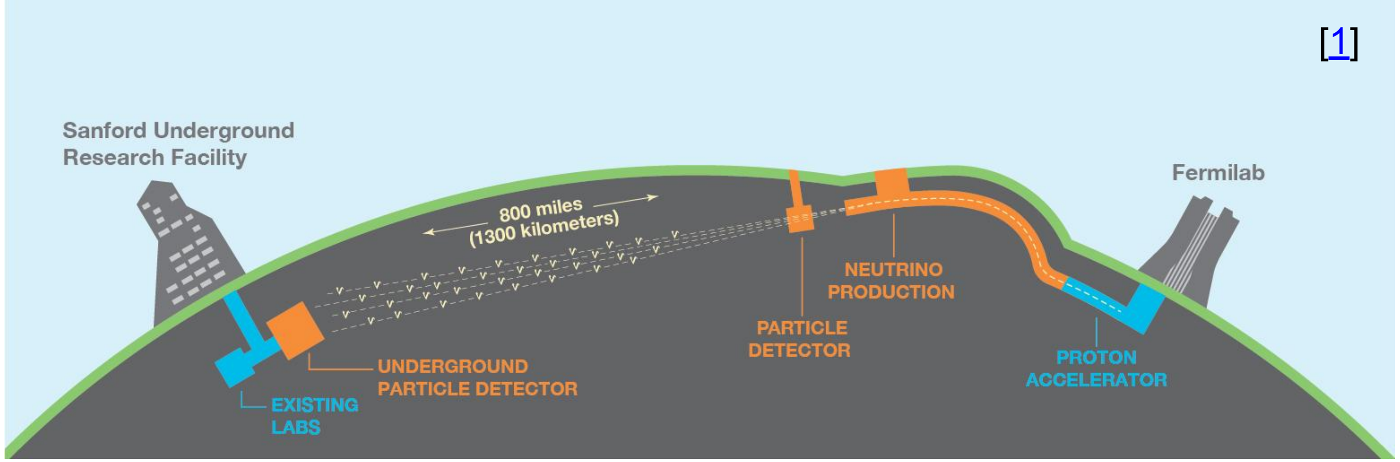
Real-Time Detection of Low-Energy Events for the DUNE Data Selection System

Judicaël Clair *[§*], Georgia Karagiorgi [§], Giuseppe Di Guglielmo [†], Luca Carloni [§]*
jsc2268@columbia.edu
[§] *Columbia University*
[†] *Fermi National Accelerator Laboratory*
[*] *on behalf of the DUNE Collaboration*

CPAD Instrumentation Frontier Workshop
Nov 30th 2022

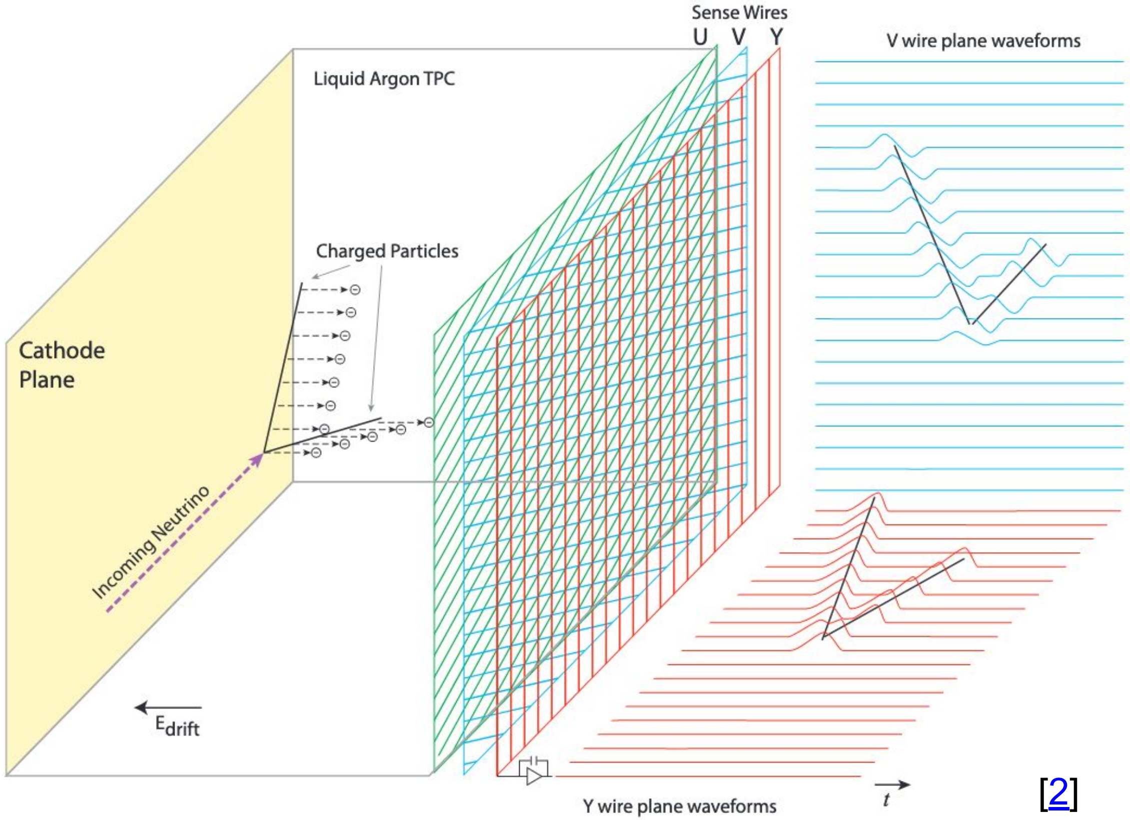
DUNE: Deep Underground Neutrino Experiment

- Beams of neutrinos and antineutrinos sent to a detector located 1300 km away and 1.5 km underground.
- The aim is to solve mysteries of the universe, e.g. why is there more matter than antimatter?
- Beam physics program includes:
  - What is the neutrino mass hierarchy?
  - Do neutrinos and antineutrinos violate CP symmetry?
- Off-beam physics program (**our focus**) includes:
  - Study of supernovae and formation of black holes; search for baryon number violation.



Far Detector: Liquid Argon Time Projection Chamber

- Operating Principle:
  - Large volume of liquid Argon.
  - Neutrinos interact with Argon atoms, producing charged particles that knock out electrons, which are then detected by planes of wires.
- DUNE:
  - 4 modules, each with 150 APAs.
  - Anode Plane Assembly (APA):
    - Planes of wires.
    - We are only considering one side of a collection plane: 480 wires.
    - Ignoring induction wires for now.



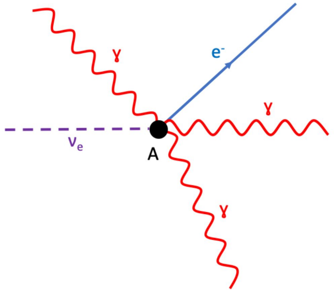
[2]

Need for Real-Time Detection

- Each wire is connected to its own 12-bit ADC @ 2 MHz.
  - Cannot save the several TB/s of data being produced continuously for over a decade.
  - Solution: detect rare events of interest in real time and only save those.
- Each side of a collection plane generates a 480x64 (*wire* x *time*) image every 32 μs [3].
- From these images we want to detect low-energy (ν-LE) supernova events:

ν-CC: charged-current scattering

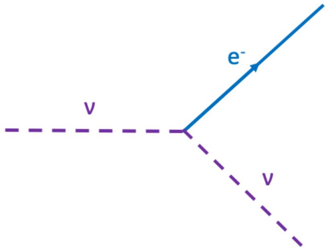
$$\nu_e \ {}^{40}Ar \rightarrow e^- \ {}^{40}K^*$$



[4]

ν-ES: elastic scattering

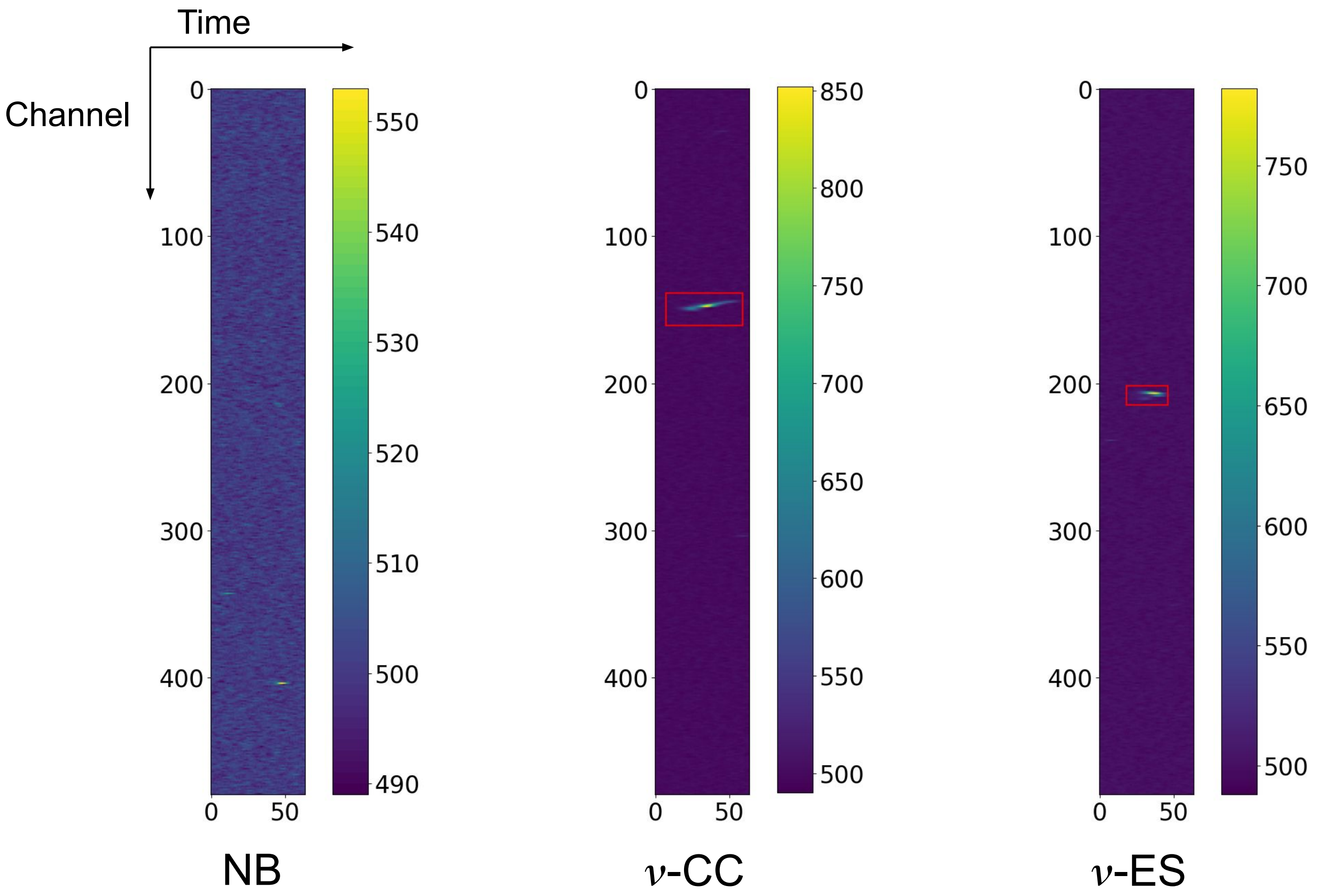
$$\overset{(-)}{\nu} \ e^- \longrightarrow \overset{(-)}{\nu} \ e^-$$



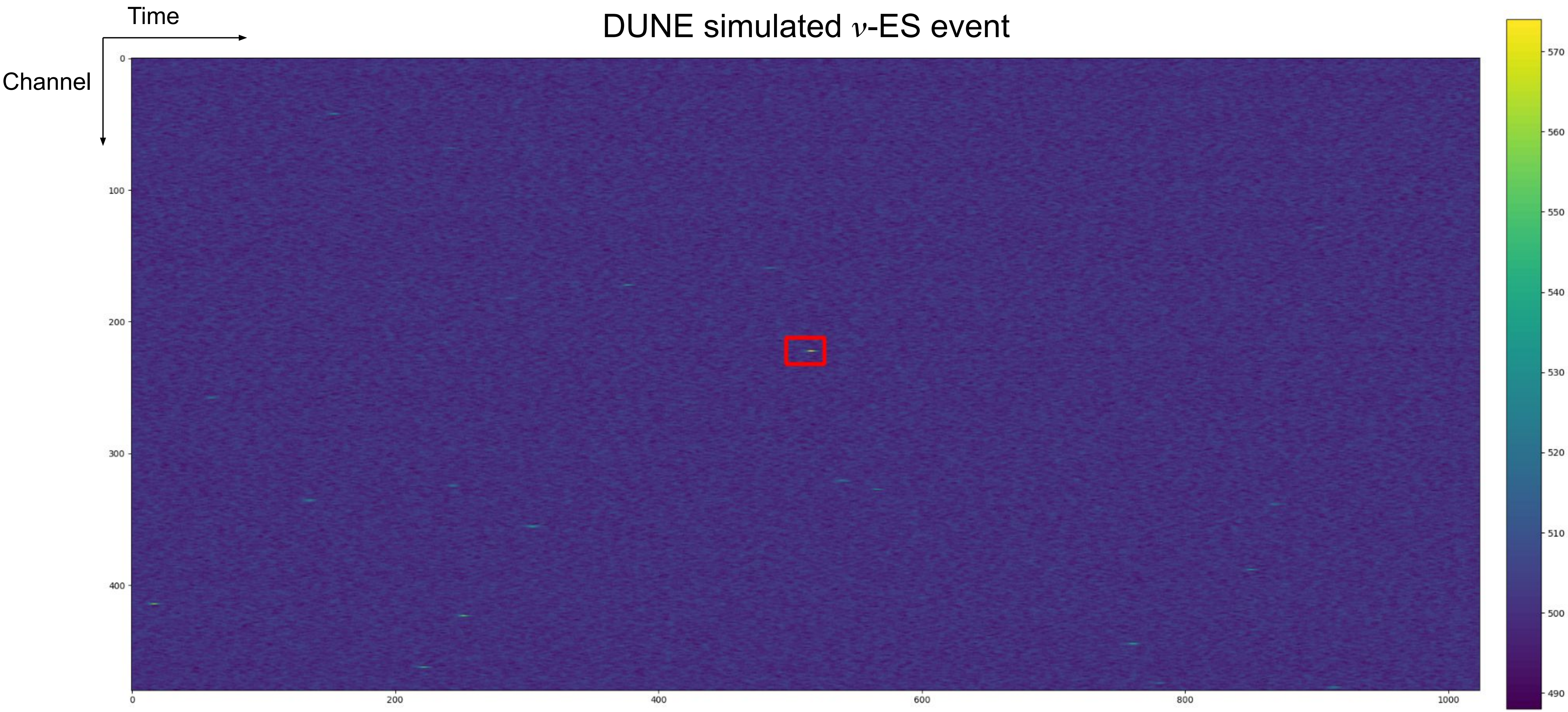
[4]

Image Classes

- NB: noise/radiological background.
- ν-CC: charged-current scattering.
- ν-ES: elastic scattering.

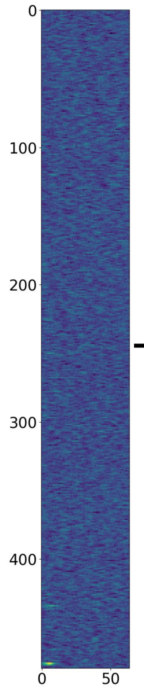


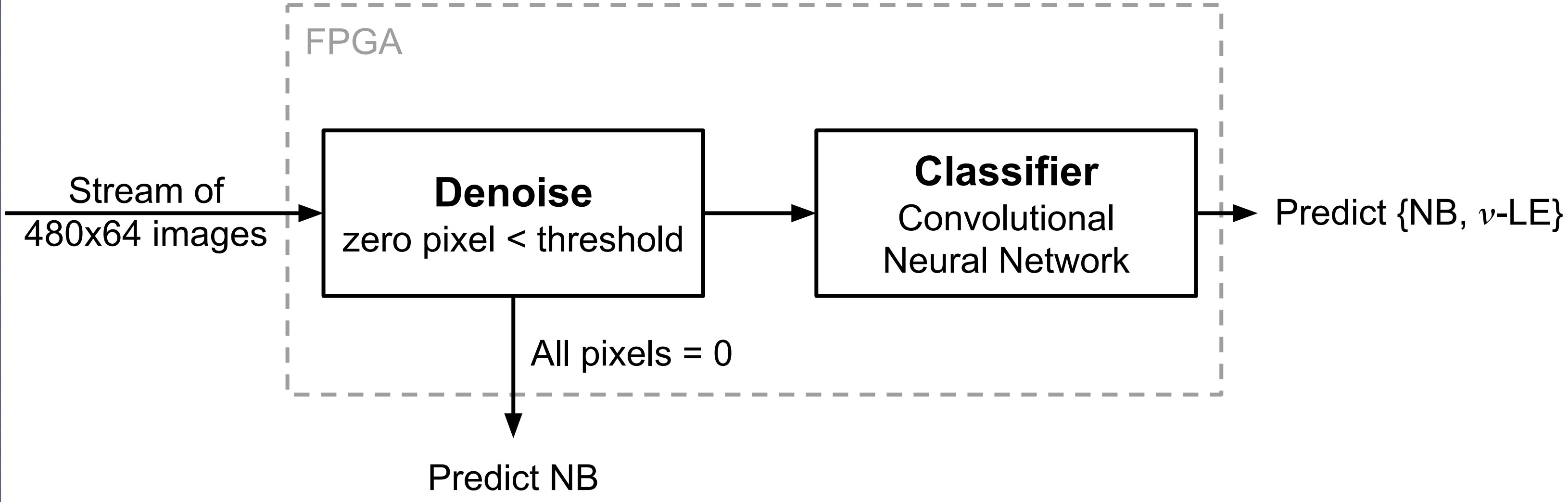
ν-LE Event Detection Difficulty

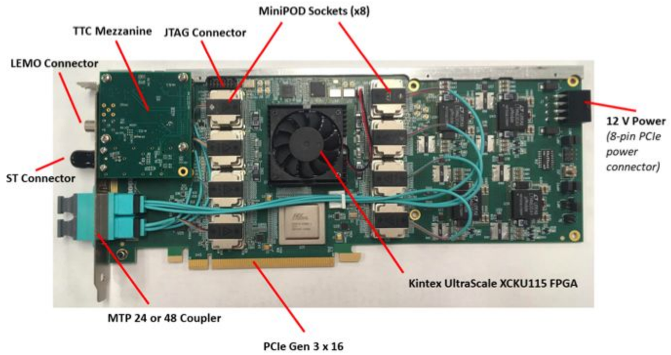


Main Goal

- Classify ν-LE events in real time with ≥ 90% efficiency.
- Reject NB images with ≫99.99% efficiency (to reduce data by a factor of ≥ 10^4).
- Each incoming 480x64 image must be processed within 32 μs to avoid buffering.
- Deployable on the Felix FPGA board (76Mb BRAM, 5.5k DSPs, 1.3M FFs, 660k LUTs).



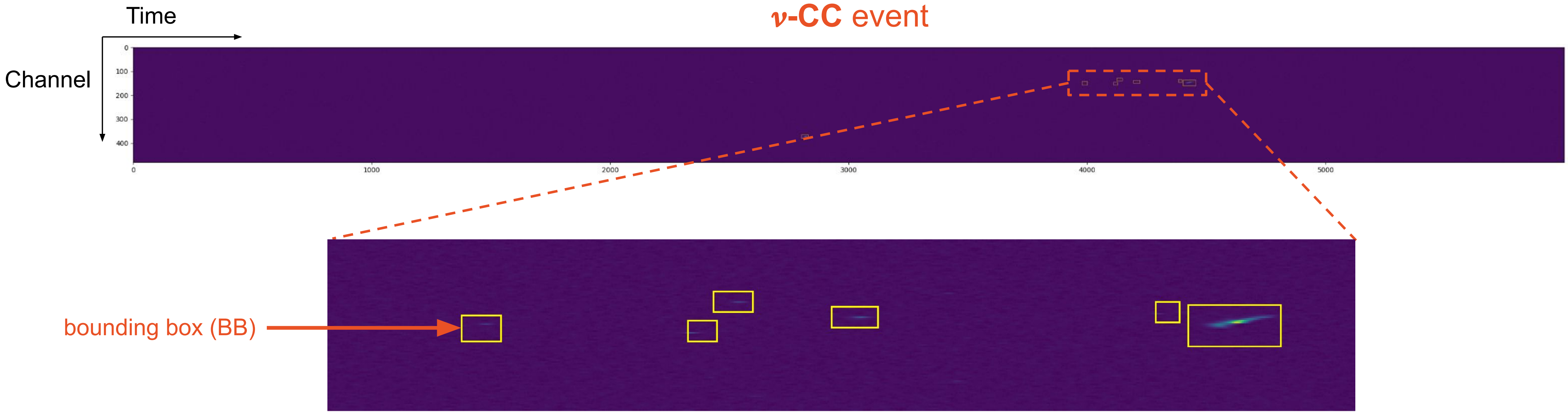




Each APA or pair of APAs will be read out by a single Felix board.

Simulation Data

- Data generated from DUNE simulations:
  - Considers both electronics noise and ambient radiological backgrounds in the detector.
  - Each simulated image is 480x6000 (needs to be sliced up into 480x64 images).
  - Each image has at most one event (ν-ES or ν-CC).
  - If image has an event, location of each particle of interest is provided.

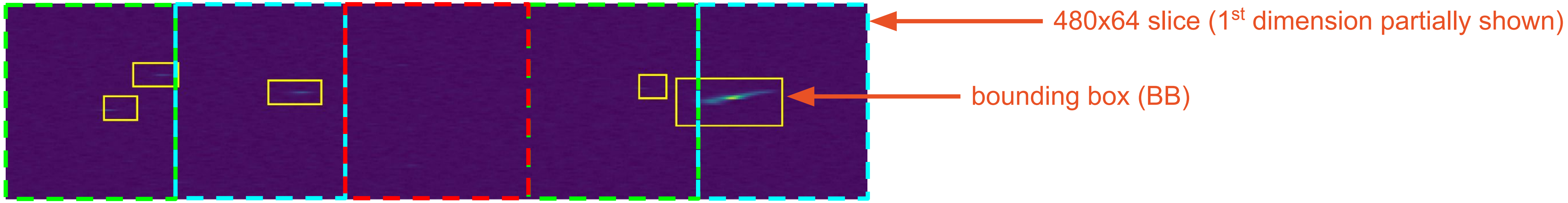


Dataset Generation

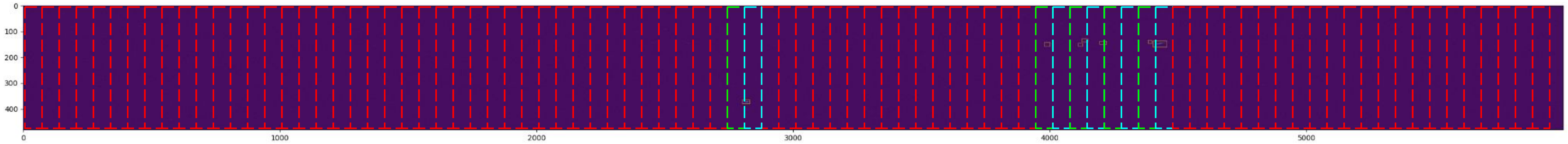
- We pre-processed the data into 480x64 images with bounding boxes (BBs) delineating ν-LE events:
  a. Create a BB at each pixel:
     i. Located at particle of interest and above *noise threshold* = 520.
     ii. Or, above *extend threshold* = 530 and near a particle of interest.
  b. Iteratively merge BBs that are close together.
  c. Slice up 480x6000 image into 480x64 images (or *slices*).
  d. Further processing to make data suitable for training, notably:
     i. Discard ν-LE slices that look too much like NB (but kept for testing).

ν-LE Image Slicing (1/2): Spanning Sets

1. Generate set of slices spanning aggregate of all BBs.



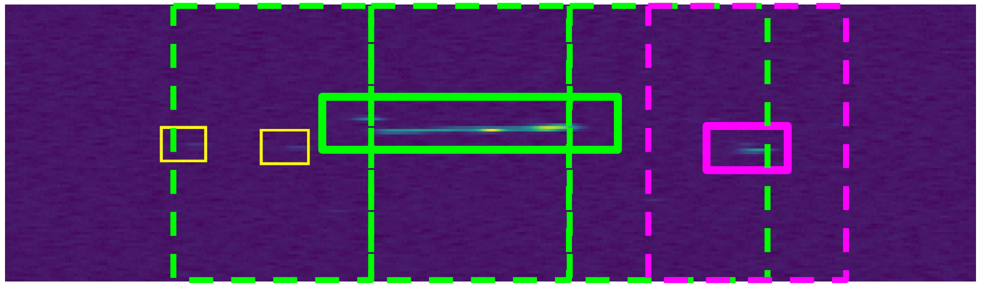
2. Generate set of slices spanning entire 480x6000 image.



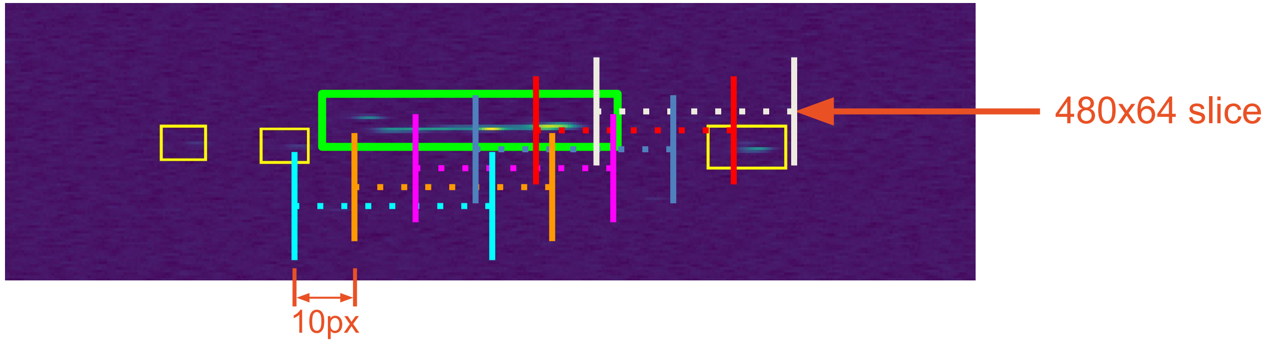
- We discard slices with only background noise (depicted in red).
- Each spanning set is a sequence of slices we may see in practice.
    - We only care that for each set spanning an event, at least 1 slice is correctly identified as ν-LE.
    - *Image-based* metrics: for each set spanning an event, combine predictions from all its slices.
    - *Slice-based* metrics: consider every slice independently.

ν-LE Image Slicing (2/2): Various Views

1. For each BB, generate contiguous set of slices centered around BB.

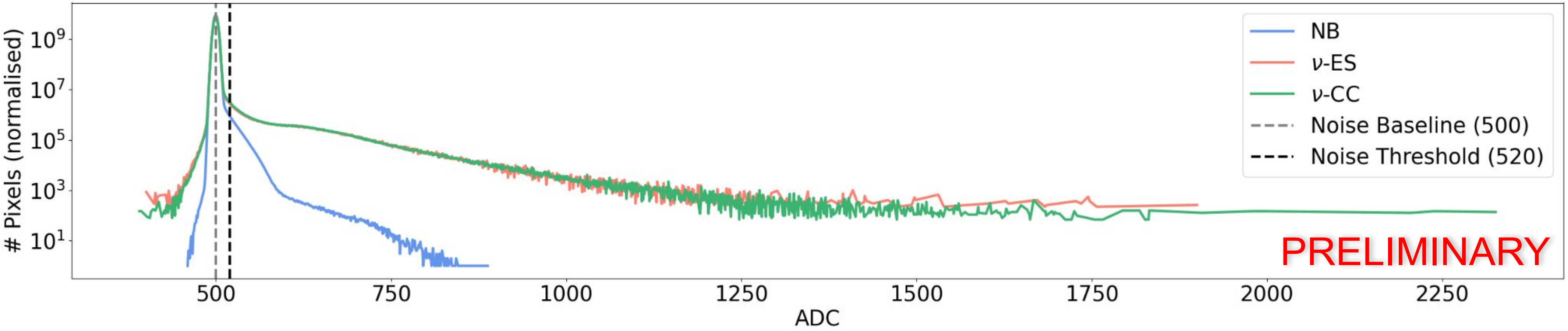


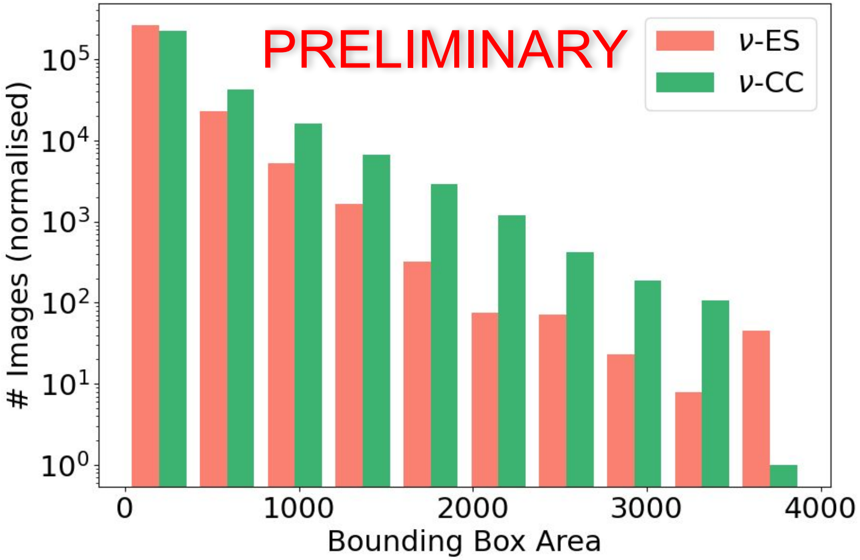
2. For each BB, generate slices each shifted by 10 pixels.



Note: only left-to-right shifted slices shown, repeat from right to left as well.

Pixel & Bounding Box Area Statistics for 480x64 Slices



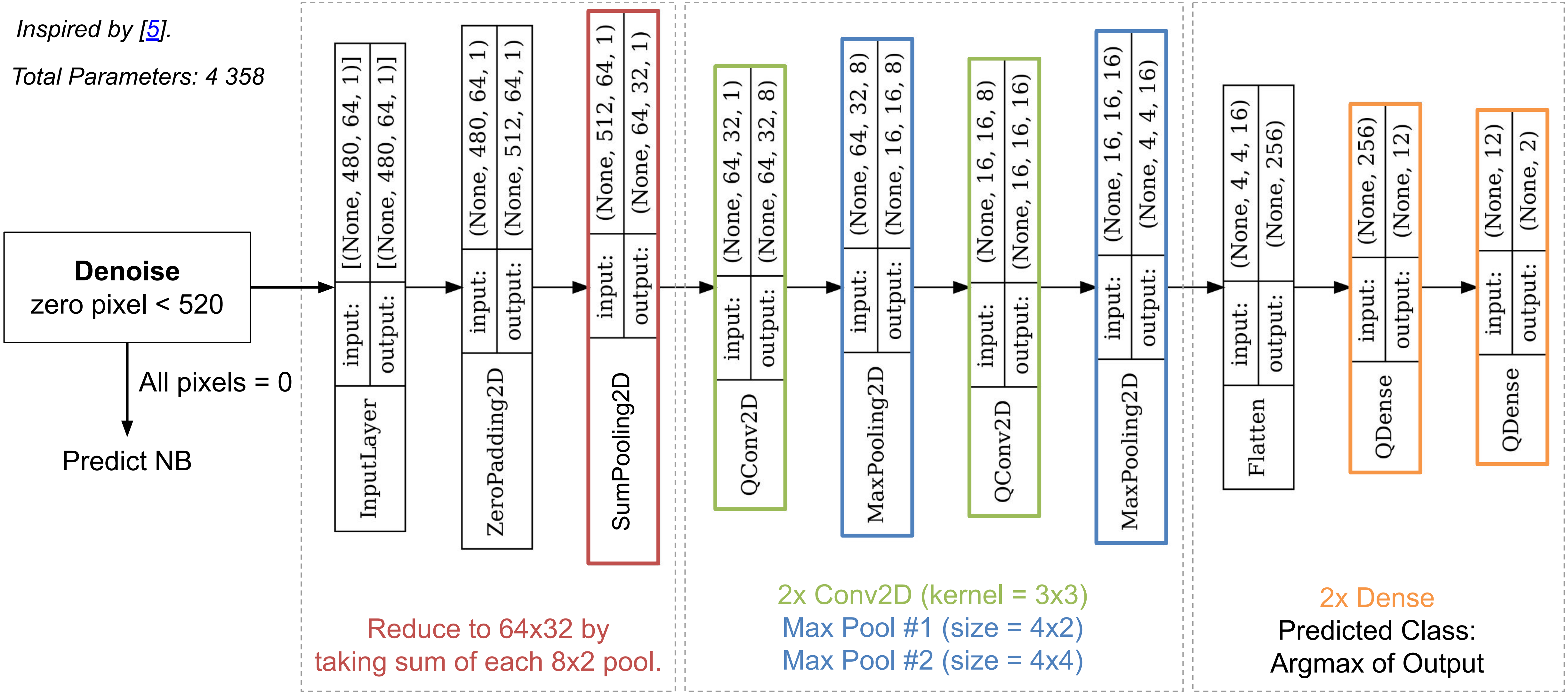


- ν-CC has larger extent on average.
- Pixel distribution for ν-ES is very similar to that of ν-CC.
- Real-time ML models unable to differentiate between ν-ES and ν-CC.
- Hence, low-level trigger classifies {NB, ν-LE}.

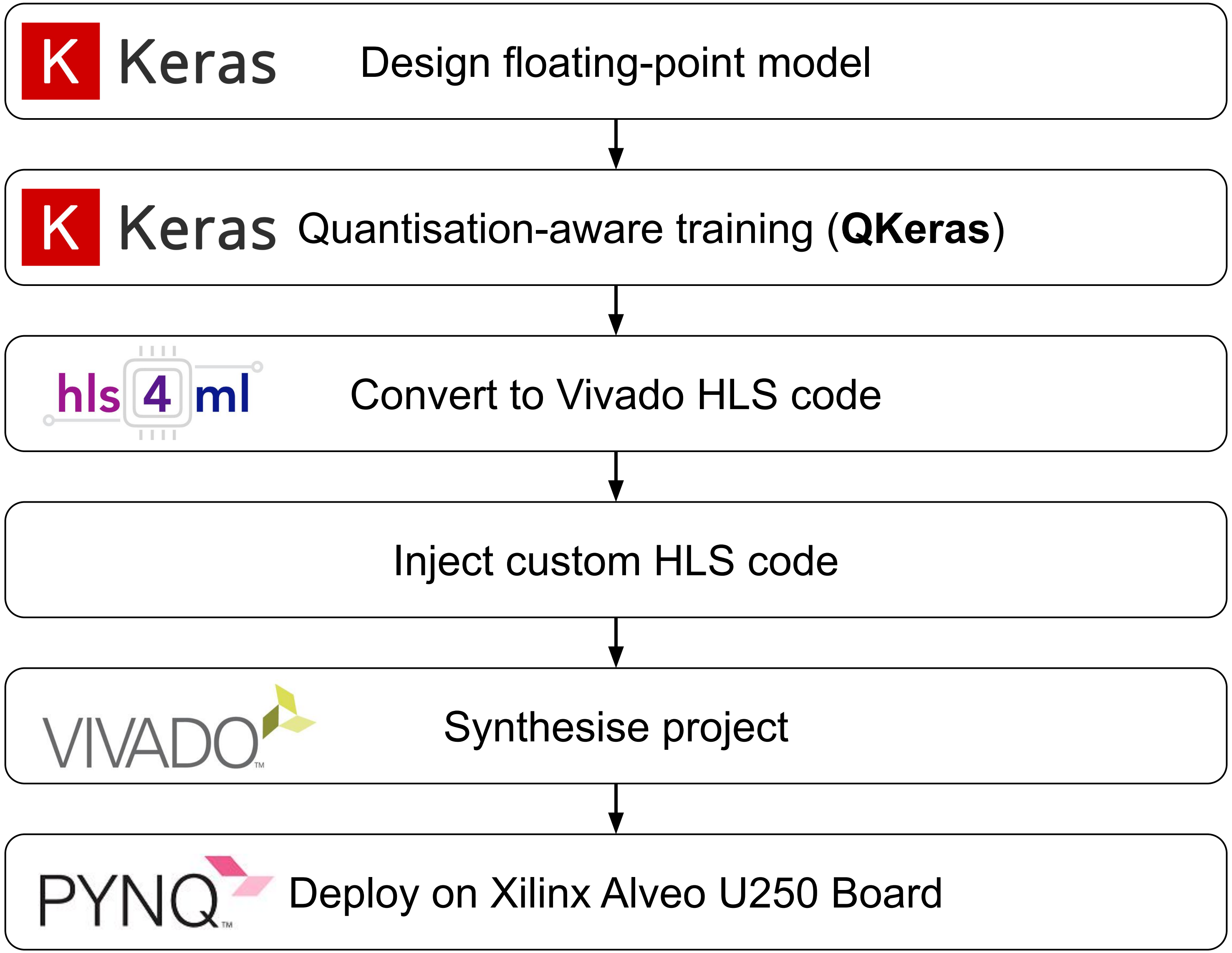
2D CNN Low-Level Trigger

Inspired by [5].

Total Parameters: 4 358



FPGA Deployment Pipeline



Preliminary Results (1/4): Low-Level Trigger

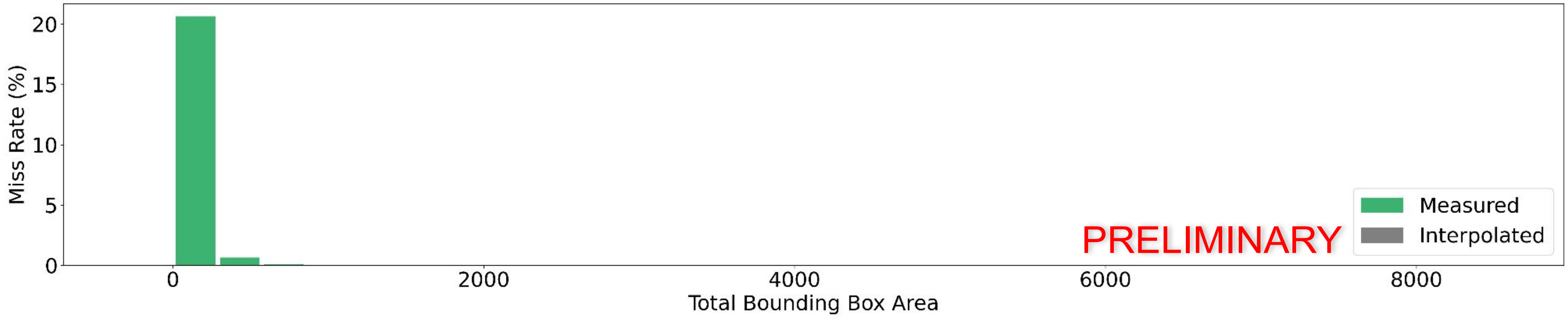
Test Set	NB (%)	LE (%)
(Slices) True NB	99.61	0.39
(Images) True LE	8.16	91.84

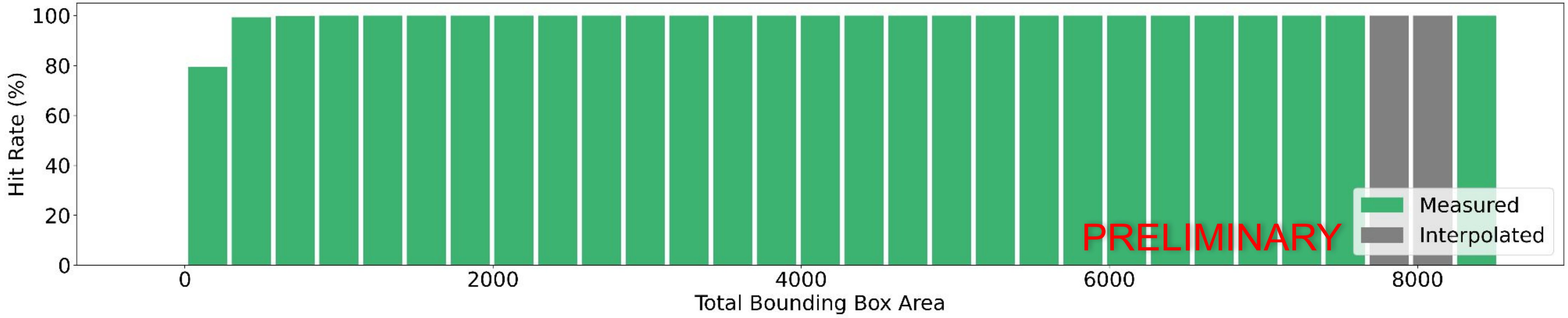
Train+Val+Test Sets	NB (%)	LE (%)
(Slices) True NB	99.61	0.39
(Images) True LE	7.85	92.15

PRELIMINARY

- 99.61% < 99.99% target for NB rejection efficiency (requirement **not** met).
- 91.84% > 90.00% target for ν-LE detection efficiency (requirement met).

Preliminary Results (2/4): Low-Level Trigger





Preliminary Results (3/4): Low-Level Trigger

Resource Utilisation

BRAM	DSP	FF	LUT
306 (5%)	44 (~0%)	48 062 (1%)	104 570 (6%)

Results shown for the Xilinx Alveo U250 board; noting that the Felix board has 76Mb BRAM, 5.5k DSPs, 1.3M FFs, 660k LUTs.

Latency *(clock @ 200 MHz)*

- Fastest unaltered hls4ml implementation: **348 μs** (with 5% LUT utilisation).
- After injecting custom HLS code: **25.18 μs** (with 6% LUT utilisation).
- 25.18 μs ≪ 32 μs latency target.

Summary

- FPGA implementation includes all pre-processing: denoising, downsampling, skipping all-zero slices.
- Resource requirement: **met by a very large margin**.
- Latency requirement: **met by a comfortable margin**.
- Accuracy requirement: **almost met**.

Preliminary Results (4/4): Low-Level Trigger

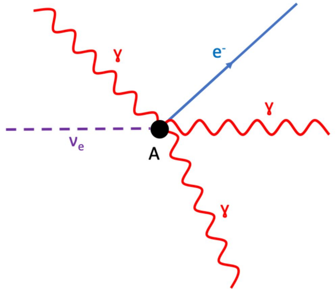
- 99.61% < 99.99% target for NB rejection efficiency (requirement **not** met). To improve:
  - Use a more complex model:
    - Initially downsample to 256x32 with additional Conv2D + MaxPool2D to compensate.
    - Resulted in a 0.03% improvement whilst significantly increasing latency & area utilisation.
  - Discard ν-LE slices from training set that look too much like noise:
    - Iterative procedure:
      1. Train model.
      2. Filter out mispredicted ν-LE slices in training set that meet certain criteria (e.g. BB area ≤ 100).
      3. Repeat from step 1.
    - Increased NB rejection efficiency from 95% to well above 99%.
    - However, makes it more difficult to detect ν-LE events that look similar to noise.
  - In progress:
    - Non-quantised version achieves 99.82% NB rejection efficiency, so weight quantisation needs fine-tuning.
    - Increase noise threshold to increase frequency of all-zero slices (assumed NB).
- Conclusion:
  - NB rejection efficiency is mostly dependent on quantisation, noise threshold, and quality of training data.
  - We will try to improve these to meet accuracy requirements.

Future Work: Supernova “Pointing”

ν-CC

[4]

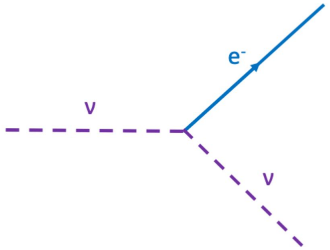
$$\nu_e\ {}^{40}Ar \rightarrow e^-\ {}^{40}K^*$$

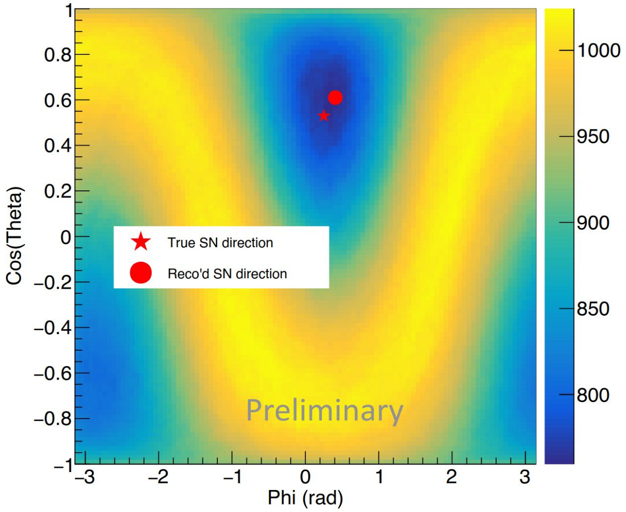


ν-ES

[4]

$$\overset{(-)}{\nu}\ e^- \longrightarrow \overset{(-)}{\nu}\ e^-$$





Example of likelihood function values for 1 supernova sample of elastic scattering events, with radiological backgrounds and noise [4]

- Unlike ν-CC, outgoing e^- direction from ν-ES is highly correlated with incoming ν direction.
- e^- direction from ν-ES can be used to determine direction to supernova (multi-messenger astrophysics).
- Accordingly, we are working on near real-time models for:
  - Precise localisation of ν-LE events (e.g. using a YOLO-like model).
  - Classification between ν-ES and ν-CC.

** in collaboration with Fermilab, Toronto and Duke University researchers*

Conclusion

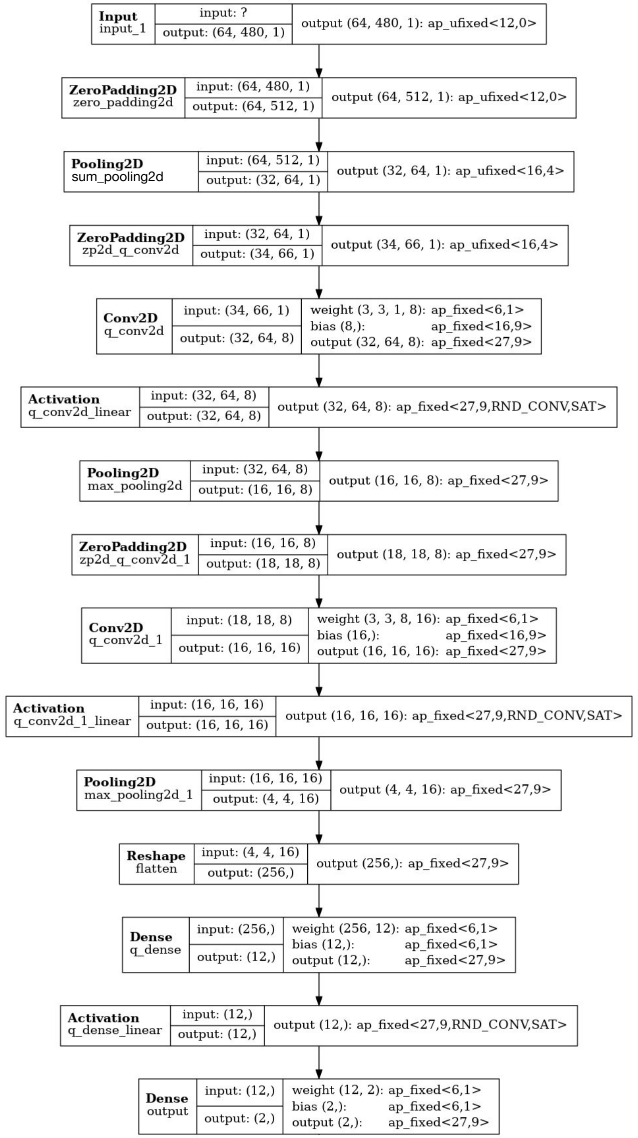
- DUNE provides an excellent opportunity to look for rare off-beam events, notably from supernova neutrinos.

- We have demonstrated the viability of ML-based algorithms for real-time detection of low-energy events:
  - Timing and resource requirements are met.
  - Room to improve accuracy.

- Future work:
  - Supernova pointing.
  - Detect lower-energy neutrinos (e.g. solar neutrinos).

Thank you!

Supplementary: 2D CNN Low-Level Trigger Quantisation



Supplementary: HLS Code Injection

- By default, hls4ml streams in 480x64 input image via a 32-bit AXI bus:
  - Each pixel is a 32-bit floating-point value.
  - Hence, 1 pixel read every clock cycle.
  - Clock period is 5 ns, so minimum latency to read input image = 480*64*5 ns = 153.6 μs.
  - Hence, inference latency ≫153.6 μs ≫32 μs = latency target.

- To meet latency requirements with negligible impact on resource utilisation:
  - Increase AXI bus width to read multiple pixels per clock cycle.
  - Directly send fixed-point pixels instead of floating-point pixels:
    - Data type of input image is `uint12_t` (recall: 12-bit ADCs).
    - Conversion from `uint12_t` to fixed point is essentially free!
      - Reinterpret cast if we keep all 12 bits.
      - Truncate if reduce precision to, say, 9 bits (minimum for denoising).
    - Significantly cheaper than converting from `uint12_t` → float → fixed.
  - Hard code all pre-processing performed before first convolutional layer:
    - Remove redundant computations (e.g. pooling over pad pixels).