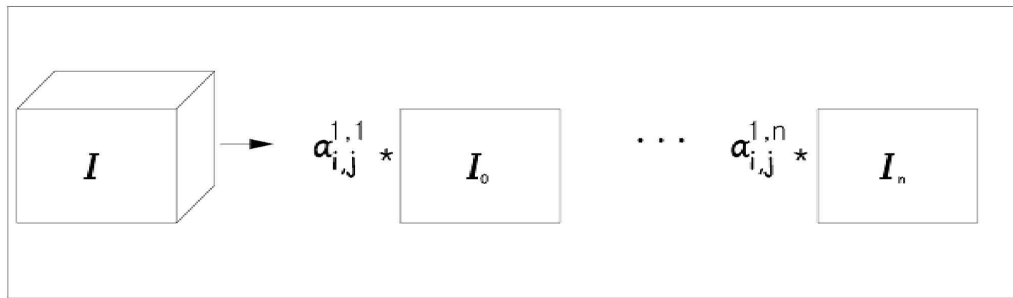
도면2



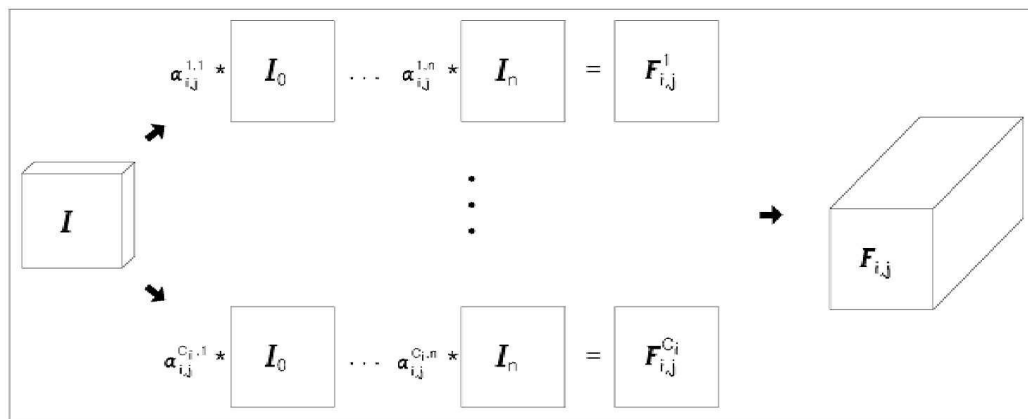
도면3



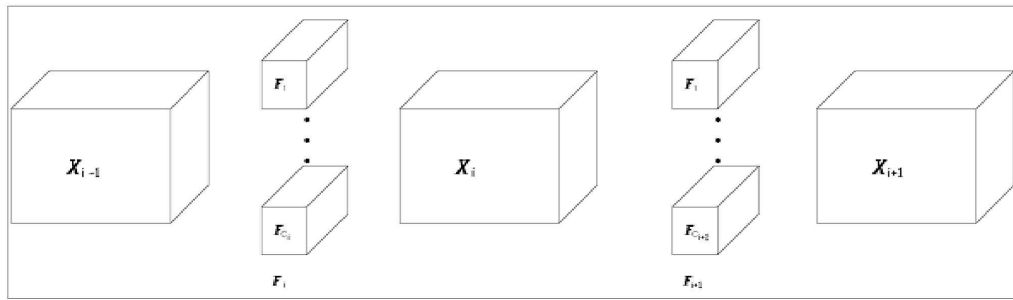
도면4



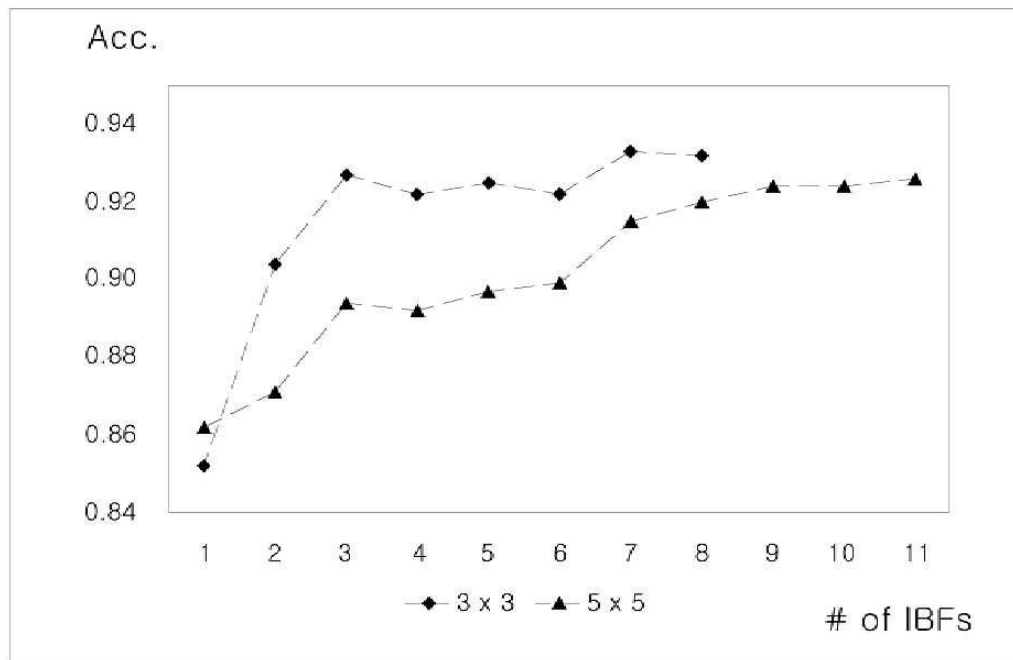
도면5



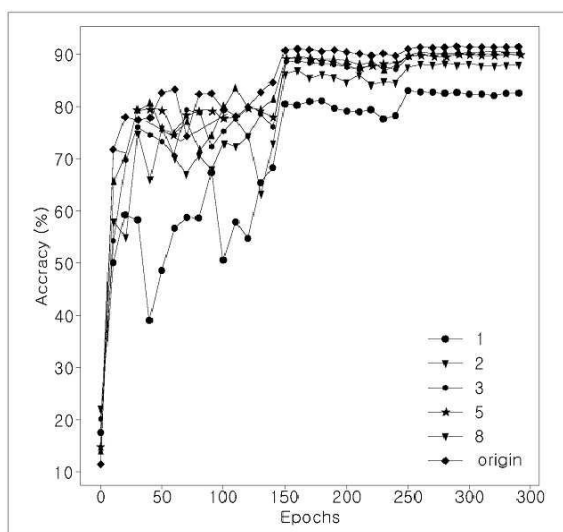
도면6



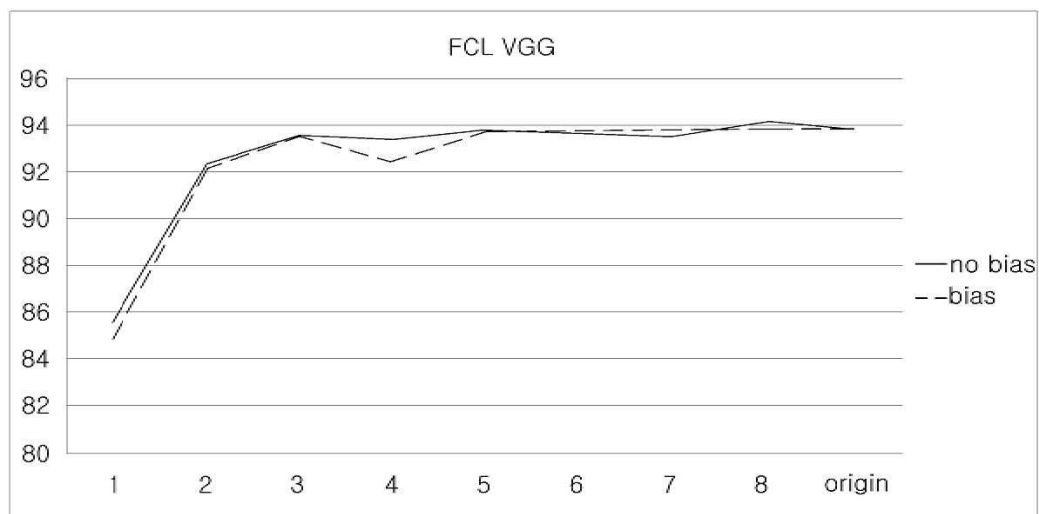
도면7



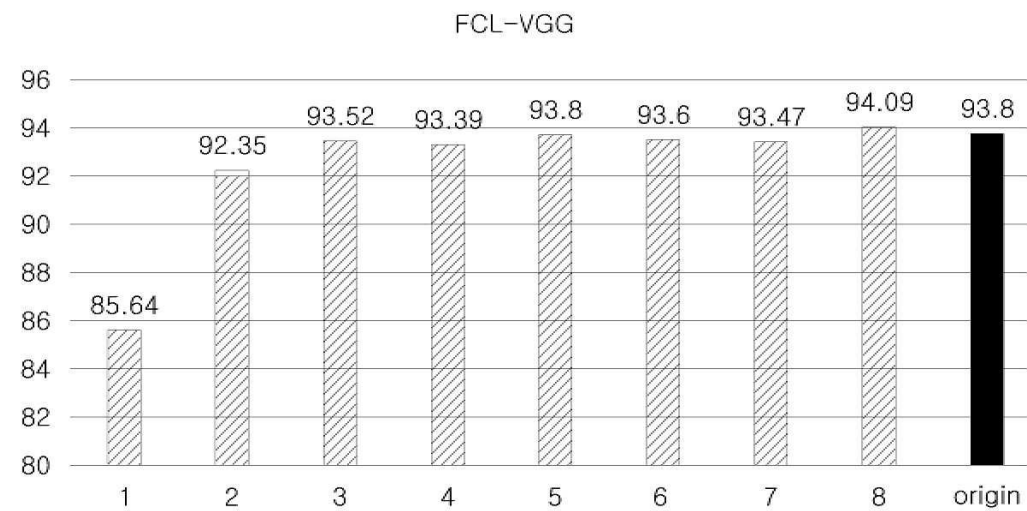
도면8



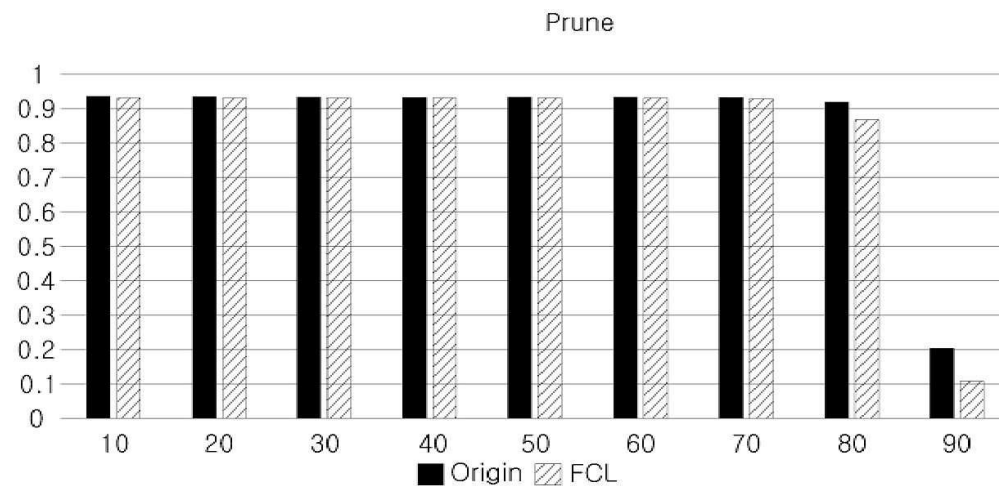
도면9



도면10



도면11



도면12

