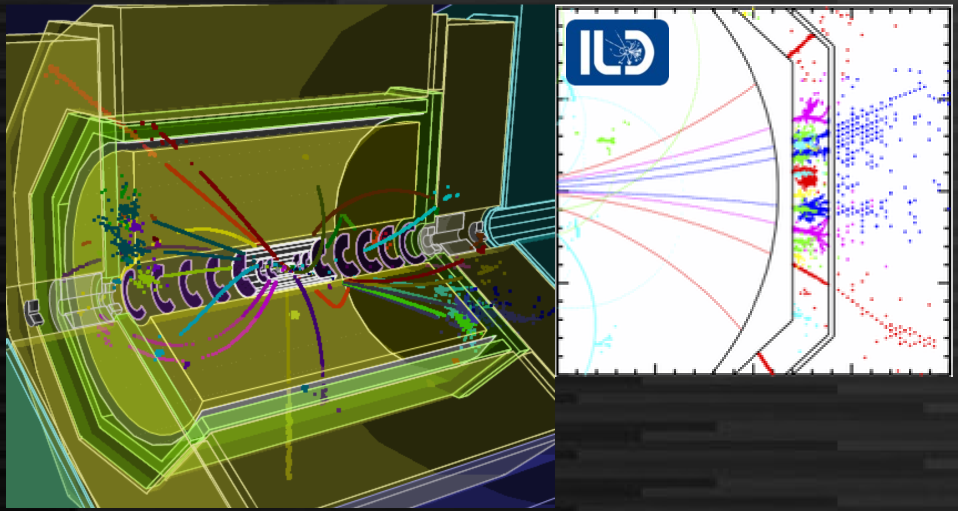
Development of particle flow algorithm with GNN for Higgs factories

Note: This talk includes recent results not fully confirmed, slides may be updated later.

Taikan Suehara / 末原　大幹
(ICEPP, The University of Tokyo)

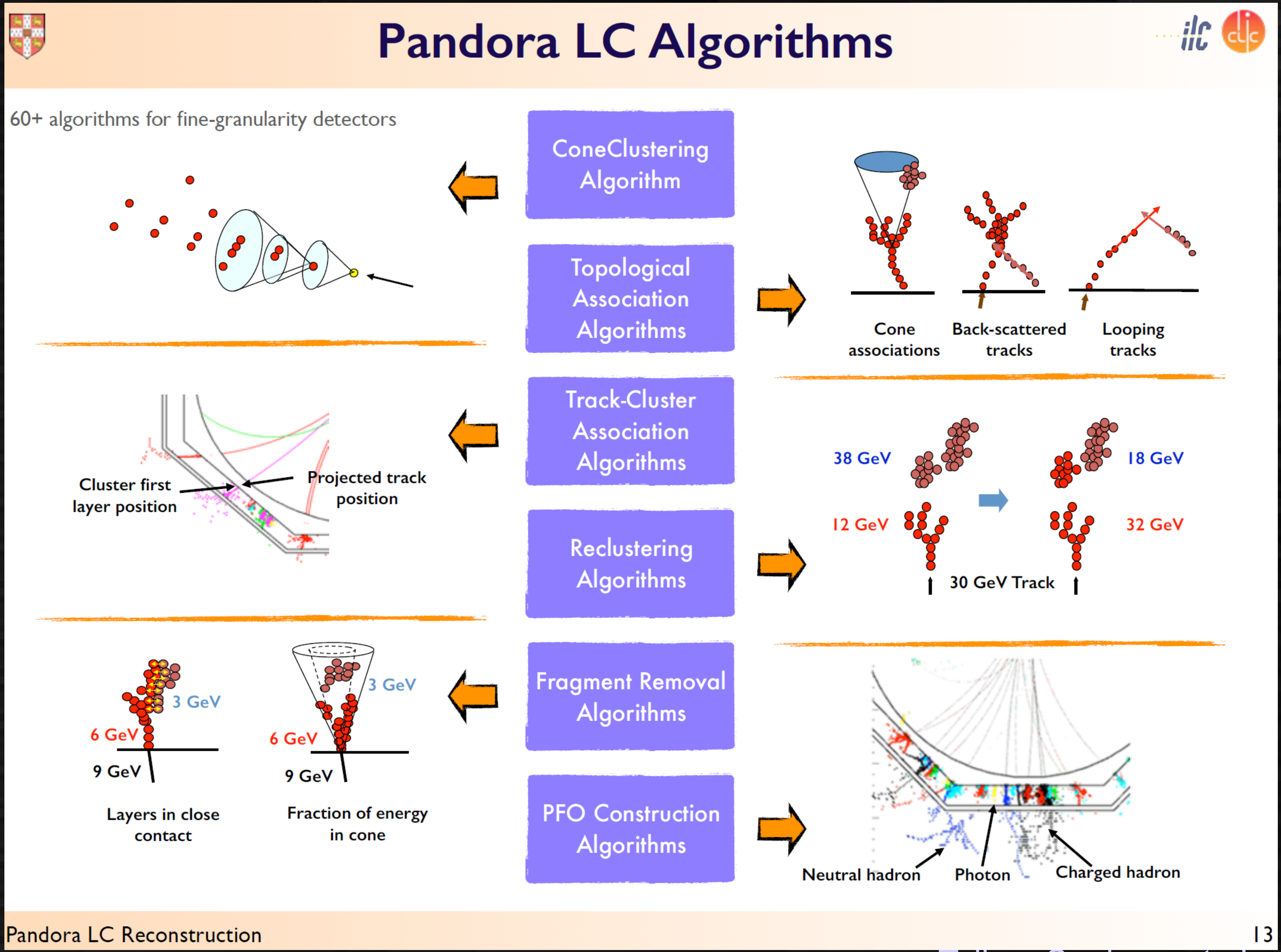
Collaborators: T. Murata (U. Tokyo), T. Tanabe (MI-6 Co.), L. Gray (Fermilab), P. Wahlen (IP Paris & ETHZ / internship at Tokyo)

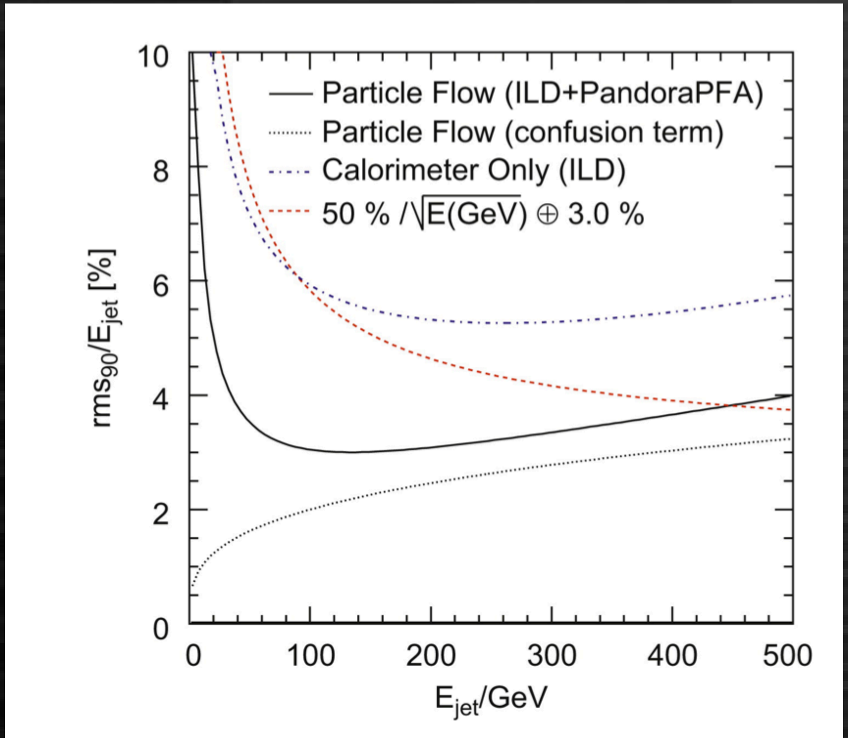
Particle flow for Higgs factories



- High granular calorimetry
  - 3D pixels for imaging EM/hadron showers at calorimeters
    - eg. 10^8 channels for ILD ECAL
  - Separation of particles inside jets
    → ~2x better energy resolution by separation of contribution from charged particles
    - Software algorithm essential (as well as hardware design)
- Particle Flow algorithm
  - Essential algorithm for high granular calorimetry
  - Complicated pattern recognition → good for DNN

Pandora ParticleFlow algorithm



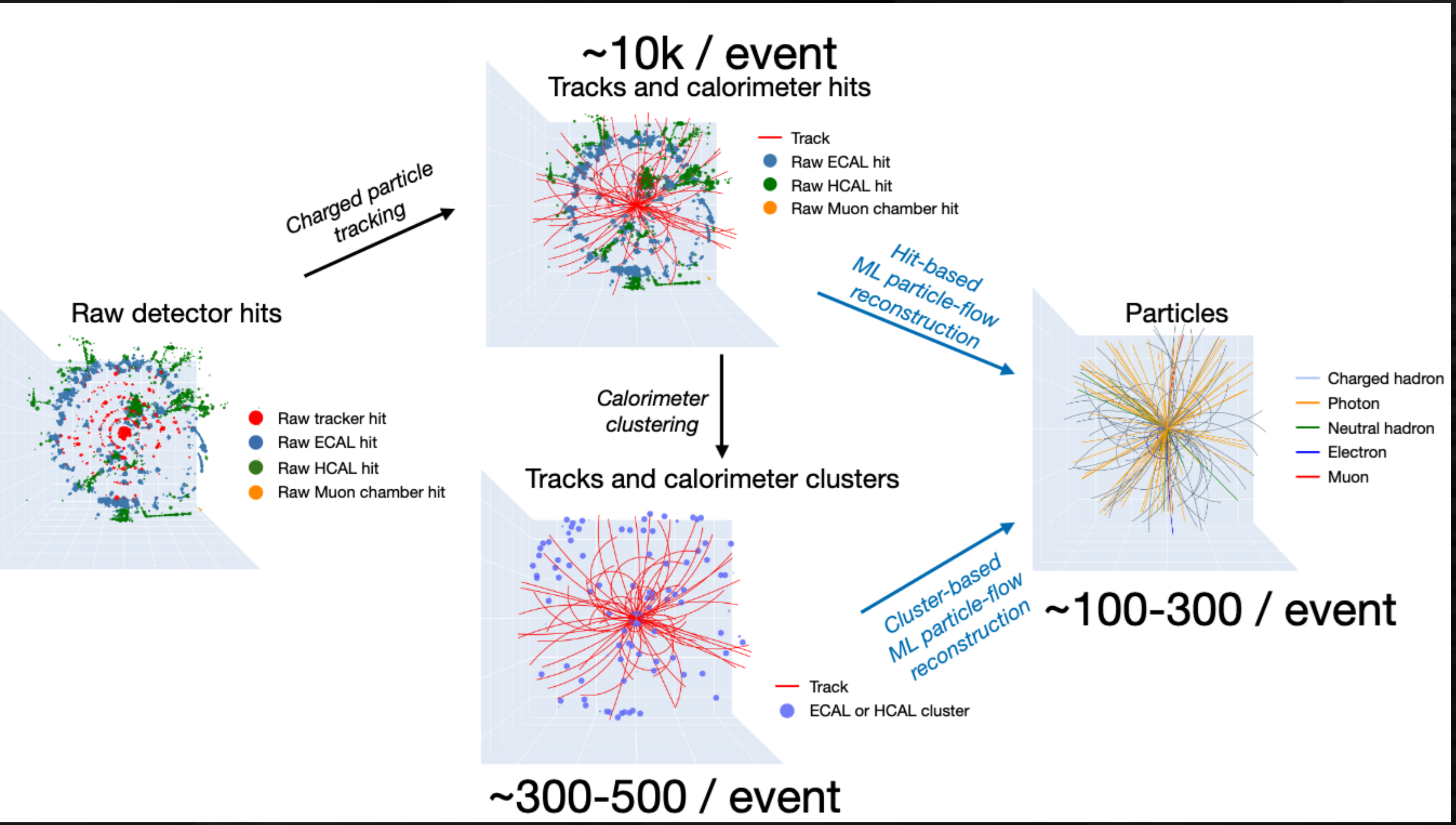


Widely used since 2008
Reasonably good performance up to ~50 GeV jets
Confusion dominates at higher energies

Motivations for DNN particle flow

- Performance improvement
  - Confusion dominant at jet energy > 100 GeV
  - More efficient way to separate cluster from charged particles should be investigated
- Integrate other functions
  - Software compensation, particle ID etc. closely related to PFA
- Detector optimization
  - Comparison with different detector settings
    - PandoraPFA too much depends on internal parameters
  - Effect of timing information to be investigated
    - With different timing resolution (1 ns, 100 ps, 10 ps, …)

Two ways for particle flow with DNN?



Track-cluster matching from calorimeter hits

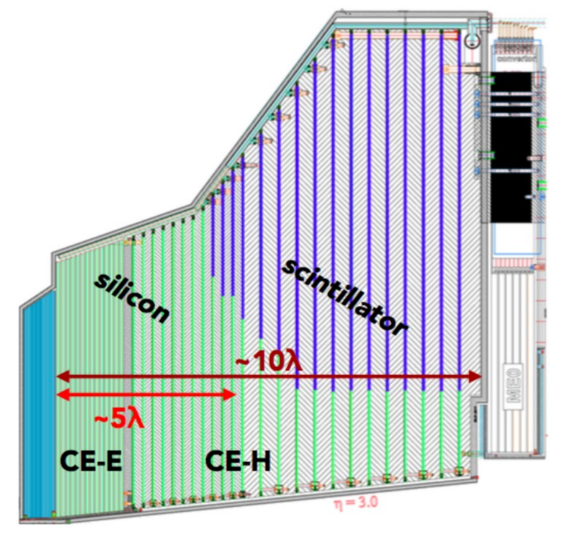
- More freedom
- Distance-based connection more efficient
- We are working this way

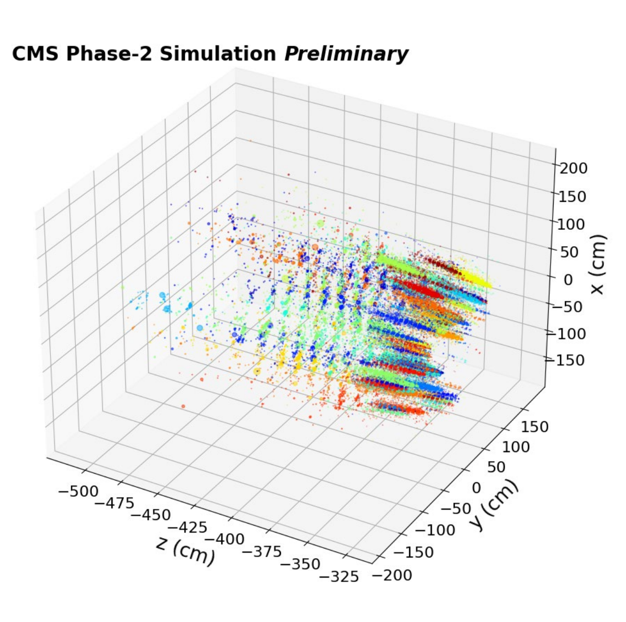
Track-cluster matching from subclusters

- Less input
- Additional clustering algorithm needed

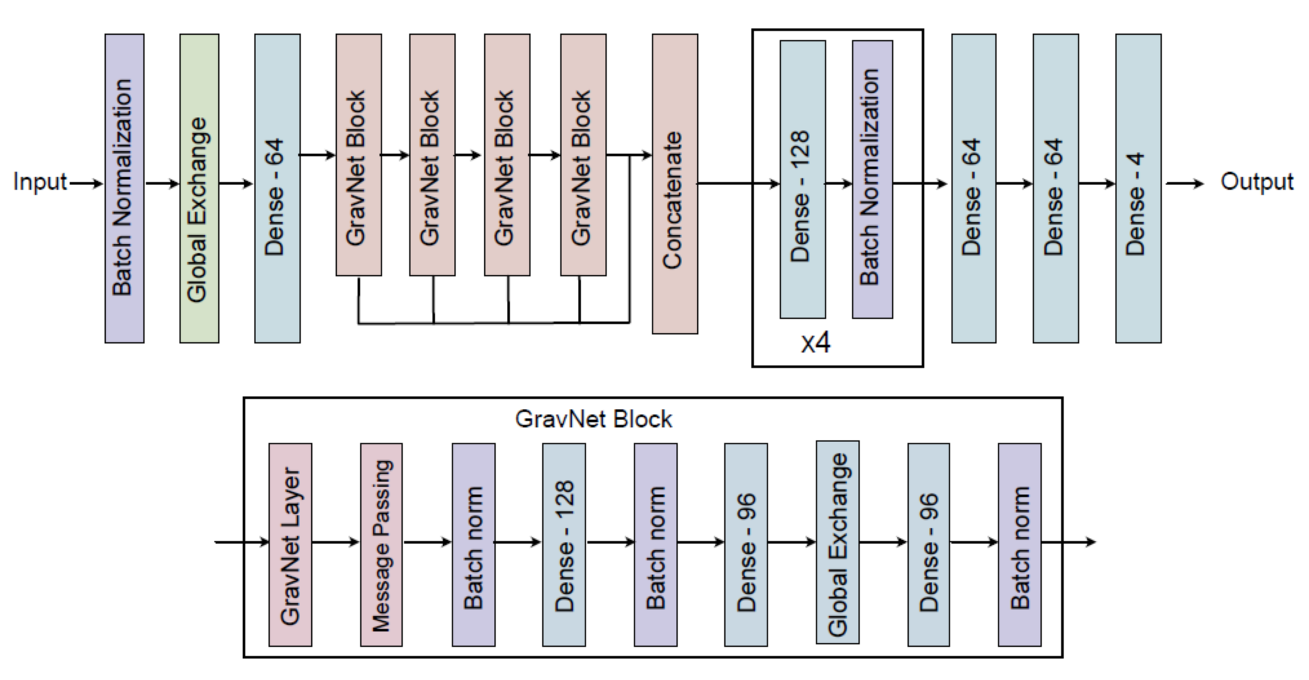
GravNet for CMS HGCAL

- CMS HGCAL
  - High granular forward calorimeter for HL-LHC upgrade at CMS
  - Similar to ILD calorimeter (silicon pixel + scintillator)
    - Inspired by CALICE development
- Reconstruction at HGCAL
  - Pileup/noise to be separated by software
  - Numerous particles from ~200 pileups
    - Difficult to handle: software algorithm critical
    - DNN reconstruction being investigated
      - Reasonable performance obtained up to ~50 pileups?





The network



Rather complicated network with ~30 hidden layers

"Object condensation" loss function is applied (shown in next page)

Input/output obtained for each hit at calorimeter

Input: Features at each hit (position, energy deposit, timing)

Output: "condensation coefficient" β, position at virtual coordinate (2-dim)
optional output of features such as energy, PID (not used now)

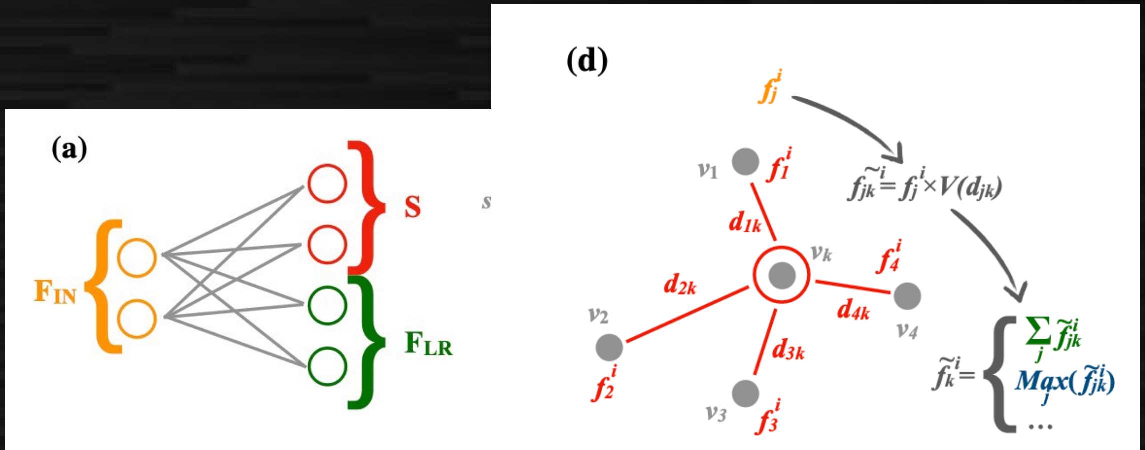
Dense (fully-connected layer) inside each hit, GravNet connects hits

GravNet and Object Condensation

GravNet

arXiv:1902.07987

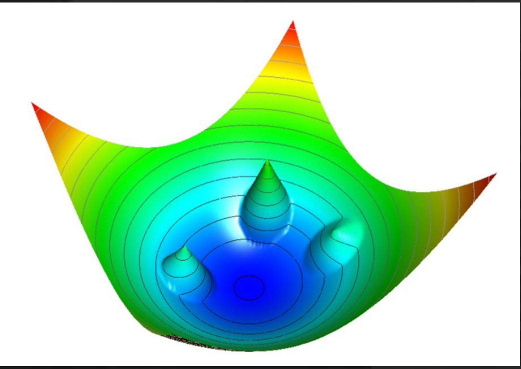
- The virtual coordinate (S) is derived from input variables with simple MLP
- Convolution using "distance" at S (bigger convolution with nearer hits)
- Repeat 2 times and concatenate the output with simple MLP



Object Condensation (loss function)

arXiv:2002.03605

$$L = L_p + s_C(L_\beta + L_V)$$



- Condensation point: The hit with largest β at each (MC) cluster
- L_V: Attractive potential to the condensation point of the same cluster and repulsive potential to the condensation point of different clusters
- L_β: Pulling up β of the condensation point
- L_p: Regression to output features (energy etc.) → currently not used

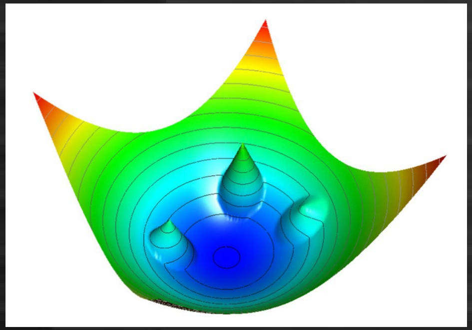
What we implemented: track-cluster matching

- PFA is essentially a problem “to subtract hits from tracks”
- HGCAL algorithm does not utilize track information
  - Only calorimeter clustering exists
- Putting tracks as “virtual hits”
  - Located at entry point of calorimeter
  - Having “track” flag (1=track, 0=hit)
  - Energy deposit = 0
- Modification on object condensation to forcibly treat tracks as condensation points (details next page)
  - Also modifying clustering algorithm to avoid double-track clusters

Current number of parameters: ~420K

Object condensation and our implementation

Object condensation loss function (the function to minimize)



$$L = L_p + s_C(L_\beta + L_V)$$

- Condensation point: The hit with largest β at each (MC) cluster
  → For each MC cluster having a track,
  the track is forcibly the condensation point regardless of β
- L_V: Attractive potential to the condensation point of the same cluster and repulsive potential to the condensation point of different clusters (no modification)
- L_β: Pulling up β of the condensation point (up to 1)
  (no modification, but β of tracks become spontaneously close to 1)
- L_p: Regression to output features (energy etc.) → currently not used

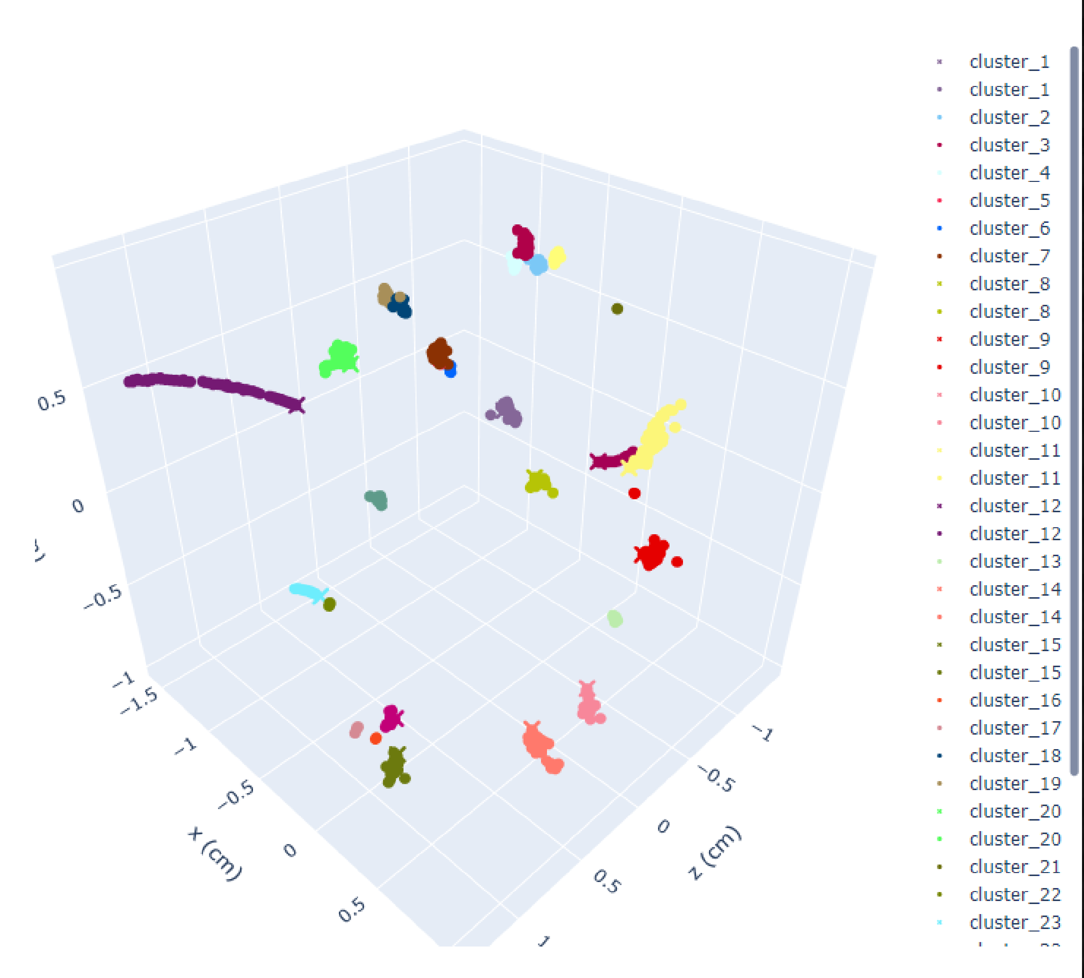
Our samples for performance evaluation

- ILD full simulation with SiW-ECAL and AHCAL
  - ECAL: 5 x 5 mm^2, 30 layers, HCAL: 30 x 30 mm^2, 48 layers
  - Taus overlayed with random direction
    - 100k events, 10 GeV x 10 taus / event → 1 million taus
    - 1M events with variable energies produced, to be tested
  - qq (q=u, d, s) sample at 91 GeV
    - ~75k events
    - Official sample for PFA calibration (other energies available)
  - Converted to awkward array stored in HDF5 format
    - A few 10 GB each

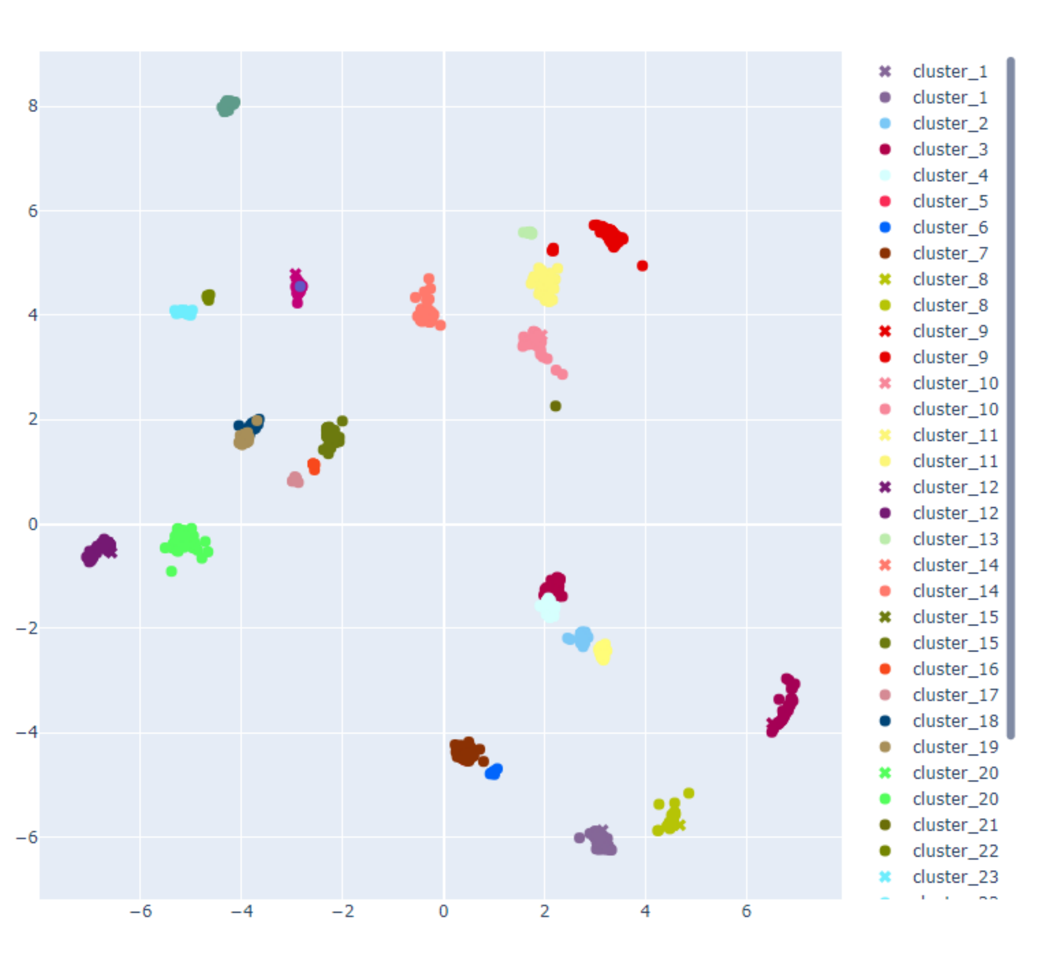
Taus: good mixture of hadrons, leptons and photons with some isolation
Good for training

Event display – looks working

10 Taus @ 10 GeV each



Real 3D coordinate



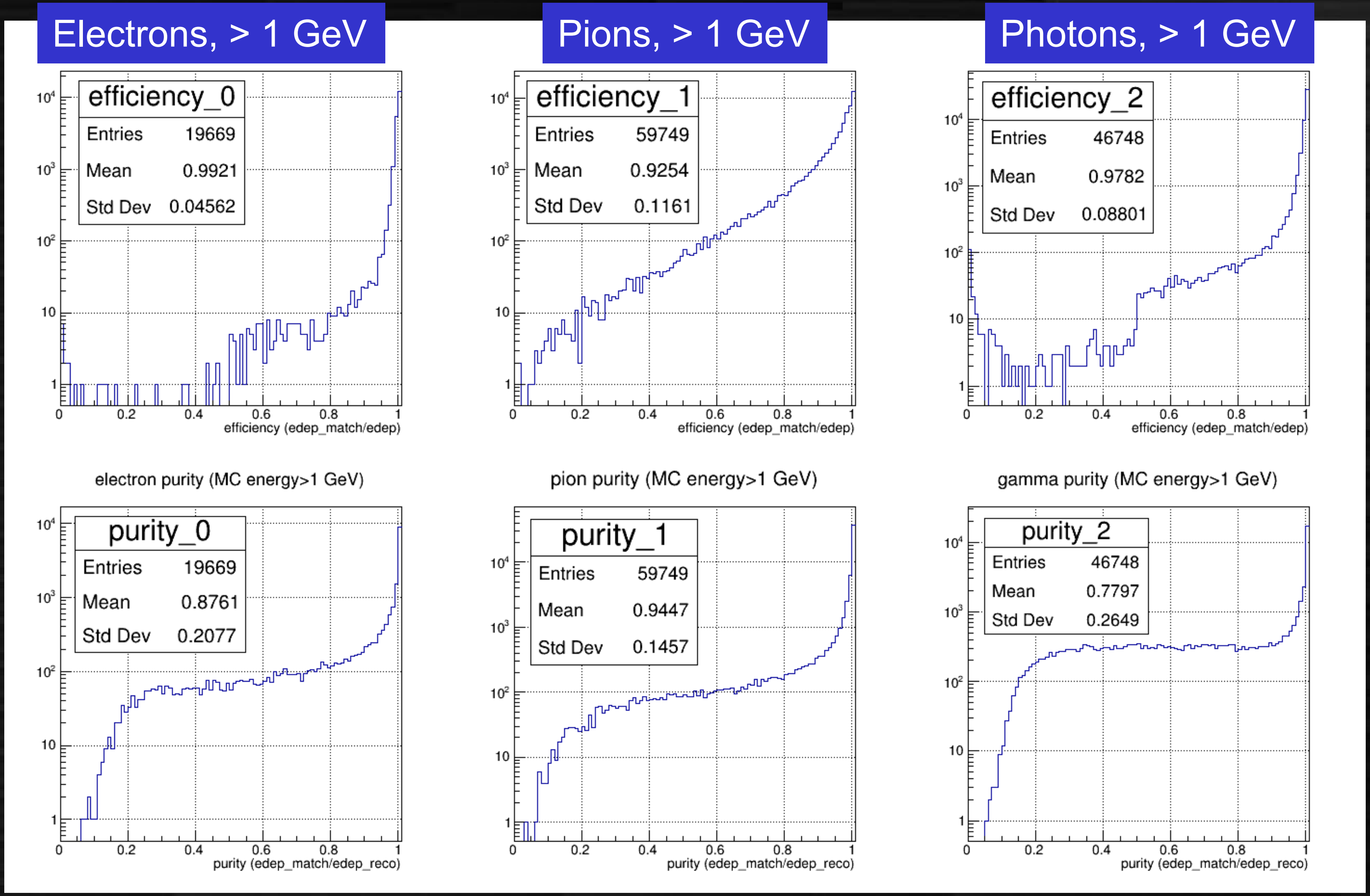
Output from GNN

Quantitative evaluation

- Make 1-by-1 connection of MC and reconstructed cluster
  - Reconstructed cluster with highest fraction of hits from the MC taken
  - Multiple reconstructed cluster may connect to one MC cluster
    - The other way does not occur
- Define 3 variables for each MC cluster
  - Edep: total energy deposit of MC cluster
  - Edep_reco: total energy deposit of matched reconstructed cluster
  - Edep_match: total energy deposit of matched reconstructed cluster included in the MC cluster
- Efficiency: edep_match / edep
- Purity: edep_match / edep_reco

All results from next page are preliminary

Efficiency & purity for GNN, tau train/tau pred



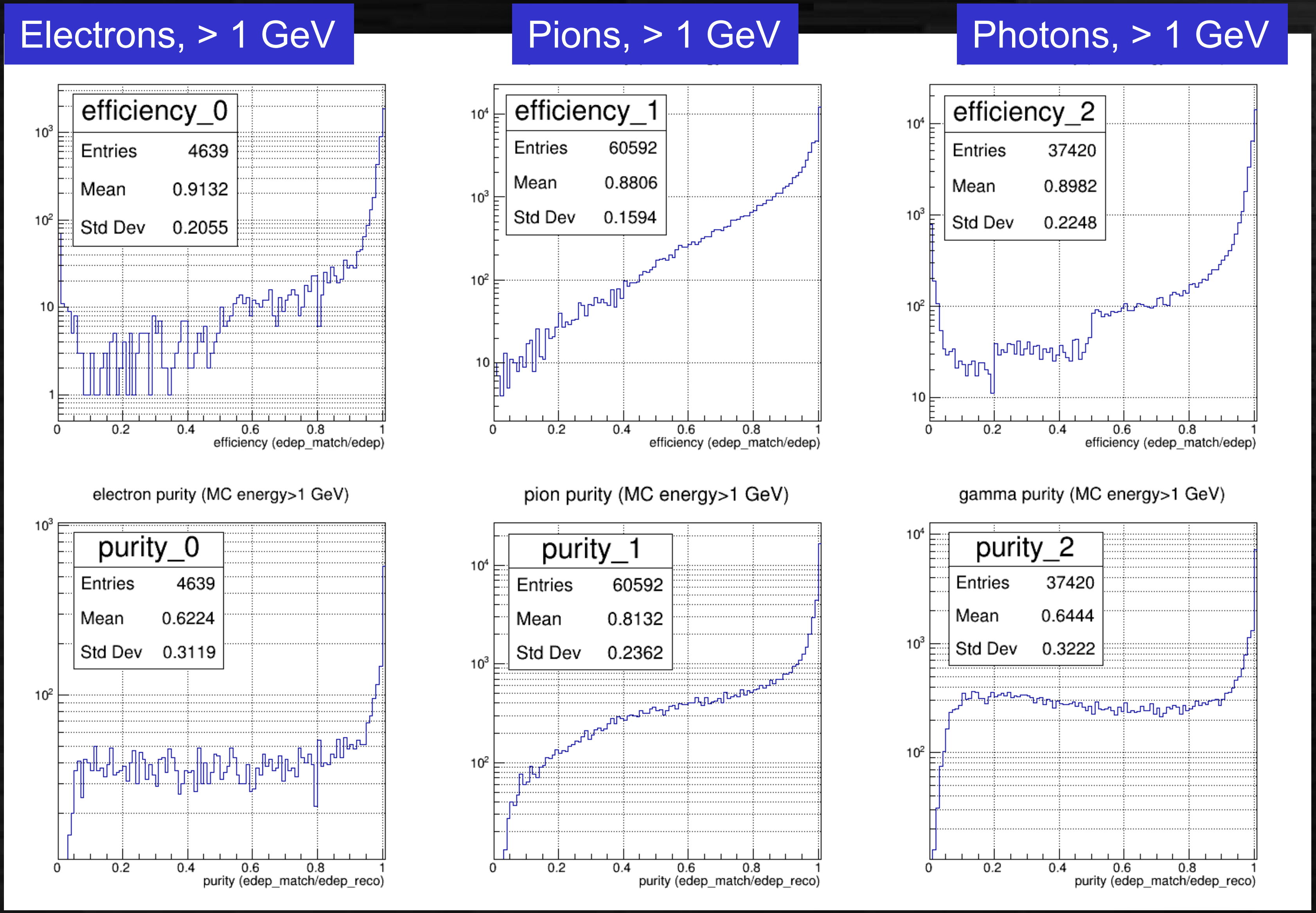
Efficiency:
>90% for all particles
slightly low in pions

Purity:
>85% for all tracks
78% for photons
→ merged photons?

Reasonably well reconstructed!

Preliminary

Efficiency & purity for GNN, tau train/qq pred

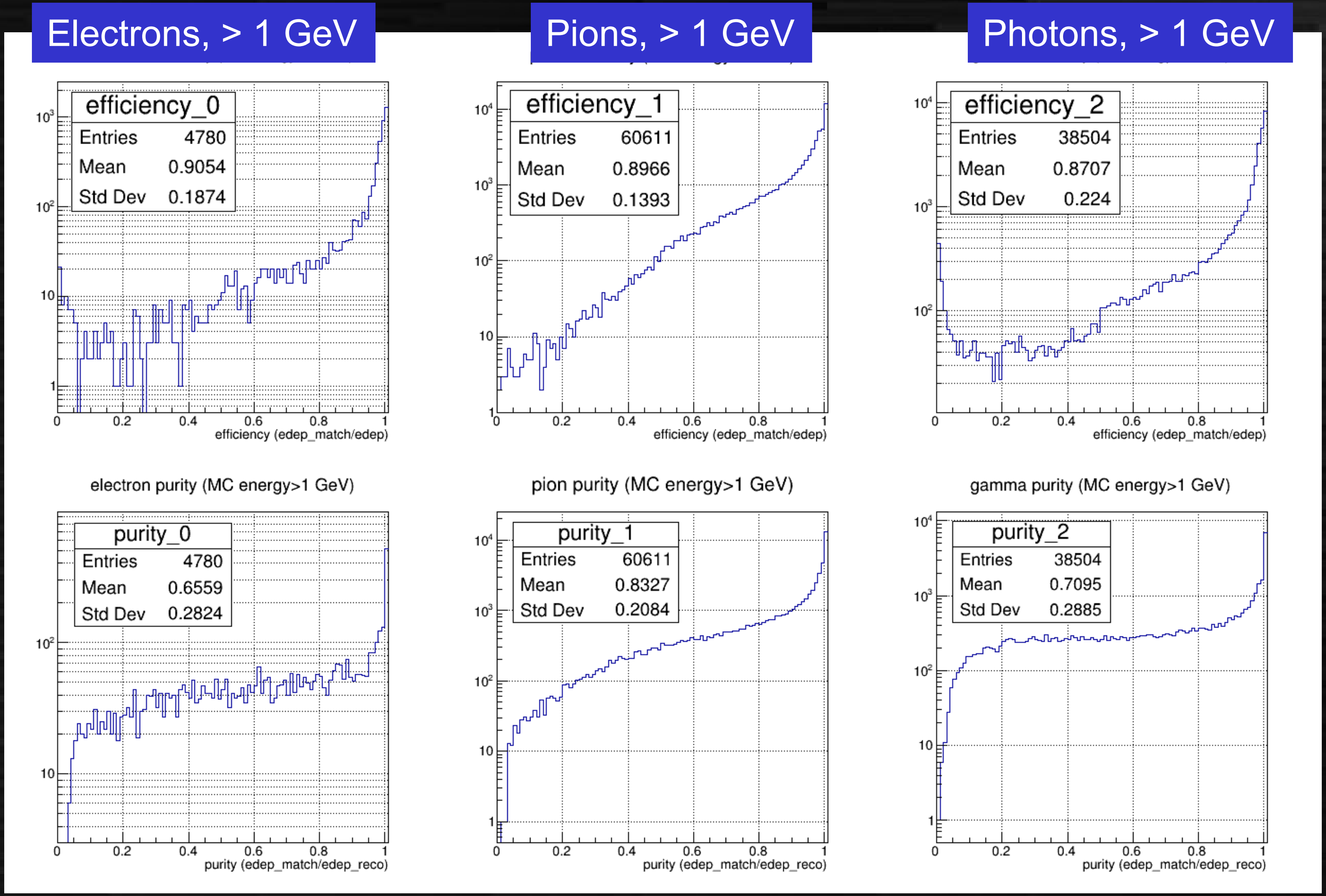


Efficiency:
>88% for all particles
slightly worse than taus

Purity:
Slightly worse in pions
Significantly worse in electrons/photons

Preliminary

Efficiency & purity for GNN, qq train/qq pred

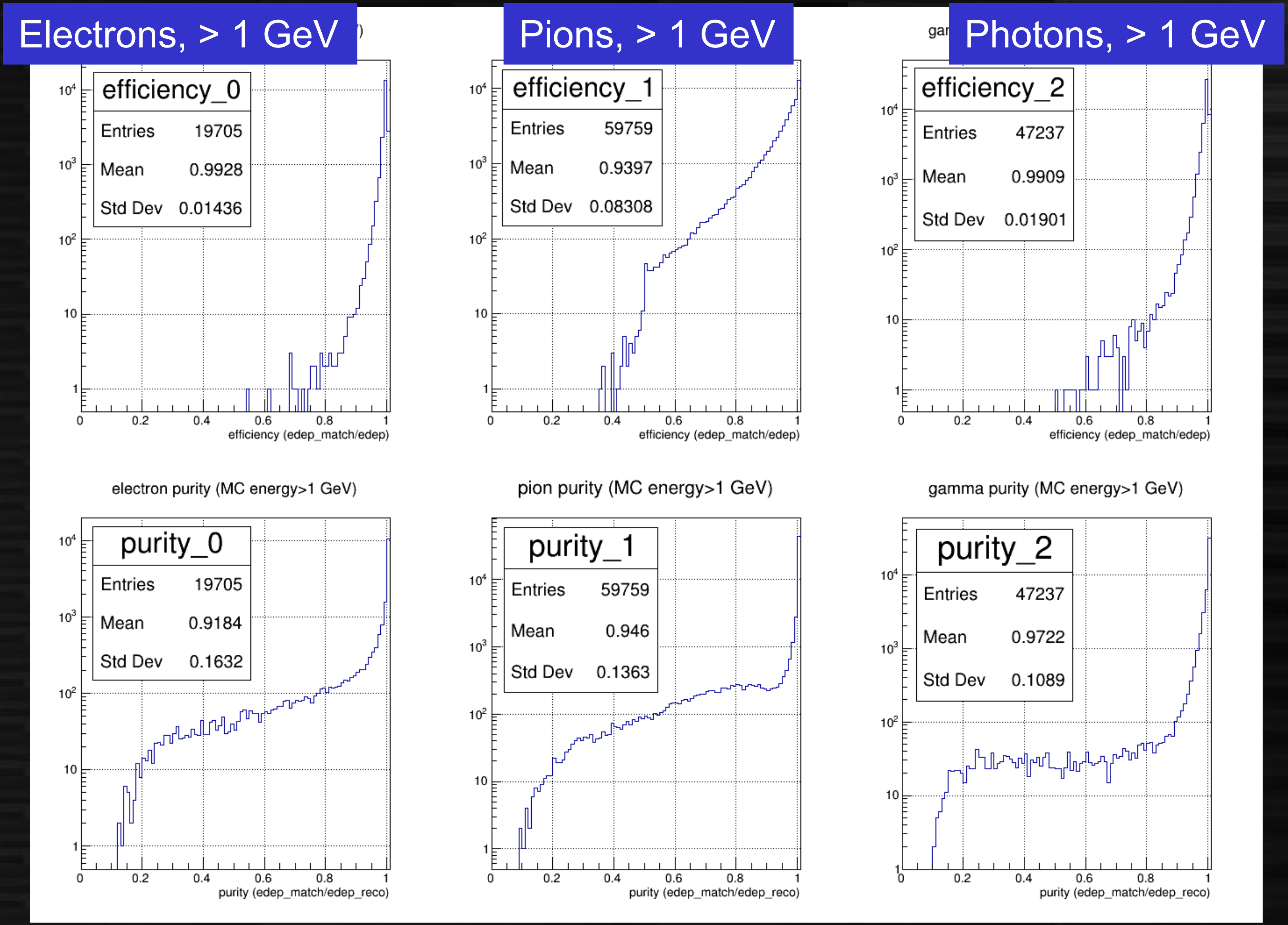


Efficiency:
Similar to tau training
Strong to different type of events

Purity:
Slightly better than tau training

Preliminary

Efficiency & purity with Pandora, ntau events

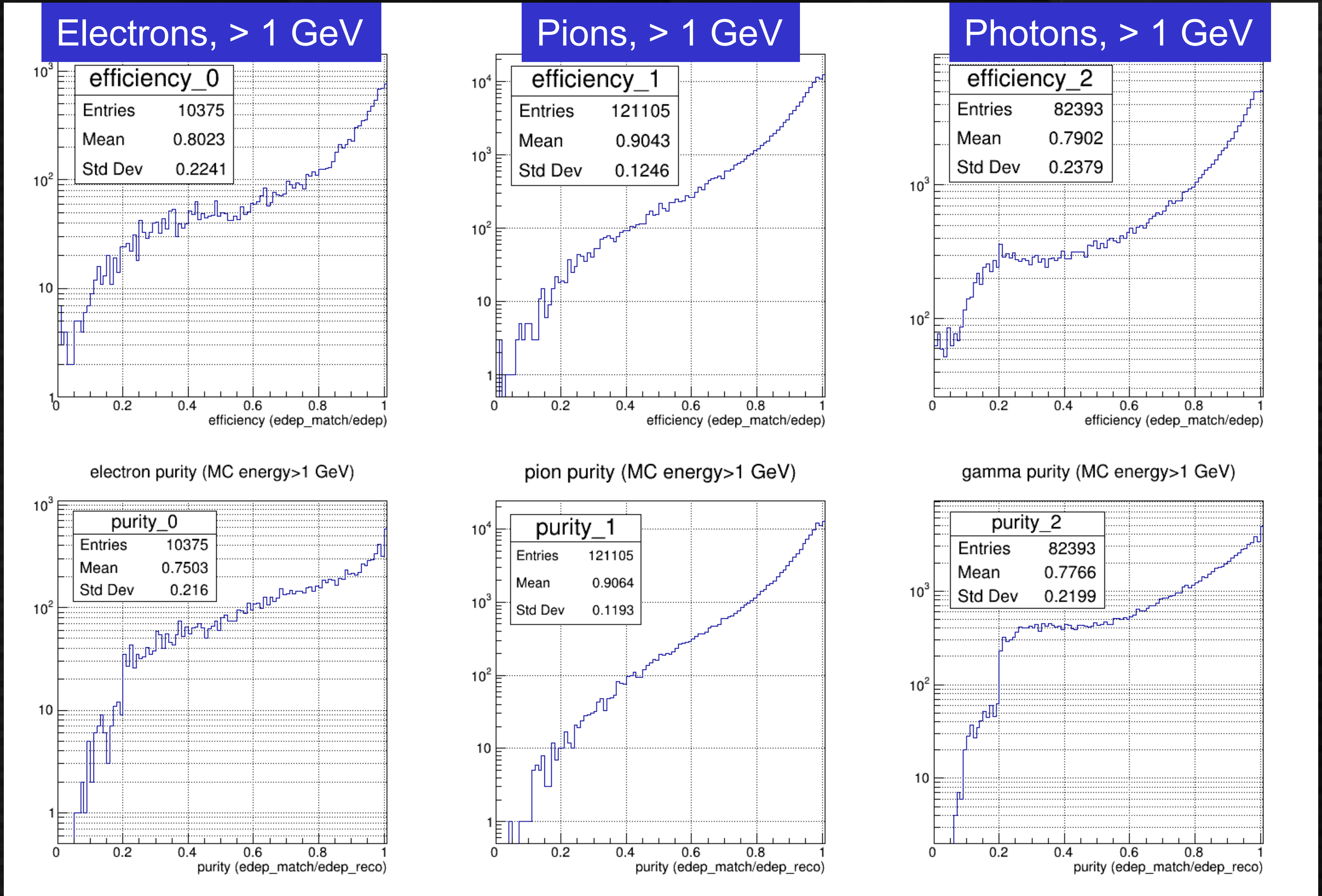


Efficiency and purity for pion is similar to GNN

Pandora is still better in photon reconstruction (esp. in purity)

Preliminary

Efficiency & purity with Pandora, qq events



Similar performance with GNN method obtained

Inconsistency with analysis using MC-cluster matching implemented in official software (ILCSoft)

Need to check definition of MC particles/tracks

Preliminary

Comparison of results (> 1 GeV)

Preliminary

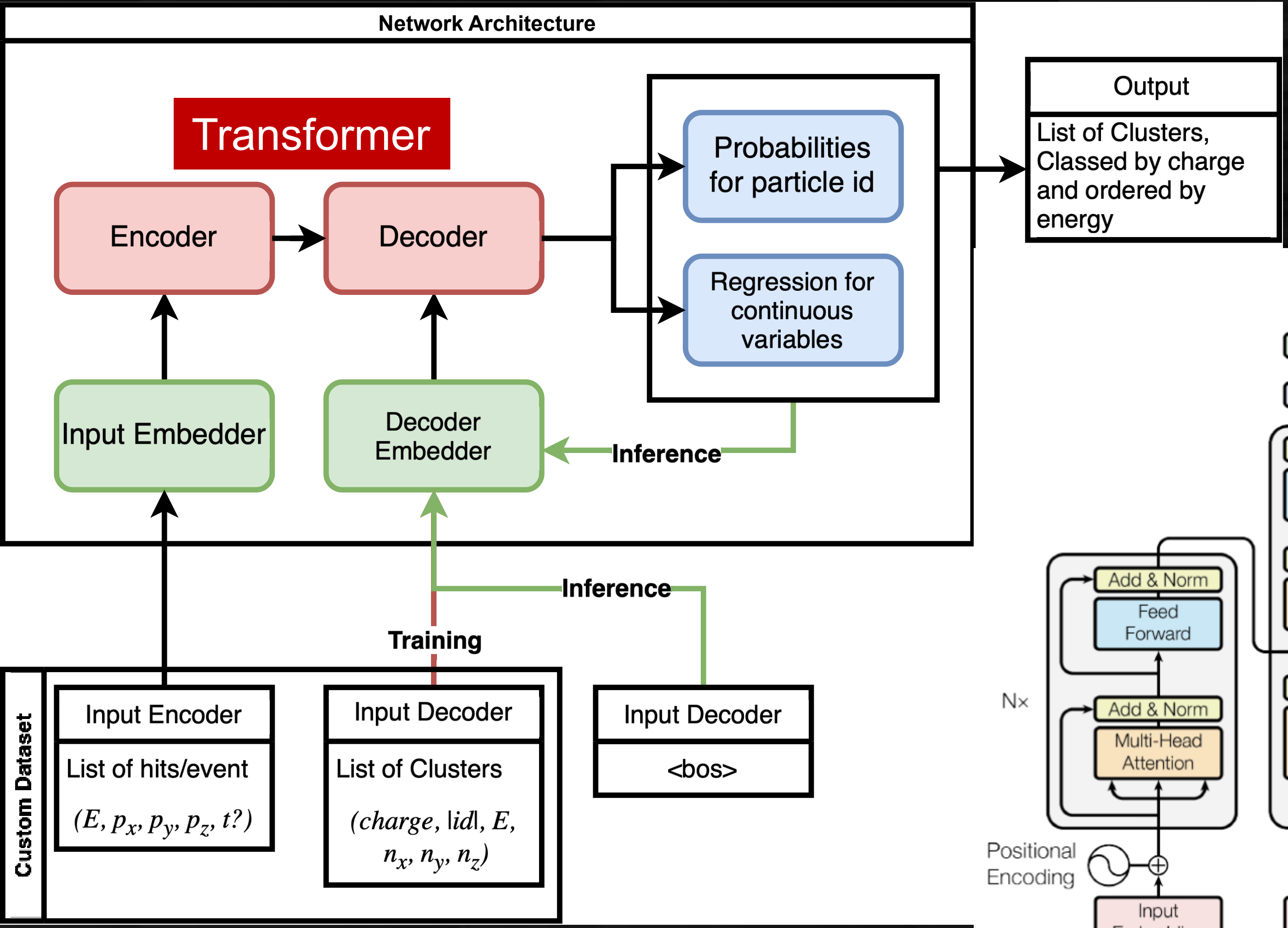
Algorithm train/test	Electron eff.	Pion eff.	Photon eff.	Electron pur.	Pion pur.	Photon pur.
GravNet 10 taus/10 taus	99.2%	92.5%	97.8%	87.6%	94.5%	78.0%
GravNet 10 taus/jets	91.3%	88.1%	89.8%	62.2%	81.3%	64.4%
GravNet jets/jets	90.5%	89.7%	87.1%	65.6%	83.3%	70.9%
PandoraPFA 10 taus	99.3%	94.0%	99.1%	91.8%	94.6%	97.2%
PandoraPFA jets	80.2%	90.4%	79.0%	75.0%	90.6%	77.7%
PandoraPFA jets (ILCSoft)	96.7%	95.5%	96.4%	97.1%	90.4%	97.7%

Still too early to conclude, but performance of GNN comparable to PandoraPFA at least on pions, which have less uncertainty related to MC truth definitions

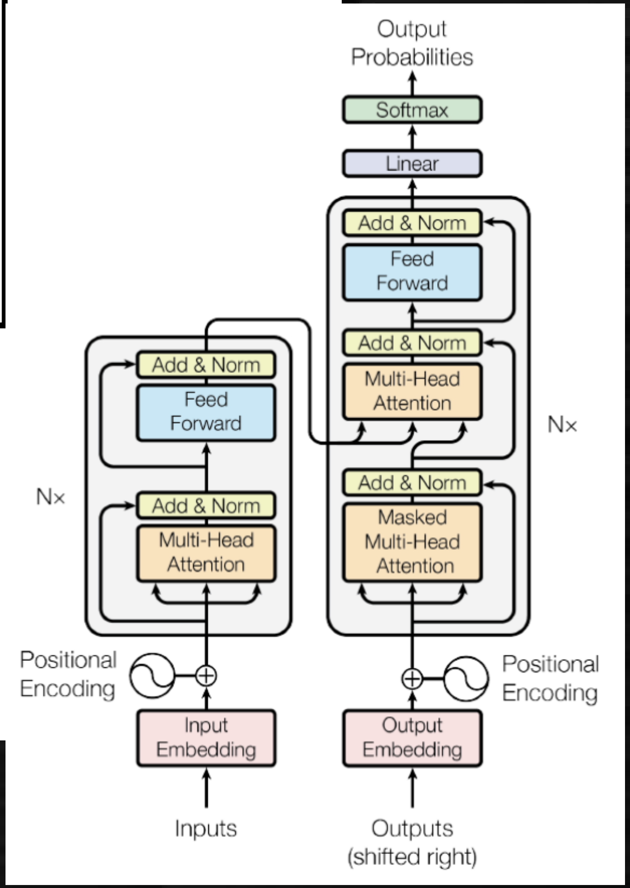
Plans for further development

- Optimizing network/input
  - Improving MC truth matching (kink tracks, photon emissions from tracks etc.)
  - Output dimension for clustering: currently 2, may be higher
  - Dependence on input sample size
    - Also number of parameters of the network
  - Other hyperparameters like learning rate etc.
  - Training with mixture of taus/jets?
- Clustering method: also a place to use NN
  - Currently applying simple clustering to collect hits around high-beta hits
- Performance study on jet energy resolution (target)
- Utilization of timing information
- Another NN: transformer (next page)

More NLP-like model: transformer



Planned structure for PFA



Transformer

Transformer: training relation among elements (hits in PFA) with (multi-head) self-attention mechanism (used in GPT etc.)

Encoder: accumulate info of all hits/tracks by transformer

Decoder:
Input cluster info one by one
Output info of next cluster
(training) MC truth clusters
(inference) just provide <bos> to derive first cluster, using output as next input until <eos> obtained
(Inspired by translation NN)

Advertisements

LCWS2024 @ U. Tokyo

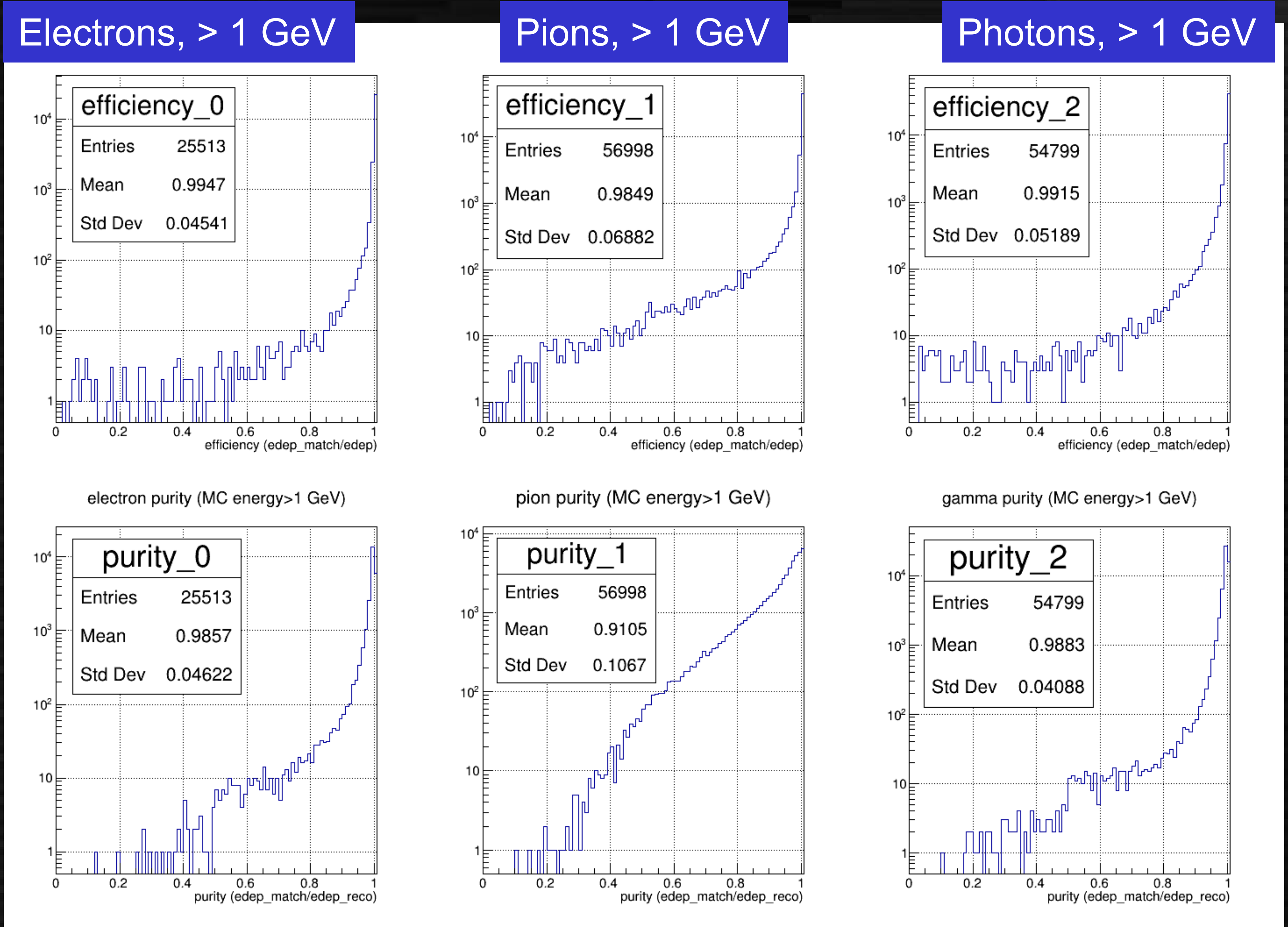
Registration deadline (early):
31st May 2024
Registration deadline (final):
30th June 2024
Workshop days:
8-11th July 2024

LCWS 2024 indico: https://agenda.linearcollider.org/e/lcws2024

Summary

- DNN-based PFA is important for further development
  - For improving performance
  - For detector design/optimization (eg. Timing)
- First implementation of track-cluster matching on GravNet/object condensation done/tested
  - Comparable performance to PandoraPFA (under investigation)
  - Still initial stage of optimization – having much hope!
  - Another methodology (transformer) being tried as well
- (additional) AI/ML should also be good for design/produce/test calorimeters, but need innovative ideas

Efficiency & purity with Pandora, ntau events

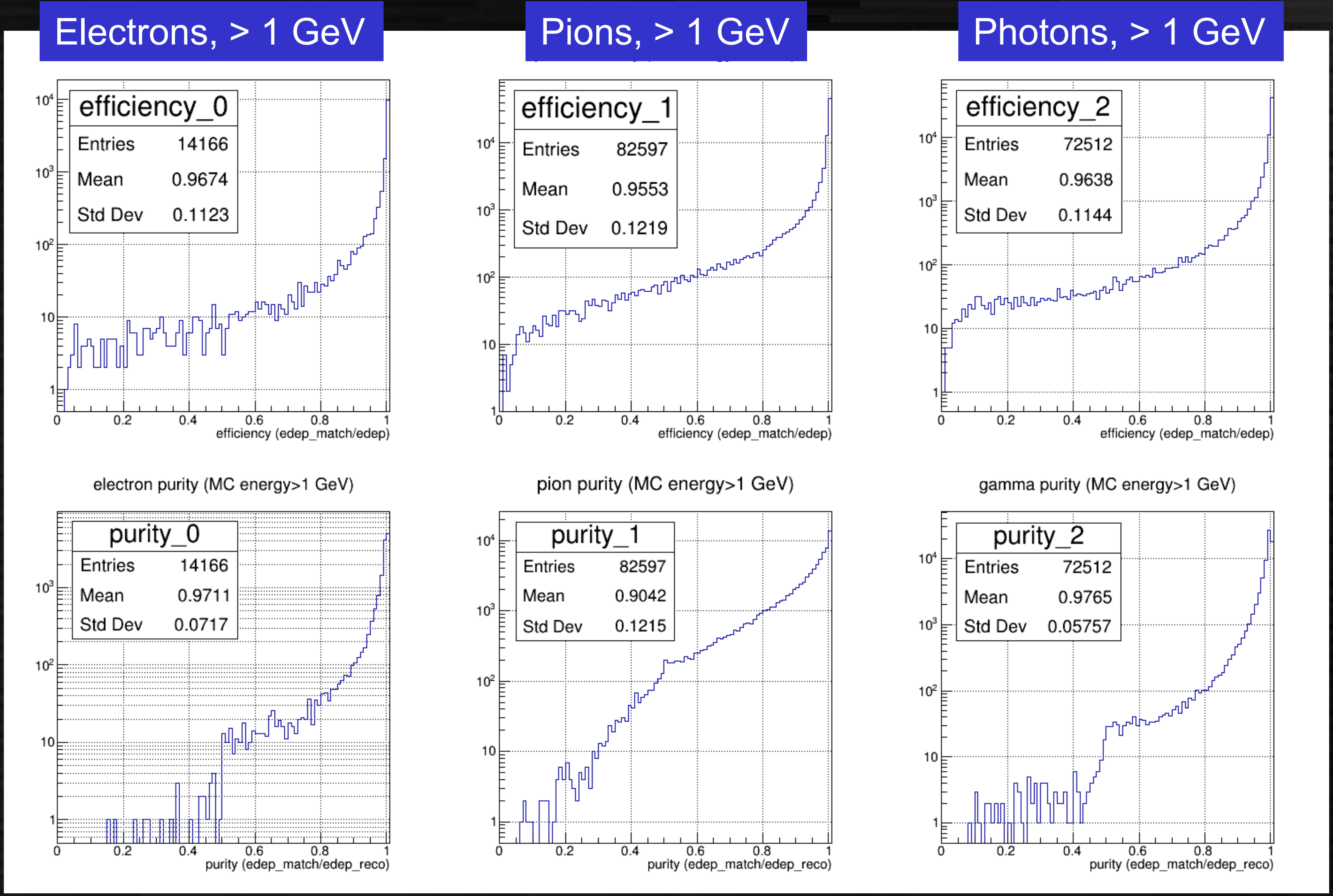


Slightly different algorithm for calculations of efficiency/purity
(to be investigated: efficiency can be overestimated)

Pandora seems still better

ILCSoft matching difference to be investigated

Efficiency & purity with Pandora, qq events



Slightly different algorithm for calculations of efficiency/purity
(to be investigated: efficiency can be overestimated

Pandora seems still better

ILCSoft matching difference to be investigated