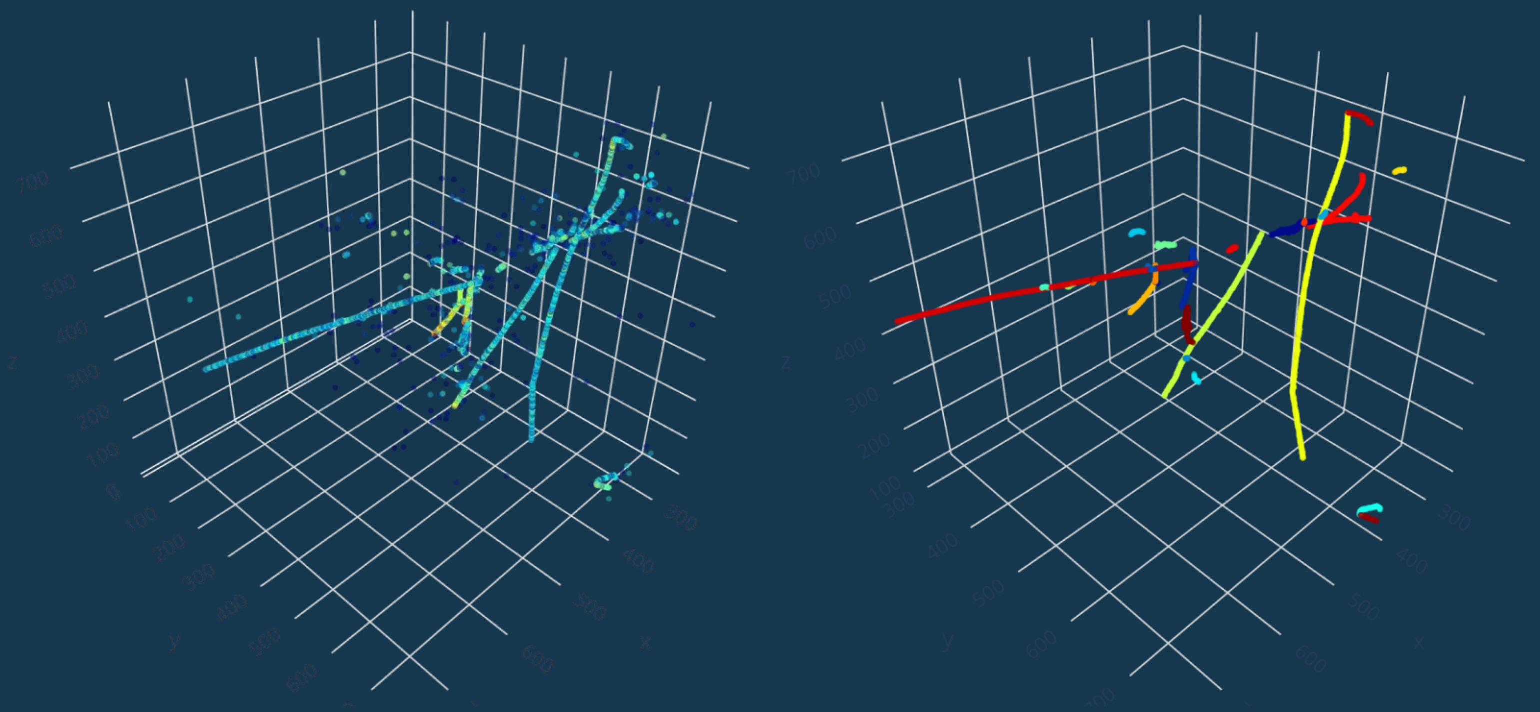


Proposal-free Deep Sparse Convolutional Neural Network for 3D Pixel Clustering

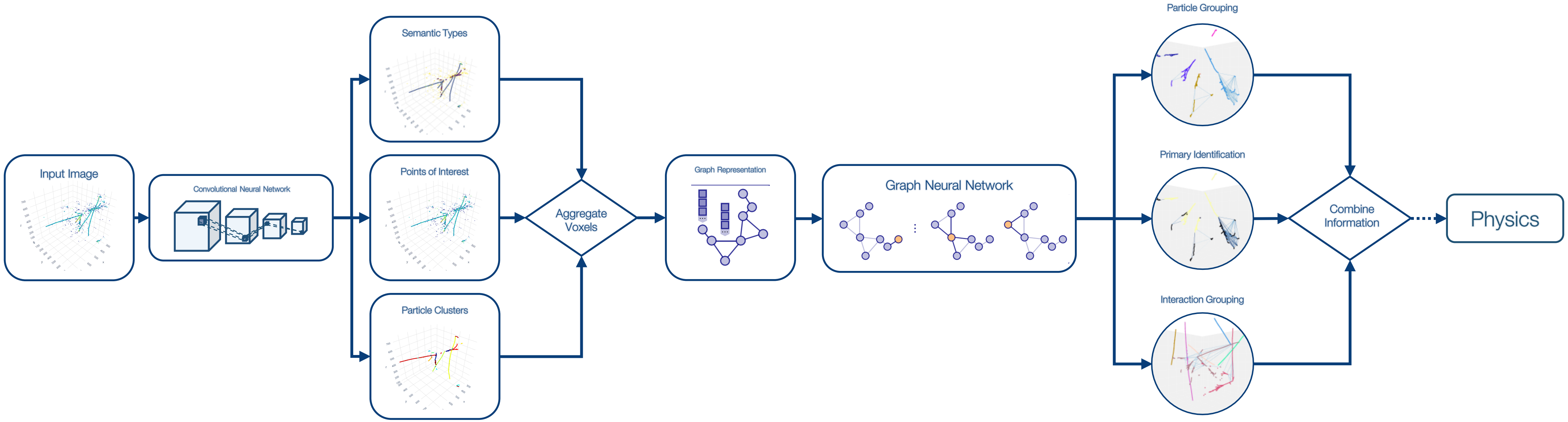
P. Cote de Soux, L. Domine, F. Drielsma, R. Itay, **D. Koh***, Q. Lin, B. Nelson, P. Tsang, K.Terao, T. Usher

U.S. DEPARTMENT OF ENERGY | Office of Science

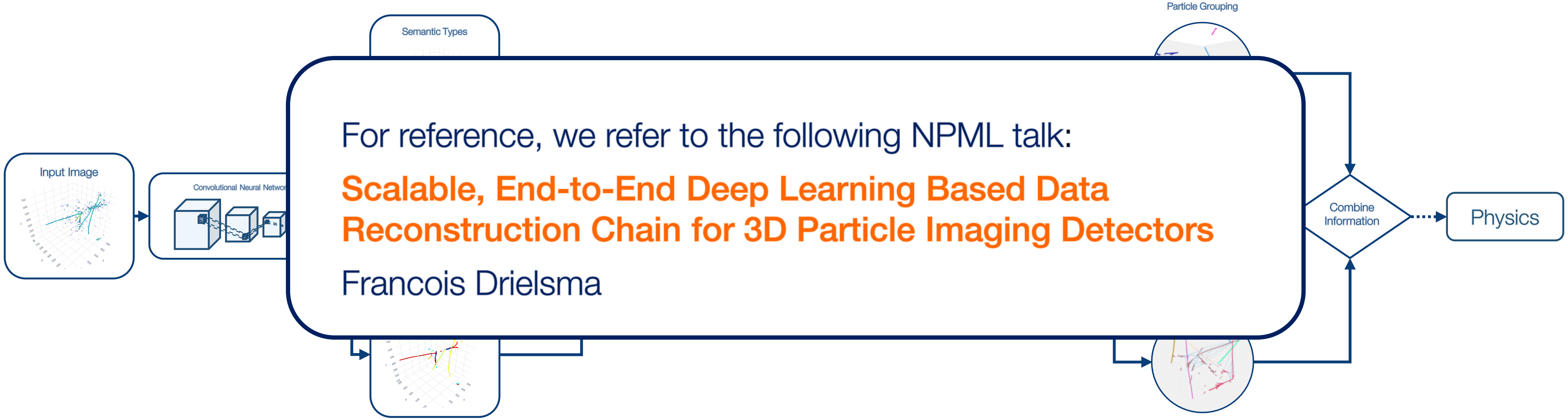
Stanford University

SLAC NATIONAL ACCELERATOR LABORATORY

In SLAC, we aim to develop a full ML-based, general purpose reconstruction chain for LArTPC data.



In SLAC, we aim to develop a full ML-based, general purpose reconstruction chain for LArTPC data.



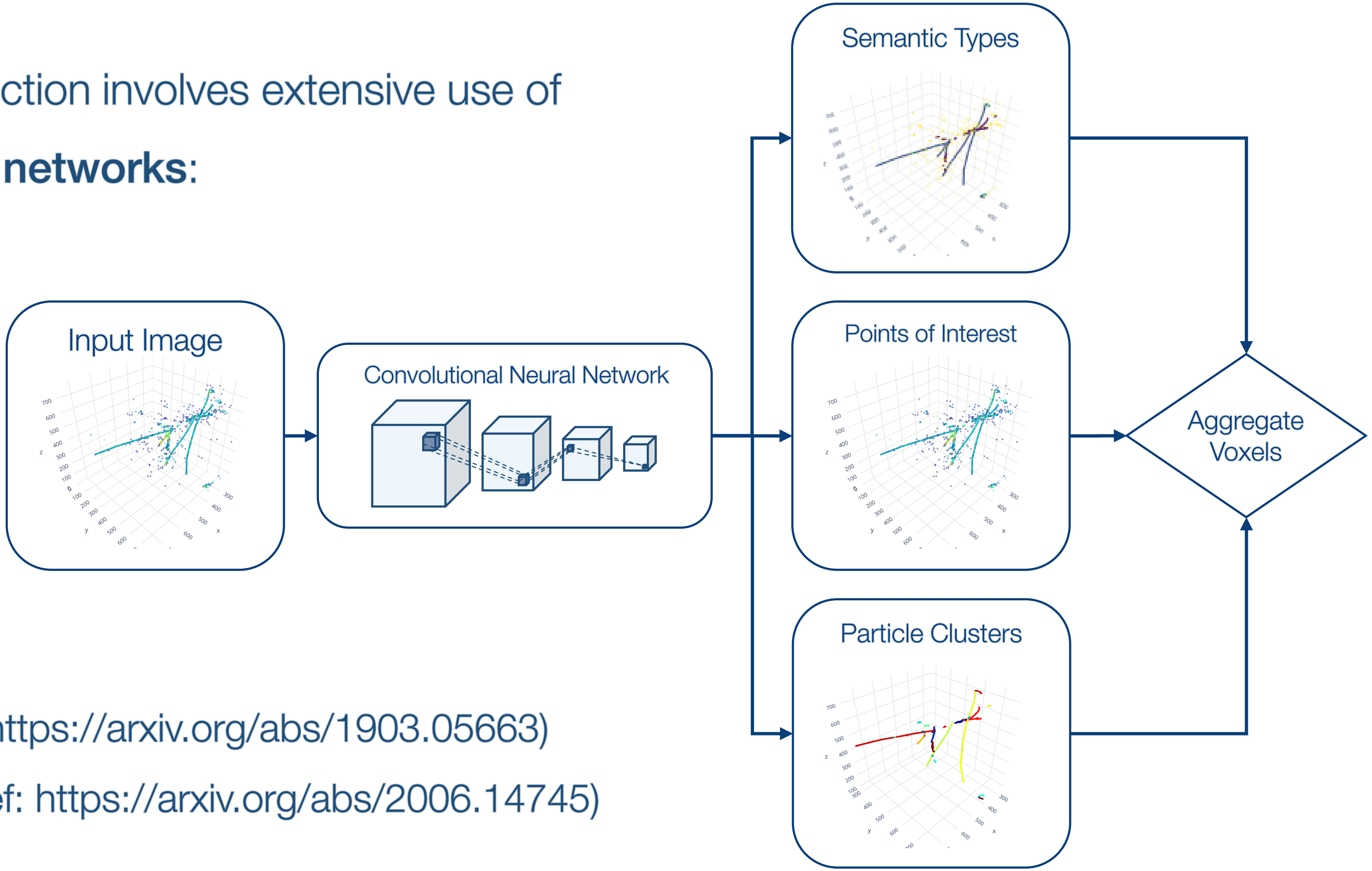
For reference, we refer to the following NPML talk:

Scalable, End-to-End Deep Learning Based Data Reconstruction Chain for 3D Particle Imaging Detectors

Francois Drielsma

SLAC NATIONAL ACCELERATOR LABORATORY

A. The first phase of reconstruction involves extensive use of **sparse convolutional neural networks**:

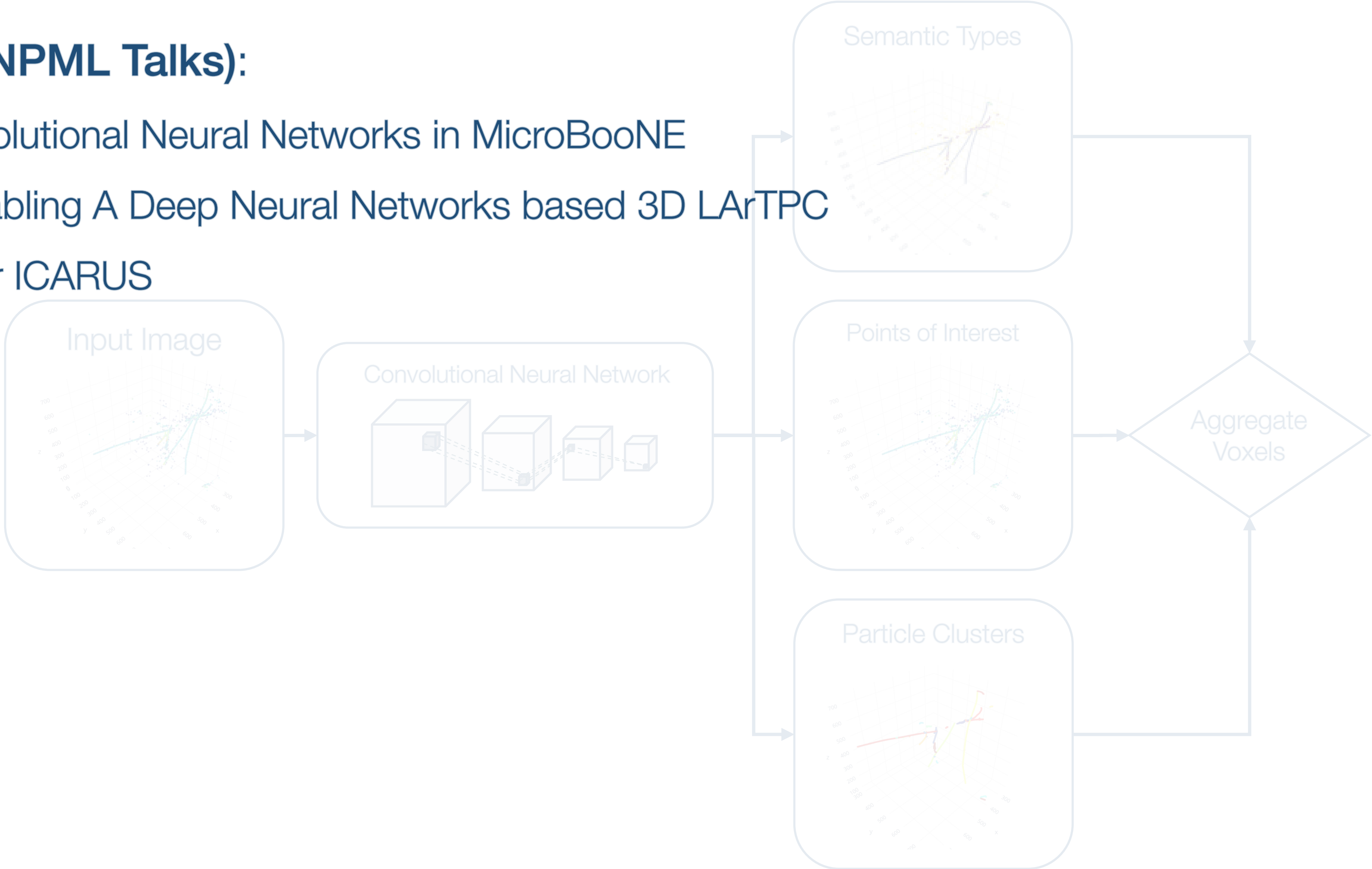


Relevant Work:

- Semantic Segmentation (ref: https://arxiv.org/abs/1903.05663)
- Points of Interest Detection (ref: https://arxiv.org/abs/2006.14745)

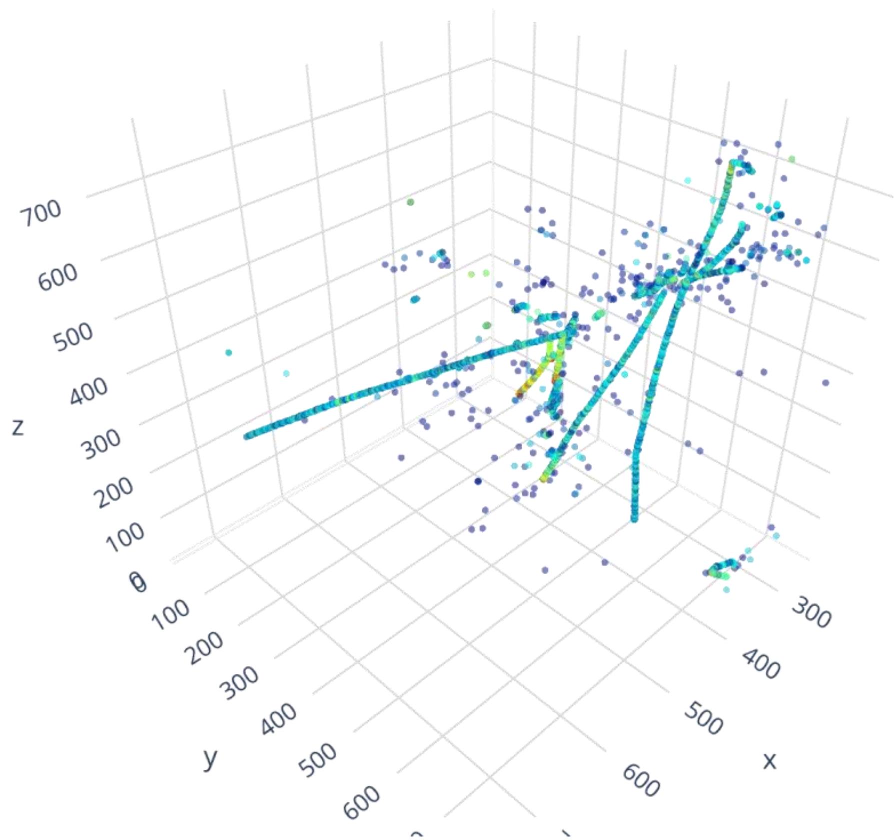
Experimental Applications (NPML Talks):

- **Ran Itay**: Using Sparse Convolutional Neural Networks in MicroBooNE
- **L. Domine and P. Tsang**: Enabling A Deep Neural Networks based 3D LArTPC Data Reconstruction Chain for ICARUS



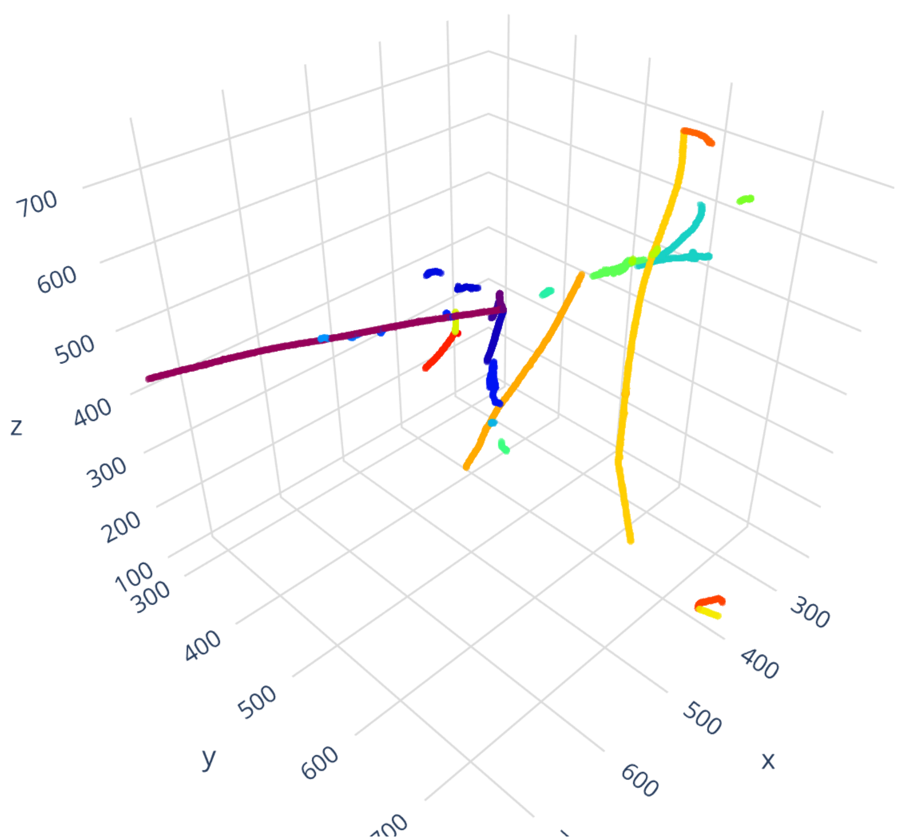
Dense Clustering: cluster voxels that belong to the same particle instance.

Energy Depositions



Remove low-E depositions with semantic labels

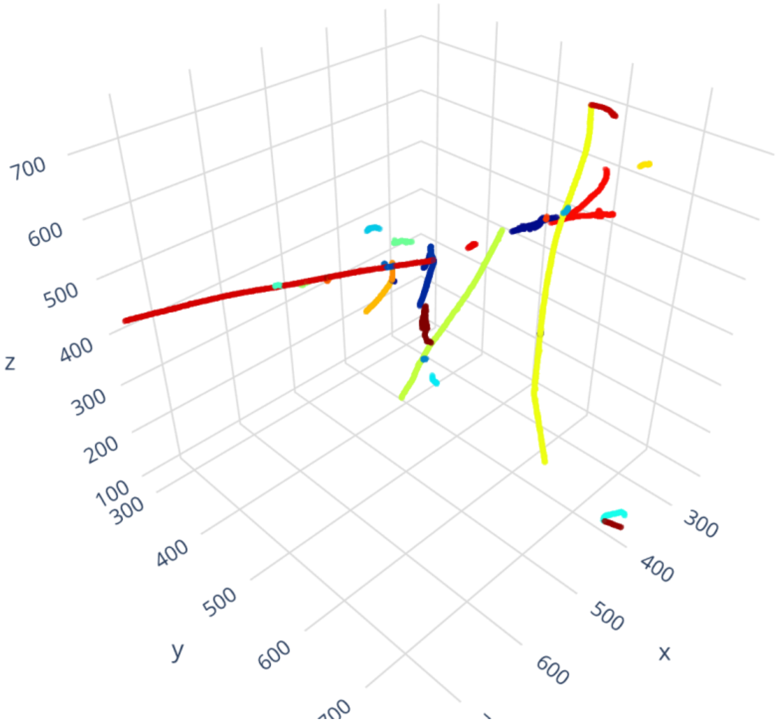
Particle Fragment Labels

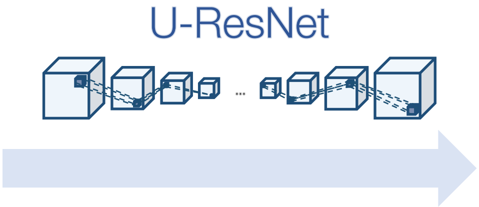


Dense Clustering: cluster voxels that belong to the same particle instance.

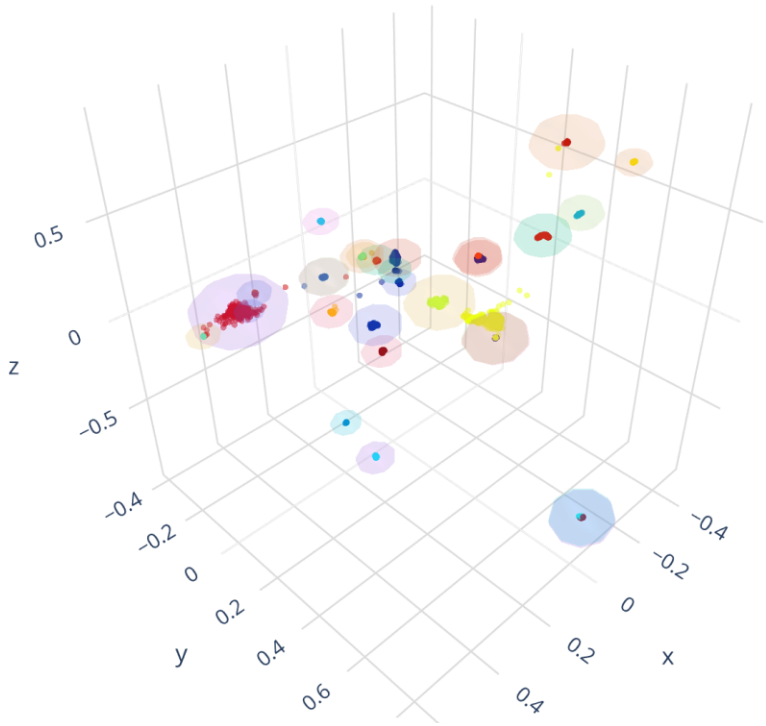
- Supervised training of embedding generation, points that belong to the same particle instance are grouped nearby each other in embedding space

Energy Depositions





Voxel Embeddings

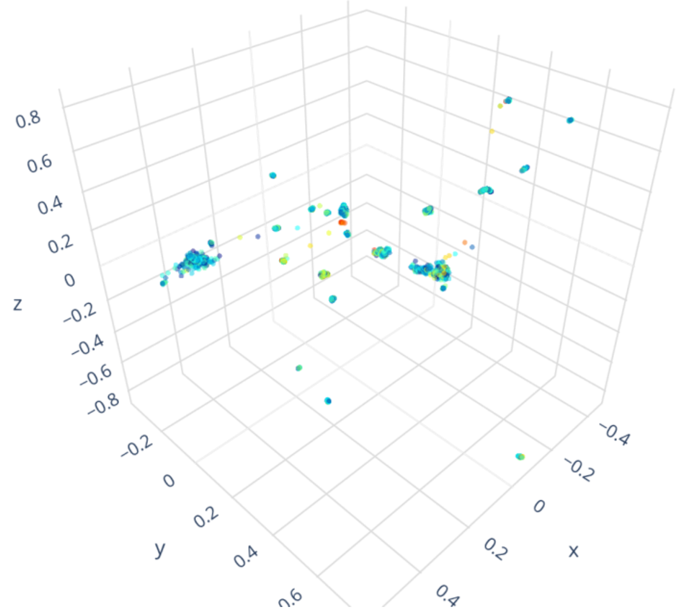


SLAC NATIONAL ACCELERATOR LABORATORY

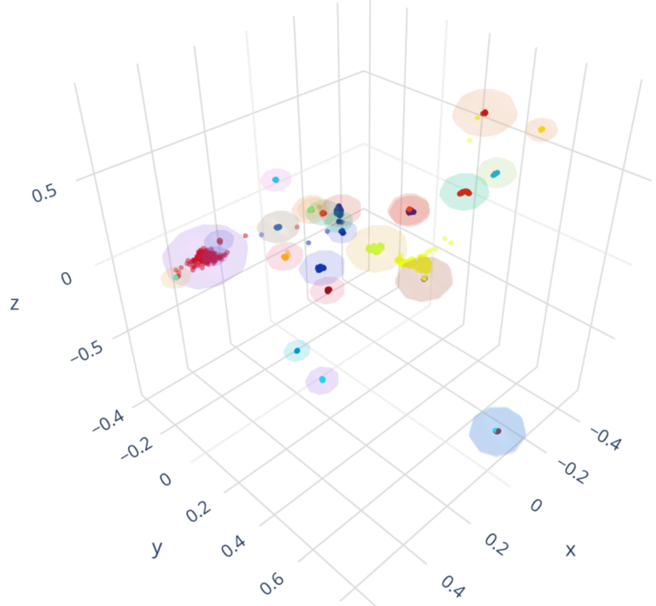
Dense Clustering: cluster voxels that belong to the same particle instance.

- Supervised training of embedding generation, points that belong to the same particle instance are grouped nearby each other in embedding space (ref: D. Koh et. al., https://arxiv.org/abs/2007.03083)
- Run a clustering algorithm on embedding space, generate predicted particle cluster labels

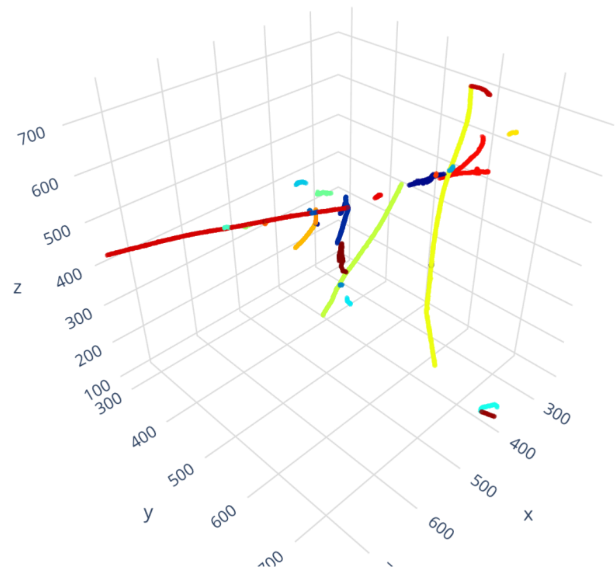
1. Voxel Embeddings



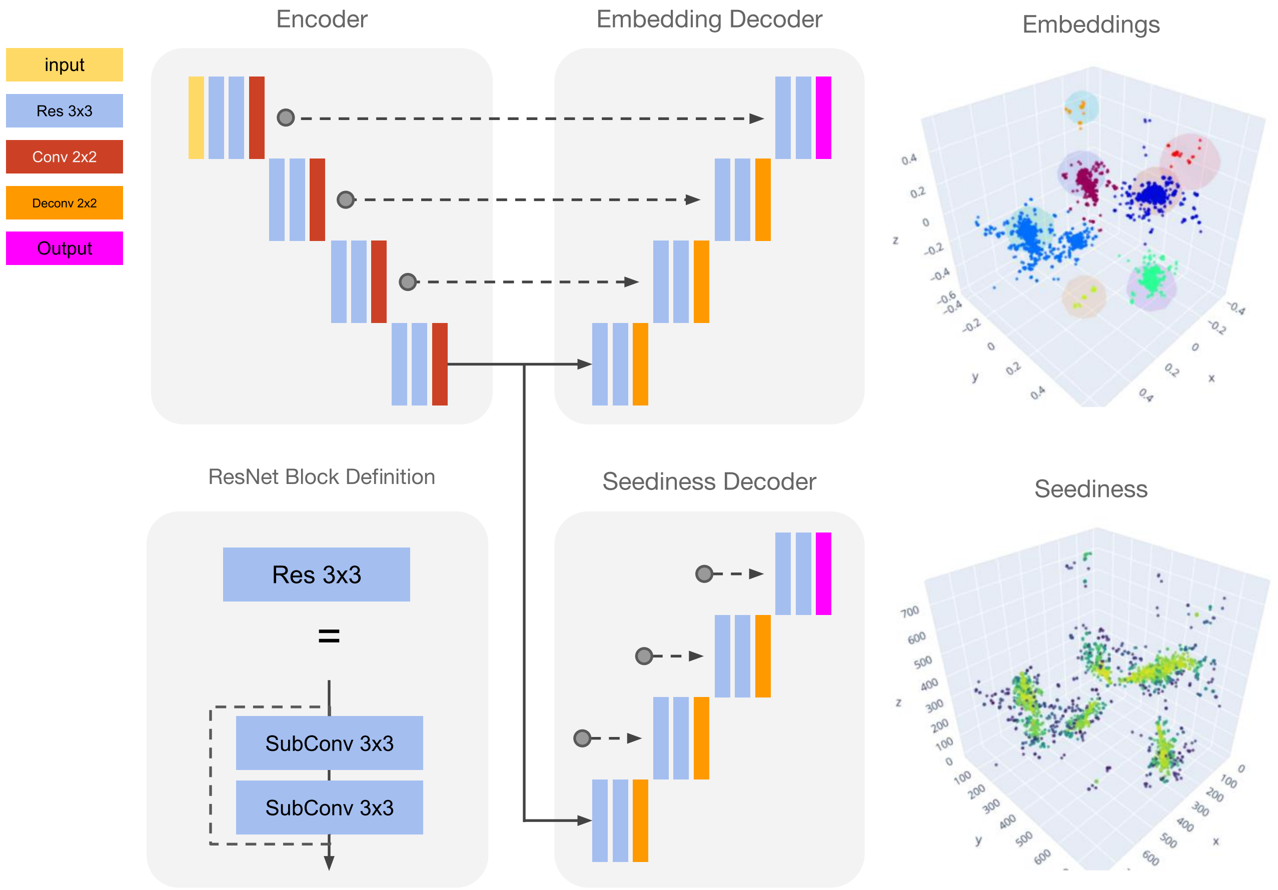
2. Gaussian Kernel Clustering



3. Predicted Cluster Labels



We use a U-ResNet with one shared encoder and two decoders:

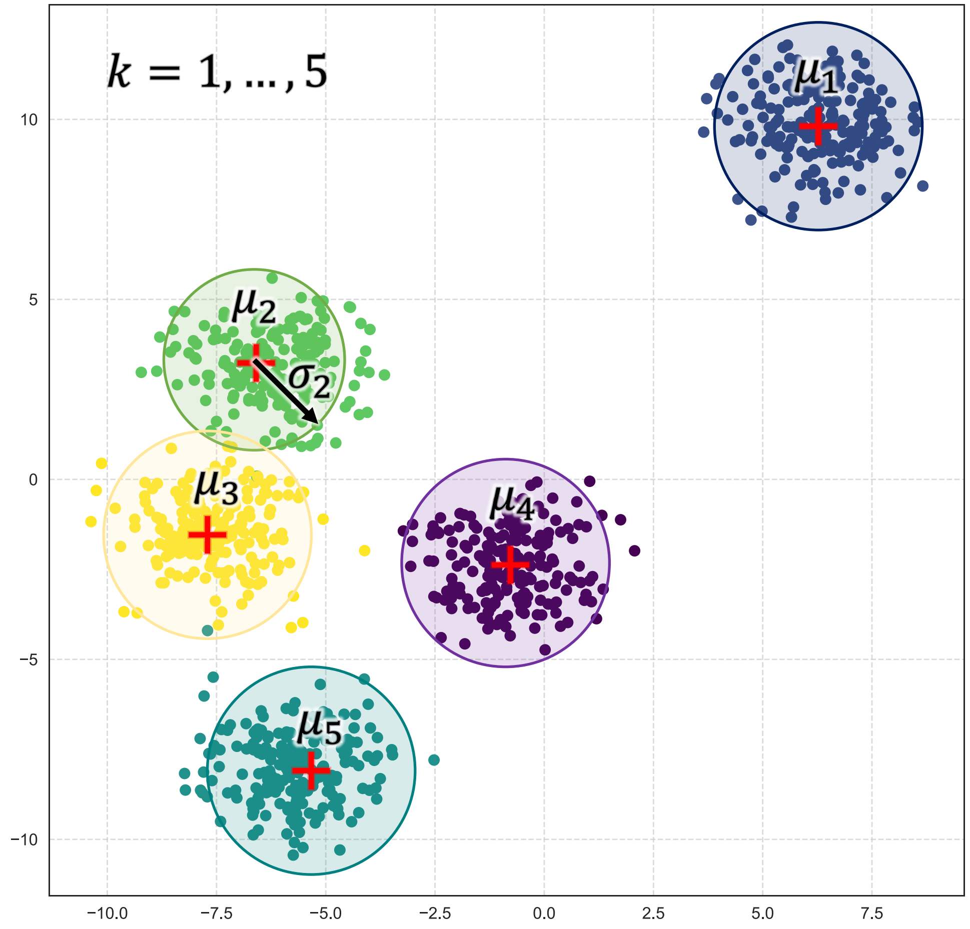


Embeddings: transform image voxel to representation in embedding space, predict size of cluster σ:

$$f_{emb}(x_i) \in [-1, 1]^3, \qquad \sigma(x_i) \in [0,2].$$

Seediness: trained to predict a score between 0 and 1, indicating proximity to embedding space centroid.

$$s_i \equiv s(x_i) \in [0,1].$$



We present a summary of the proposal-free algorithm by D. Neven et. al. (ref: https://arxiv.org/abs/1906.11109)

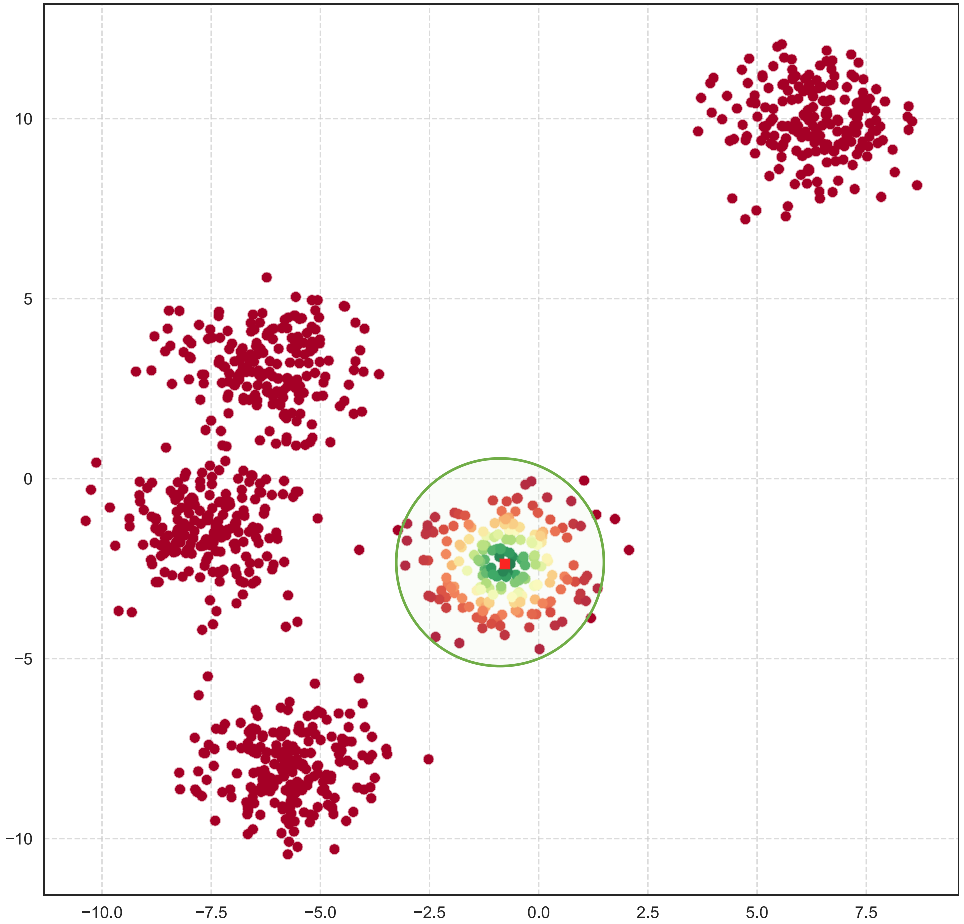
Training:

- For each true cluster k, compute centroid μ_k and average margin σ_k:

$$\mu_k = \frac{1}{n_k}\sum_{i=1}^{n_k} f_{emb}(x_i), \qquad \sigma_k = \frac{1}{n_k}\sum_{i=1}^{n_k} \sigma(x_i).$$

Note: Following scatterplots are for illustration purposes only.

SLAC NATIONAL ACCELERATOR LABORATORY



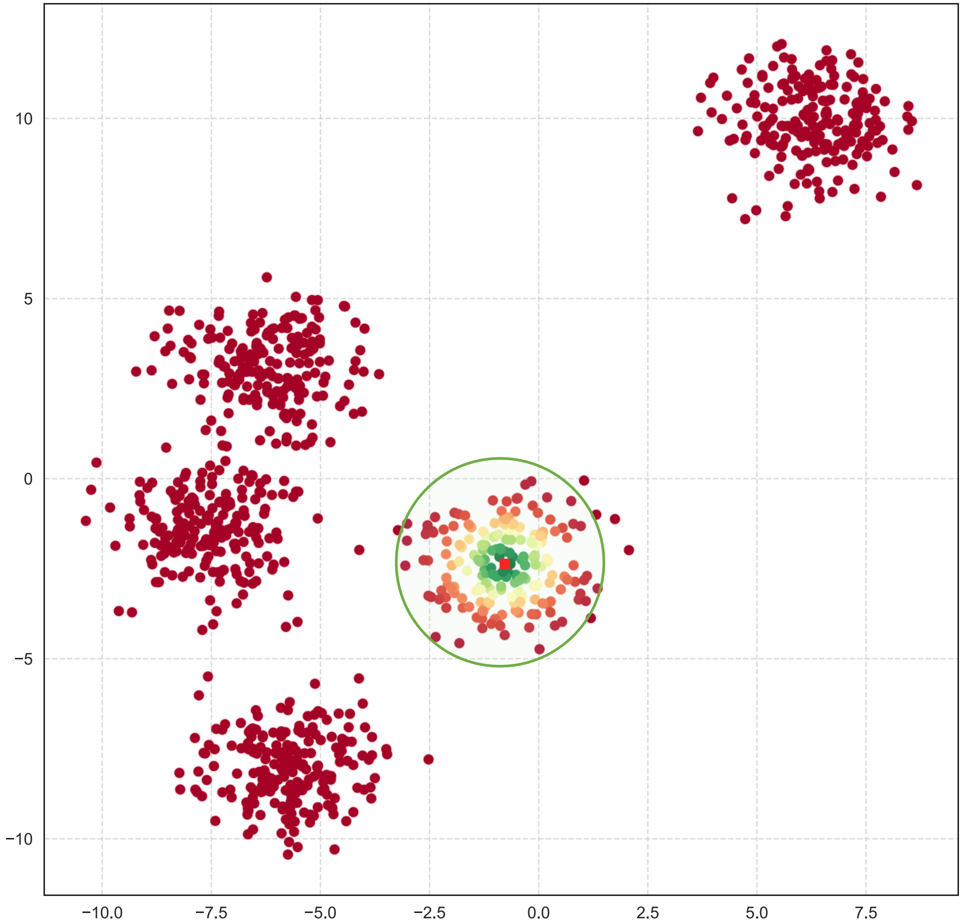
Training:

- For each true cluster k, compute centroid μ_k and average margin σ_k:

$$\mu_k = \frac{1}{n_k}\sum_{i=1}^{n_k} f_{emb}(x_i), \qquad \sigma_k = \frac{1}{n_k}\sum_{i=1}^{n_k} \sigma(x_i).$$

- Define a kernel map $p(\cdot\,; \mu_k, \sigma_k)$, where:

$$p(x; \mu_k, \sigma_k) = \exp\left(-\frac{\left|\left|f_{emb}(x_i) - \mu_k\right|\right|^2}{2\sigma_k^2}\right).$$



Training:

- For each true cluster k, compute **centroid** μ_k and average **margin** σ_k:

$$\mu_k = \frac{1}{n_k}\sum_{i=1}^{n_k} f_{emb}(x_i), \qquad \sigma_k = \frac{1}{n_k}\sum_{i=1}^{n_k} \sigma(x_i).$$

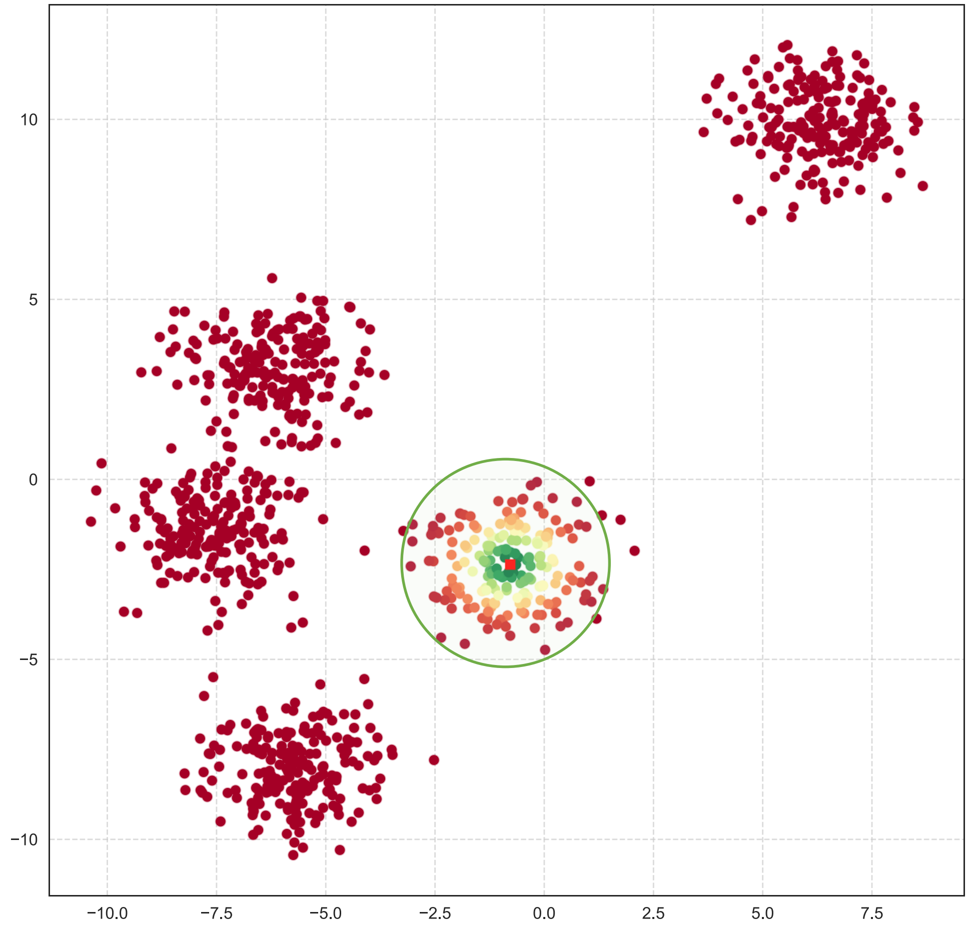
- Define a kernel map $p(\cdot\,; \mu_k, \sigma_k)$, where:

$$p_{ik} \equiv p(x_i; \mu_k, \sigma_k) = \exp\left(-\frac{\|f_{emb}(x_i) - \mu_k\|^2}{2\sigma_k^2}\right).$$

- Now problem is a binary classification over scores p:

$$L_{emb}(x_{ik}) = y_{ik}\log(p_{ik}) + (1 - y_{ik})\log(1 - p_{ik}).$$

SLAC NATIONAL ACCELERATOR LABORATORY



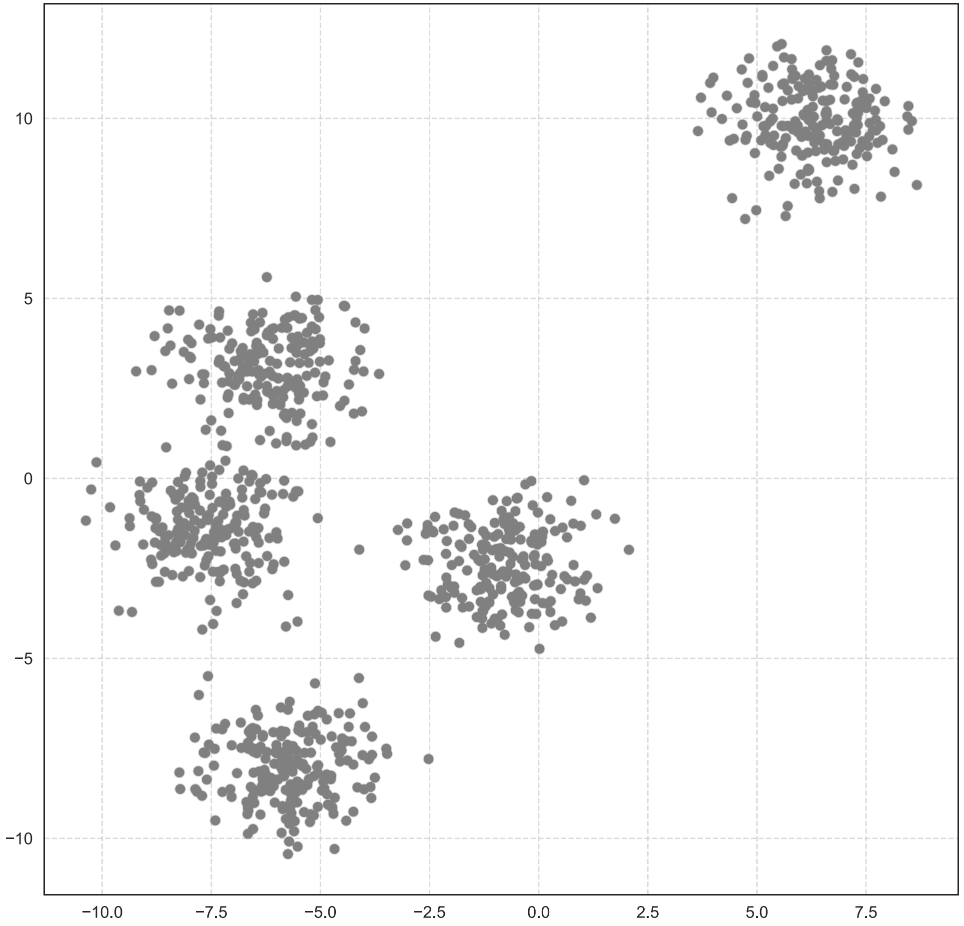
Training:

- Observe that the kernel score give a proximity score towards the centroid:

$$p_{ik} \equiv p(x_i; \mu_k, \sigma_k) = \exp\left(-\frac{||f_{emb}(x_i) - \mu_k||^2}{2\sigma_k^2}\right).$$

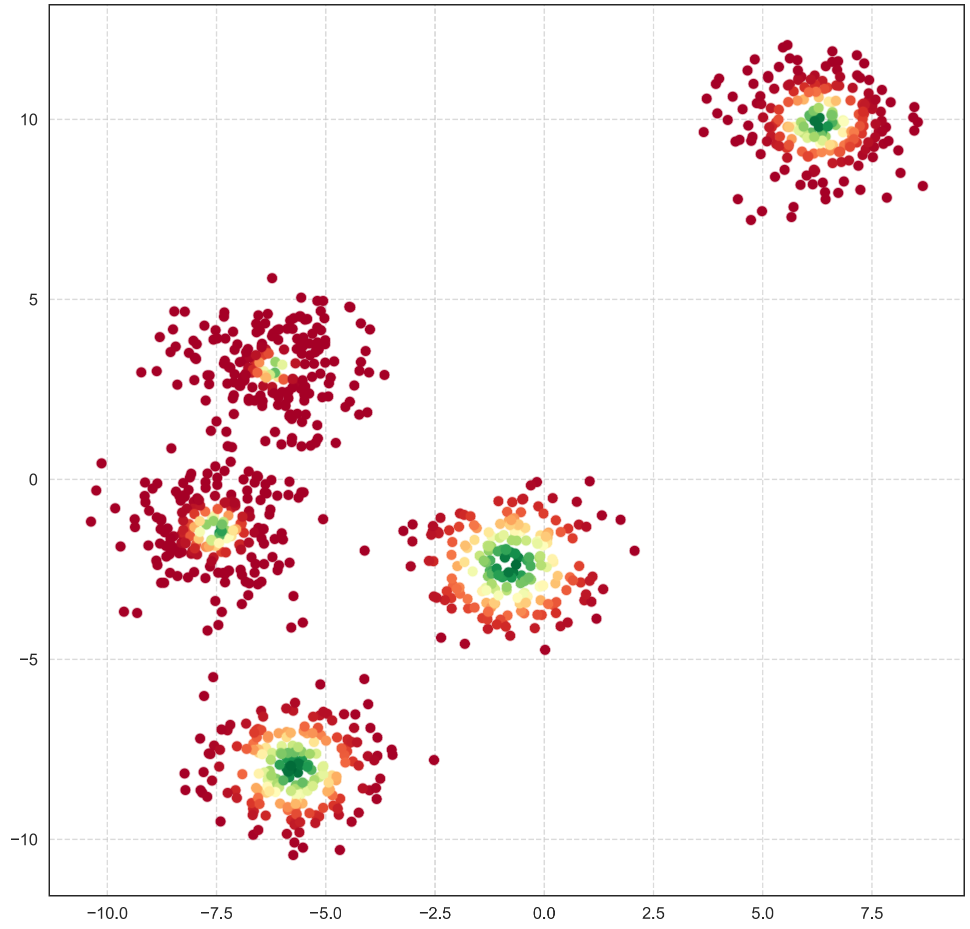
- To train seediness, we simply take p_{ik} as truth and supervise seediness branch to predict p_{ik}:

$$L_{seed}(x_i) = ||s(x_i) - p_{ik}||^2$$



Inference:

- At inference time, we do not have labels to define the centroids μ_k and margins σ_k.

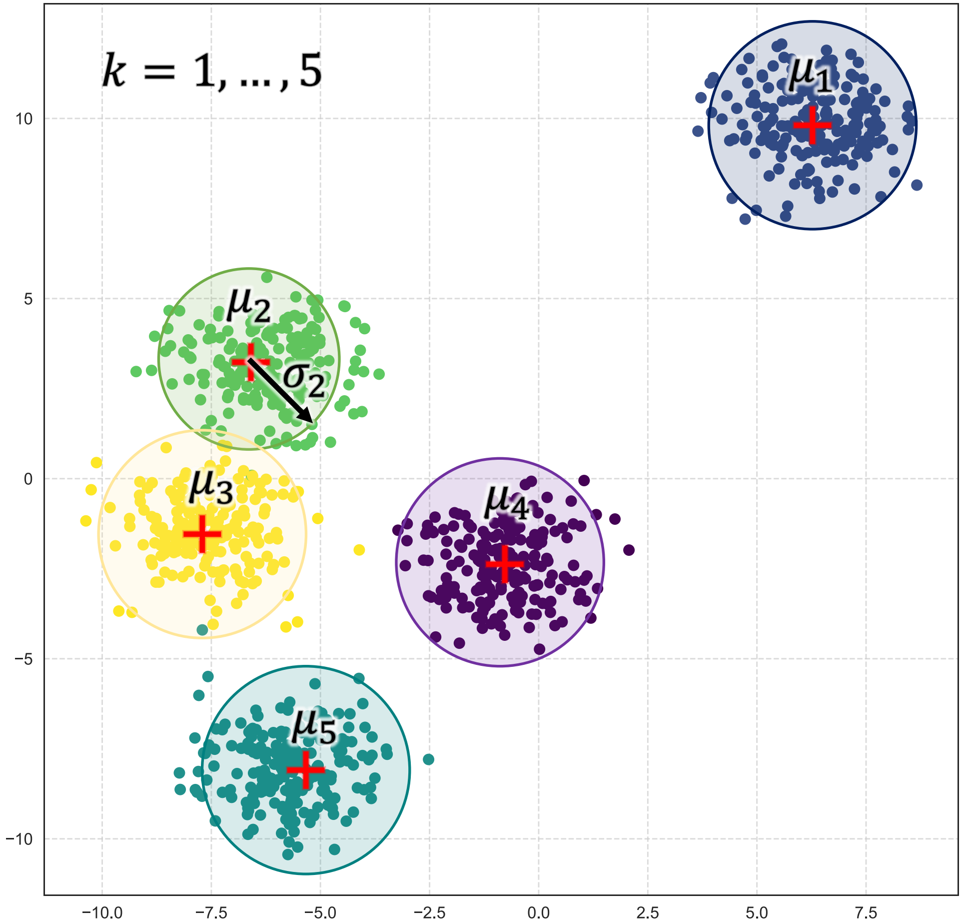


Inference:

- At inference time, we do not have labels to define the centroids μ_k and margins σ_k.
- The **seediness** predictions s_i (color) indicate likely candidates for cluster centroids.

Inference:

- At inference time, we do not have labels to define the centroids μ_k and margins σ_k.
- The **seediness** predictions s_i (color) indicate likely candidates for cluster centroids.
- Start from highest seediness point as μ_k, and assign embeddings to cluster k by thresholding on kernel score p_{ik}.



Inference:

- At inference time, we do not have labels to define the centroids μ_k and margins σ_k.
- The **seediness** predictions s_i (color) indicate likely candidates for cluster centroids.
- Start from highest seediness point μ_k, and assign embeddings to cluster k by thresholding on kernel score p_{ik}.
- Repeat until either:
  - All voxel have been clustered
  - Seediness goes below a threshold value s_0.

- Train and evaluate network on Monte-Carlo generated open dataset (PILArNet: https://arxiv.org/abs/2006.01993)
- Dataset (https://osf.io/6gvf4/): 80k Training set,~20k validation set.
- Mean Adjusted Rand Index (ARI) = **0.973**

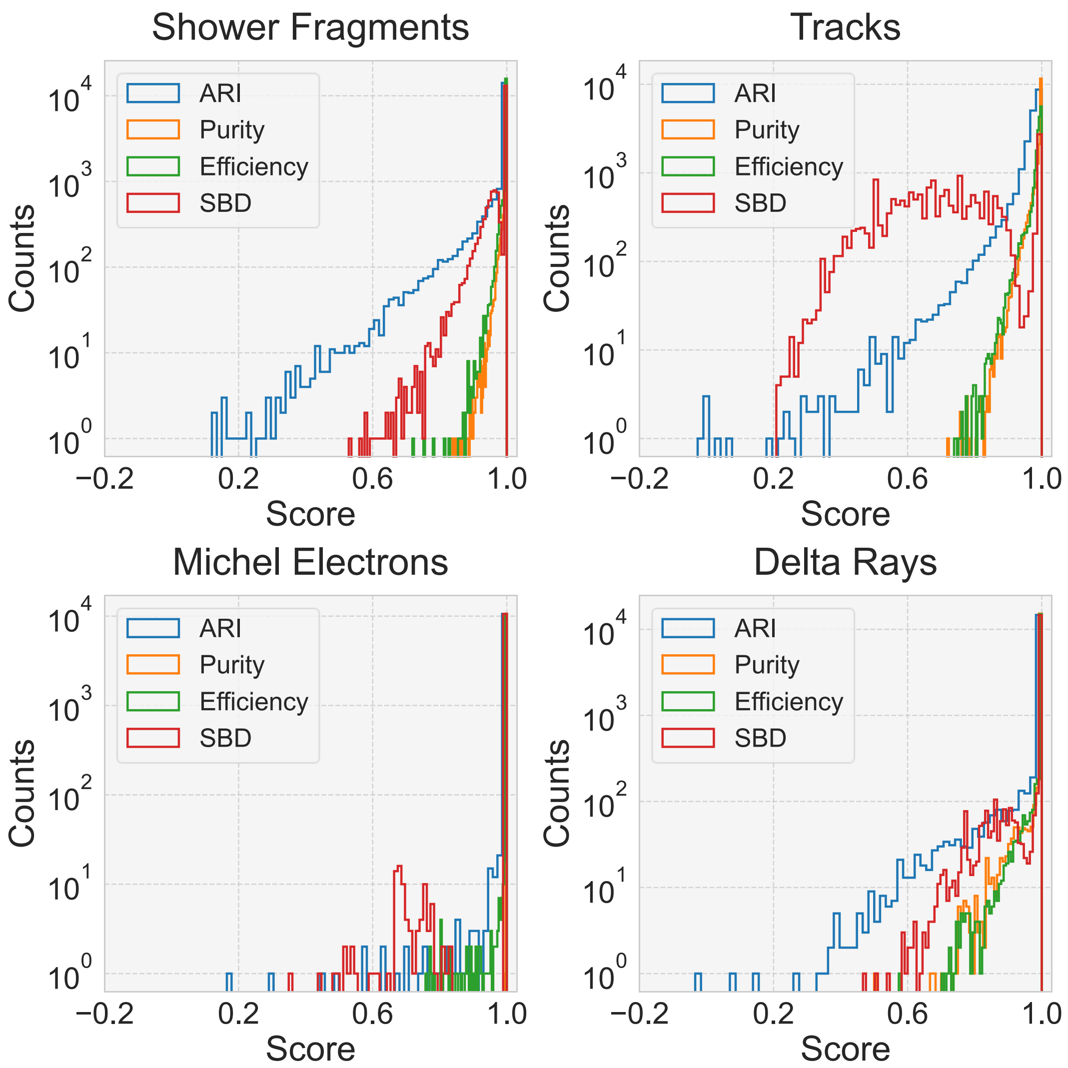
	Shower Fragments	Tracks	Michel electrons	Delta Rays
Average # per Event	15 ± 9	6 ± 3	1 ± 1	5 ± 3
Voxel Counts	21.4M	47.7M	0.793M	1.11M
Percentage of Particle Instances	57%	25%	3%	15%

- Use ground truth semantic labels and cluster within each semantic class.
- Mean Purity and Efficiency are above **99%**

	Shower Fragments	Tracks	Michel electrons	Delta Rays
ARI	0.968	0.961	0.998	0.978
Purity	0.997	0.988	0.999	0.992
Efficiency	0.996	0.983	1.000	0.998
SBD	0.978	0.717	0.998	0.982

- Use ground truth semantic labels and cluster within each semantic class.
- Mean Purity and Efficiency are above **99%**

	Shower Fragments	Tracks	Michel electrons	Delta Rays
ARI	0.968	0.961	0.998	0.978
Purity	0.997	0.988	0.999	0.992
Efficiency	0.996	0.983	1.000	0.998
SBD	0.978	0.717	0.998	0.982

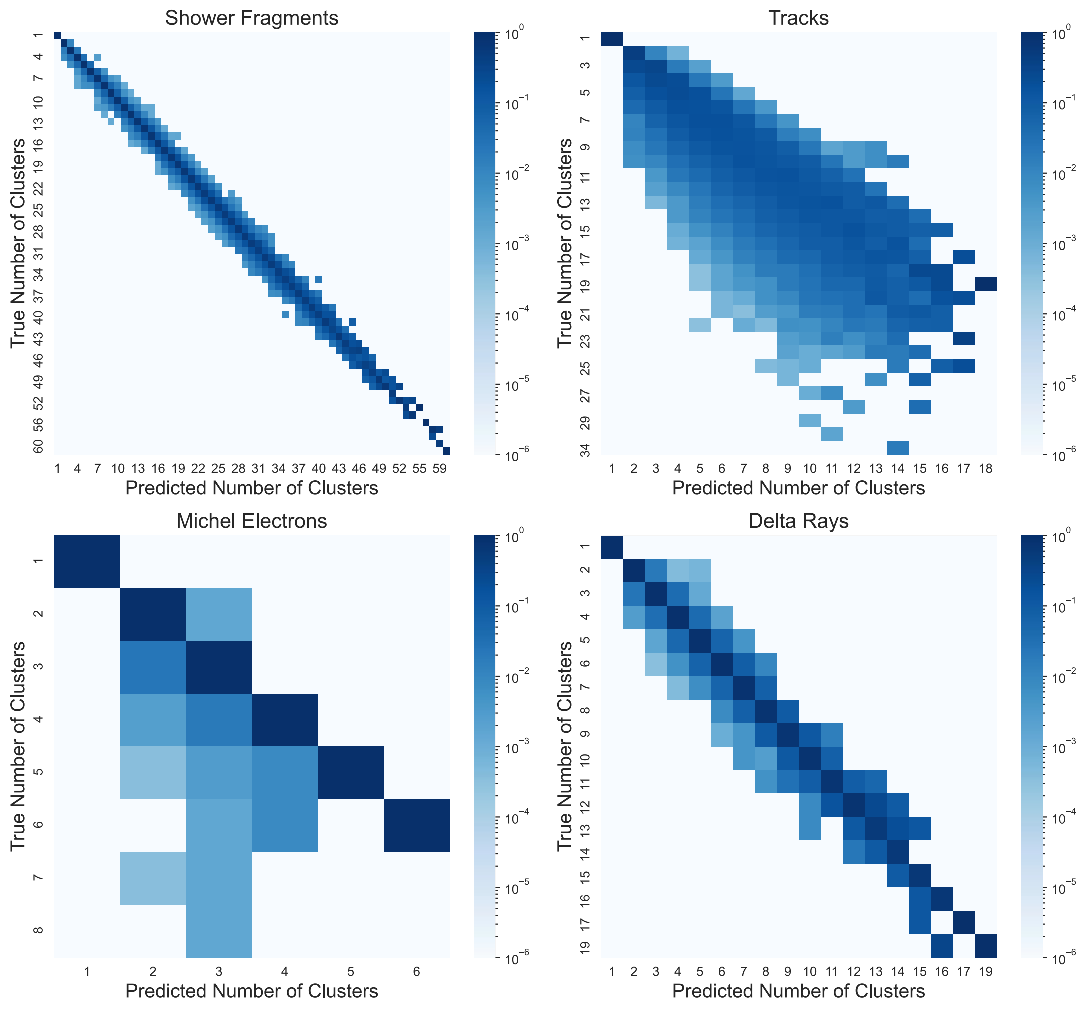


III. Results

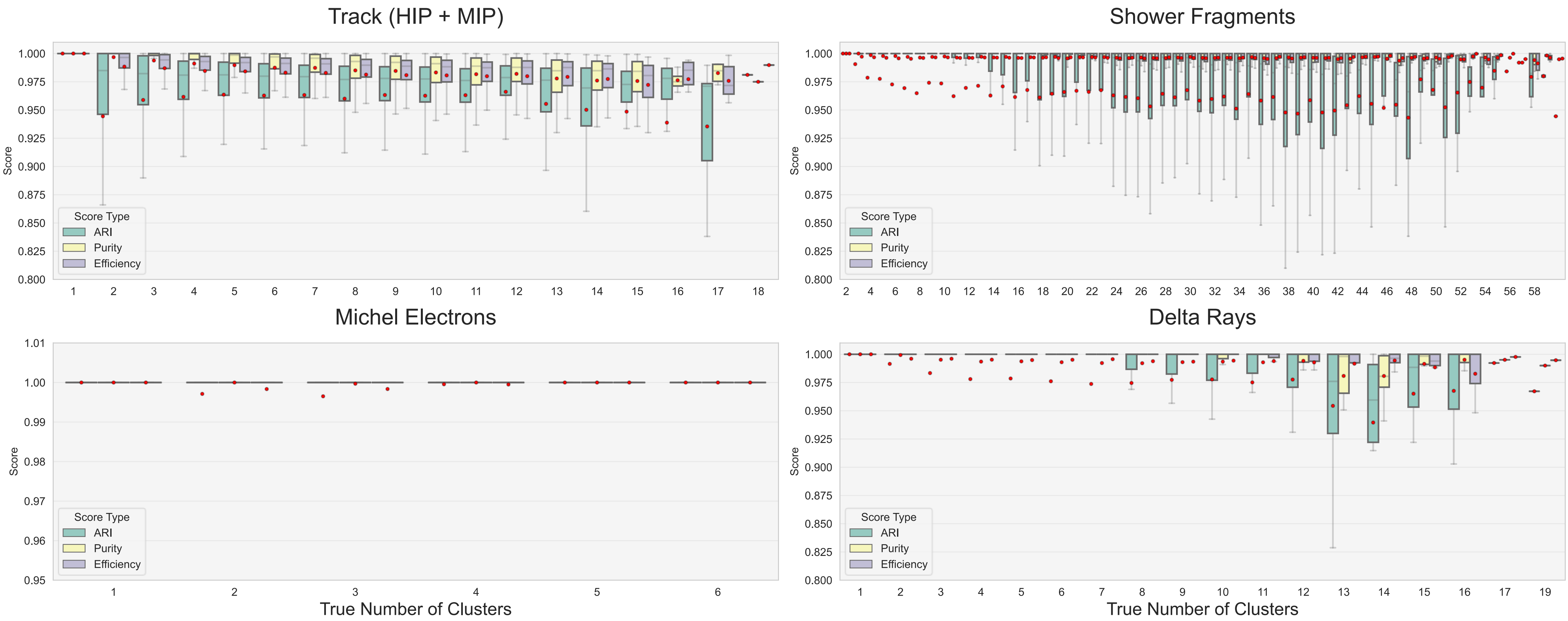
SLAC NATIONAL ACCELERATOR LABORATORY

- Use ground truth semantic labels and cluster within each semantic class.
- Mean Purity and Efficiency are above **99%**
- Mean ARI = **0.973**

	Shower Fragments	Tracks	Michel electrons	Delta Rays
ARI	0.968	0.961	0.998	0.978
Purity	0.997	0.988	0.999	0.992
Efficiency	0.996	0.983	1.000	0.998
SBD	0.978	0.717	0.998	0.982

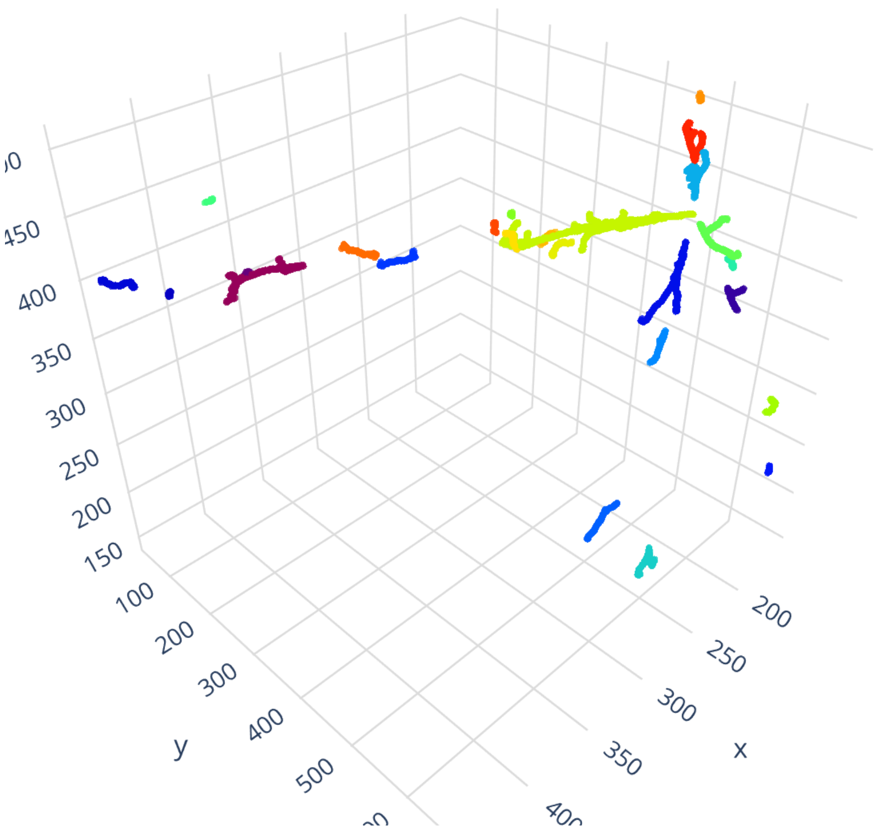


- Some performance degradation towards higher number of particles, yet averaged values are roughly stable.

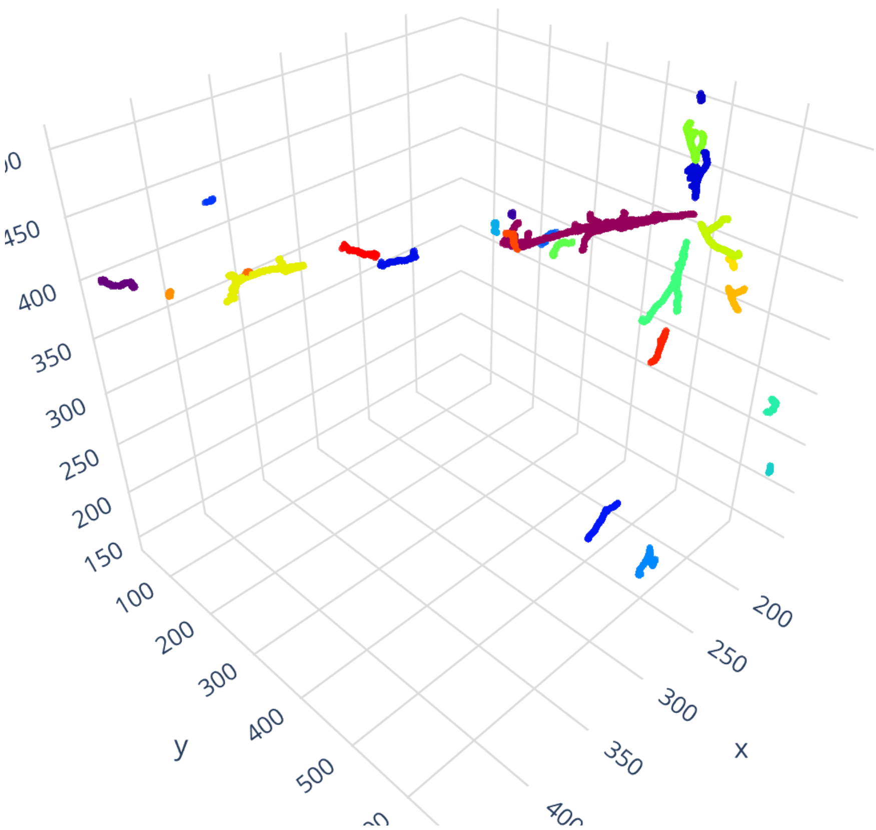


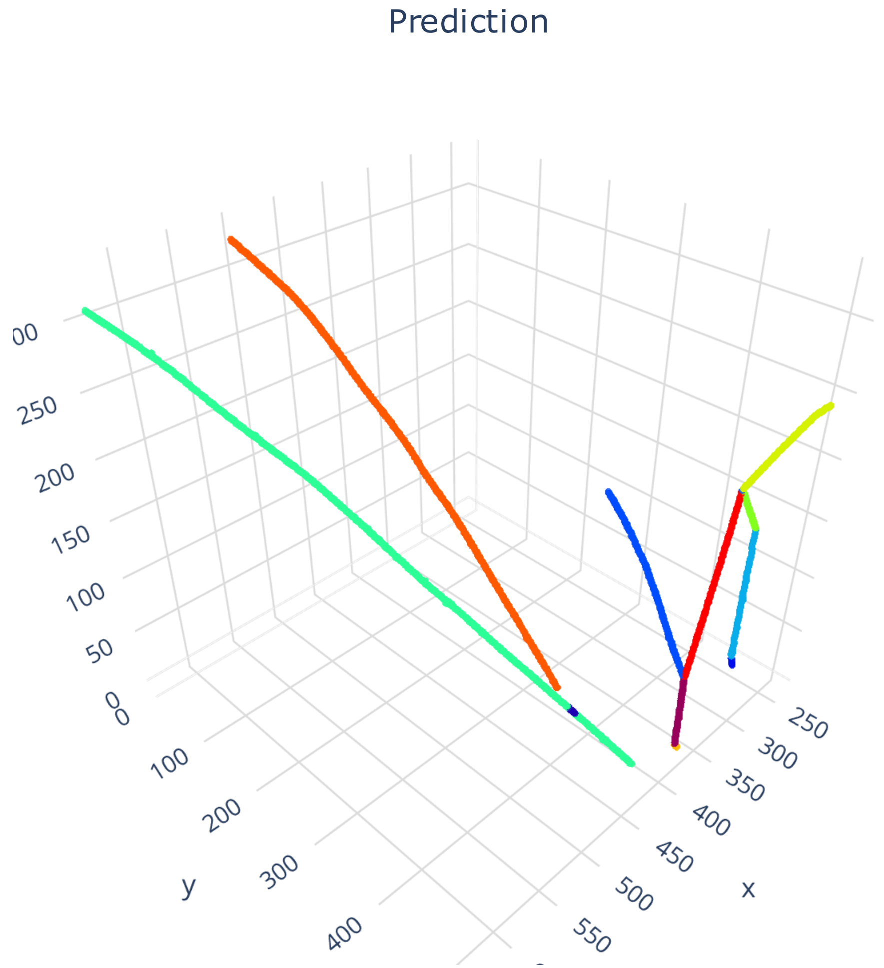
SLAC NATIONAL ACCELERATOR LABORATORY

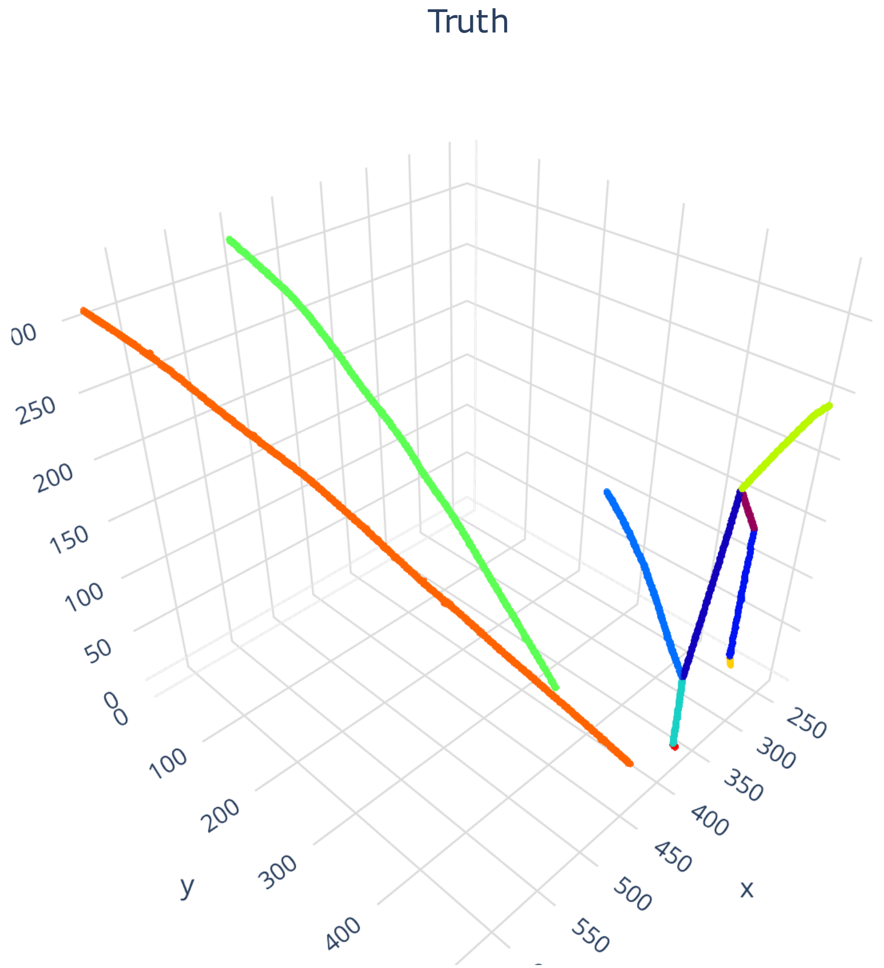
Prediction



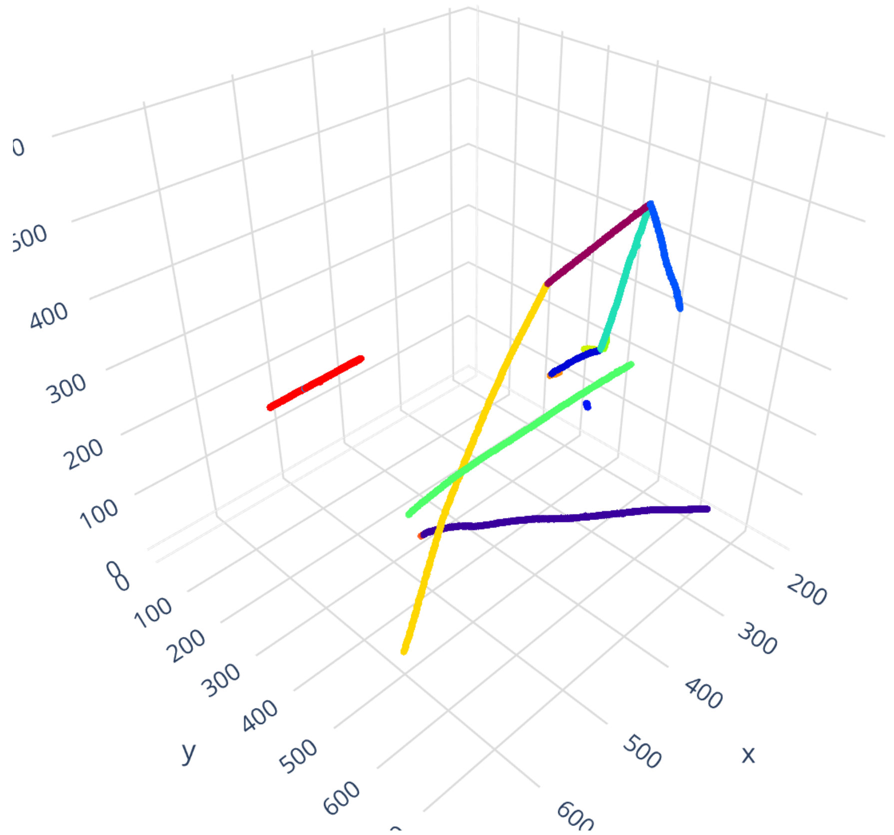
Truth



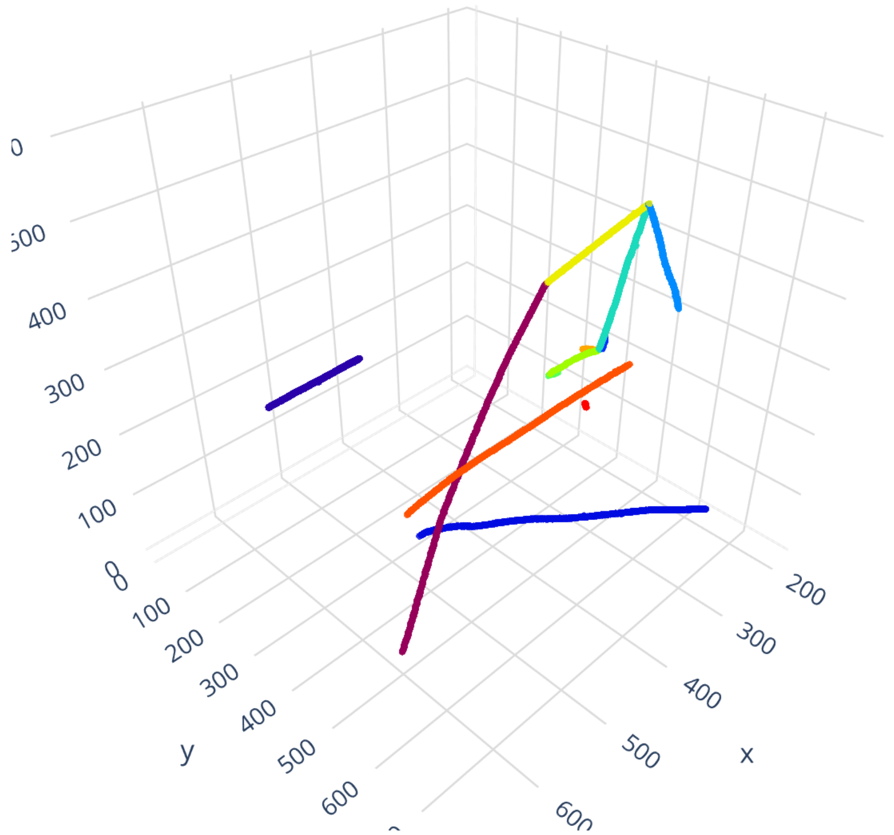

Prediction
00
250
200
150
100
50
0
0
100
200
300
400
y
250
300
350
400
450
500
550
x


Truth
00
250
200
150
100
50
0
0
100
200
300
400
y
250
300
350
400
450
500
550
x

Prediction



Truth



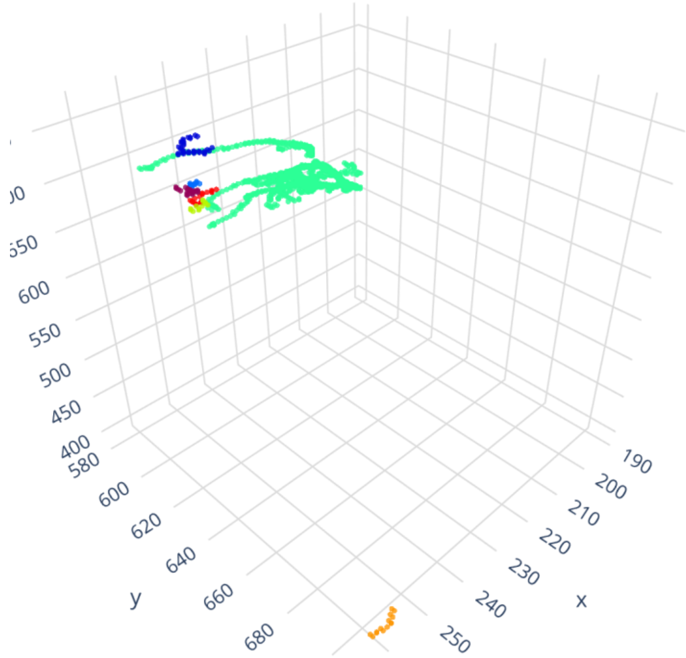
V. Conclusion

- We present a first application of proposal-free instance segmentation network to particle clustering in LArTPCs.
- Further details may be found in the arxiv preprint: D. Koh et. al., https://arxiv.org/abs/2007.03083
    - Results reproducible using Singularity containers and open github repository[1]
- Algorithm relies on accurate semantic segmentation by U-ResNet:
    - L. Domine and P. Tsang, https://arxiv.org/abs/1903.05663
- General algorithm for point cloud clustering → applications to water Cherenkov detectors.
- Future directions involve improving post-processing time complexity and addressing deficiencies for track clustering.

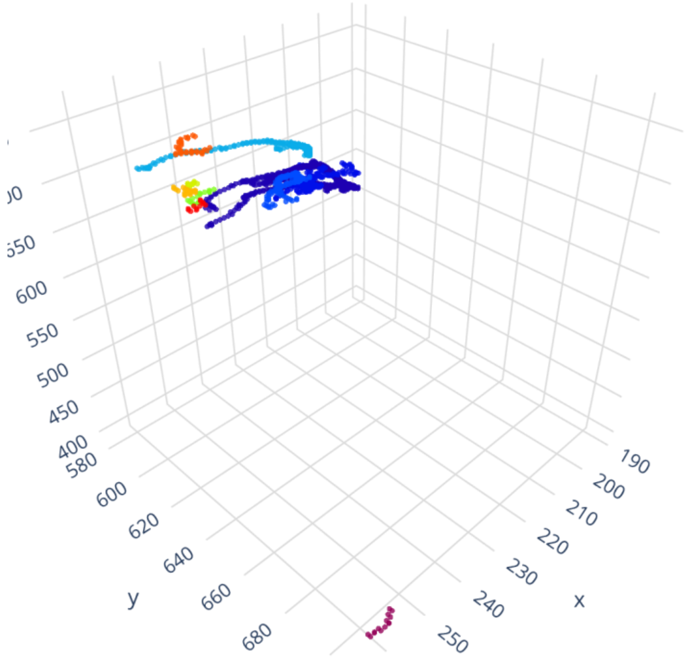
[1]: https://github.com/DeepLearnPhysics/lartpc_mlreco3d

A. Shower Mistakes

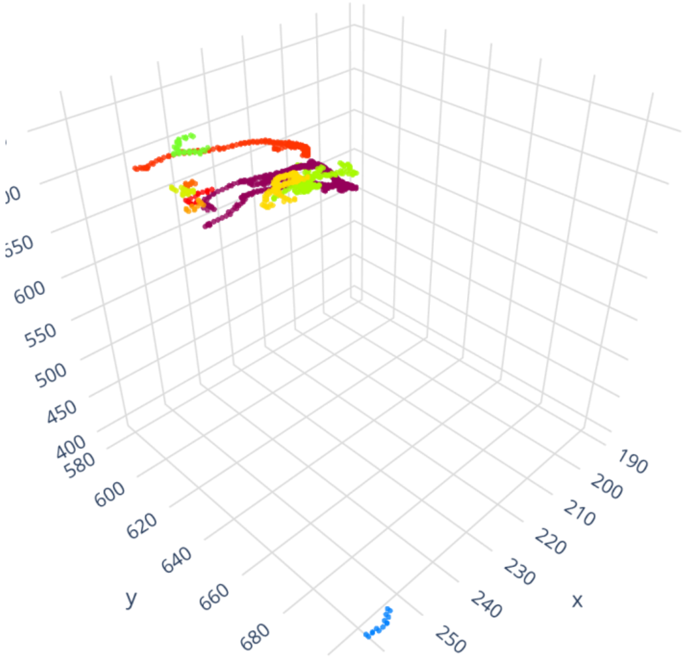
Prediction, ARI = 0.1576

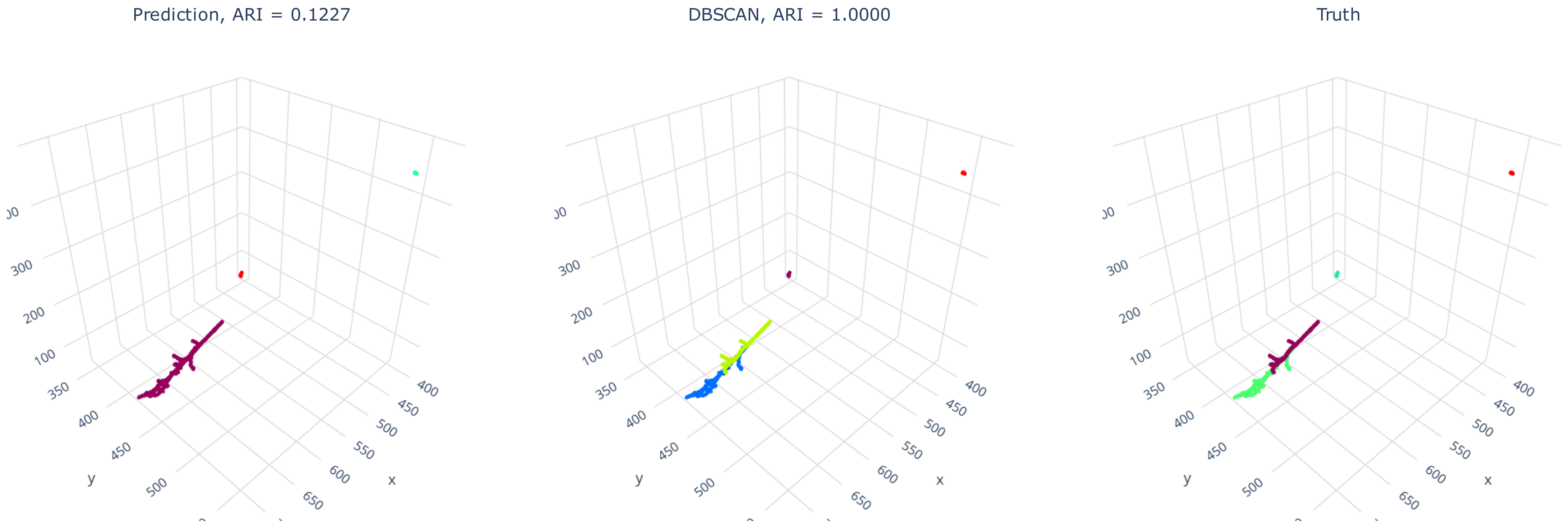


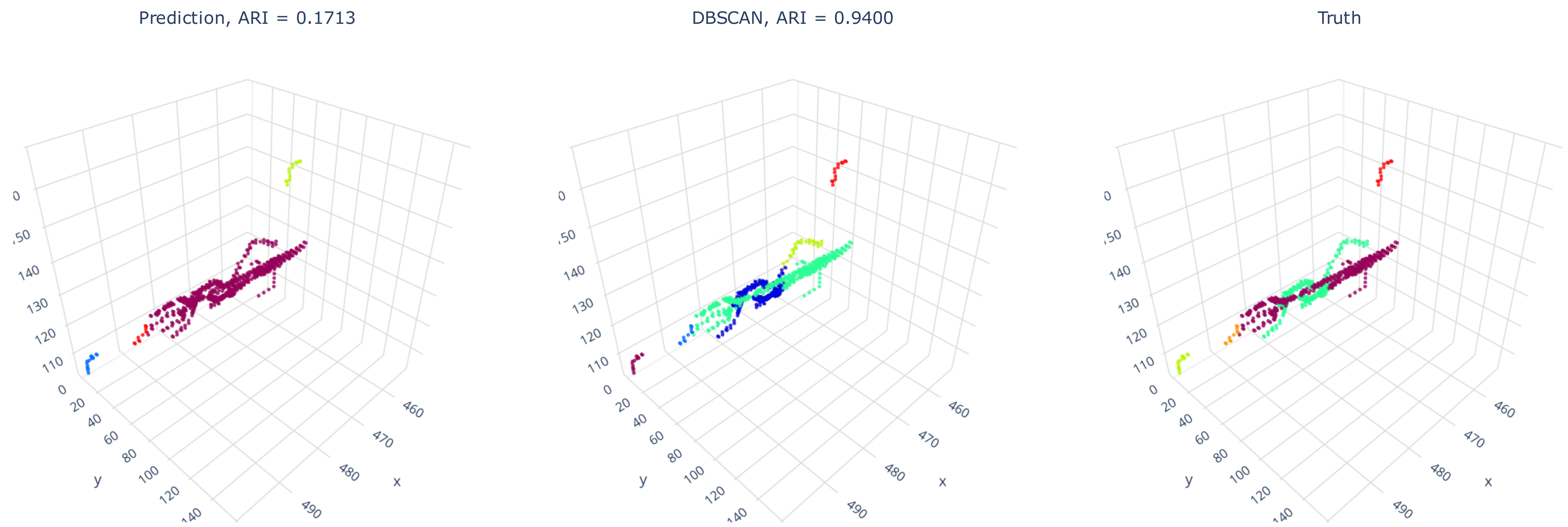
DBSCAN, ARI = 0.9994



Truth

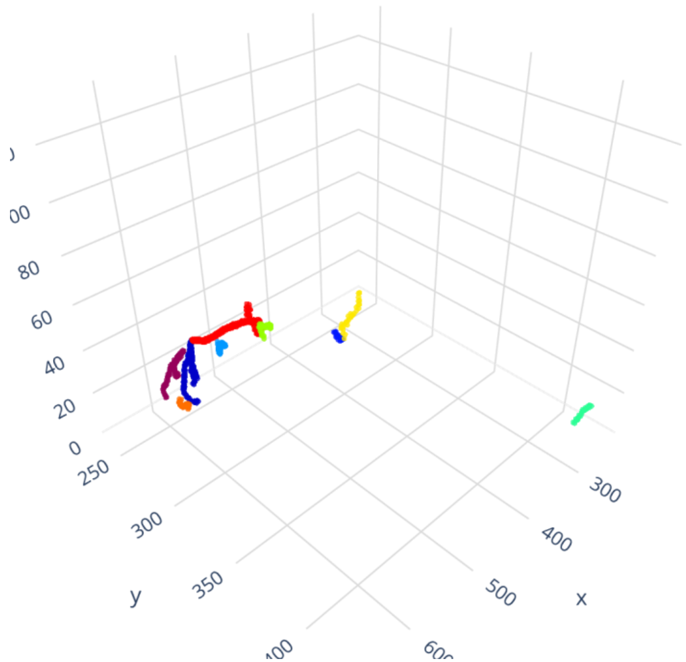



Prediction, ARI = 0.1227
DBSCAN, ARI = 1.0000
Truth
300
200
100
350
400
450
500
y
400
450
500
550
600
650
x

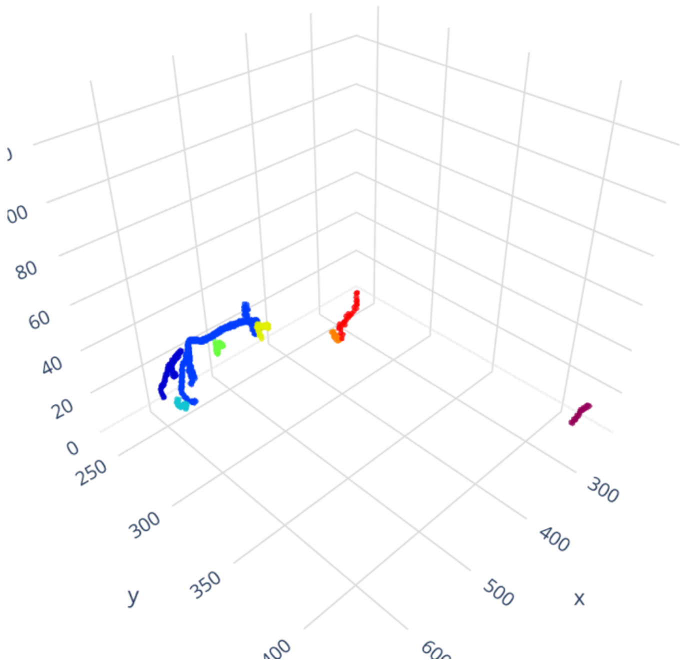

Prediction, ARI = 0.1713
DBSCAN, ARI = 0.9400
Truth
x
y
460
470
480
490
0
20
40
60
80
100
120
110
120
130
140

B. Why not DBSCAN?

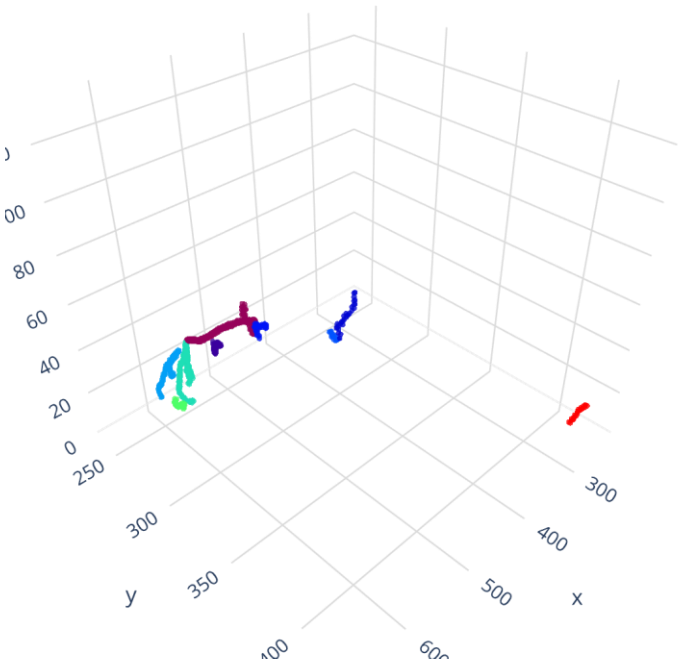
Prediction, ARI = 0.9946



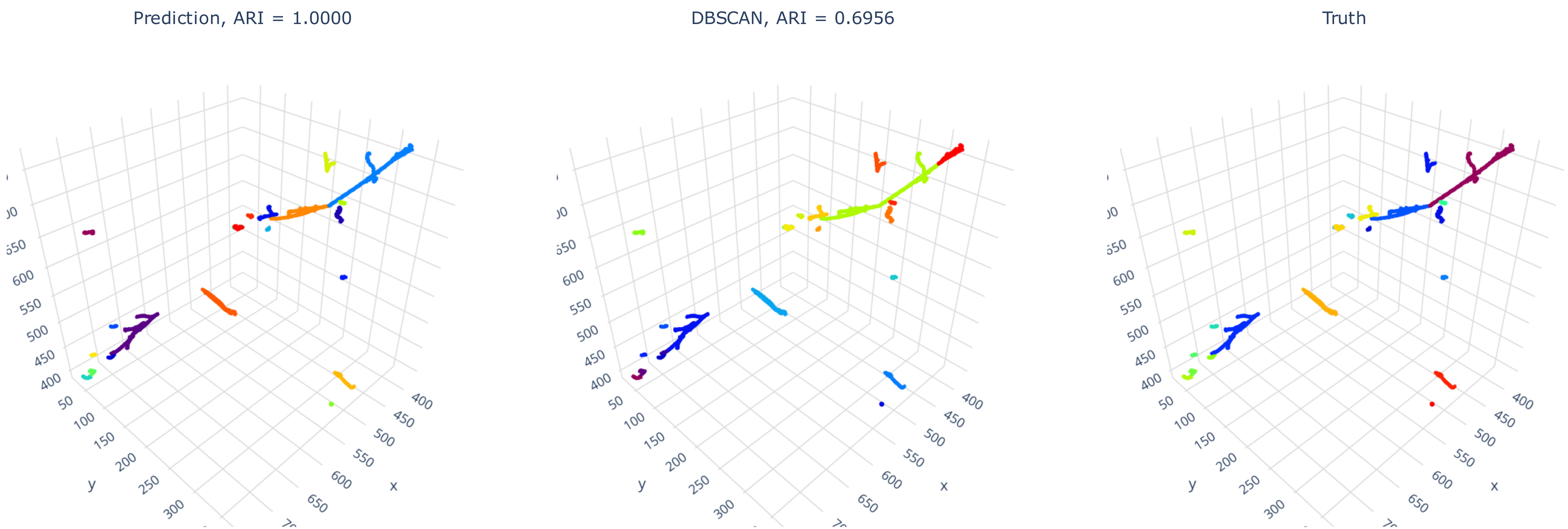
DBSCAN, ARI = 0.6057



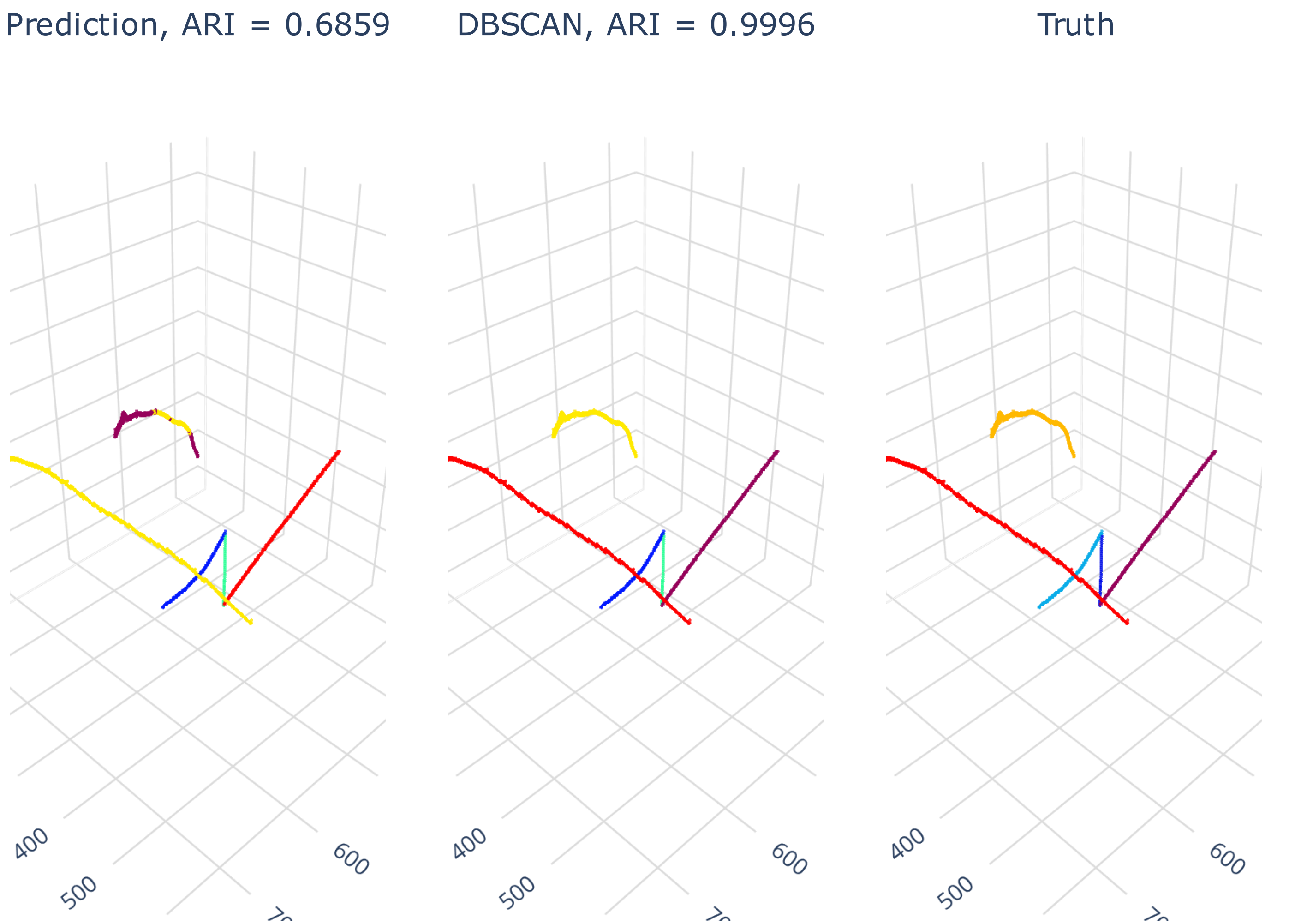
Truth

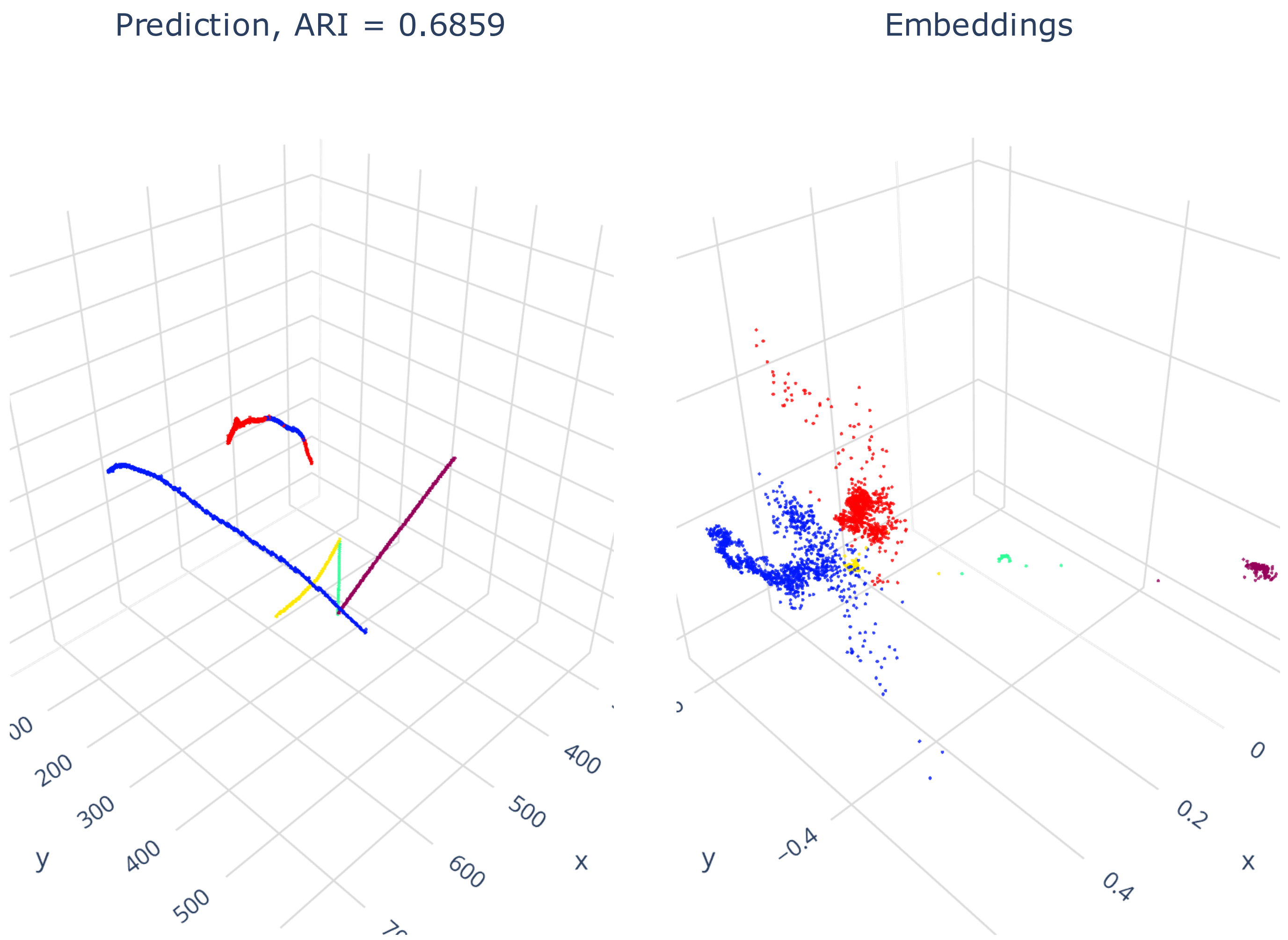


B. Why not DBSCAN?

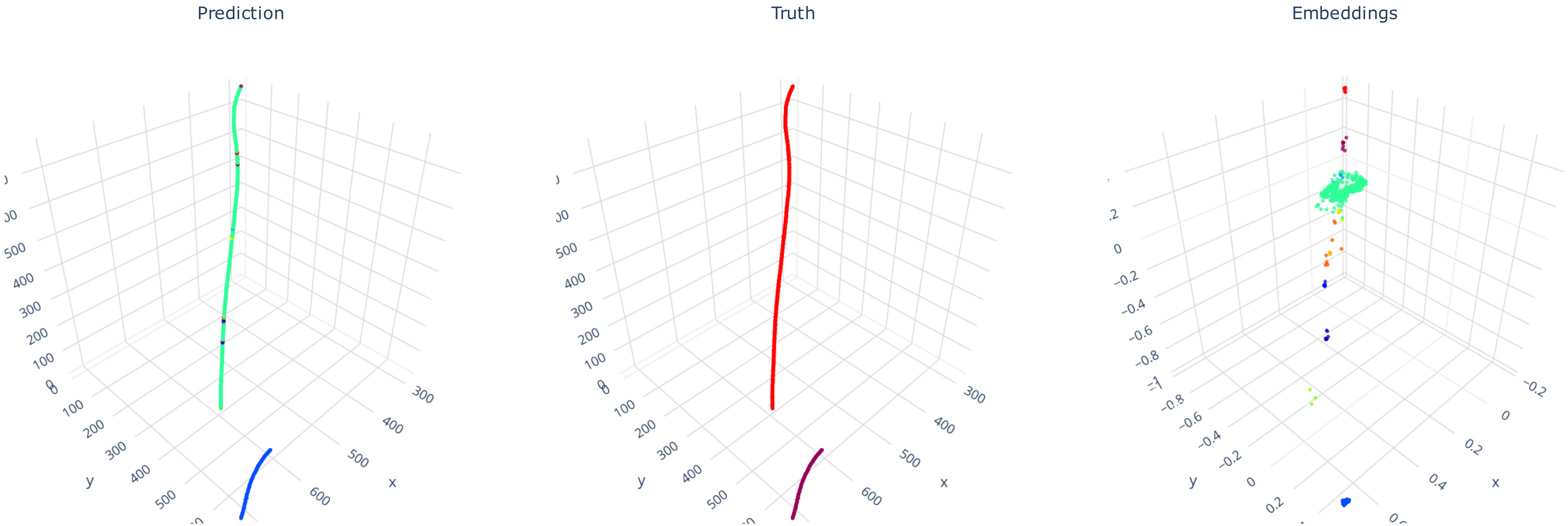


C. Track Mistakes



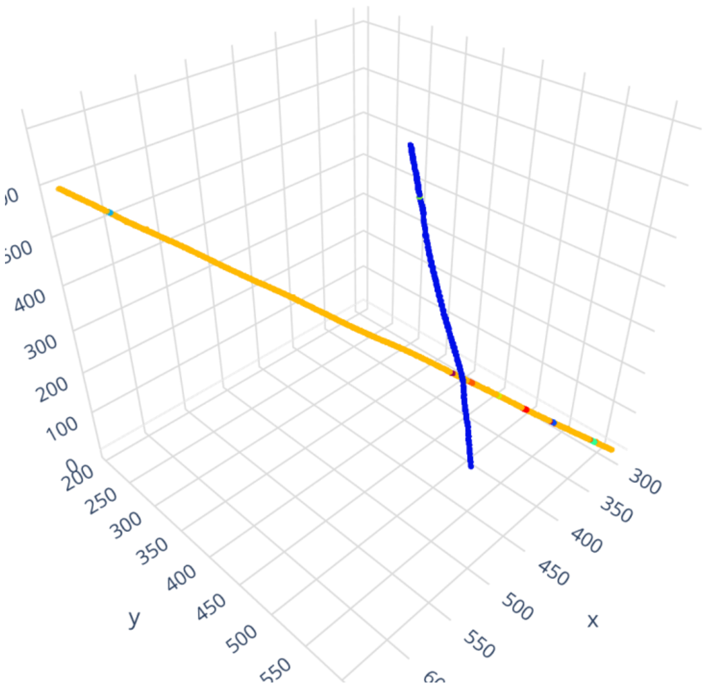

Prediction, ARI = 0.6859
Embeddings
y
x
200
300
400
500
400
500
600
y
x
−0.4
0
0.2
0.4

D. High ARI but Low SBD

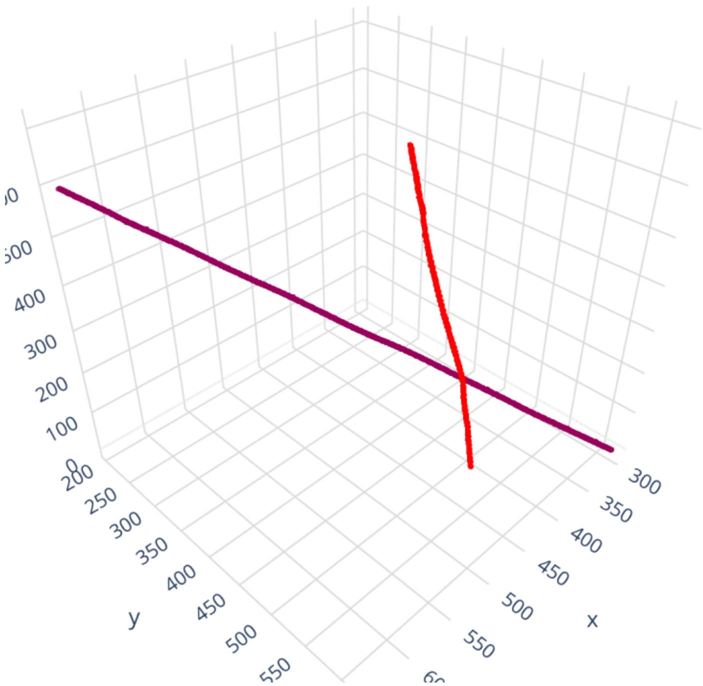


D. High ARI but Low SBD

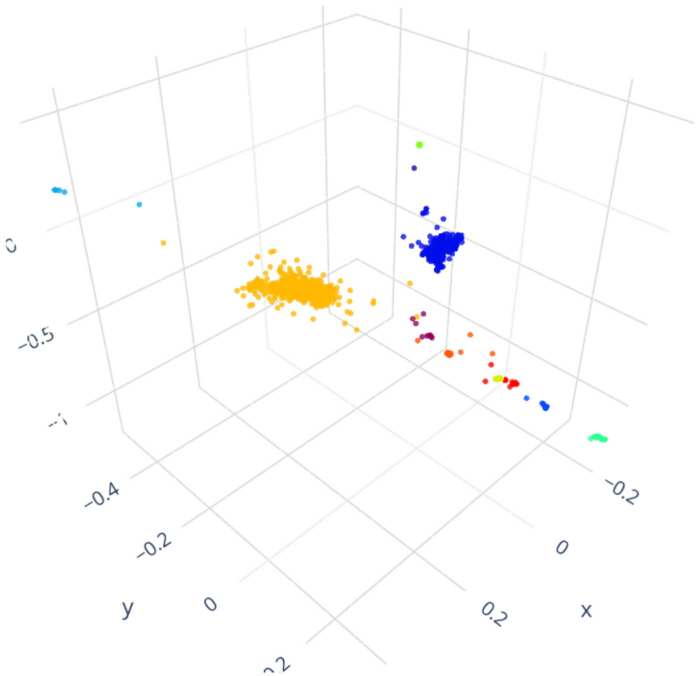
Prediction



Truth

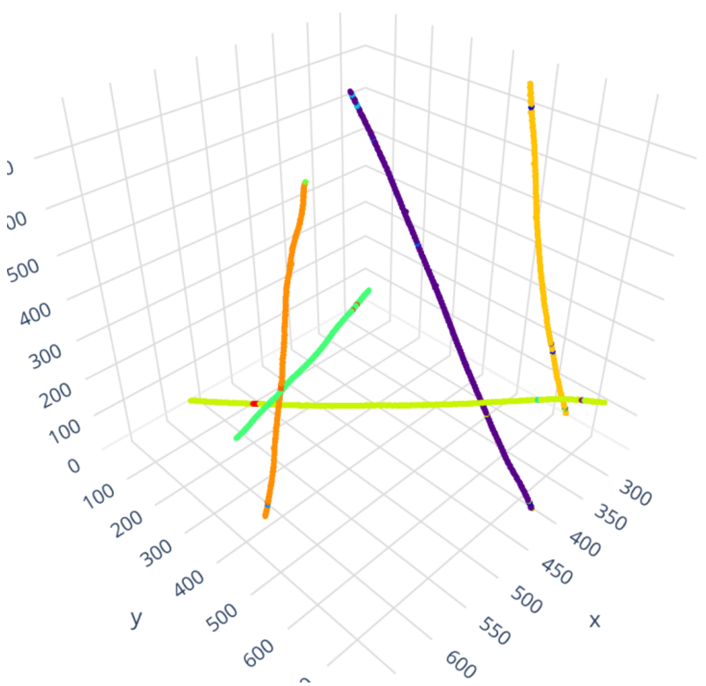


Embeddings

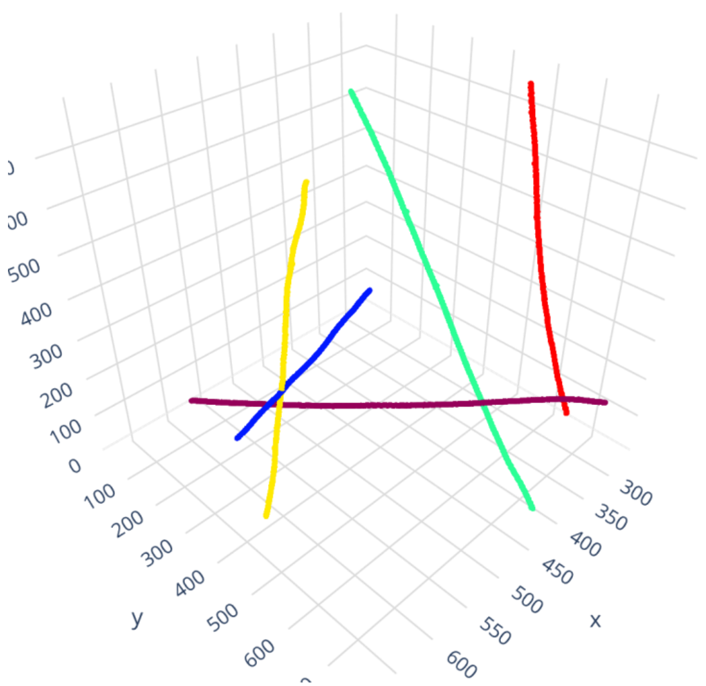


D. High ARI but Low SBD

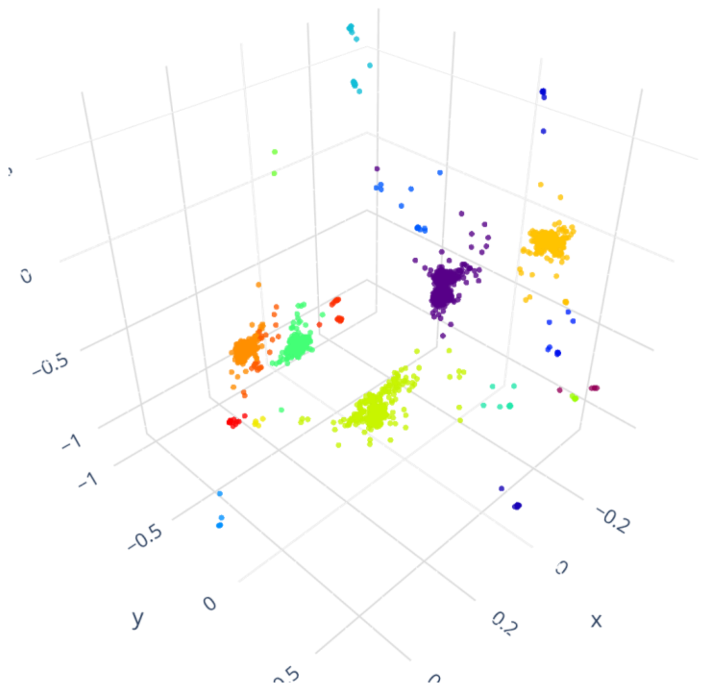
Prediction



Truth



Embeddings



E. Accuracy Metric Definitions

Definition (Purity/Efficiency for general partitions): Let $Y^* = \{S_1^*, S_2^*, \dots, S_n^*\}$ denote the ground truth partitioning of a set of points Ω, and let $\tilde{Y} = \{\widetilde{S_1}, \widetilde{S_2}, \dots, \widetilde{S_m}\}$ be the predicted partitioning of Ω. Let c_{ij} denote the entries of the contingency matrix formed by Y^* and $\tilde{Y}$. The **purity** $P(\tilde{Y}, Y^*)$ of a predicted clustering $\tilde{Y}$ with respect to ground truth Y^* is given as

$$P(\tilde{Y}, Y^*) = \frac{1}{m} \sum_{j=1}^{m} \frac{\max_{i=1,\dots,n} c_{ij}}{|\tilde{S}_j|}$$

- Purity: for a given predicted cluster of points, find the ground truth cluster with maximum pixel overlap.

Likewise, the **efficiency** $E(\tilde{Y}, Y^*)$ is defined as

$$E(\tilde{Y}, Y^*) = \frac{1}{n} \sum_{i=1}^{n} \frac{\max_{j=1,\dots,m} c_{ij}}{|S_i^*|}$$

- Efficiency: for a given ground truth cluster of points, find the predicted cluster with maximum pixel overlap.

Definition (Dice Score): Given two sets A and B, the **Dice Coefficient** is given as:

$$DSC(A,B) = \frac{2|A \cap B|}{|A| + |B|}.$$

For general partitions $Y^* = \{S_1^*, S_2^*, \dots, S_n^*\}$ and $\tilde{Y} = \{\widetilde{S_1}, \widetilde{S_2}, \dots, \widetilde{S_m}\}$, we define the **Best Dice** score of truth labels Y^* with respect to predictions $\tilde{Y}$:

$$BD(Y^*, \tilde{Y}) = \frac{1}{n} \max_{j=1,\dots,m} DSC(S_i^*, \widetilde{S_j}).$$

After symmetrizing, we obtain the **Symmetric Best Dice** score:

$$SBD(Y^*, \tilde{Y}) = \min\{BD(Y^*, \tilde{Y}), BD(\tilde{Y}, Y^*)\}.$$