(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0143059
(43) 공개일자 2024년10월02일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01) *G06F 12/0862* (2016.01)
(52) CPC특허분류
G06F 3/0611 (2013.01)
G06F 12/0862 (2013.01)
(21) 출원번호 10-2023-0037847
(22) 출원일자 2023년03월23일
심사청구일자 2023년03월23일

(71) 출원인
아주대학교산학협력단
경기도 수원시 영통구 월드컵로 206 (원천동)
한국과학기술원
대전광역시 유성구 대학로 291(구성동)
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
안정섭
서울특별시 서초구 청계산로9길 1-12, 서초포레스타6단지 607동 309호
정진우
서울특별시 동대문구 약령시로 154, 미주아파트 2동 712호
(뒷면에 계속)
(74) 대리인
특허법인로얄

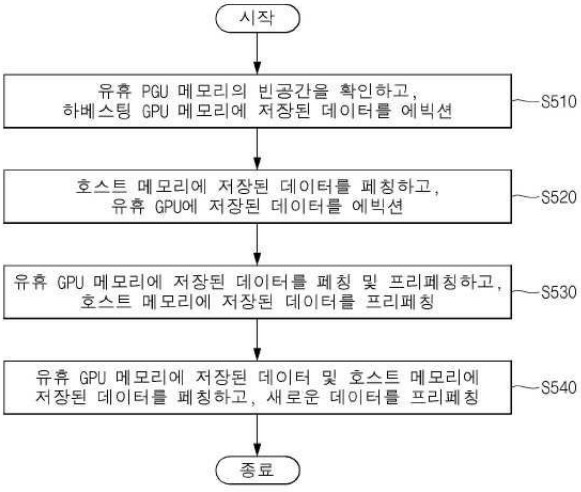
전체 청구항 수 : 총 12 항

(54) 발명의 명칭 **메모리 하베스팅 방법 및 시스템**

(57) 요 약

본 발명은 메모리 하베스팅 방법 및 시스템에 관한 것으로, 본 발명의 일 실시예에 따른 메모리 하베스팅 방법은, 하베스팅 GPU(Graphic processor Unit) 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션(eviction)하는 단계, 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 호스트 메모리로 에빅션하는 단계, 상기 호스트 메모리에 저장된 제2 데이터를 상기 하베스팅 GPU 메모리에 페칭(fetching)하는 단계, 상기 유휴 GPU 메모리에 저장된 제3 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계, 상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 것과 동시에 상기 호스트 메모리에 저장된 제5 데이터를 상기 유휴 GPU 메모리에 프리페칭하는 단계 및 상기 유휴 GPU 메모리에 저장된 제6 데이터 및 상기 호스트 메모리에 저장된 제7 데이터를 동시에 상기 하베스팅 GPU 메모리에 프리페칭하는 단계를 포함할 수 있다.

대 표 도 - 도5



(52) CPC특허분류
G06F 3/0604 (2013.01)
G06F 3/0614 (2013.01)
G06F 3/0658 (2013.01)
G06F 3/0659 (2013.01)
G06F 2212/1024 (2013.01)
G06F 2212/1032 (2013.01)
G06F 2212/6028 (2013.01)

(72) 발명자
권영진
대전광역시 서구 청사로 70, 누리아파트 113동 705호
최상진
대전광역시 서구 둔산중로 20, 나비家 아르누보팰리스 B동 1009호
김택수
서울특별시 송파구 위례성대로18길 28-12, 402호
전명재
울산광역시 울주군 언양읍 유니스트길 50, 울산과학기술원 409동 202호

이 발명을 지원한 국가연구개발사업

과제고유번호	1711152939
과제번호	2021-0-02051-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성
연구과제명	VR/AR 기반의 지능형 라이프컨설턴트 구현
기 여 율	1/1
과제수행기관명	아주대학교산학협력단
연구기간	2022.01.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

하베스팅 GPU(Graphic processor Unit) 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션(eviction)하는 단계;

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 호스트 메모리로 에빅션하는 단계;

상기 호스트 메모리에 저장된 제2 데이터를 상기 하베스팅 GPU 메모리에 페칭(fetching)하는 단계;

상기 유휴 GPU 메모리에 저장된 제3 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계;

상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 단계;

상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 것과 동시에 상기 호스트 메모리에 저장된 제5 데이터를 상기 유휴 GPU 메모리에 프리페칭하는 단계; 및

상기 유휴 GPU 메모리에 저장된 제6 데이터 및 상기 호스트 메모리에 저장된 제7 데이터를 동시에 상기 하베스팅 GPU 메모리에 프리페칭하는 단계를 포함하는, 메모리 하베스팅 방법.

청구항 2

제1항에 있어서,

상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션 하는 단계는,

NVLink를 통해 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함하는, 메모리 하베스팅 방법.

청구항 3

제1항에 있어서,

상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계는,

상기 하베스팅 GPU 메모리의 사용 메모리가 미리 설정한 값 이상인 경우, 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함하는, 메모리 하베스팅 방법.

청구항 4

제1항에 있어서,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는,

PCIe을 통해 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계를 포함하는, 메모리 하베스팅 방법.

청구항 5

제1항에 있어서,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 휴지페이지(hugepage)로 상기 호스트 메모리로 에빅션하는 단계를 포함하는, 메모리 하베스팅 방법.

청구항 6

제1항에 있어서,

상기 유휴 GPU 메모리에 저장된 상기 제6 데이터 및 상기 호스트 메모리에 저장된 상기 제7 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계는,

상기 제6 데이터에 대한 프리페칭 및 상기 제7 데이터에 대한 프리페칭은 동시에 수행되는, 메모리 하베스팅 방법.

청구항 7

적어도 하나의 하베스팅 GPU(Graphic processor Unit) 메모리, 적어도 하나의 유휴 GPU 메모리, 적어도 하나의 호스트 메모리 및 제어 장치를 포함하고,

상기 제어 장치는,

적어도 하나의 명령을 포함하는 메모리; 및

상기 적어도 하나의 명령을 수행하는 프로세서를 포함하고,

상기 적어도 하나의 명령은,

상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션(eviction)하는 단계;

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계;

상기 호스트 메모리에 저장된 제2 데이터를 상기 하베스팅 GPU 메모리에 페칭(fetching)하는 단계;

상기 유휴 GPU 메모리에 저장된 제3 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계;

상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 단계;

상기 호스트 메모리에 저장된 제5 데이터를 상기 유휴 GPU 메모리에 프리페칭하는 단계; 및

상기 유휴 GPU 메모리에 저장된 제6 데이터 및 상기 호스트 메모리에 저장된 제7 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계를 포함하는, 메모리 하베스팅 시스템.

청구항 8

제7항에 있어서,

상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션 하는 단계는,

NVLink를 통해 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함하는, 메모리 하베스팅 시스템.

청구항 9

제7항에 있어서,

상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계는,

상기 하베스팅 GPU 메모리의 사용 메모리가 미리 설정한 값 이상인 경우, 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함하는, 메모리 하베스팅 시스템.

청구항 10

제7항에 있어서,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는,

PCIe을 통해 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계를 포함하는, 메모리 하베스팅 시스템.

청구항 11

제7항에 있어서,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는,

상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 휴지페이지(hugepage)로 상기 호스트 메모리로 에빅션하는 단계를 포함하는, 메모리 하베스팅 시스템.

청구항 12

제7항에 있어서,

상기 유휴 GPU 메모리에 저장된 상기 제6 데이터 및 상기 호스트 메모리에 저장된 상기 제7 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계는,

상기 제6 데이터에 대한 프리페칭 및 상기 제7 데이터에 대한 프리페칭은 동시에 수행되는, 메모리 하베스팅 시스템.

발명의 설명

기 술 분 야

[0001] 본 발명은 메모리 하베스팅 방법 및 시스템에 관한 것으로, 특히 다수의 GPU(graphic processor unit) 메모리에 기반한 메모리 하베스팅 방법 및 시스템에 관한 것이다.

배 경 기 술

[0002] GPU(graphic processor unit)에 대한 수요가 폭발적으로 증가함에 따라 단일 서버에 여러 GPU를 장착하고 공유할 수 있도록 하는 것이 일반적이다. 업계의 많은 기업은 이러한 공유를 통해 인프라 비용을 절감하면서 딥 러닝에서 그래프 애플리케이션에 이르기까지 다양한 워크로드에 대한 수요를 충족하기 위해 다중 GPU 서버 세트로 구성된 대규모 GPU 클러스터를 구축하고 있다.

[0003] 이러한 다중 GPU 서버에서 높은 GPU 리소스 효율성을 달성하는 것은 여전히 어려운 것이 현실이다. 도 1을 참조하면, 종래 메모리 관리 기술은 여러 작업이 GPU에서 독립적으로 실행되는 공유 다중 GPU 환경에 충분히 효과적이지 않음을 알 수 있다. 수백 MB(MegaByte)에서 몇 GB(GigaByte)에 이르는 소량의 메모리가 하나 또는 몇 개의 GPU에서 유휴 상태로 남아 있을 수 있으나, 메모리 압력이 심한 다른 GPU는 동일한 서버 내의 원격 GPU보다 느린 호스트 메모리를 스왑 장치로 사용한다.

[0004] 최신 GPU 서버에는 일반적으로 NVLink와 같은 고속 상호 연결을 통해 연결된 8개 내지 16개의 GPU가 장착되어 있으므로 인접 GPU의 유휴 메모리에 액세스하는 것이 호스트보다 빠를 수 있다. 다만, 상용 다중 GPU 시스템에서 인접 GPU의 유휴 메모리를 활용하지 않는 것이 현실이다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 목적은 다중 GPU 환경(multi-GPU environment)에서 전체적인 메모리 사용률을 개선할 수 있는 메모리 하베스팅 방법 및 시스템을 제공하는 데 있다.

과제의 해결 수단

[0006] 본 발명의 일 실시예에 따른 메모리 하베스팅 방법은, 하베스팅 GPU(Graphic processor Unit) 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션(eviction)하는 단계, 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 호스트 메모리로 에빅션하는 단계, 상기 호스트 메모리에 저장된 제2 데이터를 상기 하베스팅 GPU 메모리에 페칭(fetching)하는 단계, 상기 유휴 GPU 메모리에 저장된 제3 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계, 상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 것과 동시에 상기 호스트 메모리에 저장된 제5 데이터를 상기 유휴 GPU 메모리에 프리페칭하는 단계 및 상기 유휴 GPU 메모리에 저장된 제6 데이터 및 상기 호스트 메모리에 저장된 제7 데이터를 동시에 상기 하베스팅 GPU 메모리에 프리페칭하는 단계를 포함할 수 있다.

[0007] 상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션 하는 단계는, NVLink를 통해 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함할 수 있다.

[0008] 상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계는, 상기 하베스팅

GPU 메모리의 사용 메모리가 미리 설정한 값 이상인 경우, 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함할 수 있다.

[0009] 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는, PCIe을 통해 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계를 포함할 수 있다.

[0010] 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는, 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 휴지페이지(hugepage)를 통해 상기 호스트 메모리로 에빅션하는 단계를 포함할 수 있다.

[0011] 상기 유휴 GPU 메모리에 저장된 상기 제6 데이터 및 상기 호스트 메모리에 저장된 상기 제7 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계는, 상기 제6 데이터에 대한 프리페칭 및 상기 제7 데이터에 대한 프리페칭은 동시에 수행될 수 있다.

[0012] 본 발명의 일 실시예에 따른 메모리 하베스팅 시스템은 적어도 하나의 하베스팅 GPU(Graphic processor Unit) 메모리, 적어도 하나의 유휴 GPU 메모리, 적어도 하나의 호스트 메모리 및 제어 장치를 포함하고, 상기 제어 장치는, 적어도 하나의 명령을 포함하는 메모리 및 상기 적어도 하나의 명령을 수행하는 프로세서를 포함하고, 상기 적어도 하나의 명령은, 상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션(eviction)하는 단계, 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계, 상기 호스트 메모리에 저장된 제2 데이터를 상기 하베스팅 GPU 메모리에 페칭(fetching)하는 단계, 상기 유휴 GPU 메모리에 저장된 제3 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계, 상기 유휴 GPU 메모리에 저장된 제4 데이터를 상기 하베스팅 GPU 메모리에 프리페칭(prefetching)하는 것과 동시에 상기 호스트 메모리에 저장된 제5 데이터를 상기 유휴 GPU 메모리에 프리페칭하는 단계 및 상기 유휴 GPU 메모리에 저장된 제6 데이터 및 상기 호스트 메모리에 저장된 제7 데이터를 동시에 상기 하베스팅 GPU 메모리에 프리페칭하는 단계를 포함할 수 있다.

[0013] 상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 유휴 GPU 메모리로 에빅션 하는 단계는, NVLink를 통해 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함할 수 있다.

[0014] 상기 하베스팅 GPU 메모리에 저장된 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계는, 상기 하베스팅 GPU 메모리의 사용 메모리가 미리 설정한 값 이상인 경우, 상기 하베스팅 GPU 메모리에 저장된 상기 제1 데이터를 상기 유휴 GPU 메모리로 에빅션 하는 단계를 포함할 수 있다.

[0015] 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는, PCIe을 통해 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계를 포함할 수 있다.

[0016] 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 상기 호스트 메모리로 에빅션하는 단계는, 상기 유휴 GPU 메모리에 에빅션된 상기 제1 데이터를 휴지페이지(hugepage)를 통해 상기 호스트 메모리로 에빅션하는 단계를 포함할 수 있다.

[0017] 상기 유휴 GPU 메모리에 저장된 상기 제6 데이터 및 상기 호스트 메모리에 저장된 상기 제7 데이터를 상기 하베스팅 GPU 메모리에 페칭하는 단계는, 상기 제6 데이터에 대한 프리페칭 및 상기 제7 데이터에 대한 프리페칭은 동시에 수행될 수 있다.

발명의 효과

[0018] 본 발명에 따르면, 호스트 메모리에 액세스하는 지연 시간을 효과적 감소시킬 수 있고, 이에 따라 메모리 집약적인 워크로드의 경우 GPU 메모리의 처리량이 향상될 수 있다.

도면의 간단한 설명

[0019] 본 발명에 대해 더욱 이해하기 위해 포함되며 본 출원에 포함되고 그 일부를 구성하는 첨부된 도면은 본 발명의 원리를 설명하는 상세한 설명과 함께 본 발명의 실시예를 나타낸다.

도 1은 종래 메모리 관리 방법을 설명하기 위한 개념도이다.

도 2는 본 발명의 일 실시예에 따른 메모리 하베스팅 시스템의 블록도이다.

도 3 내지 4는 본 발명의 일 실시예에 따른 메모리 하베스팅 방법을 설명하기 위한 개념도이다.

도 5는 본 발명의 일 실시예에 따른 메모리 하베스팅 방법의 흐름도이다.

도 6은 본 발명의 일 실시예에 따른 하베스팅 방법의 효과를 설명하기 위한 개념도이다.

도 7은 본 발명의 다른 실시예에 따른 제어 장치의 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0020] 본 명세서의 상술한 목적, 특징들 및 장점은 첨부된 도면과 관련된 다음의 상세한 설명을 통해 보다 분명해질 것이다. 다만, 본 명세서는 다양한 변경을 가할 수 있고 여러 가지 실시 예들을 가질 수 있는 바, 이하에서는 특정 실시 예들을 도면에 예시하고 이를 상세히 설명하고자 한다. 명세서 전체에 걸쳐서 동일한 참조번호들은 원칙적으로 동일한 구성요소들을 나타낸다. 또한, 본 명세서와 관련된 공지 기능 혹은 구성에 대한 구체적인 설명이 본 명세서의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다.

[0021] 이하, 본 명세서와 관련된 방법 및 장치에 대하여 도면을 참조하여 보다 상세하게 설명한다. 이하의 설명에서 사용되는 구성요소에 대한 접미사 "모듈" 및 "부"는 명세서 작성의 용이함만이 고려되어 부여되거나 혼용되는 것으로서, 그 자체로 서로 구별되는 의미 또는 역할을 갖는 것은 아니다.

[0022] 도 2는 본 발명의 일 실시예에 따른 메모리 하베스팅 시스템의 블록도이다. 도 3 내지 4는 본 발명의 일 실시예에 따른 메모리 하베스팅 방법을 설명하기 위한 개념도이다.

[0023] 도 2를 참조하면, 본 발명의 일 실시예에 따른 메모리 하베스팅 시스템(10)은 메모리 장치(100) 및 제어 장치(200)를 포함할 수 있다.

[0024] 메모리 장치(100) 및 제어 장치(200)는 상호간에 연결되어 통신을 수행할 있다. 메모리 장치(100)는 호스트 메모리(110), 하베스트 GPU(graphic processor unit) 메모리(120) 및 유휴 GPU 메모리(130)를 포함할 수 있다. 여기에서, 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)는 동일하게 구성될 수 있다. 또한, 각각 하나의 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)를 도시하였으나 이는 일 예시이며 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)는 복수일 수 있다.

[0025] 호스트 메모리(110)는 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)와 상호간에 PCIe을 통해 연결될 수 있고 데이터를 송수신할 수 있다. 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)는 상호간에 NVLink를 통해 연결될 수 있고 데이터를 송수신할 수 있다.

[0026] 제어 장치(200)는 메모리 장치(100)를 제어할 수 있다. 제어 장치(200)는 호스트 메모리(110), 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130)의 사용 메모리 및 유휴 메모리에 대한 정보를 획득할 수 있다. 여기에서, 사용 메모리는 현재 데이터가 저장된 메모리일 수 있고, 유휴 메모리는 데이터가 저장되지 않은 메모리일 수 있다. 제어 장치(200)는 사용 메모리 및 유휴 메모리에 대한 정보를 기초로 호스트 메모리(110), 하베스트 GPU 메모리(120) 및 유휴 GPU 메모리(130) 사이에 데이터를 송수신할 것을 명령할 수 있다.

[0027] 제어 장치(200)는 하베스팅 GPU 메모리(120)에 저장된 데이터를 유휴 GPU 메모리(130)로 에빅션(eviction)할 수 있다. 제어 장치(200)는 하베스팅 GPU 메모리(120)의 사용 메모리의 비율이 미리 설정한 값 이상인 경우, 하베스팅 GPU 메모리(120)에 저장된 데이터를 유휴 GPU 메모리(130)로 에빅션할 수 있다. 예를 들어, 미리 설정한 값은 90%일 수 있다.

[0028] 제어 장치(200)는 호스트 메모리(110)에 저장된 데이터 가운데 적어도 하나 이상을 하베스팅 GPU 메모리(120)에 페칭할 수 있다. 이와 달리 제어 장치(200)는 호스트 메모리(110)에 저장된 데이터 가운데 적어도 하나 이상을 하베스팅 GPU 메모리(120) 및 유휴 GPU 메모리(130)에 병렬적으로 페칭할 수 있다. 또한, 제어 장치(200)는 유휴 GPU 메모리(130)로 에빅션된 데이터를 호스트 메모리(110)로 에빅션할 수 있다. 이 경우, 제어 장치(200)는 데이터를 별도로 분할하지 않고 데이터를 에빅션할 수 있다(Large page eviction). 한편, 유휴 GPU 메모리(130)가 복수인 경우 제어 장치(200)는 라운드 로빈(round robin) 방식으로 데이터를 에빅션할 수 있다.

[0029] 도 3(a)을 참조하면, 하베스팅 GPU 메모리(120)에는 데이터 X 및 데이터 Y가 저장되어 있고, 유휴 GPU 메모리(130)에는 데이터 E가 저장되어 있고, 호스트 메모리(110)에는 데이터 A, 데이터 B, 데이터 C 및 데이터 D가 저장된 경우에, 제어 장치(200)는 하베스팅 GPU 메모리(120)에 저장된 데이터 X 및 데이터 Y를 유휴 GPU 메모리(130)로 에빅션할 수 있다. 이후, 제어 장치(200)는 유휴 GPU 메모리(130)로 에빅션된 데이터 X 및 데이터 Y를 호스트 메모리(110)로 에빅션할 수 있다. 이 경우, 제어 장치(200)는 데이터 X 및 데이터 Y를 휴지페이지(hugepage)로 에빅션할 수 있다. 제어 장치(200)는 호스트 메모리(110)로부터 하베스팅 메모리로 데이터 A 및

데이터 B를 페칭할 수 있다. 이 경우, 하베스팅 GPU 메모리(120)에는 데이터 A 및 데이터 B가 저장될 수 있고, 유휴 GPU 메모리(130)에는 데이터 E 데이터 X 및 데이터 Y가 저장될 수 있고, 호스트 메모리(110)에는 데이터 X, 데이터 Y, 데이터 C 및 데이터 D가 저장될 수 있다.

[0030] 도 3(b)을 참조하면, 하베스팅 GPU 메모리(120)에는 데이터 X 및 데이터 Y가 저장되어 있고, 유휴 GPU 메모리(130)에는 데이터 E가 저장되어 있고, 호스트 메모리(110)에는 데이터 A, 데이터 B, 데이터 C 및 데이터 D가 저장된 경우에, 제어 장치(200)은 호스트 메모리(110)에 저장된 데이터 A를 하베스팅 GPU 메모리(120)으로 패치하고 병렬적으로 호스트 메모리(110)에 저장된 데이터 B를 유휴 GPU 메모리(130)로 패치를 할 수 있다.

[0031] 이 경우, 하베스팅 GPU 메모리(120)에는 데이터 A가 저장될 수 있고, 유휴 GPU 메모리(130)에는 데이터 E 데이터 X, 데이터 Y 및 데이터 B가 저장될 수 있고, 호스트 메모리(110)에는 데이터 X, 데이터 Y, 데이터 C 및 데이터 D가 저장될 수 있다.

[0032] 다시 도 2를 참조하면, 제어 장치(200)는 호스트 메모리(110)에 저장된 데이터를 하베스팅 GPU 메모리(120) 및 유휴 GPU 메모리(130)에 페칭할 수 있다. 따라서, 제어 장치(200)는 페이지 폴트(page fault) 처리 비용을 줄일 수 있다.

[0033] 제어 장치(200)는 유휴 GPU 메모리(130)에 저장된 데이터를 유휴 GPU 메모리(130)로부터 하베스팅 GPU 메모리(120)로 페칭할 수 있다. 제어 장치(200)는 호스트 메모리(110)에 저장된 데이터를 호스트 메모리(110)로부터 유휴 GPU 메모리(130)로 프리페칭(Prefetching)할 수 있다. 이때 호스트 메모리(110)에 저장된 데이터를 유휴 GPU 메모리(130)로 프리페칭함과 동시에 유휴 GPU 메모리(130)에 저장된 데이터를 하베스팅 GPU(120)로 병렬적으로 프리페칭할 수 있다.

[0034] 제어 장치(200)은 유휴 GPU 메모리(130)에 저장된 데이터를 유휴 GPU 메모리(130)으로부터 하베스팅 GPU 메모리(120)으로 페칭함과 동시에 병렬적으로 호스트 메모리(110)에 저장된 데이터를 하베스팅 GPU 메모리(120)으로 페칭할 수 있다.

[0035] 도 4(a)을 참조하면, 하베스팅 GPU 메모리(120)에는 데이터 A가 저장되어 있고, 유휴 GPU 메모리(130)에는 데이터 E, 데이터 X, 데이터 Y 및 데이터 B가 저장되어 있고, 호스트 메모리(110)에는 데이터 X, 데이터 Y, 데이터 C 및 데이터 D가 저장된 경우에, 제어 장치(200)는 GPU에 저장된 데이터 B를 하베스팅 GPU 메모리(120)로 페칭할 수 있다. 제어 장치(200)는 GPU에 저장된 데이터 E를 하베스팅 GPU 메모리(120)로 프리페칭할 수 있다. 제어 장치(200)는 호스트 메모리(110)에 저장된 데이터 C 및 데이터 D를 유휴 GPU 메모리(130)에 프리페칭할 수 잇다.

[0036] 도 4(b)를 참조하면, 이러한 방식의 프리패칭의 이점은 도 4(b)를 통해 알 수 있다. 제어장치(200)이 데이터 C 및 데이터 D를 호스트 메모리(110)에서 유휴 GPU 메모리(130)으로 미리 프리패칭을 하였기 때문에 추후 페이지 C와 페이지 D가 하베스팅 GPU에서 폴트가 날 경우 폴트 비용을 효과적으로 낮출 수 있다.

[0037] 도 5는 본 발명의 일 실시예에 따른 메모리 하베스팅 방법의 흐름도이다.

[0038] 도 5를 참조하면, 제어 장치는 유휴 GPU 메모리의 빈공간을 판단하고 하베스팅 GPU 메모리에 저장된 데이터를 에빅션할 수 있다(S510). 제어 장치(예를 들어, 도 2의 제어 장치(200))는 유휴 GPU 메모리(예를 들어, 도 2의 유휴 GPU 메모리(130))에 빈공간에 대한 정보를 획득할 수 있다. 빈공간이 미리 설정한 값 이상인 경우, 제어 장치는 하베스팅 GPU 메모리(예를 들어, 도 2의 하베스팅 GPU 메모리(120))에 저장된 데이터들 가운데 적어도 하나를 유휴 GPU 메모리(예를 들어, 도 1의 유휴 GPU 메모리(130))의 빈공간에 에빅션할 수 있다. 제어 장치는 하베스팅 GPU 메모리의 사용 메모리 비율이 미리 설정한 값 이상인 경우 하베스팅 GPU 메모리에 저장된 데이터를 에빅션할 수 있다. 제어 장치는 NVLink를 통해 하베스팅 GPU 메모리에 저장된 데이터를 에빅션할 수 있다.

[0039] 제어 장치는 유휴 GPU 메모리에 저장된 데이터를 에빅션할 수 있고, 호스트 메모리에 저장된 데이터를 페칭할 수 있다(S520). 제어 장치는 S510에서 유휴 GPU 메모리로 에빅션된 데이터를 호스트 메모리(예를 들어, 도 1의 호스트 메모리(110))로 에빅션할 수 있다. 제어 장치는 유휴 GPU 메모리로 에빅션된 데이터에 대한 휴지페이지 에빅션을 수행할 수 있다. 제어 장치는 PCIe을 통해 유휴 GPU 메모리로 에빅션된 데이터를 호스트 메모리로 에빅션할 수 있다. 제어 장치는 호스트 메모리에 저장된 데이터 가운데 적어도 하나를 하베스팅 GPU 메모리에 페칭할 수 있다. 일 실시예에서, 제어 장치는 호스트 메모리에 저장된 데이터를 하베스팅 GPU 메모리 및 유휴 GPU 메모리에 병렬적으로 페칭할 수 있다.

[0040] 제어 장치는 유휴 GPU 메모리에 저장된 데이터를 페칭 및 프리페칭할 수 있고, 호스트 메모리에 저장된 데이터

를 프리페칭할 수 있다(S530). 제어 장치는 유휴 GPU 메모리에 저장된 데이터 가운데 적어도 하나를 하베스팅 GPU 메모리에 페칭할 수 있고, 유휴 GPU 메모리에 저장된 데이터 가운데 적어도 하나를 하베스팅 GPU 메모리에 프리패칭할 수 있다. 여기에서, 페칭되는 데이터 및 프리페칭되는 데이터는 다를 수 있다. 제어 장치는 호스트 메모리에 저장된 데이터 가운데 두개 이상의 데이터를 유휴 GPU 메모리에 프리페칭할 수 있다.

[0041] 제어 장치는 유휴 GPU 메모리에 저장된 데이터 및 호스트 메모리에 저장된 데이터를 페칭할 수 있고, 새로운 데이터를 프리페칭할 수 있다(S540). 제어 장치는 유휴 GPU 메모리에 저장된 데이터를 하베스팅 GPU 메모리로 페칭할 수 있고, 호스트 GPU 메모리에 저장된 데이터를 하베스팅 GPU 메모리로 페칭할 수 있다. 여기에서, 유휴 GPU 메모리에 저장된 데이터 및 호스트 GPU 메모리에 저장된 데이터는 동일할 수 있다. 또한, 제어 장치는 새로운 데이터를 유휴 GPU 메모리에 프리패칭할 수 있다.

[0042] 도 6은 본 발명의 일 실시예에 따른 하베스팅 방법의 효과를 설명하기 위한 개념도이다.

[0043] 도 6은 메모리 하베스팅 방법으로 제1 방법(base), 제2 방법(Pre-ef-host) 및 제3 방법(HUVM)에 따른 베이스 라인에서의 속도를 비교한 그래프이다. 도 6에서 제3 방법(HUVM)은 본 발명의 일 실시예에 따른 메모리 하베스팅 방법일 수 있다.

[0044] Case-1은 하베스팅 GPU 메모리(H)가 1개이고 유휴 GPU 메모리(Y)가 3개인 경우이며, 알고리즘으로 Pagerank, VGG16 및 WCC를 사용한 경우의 제1 방법 내지 제3 방법에 따른 베이스 라인에서의 속도를 비교한 그래프이다. Case-1을 참조하면, 알고리즘으로 VGG16 또는 WCC을 사용한 경우에 제1 방법 내지 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 베이스 라인에서의 속도의 차이가 없으나, 알고리즘으로 pagerank를 사용하는 경우, 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 제1 방법 및 제2 방법을 사용하여 메모리 하베스팅을 수행하는 경우에 비해 베이스 라인에서의 속도가 증가하는 것을 확인할 수 있다.

[0045] Case-2는 하베스팅 GPU 메모리(H)가 2개이고 유휴 GPU 메모리(Y)가 2개인 경우이며, 알고리즘으로 BFs, MobileNet 및 Resnet101을 사용한 경우의 제1 방법 내지 제3 방법에 따른 베이스 라인에서의 속도를 비교한 그래프이다. Case-2를 참조하면, 알고리즘으로 MobileNet 또는 Resnet101을 사용한 경우에 제1 방법 내지 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 베이스 라인에서의 속도의 차이가 없으나, 알고리즘으로 BFs를 사용하는 경우, 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 제1 방법 및 제2 방법을 사용하여 메모리 하베스팅을 수행하는 경우에 비해 베이스 라인에서의 속도가 증가하는 것을 확인할 수 있다.

[0046] Case-3은 하베스팅 GPU 메모리(H)가 2개이고 유휴 GPU 메모리(Y)가 2개인 경우이며, 알고리즘으로 WCC, BFs 및 pagerank를 사용한 경우의 제1 방법 내지 제3 방법에 따른 베이스 라인에서의 속도를 비교한 그래프이다. Case-3을 참조하면, 알고리즘으로 pagerank를 사용한 경우에 제1 방법 내지 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 베이스 라인에서의 속도의 차이가 없으나, 알고리즘으로 WCC 또는 BFS를 사용하는 경우, 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 제1 방법 및 제2 방법을 사용하여 메모리 하베스팅을 수행하는 경우에 비해 베이스 라인에서의 속도가 증가하는 것을 확인할 수 있다. case-4는 하베스팅 GPU 메모리(H)가 3개이고 유휴 GPU 메모리(Y)가 1개인 경우이며, 알고리즘으로 Wcc, Louvaion 및 Resnet101를 사용한 경우의 제1 방법 내지 제3 방법에 따른 베이스 라인에서의 속도를 비교한 그래프이다. Case-1을 참조하면, 알고리즘으로 Resnet을 사용한 경우에 제1 방법 내지 제3 방법에 따라 메모리 하베스팅을 수행하는 경우 베이스 라인에서의 속도의 차이가 없으나, 알고리즘으로 WCC 또는 Louvain를 사용하는 경우, 제3 방법에 따라 하베스팅을 수행하는 경우 제1 방법 및 제2 방법을 사용하여 메모리 하베스팅을 수행하는 경우에 비해 베이스 라인에서의 속도가 증가하는 것을 확인할 수 있다.

[0047] 도 7은 본 발명의 다른 실시예에 따른 제어 장치의 블록도이다.

[0048] 도 7의 제어 장치(700)는 도 1의 제어 장치(200)와 동일할 수 있다.

[0049] 제어 장치(700)는 적어도 하나의 프로세서(710), 메모리(720) 및 네트워크와 연결되어 통신을 수행하는 송수신 장치(730)를 포함할 수 있다. 또한, 제어 장치(700)는 입력 인터페이스 장치(740), 출력 인터페이스 장치(750), 저장 장치(760) 등을 더 포함할 수 있다. 제어 장치(700)에 포함된 각각의 구성 요소들은 버스(bus)(770)에 의해 연결되어 서로 통신을 수행할 수 있다. 다만, 제어 장치(700)에 포함된 각각의 구성요소들은 공통 버스(770)가 아니라, 프로세서(710)를 중심으로 개별 인터페이스 또는 개별 버스를 통하여 연결될 수도 있다. 예를 들어, 프로세서(710)는 메모리(720), 송수신 장치(730), 입력 인터페이스 장치(740), 출력 인터페이스 장치(750) 및 저장 장치(760) 중에서 적어도 하나와 전용 인터페이스를 통하여 연결될 수도 있다.

[0050] 프로세서(710)는 메모리(720) 및 저장 장치(760) 중에서 적어도 하나에 저장된 프로그램 명령(program comman

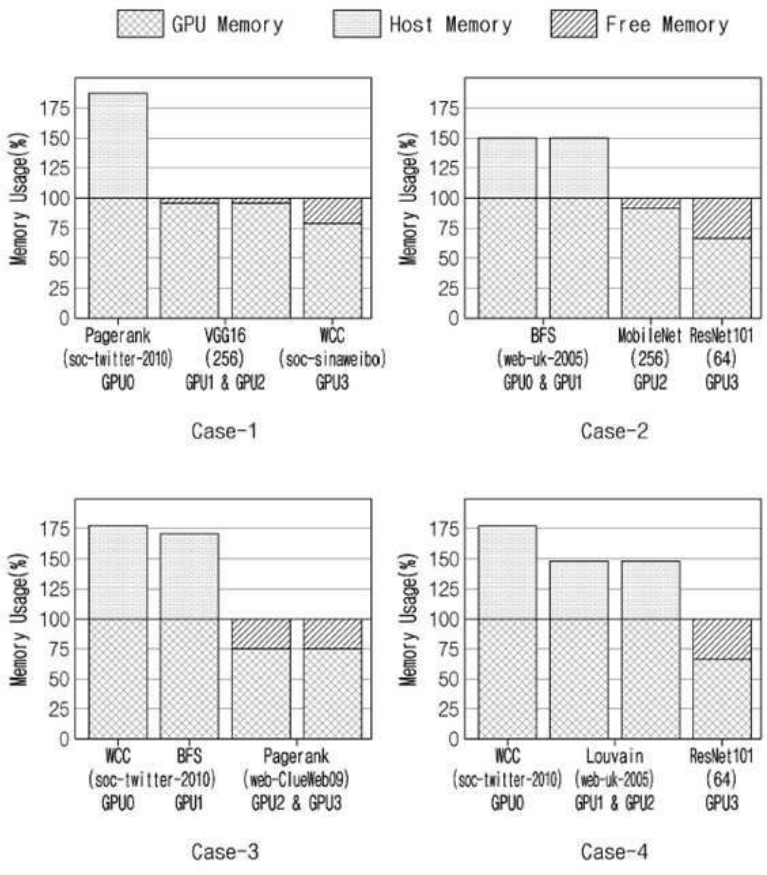
d)을 실행할 수 있다. 프로세서(710)는 중앙 처리 장치(central processing unit, CPU), 그래픽 처리 장치(graphics processing unit, GPU), 또는 본 발명의 실시예들에 따른 방법들이 수행되는 전용의 프로세서를 의미할 수 있다. 메모리(720) 및 저장 장치(760) 각각은 휘발성 저장 매체 및 비휘발성 저장 매체 중에서 적어도 하나로 구성될 수 있다. 예를 들어, 메모리(720)는 읽기 전용 메모리(read only memory, ROM) 및 랜덤 액세스 메모리(random access memory, RAM) 중에서 적어도 하나로 구성될 수 있다.

[0051] 본 발명에서 사용되는 대부분의 용어는 해당 분야에서 널리 사용되는 일반적인 것들에서 선택되지만, 일부 용어는 출원인에 의해 임의로 선택되며 그 의미는 필요에 따라 다음 설명에서 자세히 서술한다. 따라서 본 발명은 용어의 단순한 명칭이나 의미가 아닌 용어의 의도된 의미에 근거하여 이해되어야 한다.

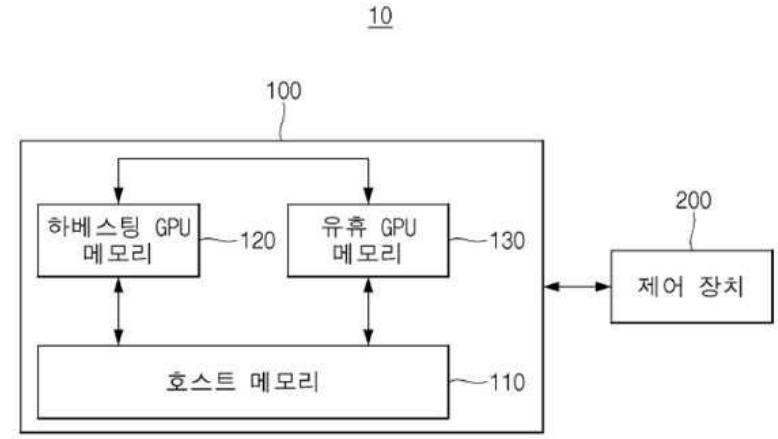
[0052] 본 발명은 본 발명의 필수적 특징을 벗어나지 않는 범위에서 다른 특정한 형태로 구체화될 수 있음은 당업자에게 자명하다. 따라서, 상술한 상세한 설명은 모든 면에서 제한적으로 해석되어서는 아니 되고 예시적인 것으로 고려되어야 한다. 본 발명의 범위는 첨부된 청구항의 합리적 해석에 의해 결정되어야 하고, 본 발명의 등가적 범위 내에서의 모든 변경은 본 발명의 범위에 포함된다.

도면

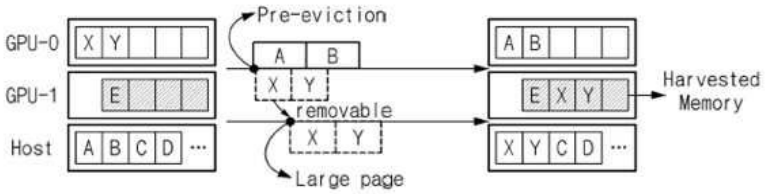
도면1



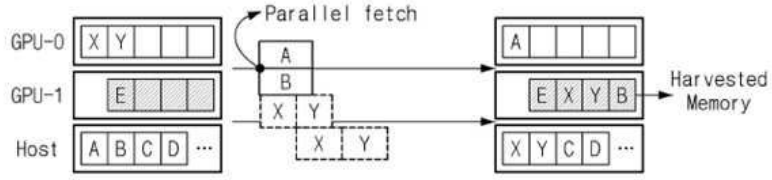
도면2



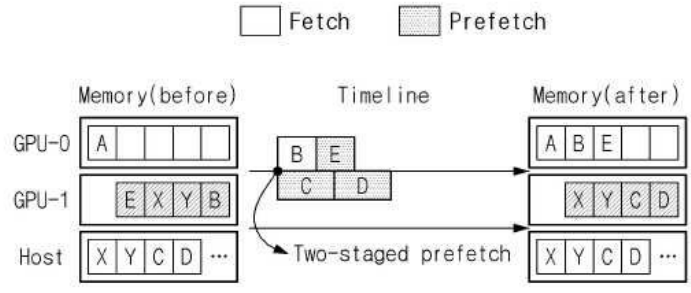
도면3a



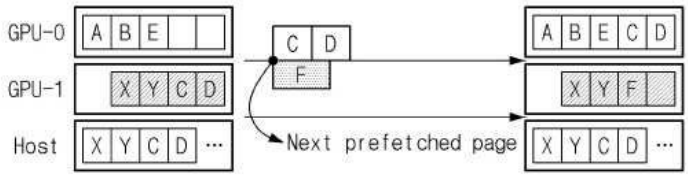
도면3b



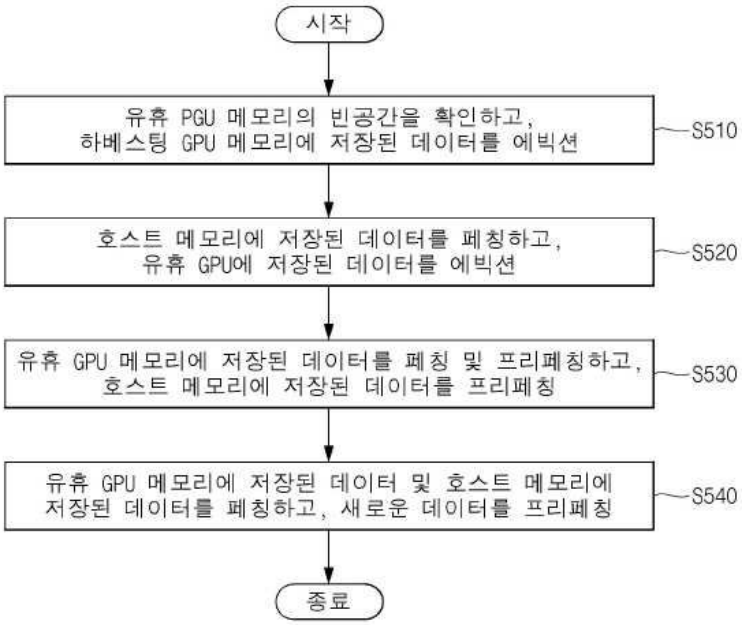
도면4a



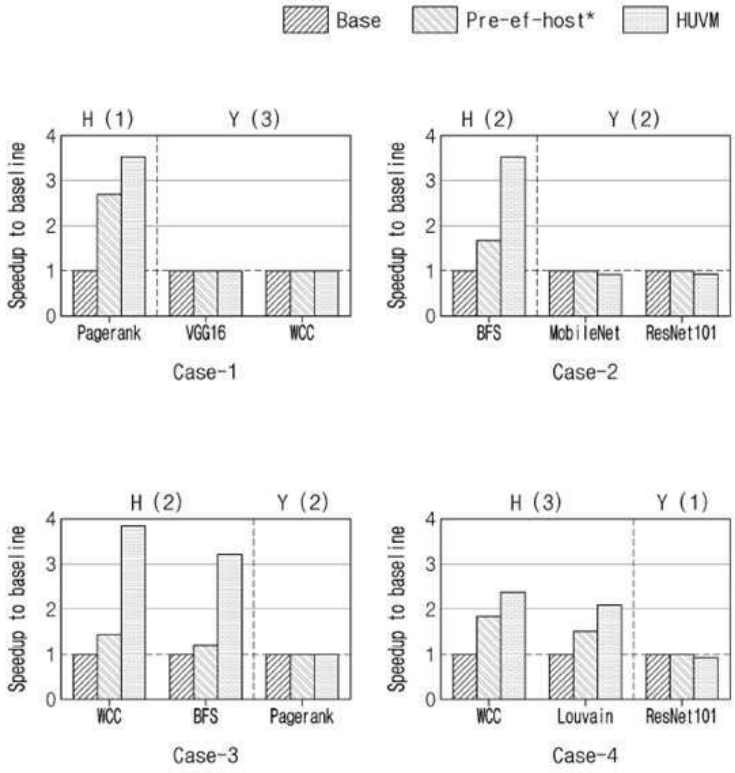
도면4b


GPU-0
A B E
GPU-1
X Y C D
Host
X Y C D ···
C D
F
Next prefetched page
A B E C D
X Y F
X Y C D ···

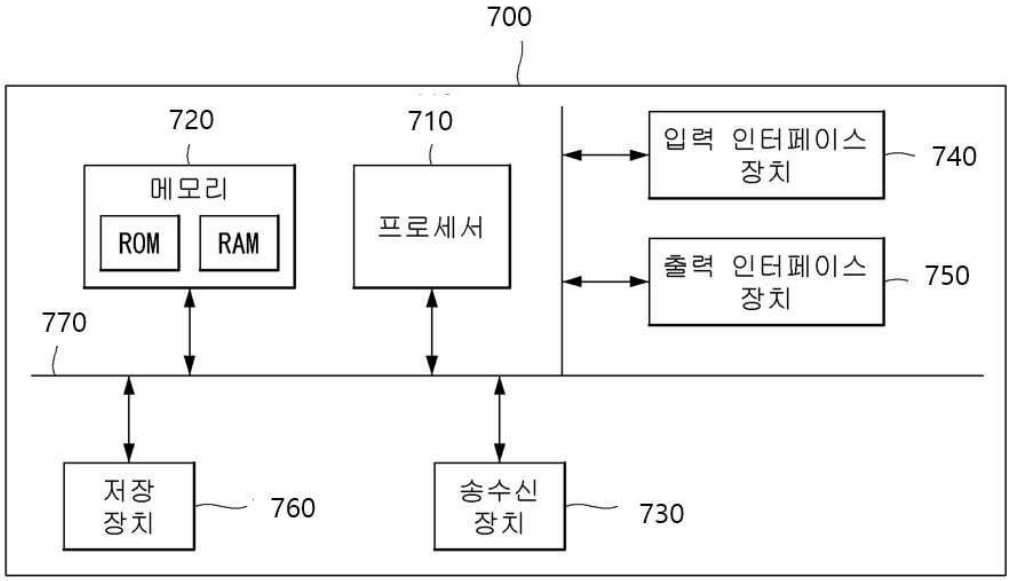
도면5


시작
유휴 PGU 메모리의 빈공간을 확인하고,
하베스팅 GPU 메모리에 저장된 데이터를 에빅션
S510
호스트 메모리에 저장된 데이터를 페칭하고,
유휴 GPU에 저장된 데이터를 에빅션
S520
유휴 GPU 메모리에 저장된 데이터를 페칭 및 프리페칭하고,
호스트 메모리에 저장된 데이터를 프리페칭
S530
유휴 GPU 메모리에 저장된 데이터 및 호스트 메모리에
저장된 데이터를 페칭하고, 새로운 데이터를 프리페칭
S540
종료

도면6


Base
Pre-ef-host*
HUVM
H (1)
Y (3)
Speedup to baseline
Pagerank
VGG16
WCC
Case-1
H (2)
Y (2)
BFS
MobileNet
ResNet101
Case-2
H (2)
Y (2)
WCC
BFS
Pagerank
Case-3
H (3)
Y (1)
WCC
Louvain
ResNet101
Case-4

도면7


700
720
메모리
ROM
RAM
710
프로세서
입력 인터페이스 장치
740
출력 인터페이스 장치
750
770
저장 장치
760
송수신 장치
730