

Accelerating Deep Neural Networks for Real-time Data Selection for High-resolution Imaging Particle Detectors

Yeon-jae Jwa

Dept. of Physics

Columbia University

New York, NY, USA

jj2429@nevis.columbia.edu

Giuseppe Di Guglielmo

Dept. of Computer Science

Columbia University

New York, NY, USA

giuseppe@cs.columbia.edu

Luca P. Carloni

Dept. of Computer Science

Columbia University

New York, NY, USA

luca@cs.columbia.edu

Georgia Karagiorgi

Dept. of Physics

Columbia University

New York, NY, USA

georgia@nevis.columbia.edu

Abstract—This paper presents the custom implementation, optimization, and performance evaluation of convolutional neural networks on field programmable gate arrays, for the purposes of accelerating deep neural network inference on large, two-dimensional image inputs. The targeted application is that of data selection for high-resolution particle imaging detectors, and in particular liquid argon time projection chamber detectors, such as that employed by the future Deep Underground Neutrino Experiment. We motivate this particular application based on the excellent performance of deep neural networks on classifying simulated raw data from the DUNE LArTPC, combined with the need for power-efficient data processing in the case of remote, long-term, and limited-access operating detector conditions.

Index Terms—convolutional neural network, deep neural network, hardware acceleration, LArTPC, particle detector

I. INTRODUCTION

Liquid Argon Time Projection Chambers (LArTPCs) represent a particle detector technology that has been widely adopted in the field of high energy physics. Over the last two decades, LArTPCs have been increasingly used for studying neutrino-argon interactions with high calorimetric (energy) and spatial resolution. LArTPCs are already in use for a number of detectors; the most recent of these detectors, MicroBooNE [1] and ProtoDUNE [2], represent a significant R&D effort which is underway to scale up the LArTPC detector technology by up to two orders of magnitude in physical detector size. This phasing approach is necessary in order to realize the future Deep Underground Neutrino Experiment (DUNE) [3], [4], which will feature the largest LArTPC detector to be ever constructed and operated at a deep underground location in Lead, South Dakota, in the United States, starting in \sim 2025.

LArTPCs, including DUNE, work by imaging particle tracks and other signatures imprinted in a large, uniform detector volume by particles produced in neutrino or other rare physics interactions. Different interactions yield distinct image topologies that are identifiable and differentiable by their spatial extent, shape, and pixel intensity, when viewed as two-dimensional projections of a three-dimensional detector region. Furthermore, the format of the detector-generated raw data represents exactly two-dimensional projections of the

activity inside the detector; as such, a potentially advantageous solution for real-time data processing and data selection (triggering) on interesting detector activity is image analysis with hardware-accelerated Deep Neural Networks (DNNs).

DNNs are already being applied successfully for the offline analysis of data recorded by existing high energy physics experiments [5], including operating LArTPCs. In the case of the latter, MicroBooNE is pioneering the use of deep learning for neutrino physics analyses (see, e.g., [6], [7]), and similar DNN-based methodologies have now been adopted for several analyses planned with the future DUNE experiment [4]. Machine learning approaches to LArTPC data analysis are gaining increasing traction (see, e.g. [8]); meanwhile, new techniques are continually being considered to improve data processing latency and resource requirements, with promising results [9].

At the same time, the success of DNNs more generally has motivated the research and development of many specialized system architectures and accelerators both in academia and in industry. An excellent overview of the challenges of accelerating DNNs in hardware and a comprehensive survey of many techniques and frameworks that have been proposed so far in the literature is provided in [10]. In terms of implementation, DNN frameworks mainly target CPUs and GPUs. In particular, GPUs offer high computational density and high level of programmability; this simplifies the interface with operating systems while providing access to powerful computational platforms for data-parallel algorithms and dense floating-point operations. GPU performance, however, comes with high power dissipation, making a GPU-based solution unsustainable for many high-performance embedded systems that require major power efficiency. Thanks to their hardware reconfigurability, Field Programmable Gate Arrays (FPGAs) are a valid alternative solution as power-aware platforms for DNN acceleration [10]. In addition to hardware developments, frameworks such as Caffe [11] and Tensorflow [12] allow a much larger user base for modern DNNs.

In this paper, we investigate the viability of DNN implementations in a variety of architecture systems, including

GPU for online data processing, and FPGA or mixed FPGA-CPU architecture systems for real-time data processing, both for the purposes of data selection (triggering) for a high-resolution and high-rate imaging detector. The application we specifically target is that of DUNE, which involves real-time streaming of data rates of the order of tens of terabits per second. The proposed data selection schemes, however, may be applicable to any LArTPC, sharing the same technology as DUNE, and particularly viable for smaller-scale ones. We note that the application of machine learning algorithms for triggering purposes has been considered for other types of particle detectors (see, e.g. [13]). However, the application proposed here for LArTPCs is a new effort, and it deals with a unique set of challenges: specifically, LArTPC triggering is governed by a much larger input (image) size, but also benefits from relaxed latency constraints due to a much slower detector response than other types of particle detectors. The targeted DUNE application and DUNE detector design are presented in Sec. II.

To motivate the application of DNNs for DUNE data selection purposes, we train and investigate the performance of a number of DNNs on simulated LArTPC raw data images. Results obtained on GPUs are presented in Sec. III, and demonstrate high efficiency in selecting rare physics interactions of interest, while maintaining a sufficiently low selection rate from background interactions and detector noise. Latency and power dissipation considerations, however, motivate the investigation of inference on FPGA or mixed FPGA-CPU systems, which have been shown to achieve significant speedup [14]. As such, in Sec. IV, we present several contributions for designing hardware acceleration of Convolutional Neural Network (CNN) inference algorithms on resource-constrained platforms like FPGAs. By using a customizable and efficient hardware accelerator design for the various layers, we show that the flexibility of the accelerator design together with the possibility of leveraging the knobs provided by High Level Synthesis (HLS) tools enable the design of high-performance accelerators that can greatly benefit the deployment of DNN models. Finally, in Sec. V, we identify DNNs which would satisfy DUNE physics and latency requirements, considering also resource utilization on an FPGA with specifications that might be suitable for DUNE readout.

II. APPLICATION USE CASE: DEEP UNDERGROUND NEUTRINO EXPERIMENT

DUNE is an international particle physics experiment that aims to study neutrinos and their oscillation patterns with unprecedented sensitivity as well as search for other rare particle interaction signatures that will inform our understanding of nature at the most fundamental level. In particular, DUNE measurements aim to elucidate the underlying mechanism responsible for the prevalence of matter over antimatter in our observable universe. To accomplish these physics goals, the DUNE far detector will employ four LArTPC modules, each holding 10 kilotons of liquid argon in total fiducial detector mass, and will operate for more than a decade in a deep

TABLE I
EXPECTED RATES OF RARE OFF-BEAM EVENTS AND OTHER OFF-BEAM SIGNATURES IN A 10 KTON (FIDUCIAL MASS) DUNE FAR DETECTOR MODULE [4].

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

^aA galactic supernova burst is expected at a rate of roughly once per century. The latest galactic supernova burst was observed in 1604 [15].

underground location at Sanford Labs, in Lead, South Dakota, beginning in the middle of the next decade.

To study neutrino oscillations, DUNE must detect interactions of neutrinos from a high-intensity pulsed beam from Fermi National Accelerator Lab, in Batavia, Illinois. Selecting and recording these interactions is straightforward since they are all expected to arrive only during a relatively short time dictated by the beam pulse structure; the latter is precisely known due to external beam timing signals informing the trigger decision. To study other rare, off-beam events such as proton decay events, neutron-antineutron oscillation events, and interactions of neutrinos from galactic supernova bursts, however, DUNE must continually process its data in order to make a data-driven decision to select and record these signatures. This is because these signatures are random in nature, and no prompt external timing signal is available to independently inform the data selection decision. The expected rate of rare off-beam events, and other off-beam interactions of interest, in DUNE is provided in Tab. I.

The DUNE system responsible for data selection must, in the end, only allow for effectively 30 petabytes of data per year to be diverted to permanent storage offline [4]. As such, given the multiple tens of terabits per second raw data rate of the DUNE far detector, a factor of 10^4 data reduction must effectively be achieved by the system, without compromising efficiency for selecting rare events of interest. Generally, a trigger efficiency of $>99\%$ is required for high energy events, including atmospheric neutrino interactions, proton decay events, neutron-antineutron oscillation events, and cosmic ray muon events in the detector. Similar trigger efficiency is also required for selecting aggregates of multiple low energy supernova neutrino interactions that are expected to occur in case of a galactic supernova burst. In that case, the trigger efficiency requirement on any individual supernova neutrino interaction can be relaxed, and a multiplicity condition can be used to boost efficiency for coincident interactions¹.

The main challenge, specific to the supernova burst trigger, is that individual supernova events are characterized by low energy deposition in the detector; as such, their observable

¹In the case of a supernova at the edge of our galaxy, for example, approximately 50 supernova neutrino interactions are expected over the span of ten seconds in each DUNE 10 kton module.

signature is similar to that of intrinsic radiological backgrounds and electronics noise in the detector, which are the dominant contributor to observable signals in the DUNE data. Consequently, in order to achieve the desired data reduction factor, significant noise and radiological background rejection is needed.

Two distinct detector designs are in development for the DUNE far detector modules. We restrict the discussion and studies presented in this paper to the so-called “single phase” LArTPC module technology, described in the following subsection, following [4]. However, both the single phase and “dual phase” technology operate on high-resolution imaging principles; we therefore expect that comparable challenges, solutions, and performance would be achievable for the “dual phase” technology for DUNE as well.

A. DUNE Single Phase Detector Design

In the case of the DUNE far detector “single phase” design, each DUNE 10 kton far detector module is segmented into 150 individual “cells” (rectangular volumes) of liquid argon, which are imaged by sensor-wire arrays, called an Anode Plane Arrays (APAs). An APA is positioned in the middle of each cell, and it consists of multiple planes of parallel wires oriented in three distinct directions relative to the vertical direction. The wires sense ionization charge (electrons) liberated by charged particles along the charged particles’ paths as they traverse the liquid argon volume enclosed in the cell; the ionization charge drifts toward the wire planes under the influence of a strong, uniform electric field applied across each cell, on either side of the APA. Given the arrival time of the ionization charge, relative to the time of the interaction (identified and recorded by detecting the prompt scintillation light produced at the time of the interaction, using a dedicated photon detection system), the drift coordinate of the event can be reconstructed. The ionization signals recorded as a function of wire number across each wire plane, and as a function of time, can then be mapped into a two-dimensional projected view of the cell, for a given time; this makes it possible to reconstruct a three-dimensional view of any interaction inside a cell by matching signals across the three stereoscopic views (one per plane).

The studies in this paper involve only signals from vertically oriented wire planes. One such plane exists on each side of the APA, and makes up a so called charge “collection” wire plane. Due to the electric field configuration and readout electronics response, recorded signals on collection wires are unipolar; as such, their amplitudes and integrals, in particular, correlate highly with the amount of ionization charge arriving at each wire. We refer to channel vs. time data which spans the equivalent of a collection plane times drift time (drift length on one side of the APA divided by wire signal sampling rate) as an “APA-frame”. For the DUNE APA cell physical dimensions and nominal electric field configuration, the APA-frame drift time corresponds to 2.25 ms.

Simulations of APA-frames representative of several topologies of interest, from Tab. I, are shown in Fig. 1. APA-frames are simulated using the LArSoft framework [16], [17] and

DUNE Monte Carlo generation tools [18]. DUNE Monte Carlo generation configuration parameters are set to the `dunetpc v07_13_00` default values, except for the electronics noise RMS levels, which are artificially enhanced for conservatism; specifically, in our simulations, we increase the collection plane electronics noise RMS by 40% relative to the default value. All APA-frames with topologies of interest also include default radiological background and electronics noise.

B. DUNE Data Acquisition System Design

DUNE will have to operate continually, for more than a decade, streaming data out of its LArTPC detectors at a total rate of multiple tens of terabits per second. For reference, MicroBooNE [1] and ProtoDUNE [2], the two largest currently operating LArTPCs, stream images continually at a data rate of greater than 260 and 490 gigabits per second, respectively. Unlike DUNE, these experiments do not have a rare event search physics scope. Data reduction for these detectors is therefore achieved through a combination of external trigger signals informing when to record a small subset of that data, and additional real-time compression, filtering and/or zero-suppression carried out in FPGA and/or CPU (see, e.g. [1], [19]).

Differently from these detectors, the DUNE detector must be capable of processing its data in real time, or, in an online fashion, in order to make data-driven decisions to record what might be rare physics events. DUNE’s data acquisition system (DAQ), and in particular its data selection (sub)system, must do so with negligible dead-time, to maximize the detector’s physics sensitivity to rare signatures. An additional constraint is power distribution limitations at the (underground) detector site. Specifically, the DUNE far detector DAQ is limited to 500 kVA of power underground, or 125 kVA per 10 kton module, plus an additional 50 kVA of power available on the surface for back-end DAQ [4].

The baseline DUNE DAQ design is documented in detail in [4]. It employs a multi-level data selection system. First, a low-level data selection decision is achieved on a combination of CPU and FPGA resources. This level of data selection is executed independently on a per-APA basis, while the second level, to first order, aggregates low-level information from all APAs in a single module to make a module-level trigger decision. The module-level trigger decision is executed on CPU resources, and its latency is limited to a few seconds. When formed, a module-level trigger decision instructs the readout of several milliseconds worth of continuous data from all 150 APAs in the module, or, in the case of a supernova burst trigger decision, 100 seconds worth of continuous data from all 150 APAs. Non-supernova burst trigger decision rates of up to $O(1)$ Hz are possible, while supernova burst trigger decision rates are limited to one per month. These upper limits on trigger rates include fake triggers on accidental backgrounds and noise; therefore, background noise considerations are especially important in the case of supernova burst triggers. Additional data down-selection can be achieved by the use

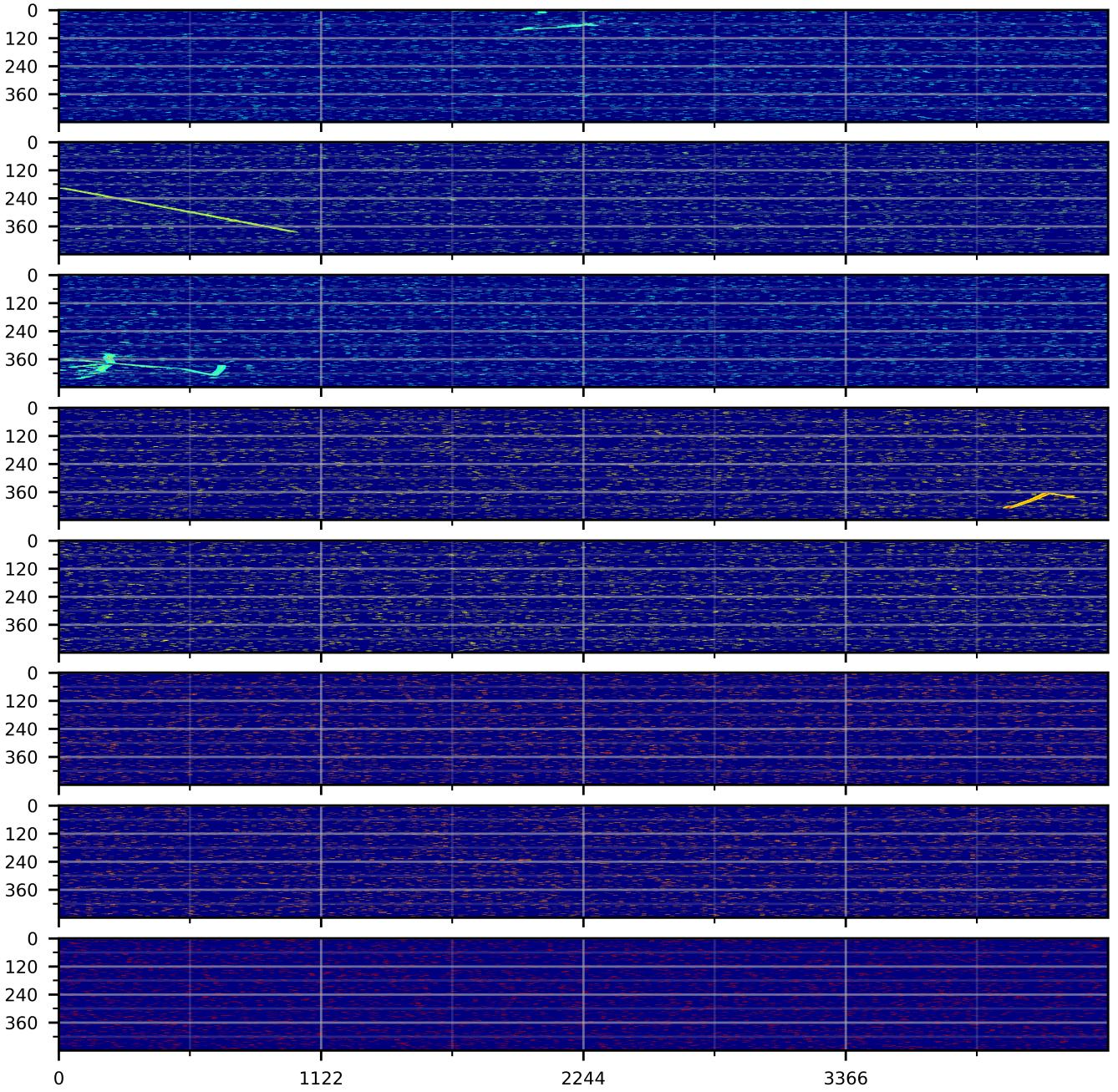


Fig. 1. Simulated APA-frames, representative of three main types of signatures of interest. The top four frames correspond to high energy events; the lower four frames correspond to low energy supernova neutrino events (first two) and empty events including only background noise (bottom two). APA-frames are defined according to the APA drift volume in which the interaction originates. The y axis of each frame corresponds to collection plane channel number; the x axis corresponds to time tick (2 MHz) across a full 2.25 ms readout.

of a high-level filter farm, which is envisioned to employ data selection techniques similar to the ones presented in this work.

III. DNN-BASED LArTPC DATA SELECTION

To motivate DNN-based LArTPC data selection, we have studied a number of DNNs in terms of their performance on classifying simulated DUNE far detector single phase APA-frames. We have considered a multi-class data classification scheme, where the different classes represent different types

of off-beam physics events of interest that can occur in the DUNE far detector, as well as non-physics events (intrinsic to the detector materials radiological backgrounds and electronics noise backgrounds).

The methodology we followed assumes that a two-level data selection system is used to (1) first generate a low-level data selection decision, specifically the classification of APA-frames according to their content with the use of a DNN, and to (2) subsequently process those decisions further in order to

make a module-level data selection decision. More specifically, the module-level data selection stage keeps track of information² from APA-frames that have been tagged as a certain type of interaction over the entire 10 kton detector module, over a given time interval. In this way, for example, a supernova burst trigger decision can be generated at the module level if multiple APA-frames are tagged by the “low-level” trigger as containing supernova neutrino interactions over a short amount of time (typically on the order of seconds). Our studies focus particularly on the low-level stage of processing.

The APA-frames stream continually from each DUNE detector module, at a rate of 200 frames (one for each drift volume) per 2.25 ms. Each frame is 480 channels wide by $(2.25 \text{ ms}) \times (2 \text{ MHz}) = 4500$ samples³ wide, corresponding to a total of 4.15 megapixels, with 12-bit color resolution. Because of the large APA-frame size (3.2 GB), significant down-sizing is necessary in order to fit APA-frames into image sizes typically processed by DNNs. Down-sizing is also applied in anticipation of the limited resources available on FPGAs that the DUNE far detector data selection system will employ for low-level data selection [4], which we consider to be a candidate hardware platform for DNN deployment.

Two methods were followed to pre-process APA-frames in preparation for DNN classification; classification was carried out with a VGG16b network [20] trained and tested independently for each method on a GPU:

- Method 1: In the first method, noise removal was minimally applied to each APA-frame, and the resulting image was re-sized by down-sampling it into a 600×600 image, to be used for DNN inference.
- Method 2: In the second method, aggressive noise removal was applied to each APA-frame before down-sizing the image for inference, followed by cropping around a signal “region of interest” (ROI), and re-sizing the resulting ROI (by down-sampling or up-sampling) into a 64×64 image. The noise removal and ROI finding were informed by studying the ADC distributions of simulated APA-frames of different signatures, as shown in Fig. 2. Examples of ROIs are shown in Fig. 3.

For both methods, the images were used to train a customized VGG16b network, and the resulting network was tested on a statistically independent sample of images, prepared in the same way, for accuracy and inference speed. The tests were performed on a single GPU, NVIDIA GeForce GTX 1080 Ti.

Inference results on GPU from each method for VGG16b are summarized in Tabs. II and III. The tables show the number of ROI images used for training and testing for each sample; and resulting accuracy, identified in terms of the fraction of input images in the testing case which get classified under each label: background noise (NB), low energy supernova neutrino interaction (LE), or high energy interaction (HE). The given fractions are inclusive of all event energies. Finally, per-

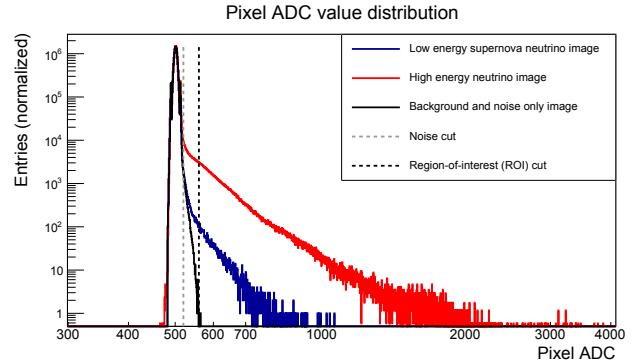


Fig. 2. Pixel ADC distributions of frame images for the three classes in consideration: background noise (black), low energy neutrino (blue), and high energy neutrino (red) images. The pixel ADC values range from 0 to 4095 (12-bit ADC). The distributions are absolutely normalized to 100 images with 480×4488 pixels each. The background noise distribution peaks below 520 for all frames. The dashed vertical lines indicate cuts that were used in pre-processing input images for the networks, in order to de-noise the raw images and to select regions with candidate physics interactions. Based on these distributions, a noise removal cut (indicated by the dashed gray line) and an ROI cut (indicated by the dashed black line) was set to 520 and 560 ADC, respectively.

TABLE II
GPU INFERENCE RESULTS USING METHOD 1, OBTAINED WITH A VGG16B NETWORK (TRAINING FOR 2 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	51,100	99,000	91.45	8.49	0.06	
LE	44,900	29,800	3.17	96.83	0	27.7 \pm 8.6
HE	52,828	67,178	6.03	3.48	90.48	

APA-frame inference times are provided, in milliseconds, and include image input i/o from host (GPU server) memory. The key table parameters are the correct classification rates of low energy and high energy frames, both of which are required to be high by DUNE physics performance requirements, as well as the mis-classification rate of noise frames as low energy frames, which should be as low as possible by data reduction requirement considerations. Both methods are found to yield comparable results in terms of classification accuracy. More specifically, the networks are able to select high energy and low energy frames with efficiencies in excess of 95% and 90%, respectively. Required efficiency for high energy frames should be $> 99\%$ for interactions with visible energy in excess of 100 MeV. The obtained efficiencies are integrated over all energies (which extend below 100 MeV); it is expected that a HE efficiency calculated relative to interactions with visible energy in excess of 100 MeV would be higher. While signal efficiency performance is comparable for the two methods, Method 2 performs much better with respect to mis-classification rates for background noise frames as LE frames, where a false pass rate of 0.35% is achieved.

Inference latency for the two methods is also comparable, although Method 2 inference is faster by more than a factor of five, due to the reduced size of the input image. Latency

²Spatial coordinate, type of interaction, etc.

³More specifically, 4488 samples are used for simulation purposes.

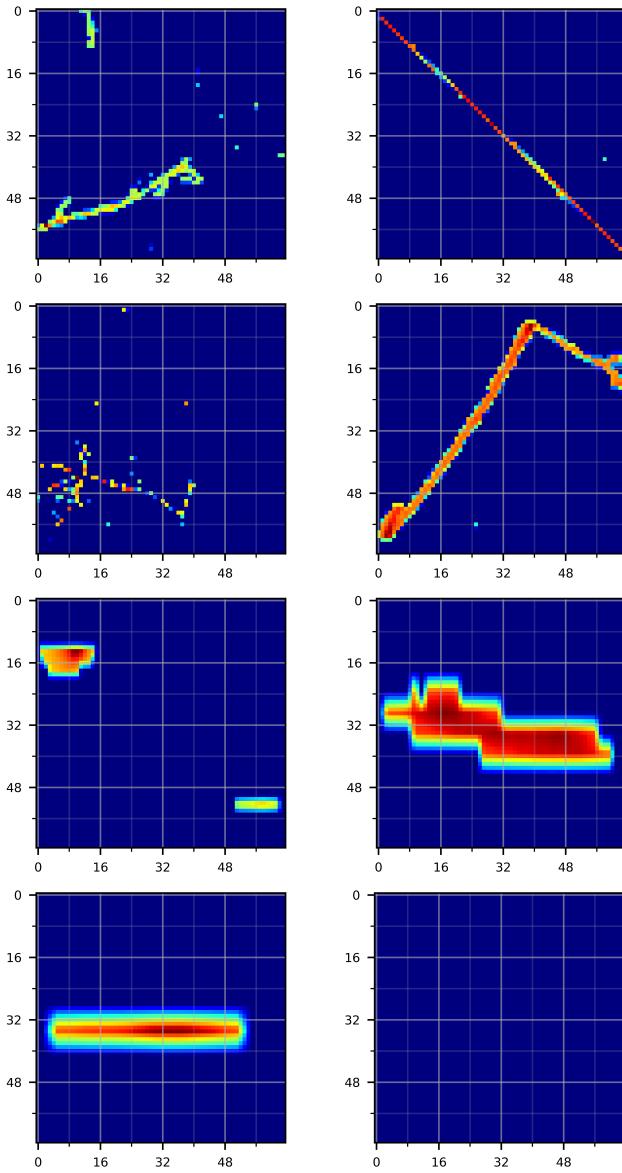


Fig. 3. ROIs extracted using Method 2 for the simulated frames shown in Fig. 1. The y axis represents channel space; the x axis represents time space. The top four panels correspond to high energy interactions; the subsequent two correspond to low energy interactions; the bottom two correspond to background noise (typically empty frames, after noise removal, or noise artifacts). Noise removal is achieved by zero-suppressing pixels with ADC values below 520; an ROI is defined by first finding the smallest contiguous rectangular region in a frame that contains at least one pixel value exceeding 560 ADC, padded by five (5) additional pixels in each direction (left, right, top, or bottom); the resulting region is down-sized or up-sized, by down-sampling or up-sampling to fit into a 64×64 image, as shown here, defined as an ROI, and is then fed into a DNN for inference.

considerations determine whether frame-by-frame inference can be applied during the low-level data selection stage of the DUNE far detector DAQ system; such application would have to keep up with the frame rate of 66.6×10^3 fps. In the case of Method 1, if we required that every frame go through image classification, the observed latency of 27.7 ms (an order of magnitude off 2.25 ms even with a 150-fold parallelization)

TABLE III

GPU INFERENCE RESULTS USING METHOD 2, OBTAINED WITH THE VGG16B NETWORK (TRAINING FOR 13 EPOCHS AND LEARNING RATE SET TO 10^{-4}). NB* CORRESPONDS TO EXPLICITLY NON-EMPTY BACKGROUND NOISE ROIS, CONTAINING NOISE ARTIFACTS, WHICH REPRESENT APPROXIMATELY 2% OF THE REGIONS FOUND AFTER NOISE REMOVAL.

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	
LE	12,050	3,970	3.78	95.04	1.18	
HE	10,137	3,417	2.99	6.88	90.14	5.0 ± 0.3

TABLE IV

GPU INFERENCE RESULTS USING METHOD 2, OBTAINED WITH THE CNN_S NETWORK (TRAINING FOR 48 EPOCHS AND LEARNING RATE SET TO 2×10^{-3}).

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	
HE	10,137	3,417	3.63	6.15	90.22	1.6 ± 0.1

would preclude such application during low-level data selection, unless a more-than-10-fold parallelization of frame-by-frame processing were to be implemented; application at high-level filter stage, however, is viable, because a relatively low module-level trigger rate (for example of order 1 Hz readout of 200 APA-frames) would make data rate handling more manageable. In the case of Method 2, the inference latency (comparable to APA-frame length of 2.25 ms) is far more promising for a frame-by-frame online low-level data selection implementation; furthermore, the processing time requirement for this method can be relaxed further based on the additional reduction of frame rate gained by the aggressive noise removal and ROI formation pre-processing stage. We have found that after noise removal and ROI finding, only 2% of the 2.25 ms-long background noise frames survive. Considering that most APA-frames that DUNE will be reading out will contain only background noise, we expect that the average frame rate reduction factor gained will be close to that of the background noise reduction factor. Hence, directing only ROIs containing non-zero pixels to network inference, for example, could relax the processing time requirement by a factor of 50.

Additional fake (background noise) trigger reduction is possible at the module-level data selection stage, by aggregating APA-frames classified as LE and considering their coincidence over the anticipated duration (10 seconds) of a supernova burst, following the methodology for supernova burst triggering in [4]. Findings from preliminary studies [4], [21] support the successful application of the coincidence-based methodology fed by CNN-based (using a VGG16b network) low-level information.

The promise of Method 2 for online application for low-level data selection further motivates the use of smaller networks, and, for the purposes of further acceleration on FPGA,

TABLE V

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	

TABLE VI

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	

smaller input images. The second method was therefore further explored for a number of other customizable networks, besides VGG16b [20], including a smaller, simpler CNN than VGG16b, referred to as CNN_s [22], a Multi-Layer Perceptron (MLP) network [23], and a ResNet14b network [24]. Results from the three additional networks are provided in Tabs. IV through VI, to be considered in comparison with VGG16b results in Table III. The best performance is obtained with VGG16b and CNN_s. The simple CNN (CNN_s) performs comparably with VGG16b in terms of the accuracy, albeit with slightly higher pass rate ($\sim 0.5\%$) on background noise ROIs. MLP and ResNet14b also have comparable pass rates ($\sim 0.5\%$) for background noise, but the accuracies for low energy and high energy ROIs are not as high as those for VGG16b or CNN_s. Inference times with CNN_s (on a single GPU card) are an order of magnitude lower than for VGG16b, due to the reduced number of layers and convolutions per layer.

Finally, we note that lower background noise pass rates could be achievable using a variation of a CNN-based selection. For example, in [21], Method 1 is used to train against six classes: NB, LE, plus the four subclasses of the HE class including atmospheric neutrino interactions (atm), nucleon decay (ndk), neutron antineutron oscillation (nnbar), and cosmic interactions (cosmic). Rather than classifying frames in terms of the six labels according to the label returning the highest score, a cut on the NB classification score is applied in order to reject frames with high enough NB scores, and select all surviving APA-frames. Results based on this classification scheme are summarized in Tab. VII as a function of NB score cut. The number of ROI images used for training and testing for each sample in Tab. VII correspond to those given in Tab. II. The main difference relative to Tab. II is that accuracy is identified in terms of the fraction of input images in the testing case which have NB score lower than what is indicated on the left column. (Here, too, fractions are inclusive of all

energies.) The average inference time is comparable to that presented in Tab. II, and includes image input i/o from host (GPU server) memory.

IV. CNN IMPLEMENTATION IN FPGA

The accuracy performance of CNN_s obtained with reduced-size raw data images, combined with the reduced size of the network relative to VGG16b, motivate studies for further hardware acceleration. Hardware accelerators can be designed according to two main different approaches [25]: the designer can tightly couple the hardware functional unit inside the pipeline of a processor core or choose a loose out-of-core coupling architecture. Loosely-coupled accelerators (LCA) are hardware accelerators capable of performing Direct-Memory Access (DMA) to external main memory. LCAs are located outside the processor cores, for example on the FPGA fabric, and interact with the rest of the chip through on-chip interconnects. They can implement coarse-grain operations with dedicated datapaths that can accelerate a complete application functionality (e.g. the convolutional layers in the case of a CNN). To implement our accelerators we adopted the LCA approach, as it represents a perfect match in terms of reconfigurability and flexibility with FPGAs and embedded SoCs.

The bottleneck for the performance of inference of CNNs are the convolutional layers, which alone are responsible for more than 90% of the computations performed on networks like VGG16b. Thus we chose to specifically design a *convolutional* LCA for our CNNs.

While the workloads of many accelerators described in literature are fixed and known at design time [26], a convolutional layer has a number of parameters that are known only at run time (input dimensions, number of input channels, number of filters per layer, etc.). Different configurations of these hyperparameters lead to drastic changes in memory requirements and computational capabilities. Thus, we chose to design a LCA that is *configurable* at run-time.

We used High-Level Synthesis (HLS) to obtain the FPGA implementation starting from specifications made in C/C++ [27]. Current HLS tools enable an effective exploration of the design space of an accelerator to obtain many alternative implementations which are trade-offs of resource/power requirements and performance [28]–[30].

A. Accelerator Architecture

Figure 4 illustrates the main components and memories of our configurable convolutional LCA. It embeds three private local memories (for storing the input and output features and the filter weights), a patch extractor (for data reordering), and several multiply-and-accumulate engines which are the core of the convolution operations. The accelerator communicates with the rest of the chip through AXI4 interconnects [31].

Private Local Memory. Custom hardware accelerators allow designers to tune the microarchitecture and enable higher level of optimization to meet a specific configuration and workload, providing high performance and energy efficiency [32].

TABLE VII

GPU INFERENCE RESULTS USING METHOD 1, OBTAINED WITH THE VGG16B NETWORK (TRAINING FOR 2 EPOCHS AND LEARNING RATE SET TO 2×10^{-4}), TRAINED ON SIX CLASS LABELS. SEE TEXT FOR MORE DETAILS.

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

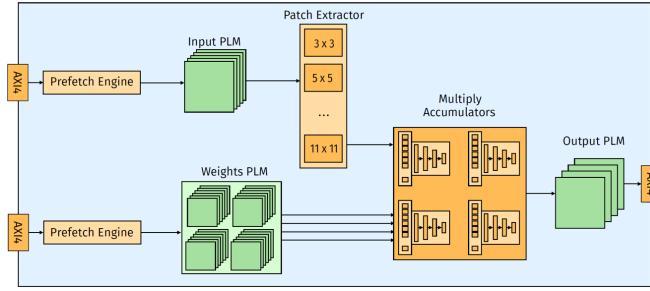


Fig. 4. Overview of the configurable loosely-coupled accelerator.

General purpose processors (CPUs) leverage the hierarchy of caches and memory to provide the best solution in terms of bandwidth and latency across a variety of applications. Similarly, GPUs offer very high bandwidth and massive availability of parallel computational cores (CUDA cores for GPUs NVIDIA). When implementing custom hardware accelerator on FPGAs, resource utilization and allocation is an important design constraint. The designer should carefully optimize the accelerator to reuse data as much as possible, thus balancing communication versus computation and reducing expensive memory transfers from the off-chip main memory. This requires the use of private local memories (PLMs), which offer low latency, high bandwidth memory and customizable word widths. They do so, by providing many banks and ports that are exclusively accessed by the datapath logic of the LCA that embeds them [25]. Careful design and tailoring of these structures for input/output ports, partitioning, and resource allocation is essential to constantly provide data to be fed to all the high-performance computational engines.

Patch Extractor. The patch extractor is an optimized module for retrieving the portion of the input features where the filters are applied. This operation is highly dependent on the choice of hyperparameters. Due to the irregular access pattern that this module performs while fetching data from the Input PLM, we decided to have various implementations for the most common cases, from the smallest 3×3 filters up to bigger 11×11 filters. At run time, accordingly to the settings of the convolutional layer, the accelerator would choose and enable the correct patch extractor.

Multiply-and-Accumulate (MAC). The computational core of convolutional layers lies in the MAC operation. The amount of MAC per input image added up quickly from few thou-

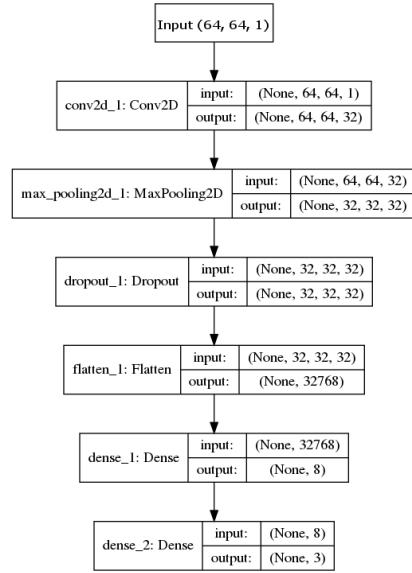


Fig. 5. Overview of our customized CNN, CNN_s.

sands for LeNet network [33] up to tens billions for VGG16 network [20]. To meet this computation requirement, our accelerator embeds several MAC engines. Each of these works on an independent input filter, allowing the parallelization of the computation of the output activation map across multiple filters. Internally, each MAC is implemented with a set of multipliers and accumulators. Changing the number of those components directly affects the degree of parallelism.

B. Performance and Power Analysis

We ran our tests on a Xilinx Embedded FPGA (Zynq UltraScale+ XCZU9EG MPSoC) that combines both an ARM Cortex-A53 64 bits multi-core processor and FPGA fabric fabricated in 16 nm technology. Overall, it represents a state-of-the-art *embedded* platform for a fair evaluation between FPGA acceleration of deep-learning inference tasks and pure software execution. We implemented a customized CNN, CNN_s (*DUNE-CNN-01*), in C language as a reference for our performance and power analysis. Figure 5 provides an overview of CNN_s.

Table VIII summarizes the results. The inference time of our customized CNN_s for a single image is 0.0855 seconds when executed as software on the ARM Cortex-A53 CPU. The

TABLE VIII

PERFORMANCE AND POWER ANALYSIS RESULTS ON THE EMBEDDED FPGA (ZYNQ ULTRASCALE+ XCZU9EG MPSoC).

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

CPU runs at 1.2 GHz. The inference time of the same network when it leverages the FPGA-acceleration is 0.0511 seconds. The accelerator runs at 100MHz on the FPGA fabric. The total power for the processing system (CPU) and for the FPGA accelerator are 2.871 Watts and 1.110 Watts respectively, as reported in Vivado Power Analysis. The energy efficiency of the FPGA implementation is more than 4 times better than the embedded CPU.

V. VIABILITY OF DNN APPLICATION FOR DUNE DATA SELECTION

Our studies demonstrate that DNNs in general can meet trigger efficiency requirements for selecting off-beam rare events in the DUNE far detector. In addition, for several CNNs (e.g. VGG16b), sufficiently low fake trigger rates can be met, such that the required data reduction factor of 10^4 can be achieved for high energy triggering and potentially also for low energy triggering with a subsequent module-level data selection stage; the latter is the subject of future investigations.

For the case of an online data selection implementation where inference is carried out exclusively in GPUs, out of the four DNNs considered, CNN_s is identified as the most viable option for GPU deployment at the low-level data selection stage. We assume that the necessary pre-processing from preparing the ROIs, which consists of operations which are commonly done in FPGA, can keep up with the raw detector APA-frame rate, and consider only the inference stage latency for the purposes of this discussion. Given that the inference time for an ROI with CNN_s is comparable to the APA-frame length (2.25 ms), CNN_s should on average keep up with frame-by-frame selection, with each APA's frames processed in a separate GPU card; this, however, implies that a 200-fold parallelization would be needed (across 200 GPU cards) to facilitate low-level data selection for a 10 kton module; this is unfeasible given power restrictions underground at the far detector location. On the other hand, a factor of 50 reduction in required GPU processing would be possible if a pre-processing step were to be added to remove empty ROIs before the inference stage. Such a step would remove all but 2% of the background noise ROIs from the inference stage, allowing for, on average, 112.5 ms per ROI for inference. The same scheme would make VGG16b viable for online inference no GPU as well, which yields characteristically higher efficiency for all rare events of interest.

In the case of FPGA inference we find that a factor of four (4) increase in energy efficiency (img/s/W) is possible over a software implementation in CPU of the same (CNN_s)

algorithm, motivating consideration of deployment of CNNs for low-level data selection on FPGA. The performance improvement over a software implementation is comparable for both inference speed (factor of 1.7) and power efficiency (factor of 2.6). Furthermore, we find that for smaller networks, such as for CNN_s, the resource allocation requirements for a full network implementation processing ROIs of 64×64 size are comparable with those available in state-of-the-art FPGAs, a desirable feature for simplified parallelization and for minimizing costs.

VI. SUMMARY

Acceleration of DNNs for real-time data selection is motivated by a number of up and coming high-resolution imaging particle detectors, in particular LArTPCs which work by imaging particle traces that are identifiable by their distinct topologies (spatial extent, shape, and pixel intensity) in two-dimensional view projections of three-dimensional detector regions. We have investigated the viability of DNN application for the purposes of real-time or online data selection (triggering) for such detectors, with a particular focus on the future DUNE experiment. Data selection is achieved by frame-by-frame classification of raw data streamed in channel vs. time space from 200 independent, self-contained regions of one of four DUNE far detector modules, assuming a single phase design.

Using simulated DUNE raw data images (APA-frames), we have found that such techniques yield promising results in terms of image classification accuracy, for a large variety (in terms of depth and size) of networks. Sufficiently high trigger efficiencies are achieved for selection of APA-frames with high energy interactions; lower trigger efficiencies are achieved for APA-frames with low energy interactions. However, supernova burst trigger efficiency can be optimized further by exploiting a higher-level decision which aggregates selected APA-frames over time, following the approach in [4].

We have further shown that latency and power considerations make the implementation of DNNs on GPUs for online inference viable for smaller networks and with significantly re-sized and down-selected ROI image inputs. Larger networks with re-sized full-frame information are viable only for the high level filter stage, at this time.

Finally, we have shown that implementation of DNNs on FPGAs for real-time inference at the low-level stage is promising, and have provided a viable path for development and optimization.

ACKNOWLEDGMENT

The authors thank J. Hewes, Y. Zhou, and S. Koo for early contributions to the development of CNNs and simulation tools used in the GPU studies, S. Rossi for a preliminary analysis of the FPGA implementation of the CNN, and K. Terao for valuable input and feedback to this work. This material is based upon work supported by the National Science Foundation under Grant No. PHY-1753228, and work supported in

part by the Research Initiatives in Science and Engineering (RISE) program of Columbia University.

REFERENCES

- [1] R. Acciarri *et al.* [MicroBooNE Collaboration], “Design and construction of the MicroBooNE detector,” JINST, vol. 12, no. 02, 2017.
- [2] B. Abi *et al.* [DUNE Collaboration], “The Single-Phase ProtoDUNE technical design report,” FERMILAB-DESIGN-2017-02.
- [3] B. Abi *et al.* [DUNE Collaboration], “The DUNE far detector interim design report, Volume 2: Single-phase module,” FERMILAB-DESIGN-2018-03.
- [4] DUNE Collaboration, “Deep Underground Neutrino Experiment (DUNE) technical design report,” in preparation.
- [5] A. Radovic *et al.*, “Machine learning at the energy and intensity frontiers of particle physics,” Nature, vol. 560, no. 7716, pages 41-48, 2018.
- [6] C. Adams *et al.* [MicroBooNE Collaboration], “A deep neural network for pixel-level electromagnetic particle identification in the MicroBooNE liquid argon time projection chamber,” FERMILAB-PUB-18-231-ND.
- [7] R. Acciarri *et al.* [MicroBooNE Collaboration], “Convolutional neural networks applied to neutrino events in a liquid argon time projection chamber,” JINST, vol. 12, no. 03, 2017.
- [8] L. Domine and K. Terao, “Scalable deep convolutional neural networks for sparse, locally dense liquid argon time projection chamber Data,” arXiv:1903.05663.
- [9] V. Sze *et al.*, “Efficient processing of deep neural networks: A tutorial and survey,” Proc. of the IEEE, vol. 105, no. 12, pages 2295-2329, 2017.
- [10] B. Falsafi *et al.*, “FPGAs versus GPUs in data centers,” IEEE Micro, vol. 37, no. 1, pages 60-72, Jan. 2017.
- [11] Y. Jia *et al.*, “Caffe: Convolutional architecture for fast feature embedding,” Proc. Intl. Conf. on Multimedia, pages 675-678, Nov. 2014.
- [12] M. Abadi *et al.*, “Tensorflow: A system for large-scale machine learning,” In OSDI, vol. 16, pages 265-283, 2016.
- [13] J. Duarte *et al.*, “FPGA-accelerated machine learning inference as a service for particle physics computing,” arXiv:1904.08986 [physics.data-an].
- [14] P. Coussy *et al.*, “An introduction to high-level synthesis,” IEEE Design & Test of Computers, 26(4):8-17, 2009.
- [15] Clark, D. H. and Stephenson, F. R., “The Historical Supernovae,” Supernovae: A survey of current research; Proceedings of the Advanced Study Institute. Cambridge, England: Dordrecht, D. Reidel Publishing Co., pages 355-370, 1982.
- [16] LArSoft release redmine 06.60.00
<https://cdcvn.fnal.gov/redmine/projects/larsoft/wiki/ReleaseNotes066000>.
- [17] LArSoft release redmine 07.13.00
<https://cdcvn.fnal.gov/redmine/projects/larsoft/wiki/ReleaseNotes071300>.
- [18] dunetpc release redmine 07.13.00
<https://cdcvn.fnal.gov/redmine/projects/dunetpc>
- [19] MicroBooNE Collaboration, “The Continuous Readout Stream of the MicroBooNE Liquid Argon Time Projection Chamber for Detection of Supernova Neutrinos,” MicroBooNE Public Note, MICROBOONE-NOTE-1030-PUB, 2019.
- [20] Simonyan, Karen and Zisserman, Andrew, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv:1409.1556.
- [21] G. Ge, Y. Jwa and G. Karagiorgi, “ML-based triggering for DUNE”, DUNE docd-id-11311, 2018.
- [22] Krizhevsky, Alex and Sutskever, Ilya and E. Hinton, Geoffrey, “ImageNet classification with deep convolutional neural networks,” Neural Information Processing Systems, vol. 25, 2012.
- [23] Frank Rosenblatt, “The perceptron: a probabilistic model for information storage and organization in the brain,” Cornell Aeronautical Laboratory, Psychological Review, vol. 65, No. 6, 1958.
- [24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep residual learning for image recognition,” arXiv:1512.03385.
- [25] Cota, Emilio G. *et al.*, “An analysis of accelerator coupling in heterogeneous architecture,” ACM/EDAC/IEEE Design Automation Conference (DAC), 2015.
- [26] Piccolboni, Luca *et al.*, “Broadening the exploration of the accelerator design space in embedded scalable platform,” IEEE High Performance Extreme Computing Conference (HPEC), 2017.
- [27] R. Nane *et al.*, “A survey and evaluation of FPGA high-level synthesis tools,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016.
- [28] A. Prost-Boucle *et al.*, “A fast and autonomous HLS methodology for hardware accelerator generation under resource constraints,” In Euromicro Conf. on Digital System Design, pages 201-208, 2013.
- [29] S. Rossi, “Accelerating convolutional neural networks with high-level synthesis,” M.S. Thesis, Columbia University in the city of New York, New York (USA), 2018.
- [30] X. Liu *et al.*, “High level synthesis of complex applications: An H.264 video decoder,” In Proc. Intl. Symp. on Field-Programmable Gate Arrays, pages 224-233, 2016.
- [31] ARM AMBA, “AXI and ACE Protocol Specification,” 2011.
- [32] Borkar, Shekhar and Andrew A. Chien, “The future of microprocessors,” Communications of the ACM, vol. 54, no. 5, 2011.
- [33] LeCun, Yann *et al.*, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, 1998.