Hall C Winter Collaboration Meeting 2026
Flash talk : AI/ML model development using Hall C NPS data

Chi-Kin Tam (CUA)

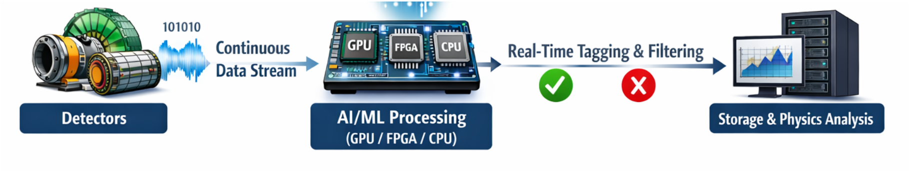
Collaboration : Brad Sawatzky, Dmitry Romanov, Tanja Hornt, Joshua Crafts, Baptiste Fraisse

tamc@cua.edu / ckin@jlab.org

Introduction: Streaming Readout (SRO)

The Paradigm Shift

- **Triggered DAQ:** Decisions made in hardware; only selected data is read out.
- **Streaming Readout:** Continuous, ”triggerless” data stream; all signals are timestamped and moved to a compute buffer.



Conceptual Workflow: Front-End → Network → Software Farm

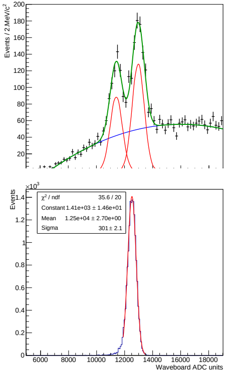
- ▷ **Real-time Analysis:** Eliminates hardware deadtime; Access data from all detectors when making decisions.
- ▷ **Software Flexibility:** Complex algorithms (e.g., AI/ML) can be applied to enhance performance
- ▷ **Standard for EIC:** SRO is the baseline for the Electron-Ion Collider to maximize physics reach.

JLab SRO Tests: Validating the Paradigm

Jlab SRO tests

- On-Beam test at Hall B (CLAS12), Hall D (GlueX).
- Reproduced σ_E in trigger modes and tested AI algorithm.

- ▷ Need new techniques for identifying complex event topologies over the background embedded within the continuous data flow
- ▷ Need to demonstrate AI algorithm for improving the identification of events of interest in high-background environments



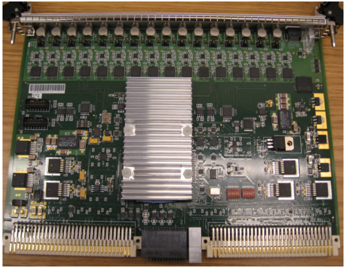
Hall C

- ▷ As a first step existing data of the NPS RG1A experiments, e.g., E12-13-010/007 taken with a loose (VTP) trigger allow for training and optimizing an AI filtering algorithm
- ▷ replicated for the full SRO, e.g., in the next phase of NPS experiments

Hardware for SRO : fADC250 and VTP module

Jlab FADC250

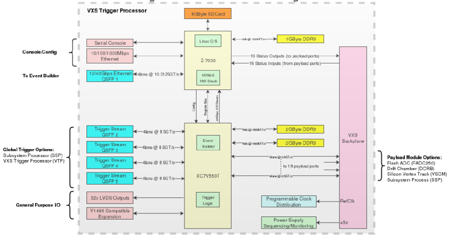
- general purpose triggered readout detector
- adapted for SRO by utilizing the VXS serial links
- charge integration around pulses and send to VTP module



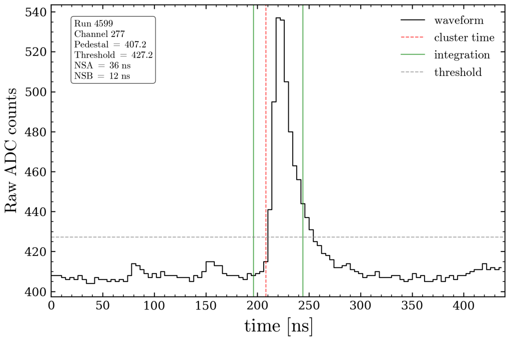
VXS Trigger Processor (VTP) Module

- support 16 FADC250 modules, FPGA-based data processing
- fiber optics serial links to other crates
- 1Gbps Ethernet connection for configuration, 10/40Gbps Ethernet for CODA ROC readout
- high speed readout can be used to read from compatible Jlab VXS electronics

Figure 2a: VTP Hardware Block Diagram



Cluster reconstruction from RAW waveform



$$\text{Pulse integral} = \text{GAIN} \cdot \sum_{n=\text{N}-\text{NSB}}^{n+\text{NSA}-1} (\text{WF}[n] - \text{ped.})$$

Parameter	Value	Unit
GAIN	0.598	MeV/ADC unit
TET	20	ADC unit
PED	~ 400	ADC unit
NSA	12	ns
NSB	36	ns

Emulate fADC250 processing

- fADC250 readout window : 440 ns (110 samples)
- Start from **RAW** waveform
- Pulse finding by leading edge algorithm
- Pulse time used later as the cluster time
- VME software-configured windows NSA, NSB

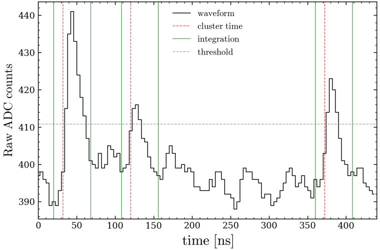
Notes:

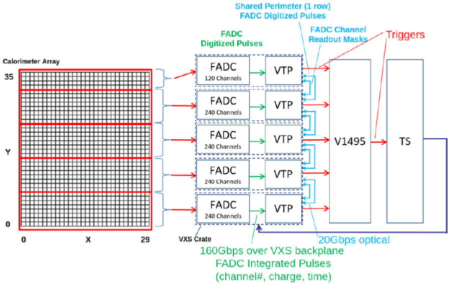
- TET is defined relative to the pedestal.
- PED is channel dependent.

Clustering from RAW waveform : VTP

What VTP module sees :

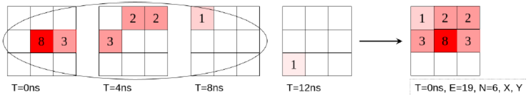
- Energies and rise times of each pulses
- Full $1\ \mu s$ window as opposed to 440 ns





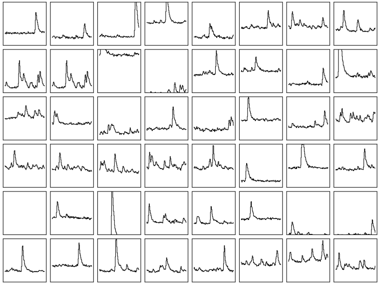
For each 3x3 grid :

- Filter out pulses within 20 ns window. Triggered if
  - At least 1 hit
  - $E_{\text{seed}} > 150\ \text{MeV}$
  - E_{seed} is local maximum

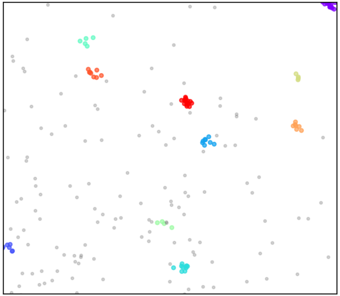


ML Task Definition

- **Input** : Point Cloud of signals with the Waveform as features.
  - ▷ pedestal provided to the model, either from VME config or calibration.
- **Truth** : Cluster IDs, either from VTP or HCana reconstruction.
- **Goal** : Reconstruct clusters directly from RAW waveforms.



ML Model →



Progress

Table: Status of Data Extraction and Machine Learning Implementation

Data Source	**Extracted**	**ML Ready**	**Implementation Notes**
VTP Reconstruction	✓	—	Existing algorithm used
HCANA Reconstruction	✓	✓	Cluster seed recognized
Geant4 Simulation	—	—	Not yet initiated

Note: Detail of the Geant4 Simulation can be found in Avnish's talk earlier (15:15).

Object Condensation

Eur. Phys. J. C 80, 886 (2020)

Attractive loss

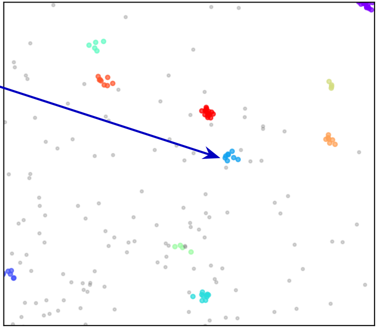
$$V_{\text{attr}}(x) = ||x - \underbrace{x_{\text{seed}}}||^2 q_{\text{seed},k}$$

- Signals associated to the same cluster are pulled towards their cluster seed in latent space.
- minimize latent x distance for points (k) in each cluster.

Repulsive loss

$$V_{\text{repul}}(x) = \max(0, \lambda - ||x - x_\beta||)q_{\beta k}$$

- Signals from different objects are pushed away
- minimized signals from different clusters are well-separated.



Latent space distribution

Object Condensation

Attractive loss

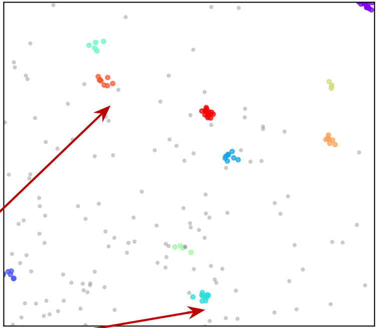
$$V_{\mathrm{attr}}(x) = ||x - \underbrace{x_{\mathrm{seed}}}||^2 q_{\mathrm{seed},k}$$

- Signals associated to the same cluster are pulled towards their cluster seed in latent space.
- minimize latent x distance for points (k) in each cluster.

Repulsive loss

$$V_{\mathrm{repul}}(x) = \max(0, \lambda - ||x - x_\beta||) q_{\beta k}$$

- Signals from different objects are pushed away
- minimized signals from different clusters are well-separated.



Latent space distribution

Object Condensation

Coward Loss

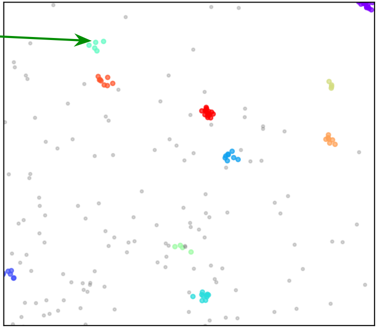
$$V_{\text{coward}} = \frac{1}{K}\sum_i (1-\beta_i)$$

- apply to true cluster seeds only
- encourage cluster seeds to have high $\beta \approx 1$

Repulsive loss

$$V_{\text{noise}} = \underbrace{s_B}_{\text{s/n ratio}} \sum_i n_i \beta_i$$

- apply to background / noise nodes only
- regularize backgrounds to stay $\beta \approx 0$



Latent space distribution

Object Condensation

Coward Loss

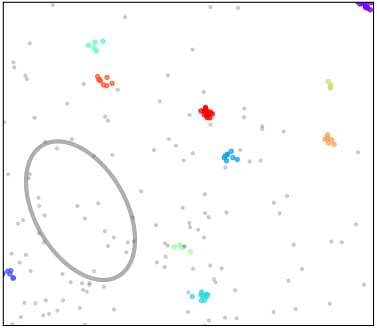
$$V_{\text{coward}} = \frac{1}{K} \sum_i (1 - \beta_i)$$

- apply to true cluster seeds only
- encourage cluster seeds to have high $\beta \approx 1$

Repulsive loss

$$V_{\text{noise}} = \underbrace{s_B}_{\text{s/n ratio}} \sum_i n_i \beta_i$$

- apply to background / noise nodes only
- regularize backgrounds to stay $\beta \approx 0$



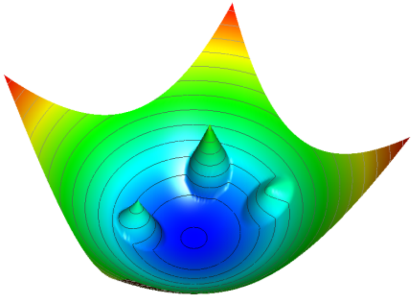
Latent space distribution

Object Condensation

Total Loss function

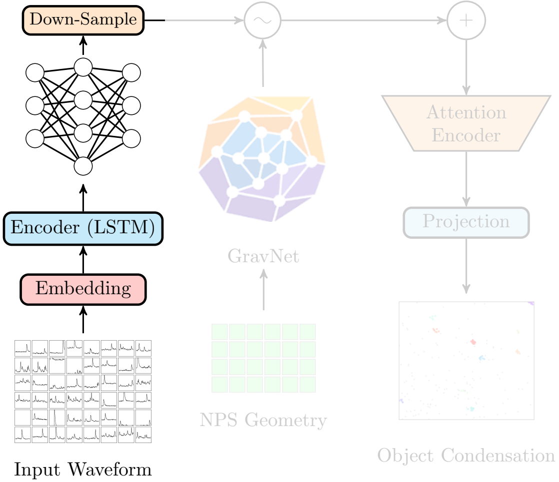
$$\mathcal{L} = aV_{\mathrm{attr}} + bV_{\mathrm{repul}} + cV_{\mathrm{coward}} + dV_{\mathrm{noise}}$$

- Carefully tune hyperparameters a, b, c, d to balance different loss terms.



Eur. Phys. J. C 80, 886 (2020)

Architecture : Encoding Waveform

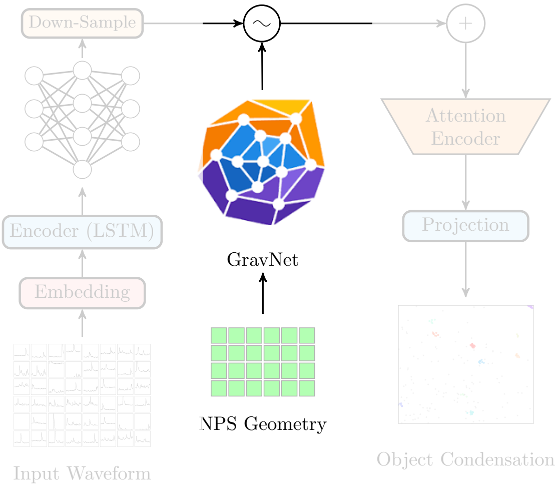


- Input **pedestal-subtracted waveform**, dimension [1080 channels][110 samples]
- Embed waveform sample into higher dimension.

- Long-Short Term Memory (LSTM) network compresses WF into latent feature vectors.
- Usual features in literature : **peak time**, **pulse integral**, **rise time**, **fall time**, **FWHM**, **number of peaks**, etc.

- Down-Sampling the background waveform to reduce memory overhead.
  - For each WF associated with a cluster, keep $3-5$ WFs associated with noise.

Architecture : GravNet



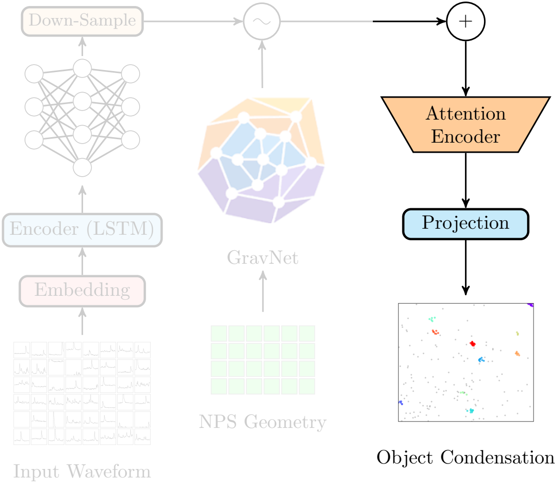
GravNet Layer

- takes the NPS geometry
- project to
  - coordinate in latent space
  - feature space for each vertex
- build kNN graph $\rightarrow$ aggregate features from neighbors scaled by **Grav**itational potential !

$$\overbrace{\mathbf{x}_i^{(k)}}^{\text{node } i} = \gamma^{(k)}\left(\mathbf{x}_i^{(k-1)}, \bigoplus_j \underbrace{\phi^{(k)}}_{\text{func.}}\left(\mathbf{x}_i^{(k-1)}, \mathbf{x}_j^{(k-1)}\right)\right)$$

- used as **Positional encoding** in downstream attention layers

Architecture



Attention Layers

- Input : learned geometry from GravNet + latent WF features
- Assign attention scores $\rightarrow$ correlate different channels

Object Condensation Head

- Compress into object representation
  - latent space coordinate $x \in \mathbb{R}^2$
  - Cluster seedness score $\beta \in (0.0, 1.0)$
- Use Object Condensation loss to train the model.

Result

```
config={
    "epochs": 20,
    "save_dir": save_dir,
    "log_dir": log_dir,
    "mnt_metric": "loss",
    "mnt_mode": "min",
    "save_period": 1,
    "early_stop": 5,
    # trainer config
    "mask_scale": 3.0,
    "q_min": 0.2,
    "noise_idx": 0,
    "margin": 1.0,
    "attr_scale": 3.0,
    "repul_scale": 0.5,
    "coward_scale": 4.0,
    "noise_scale": 0.02,
},
```

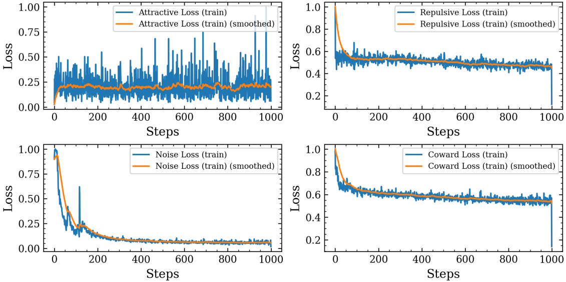
Total Loss function

$$\mathcal{L} = aV_{\text{attr}} + bV_{\text{repul}} + cV_{\text{coward}} + dV_{\text{noise}}$$

- Carefully tune hyperparameters a, b, c, d to balance different loss terms.

- Gradual decrease of $V_{\text{repul}}, V_{\text{noise}}, V_{\text{coward}}$ indicates
  - points from different clusters are well-separated
  - separate real signals from noise
  - recognize center of cluster

- Flat V_{attr} indicates inability to recognize points from the same cluster !



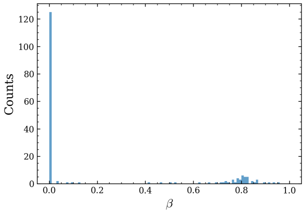
Inference Procedure

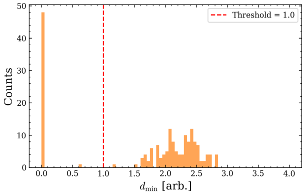
Recall Model Outputs

- Latent space coordinate $x \in \mathbb{R}^2$
- Cluster seedness score $\beta \in (0.0, 1.0)$

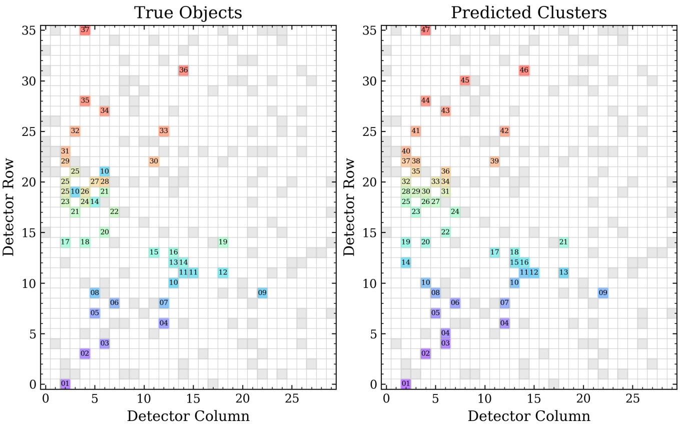
Clustering procedure

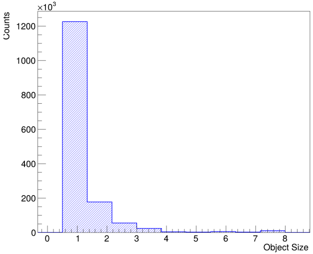
1. Choose hyperparameters $\beta_{\text{threshold}}, d_{\text{threshold}}$
2. Select points with $\beta > \beta_{\text{threshold}}$ as cluster seeds
3. Assign cluster membership based on distance in latent space d
4. Reject and mark as noise if $d > d_{\text{threshold}}$.





Inference Result





- ✓ Able to recognize cluster seeds and background noise.
- × Unable to pull points from the same cluster → data overwhelmed by singleton clusters.

Summary and Outlook

Summary

- ✓ Developed ML-based cluster reconstruction for NPS.
- ✓ Successfully separated noise and identified cluster seeds.
- ✓ **Data Status**: Reconstructed VTP FPGA-triggered cluster memberships are ready for ML training.

Outlook

- → **Verification**: Test on Geant4 simulated clusters using high-level features.
- → **EIC SRO**: Construct streaming readout frames composed of $2\ \mu$s windows.
- → **Model**: Pivot to transfer learning by decoupling waveform encoding and clustering steps.

`JeffersonLab/nps-sro-ml`

Thank you for your attention!