(Fast) Machine Learning for Neutrinos

Georgia Karagiorgi, Columbia University

Fast Machine Learning
IRIS-HEP Blueprint Workshop

Sep. 10-13, 2019 at Fermi National Lab

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

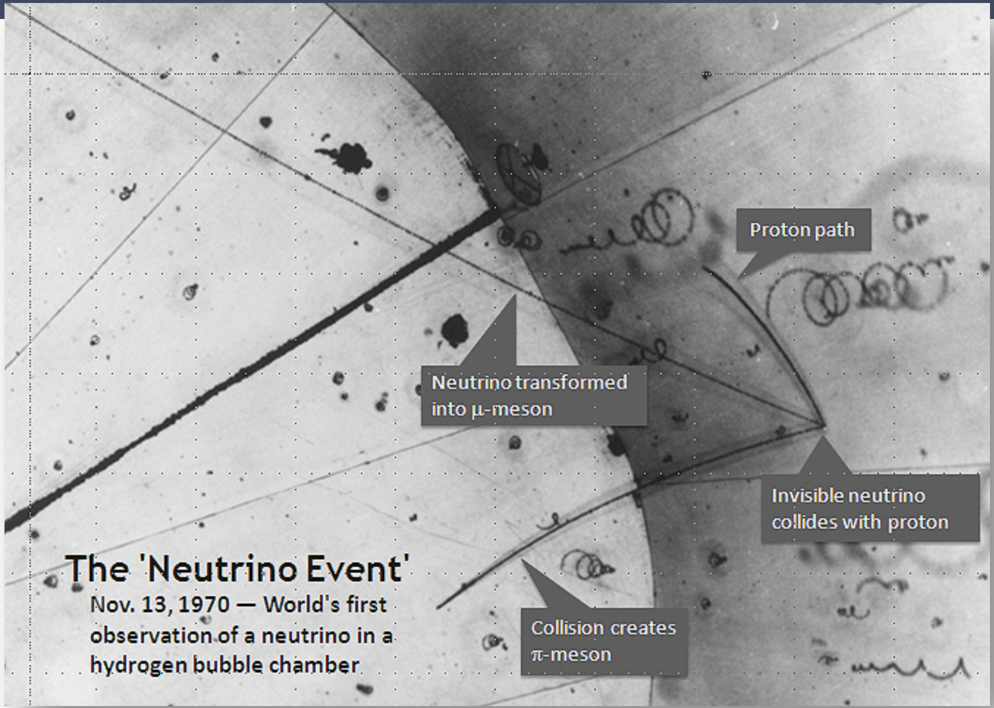
(Fast) Machine Learning for Neutrinos

This talk:

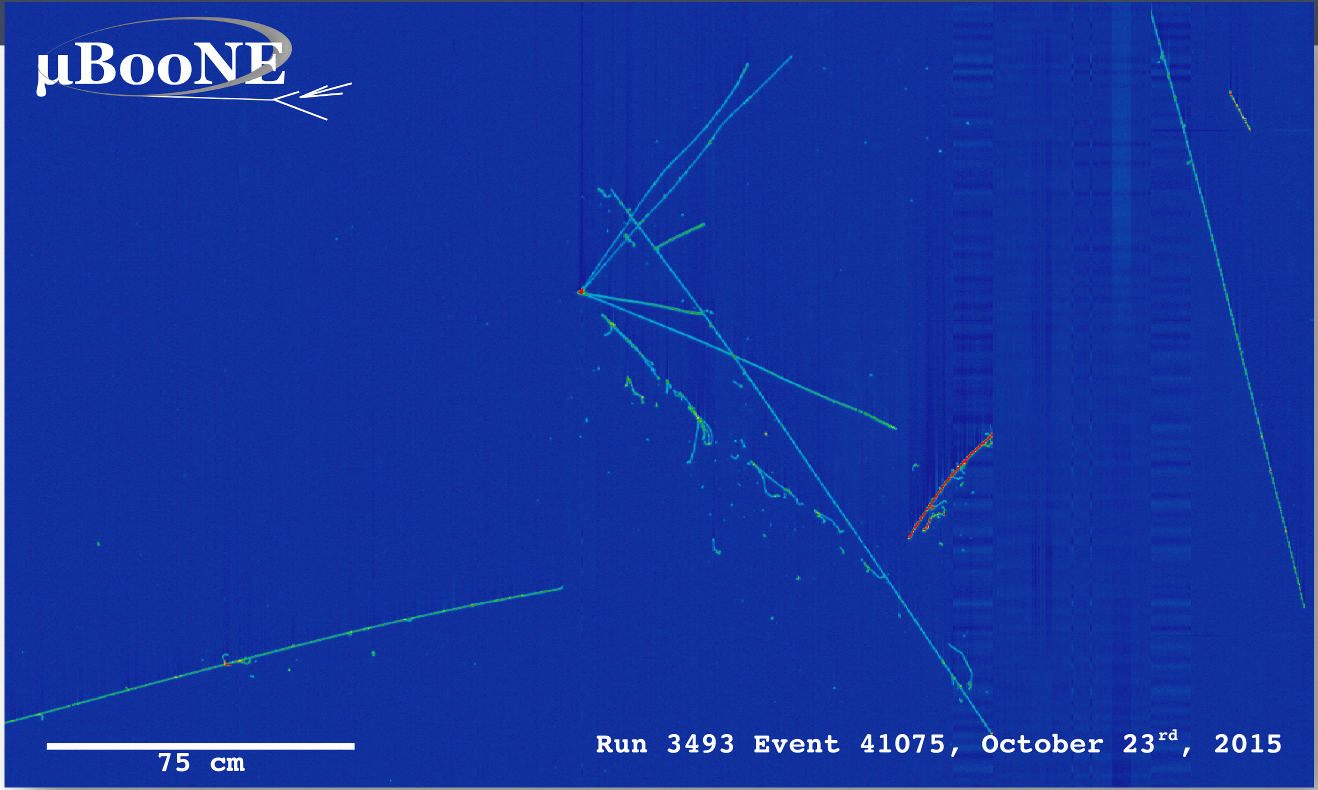
- Motivate need for *fast* ML for neutrinos
- Deep Underground Neutrino Experiment (DUNE) as application case study,

  based on collaborative work with Y. Jwa, G. Di Guglielmo, and L. Carloni, Columbia U.

Neutrino detection 49 years ago…



...and neutrino detection today!



One of the first neutrino events observed in the MicroBooNE Liquid Argon Time Projection Chamber

- Different detector technology
- Similar images
- Automated, often continuous readout
  → lots and lots of data!

Machine Learning for Neutrinos

The data outputs of many neutrino detectors can be viewed as **images**, which invites the application of **computer-vision techniques** for data analysis, and event identification.

nature
International journal of science

Review Article | Published: 01 August 2018

Machine learning at the energy and intensity frontiers of particle physics

Alexander Radovic, Mike Williams, David Rousseau, Michael Kagan, Daniele Bonacorsi, Alexander Himmel, Adam Aurisano, Kazuhiro Terao & Taritree Wongjirad

Nature **560**, 41–48 (2018) | Download Citation

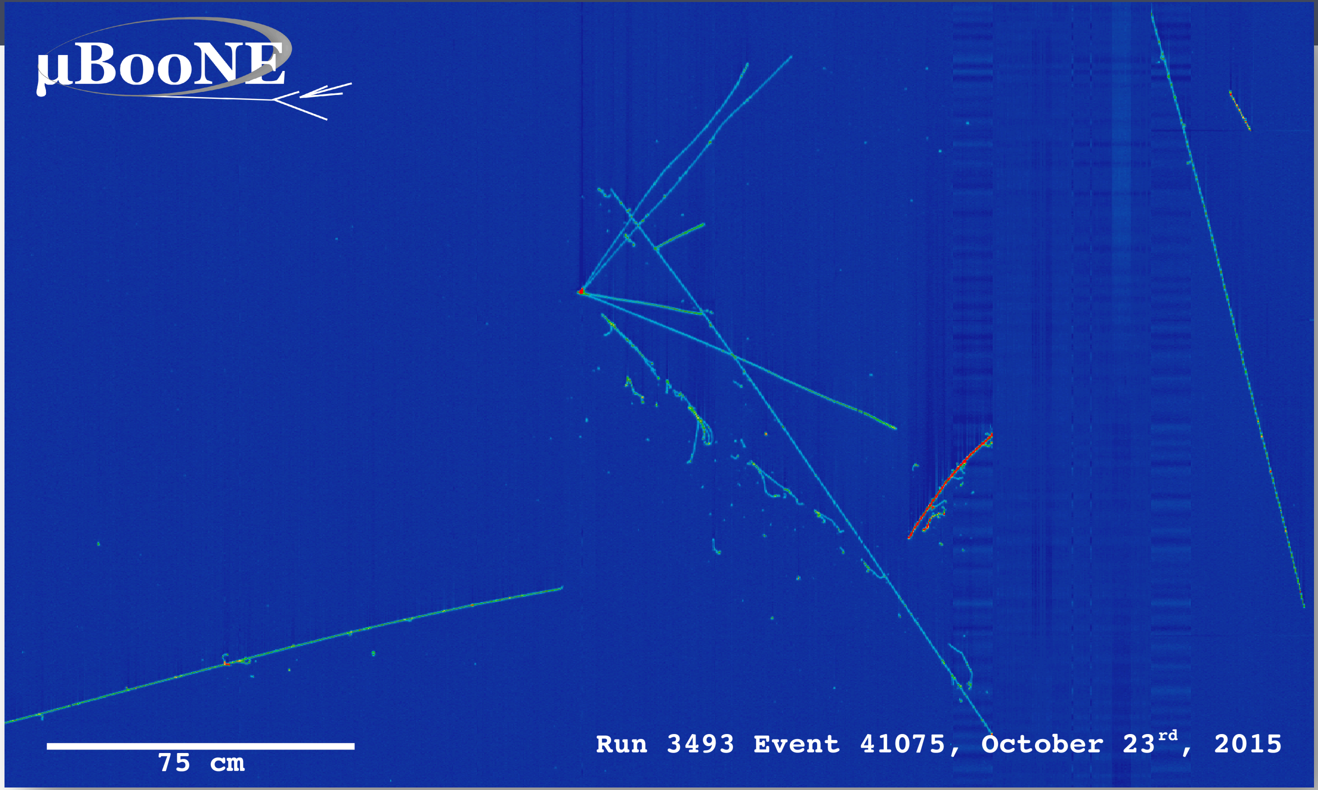
"…by taking advantage of **accelerated computing** on GPUs, these CNNs can run much faster than the conventional algorithms used by previous neutrino experiments. This makes them **ideally suited to the task of real-time image classification and object detection**."

“(Fast) Machine Learning for Neutrinos”

or

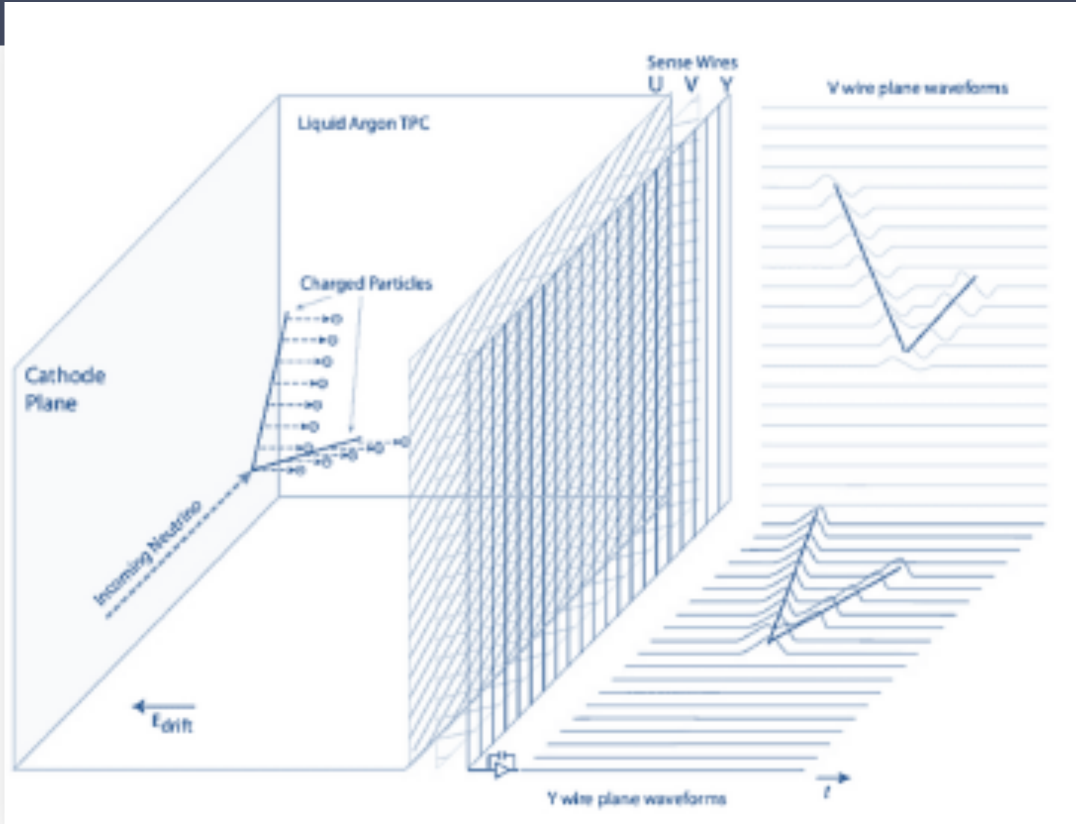
“Acceleration of CNNs for real-time inference”

Special case: LArTPC Technology



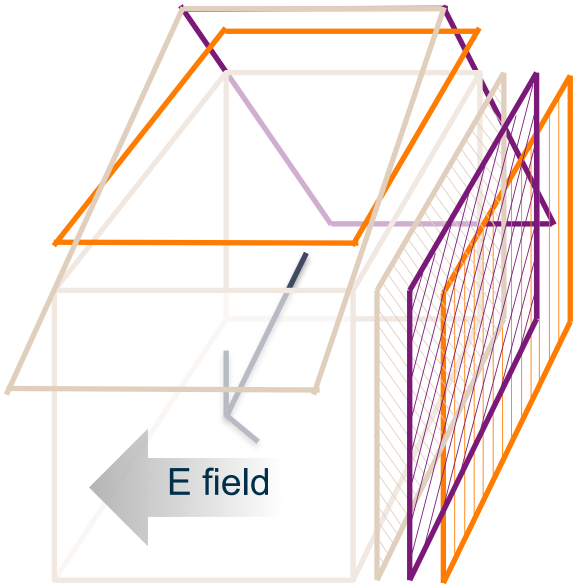
One of the first neutrino events observed in the MicroBooNE **Liquid Argon Time Projection Chamber**

LArTPC operating principles

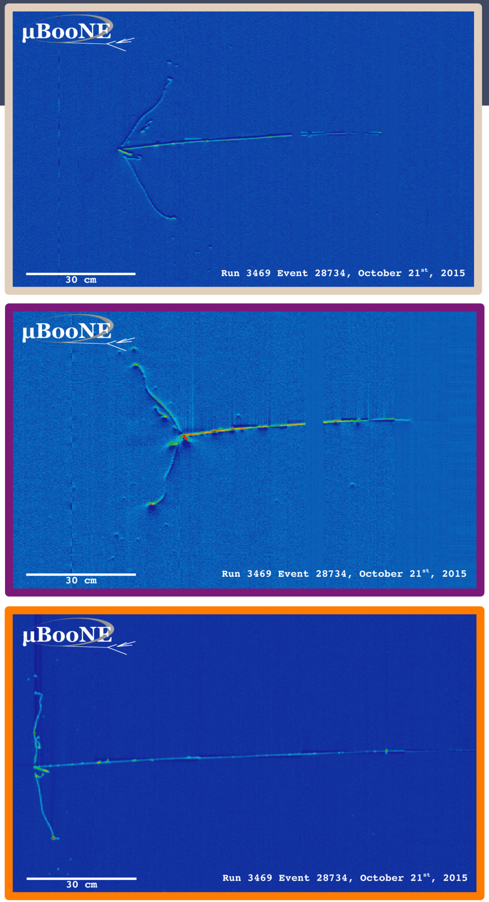


- particle-imaging detector
- stereoscopic "video capture" of activity within detector volume with sub-mm spatial resolution
- high-resolution "video" streams:
  - O(10) megapixel per O(1) ms for a volume the size of ~a small room
  - Usually 12-bit resolution

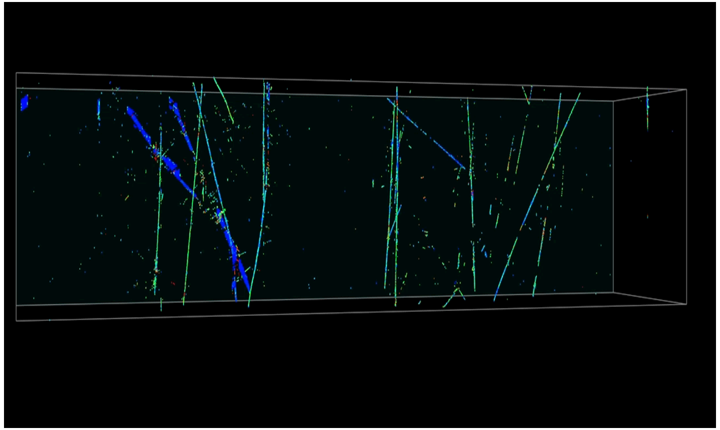
MicroBooNE



"school bus"-sized detector



Three charge sensing planes, provide three 2D projected views of detector volume

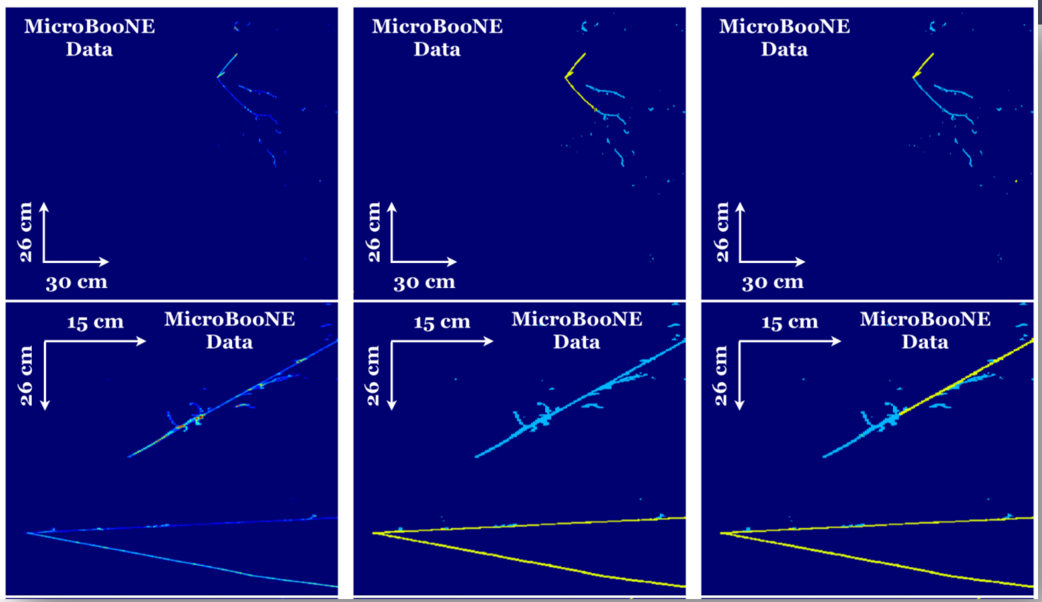


625 frames per plane per second, ~2700 x 3200 = 8.6M pixels each

Machine Learning @ MicroBooNE

MicroBooNE is pioneering machine learning applications for LArTPCs – offline analysis.

[1]



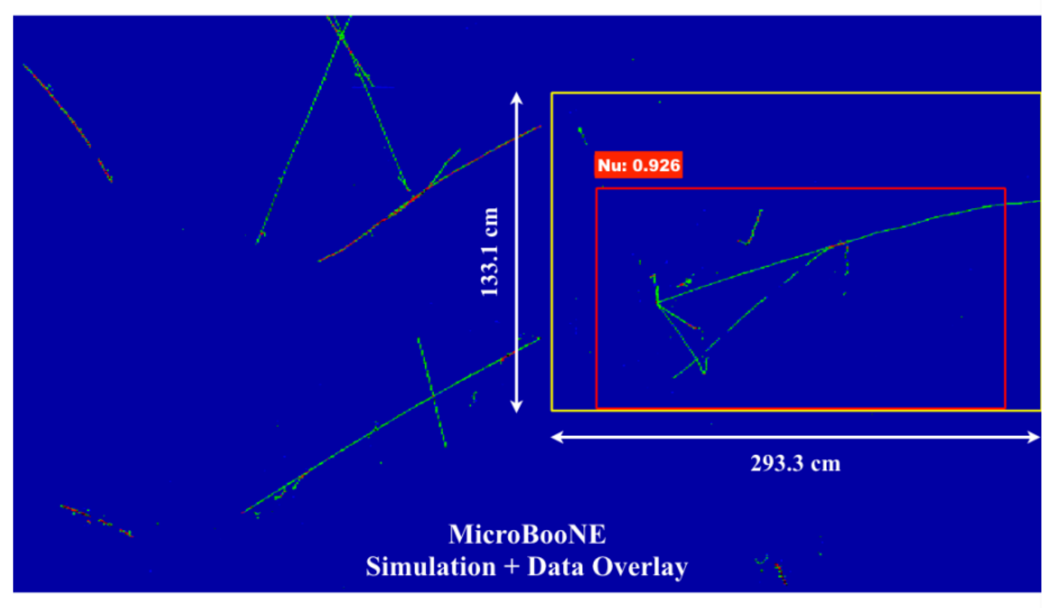
See, e.g.:

[1] "Deep neural network for pixel-level electromagnetic particle identification in the MicroBooNE liquid argon time projection chamber," Phys. Rev. D99 (2019) No. 9, 092001.

[2] "Convolutional Neural Networks Applied to Neutrino Events in a Liquid Argon Time Projection Chamber," JINST 12 (2017) No. 03, P03011.

Machine Learning @ MicroBooNE

MicroBooNE is pioneering machine learning applications for LArTPCs – offline analysis.



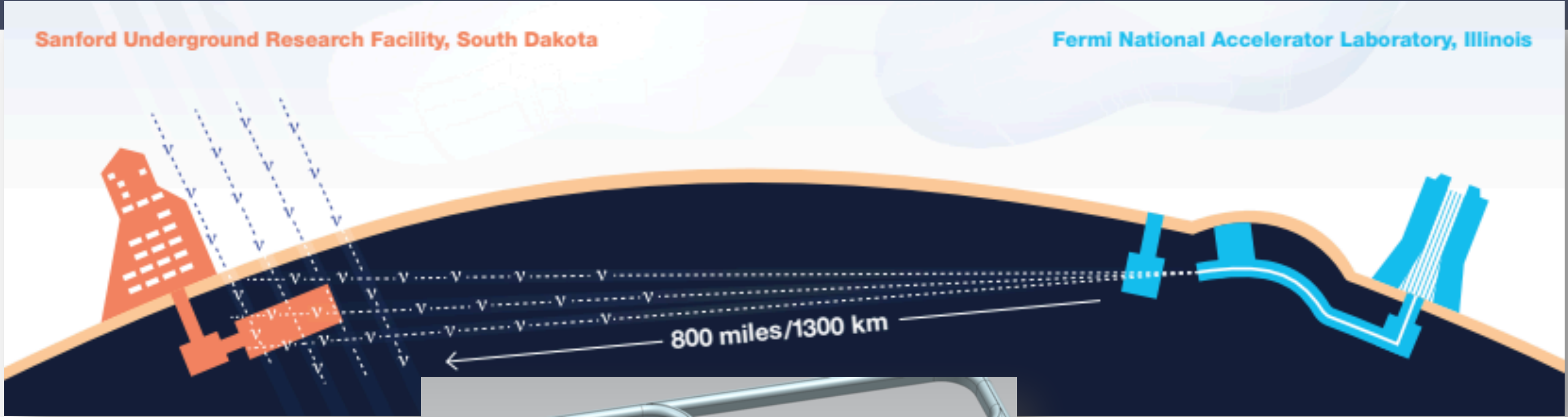
CNNs can be trained to do particle classification, particle and neutrino detection, and neutrino event identification [2].

See, e.g.:

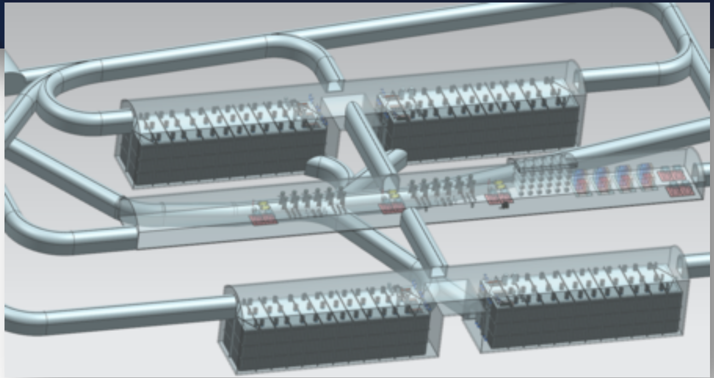
[1] "Deep neural network for pixel-level electromagnetic particle identification in the MicroBooNE liquid argon time projection chamber," Phys. Rev. D99 (2019) No. 9, 092001.

[2] "Convolutional Neural Networks Applied to Neutrino Events in a Liquid Argon Time Projection Chamber," JINST 12 (2017) No. 03, P03011.

Application Case: **DUNE** (~500x MicroBooNE!)



4 neutrino detector modules
1 mile underground



Primary physics goals of DUNE:

- Leptonic CP violation and neutrino mass hierarchy
- Off-beam rare event searches

DUNE rare event searches

Interaction Type	Event Type	Expected Rate
Rare off-beam events		
Proton decay	High Energy (HE)	< 1 / year
Neutron-antineutron oscillation	High Energy (HE)	< 1 / year
Galactic supernova burst[a]	Low Energy (LE)	< 1 / year
Other off-beam events		
Atmospheric neutrinos	High Energy (HE)	1200 / year
Cosmic ray muons	High Energy (HE)	1.3×10^6 / year

neutrinos from nearby supernova bursts

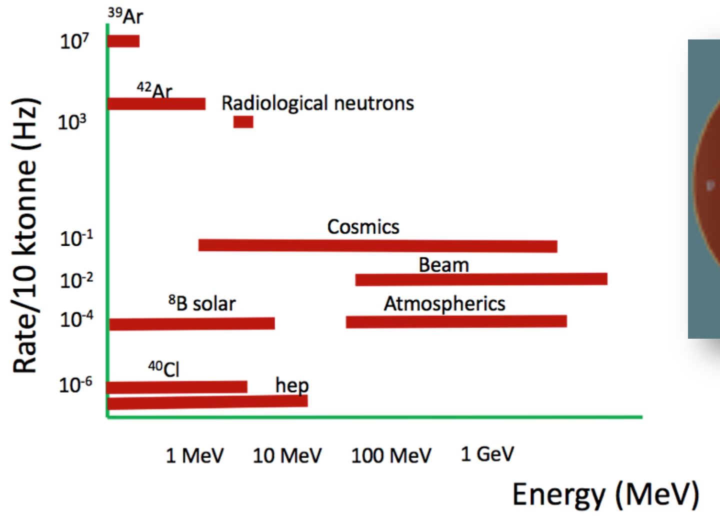
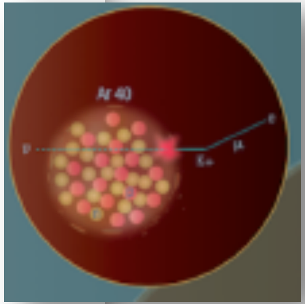


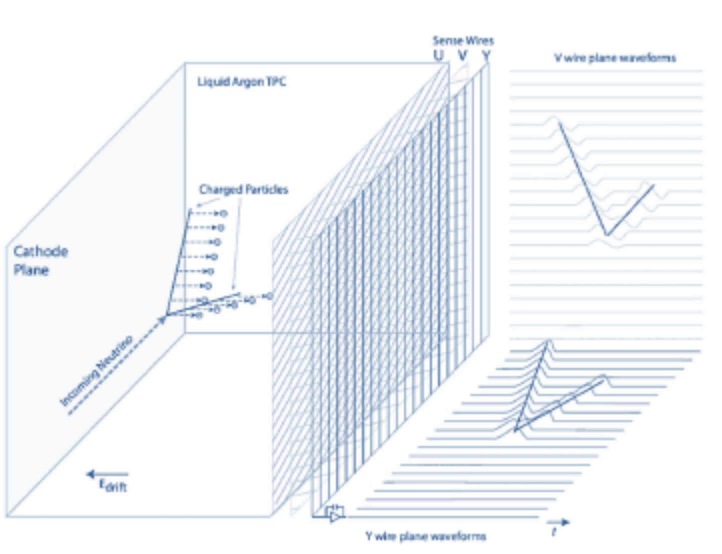
Figure 1.2: Expected physics-related activity rates in a single 10 kt module.

[DUNE TDR]



proton decay, baryon number violation

DUNE's Data (Selection) Challenge



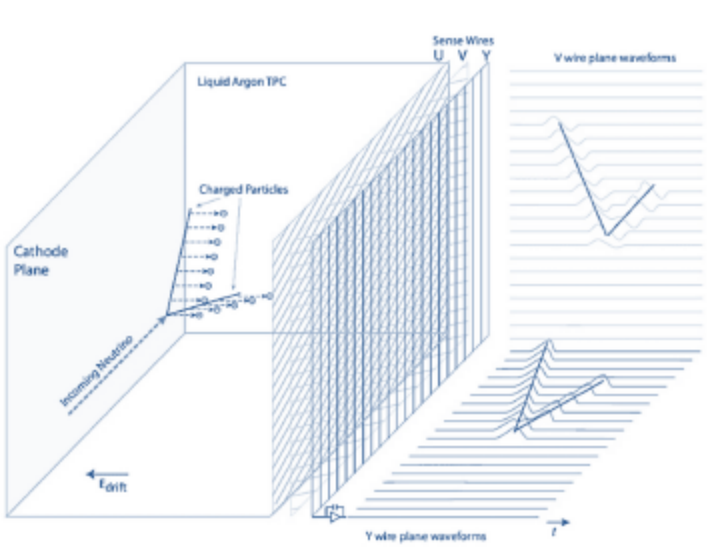
High-resolution "video" streams:

- from up to 4x150 independent detector volumes
- 11.5 megapixel frames (all 3 planes) per 2.25ms
- 12-bit resolution

A total of **~40 terabits per second**

100% live time
continuous operation for more than a decade

DUNE's Data (Selection) Challenge



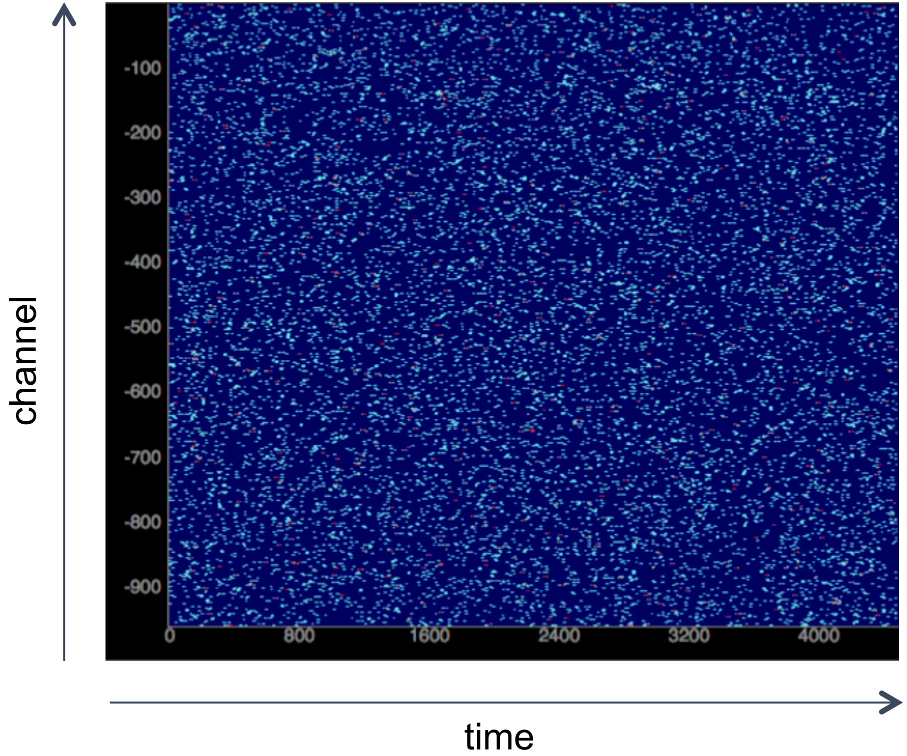
Rare event searches require **data-driven self-triggering**
Data selection system must digest full data stream and identify signatures of interest.

Requires:

- **Fast and efficient data processing** for trigger decision (2.25 ms)
- **Large buffering** to hold data while decision is being made (a full drift for DUNE SP is 2.6 GB)
- Orders of magnitude more buffering and processing for a **supernova burst trigger**, which looks for correlated signatures in O(10) seconds!

DUNE signatures

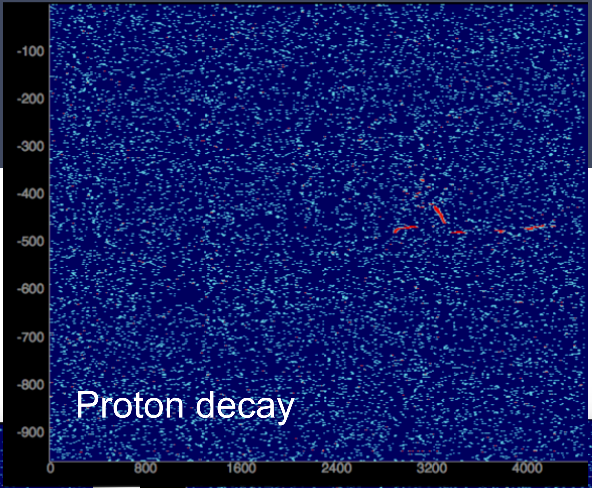
Deep Underground = "quiet" environment

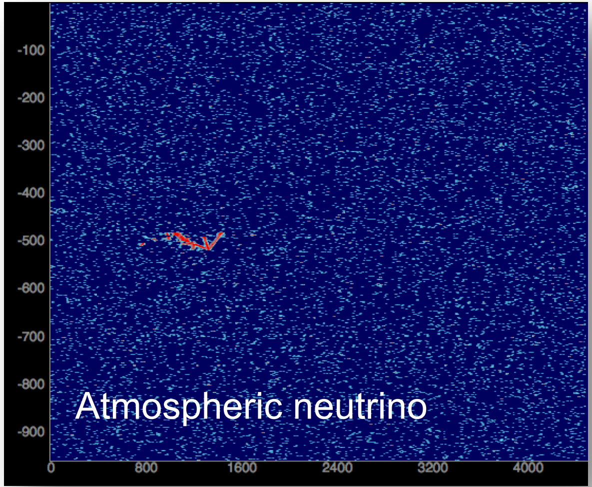


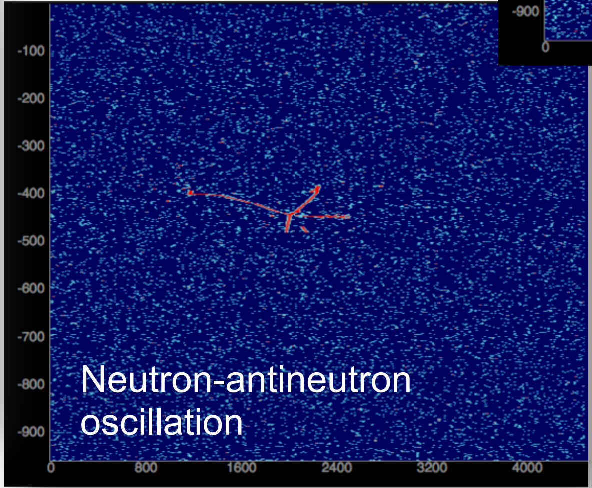
Single frame from high-resolution video: One of three 2D views from one of hundreds of cells in the detector

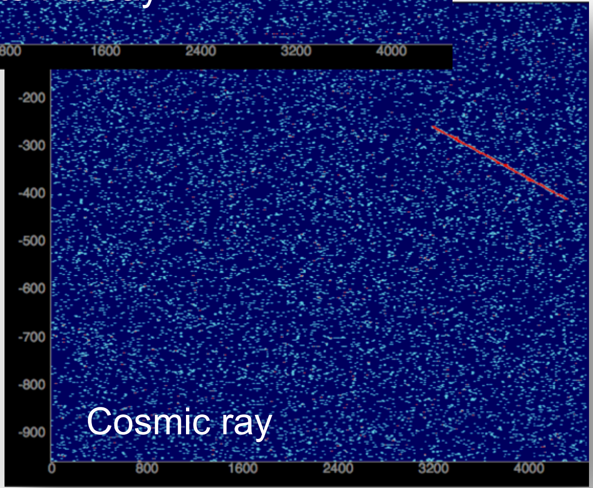
"Static" is noise and small energy deposits from radiological impurities in the detector

DUNE signatures: HE events





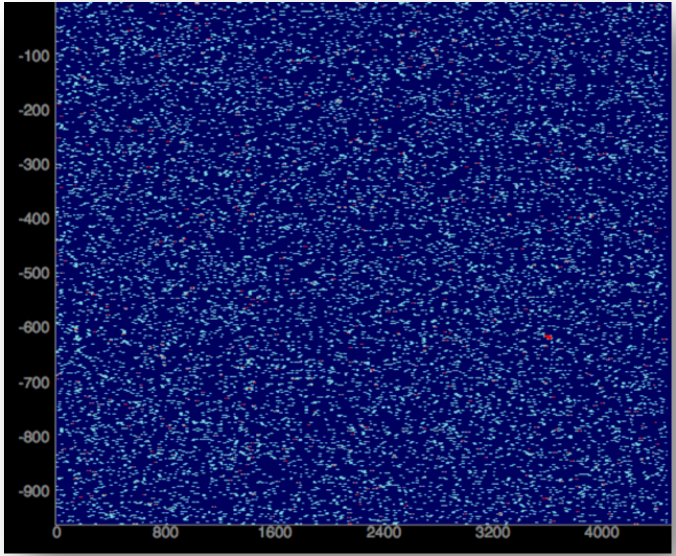


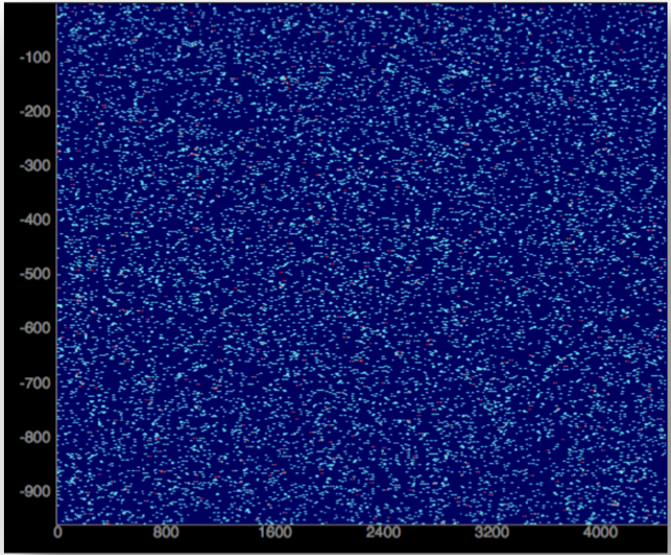


[simulation]

DUNE signatures: HE events

Special challenge: **neutrinos from supernova core collapse**
Very low energy and small (in extent) topology, similar to radiological background activity in the detector



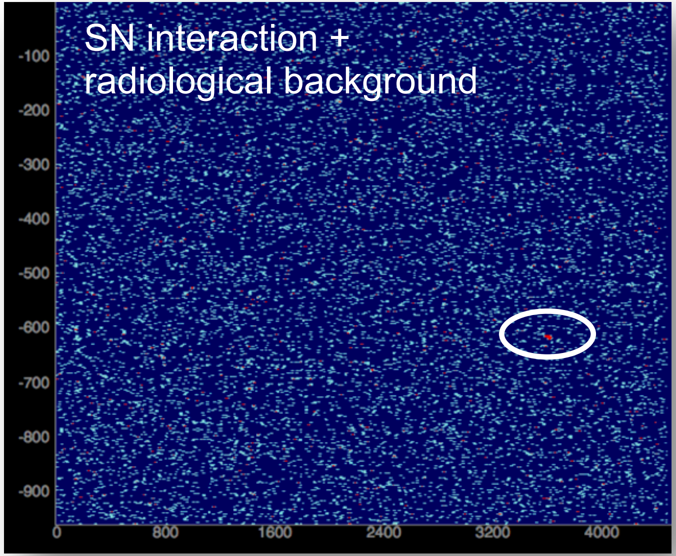


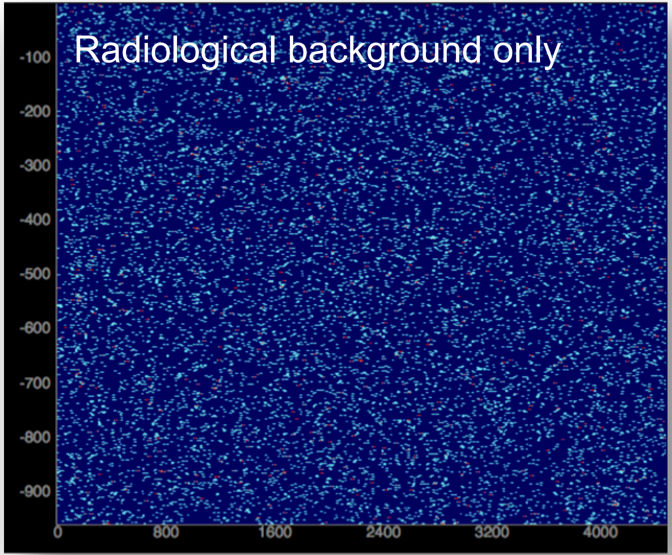
Need $O(10^4)$ background suppression, while maintaining high efficiency to a frame containing a supernova neutrino interaction

[simulation]

DUNE signatures: HE events

Special challenge: **neutrinos from supernova core collapse**
Very low energy and small (in extent) topology, similar to radiological background activity in the detector





Need $O(10^4)$ overall background suppression, while maintaining high efficiency to a frame containing a supernova neutrino interaction

[simulation]

ML-based data selection (trigger) in DUNE

Raw LArTPC data format ideally suited for image analysis!

E.g., **Convolutional Neural Networks (CNNs)*** could be applied for real-time image classification, using hardware acceleration (FPGA), or online in GPU or CPU.

**translation-invariant feature extraction*

A data selection (trigger) scheme could, e.g.,

- Work with only one projection (2D), preferably collection plane
- Down-sample and resize image if/as needed
- Classify via CNN as whether it contains an interaction of interest
- For supernova interaction-containing frames, consider them in coincidence with frames across the entire the detector over a 10 second period (higher-level data selection decision)

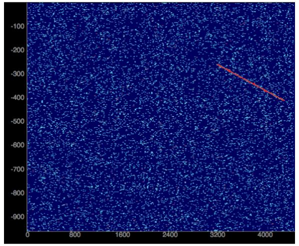
Case study 1: Full stream, frame-by-frame classification

Starting with raw LArTPC images, how well can a CNN classify data?

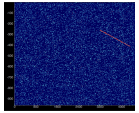
Consider three classes:

background (NB)/**supernova-like low energy activity (LE)**/**high-energy activity (HE)**

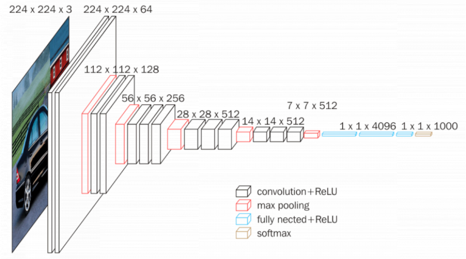
Train CNN (vgg16b) for classification on GPU, and test (on any given platform)



Raw image input (4488x480)



Down-sampling, resizing (600x600)



CNN classification

NB

LE

HE

Selection (e.g., by highest class score)

Case study 1: Full stream, frame-by-frame classification

Results obtained with vgg16b network:
(600x600 input image)

Sample	Train Size	Test Size	Accuracy (%)			Inference Time (ms)
			ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	
NB	51,100	99,000	91.45	8.49	0.06	
LE	44,900	29,800	3.17	96.83	0	27.7±8.6
HE	52,828	67,178	6.03	3.48	90.48	

- **HE events correctly classified with > 90% efficiency**
- **LE events correctly classified with > 95% efficiency**
- **~8.5% mis-classification rate of background frames as containing LE events, but could be further reduced with a higher-level selection**

Still, **inference time is ~28ms for a 2.25ms image** → x10 off what might be a reasonable goal even with a 200-fold parallelization (200 images/2.25 ms/module)

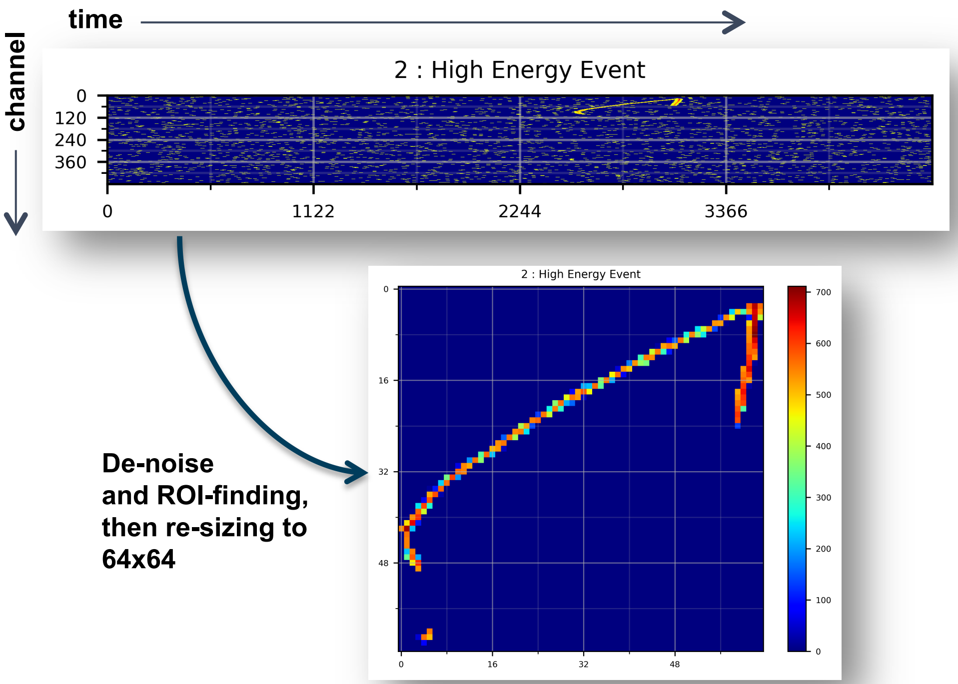
Case study 2: Down-selection, frame-by-frame classification

Long inference time: A consequence of both input image size and network architecture!

Smaller input images?

Pre-processing of collection plane images for

1. De-noising
2. Region of Interest (ROI) finding
3. Re-sizing to 64x64



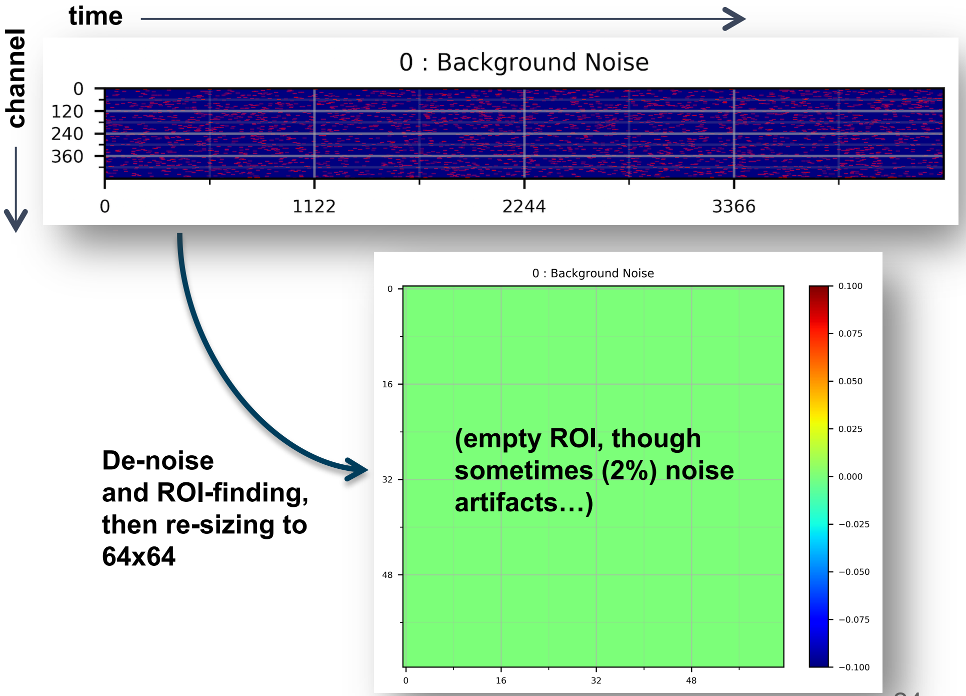
Case study 2: Down-selection, frame-by-frame classification

Long inference time: A consequence of both input image size and network architecture!

Smaller input images?

Pre-processing of collection plane images for

1. De-noising
2. Region of Interest (ROI) finding
3. Re-sizing to 64x64



Case study 2: Down-selection, frame-by-frame classification

Results obtained with vgg16b network:
(64x64 input image)

Sample	Train Size	Test Size	Accuracy (%)			Inference Time (ms)
			ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	
NB	12,023	4,027	99.65	0.35	0	
NB*	12,023	293	79.9	19.8	0.34	5.0±0.3
LE	12,050	3,970	3.78	95.04	1.18	
HE	10,137	3,417	2.99	6.88	90.14	

- **HE events correctly classified with > 90% efficiency**
- **LE events correctly classified with > 95% efficiency**
- **~0.4% mis-classification rate of background frames as containing LE events, and could be further reduced with a higher-level selection**

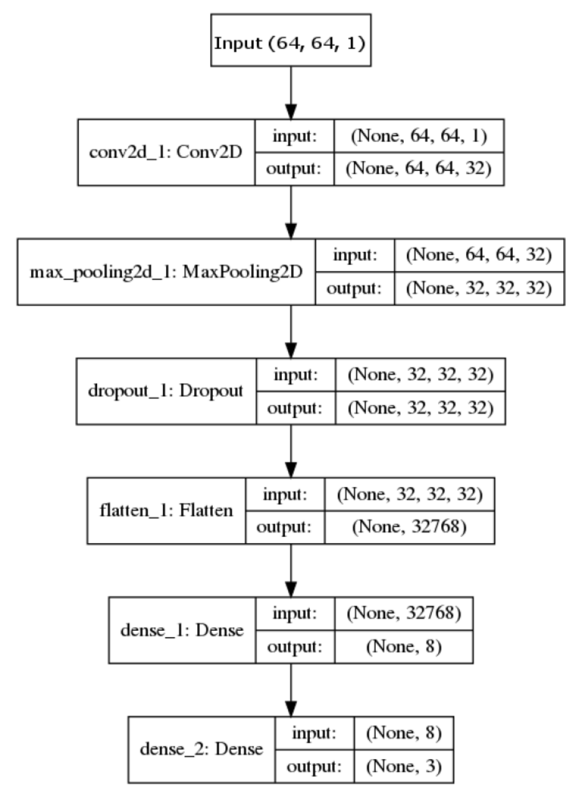
Inference time is ~5ms for a 2.25ms image → x2 off what might be a reasonable goal, assuming a 200-fold parallelization (200 images/2.25 ms/module)

*Added advantage: 98% of NB ROIs are empty, so inference stage could be skipped, gaining x50 in classification rate!

Case study 2: Down-selection, frame-by-frame classification

Smaller input images help!
What about smaller networks?

Consider smaller CNN, “CNN_s”



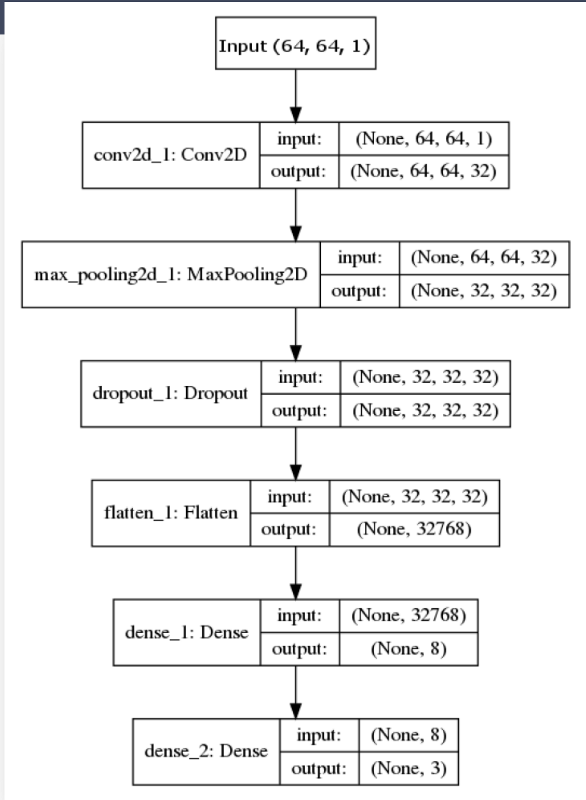
Case study 2: Down-selection, frame-by-frame classification

Smaller input images help!
What about smaller networks?

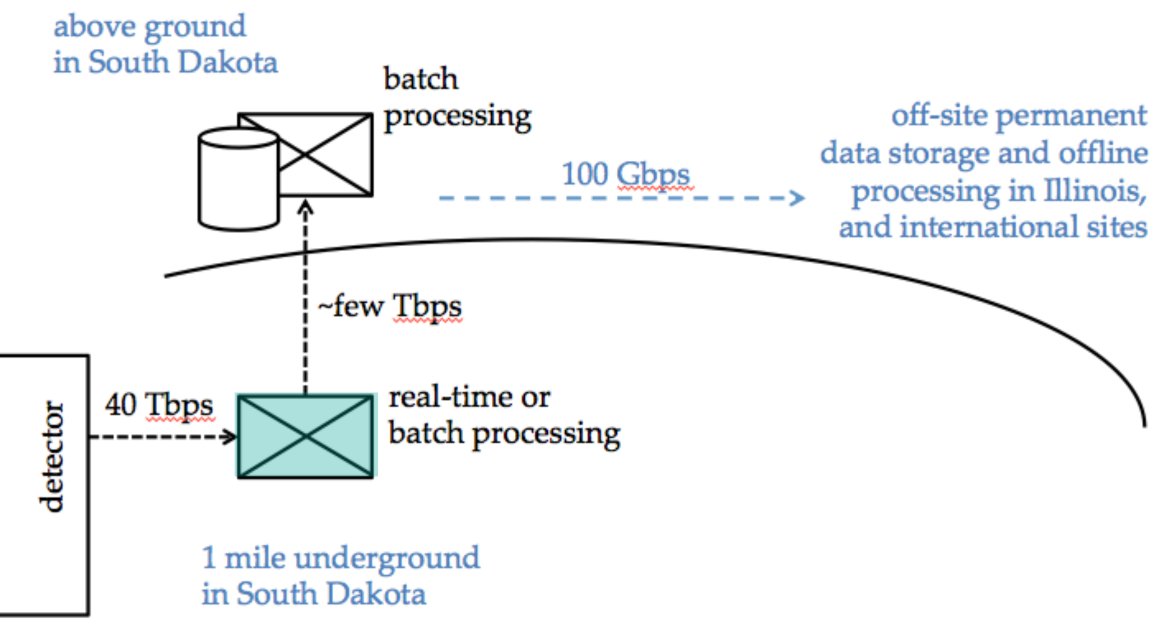
Consider smaller CNN, “CNN_s”

Sample	Train Size	Test Size	Accuracy (%)			Inference Time (ms)
			ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	
NB	12,023	4,027	99.53	0.47	0.12	
LE	12,050	3,970	4.01	94.48	1.51	1.6±0.1
HE	10,137	3,417	3.63	6.15	90.22	

Comparable efficiencies as with vgg16b (64x64).
>3x reduction in inference time → online classification is possible for 200-fold parallelization



Online vs. real-time implementation: Considerations for DUNE



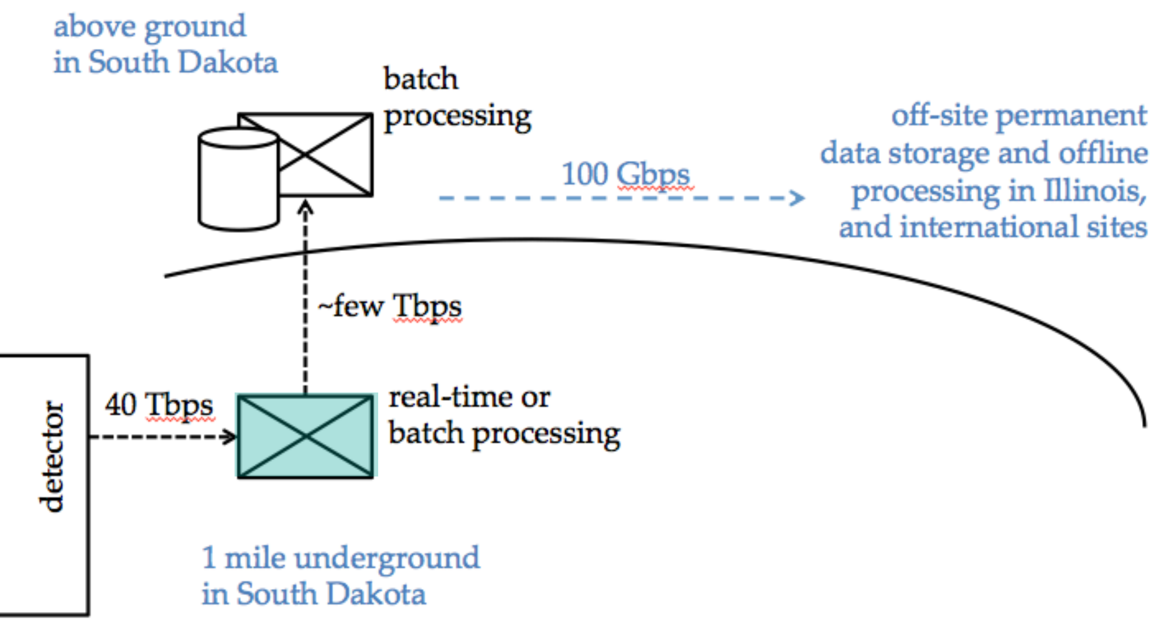
[DUNE TDR, in preparation]

Online implementation, e.g. in GPU:

GPU advantages: High computational density, level of programmability, data-parallelism, flexibility

Downsides: Long-term operation reliability and power utilization, especially underground

Online vs. real-time implementation: Considerations for DUNE



[DUNE TDR, in preparation]

Real-time implementation, in FPGA:

FPGA: power-aware platform for CNN acceleration
Concerns for application: resource constrains given network size, input image size are large

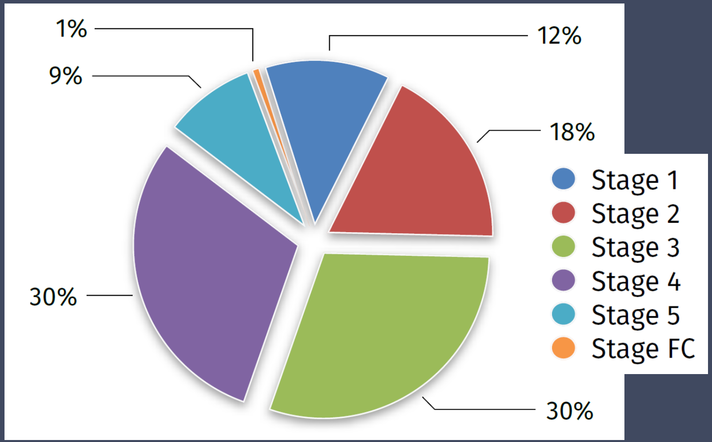
FPGA advantages: power-efficiency enables upstream implementation, deterministic

Downsides: Ease of programmability, resource constraints

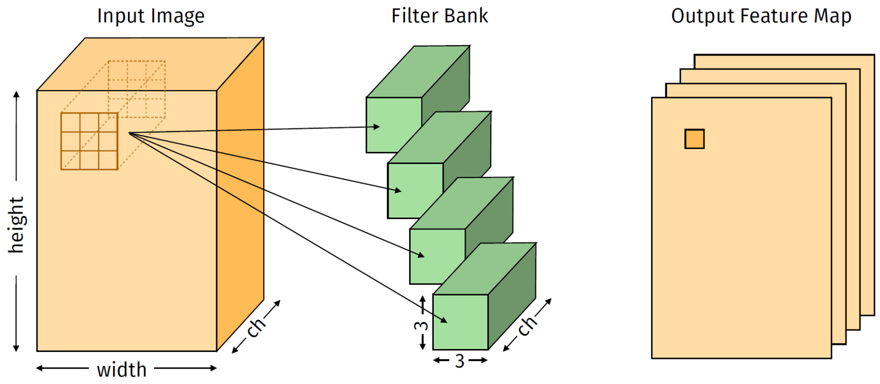
Convolutional Layers

Convolutional layers are the most computational intensive part in CNNs

$$Y_{k,i,j} = \sum_{c=0}^{C-1} X_c * W_{k,c} + B_k = \left[\sum_{c=0}^{C-1} \sum_{x=0}^{F-1} \sum_{y=0}^{F-1} X_{c,i+x-\frac{F}{2},j+y-\frac{F}{2}} \cdot W_{k,c,x,y} \right] + B_k$$



Distribution of floating-point operations per stage in vgg16b



Accelerating CNN's for real-time inference

- **Exploring CNN acceleration** using a customizable and efficient hardware accelerator design for the various layers of CNN, utilizing High Level Synthesis (HLS)-based design flow

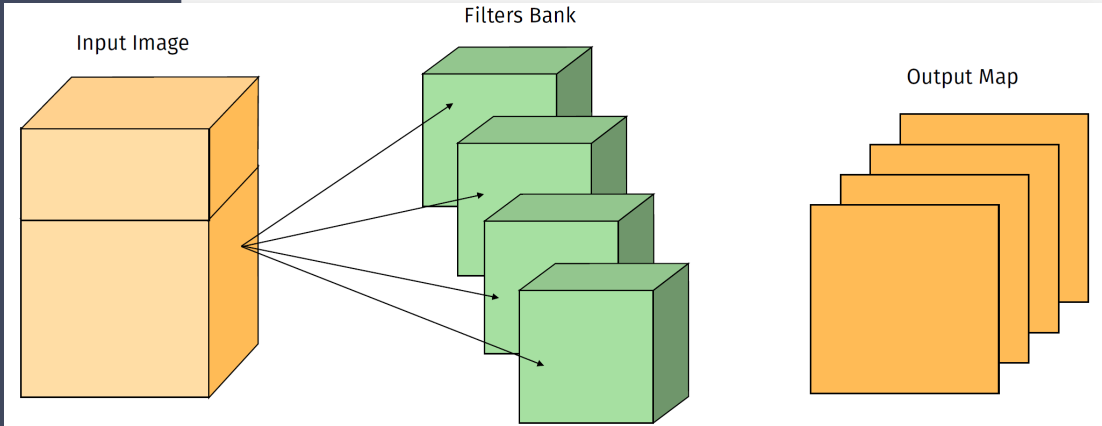
- Flexibility for optimization (processing time, efficiency, resource utilization)

Xilinx ZynqMP UltraScale+ XCZU9EG

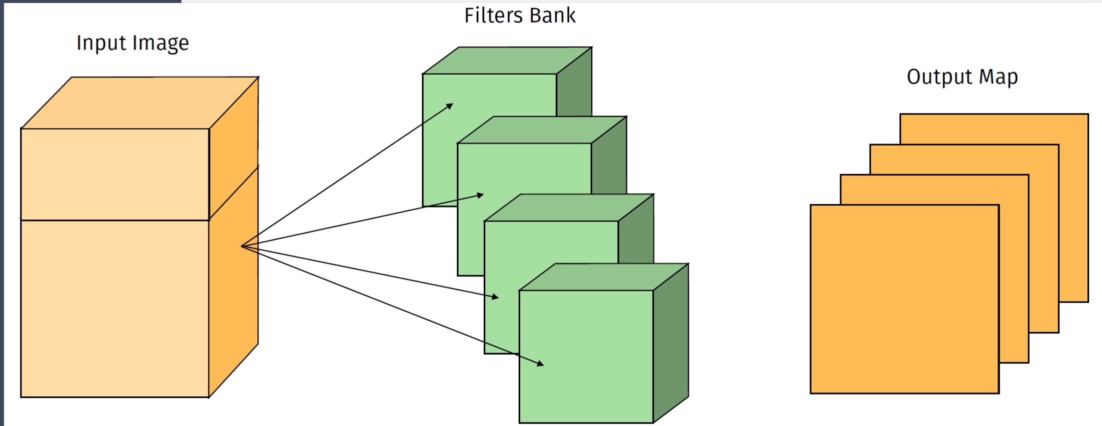
[Studies and results forthcoming in IEEE proceedings]

Balance of Computation and Communication

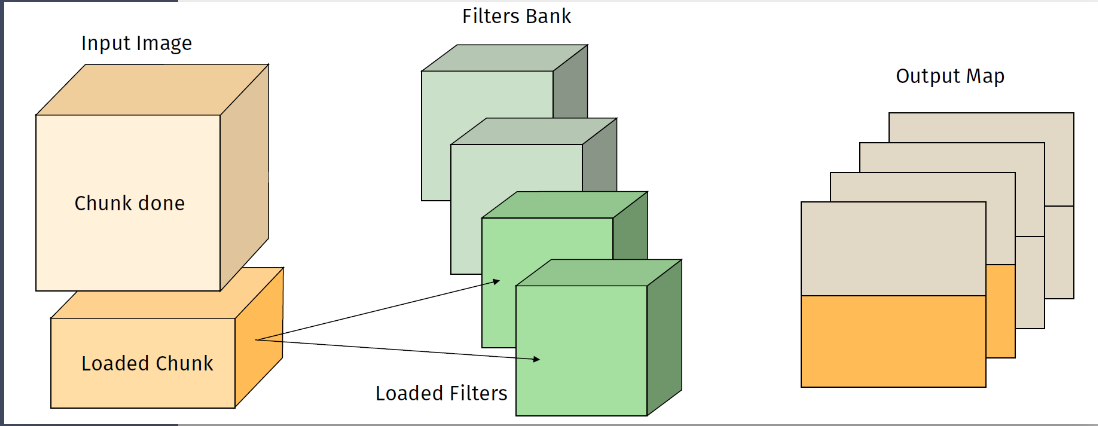
Carefully design the algorithm to reuse data as much as possible, thus reducing expensive memory transfers from and to off-chip DRAM



Balance of Computation and Communication

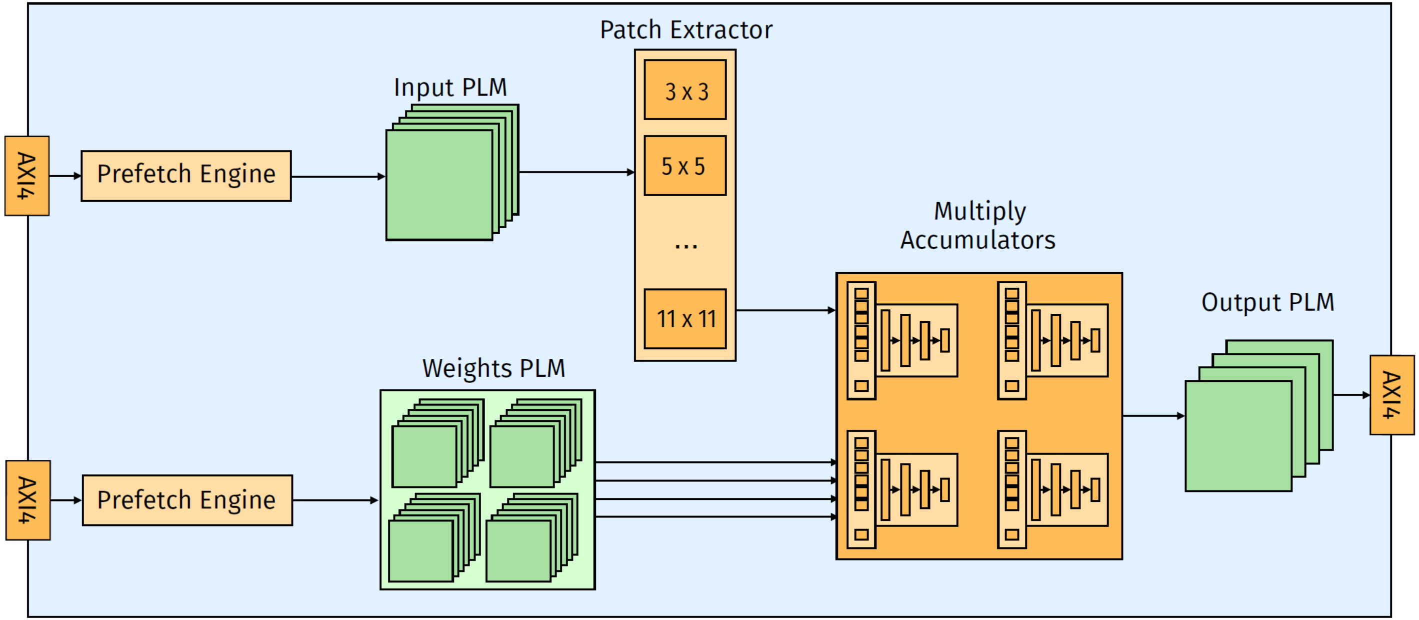


Tailoring Private Local Memory



Both inputs and weights are divided in chucks and the computation is done only with the on-chip copy of the data

Accelerator structure: Highly-configurable accelerator



Allows exploration of large (in size) networks (e.g. vgg16)

Performance

Implemented customized CNN_s on
(1) **FPGA accelerator** and
(2) in C on **ARM Cortex-A53 CPU as a reference**
for performance and power analysis.

Platform	Model	Time (s)	Power (W)	Energy Efficiency (img/s/W)
ARM C-A53	CNN_s	0.0855	2.871	4.074
FPGA	CNN_s	0.0511	1.110	17.630

Performance when leveraging FPGA for acceleration*:

1.7x **average speedup** for inference
2.6x **more power efficient** w.r.t software implementation on ARM Cortex A53

**Note:* Accelerator designed for optimization of even larger networks.
Much higher energy efficiency is achieved for larger networks than CNN_s,
but resource allocation is an issue → more communication → longer latency

Summary

Neutrino detection involves large, uniform detector volumes, often sparsely occupied with activity, where neutrino interactions can happen anywhere within the detector volume.

→ **Ideal for applications of image analysis, CNNs, deep CNNs…**

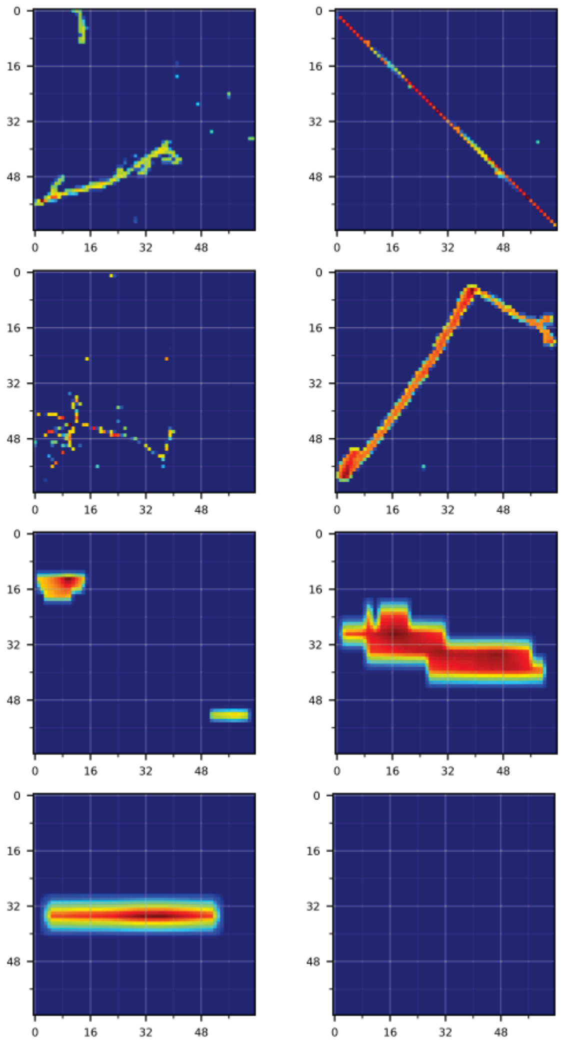
In recent years, the **size and resolution of data** from neutrino detectors has been growing drastically. Current technologies, most notably **LArTPC's (DUNE)** are faced with **major data-processing challenges**.

Fast ML needs are increasing for **offline** analysis.
With sufficient acceleration, **real-time** ML could prove advantageous for long-term operating detectors.

Ongoing efforts demonstrate **viability of CNN-based real-time/online triggering in DUNE** and invite further exploration of such application.

Thank you!
Questions?

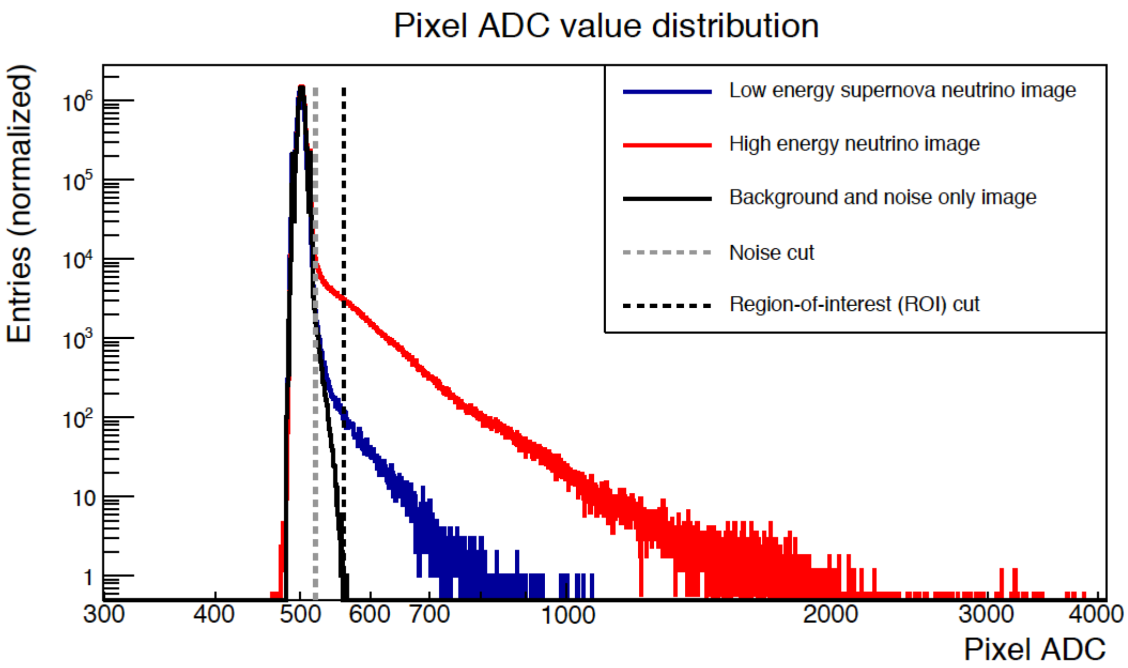
COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK



HE ROIs

LE ROIs

Noise ROIs



Noise removal cut: dashed gray line, at 520 ADC
ROI cut: dashed black line, at 560 ADC

GPU INFERENCE RESULTS USING METHOD 2, OBTAINED WITH THE MLP_1 NETWORK (TRAINING FOR 65 EPOCHS AND LEARNING RATE SET TO 2×10^{-4}).

Sample	Train Size	Test Size	Accuracy (%)			Inference Time (ms)
			ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	
NB	12,023	4,027	99.50	0.45	0.05	
LE	12,050	3,970	4.48	89.70	5.82	1.0±0.08
HE	10,137	3,417	7.29	13.08	79.63	

GPU INFERENCE RESULTS USING METHOD 2, OBTAINED WITH THE RESNET50 NETWORK (TRAINING FOR 30 EPOCHS AND LEARNING RATE SET TO 10^{-5}).

Sample	Train Size	Test Size	Accuracy (%)			Inference Time (ms)
			ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	
NB	12,023	4,027	99.28	0.55	0.17	
LE	12,050	3,970	3.55	88.89	7.56	15.3±1.2
HE	10,137	3,417	2.84	15.13	82.03	

Alternate classification scheme:

NB cut	Accuracy (%)						
	ϵ_{NB}	ϵ_{LE}	ϵ_{HE}	$\epsilon_{HE:nnbar}$	$\epsilon_{HE:ndk}$	$\epsilon_{HE:atm}$	$\epsilon_{HE:cosmic}$
0.1	0.73	88.18	96.12	99.98	99.29	92.24	92.57
0.01	0.14	83.27	95.68	99.98	99.18	91.01	92.46
0.001	0.033	77.11	95.21	99.98	99.05	89.76	92.23
0.0001	0.011	69.74	94.61	99.97	98.74	88.39	91.71
0.00001	0.002	60.73	93.79	99.95	98.22	86.61	90.97

Xilinx ZynqMP UltraScale+ XCZU9EG

		CPU		Accelerator		
	MFLOP	**Time**	**GFLOPS**	**Time**	**GFLOPS**	**Speedup**
conv1_1	86.7	2.17	0.04	0.21	0.41	10.31
conv1_2	3699.4	51.05	0.07	3.66	1.01	13.95
conv2_1	1849.7	25.24	0.07	1.82	1.02	13.87
conv2_2	3699.4	51.27	0.07	3.46	1.07	14.82
conv3_1	1849.7	24.84	0.07	1.72	1.08	14.44
conv3_2	3699.4	50.85	0.07	3.37	1.10	15.09
conv3_3	3699.4	51.24	0.07	3.37	1.10	15.20
conv4_1	1849.7	25.23	0.07	1.68	1.10	15.02
conv4_2	3699.4	50.68	0.07	3.34	1.11	15.17
conv4_3	3699.4	50.68	0.07	3.34	1.11	15.17
conv5_1	924.8	12.46	0.07	0.84	1.10	14.83
conv5_2	924.8	12.46	0.07	0.84	1.10	14.83
conv5_3	924.8	12.46	0.07	0.84	1.10	14.83

15x average speedup
45x more power efficient
w.r.t software implementation
on ARM Cortex A53

	Time (s)	**Power** (W)	**PET** (Img/s/W)
ARM A53	420	3.2	0.001
Xilinx XCZU9EG FPGA	**28**	**0.8**	**0.045**