National Science Foundation's Industry/University Cooperative Research (I/UCRC) Program

3D Generative Adversarial Networks inference implementation on FPGAs

Chao Jiang, David Ojika, Herman Lam:
*SHREC @ UF**

Federico Carminati, Sofia Vallecorsa:
CERN openlab

Thorsten Kurth, Prabhat:
*NERSC Berkeley Lab***

Bhavesh Patel:
Dell EMC

* SHREC: NSF Center for Space, High-Performance, and Resilient Computing, University of Florida

** NERSC: National Energy Research Scientific Computing Center, Lawrence Berkeley National Lab

Mission-Critical Computing
NSF CENTER FOR SPACE, HIGH-PERFORMANCE, AND RESILIENT COMPUTING (SHREC)

University of Pittsburgh
BYU BRIGHAM YOUNG UNIVERSITY
Virginia Tech
UF UNIVERSITY of FLORIDA
CERN openlab
CHEP 2019

Outline

- Hetergeneous computing for deep learning
  - Data pre-processing; model training; model inference
  - Focus on FPGA-acceleration of inference stage
- Experimental platforms & tools
  - Intel PAC10 card; OpenVINO; DLA* design suite

- Case studies
  - 3DGAN *(new)*:
    - Initial results
- Conclusions & going forward

* DLA: Deep Learning Accelerator

Heterogeneous Computing[1] for Deep Learning

Motivation

- Deep learning becoming *pervasive* for mission-critical computing
  - *Heterogeneous computing** offers unique capabilities to accelerate DNNs[2]

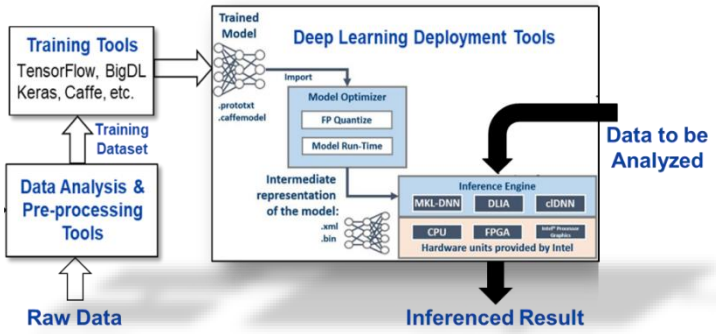
Goal

Perform design-space exploration:

- Of *emerging HGC[1] archs/tools* and *DNN models*
- For *acceleration* of selected mission-critical apps

Approach

Focus on use of *FPGAs* to *accelerate inference stage* of the HGC workflow



Stages of HGC workflow

- Data analysis & pre-processing
- Model training
- DNN inference

Collaborating partners

- ***NERSC**:*** HepCNN, CosmoGAN model support
- ***CERN openlab:*** 3D GAN model support
- ***Dell:*** SHREC membership support, equipment
- ***Intel:*** Deep-learning tools; engineering support

DNN Models *from NERSC & CERN Openlab*

- HEP-CNN
- CosmoGAN
- 3D GAN

*HGC: Heterogeneous Computing (CPU+GPU+FPGA)

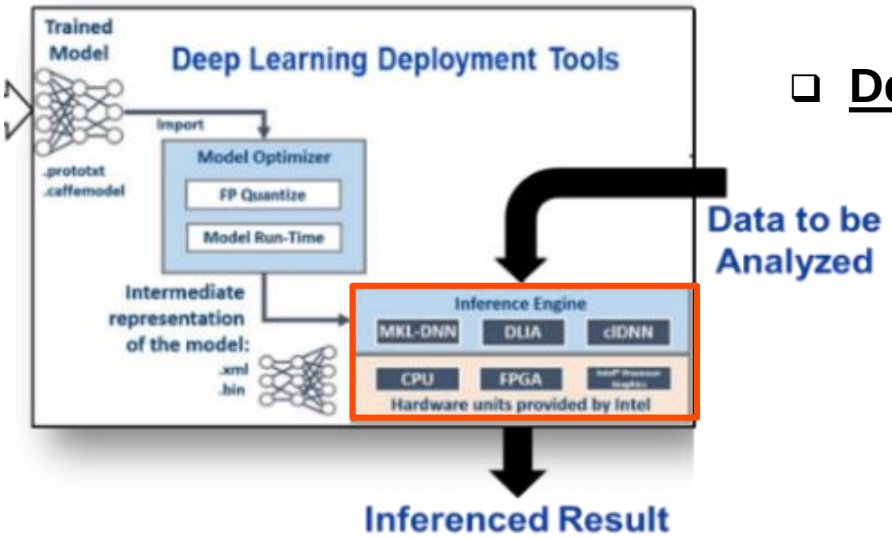
**NERSC: National Energy Research Scientific Computing Center, Lawrence Berkeley National Lab

FPGA Acceleration for DNN Inference

Experimental Setup & Tools

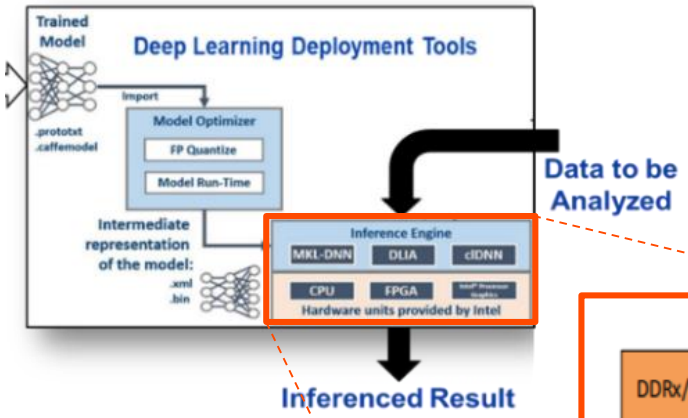
- **Experimental platforms**
  - **Dell EMC server:** 2x Intel Xeon Gold 6130 CPU
  - **Intel PAC:** Arria 10 GX FPGA

- **Intel OpenVINO Toolkit**
  - Model Optimizer
    - Convert mainstream deep learning framework model *(TensorFlow, Caffe, etc.)* into unified *intermediate representations (IR)*
  - Inference Engine
    - API library for mapping IR onto *Intel hardware platforms (CPU, GPU, FPGA, etc)*
    - Integrated with *Deep Learning Accelerator suite* for *FPGA acceleration*



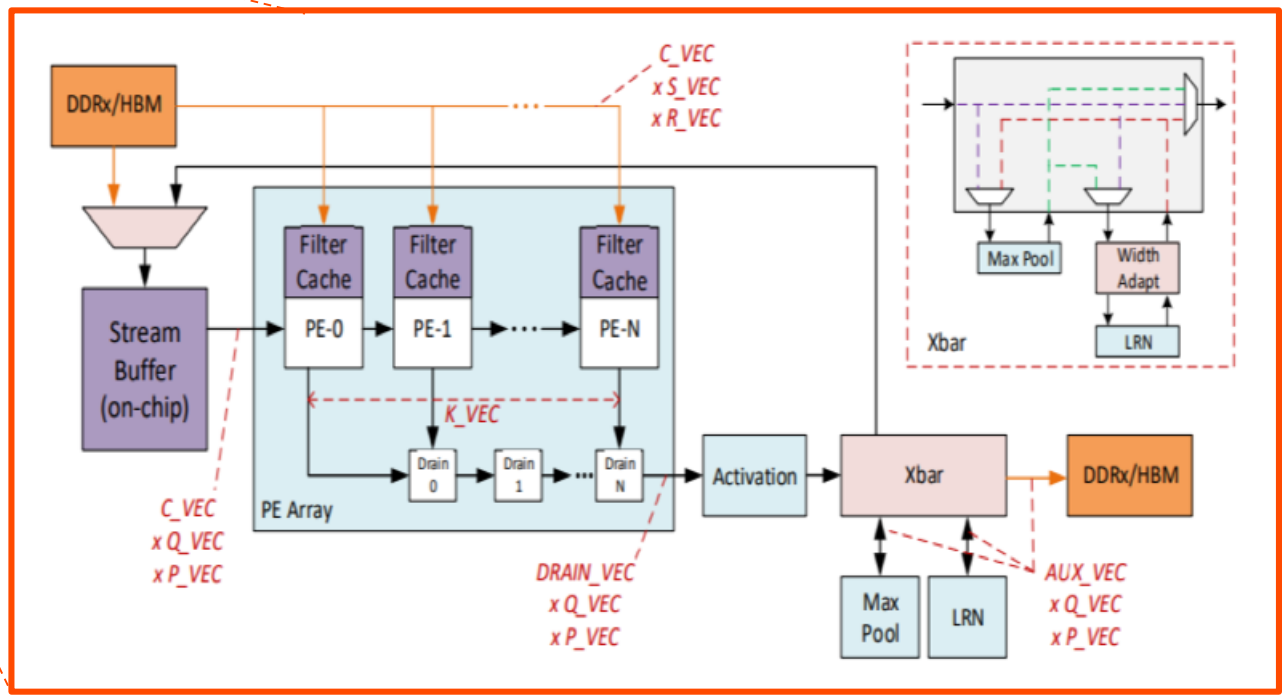
- **Deep Learning Accelerator suite (DLA)**
  - *OpenCL-based* implementation of DNN inferencing hardware architecture
  - Source code acquired through NDA with Intel to be *optimized for various applications*

Deep Learning Accelerator Suite (DLA [1])



- *OpenCL-based* implementation of DNN inferencing hardware architecture
- Source code acquired through NDA with Intel to be *optimized for various applications*

- DDR/HBM
- Stream Buffer
- PEs: processing elements
- Activation module
- Xbar
- Max Pool module
- LRN: normalization



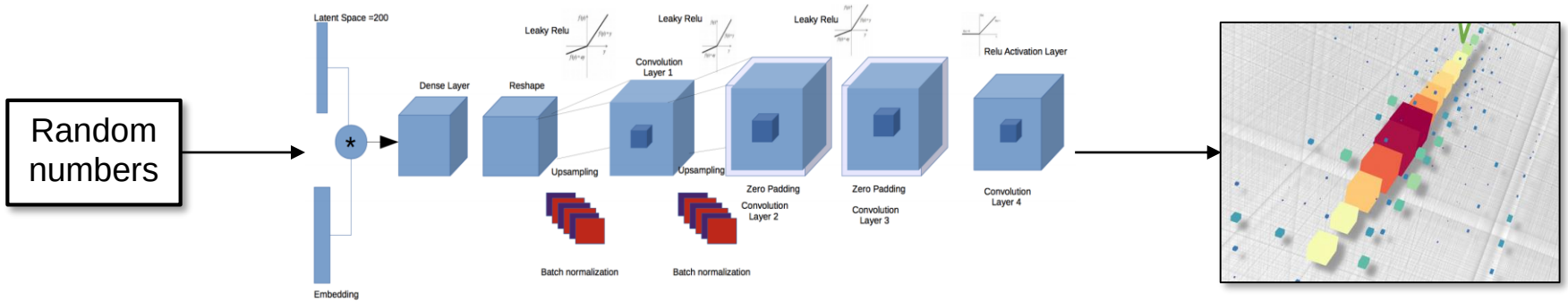
[1] Abdelfattah, M.S. et al., DLA: Compiler and FPGA Overlay for Neural Network Inference Acceleration. arXiv e-prints arXiv:1807.06434 (Jul 2018)

Outline

- Hetergeneous computing for deep learning
  - Data pre-processing; model training; model inference
  - Focus on FPGA-acceleration of inference stage
- Experimental platforms & tools
  - Intel PAC10 card; OpenVINO; DLA* design suite
- Case studies
  - 3DGAN *(new)*:
    - Initial results
- Conclusions & going forward

* DLA: Deep Learning Acceleration

Case Study: 3DGAN[1] Model from CERN openlab



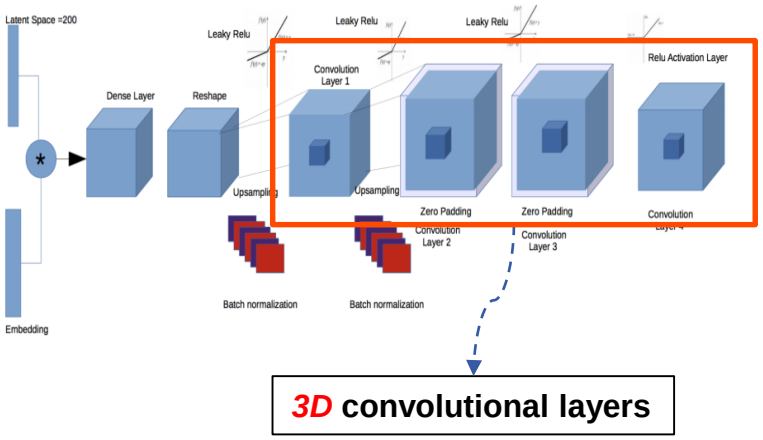
Particle shower simulation

- Generate a 3D simulation of a particle detector in HEP experiments
- Developed and trained by openlab at CERN using *3DGAN* topology with upsampling + 3D convolutional layers

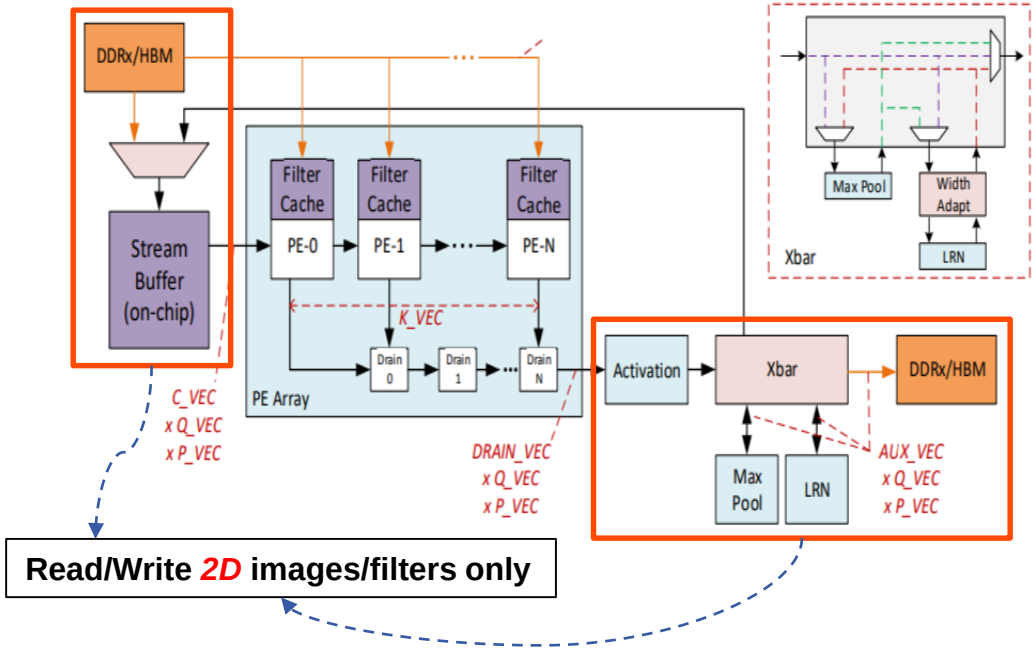
[1]

FPGA Primitive to Support 3D Convolutional Layer

3DGAN model structure



System-level architecture of DLA



Problem:

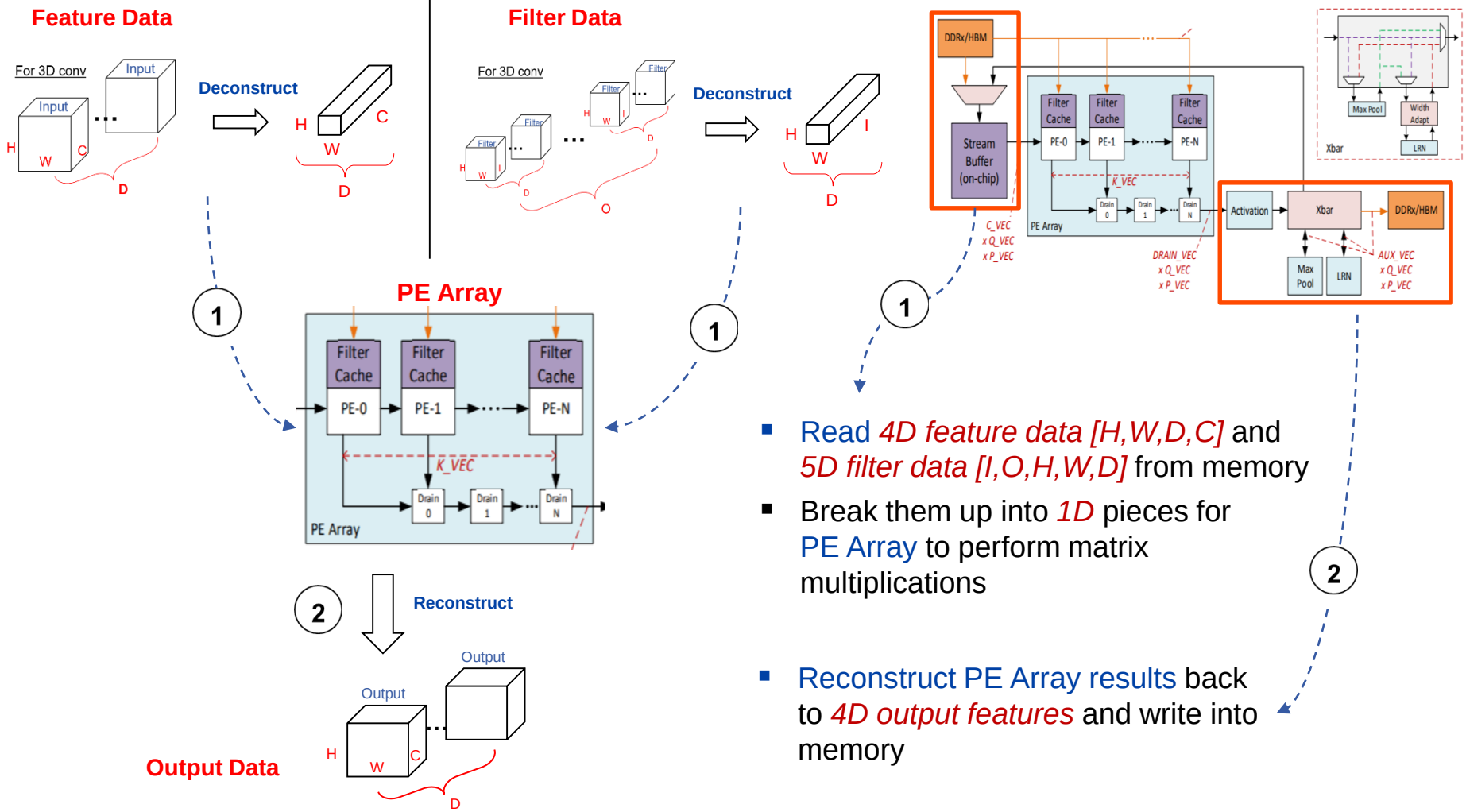
- DLA can not inference *3D convolutional layers on FPGA* natively

Solution:

- *Customize* DLA to implement *3D convolutional primitive*

[1] Abdelfattah, M.S. et al., DLA: Compiler and FPGA Overlay for Neural Network Inference Acceleration. arXiv e-prints arXiv:1807.06434 (Jul 2018)

FPGA Primitive to Support 3D Convolutional Layer



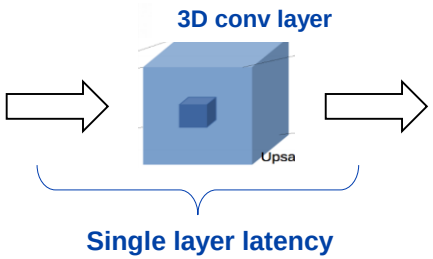
- Read *4D feature data [H,W,D,C]* and *5D filter data [I,O,H,W,D]* from memory
- Break them up into *1D* pieces for PE Array to perform matrix multiplications
- Reconstruct PE Array results back to *4D output features* and write into memory

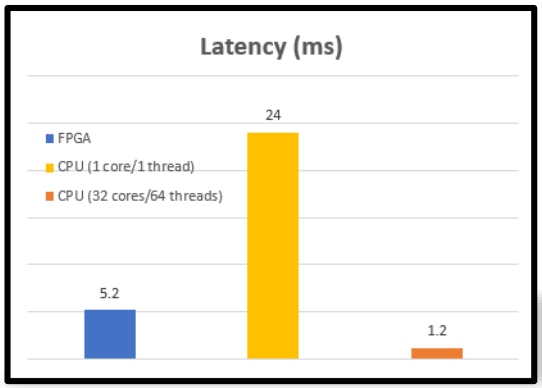
[1] Abdelfattah, M.S. et al., DLA: Compiler and FPGA Overlay for Neural Network Inference Acceleration. arXiv e-prints arXiv:1807.06434 (Jul 2018)

3DGAN: Initial Results (Single Conv3d Layer)

FPGA vs. CPU performance *for single 3D convolutional layer*

Bit precision	FPGA * Latency	CPU (1 core/1 thread) ** Latency	CPU (32 core/64 thread) ** Latency	FPGA speedup vs. 1 core CPU
FP16	***5.2 ms/layer***	**24 ms/layer**	**1.2 ms/layer**	***4.6x***





* Arria 10 at 20 nm process
** Intel Xeon Gold 6130 CPU at 14 nm process

Conclusion:

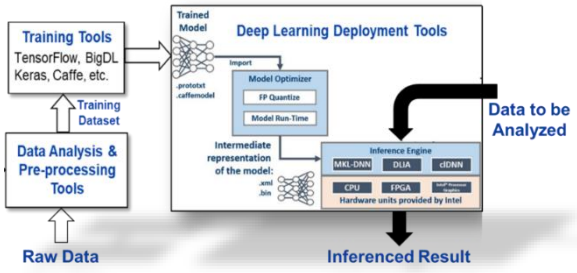
- Successfully inference *3D conv layer* on FPGA with customized DLA

Going forward:

- *Integrate* the customized DLA OpenCL code with *Intel OpenVINO toolkit* to performance complete 3DGAN inference
- Perform *design space exploration* for further optimisation

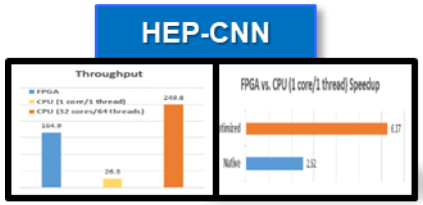
Summary & Conclusions

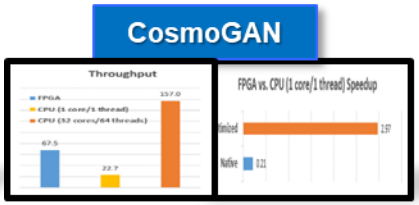
- Heterogeneous computing for deep learning

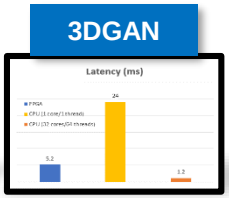


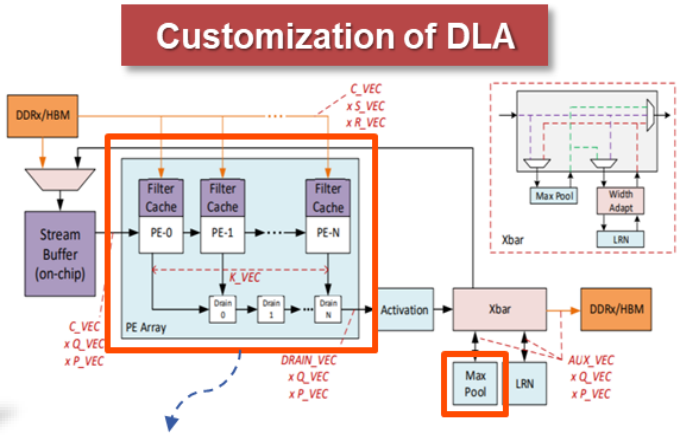
- Collaboration with CERN openlab & NERSC on scientifically relevant DNN
- SHREC: Focused on FPGA-acceleration of inference stage

- Exploration of FPGA-based platforms & tools
  - Intel PAC Arria 10 card; OpenVINO; DLA design suite
  - Explore use and improvement of state-of-art tools









Conclusions & Going Forward

- Continue to explore & improve FPGA-based DNN platforms & tools
  - Scale up multiple FPGAs for faster inference
  - Explore FPGA+ for efficient DNN model training
- Appropriate use of FPGA-based DNN platforms
  - Compare FPGA-based platform vs. CPU, GPU, & other emerging devices (energy, size, weight, cost, etc.)
  - Determine appropriate missions for FPGA-based systems

QUESTIONS