반도체공학회 논문지 (TSE: Transactions on Semiconductor Engineering), Vol. 2 , No. 4, Dec., 2024
https://doi.org/10.22895/tse.2024.0016 ISSN (On-line) 3022-2028

계층적 정렬 기반 실시간 가우시안 스플래팅 렌더링 FPGA 프로세서

(A Hierarchical Sorting based Real-Time Gaussian Splatting Rendering FPGA Pro-cessor)

이홍석 [1], 박원훈 [1], 안상혁 [1], 김민성 [1], 유회준 [1,+]
(Hongseok Lee [1], Wonhoon Park [1], Sanghyuk Ahn[1], Minsung Kim[1], and Hoi-Jun Yoo[1,+])

요 약

3D 가우시안 스플래팅 렌더링은 AR/VR 등의 3D 환경 렌더링 기술쪽의 SOTA 이다. 빠른 렌더링과 적은 훈련 데이터로도 고품질의 이미지를 렌더링 하지만, 엣지 디바이스에서 3D 가우시안 렌더링을 달성하는건 각 연산 과정에서 독립적으로 생기는 문제로 인해 제한이 있다. 본 논문에서는 계층적 정렬 연산, 재구성 가능한 연산코어와 중요도 기반 구면조화 함수 변환을 활용해 엣지(FPGA)에서 빠르고 고품질의 렌더링을 달성했다.

ABSTRACT

3D Gaussian Splatting Rendering is a SOTA on 3D environment rendering technology regarding AR/VR. It is able to render high quality image with less training data then previous methods, while maintaining fast rendering. Yet, achieving this on edge device is limited due to independently occurring images in each processing stages. This paper proposes hierarchical sorting computation, reconfigurable matmul core and importance based spherical harmonics band evaluation to achieve fast and high quality rendering in FPGA.

KEY WORDS
키워드; 3D 가우시안 스플래팅, 고품질 렌더링, 3D 환경 모델링

I. 서 론

최근 3D 가우시안 스플래팅(3DGS)은 3D 환경을 렌더링함에 있어 새로운 방법으로 주목받고 있다[1]. 기존의 Neural Radiance Field (NeRF) 와는 다르게, 3DGS는 비등방성 공분산 매개변수를 활용해 적은 훈련 데이터로도 고품질의 투명 및 반사장면을 렌더링 할 수 있다[2]. 렌더링의 주요 단계는 다음과 같다. 1) Frustum Culling – 보이는 범위 밖의 가우시안 제거 2) 2D Feature Computation – 3D 장면 매개변수에서 이미지 매개변수 계산 3) 가우시안을 깊이 순으로 정렬 4) Volume Rendering(VR) – 정렬된 가우시안의 가중 합산으로 픽셀 색상 얻기.

그러나 엣지 디바이스에서 실시간 3D 가우시안 렌더링을 달성하는 것은 다음 세가지 이유로 어려움을 겪고 있다. 1) 계산 집약적 특성으로 인한 VR 단계의 시스템 지연 (59.4% 지분 차지) 2) 구면조화 함수 매개변수가 On-chip 메모리 보다 큼 (~114배) 3) 각 단계에서 다양한 유형의 행렬 연산 지원 필요.

따라서 이 문제를 해결하기 위해 다음 세가지 기능을 갖춘 재구성 가능한 FPGA

[1] KAIST: +Corresponding author: Hoi-Jun Yoo, hjyoo@kaist.ac.kr
(Received Nov. 29, 2024, Revised Dec. 21, 2024, Accepted Dec. 27, 2024)

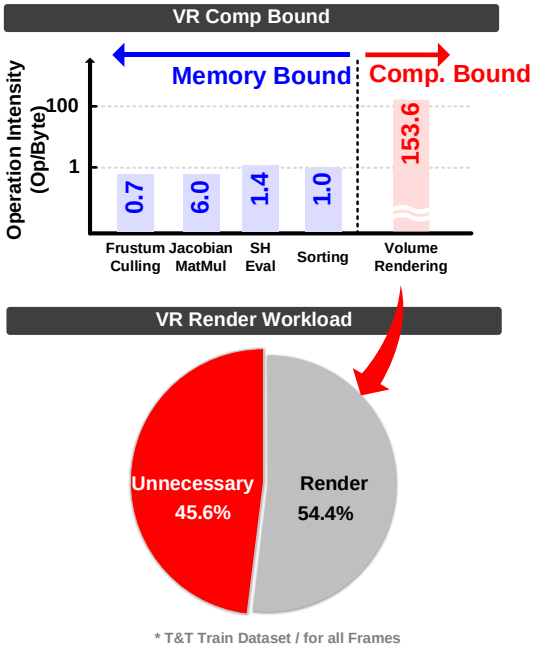


그림 1. 가우시안 스플래팅 연산분석

프로세서를 제안한다: 1) VR에서의 연산을 줄이기 위한 3단계 파이프라인 구조의 계층적 정렬 연산 유닛 (HGSU) 2) 균일한 (homogeneous) 구조에서 다양한 차원의 행렬 곱셈을 가속화하도록 설계된 차원 재구성 가능한 연산 코어 (DRMC) 3) 중요도 기반 구면조화 함수 변환 (IBSHE)를 통한 선택적 가우시안 특성 계산 처리.

II. 본 론

1. 전체 구조도

다음 그림 2.는 제안된 가속기의 전체 구조도를 보여준다. 두개의 주요 코어 (HGSU, DRMC) 가 존재한다. 두개 코어는 타일 매개변수 용 전역 메모리가 사이에 존재한다. HGSU는 8개의 32개 요소 정렬하는 정렬 유닛으로 구성되며, 각각의 정렬 유닛은 상위 컨트롤러가 연결되어 있다. 입력 및 출력 메모리는 각각 32KB으로 구성된다. 파이프라인에 따라 컨트롤러는 출력된 정렬 데이터를 구면조화 평가를 위한 SH Scorer 또는 VR 용 DRMC로 제공한다. DRMC는 144개의 PE 를 지닌 FP16 MAC으로 구성되며 입력 메모리는 128 KB, 출력 메모리는 64 KB이다. 결과는 상위 축적기에 누적되며 이는 LUT로 합성되어

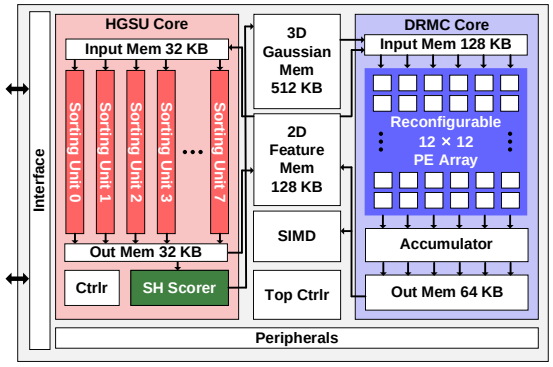


그림 2. 프로세서 구조도

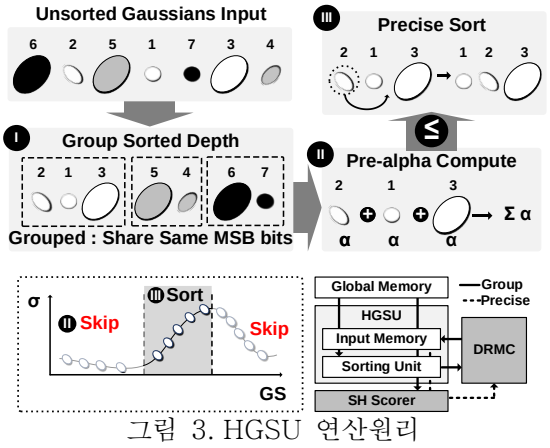


그림 3. HGSU 연산원리

있다. 전역 3D 매개변수 메모리 (512 KB)와 2D 매개변수 메모리 (128KB)는 계산에 필요한 매개변수 값을 저장한다. SIMD는 지수 계산 및 임계값 비교에 사용한다.

2. 프로세서 기능

(1) 계층적 정렬 연산

그림 3.는 VR 단계에서 필요한 가우시안 수를 줄이기 위한 HGSU의 동작을 보여준다. 이 방식은 기존 정밀 정렬 방식 이전에 두 단계를 추가하여 진행한다: 1) 가우시안 그룹화 2) 사전 알파 계산. 본래 정밀 정렬은 16b 가우시안의 깊이 값을 4b 단위로 기수를 나누어 정렬한다. 가우시안 그룹화 단계에서는 정렬되지 않은 가우시안 깊이가 16×16 픽셀의 타일 크기 단위로 연산된다. 전체 16b을 사용하는 대신, MSB 쪽 8b을 공유하는 가우시안들이 그룹으로 묶인다. 동일 그룹 내 가우시안들은 비슷한 공간을 점유하는 특성을 보여 이들의 점유값 또한 유사하게 나타나는 특성이 있다. 이 점을 활용하여 사전 알파 단계에서 각 그룹의 알파값을 예측하고 알파 값이 낮을 것으로 예상되는 그룹은 건너뛴다. 마지막으로 정밀 정렬 단계에서 남은

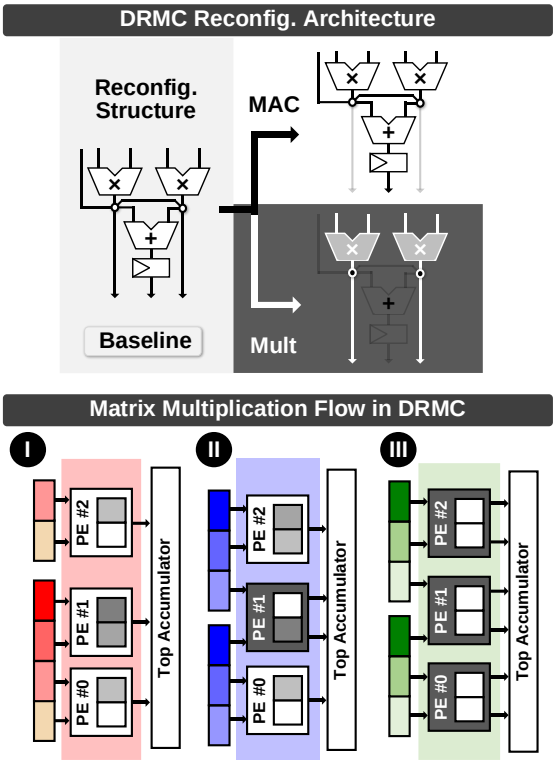


그림 4. DRMC 연산 구조

가우시안 그룹을 16b 정밀도로 정렬한 후 VR 단계에 입력한다. 이러한 제거 과정을 통해 24.3%의 가우시안이 감소하여 처리량이 1.2배 증가하며, PSNR 감소는 1 dB 미만의 감소로 영향이 미미하다.

(2) 다양한 차원의 행렬 곱셈 연산 코어

그림 4.은 서로 다른 행렬 곱셈 차원을 최대 지원 가능한 재구성 MAC 구조를 보여준다. 예시로 Frustum Culling 에서는 카메라와 이미지 투영을 위해 4×4 차원이, 2D 공분산 계산에는 3×3 차원이, VR에서는 스칼라 곱셈이 필요하다. 해당 연산들은 각기 짝수, 홀수 또는 스칼라 곱셈을 분류한다. 이러한 다른 특성을 가진 연산을 서로 다른 연산 구조로 가속화 시도할 경우, PE 사용율이 53.7%로 매우 크게 떨어진다.

본 논문에서는 균일한 차원 재구성 가능한 행렬곱셈 코어 구조를 통해 모든 행렬 연산을 유연하게 지원할 수 있도록 재구성 가능한 MAC 유닛을 제안한다. 예를 들어 Frustum Culling 작업을 지원하기 위해 모든 PE는 MAC 모드로 동작하며 PE#0은 p0 = (a11b11 + a12b21)을 계산하고, PE#1은 p1 = (a13b31 + b14b41)을 계산한다. 마지막으로 상위 축적기가 c11 =

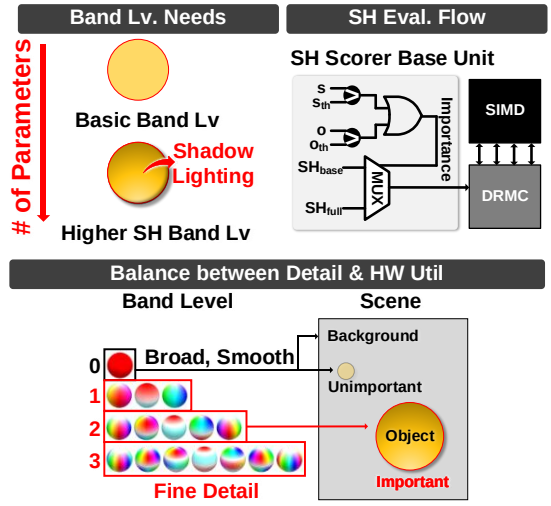


그림 5. 구면조화 중요도 평가 방법 및 유닛

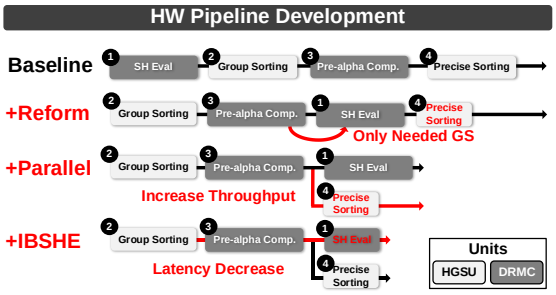
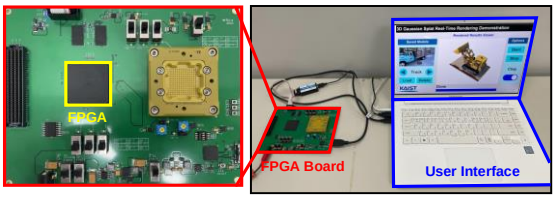


그림 6. 구면조화 적용 파이프라인

p0 + p1 을 생성한다. 투영 변환 연산의 경우, 세개의 PE 당 가운데 PE는 Mult 모드로 전환되어 개별 곱셈기만 사용한다. 이 방식은 모든 연산에서 PE를 최대한으로 활용해 연산 자원을 활용해 처리량을 1.4배 향상 시킬 수 있다.

(3) 중요도 기반 구면조화 함수평가

그림 5. 는 구면조화 중요도 평가 방법 및 유닛을 보여주고, 그림 6.은 변환 파이프라인을 보여준다. 첫 번째 단계는 HGSU에 의해 수정된 3DGS 파이프라인이 처리량 극대화를 위해 재구성한다. 그룹 정렬 및 사전 알파 계산이 구면조화 변환 앞에 진행된다. 사전 알파 계산은 가우시안 그룹을 건너뛰는 인덱스를 구면조화 변환 단계로 보내어, 해당 평가에 필요한 가우시안 수를 24.3% 줄인다. 또한 구면조화 평가 단계와 정밀 정렬 단계가 병렬화되어 각각 DRMC 코어와 HGSU 클러스터에서 실행된다. 그러나 파이프라인 재구성 시 구면조화 변환 지연 시간이 여전히 정밀 정렬 단계보다 크다. 이 문제를



FPGA Specifications	
Platform	Cyclone V
Clock	200 MHz
Logic Util.	103,342
DSP	288
Block Mem.	7,340,032

Implementation Results	
System	3DGS
Bit Precision	FP16
Avg. Perf.	66.6 FPS
Power	3.6 W

그림 7. FPGA 데모 시스템과 렌더링 결과

해결하기 위해 중요도 기반 구면조화 평가가 제안되었다., 각 가우시안은 0단계에서 3단계까지 여러 단계로 구성되어 더 높은 단계는 더 세밀한 묘사를 보여준다. 따라서 가우시안의 중요도 여부를 분류하고, 중요하지 않은 가우시안의 경우 기본 구면조화 단계를 사용하여 변환를 수행한다. Scoring Unit은 크기와 불투명도에 대한 임계값을 확인 후 OR 연산 적용하여 중요도 여부(flag) 를 생성한다. Flag가 0이면 두개의 가우시안은 기본 단계로 계산되며 그렇지 않으면 전체 단계가 변환되어 단일 가우시안의 색상을 반환한다. 이러한 단계를 종합하여 메모리 요구량이 67.2% 감소하고 처리량은 9.3 배 증가했다. 이러한 부분을 감안했을 때 PSNR 손실이 0.4 dB에 불과할 정도로 영향은 미미하다.

III. 결론

그림 7 은 제안된 3DGS 렌더링을 위한 FPGA 데모 시스템과 측정 결과를 보여준다. 제안된 프로세서는 인텔 Cyclone V 에서 구현되었고, 최대 클럭 주파수는

표 1. 다른 렌더링 시스템과 비교표

	ACM TOG'22	ASSCC'23	Edge GPU[3]	This Work
Platform	ASIC(Simul)	Cyclone V	GPU	Cyclone V
Application	Implicit NeRF	Explicit NeRF	3DGS	3DGS
PSNR (dB)[1]	28.9	29.1	26.0	37.1
Frequency (MHz)	400	200	1400	200
Power (mW)	0.3	5.8	7500	3.6
Bit Precision	FXP9	FXP4/2	FP16	FP16
Frame Rate (FPS)[2]	0.02	33.6	30	66.6
Energy per Frame	1414.0	172.6	250	54.1

1) Synthetic NeRF Lego Dataset rendered results
2) 800 * 800 Resolution
3) Jetson Orin Nano

200MHz 이다. 기존의 3D 렌더링 FPGA 프로세서와 비교했을 때, 제안된 가속기는 66.6 FPS 의 렌더링 속도와 3.6 W 의 전력 소비로 큰 성과를 거두었다. 이는 Synthetic NeRF Lego Dataset 에서 기존의 FPGA 프로세서보다 높은 정확도를 달성한 것이다. 각 기능이 적용된 Synthetic NeRF Lego 및 T&T 트럭 데이터셋에서 렌더링 품질 결과를 보여준다.

표 1 을 통해 다른 렌더링 시스템과 비교 시 기존의 NeRF 렌더링 시스템보다 확연히 높은 PSNR 과 빠른 FPS 를 보여준다. 같은 시스템의 Edge GPU 랑 비교했을 시에도 빠른 FPS 와 높은 퀄리티를 보여준다.

결론적으로, 계층적 정렬과 선택적 구면조화 변환을 결합한 재구성 가능한 연산 구조를 갖춘 3DGS 프로세서를 제안하여 각 처리 단계에서 문제를 해결하고, 66.6 FPS 의 고품질 3D 렌더링 시스템을 구현했다.

감사의 글

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 인공지능반도체고급인재양성사업 연구결과로 수행되었음 (IITP-2024-RS-2023-00256472)

참고 문헌

논문지 논문 인용
[1] B. Kerbl, G.Kopanas, T.Lemkuhler, and G. Drettakis, "3D Gaussian Splatting for Real-Time Radiance Field Rendering", *ACM Trans actions on Graphics*, vol. 42, July 2023

학술 대회 논문 인용

[2] C. Blanchard, L. Gupta, and S. Nanisetty, “Analyzing 3D Gaussian Splatting and Neural Radiance Fields: A Comparative Study on Complex Scenes and Sparse Views”, cs.tornto.edu.
[3] C. Rao et al., “ICARUS: A Specialized Architecture for Neural Radiance Fields Rendering, “ ACM Trans. Graph 2022
[4] J. Park et al., “A 33.6 FPS Embedding based Real-time Neural Rendering Accelerator with Switchable Computation Skipping Architecture on Edge Device”, ASSCC 2023.

기타 인용
[5] NVIDIA Inc., NVIDIA Jetson Orin Nano, https://www.nvidia.com/en-us/auonomous-machies/embedded-systems/jetson-orin

이 홍 석 (Hongseok Lee)

2024년 2월 : KAIST 전기 및 전자공학부 졸업
2024년 3월~현재 : KAIST 인공지능반도체 대학원 석사과정

<관심분야> 시스템반도체 설계, 딥러닝 프로세서

박 원 훈 (Wonhoon Park), 학생회원

2021년 8월 : KAIST 전기 및 전자공학부 졸업
2023년 8월 : KAIST 전기 및 전자공학부 석사
2023년 9월~현재 : KAIST 전기 및 전자공학부 박사과정

<관심분야> 딥러닝 프로세서, 딥러닝 시스템 반도체 설계

안 상 혁 (Sanghyuk Ahn)

2023년 2월 : 성균관대학교 전기전자공학부 졸업
2023년 3월~현재 : KAIST 전기 및 전자공학부 석사과정

<관심분야> 딥러닝 프로세서, 프로세싱-인-메모리(PIM), 뉴로모픽 하드웨어 설계

김 민 성 (Minsung Kim)

2024년 2월 : 고려대학교 전자전자공학부 졸업
2024년 3월~현재 : KAIST 인공지능반도체 대학원 석사과정

<관심분야> 딥러닝 프로세서, 프로세싱-인-메모리(PIM), 뉴로모픽 하드웨어 설계

유 회 준 (Hoi-Jun Yoo), 평생회원

1983년 2월 : 서울대학교 전자공학과 졸업
1988년 8월 : KAIST 전기 및 전자공학부 박사
1988년 9월~1990년 12월: 미국 벨연구소 연구원
1991년 2월~1995년 2월: 현대전자 반도체연구소 DRAM 설계실장
1998년 2월~현재: KAIST 전기 및 전자공학부 교수

<관심분야> 집적회로 설계, 멀티미디어 SoC 설계, 고속 및 저전력 메모리