A-SSCC 2025 Review

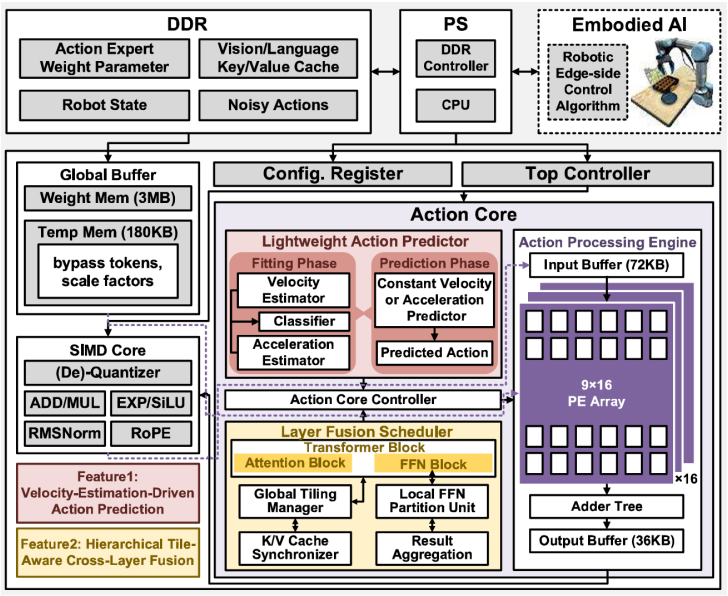
KAIST 전기 및 전자공학부 석사과정 이가은

Session 7 Application-Driven FPGA Circuits and Systems

이번 A-SSCC 2025의 Session 7에서는 엣지 환경에서의 실시간 인공지능 가속 및 고성능 비디오·로보틱스 처리를 위한 하드웨어 가속기를 주제로 총 4편의 논문이 발표되었다. 본 세션에서는 로봇 행동 계획, 비디오 트랜스포머, 3D 렌더링 등 계산 복잡도와 메모리 요구량이 큰 AI 워크로드를 대상으로, FPGA 기반의 저지연·고효율 가속 구조를 제안한 연구들이 소개되었다. 특히 토큰 수 감소, 예측 기반 연산 생략, 캐시 및 프리패칭을 활용한 메모리 병목 완화 등 알고리즘–하드웨어 협업(Co-design) 기법이 공통적으로 강조되었으며, 엣지 디바이스에서 실시간 성능을 달성하기 위한 다양한 설계 전략이 집중적으로 논의되었다. 본 리뷰에서는 Session 7에 포함된 논문 중에서도, 2편의 논문을 살펴보고자 한다.

#7-1 본 논문은 Fudan University 와 ZTE Corporation 에서 발표한 연구로, Embodied Artificial Intelligence(EAI) 환경에서 로봇의 실시간 행동 계획을 수행하기 위한 FPGA 기반 하드웨어 가속기를 제안한다. 최근 로보틱스 분야에서는 시각 정보와 언어 지시를 입력으로 받아 연속적인 행동 시퀀스를 생성하는 Vision-Language-Action(VLA) 모델이 주목받고 있으나, diffusion 또는 transformer 기반 구조로 인해 계산 복잡도와 지연 시간이 커 엣지 환경에서 실시간 제어 주기를 만족시키기 어렵다. 기존 연구들은 병렬화나 모델 경량화를 통해 성능 개선을 시도해왔지만, 행동 계획과 같이 시간적으로 연속된 토큰을 생성하는 문제에서는 예측 가능한 구간에서도 동일한 신경망 연산이 수행되어 불필요한 연산과 전력 소모가 발생한다.

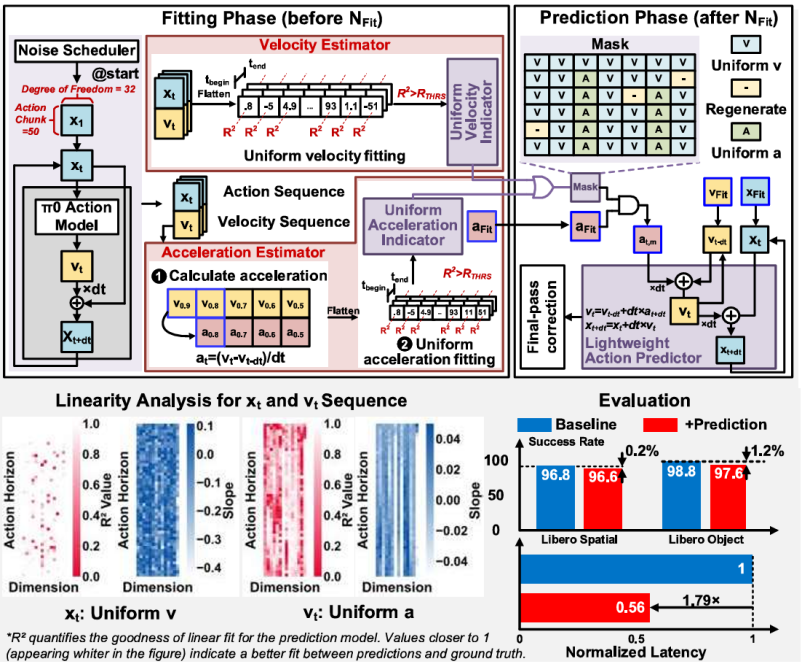
본 논문에서는 이러한 문제를 해결하기 위해 속도 추정 기반 행동 예측(Velocity-Estimation-Driven Behavior Prediction)이라는 새로운 접근 방식을 제안한다. 제안된 방법은 행동 생성 과정을 시간 축에서 분석하여, 모든 토큰을 동일하게 처리하는 대신 토큰의 동역학적 특성에 따라 처리 방식을 달리하는 것을 핵심으로 한다. 구체적으로, 로봇의 최근 행동 이력을 기반으로 각 자유도에 대한 속도 및 가속도를 추정하고, 해당 구간이 선형 운동으로 근사 가능한지를 판단한다. 이 과정에서 결정 계수(R^2)를 사용하여 행동 변화의 선형성을 정량적으로 평가하며, 일정 임계값 이상일 경우 해당 구간을 예측 가능 구간으로 분류한다.



[그림 1] Overall architecture

예측 가능 구간에서는 신경망 기반 행동 생성 대신, 이전 상태에서 추정된 속도 또는 가속도를 이용한 외삽 연산을 통해 다음 행동을 생성한다. 반면, 장애물 회피나 방향 전환과 같이 비선형성이 큰 구간에서는 기존 diffusion 기반 행동 생성 모델을 그대로 적용한다. 이를 통해 전체 행동 시퀀스 중 상당 부분을 차지하는 선형 구간에서 고비용 신경망 연산을 제거할 수 있으며, 계산 복잡도와 전력 소모를 동시에 줄일 수 있다.

하드웨어 아키텍처 측면에서, 논문에서는 이러한 알고리즘적 아이디어를 반영한 계층적 파이프라인 구조를 FPGA 상에 구현하였다. 전체 시스템은 크게 행동 입력 처리 모듈, 속도 추정 및 선형성 평가 모듈, 예측 기반 행동 생성 모듈, 그리고 신경망 추론 모듈로 구성된다. 속도 추정 모듈은 최근 여러 시점의 행동 데이터를 저장하고, 선형 회귀 연산을 수행하여 속도 벡터를 계산한다. 이후 선형성 평가 결과에 따라 Prediction Phase 또는 Correction Phase 로 동작 경로가 분기된다. Prediction Phase 에서는 간단한 산술 연산을 통해 행동을 외삽하며, 이 과정은 매우 낮은 지연 시간과 전력으로 수행된다. Correction Phase 에서는 일정 주기마다 신경망 추론 결과를 사용하여 예측 결과를 보정함으로써, 장시간 동작 시 발생할 수 있는 누적 오차를 제한한다. 이러한 구조는 실시간성을 유지하면서도 행동 정확도를 안정적으로 확보할 수 있도록 설계되었다. 또한 FPGA 자원 활용 측면에서도, 고비용 연산 유닛의 활성 빈도를 줄여 전체 시스템 효율을 향상시킨다.

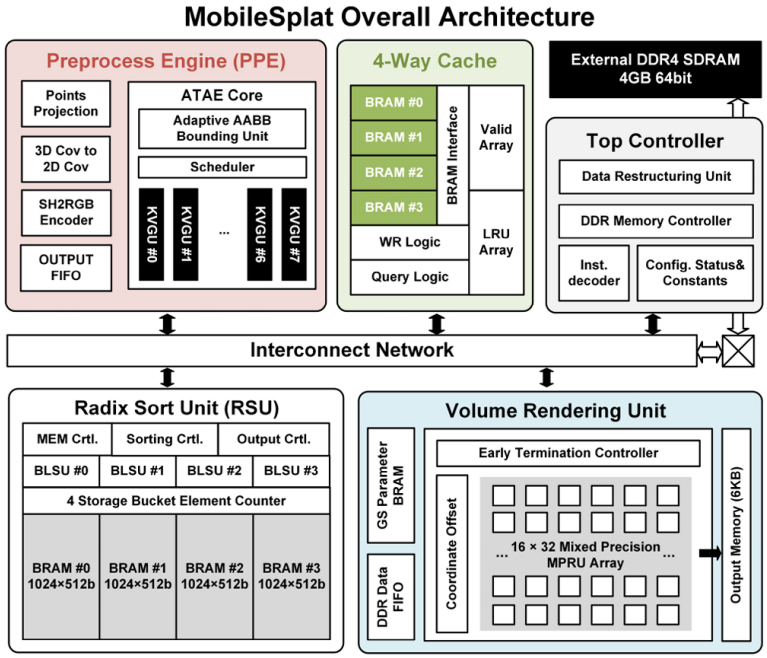


[그림 2] Velocity-estimation-driven behavior prediction mechanism,

제안된 가속기는 실제 로봇 행동 계획 시나리오에서 평가되었으며, 최대 102.5 Hz의 행동 계획 주기를 달성하였다. 또한 예측 기반 연산 생략을 통해 평균 처리 속도가 최대 1.79배 향상되었고, 행동 성공률 역시 기존 방식과 유사한 수준을 유지하였다. 종합적으로 본 논문은 Embodied AI 행동 생성 문제를 단순한 신경망 가속이 아닌, 로봇 동작의 시간적·물리적 특성을 활용한 시스템 수준 최적화 문제로 재정의하였다.

#7-2 본 논문은 Tsinghua University Shenzhen 캠퍼스 연구진과 AMD 의 공동 연구로, 3D Gaussian Splatting 을 모바일 및 엣지 환경에서 실시간으로 구현하기 위한 FPGA 기반 렌더링 프로세서 MobileSplat 을 제안한다. 3D Gaussian Splatting 은 NeRF 대비 빠른 렌더링 속도와 높은 시각적 품질로 다양한 3D 비전 응용에서 주목받고 있으나, 다수의 Gaussian primitive 를 타일 단위로 처리해야 하므로 연산량과 메모리 접근 비용이 커 엣지 환경에서의 실시간 구현이 어렵다.

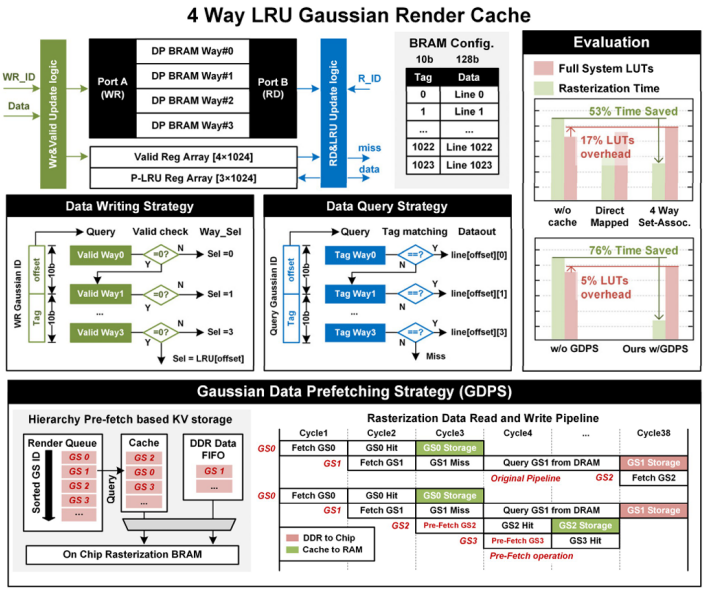
기존 연구들은 GPU 기반 병렬 처리로 성능을 확보해왔으나, 전력 소모와 시스템 비용 측면에서 모바일 환경에 적합하지 않으며, Gaussian 데이터의 반복적 접근으로 인한 메모리 병목을 근본적으로 해결하기 어렵다. 이러한 한계를 해결하기 위해 본 논문은 Gaussian Splatting 의 연산 및 데이터 접근 특성을 분석하고, 불필요한 연산과 메모리 접근을 구조적으로 제거하는 하드웨어 아키텍처를 제안한다.



[그림 3] Overall Architecture of MobileSplat

본 논문에서 제안한 MobileSplat 의 핵심 설계 철학은 연산 유닛의 처리 속도를 높이는 것보다, 렌더링 파이프라인 전반에서 처리해야 할 데이터의 양 자체를 줄이고 데이터 이동을 최소화하는 데 있다. 이를 위해 첫 번째로 도입된 기법은 Pre-culling Architecture 이다. 제안된 Pre-culling 구조는 Gaussian 의 투영 범위를 사전에 분석하여, 특정 타일에 영향을 미치지 않는 Gaussian 을 렌더링 파이프라인 초기에 제거함으로써 불필요한 연산을 방지하며, 평균적으로 처리해야 하는 타일 수를 줄인다. 두 번째 핵심 요소는 Gaussian 데이터의 반복적인 접근 특성을 활용한 온칩 캐시 구조이다. MobileSplat 은 4-way set-associative 구조의 Gaussian Render Cache 를 FPGA 내부 BRAM 으로 구현하여 최근에 사용된 Gaussian 데이터를 저장한다. 세 번째로, 본 논문에서는 Gaussian Splatting 의 시각적 특성을 고려한 Mixed-Precision Computation Path 를 제안한다. 모든 연산을 동일한 고정밀도로 수행하는 대신, 최종 화질에 미치는 영향이 상대적으로 작은 연산은 저정밀도로 처리하고 민감한 연산만 고정밀 경로를 통해 수행함으로써 연산량과 전력 소모를 줄인다. 이러한 접근은 최근 AI 및 그래픽스 하드웨어 설계에서 널리 사용되고 있는 정확도–효율 트레이드오프를 잘 반영한 사례라 할 수 있다. 메모리 병목 완화를 위해 제안된 Gaussian Data Prefetching Strategy(GDPS) 또한 본 논문의 중요한 기여 중 하나이다. GDPS 는 Gaussian 데이터 접근 패턴을 분석하여 향후 필요할 것으로 예상되는 데이터를 미리 DRAM 에서 읽어와 내부

파이프라인에 공급함으로써, 외부 메모리 접근 지연을 내부 연산과 겹쳐 숨기고 전체 렌더링 지연을 효과적으로 완화한다.



[그림 4] 4 Way LRU Gaussian Render Cache

실험 결과, 제안된 MobileSplat 은 다양한 3D 장면에 대해 최대 105 FPS 의 실시간 렌더링 성능을 달성하였다. Pre-culling 기법을 통해 평균 타일 수를 최대 35%까지 감소시켰으며, Gaussian Render Cache 와 GDPS 를 통해 rasterization 단계의 수행 시간을 크게 단축하였다. 또한 제한된 FPGA 자원 내에서 구현이 가능함을 보였고, 전력 효율 측면에서도 모바일 환경에 적합한 수준을 달성하였다. 종합적으로 본 논문은 3D Gaussian Splatting 의 병목을 연산량 증가의 문제로만 보지 않고, 데이터 이동과 메모리 접근 관점에서 재정의하였다는 점에서 의의가 크다.

저자정보

이가은 석사과정 대학원생

- 소속 : 한국과학기술원 전기 및 전자공학부
- 연구분야 : 디지털 회로 설계
- 이메일 : gelee@ics.kaist.ac.kr
- 홈페이지 : https://idec.or.k