

DVS 카메라를 이용한 Spiking Neural Network 시뮬레이션을 위한 인터페이스 개발

권용인, 허인구, 이종원, 백윤홍
서울대학교 전기정보공학과

e-mail:yikwon, igheo, jwlee@optimizer.snu.ac.kr, ypaek@snu.ac.kr

Implementing Interface for Spiking Neural Network Simulation for DVS Camera

Yong-in Kwon, In-gu Heo, Jong-won Lee, Yun-heong Paek
Dept. of Electrical Engineering, SNU

요 약

DVS 카메라는 인간의 눈을 모델링하여 만들어져서 화면의 변화에 반응하여 Address - Event - Representation 데이터를 생성하고 이 데이터는 jAER Viwer를 통해 확인할 수 있다. 이렇게 생성된 DVS 카메라의 데이터를 Spiking Neural Network의 입력으로 주기 위해 GPU를 이용한 Spiking Neural Network 시뮬레이터인 GPUSNN과 jAER 사이에 인터페이스가 필요하다. 이 인터페이스를 이용하면 GPUSNN을 통해 비전 알고리즘을 빠르고 효과적으로 Spiking Neural Network 시뮬레이션을 할 수 있을 것이다.

1. 서론

최근 Spiking Neural Network (SNN) 모델이 여러 Neural Network 사이에 현실적인 모델로 떠오르고 있다. Firing Rate-based 모델들과는 다르게 SNN 모델은 Spike의 정확한 시간 구조를 포함하기 때문에 여러 가지 흥미로운 성질들을 표현할 수 있다. SNN 모델은 또한 높은 생물학적 정확성을 가지고 있어, 여러 가지 두뇌구조의 특성들을 모델링 할 수 있다.

하지만 대부분의 소프트웨어 시뮬레이션에서 Neural Network은 CPU에서 순차적으로 처리되고, 한 번에 하나의 Neuron만이 계산이 된다. 결과적으로 Network가 커지면 커질수록 성능이 떨어지고 인간의 두뇌와 같은 방대한 크기의 Neural Network을 시뮬레이션 하려면 슈퍼컴퓨터나 분산컴퓨터로도 힘들다.

전통적인 CPU 아키텍처는 병렬처리가 불가능하기 때문에 Graphic Processing Units(GPUs)가 그 대안으로 떠오르고 있다. 게이밍 마켓이 제공하는 강력하고 비교적 값이 싼 병렬수행이 가능한 플랫폼인 GPU들은 원래 그래픽 렌더링을 위해 개발되었고 CPU의 대용으로 사용하기는 힘들었다. 하지만 NVIDIA의 CUDA(Compute Unified Device Architecture)와 같은 여러 프레임워크가 만들어짐으로서 프로그래머가 C 코드를 이용하여 GPU를 이용하여 병렬 연산을 쉽게 할 수 있게 되었다. 이에 따라 CUDA 그래픽 프로세서를 이용하여 Large-scale의 Spiking Neural Network를 효율적으로 시뮬레이션하는 툴인 GPUSNN이 개발됨에 따라 간단한 설정 만으로

Spiking Neural Network를 시뮬레이션 할 수 있게 되었다.

Dynamic Vision Sensor(DVS)는 생물학적 시각 시스템에서 기인하여 개발된 CMOS Vision Sensor이다. DVS 카메라를 이용하여 Vision 입력을 받고, 입력받은 데이터를 Vision 알고리즘을 적용한 GPUSNN 툴을 이용하여 시뮬레이션 하면 물체인식, 동작인식 등의 다양한 어플리케이션을 개발 할 수 있을 뿐만 아니라, 생물학적 두뇌 연구에도 큰 도움이 된다.

2. 시뮬레이터

GPUSNN 시뮬레이터는 Izhikevich의 Spiking Neural Network를 모델링했다. 이 모델은 다른 클래식한 모델들보다 더 다양한 Neural Response를 생성해낼 수 있다. 또한 다른 모델보다 훨씬 적은 계산량을 필요로한다. Izhikevich neuron은 다음의 표현들에 의해 표현된다.

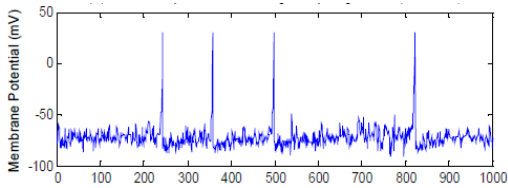
$$v' = 0.04v^2 + 5v + 140 - u + I$$

$$u' = a(bv - u)$$

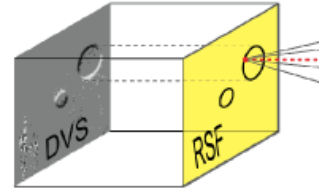
$$if (v \geq +30 mV) \text{ then } \{ v = c \text{ and } u = u + d \}$$

v는 Neuron의 membrane 전위를 나타내고, u는 recovery 변수이다. membrane 전위가 30mV를 넘어서면 firing하고 다시 초기값을 가지게 된다. 그림 1

은 membrane 전위가 변하고 firing하는 예이다.

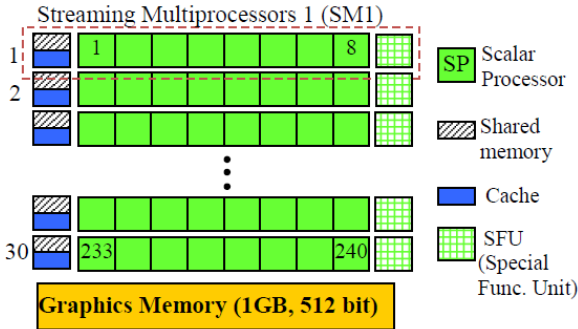


(그림 1) Spiking Neuron의 Membrane 전위



(그림 3) CUDA GPU Architecture

3. GPU 아키텍처 및 GPUSNN



(그림 2) CUDA GPU Architecture

그림 2는 NVIDIA의 CUDA GPU 아키텍처의 간단한 모습이다. CUDA GPU 아키텍처는 스트리밍 멀티프로세서(Streaming Multiprocessors)의 어레이를 갖고 있으며 각각의 스트리밍 멀티프로세서는 여덟개의 floating - point scalar processors(SPs)를 갖고있으며 하나의 스페셜 평션 유닛(SFU) 등을 갖고 있다.

CUDA 에서는 Generality와 Efficiency 사이에 Trade-off를 만드는 여러 가지 특성이 존재한다. CUDA GPU에서 Spiking Neural Network를 시뮬레이션하는 성능에 영향을 주는 요소는 다음과 같다. - 병렬수행, 메모리 Bandwidth, 메모리 사용량, 쓰레드 divergence

위의 요소들은 서로 연관되어있으므로 네 가지의 모든 요소가 최적화되어야 GPU에서의 수행이 효율적일 수 있다. 따라서 각각의 요소들을 최적화시키는 것이 중요하다.

GPUSNN은 CUDA GPU의 병렬 아키텍처 구조를 이용하여 Spiking Neural Network를 시뮬레이션 하는 툴이다. GPUSNN은 10만개의 뉴런과 평균 7Hz 속도로 firing 하는 5천만개의 시냅틱 연결을 시뮬레이션 할 때 CPU 버전보다 최고 26배 속도가 빠르다. 그리고 10만개의 뉴런과 천만개의 시냅틱 연결을 시뮬레이션 한다면 리얼타임에 비해 1.5배 정도만 느리다. 따라서 GPUSNN을 이용한다면 Vision 등과 같이 리얼타임 환경이 필요한 알고리즘을 시뮬레이션 할 수 있다.

4. DVS 카메라

GPUSNN에 입력으로 쓰이는 카메라는 그림 3과 같다

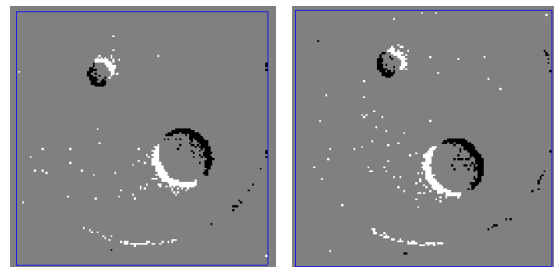
DVS는 화면에서 움직임에 반응하여 AER(Address - Event - Representation) 이벤트를 생성한다. 불필요한 데이터와 컴퓨팅 타임을 줄이기 위해서 DVS는 refractory spike filter(RSF)를 통해 먼저 필터링을 한다. RSF는 refractory 타임 보다 작은 시간동안의 변화는 필터링 시켜버린다.

5. jAER Interface

jAER 은 AER 시스템으로 리얼타임 spike-based 프로 세싱을 위한 java 프로젝트이다. jAER은 USB 인터페이스를 통해 DVS와 PC를 연결시켜주고 모니터를 통해 AER 데이터를 표시해준다. 이러한 jAER과 GPUSNN을 연결시켜주기 위한 인터페이스가 필요하고 이러한 인터페이스를 통해 DVS 로부터 읽어들이는 AER정보가 jAER을 통해 PC 화면에 표시되고 jAER 인터페이스를 통해 GPUSNN에 전달되어 Spiking Neural Network 시뮬레이션이 된 후 다시 jAER 인터페이스를 통해 jAER에 전달되어 PC를 통해 시뮬레이션 결과를 확인할 수 있다.

6. 실험 및 결론

jAER Interface의 동작을 실험하기 위해 DVS로부터 생성된 AER 데이터가 jAER로부터 GPUSNN으로 전달 하였다. GPUSNN의 Spiking Neural Network 구성은 128 * 128 의 1-Layer 구조로 입력을 받는 즉시 firing 하도록 구성하였다. 결과적으로 GPUSNN은 입력받은 AER을 약간의 딜레이로 버퍼링 해주게 된다. 그림 4는 실험 결과이다.



(그림 4) AER Viewer (GPUSNN시뮬레이션 전과 후)

이 논문에서는 DVS 카메라를 이용한 Spiking Neural Network 시뮬레이션을 위해 GPUSNN과 jAER을 연결하는 jAER Interface를 만들고, GPUSNN에 버퍼링 알고리즘을 적용함으로써 jAER Interface의 동작을 확인하였다.

GPUSNN 시뮬레이터와 DVS 카메라를 연결하면 Vision 알고리즘을 적용한 Spiking Neural Network 시뮬레이션을 GPU를 이용하여 효과적으로 수행할 수 있을 거라 기대되며, 사물인식이나 동작 인식 등의 여러 Vision 알고리즘을 GPUSNN에 적용 가능하게 될 것이다.

참고문헌

- [1] Jayram Moorknikara “Efficient Simulation of Large-Scale Spiking Neural Networks Using CUDA Graphics Processors” International conference on neural networks.
- [2] E. M. Izhikevich “Large-scale model of mammalian thalamocortical systems” Proceedings of the National Academy of Sciences, 2008
- [3] E. M. Izhikevich, “Simple model of spiking neurons” Neural Networks, IEEE Transactions on, vol. 14, no. 6, pp. 1569-1572, 2003
- [4] NVIDIA Programming manual Version 2.0. See Appendix A for technical specifications
- [5] J. M. Nageswarana “Computing Spike-based Convolutions on GPUs” ISCAS, pp. 197-1920, 2009
- [6] Lichtsteiner, P., C. Posch “A 128X128 120dB 15us Latency Asynchronous Temporal Contrast Vision Sensor” IEEE JSSC, 43(2) 566-576, 2008

ACKNOWLEDGE

본 연구는 교육과학기술부/한국과학재단 우수연구센터 육성사업(과제번호 2011-0000975), 2011년도 정부(교육과학기술부)의 재원으로 한국과학재단의 국가지정연구실사업(No.2011-0018609) 및 IDEC의 지원을 받아 수행되었습니다.