



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0123521
(43) 공개일자 2024년08월14일

(51) 국제특허분류(Int. Cl.)
G06N 3/098 (2023.01)

(52) CPC특허분류
G06N 3/098 (2023.01)

(21) 출원번호 10-2023-0015973

(22) 출원일자 2023년02월07일

심사청구일자 2023년02월07일

(71) 출원인
고려대학교 산학협력단

서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)

(72) 발명자

백상현

서울특별시 서초구 고무래로 35, 109동 1004호
차동주

서울특별시 동작구 사당로 54, 201호

정대영

서울특별시 동대문구 정릉천동로 36, 109동 1703호

(74) 대리인

김등용

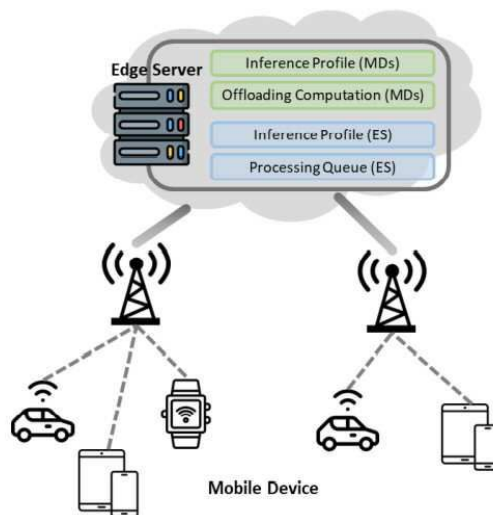
전체 청구항 수 : 총 7 항

(54) 발명의 명칭 심층 신경망 응용을 위한 분할 컴퓨팅 방법 및 장치

(57) 요약

심층 신경망 응용을 위한 분할 컴퓨팅 장치 및 방법이 개시된다. 상기 분할 컴퓨팅 방법은 적어도 프로세서를 포함하는 분할 컴퓨팅 장치에 의해 수행되고, 복수의 모바일 단말들 각각으로부터 계층 분할 지점 결정 요청을 수신하는 단계, 상기 계층 분할 지점 결정 요청에 응답하여, 상기 복수의 모바일 단말들 각각의 계층 분할 지점을 결정하는 단계, 상기 계층 분할 지점을 상기 복수의 모바일 단말들 각각으로 송신하는 단계, 상기 복수의 모바일 단말들 각각으로부터 제1 중간 데이터를 수신하는 단계, 상기 제1 중간 데이터 중간 모델에 입력하여 제2 중간 데이터를 도출하는 단계, 및 상기 제2 중간 데이터를 상기 모바일 단말들 각각으로 송신하는 단계를 포함한다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호	1711152525
과제번호	2021-0-01810-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	양자세계를 연결하는 초신뢰 양자인터넷 요소 기술 개발
기 여 율	1/2
과제수행기관명	고려대학교산학협력단
연구기간	2022.01.01 ~ 2022.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1711160326
과제번호	2022-0-01015-001
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	6G핵심기술개발
연구과제명	지능형 6G 모바일 코어 네트워크 후보 요소기술 개발
기 여 율	1/2
과제수행기관명	고려대학교 산학협력단
연구기간	2022.04.01 ~ 2022.12.31

명세서

청구범위

청구항 1

적어도 프로세서를 포함하는 분할 컴퓨팅 장치에 의해 수행되는 분할 컴퓨팅 방법에 있어서,
복수의 모바일 단말들 각각으로부터 계층 분할 지점 결정 요청을 수신하는 단계;
상기 계층 분할 지점 결정 요청에 응답하여, 상기 복수의 모바일 단말들 각각의 계층 분할 지점을 결정하는 단계;
상기 계층 분할 지점을 상기 복수의 모바일 단말들 각각으로 송신하는 단계;
상기 복수의 모바일 단말들 각각으로부터 제1 중간 데이터를 수신하는 단계;
상기 제1 중간 데이터 중간 모델에 입력하여 제2 중간 데이터를 도출하는 단계; 및
상기 제2 중간 데이터를 상기 모바일 단말들 각각으로 송신하는 단계를 포함하는 분할 컴퓨팅 방법.

청구항 2

제1항에 있어서,
상기 계층 분할 지점은, 분할 컴퓨팅을 위해 임의의 심층 신경망 모델을 분할하는 지점을 의미하고,
상기 계층 분할 지점에 따라 상기 임의의 심층 신경망 모델은 헤드(head) 모델, 중간(middle) 모델, 및 테일(tail) 모델로 분할되고,
상기 제1 중간 데이터는 상기 헤드 모델의 출력 데이터이고,
상기 제2 중간 데이터는 상기 중간 모델의 출력 데이터인,
분할 컴퓨팅 방법.

청구항 3

제1항에 있어서,
상기 계층 분할 지점을 결정하는 단계는,
상기 모바일 단말들의 평균 추론 지연 시간을 최소화하기 위한 목적 함수를 이용한 최적화 문제를 풀이함으로써, 상기 계층 분할 지점을 결정하는,
분할 컴퓨팅 방법.

청구항 4

제3항에 있어서,
상기 계층 분할 지점을 결정하는 단계는 적어도 하나의 제약 조건을 이용하여 상기 최적화 문제를 풀이하고,
상기 적어도 하나의 제약 조건은, 상기 복수의 모바일 단말들 각각에 대한 제1 계층 분할 지점은 최대 1 개가 존재하여야 한다는 제1 제약 조건을 포함하는,
분할 컴퓨팅 방법.

청구항 5

제4항에 있어서,
상기 적어도 하나의 제약 조건은, 상기 복수의 모바일 단말들 각각에 대한 제2 계층 분할 지점은 최대 1 개가 존재하여야 한다는 제2 제약 조건을 포함하는,

분할 컴퓨팅 방법.

청구항 6

제5항에 있어서,

상기 적어도 하나의 제약 조건은, 상기 제1 계층 분할 지점은 상기 제2 계층 분할 지점보다 작거나 같아야 한다는 제3 제약 조건을 포함하는,

분할 컴퓨팅 방법.

청구항 7

제6항에 있어서,

상기 적어도 하나의 제약 조건은, 수학식 1로 표현되는 제4 제약 조건을 포함하고,

$$\frac{(\sum_{m=1}^M P_m)^2}{M \sum_{m=1}^M P_m^2} \geq \theta_F \quad (\frac{1}{M} \leq \theta_F \leq 1)$$

상기 수학식 1은 이고,

상기 P_m 은 모바일 단말 m 이 상기 분할 컴퓨팅 장치로 오프로딩하는 연산량이고,

상기 M 은 상기 모바일 단말의 개수이고,

상기 θ_F 는 미리 정해진 값을 갖는 공평성 임계치(fairness threshold)인,

분할 컴퓨팅 방법.

발명의 설명

기술 분야

[0001] 본 발명은 멀티 모바일 단말 엣지 컴퓨팅 환경에서의 분할 컴퓨팅에 관한 것이다.

배경 기술

[0002] 기존의 DNN(Deep Neural Network, 심층 신경망) 추론 방식은 전체 DNN 모델을 단일 장치(e.g., 모바일 단말 또는 엣지 서버 등)에서 처리하여 결과를 도출하는 구조를 갖는다. 추론(Inference)이란 DNN 모델 파라미터들의 학습(training)이 완료된 이후 학습된 모델을 바탕으로 학습에 사용되지 않은 새로운 데이터를 입력으로 넣어 결과를 도출하는 과정이다.

[0003] 기존의 DNN 추론 방식은 전체 DNN 추론을 모바일 단말에서 수행하는 경우와 엣지 서버에서 수행하는 경우로 크게 두 가지의 방식이 있다. 먼저, 전체 DNN 추론을 모바일 단말에서 수행하는 경우에는 로데이터(raw data)를 다른 장치로 보낼 필요 없이, 모바일 단말에서 자체적으로 처리하지만, 모바일 단말의 부족한 컴퓨팅 자원으로 인해 높은 추론 지연 시간을 발생시킬 수 있다. 반면, 로데이터를 엣지 서버로 전송하여 전체 DNN 추론을 엣지 서버에서 할 때는 컴퓨팅 자원이 충분하여 추론 지연 시간 측면에서는 이득을 볼 수 있지만, 높은 전송 지연 시간 시간이 발생하는 문제점이 있다.

[0004] 분할 컴퓨팅(Split Computing)은 위와 같이 단일 장치에서의 DNN 추론 방식이 가지는 문제점을 극복하기 위해 여러 장치를 통해 순차적으로 추론을 하는 구조를 제시한다. 먼저, 분할 컴퓨팅은 DNN 모델의 특정 계층을 계층 분할 지점(Split Point)을 선택하여 DNN 모델을 앞부분(head)과 뒷부분(tail)으로 나눈다. 이를 기반으로 앞부분은 모바일 단말에서 처리하고, 앞부분의 중간 데이터(intermediate data)를 엣지 서버로 전송하여 뒷부분은 엣지 서버에서 처리하는 구조를 가진다. 이러한 구조는 여러 장치의 컴퓨팅 자원, 통신 상태, 에너지 등의 상태에 따라서 적절한 계층 분할 지점을 선택하여 기존의 단일 장치에서의 DNN 추론 방식의 문제점을 극복한다. 계층 분할 지점에 따라 추론 지연 시간, 장치 자원 사용량 등이 달라지기 때문에 적절한 계층 분할 지점을 선택하는 것이 중요하다.

[0005] 최근에는, 인공지능 기술이 발전하면서 다양한 구조를 가진 DNN 모델이 등장하고 있다. 저해상도의 이미지를 입력으로 받아, 고해상도의 이미지를 출력으로 도출하는 super resolution 모델은 데이터 크기가 상당히 결과를

도출하는 구조로 되어 있다. 이러한 구조를 가진 DNN 모델은, 엣지 서버에서 DNN 모델의 뒷부분을 처리하고 결과를 다시 모바일 단말로 반환하는데 상당히 많은 전송 오버헤드를 발생시켜 추론 지연 시간이 급증하게 된다. 이러한 문제점을 해결하기 위해 본 발명에서는 계층 분할 지점을 두 개로 결정하여, DNN 모델을 앞부분(head), 중간 부분(middle), 뒷부분(tail)으로 나누는 구조를 제안한다. 즉, 두 개의 계층 분할 지점을 결정하는 방안을 제안함으로써, 모바일 단말에서 앞부분을 처리하고, 엣지 서버에서 중간 부분을 처리하고, 뒷부분을 다시 모바일 단말에서 처리함으로써 데이터 크기가 큰 출력을 엣지 서버에서 모바일 단말로 전송하지 않아도 되는 구조를 제안한다.

[0006] 또한, 하나의 엣지 서버가 서비스 제공자(service provider)로서 여러 개의 모바일 단말들에 서비스를 제공해주는 환경에서, 컴퓨팅 자원이 부족한 모바일 단말들은 컴퓨팅 자원이 충분한 모바일 단말들보다 더 많은 양의 연산량을 오프로딩(offloading) 한다. 각각의 모바일 단말들이 엣지 서버의 자원을 위해 비용을 지불하는 환경에서, 컴퓨팅 자원이 많다는 이유로 엣지 서버의 자원을 충분히 활용하지 못하는 것은 컴퓨팅 자원이 충분한 모바일 단말에는 불공평하다는 문제점이 발생한다. 이에 따라, 본 발명에서는 여러 모바일 단말들이 엣지 서버의 자원을 공평하게 사용(공평한 오프로딩 기회)할 수 있는 계층 분할 지점을 결정하는 방안을 제안한다.

[0007] 이를 위해 본 발명에서는 분할 컴퓨팅 시 추론 시간을 최소화하고 모바일 단말 간 엣지 서버로 오프로딩하는 연산량에 대한 공정성(fairness)를 보장하는 계층 분할 지점을 결정하기 위한 F2SC(Fast and Fair Split Computing) 알고리즘을 제안한다. 먼저, 엣지 서버가 여러 모바일 단말에 공평한 오프로딩 기회를 보장해주기 위해 JFI(Jain' Fairness Index)를 이용하여 문제 공식화를 하고, 준최적(sub-optimal)의 결과를 현실적인 시간 내에 구할 수 있는 F2SC 알고리즘을 제안한다.

선행기술문헌

특허문헌

[0008] (특허문헌 0001) 대한민국 등록특허 제2113662호 (2020.05.22. 공고)
(특허문헌 0002) 대한민국 공개특허 제2022-0113714호 (2022.08.16. 공개)

비특허문헌

[0009] (비특허문헌 0001) R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System," in arXiv:cs/9809099, September 1998.

발명의 내용

해결하려는 과제

[0010] 본 발명이 이루고자 하는 기술적인 과제는 추론 지연 시간을 최소화하고 공정성을 보장하는 분할 컴퓨팅 방법 및 장치를 제공하는 것이다.

과제의 해결 수단

[0011] 본 발명의 일 실시예에 따른 분할 컴퓨팅 방법은 적어도 프로세서를 포함하는 분할 컴퓨팅 장치에 의해 수행되고, 복수의 모바일 단말들 각각으로부터 계층 분할 지점 결정 요청을 수신하는 단계, 상기 계층 분할 지점 결정 요청에 응답하여, 상기 복수의 모바일 단말들 각각의 계층 분할 지점을 결정하는 단계, 상기 계층 분할 지점을 상기 복수의 모바일 단말들 각각으로 송신하는 단계, 상기 복수의 모바일 단말들 각각으로부터 제1 중간 데이터를 수신하는 단계, 상기 제1 중간 데이터 중간 모델에 입력하여 제2 중간 데이터를 도출하는 단계, 및 상기 제2 중간 데이터를 상기 모바일 단말들 각각으로 송신하는 단계를 포함한다.

발명의 효과

[0012] 본 발명의 실시예들에 의한 경우, 추론 지연 시간을 최소화하는 동시에 모바일 단말 간 엣지 서버로 오프로딩하

는 연산량에 대한 공정성을 보장하는 계층 분할 지점 결정이 가능하다.

도면의 간단한 설명

- [0013] 본 발명의 상세한 설명에서 인용되는 도면을 보다 충분히 이해하기 위하여 각 도면의 상세한 설명이 제공된다.
- 도 1은 본 발명에서 제안하는 분할 컴퓨팅 시스템을 도시한다.
- 도 2는 두 개의 계층 분할 지점이 결정된 DNN 모델을 도시한다.
- 도 3은 계층 분할 지점 결정 알고리즘의 수도 코드를 도시한다.
- 도 4는 엣지 서버 처리 큐의 예약 예시1을 나타낸다.
- 도 5는 엣지 서버 처리 큐의 예약 예시2를 나타낸다.
- 도 6은 본 발명의 일 실시예에 따른 분할 컴퓨팅 방법을 설명하기 위한 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0014] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시예들에 대해서 특정한 구조적 또는 기능적 설명들은 단지 본 발명의 개념에 따른 실시예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시예들은 다양한 형태로 실시될 수 있으며 본 명세서에 설명된 실시예들에 한정되지 않는다.
- [0015] 본 발명의 개념에 따른 실시예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.
- [0016] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.
- [0017] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0018] 본 명세서에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0019] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0020] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시예들을 상세히 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.
- [0021] DNN(Deep Neural Network, 심층 신경망) 모델의 계층 분할 지점을 결정하기 위한 시스템 모델을 소개하고, 분할 컴퓨팅의 전반적인 과정을 자세히 설명한다.

- [0022] 도 1은 모바일 단말간 공평성(또는 공정성)을 보장하고 추론 지연 시간을 최소화하는 분할 컴퓨팅을 수행하기 위한 시스템 모델(분할 컴퓨팅 시스템 등으로 명명될 수 있음)이다. 복수의 모바일 단말들과 하나의 (엣지) 서버(분할 컴퓨팅 장치 등으로 명명될 수 있음)로 구성되고, DNN 추론을 수행하기 위해 모바일 단말과 엣지 서버에서 분할 컴퓨팅을 활용한다.
- [0023] 모바일 단말은 m ($m \in \mathbb{M} = \{1, 2, 3, \dots, M\}$) 으로 표현하고, 각각의 모바일 단말은 서로 다른 컴퓨팅 자원 (Floating point Operations per Second, FLOPS) C_m 을 가진다. 모바일 단말에서 DNN 기반의 애플리케이션이 발생하면 모바일 단말들은 엣지 서버로 적절한 계층 분할 지점에 대한 결정을 요청한다.
- [0024] 엣지 서버에서 계층 분할 지점을 결정하기 위해 먼저 모바일 단말의 정보가 필요하다. 모바일 단말이 추론하는데 발생하는 추론 지연 시간 정보(inference profile)를 엣지 서버에 미리 전송하는 것으로 가정한다. 엣지 서버는 모바일 단말과 엣지 서버의 추론 지연 시간 정보, 엣지 서버의 처리 큐(processing queue) 정보, 모바일 단말 간 오프로딩하는 연산량에 대한 공평성을 고려하여 적절한 계층 분할 지점을 두 개(실시예에 따라, 단일의 계층 분할 지점을 결정할 수도 있음)로 결정하여 모바일 단말에게 알려준다. 모바일 단말은 결정된 두 개의 계층 분할 지점을 바탕으로 DNN 추론을 시작하게 된다. 엣지 서버의 컴퓨팅 자원은 C_e 로 표현한다. 또한, 엣지 서버는 동시에 여러 태스크를 처리 할 수 없는 것으로 가정한다.
- [0025] 모바일 단말에서 발생하는 DNN 기반의 애플리케이션은 총 L 개의 층으로 구성되어 있고, k ($k \in \mathbb{L}$) 번째 계층을 처리하기 위해 필요한 연산량(Floating point Operations, FLOPs)은 R_k 로 표현하고, k 번째 계층을 처리하고 발생하는 중간 데이터의 크기는 I_k 로 표현한다.
- [0026] 모바일 단말 m 의 첫 번째 계층 분할 지점이 l ($l \in \mathbb{L}$) 로 결정되고 두 번째 계층 분할 지점이 l' ($l' \in \mathbb{L}$) 로 결정되면, DNN 모델을 도 2와 같이 헤드(head), 중간(middle), 테일(tail) 모델로 나눌 수 있다. 이에 따라, 모바일 단말에서 헤드 모델을 처리하고, 엣지 서버에서 중간 모델(또는 미들 모델)을 처리하고, 테일 모델을 다시 모바일 단말에서 처리함으로써 크기가 큰 출력을 엣지 서버에서 모바일 단말로 반환하지 않아도 되는 구조를 가진다.
- [0027] 모바일 단말 m 에서 발생한 DNN 기반의 애플리케이션이 두 개의 계층 분할 지점이 결정되었을 때, DNN 추론 시간 및 과정은 다음과 같다. 먼저, 헤드 모델을 모바일 단말에서 처리하는데 발생하는 시간을 $T_{m,l,l'}^{d_1}$ 로 표현하고 $\sum_{k=0}^l \frac{R_k}{C_m}$ 로 계산한다. 모바일 단말에서 헤드 모델을 처리하고 나온 I_l 크기의 첫 번째 중간 데이터를 엣지 서버로 전송하는데 발생하는 시간을 $T_{m,l,l'}^{t_1}$ 로 표현하고, 통신 상태가 B 라 할 때, $\frac{I_l}{B}$ 로 계산할 수 있다. 또한, 엣지 서버에서 중간 모델을 처리하는데 발생하는 시간을 $T_{m,l,l'}^s$ 로 표현하고 $\sum_{k=l+1}^{l'} \frac{R_k}{C_s}$ 로 계산한다. 엣지 서버에서 중간 모델을 처리하고 나온 $I_{l'}$ 크기의 두 번째 중간 데이터를 모바일 단말로 전송하는데 발생하는 시간을 $T_{m,l,l'}^{t_2}$ 로 표현하고, $\frac{I_{l'}}{B}$ 로 계산한다. 모바일 단말이 $I_{l'}$ 크기의 두 번째 중간 데이터를 엣지 서버로부터 전송 받으면, 테일 모델을 모바일 단말에서 처리하게 되는데, 이 때 발생하는 처리 지연 시간을 $T_{m,l,l'}^{d_2}$ 로 표현하고, $\sum_{k=l'+1}^{L+1} \frac{R_k}{C_m}$ 로 계산할 수 있다.
- [0028] 앞서 계산한 처리 지연 시간 및 전송 지연 시간은 첫 번째 계층 분할 지점 l 과 두 번째 계층 분할 지점 l' 이 같지 않은 경우이다. 반면, 첫 번째 계층 분할 지점과 두 번째 계층 분할 지점이 같은 경우에는 전체 DNN 모델을 모바일 단말에서 자체적으로 처리하겠다는 의미이므로, $T_{m,l,l'}^{t_1}, T_{m,l,l'}^s, T_{m,l,l'}^{t_2}$ 의 값이 0 이 된다.
- [0029] 또한, 엣지 서버의 처리 큐는 다음과 같은 방식으로 업데이트한다. 모바일 단말 m 에서 발생한 DNN 기반의 애플리케이션이 엣지 서버에 도착하는 시점이 $t_{m,l,l'}^a$ 이고, δ 가 괄호안의 수식이 참이면 1, 거짓이면 0을 반환하는 표

시 함수(indicator function)라고 할 때, $t+1$ 시점에 엡지 서버의 처리 큐는 $Q(t+1) = \max(Q(t) + \sum_m (\delta(t = t_{m,l,l'}^a) \cdot T_{m,l,l'}^s) - 1, 0)$ 로 계산한다. 엡지 서버가 이미 처리중인 태스크가 있는 경우에는 엡지 서버에서의 대기 시간 $Q(t_{m,l,l'}^a)$ 가 발생한다.

[0030] 이하에서는 모바일 단말들이 엡지 서버로 오프로딩하는 연산량에 대한 공평성을 보장하고, 모바일 단말들의 평균 DNN 추론 지연 시간을 최소화하기 위한 계층 분할 지점 결정에 대한 최적화 문제를 공식화 한다.

[0031] 1. 목적 함수

[0032] 먼저, $T_{m,l,l'}$ 은 모바일 단말 m 의 두 개의 계층 분할 지점이 $l, l' (l, l' \in \mathbb{L})$ 번째 계층으로 결정되었을 때 발생하는 DNN 추론 지연 시간이다. $T_{m,l,l'}$ 은 앞서 계산한 $T_{m,l,l'}^{d_1}, T_{m,l,l'}^{l_1}, Q(t_{m,l,l'}^a), T_{m,l,l'}^s, T_{m,l,l'}^{l_2}, T_{m,l,l'}^{d_2}$ 의 합으로 계산된다. 또한, $x_{m,l,l'} (x_{m,l,l'} \in \{0,1\})$ 는 모바일 단말 m 의 계층 분할 지점이 l 과 l' 번째 계층이라면 1을, 아니라면 0의 값을 가지게 되는 이진 결정 변수이다. 이를 기반으로 모바일 단말들의 평균 추론 지연 시간 최적화를 위한 목적 함수는 수학적 식 1과 같다.

[0033] [수학적 식 1]

$$\min \frac{\sum_{m=1}^M \sum_{l=0}^L \sum_{l'=0}^L T_{m,l,l'} \cdot x_{m,l,l'}}{M}$$

[0034] [수학적 식 1]

[0035] 2. 제약 조건

[0036] 모든 모바일 단말들에 대해 첫 번째 계층 분할 지점은 최대 1 개가 존재할 수 있으므로, 수학적 식 2의 제약 조건을 만족해야 한다.

[0037] [수학적 식 2]

$$\sum_{l=0}^L x_{m,l,l'} = 1 \quad (\forall m, l')$$

[0038] [수학적 식 2]

[0039] 모든 모바일 단말들에 대해 두 번째 계층 분할 지점은 최대 1 개가 존재할 수 있으므로, 수학적 식 3의 제약 조건을 만족해야 한다.

[0040] [수학적 식 3]

$$\sum_{l'=0}^L x_{m,l,l'} = 1 \quad (\forall m, l)$$

[0041] [수학적 식 3]

[0042] 첫 번째 계층 분할 지점은 두 번째 계층 분할 지점보다 작거나 같아야 하므로, 수학적 식 4의 제약 조건도 함께 만족해야 한다.

[0043] [수학적 식 4]

$$\sum_{l=0}^L x_{m,l,l'} \cdot l \leq \sum_{l'=0}^L x_{m,l,l'} \cdot l'$$

[0044] [수학적 식 4]

[0045] 모바일 단말 m 이 엡지 서버로 오프로딩하는 연산량을 P_m 이라고 할 때, P_m 은 수학적 식 5와 같은 방식으로 계산한다.

[0046] [수학적 식 5]

$$P_m = \sum_{k=l+1}^{l'} R_k$$

[0047] [수학적 식 5]

[0048] Jain's Fairness Index 식(R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System," in *arXiv:cs/9809099*, September

1998.)을 기반으로 최소 보장 되어야 하는 공정성 임계치(fairness threshold) θ_F 를 설정해 주어, 모바일 단말들이 엡지 서버로 오프로딩하는 연산량에 대한 공정성을 보장해 줄 수 있도록 수학적 6을 만족해야 한다.

[수학적 6]

$$\frac{(\sum_{m=1}^M P_m)^2}{M \sum_{m=1}^M P_m^2} \geq \theta_F \quad \left(\frac{1}{M} \leq \theta_F \leq 1 \right)$$

$x_{m,i,i'}$ 는 모바일 단말 m의 첫번째 계층 분할 지점이 1번째이고, 두번째 계층 분할 지점이 1' 번째 계층이라면 1을, 아니라면 0의 값을 가지게 되는 이진 결정 변수이므로, 수학적 7을 만족해야 한다.

[수학적 7]

$$x_{m,i,i'} \in \{0,1\}$$

이하에서는 모바일 단말간 공정성을 보장해주고, 모바일 단말들의 평균 DNN 추론 지연 시간을 최소화하는 최적화 문제를 빠르고 효율적인 시간 내에 해결할 수 있는 휴리스틱 알고리즘을 설명한다.

도 3은 제안하는 휴리스틱 알고리즘의 의사 코드(pseudo code)이다. 먼저, 1번째 줄에서 모바일 단말을 컴퓨팅 자원 기준으로 오름차순으로 정렬하여, 컴퓨팅 자원이 가장 부족한 모바일 단말부터 계층 분할 지점을 결정할 수 있도록 한다. 2 번째 줄에서는 공정성 임계치(fairness threshold) θ_F 값이 주어지면, 다음과 같은 방식으로 K 값을 구한다. 모든 모바일 단말 m에 대하여, $P_{min} \leq P_m \leq P_{max}$ 와 $P_m > 0$ 를 만족할 때, Jain's Fairness Index 는 최소 $\frac{4K}{(K+1)^2}$ ($K = \frac{P_{max}}{P_{min}}$)를 만족한다. 이를 바탕으로 공정성 임계치 θ_F 값이 주어지면 K 값을 계산할 수 있고, 계산한 K 값을 통해 P_m 에 대한 범위를 P_{min} 에 대한 식으로 표현할 수 있다. 변수인 P_{min} 을 특정 상수로 결정해 주기 위해 P_{min} 과 P_{max} 의 중앙값을 나타내는 $\alpha (= \frac{P_{min}+P_{max}}{2} = \frac{(K+1)}{2} P_{min})$ 를 정의하여 15 번째 줄과 같이 α 에 대해 탐색함으로써 공정성 임계치 θ_F 를 만족하는 최종적인 P_m 의 범위를 구한다. 다음으로는, 알고리즘에 필요한 변수를 초기화한다. L_{final} 은 M 개의 모바일 단말에 대해 최종적으로 결정되는 계층 분할 지점을 저장하는 리스트이다. L_{tot} 은 α 값에 따른 M 개의 모바일 단말에 대한 계층 분할 지점을 저장하는 리스트이다. L_{md} 는 하나의 모바일 단말 m이 결정할 수 있는 계층 분할 지점을 저장하는 리스트이다.

4~13 번째 줄에서는 K 값이 1인 경우에 동작하는 의사 코드이다. K 값이 1인 경우는 공정성 임계치 θ_F 값이 1 인 경우로, 모든 모바일 단말들이 동일한 연산량을 엡지 서버로 오프로딩 한다. 오름차순으로 정렬된 모바일 단말 중 첫 번째 모바일 단말이 DNN 추론 지연 시간이 최소화되는 계층 분할 지점을 결정하고, 첫 번째 모바일 단말의 결정된 계층 분할 지점을 나머지 모바일 단말도 동일하게 가질 수 있도록, 11~12 번째 줄과 같이 처리한다.

반면에, K 값이 1이 아닌 경우, 모바일 단말들의 계층 분할 지점 결정 방법은 15~33 번째 줄과 같다. 20 번째 줄은 탐색한 두 개의 계층 분할 지점이 공정성 임계치 θ_F 를 만족하는지 확인하고, 만족한다면 해당 계층 분할 지점을 L_{md} 에 저장하여 모바일 단말 m의 후보 계층 분할 지점으로 저장한다. 하나의 모바일 단말 m에 대한 계층 분할 지점의 탐색이 종료되면, 25 번째 줄과 같이 L_{md} 에 저장된 후보 계층 분할 지점 중 DNN 추론 지연 시간이 최소화되는 계층 분할 지점 (l^*, l'^*)을 기반으로 엡지 서버의 처리 큐를 예약하고, 해당 계층 분할 지점을 L_{mds} 에 저장한다. 엡지 서버의 처리 큐를 예약하는 방식은 도 4, 도 5와 같다. 먼저, 태스크 1과 2가 엡지 서버의 처리 큐에 먼저 예약된 상태에서, 길이가 4인 태스크 3이 시점 6에 도착했을 때, 엡지 서버의 처리 큐에 태스크 3을 연속적으로 처리할 수 있는 공간이 있으면, 도 5와 같이 태스크 3을 예약하고, 그렇지 않으면 도 4와 같이 엡지 서버의 처리 큐의 가장 뒤에 예약하게 된다. 이렇게 하나의 α 값에 대하여 모든 모바일 단말의 계층 분할 지점이 결정되면, L_{mds} 를 L_{tot} 에 저장하고, L_{mds} 를 초기화한다.

33 번째 줄에서는 모든 α 값에 대하여 탐색하여 모바일 단말의 계층 분할 지점이 결정되면, L_{tot} 에 저장된 모바일

단말의 후보 계층 분할 지점 중 DNN 추론 지연 시간의 평균이 가장 최소화되는 모바일 단말들의 계층 분할 지점을 L_{final} 에 저장하여, 모바일 단말들의 최종적인 계층 분할 지점을 결정한다.

[0059] 이하에서는 공식화한 최적화 문제를 Brute-force 방식으로 최적의 해를 찾을 때와 앞서 제안한 휴리스틱 알고리즘을 통해 해를 찾을 때의 알고리즘 복잡도를 설명한다. 먼저, Brute-force 방식으로 최적의 해를 찾을 때, 알고리즘의 복잡도는 $O((\Omega(L)^2)^{\Omega(M)})$ 로 표현한다. $O()$ 는 Big-O 표기법을 나타낸다. 또한, $\Omega(M)$ 과 $\Omega(L)$ 은 각각 집합 M 과 L 에 포함된 원소의 수(모바일 단말의 개수, DNN 모델의 계층 개수)를 나타낸다. 따라서, Brute-force 방식으로 최적의 해를 찾기에는 알고리즘의 복잡도가 높아 많은 연산 시간 및 컴퓨팅 자원이 요구된다. 따라서 본 발명에서는 현실적인 시간 내에 최적의 해를 구할 수 있는 휴리스틱 알고리즘을 제안하였다.

[0060] 제안하는 휴리스틱 알고리즘의 복잡도는 $O(a\Omega(M)\Omega(L)^2)$ 로 표현할 수 있다. 여기서 a 는 공정성 임계치 θ_F 를 만족하는 P_m 의 범위를 구하기 위해, 최적의 α 값을 탐색하기 위한 값으로, 상수의 값을 가진다. 따라서, 지수 함수 형태의 복잡도를 가지는 Brute-force 방식에 비해 제안한 휴리스틱 알고리즘의 복잡도는 다항 함수의 형태를 가지므로 매우 낮은 복잡도를 가지며, 빠른 시간내에 최적의 해를 구할 수 있다.

[0061] 도 6은 본 발명의 일 실시예에 따른 분할 컴퓨팅 방법을 설명하기 위한 흐름도이다. 분할 컴퓨팅 방법은 계층 분할 지점 결정 방법 등으로 명명될 수도 있다. 또한, 분할 컴퓨팅 방법은 적어도 프로세서(processor) 및/또는 메모리(memory)를 포함하는 컴퓨팅 장치에 의해 수행될 수 있다. 따라서, 분할 컴퓨팅 방법의 단계들 중 적어도 일부는 컴퓨팅 장치의 프로세서의 동작으로 이해될 수도 있다. 컴퓨팅 장치는 PC(Personal Computer), 서버(server) 등을 포함한다. 이하에서, 분할 컴퓨팅 방법을 설명함에 있어, 앞선 기재와 중복되는 내용에 관하여는 그 구체적인 설명을 생략하기로 한다.

[0062] 우선, 복수의 모바일 단말들 각각으로부터 계층 분할 지점 결정 요청을 수신한다(S110). 즉, 심층 신경망 모델을 이용한 추론 동작을 수행할 필요가 발생했을 때, 복수의 모바일 단말들 각각은 계층 분할 지점 결정 요청을 서버(분할 컴퓨팅 장치로 명명될 수도 있음)로 송신할 수 있다. 이때, 복수의 모바일 단말들 각각에는 동일한 심층 신경망 모델(학습이 완료된 모델을 의미할 수 있음)이 저장되어 있거나, 또는 상이한 심층 신경망 모델이 저장되어 있을 수 있다.

[0063] 다음으로, 모바일 단말들 각각에 대한 계층 분할 지점을 도출하고, 도출된 계층 분할 지점에 대한 정보를 각 모바일 단말로 전송할 수 있다(S120). 이때, 계층 분할 지점은 모바일 단말 별로 상이하거나 동일할 수 있다. 계층 분할 지점은 앞서 기재한, 목적 함수와 적어도 하나의 제약 조건을 이용한 최적화 문제의 풀이와 제안한 알고리즘을 이용하여 도출될 수 있다.

[0064] 계층 분할 지점에 대한 정보를 수신한 각 모바일 단말들 각각은 수신된 계층 분할 지점에 대응하는 헤드 모델에 대한 추론을 수행하고, 추론 수행의 결과물인 제1 중간 데이터를 서버로 송신할 수 있다(S130). 이때, 각 모바일 단말은 자신의 계층 분할 지점에 맞는 헤드 모델에 대한 추론 동작을 수행한다. 따라서, 각 모바일 단말에는 임의의 계층 분할 지점에 대응하는 모델들이 미리 저장되어 있을 수 있다.

[0065] 제1 중간 데이터를 수신한 서버는 제1 중단 데이터를 중간 모델에 입력함으로써 추론 동작을 수행할 수 있다(S140). 서버에 의한 추론 동작의 결과로써, 제2 중간 데이터가 획득될 수 있고, 제2 중간 데이터는 대응되는 각 모바일 단말로 송신될 수 있다.

[0066] 마지막으로, 각 모바일 단말은 서버로부터 수신한 제2 중간 데이터를 테일 모델에 입력함으로써, 최종 결과를 도출할 수 있다.

[0067] 본 명세서에서 언급된 심층 신경망은 임의의 인공 신경망을 의미할 수 있으며, 본 발명의 권리범위가 인공 신경망의 종류에 제한되는 것이 아님은 명백하다. 하나의 예로써, 심층 신경망은 FSRCNN(Fast Super-Resolution Convolution Neural Network)일 수 있다.

[0068] 이상에서 설명된 장치는 하드웨어 구성 요소, 소프트웨어 구성 요소, 및/또는 하드웨어 구성 요소 및 소프트웨어 구성 요소의 집합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성 요소는, 예를 들어, 프로세서, 컨트롤러, ALU(Arithmetic Logic Unit), 디지털 신호 프로세서(Digital Signal Processor), 마이크로컴퓨터, FPA(Field Programmable 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(Operation System, 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 기술 분야에서 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.

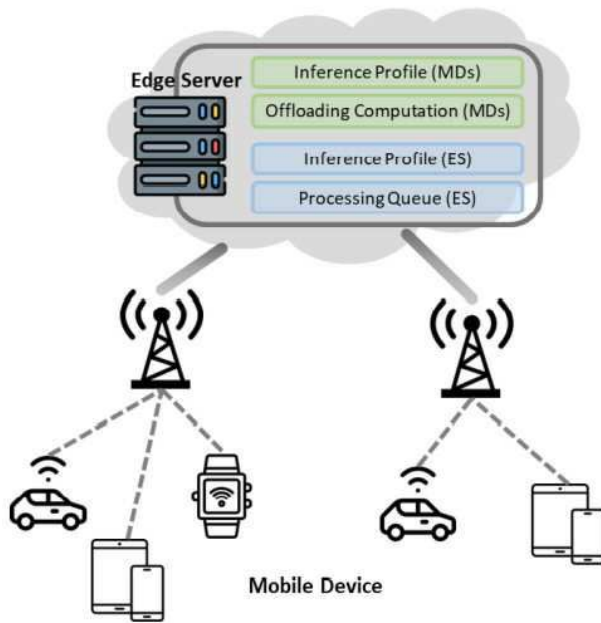
[0069] 소프트웨어는 컴퓨터 프로그램(Computer Program), 코드(Code), 명령(Instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(Collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성 요소(Component), 물리적 장치, 가상 장치(Virtual Equipment), 컴퓨터 저장 매체 또는 장치, 또는 전송되는 신호 파(Signal Wave)에 영구적으로, 또는 일시적으로 구체화(Embody)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.

[0070] 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), 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 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

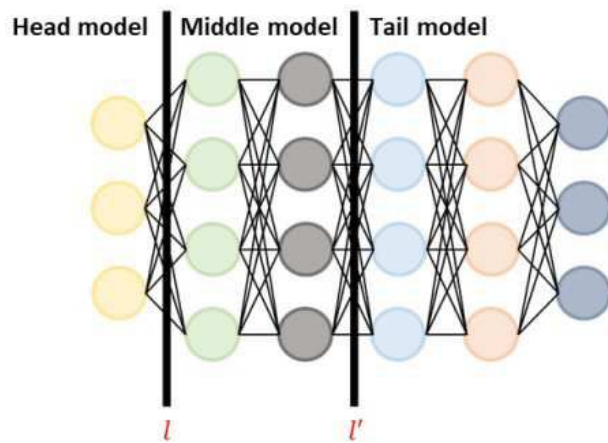
[0071] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성 요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성 요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 등록청구범위의 기술적 사상에 의해 정해져야 할 것이다.

도면

도면1



도면2



도면3

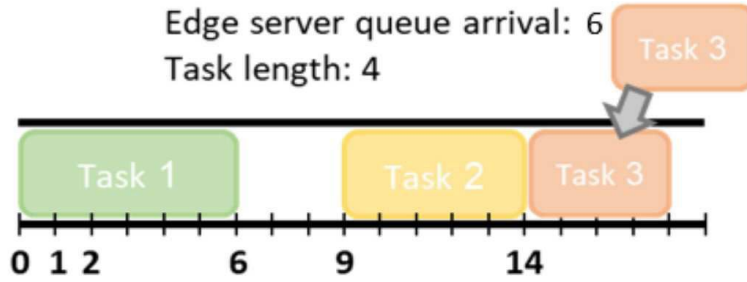
Algorithm 1 Fast and Fair Split Computing (F2SC) Algorithm.

```

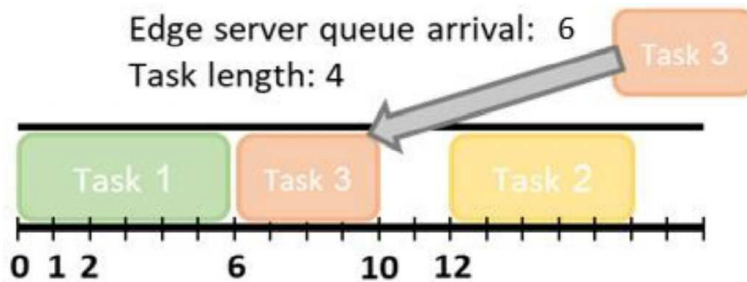
1: Sort the list of mobile devices in ascending order based on computing capability  $C_m$ 
2: Calculate  $K$  with fairness threshold  $\theta_F$ 
3: Initialize the lists  $L_{final}, L_{tot}, L_{mds}, L_{md} = []$ 
4: if  $K = 1$  then
5:   for  $l = 0$  to  $L$  do
6:     for  $l' = 1$  to  $L$  do
7:       Insert  $(l, l')$  into  $L_{md}$ 
8:     end for
9:   end for
10:   $(l^*, l'^*) = \underset{(l, l') \in C}{\operatorname{argmin}} T_{1, l, l'}$ 
11:  for  $m = 1$  to  $M$  do
12:    Insert  $(l^*, l'^*)$  into  $L_{final}$ 
13:  end for
14: else
15:  for  $\alpha = 1$  to  $\frac{K+1}{2K} \sum_{k=0}^L R_k$  do
16:    for  $m = 1$  to  $M$  do
17:      for  $l = 0$  to  $L$  do
18:        for  $l' = 1$  to  $L$  do
19:          Calculate offloaded computations  $P_{(l, l')}$ 
20:          if  $\frac{2\alpha}{K+1} \leq P_{(l, l')} \leq \frac{2\alpha K}{K+1}$  then
21:            Insert  $(l, l')$  into  $L_{md}$ 
22:          end if
23:        end for
24:      end for
25:       $(l^*, l'^*) = \underset{(l, l') \in L_{md}}{\operatorname{argmin}} T_{m, l, l'}$ 
26:      Reserve for edge server's processing queue
27:      Insert  $(l^*, l'^*)$  into  $L_{mds}$ 
28:       $L_{md} = []$ 
29:    end for
30:    Insert  $L_{mds}$  into  $L_{tot}$ 
31:     $L_{mds} = []$ 
32:  end for
33:  Insert the list of mobile devices' split point that minimizes the average DNN inference latency among  $L_{tot}$  into  $L_{final}$ 
34: end if

```

도면4



도면5



도면6

