End-to-end ML-based Reconstruction Chain for the DUNE Near Detector

François Drielsma

on behalf of the SLAC ML group

drielsma@slac.stanford.edu

May 13, 2020

U.S. DEPARTMENT OF ENERGY
Office of Science

SLAC NATIONAL ACCELERATOR LABORATORY

ML for LArTPCs

SLAC Group

SLAC NATIONAL ACCELERATOR LABORATORY

At SLAC, research supported by DoE ML grants (**K. Terao**):

- **Deep-learning-based data reconstruction chain for liquid argon time-projection chambers**
- μBOONE, pDUNE, ICARUS, ArgonCube 2x2, DUNE

Group consists of three **scientists**, three **postdocs**, three **grad students**

T. Usher
ICARUS

F. Drielsma
ICARUS

Q. Lin
ICARUS

L. Domine
ICARUS

P. Tsang
pDUNE

R. Itay
μBOONE

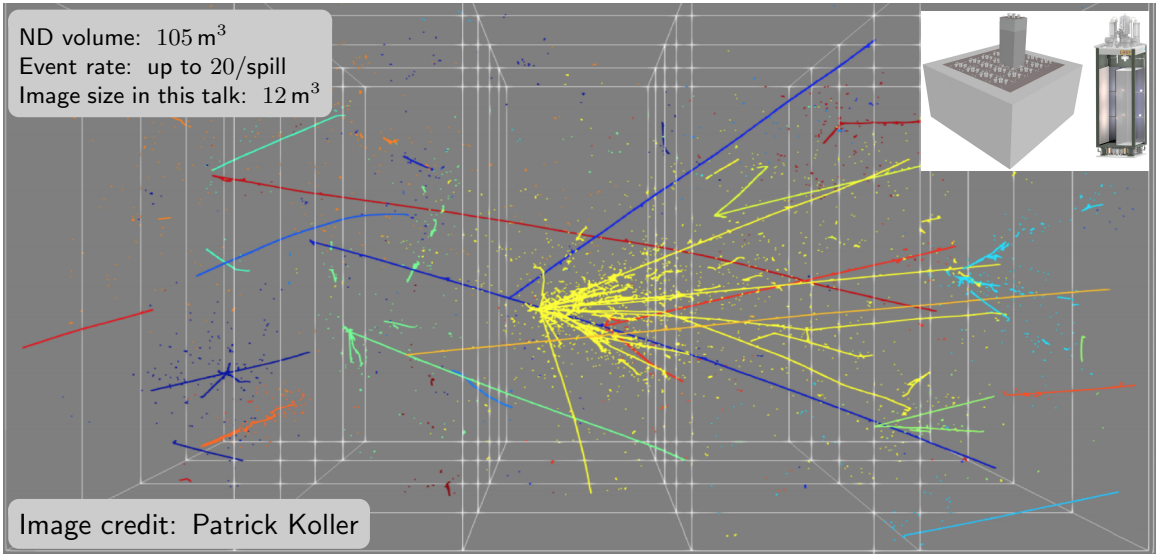
D.H. Koh
ICARUS

P. Cotes de Soux

High rate LArTPCs

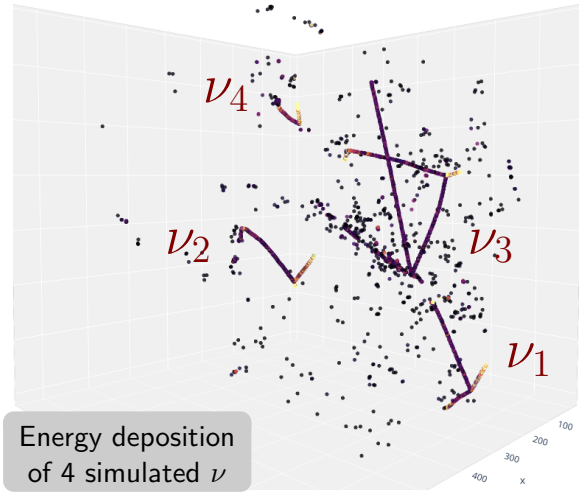
DUNE ND

SLAC NATIONAL ACCELERATOR LABORATORY



High rate LArTPCs | NuMI Example

SLAC NATIONAL ACCELERATOR LABORATORY



Energy deposition of 4 simulated ν

ν_1 0.70 GeV ν_μ CC interaction

- 0.17 GeV μ^-, 1.09 GeV p^+, 0.37 GeV π^+

ν_2 0.81 GeV ν_μ CC interaction

- 0.30 GeV μ^- (32 MeV e^-), 1.13 GeV p^+, 0.99 GeV p^+

ν_3 3.14 GeV ν_μ CC interaction

- Primary: 1.81 GeV μ^-, 0.88 GeV π^+, 0.22 GeV $\pi^0 \rightarrow$ (198 MeV γ + 24 MeV γ)
- Secondary: 0.31 GeV π^-, 1.03 GeV p^+
- Tertiary: 1.02 GeV p^+

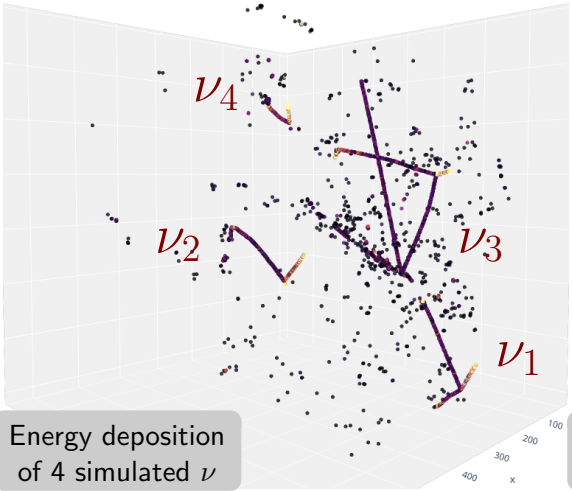
ν_4 0.35 GeV ν_μ CC interaction

- 0.18 GeV μ^- (24 MeV e^-), 1.06 GeV p^+

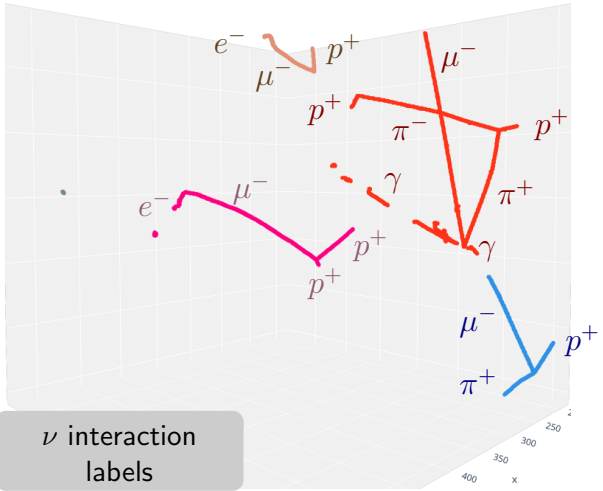
High rate LArTPCs

NuMI Example

SLAC NATIONAL ACCELERATOR LABORATORY



Energy deposition of 4 simulated ν

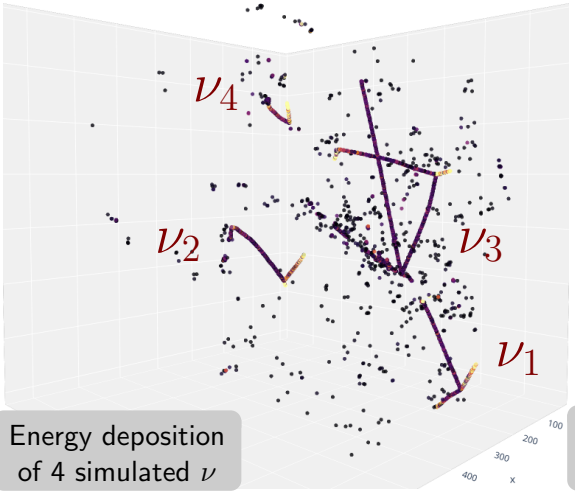


ν interaction labels

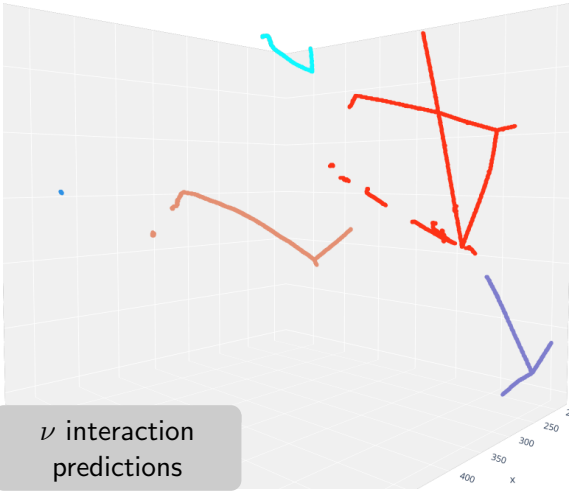
High rate LArTPCs

NuMI Example

SLAC NATIONAL ACCELERATOR LABORATORY



Energy deposition of 4 simulated ν

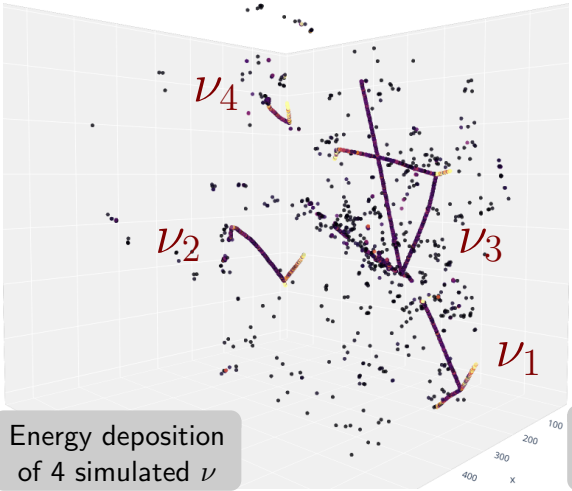


ν interaction predictions

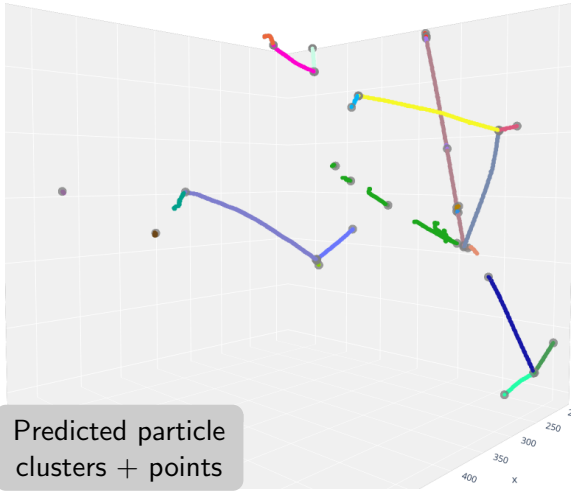
High rate LArTPCs

NuMI Example

SLAC NATIONAL ACCELERATOR LABORATORY

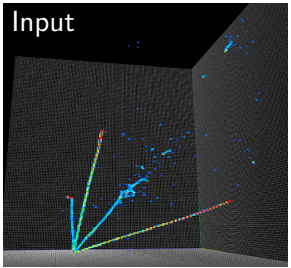


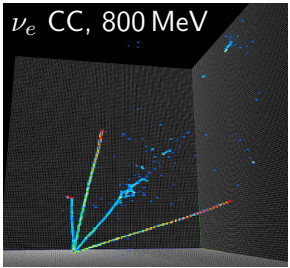
Energy deposition of 4 simulated ν

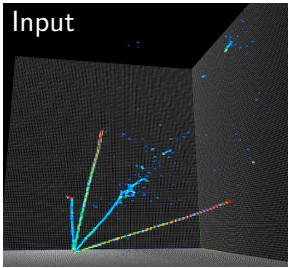


Predicted particle clusters + points

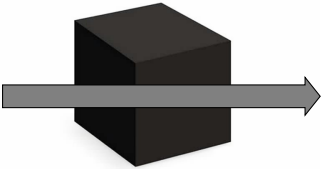
Hierarchical feature extraction | Big Picture

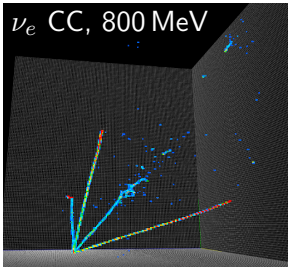






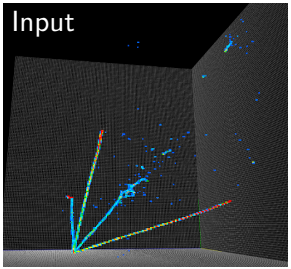
CNN ?





Hierarchical feature extraction | Big Picture

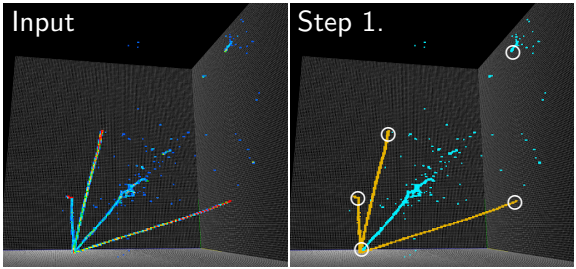
Enforce extraction of **hierarchical physics features**



Hierarchical feature extraction | Big Picture

Enforce extraction of **hierarchical physics features**

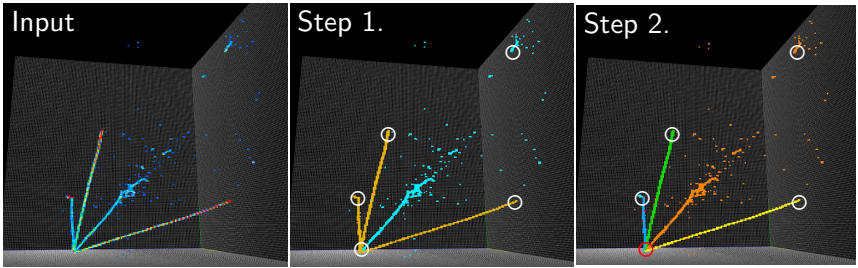
1. Pixel feature extraction + key points (particle start/end)



Hierarchical feature extraction | Big Picture

Enforce extraction of **hierarchical physics features**

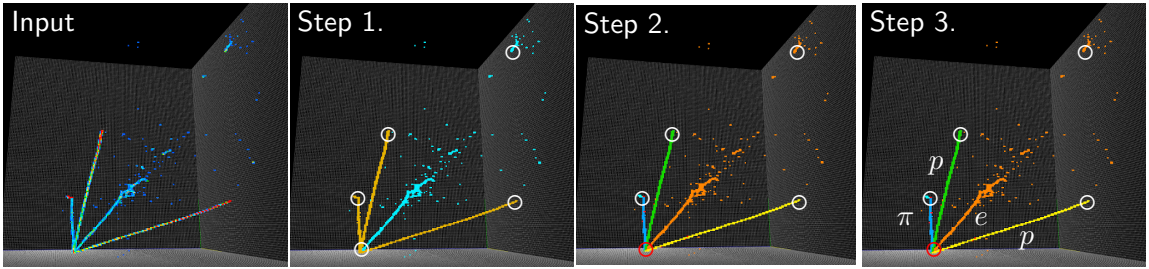
1. Pixel feature extraction + key points (particle start/end)
2. Vertex finding + particle clustering



Hierarchical feature extraction | Big Picture

Enforce extraction of **hierarchical physics features**

1. Pixel feature extraction + key points (particle start/end)
2. Vertex finding + particle clustering
3. Particle type + energy/momentum

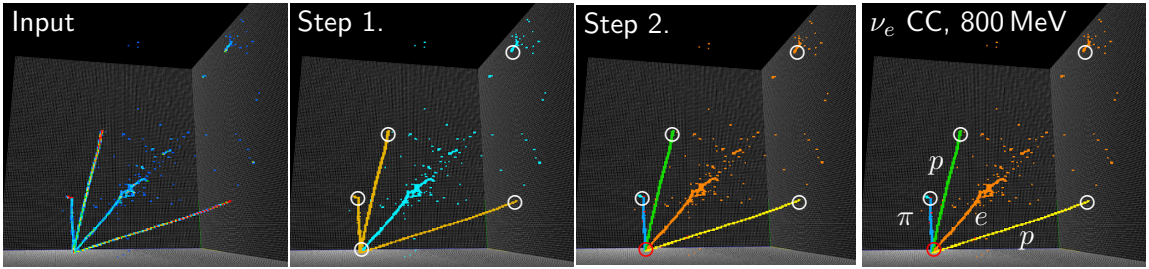


Hierarchical feature extraction | Big Picture

SLAC NATIONAL ACCELERATOR LABORATORY

Enforce extraction of **hierarchical physics features**

1. Pixel feature extraction + key points (particle start/end)
2. Vertex finding + particle clustering
3. Particle type + energy/momentum
4. Interaction (particle flow) reconstruction



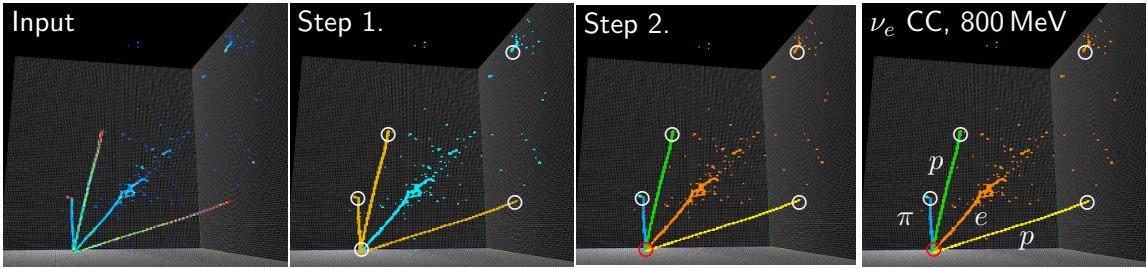
Hierarchical feature extraction | Big Picture

SLAC NATIONAL ACCELERATOR LABORATORY

Enforce extraction of **hierarchical physics features**

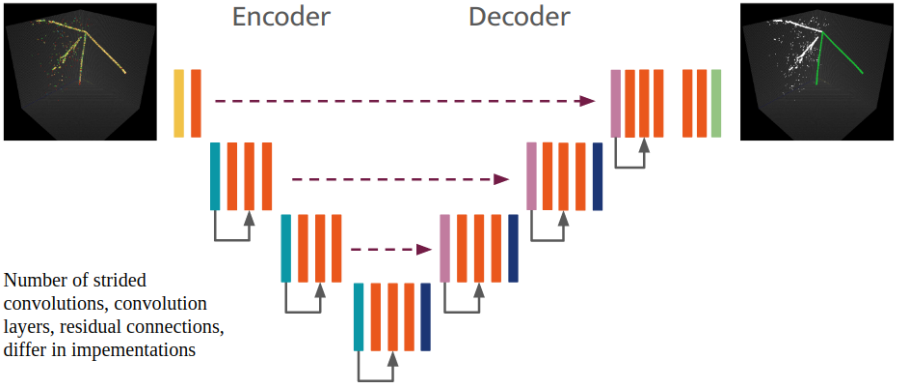
1. Pixel feature extraction + key points (particle start/end)
2. Vertex finding + particle clustering
3. Particle type + energy/momentum
4. Interaction (particle flow) reconstruction

Whole chain can be optimized together



Pixel feature extraction | Architecture

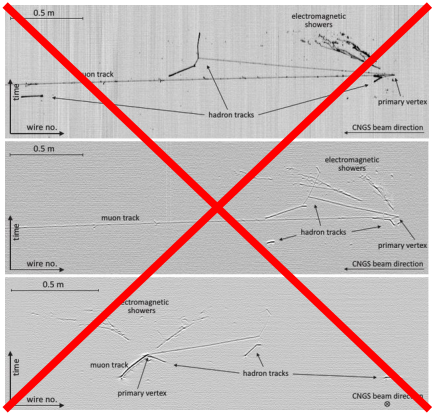
UNet + Residual connections + Sparse convolution → **Sparse UResNet**



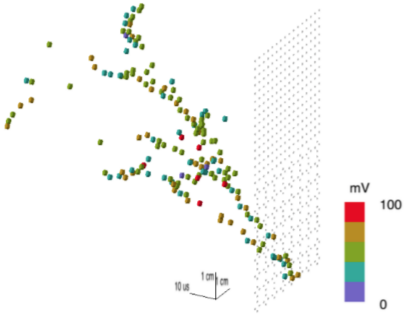
arXiv:1903.05663

L. Domine

Pixel feature extraction | Input

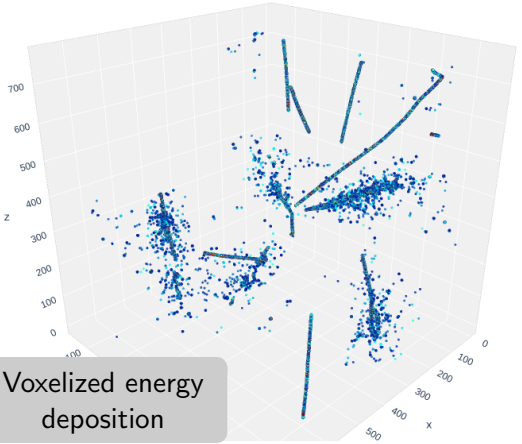


ICARUS, arXiv:1210.5089

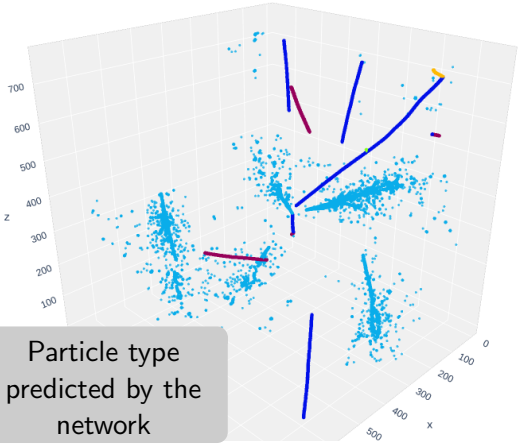


LArPix, arXiv:1808.02969

Pixel feature extraction | Particle type



Voxelized energy deposition



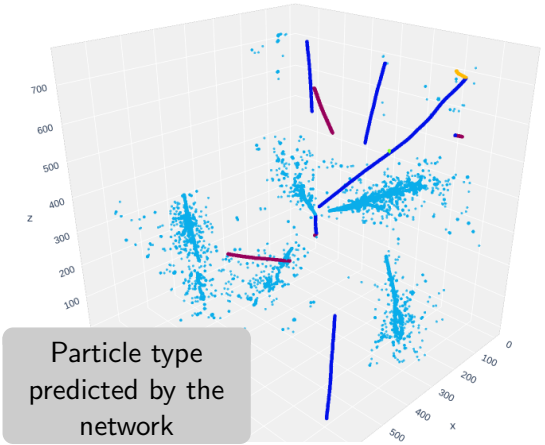
Particle type predicted by the network

Pixel feature extraction | Particle type

Particle type identification accuracy:

Particle type	Voxel fraction	Accuracy
HIP	17 %	98.2 %
MIP	34 %	99.4 %
Showers	47 %	99.2 %
Delta rays	1 %	96 %
Michel	1 %	94.7 %
Total		**99 %**

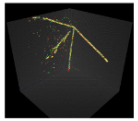
Network adapted to very sparse data,
see paper for details: arXiv:1903.05663

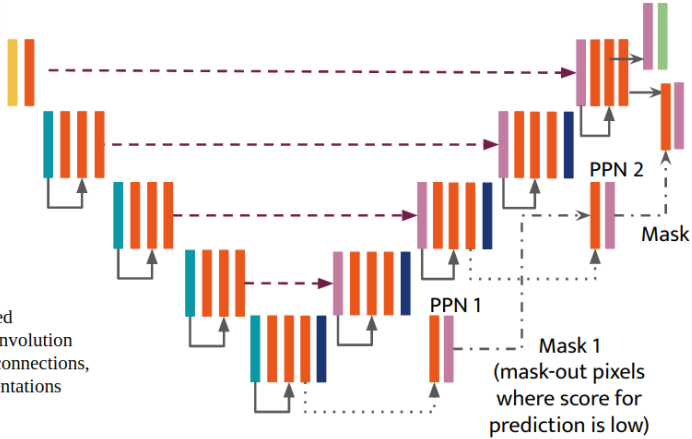


Particle type predicted by the network

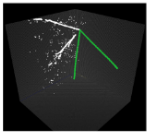
Pixel feature extraction

Point proposal





Number of strided convolutions, convolution layers, residual connections, differ in impementations

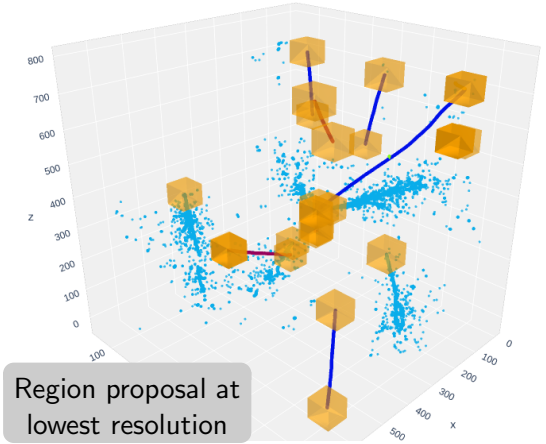


input
conv
conv-s2-finc
tconv-s2-fde
conv-fdec
softmax
Residual connections
Concatenation

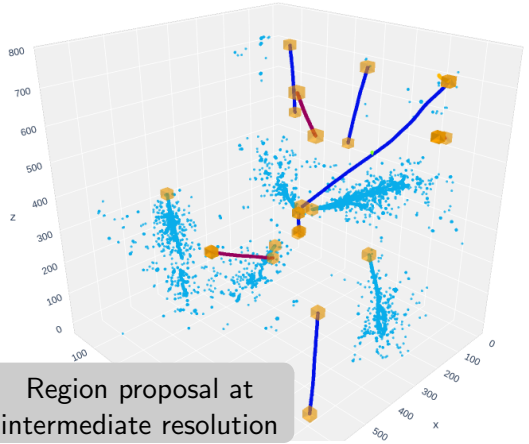
doi.org/10.5281/zenodo.1300713

L. Domine

Pixel feature extraction | Point proposal



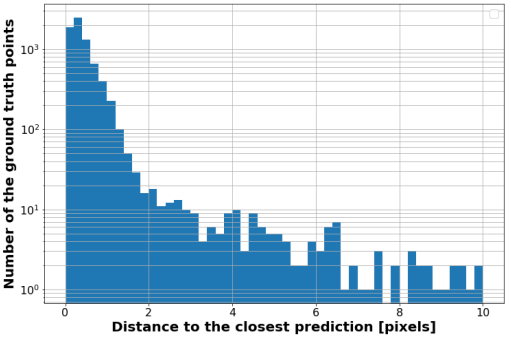
Region proposal at lowest resolution

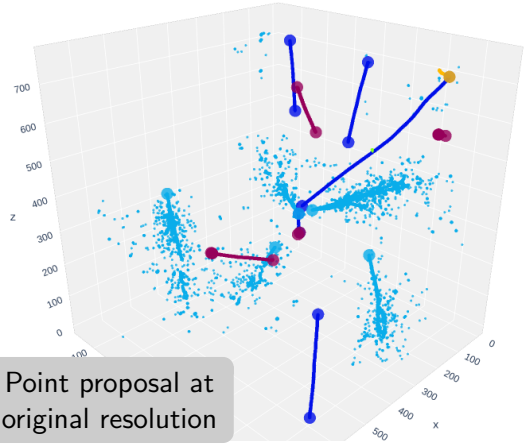


Region proposal at intermediate resolution

Pixel feature extraction | Point proposal

Point proposal efficiency (97 % $<$ 10 px):



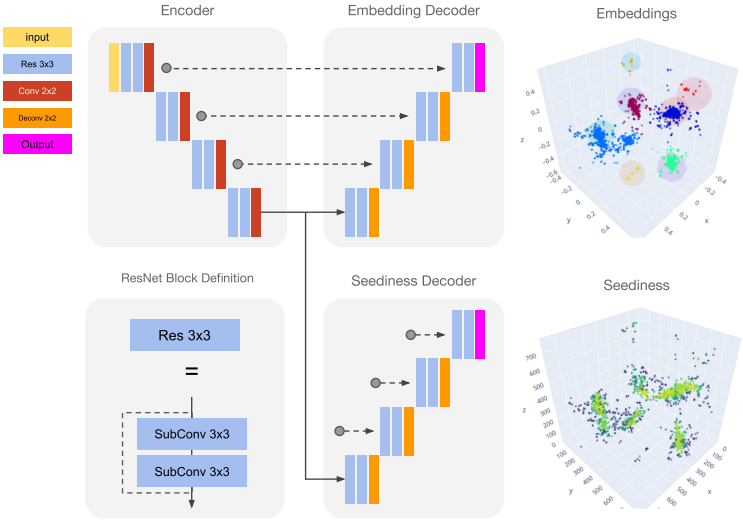


Point proposal at original resolution

doi.org/10.5281/zenodo.1300713

Pixel feature extraction

Dense clustering



Network predicts 3 things:

- **Embedding**: space in which fragments are spatially separated
- **Seediness**: likelihood that a voxel is a cluster centroid in embedding space
- **Margin**: Size of the cluster in embedding space

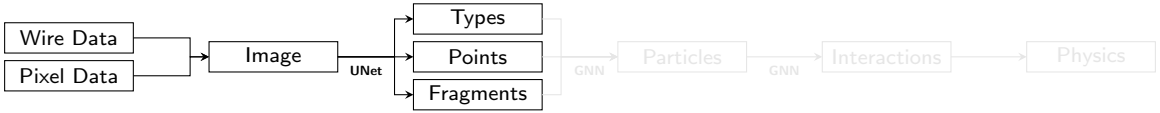
D. H. Koh

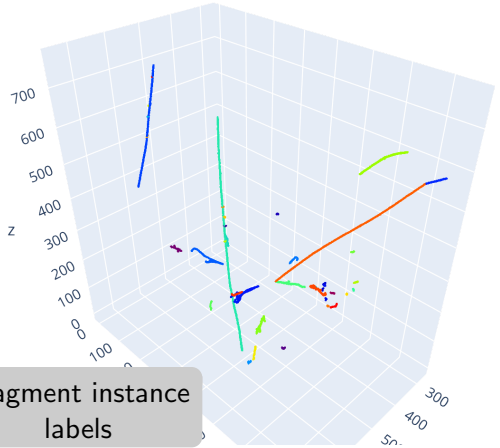
arXiv:1906.11109

Pixel feature extraction

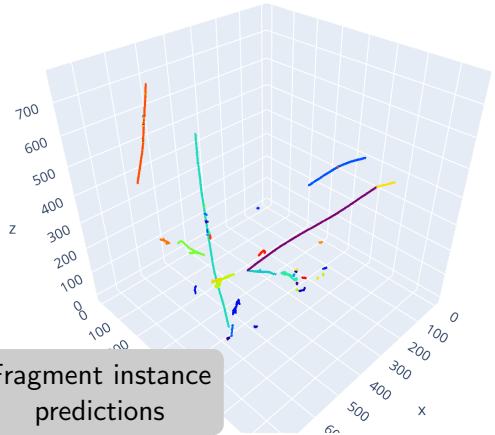
Dense clustering

SLAC NATIONAL ACCELERATOR LABORATORY



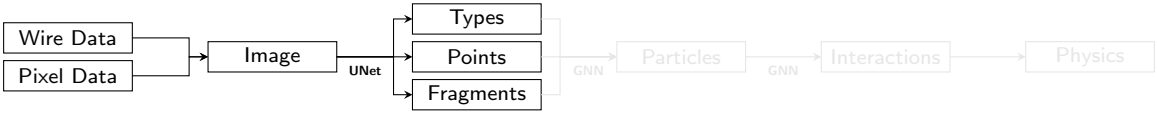


Fragment instance labels



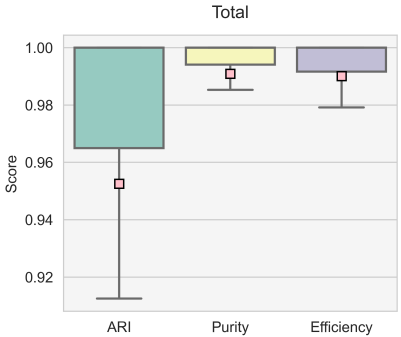
Fragment instance predictions

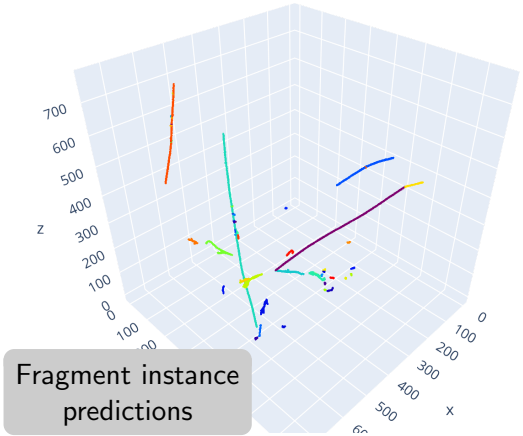
Pixel feature extraction



Dense clustering accuracy:

- Mean ARI: **95.2 %**





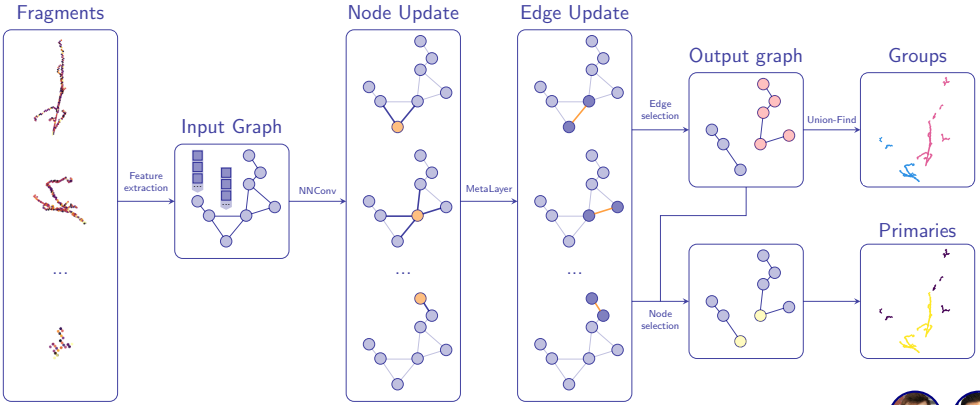
Fragment instance predictions

Shower clustering | Architecture

SLAC NATIONAL ACCELERATOR LABORATORY

Graphical Neural Networks (GNN) are ideal to cluster spatially detached objects:

- Based on *nodes* and *edges*. Features propagate by *message passing* (MP)



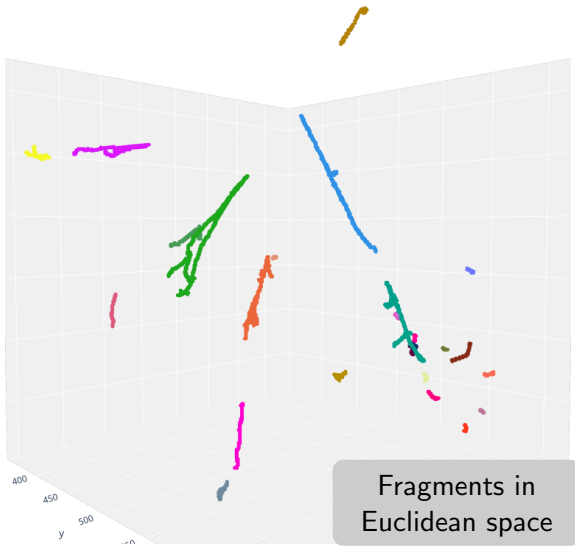
F. Drielsma, Q. Lin, P. Cotes de Soux

Shower clustering

Network input

Input:

- Fragmented EM showers



Fragments in Euclidean space

Shower clustering

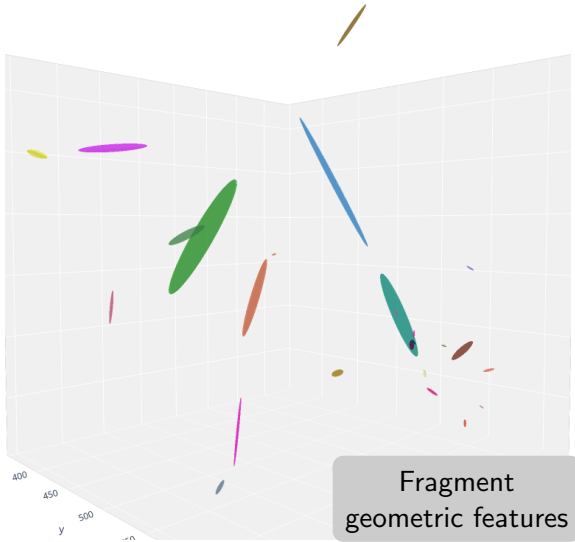
Network input

Input:

- Fragmented EM showers

Node features:

- Centroid
- Covariance matrix + PCA
- Start point + direction



Fragment geometric features

Shower clustering

Network input

SLAC NATIONAL ACCELERATOR LABORATORY

Input:

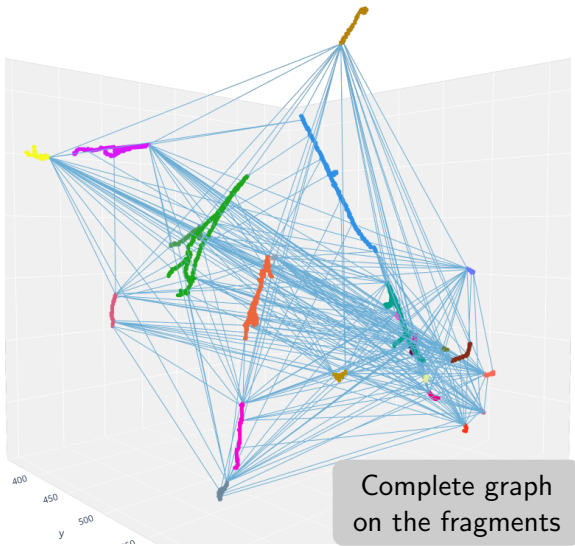
- Fragmented EM showers

Node features:

- Centroid
- Covariance matrix + PCA
- Start point + direction

Input graph:

- Connect every node with every other node (**complete graph**)



Complete graph on the fragments

Shower clustering

Network input

Input:

- Fragmented EM showers

Node features:

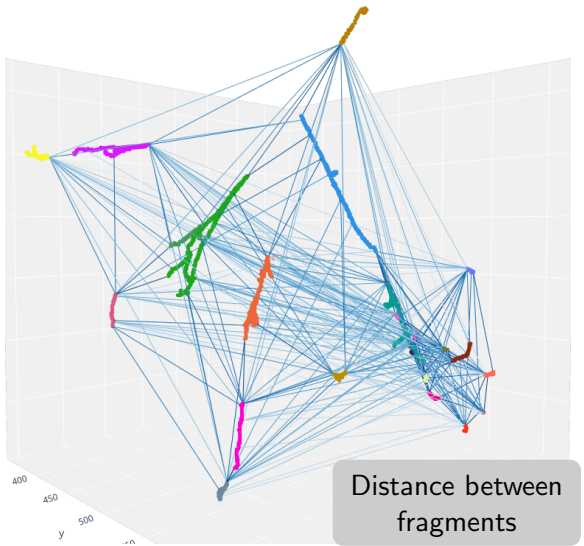
- Centroid
- Covariance matrix + PCA
- Start point + direction

Input graph:

- Connect every node with every other node (**complete graph**)

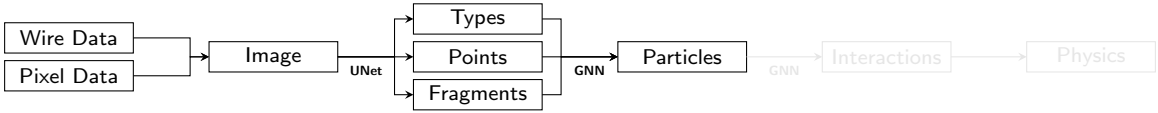
Edge features:

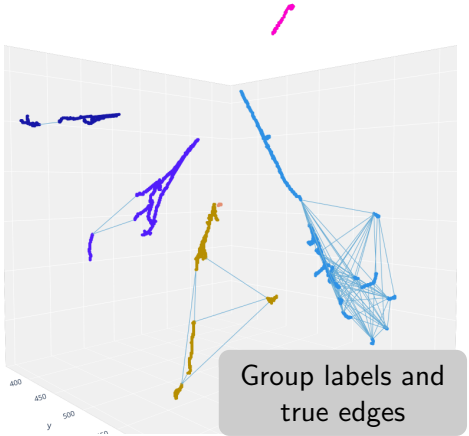
- Displacement vector (+variations)
- Closest points of approach



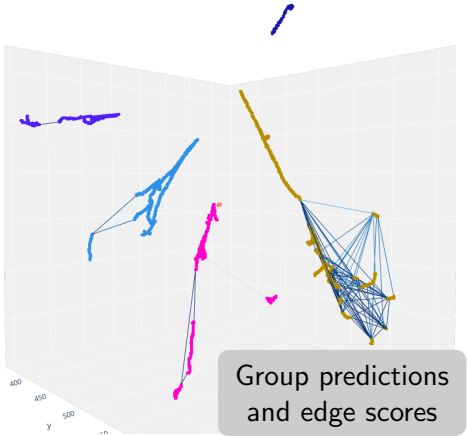
Shower clustering

Performance





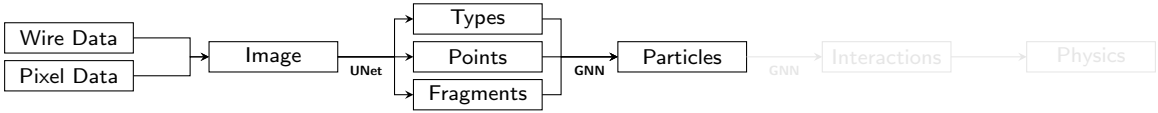
Group labels and true edges



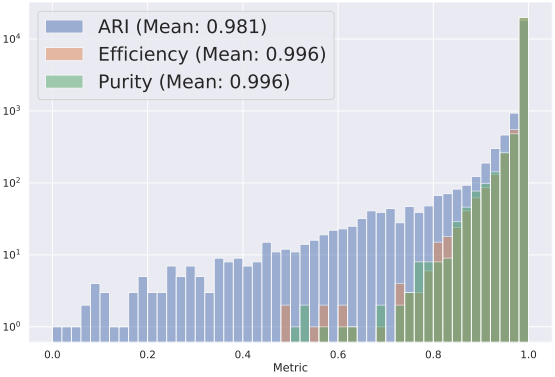
Group predictions and edge scores

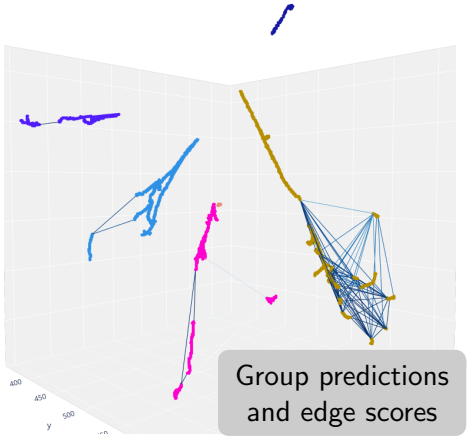
Shower clustering

Performance

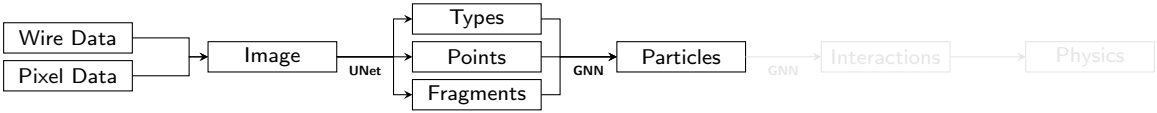


Shower clustering accuracy:





Shower Clustering | Start identification

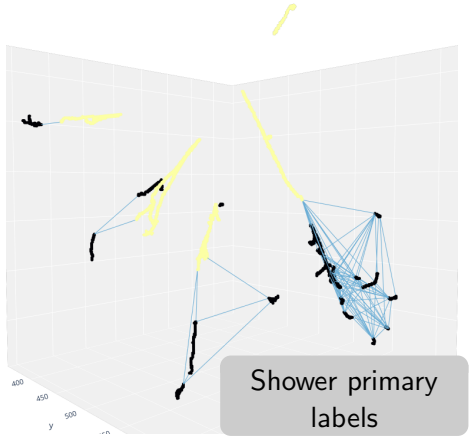


Identifying shower starts is a very useful task

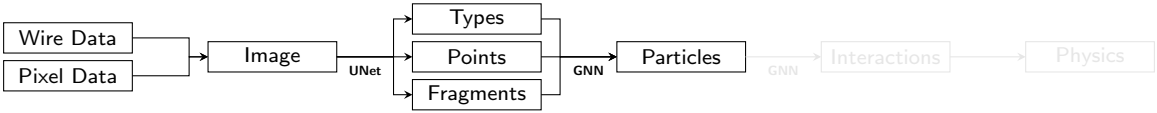
- Start point, direction
- Shower matching for π_0 reconstruction

Given the **predicted** partition of nodes:

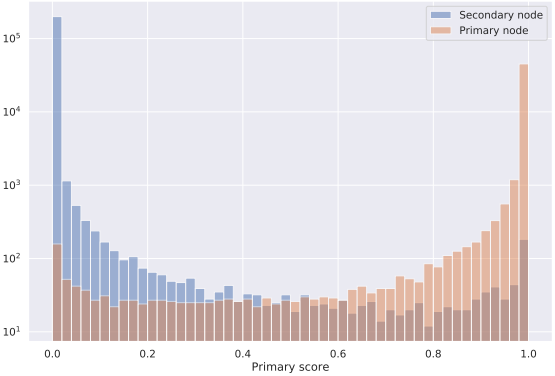
- In each group, identify the node with the highest primary score, tag as primary
- All other fragments are secondaries

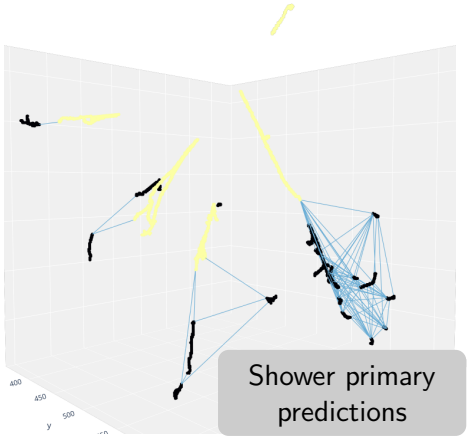


Shower Clustering | Start identification

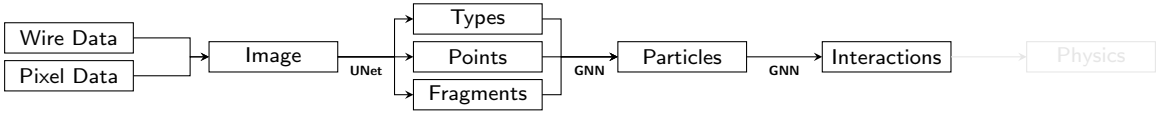


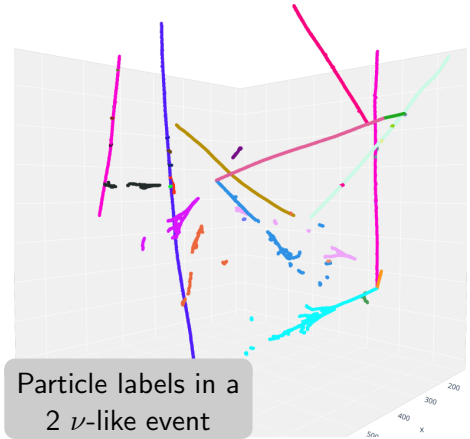
Shower primary accuracy (**99.6 %**):





Interaction Clustering | Performance





Particle labels in a 2 ν-like event

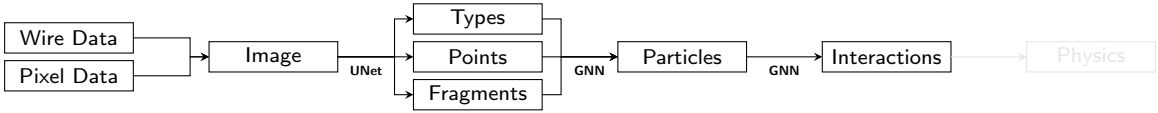
The shower clustering task can be extended to **interaction clustering**:

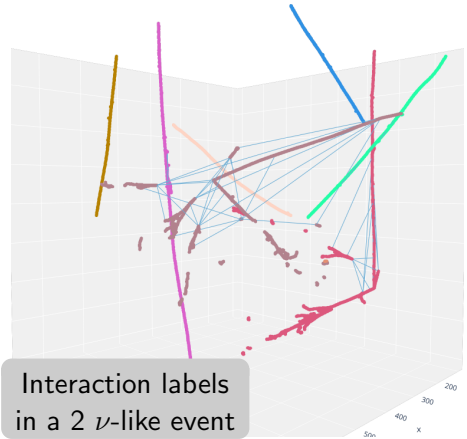
- Interaction group = particles that originate from the same vertex
- Fragment ID $\rightarrow$ Particle ID
- Particle ID $\rightarrow$ Interaction ID

Crucial to:

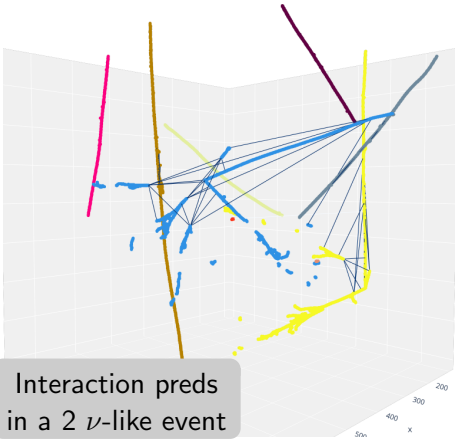
- Separate signal from background
- **Resolve pileup**

Interaction Clustering | Performance



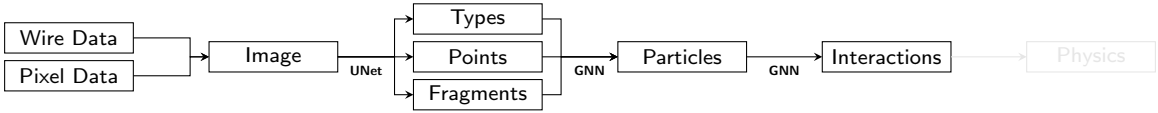


Interaction labels in a 2 ν-like event

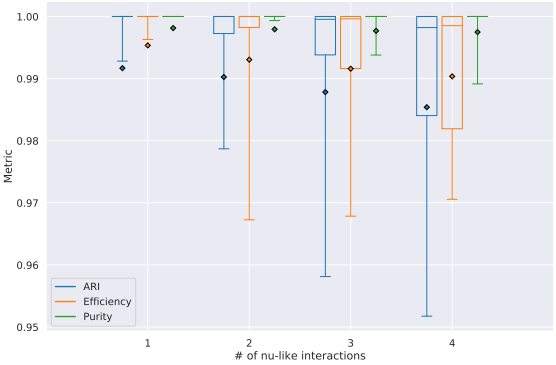


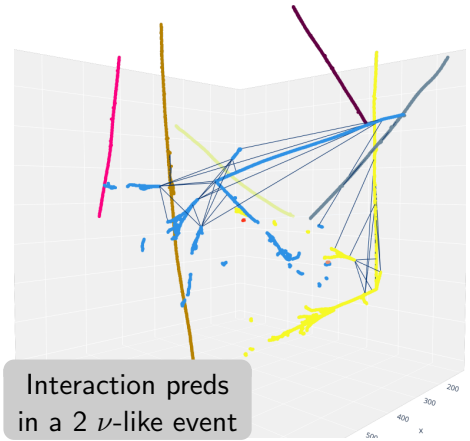
Interaction preds in a 2 ν-like event

Interaction Clustering | Performance



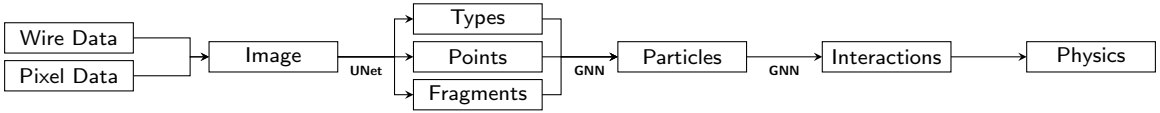
Interaction clustering performance:





Interaction preds in a 2 ν-like event

Summary



ML-based hierarchical feature extraction very successful

What is now available:

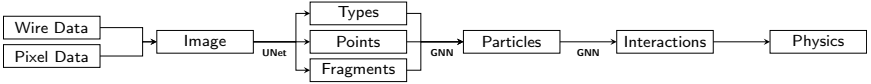
- List of interactions per event
- List of particles per interaction
- Particle class (e.g. track vs shower)
- Start and end point of particles

What we don't have:

- Momentum/energy estimation
- Particle identification
- Interaction vertex
- Particle flow

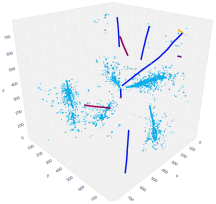
$\rightarrow$ Goals for this summer research !

Now engaging physics analysis needs in SBN/DUNE: your help is very welcome !



SLAC NATIONAL ACCELERATOR LABORATORY

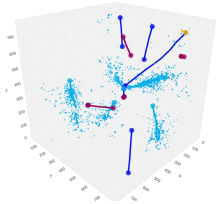
Particle types

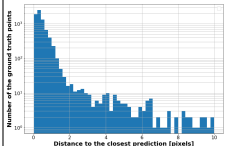


Particle type	Accuracy
HIP	98.2 %
MIP	99.4 %
Showers	99.2 %
Delta	96 %
Michel	94.7 %
Total	**99 %**

See paper for details:
arXiv:1903.05663

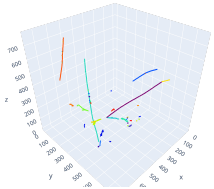
Point proposal





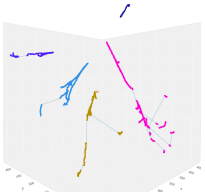
	< 10 voxels
Eff.	97.2 %
Pur.	97.6 %

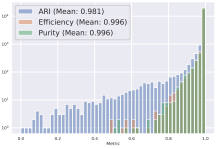
Dense clustering



	Mean
Eff.	99.0 %
Pur.	99.1 %
ARI	95.2 %
SBD	91.8 %

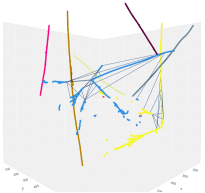
Shower clustering

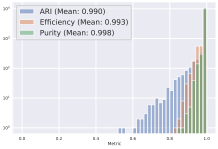




	Mean
Eff.	99.6 %
Pur.	99.6 %
ARI	98.1 %

Interaction grouping





	Mean
Eff.	99.3 %
Pur.	99.8 %
ARI	99.0 %

Here $\#\nu = 2$/image

Back-up slides

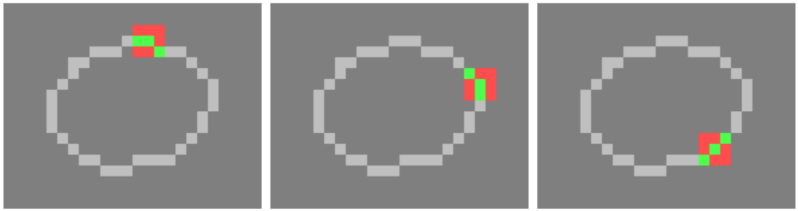
Submanifold Sparse Convolutions

1. Resources waste of dense convolutions on sparse data
2. Dilation problem
   - One nonzero site leads to 3d nonzero sites after 1 convolution
   - How to keep the same level of sparsity throughout the network?

https://arxiv.org/pdf/1711.10275.pdf

In more details, two new operations:

- Sparse convolutions (SC)
  - Discards contribution of non-active input sites
  - Output site active if at least one input site is active
- Sparse submanifold convolutions (SSC)
  - Output size = Input size
  - Output site active iff center of receptive field active
  - Only compute features for active output sites



https://arxiv.org/pdf/1711.10275.pdf

PPN outputs **voxel location**, **position within voxel** and **point class**

Three components to the point proposal loss:

- Pixel classification loss at each of three depth (pixel contains point or not)

$$\mathcal{L}_{\text{class},i} = \frac{1}{N_i} \sum_{k=1}^{N_i} y_k \log(p_k) + (1 - y_k) \log(1 - p_k)$$

- L^1 distance from true point at highest resolution **on active voxels**

$$\mathcal{L}_{\text{dist}} = \frac{1}{N_3^*} \sum_{k=1}^{N_3^*} \min_j ||\vec{p}_i - \vec{q}_j||$$

- Particle type loss at highest resolution **on active voxels**

$$\mathcal{L} = \frac{1}{N_3^*} \sum_{k=1}^{N_3^*} \sum_c y_{k,c} \log(p_{k,c})$$

Fragment clustering | Loss

Instance Segmentation:

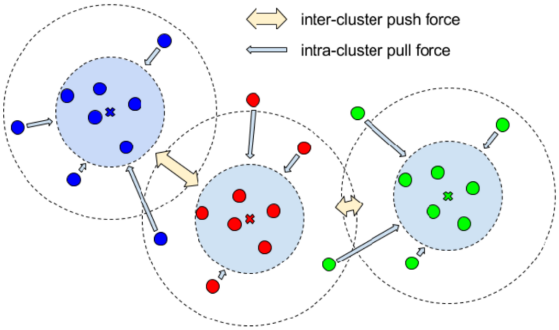
- Three component loss: pull together points that belong to the same cluster, keep distance between clusters, and regularization

$$L = \alpha L_{var} + \beta L_{dist} + \gamma L_{reg},$$

$$L_{var} = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{N_c}\sum_{i=1}^{N_c}[\max(0, \|\mu_c - x_i\| - \delta_v)]^2$$

$$L_{dist} = \frac{1}{C(C-1)}\sum_{\substack{c_A,c_B=1 \\ c_A \neq c_B}}^{C}[\max(0, 2\delta_d - \|\mu_{c_A} - \mu_{c_B}\|)]^2$$

$$L_{reg} = \frac{1}{C}\sum_{c=1}^{C}\|\mu_c\|$$

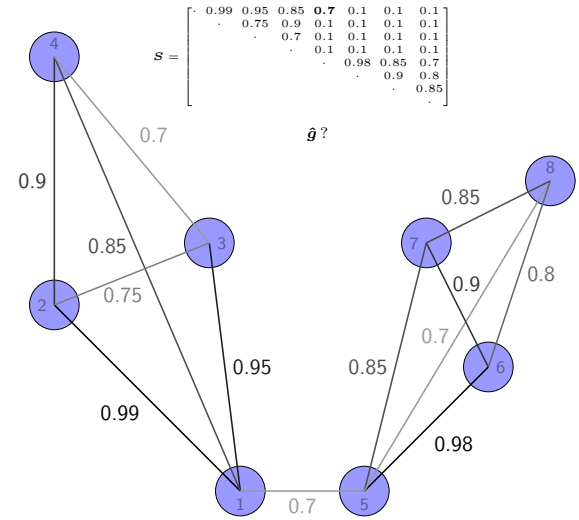


arXiv:1708.02551

Shower clustering

The network predicts a score matrix $\boldsymbol{S}$, estimate of the true adjacency matrix $\boldsymbol{A}$

- How to recover a set partition $\hat{\boldsymbol{g}}$?

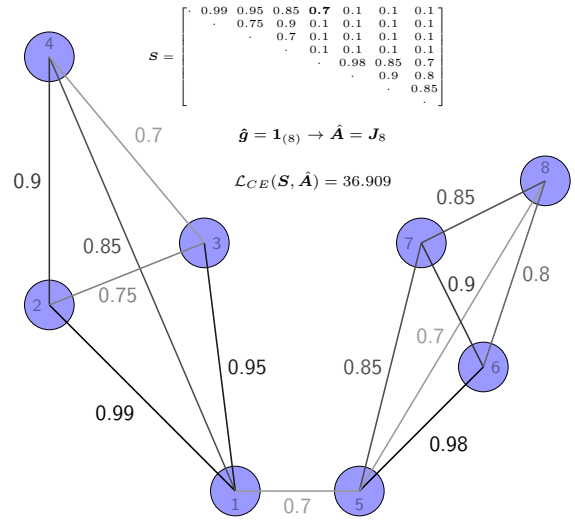


The network predicts a score matrix $\boldsymbol{S}$, estimate of the true adjacency matrix $\boldsymbol{A}$

- How to recover a set partition $\hat{\boldsymbol{g}}$?

We want the partition $\hat{\boldsymbol{g}}$ that minimizes the CE loss, given $\hat{A}_{ij} = \delta_{\hat{g}_i, \hat{g}_j}$, e.g.

$$\hat{\boldsymbol{g}} = \min_{\boldsymbol{g} \in G} \mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}(\boldsymbol{g}))$$



Shower clustering

Inference

SLAC NATIONAL ACCELERATOR LABORATORY

The network predicts a score matrix $\boldsymbol{S}$, estimate of the true adjacency matrix $\boldsymbol{A}$

- How to recover a set partition $\hat{\boldsymbol{g}}$?

We want the partition $\hat{\boldsymbol{g}}$ that minimizes the CE loss, given $\hat{A}_{ij} = \delta_{\hat{g}_i, \hat{g}_j}$, e.g.

$$\hat{\boldsymbol{g}} = \min_{\boldsymbol{g} \in G} \mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}(\boldsymbol{g}))$$

G is the set of all possible partitions

- Bell number, huge ($B_{20} \simeq 5 \times 10^{13}$), cannot brute force, how to optimize?
- Start with an empty graph

$$\boldsymbol{S} = \begin{bmatrix} \cdot & 0.99 & 0.95 & 0.85 & \mathbf{0.7} & 0.1 & 0.1 & 0.1 \\ & \cdot & 0.75 & 0.9 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & \cdot & 0.7 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & & \cdot & 0.1 & 0.1 & 0.1 & 0.1 \\ & & & & \cdot & 0.98 & 0.85 & 0.7 \\ & & & & & \cdot & 0.9 & 0.8 \\ & & & & & & \cdot & 0.85 \\ & & & & & & & \cdot \end{bmatrix}$$

$$\hat{\boldsymbol{g}} = (1, 2, 3, 4, 5, 6, 7, 8) \rightarrow \hat{\boldsymbol{A}} = \boldsymbol{I}_8$$

$$\mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}) = 29.998$$

The network predicts a score matrix $\boldsymbol{S}$, estimate of the true adjacency matrix $\boldsymbol{A}$

- How to recover a set partition $\hat{\boldsymbol{g}}$?

We want the partition $\hat{\boldsymbol{g}}$ that minimizes the CE loss, given $\hat{A}_{ij} = \delta_{\hat{g}_i, \hat{g}_j}$, e.g.

$$\hat{\boldsymbol{g}} = \min_{\boldsymbol{g} \in G} \mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}(\boldsymbol{g}))$$

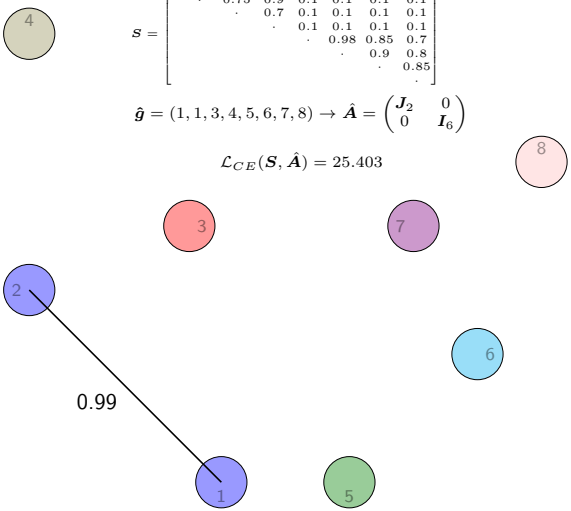
G is the set of all possible partitions

- Bell number, huge ($B_{20} \simeq 5 \times 10^{13}$), cannot brute force, how to optimize?
- Start with an empty graph
- Iteratively attempt to add most likely edges until edge score < 0.5

$$\boldsymbol{S} = \begin{bmatrix} \cdot & 0.99 & 0.95 & 0.85 & \mathbf{0.7} & 0.1 & 0.1 & 0.1 \\ & \cdot & 0.75 & 0.9 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & \cdot & 0.7 & 0.1 & 0.1 & 0.1 & 0.1 \\ & & & \cdot & 0.1 & 0.1 & 0.1 & 0.1 \\ & & & & \cdot & 0.98 & 0.85 & 0.7 \\ & & & & & \cdot & 0.9 & 0.8 \\ & & & & & & \cdot & 0.85 \\ & & & & & & & \cdot \end{bmatrix}$$

$$\hat{\boldsymbol{g}} = (1, 1, 3, 4, 5, 6, 7, 8) \rightarrow \hat{\boldsymbol{A}} = \begin{pmatrix} \boldsymbol{J}_2 & 0 \\ 0 & \boldsymbol{I}_6 \end{pmatrix}$$

$$\mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}) = 25.403$$



Shower clustering

Inference

The network predicts a score matrix $\boldsymbol{S}$, estimate of the true adjacency matrix $\boldsymbol{A}$

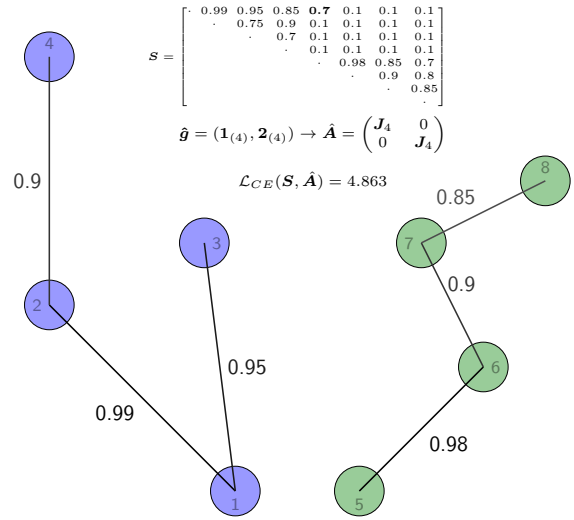
- How to recover a set partition $\hat{\boldsymbol{g}}$?

We want the partition $\hat{\boldsymbol{g}}$ that minimizes the CE loss, given $\hat{A}_{ij} = \delta_{\hat{g}_i, \hat{g}_j}$, e.g.

$$\hat{\boldsymbol{g}} = \min_{\boldsymbol{g} \in G} \mathcal{L}_{CE}(\boldsymbol{S}, \hat{\boldsymbol{A}}(\boldsymbol{g}))$$

G is the set of all possible partitions

- Bell number, huge ($B_{20} \simeq 5 \times 10^{13}$), cannot brute force, how to optimize?
- Start with an empty graph
- Iteratively attempt to add most likely edges until edge score < 0.5



Shower clustering

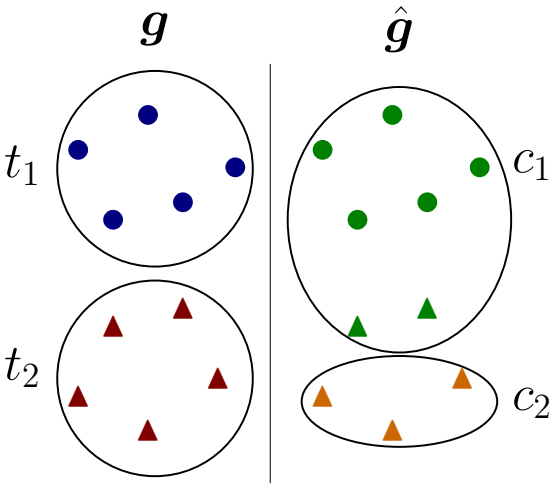
Metrics

Purity $= \frac{1}{N}\sum_{i=1}^{n_p} \max_j |c_i \cap t_j|$

- c_i predicted cluster
- t_j true cluster with highest count in c_i

Efficiency $= \frac{1}{N}\sum_{i=0}^{n_t} \max_j |c_j \cap t_i|$

- c_j pred. cluster with highest count in t_i
- t_i true cluster



Purity $= \frac{1}{N}\sum_{i=1}^{n_p} \max_j |c_i \cap t_j|$

- c_i predicted cluster
- t_j true cluster with highest count in c_i

Efficiency $= \frac{1}{N}\sum_{i=0}^{n_t} \max_j |c_j \cap t_i|$

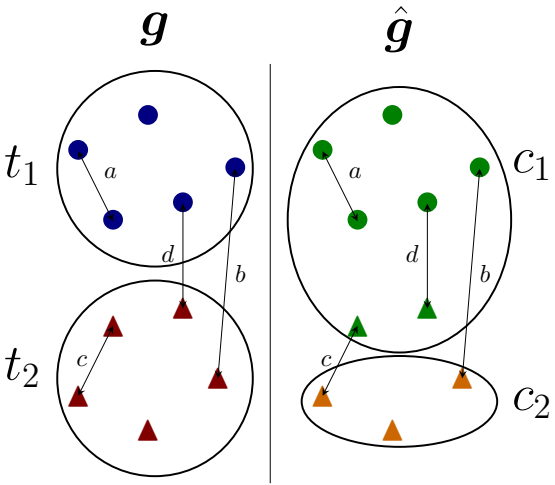
- c_j pred. cluster with highest count in t_i
- t_i true cluster

Adjusted Rand Index (ARI)

- Measure of overlap of prediction and truth, adjusted for random chance

$\text{RI} = \frac{a+b}{a+b+c+d}$

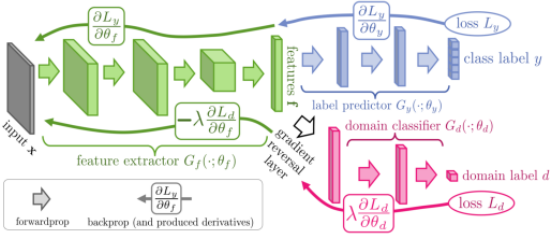
$\text{ARI} = \frac{RI - E(RI)}{1 - E(RI)}$



Domain discrepancies | GAN

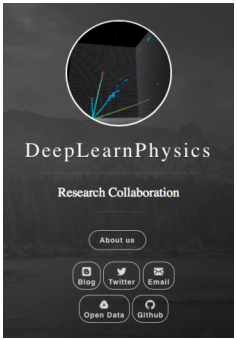
What can we do about imperfect simulation ?

- Issue: the signal distribution learned by the algorithm may be different in two domains!
- Mitigation techniques in ML domain ?
  - Can try CNN to locate where it is
  - Can try CNN to fix the discrepancy
  - Can try a training technique to minimize the effect



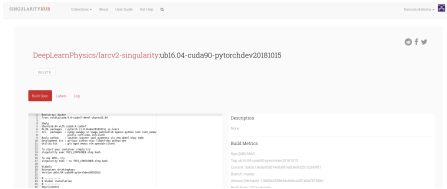
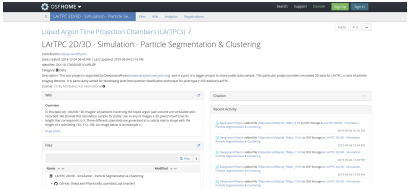
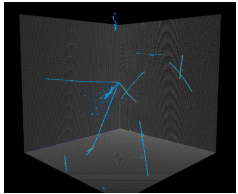
Maximize the loss to discriminate data vs. simulation, feature extractors are penalized to key on simulation specific information

Domain-Adversarial Training of Neural Networks: J. Mach. Learn. Res. 17 (2016)



DeepLearnPhysics: Collaboration for ML technique R&D

- Open simulation sample (used throughout this talk)
  - Open real data ? Soon ! (3D prototype R&D at SLAC)
- Open source container (Singularity)
- Open source code (GitHub)
  - All the code used to make this talk is available

→ **Reproducible results** !

- Readers have reproduced arXiv:1903.05663