Applying Deep Neural Network Techniques for LArTPC Data Reconstruction

Laura Domine (Stanford University / SLAC)

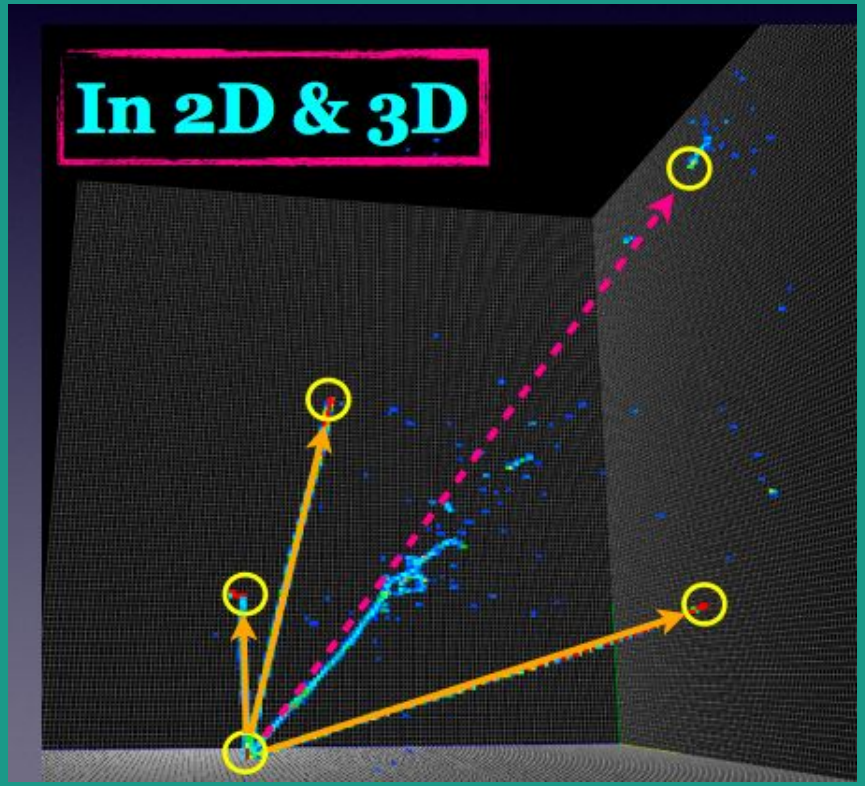
Fermilab Machine Learning Group meeting - 11/7/18

SLAC

Plan

1. LArTPC & Deep Learning
2. Examples of applications: UResNet & PPN networks
3. Sparse convolutions

LArTPC
& Deep Learning



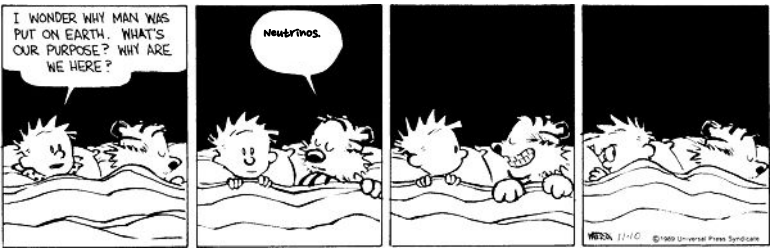
Liquid Argon Time Projection Chamber (LArTPC)

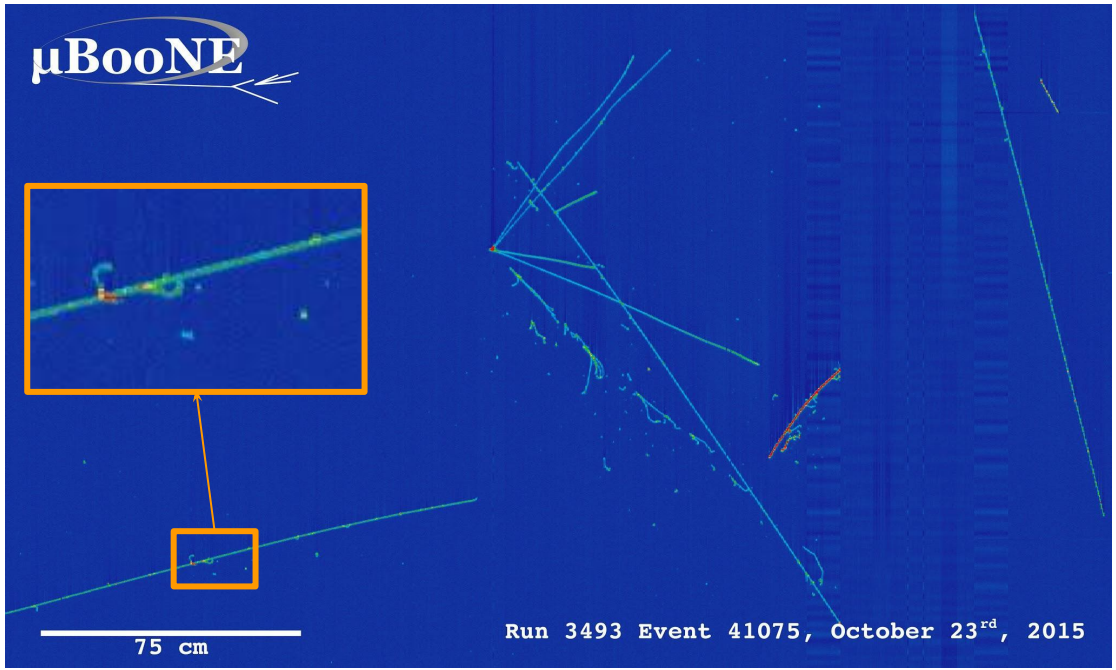
Neutrino detectors

Ex: MicroBooNE @ Fermilab, 150 tons

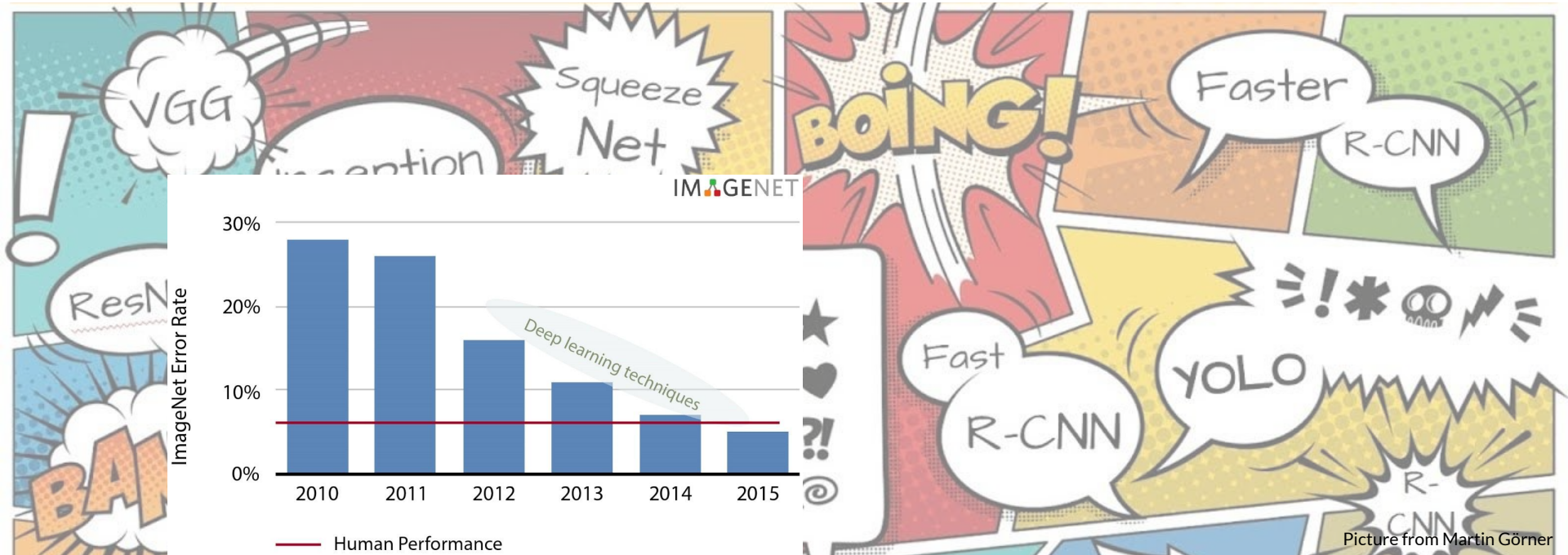
2D or 3D data

Bigger and bigger! (DUNE)





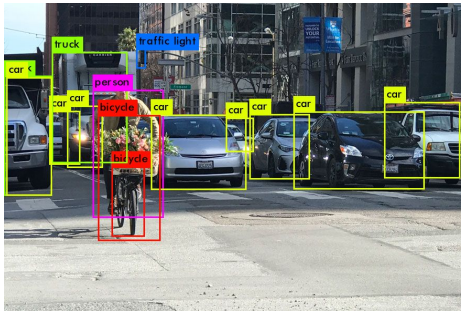
Deep Neural Networks (DNN) & Computer Vision



Picture from Martin Görner

Deep Neural Networks (DNN) & Computer Vision

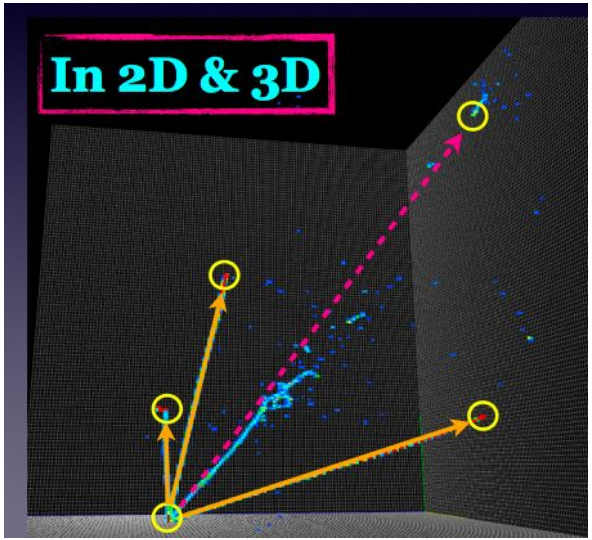
Object detection & classification



Semantic segmentation

Towards a full reconstruction chain with DNN

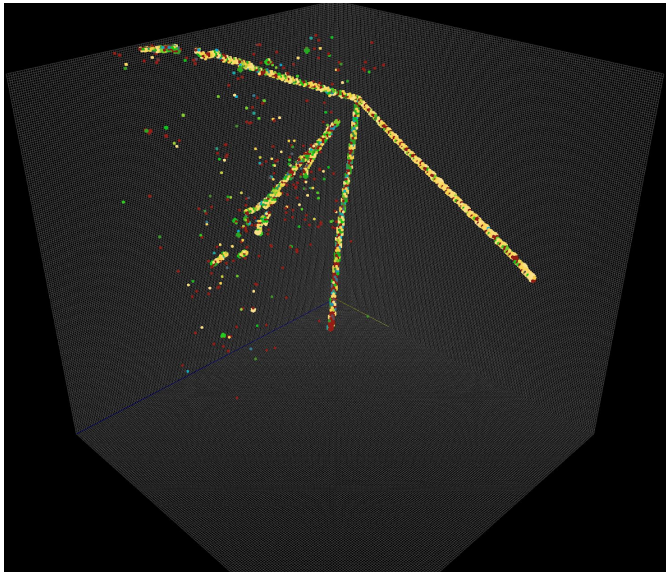
- Currently: Lots of heuristic algorithms
- Goal: Replace them with a set of DNN algorithms which ideally will
  - Run faster
  - Have a better performance



Towards a full reconstruction chain with DNN

Steps:

1. **Point detection (track edge)**

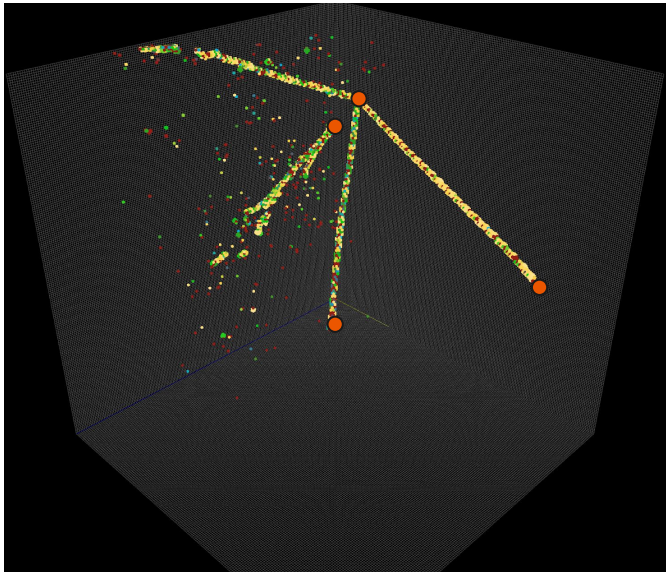


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

1. ~~Point detection (track edge)~~
   PPN

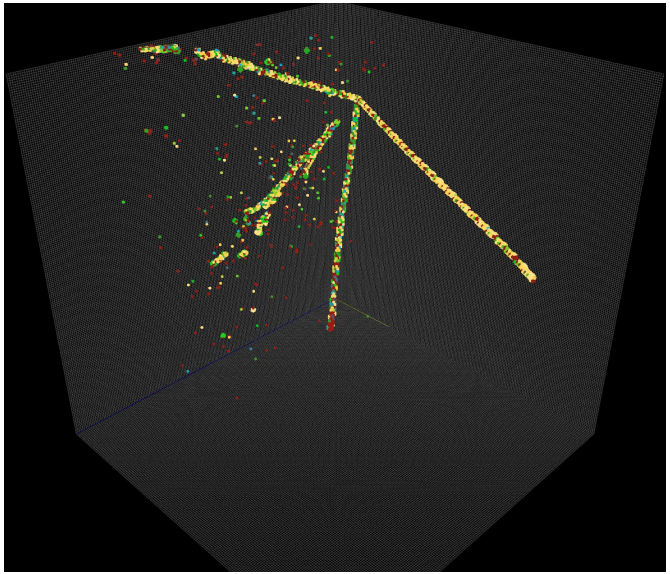


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

1. ~~Point detection (track edge)~~
   PPN
2. **Pixel-wise labeling (particle track vs electromagnetic shower)**

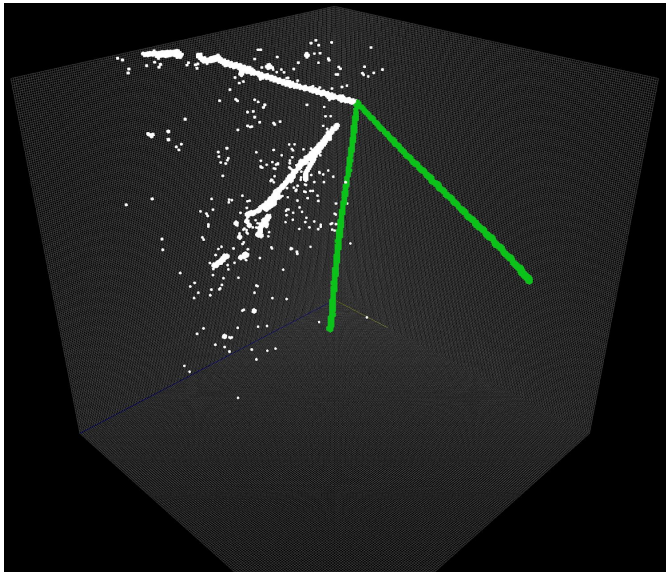


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

1. ~~Point detection (track edge)~~
   PPN
2. ~~Pixel-wise labeling (particle track vs electromagnetic shower)~~
   UResNet

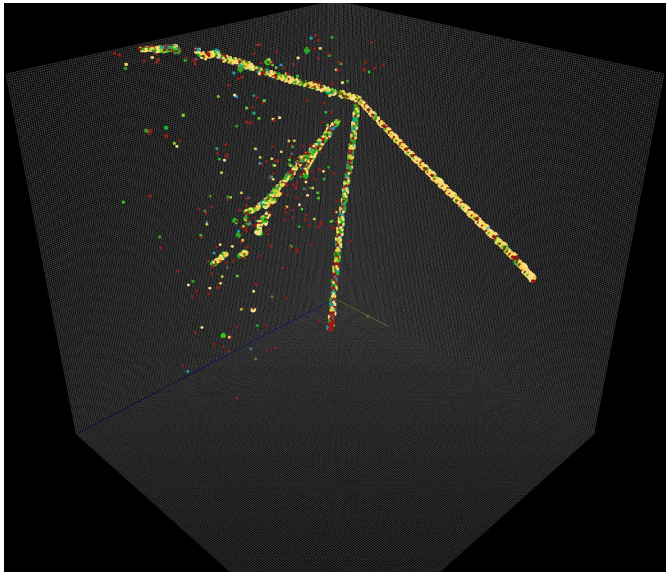


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

1. ~~Point detection (track edge)~~
   PPN
2. ~~Pixel-wise labeling (particle track vs electromagnetic shower)~~
   UResNet
3. **Clustering of energy deposits and instance segmentation**

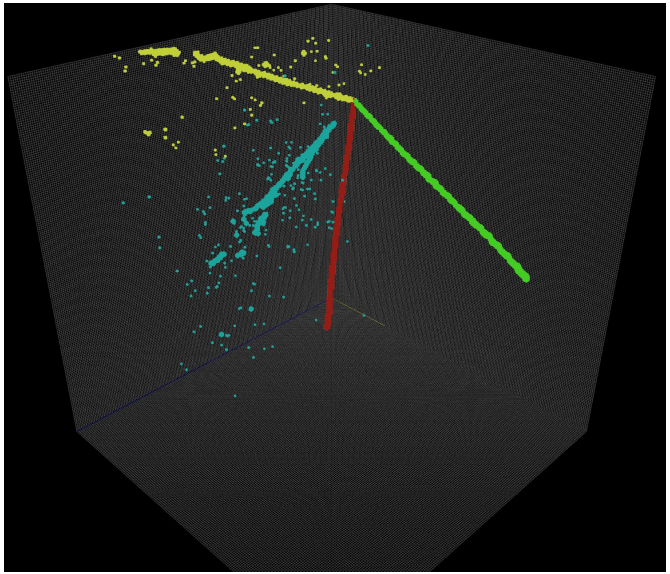


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

1. ~~Point detection (track edge)~~
   PPN
2. ~~Pixel-wise labeling (particle track vs electromagnetic shower)~~
   UResNet
3. **Clustering of energy deposits and instance segmentation**
   Work in progress!

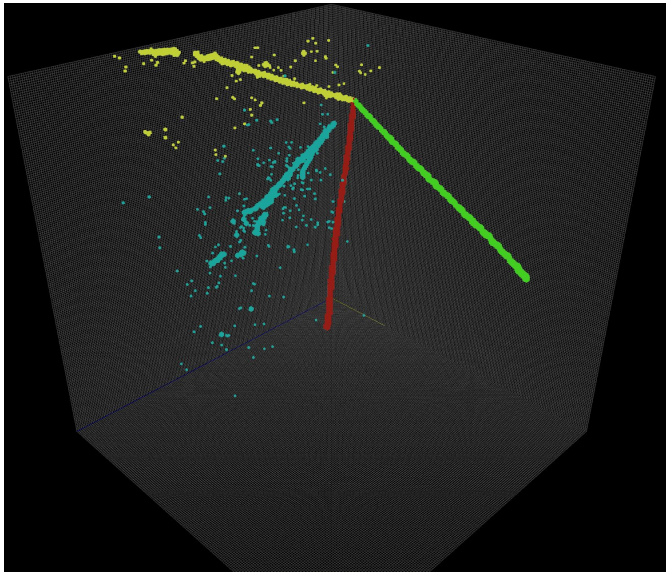


Non-contractual picture - Actual product may differ

Towards a full reconstruction chain with DNN

Steps:

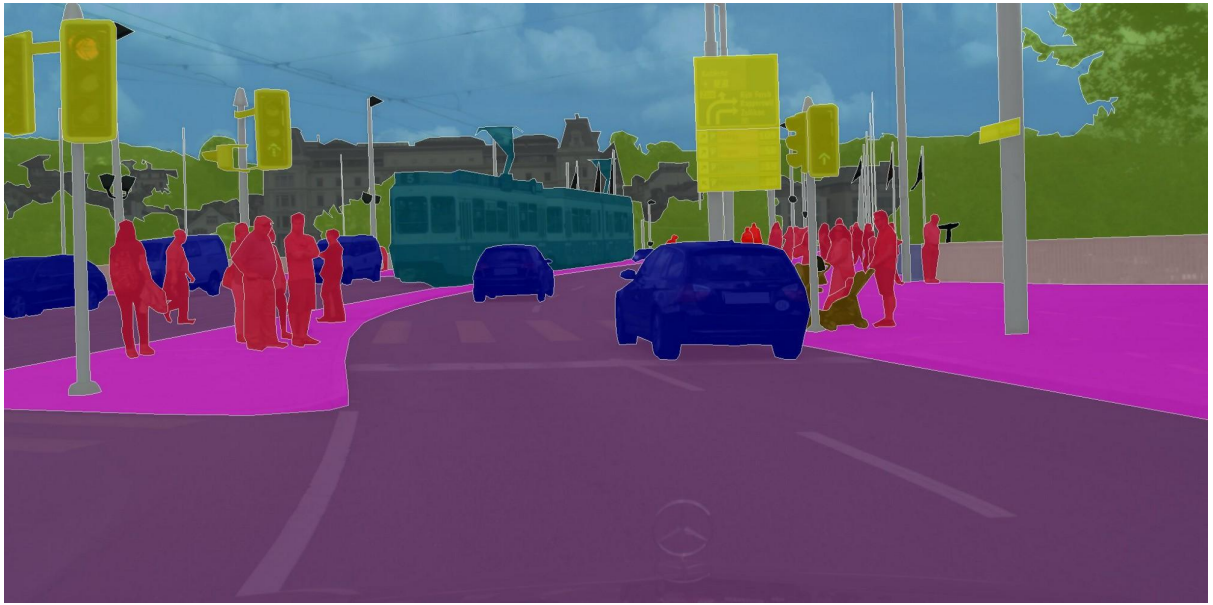
1. ~~Point detection (track edge)~~
   PPN
2. ~~Pixel-wise labeling (particle track vs electromagnetic shower)~~
   UResNet
3. Clustering of energy deposits and instance segmentation
   Work in progress!
4. ***Particle identification and energy estimate***
5. ***Hierarchical reconstruction***



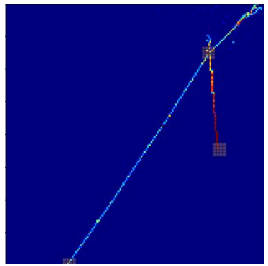
Non-contractual picture - Actual product may differ

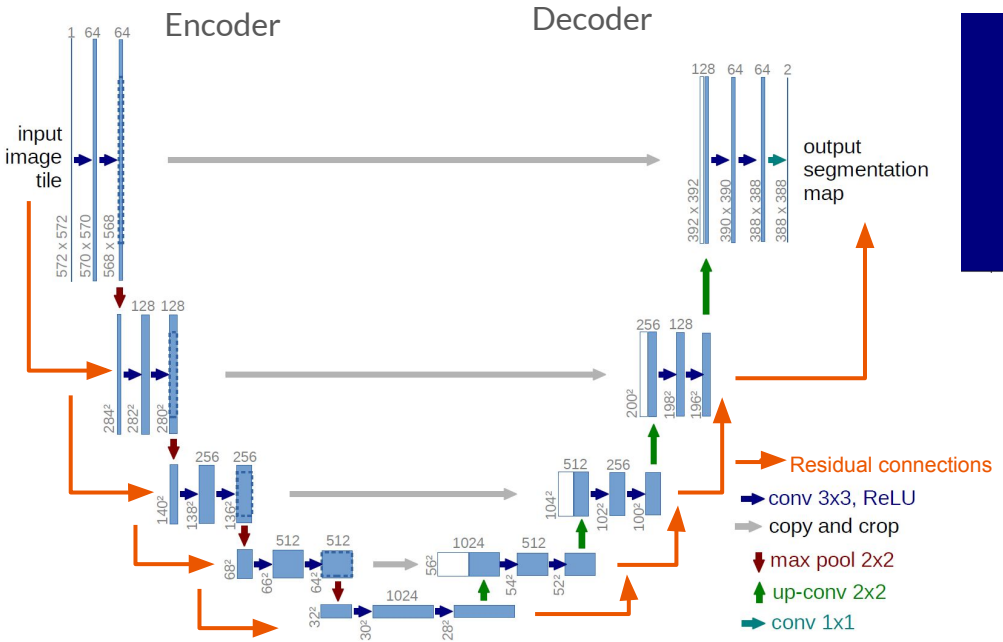
Examples of applications: UResNet and PPN networks

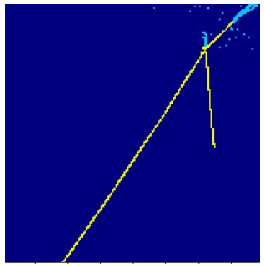
Semantic Segmentation: UResNet



Semantic Segmentation: UResNet

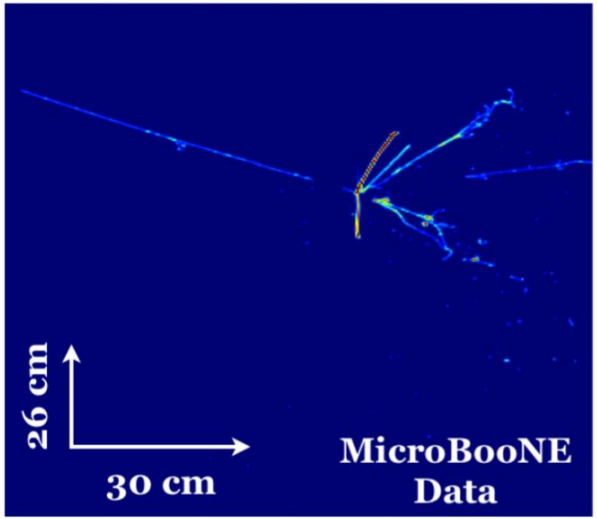




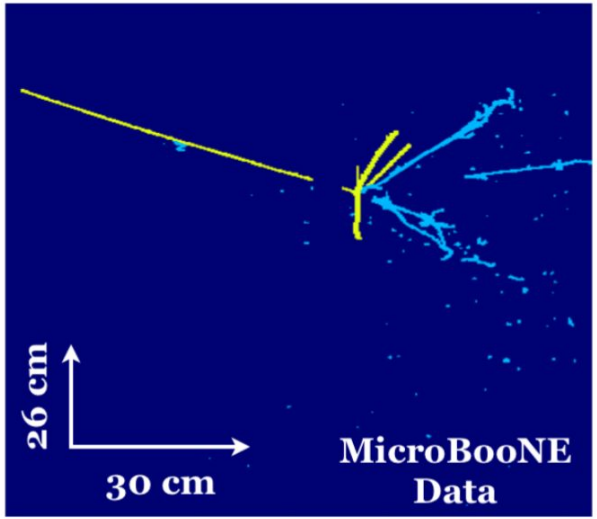


Semantic Segmentation: UResNet

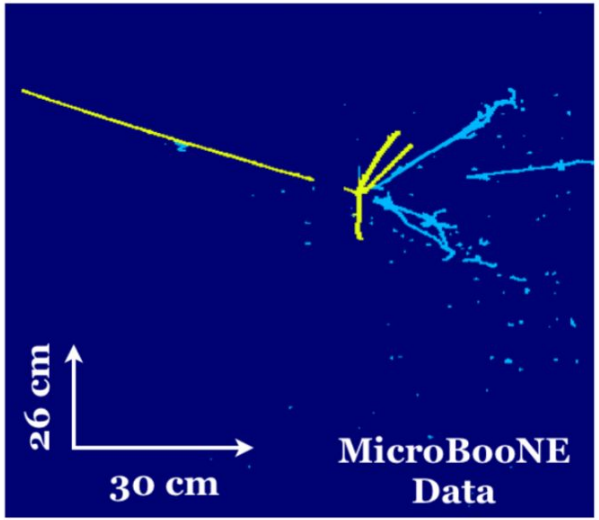
Data



Physicist's label



Network's output



A Deep Neural Network for Pixel-Level Electromagnetic Particle Identification in the MicroBooNE Liquid Argon Time Projection Chamber.
https://arxiv.org/abs/1808.07269

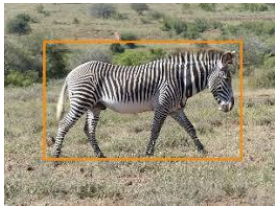
Point-finding: PPN

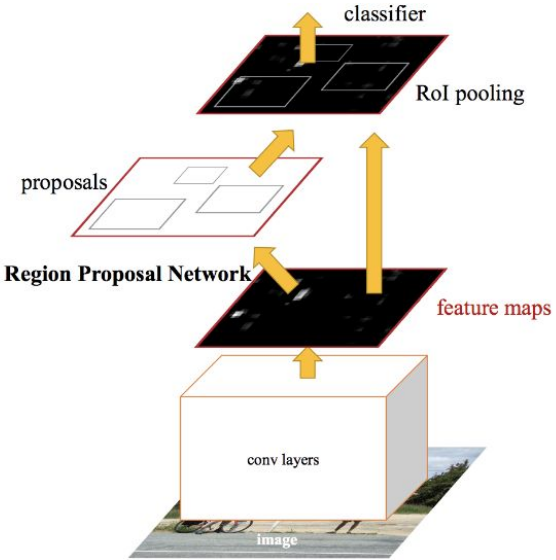
Inspired by **Faster-RCNN** architecture

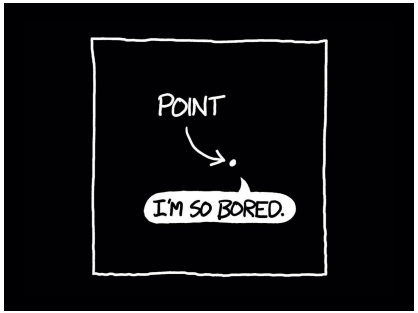
- Region Proposal Network detects regions of interest
- Replace regions with points = **Pixel Proposal Network** (PPN)

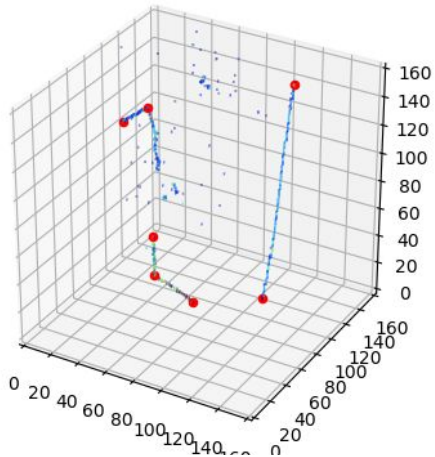
  Why not Mask-RCNN?

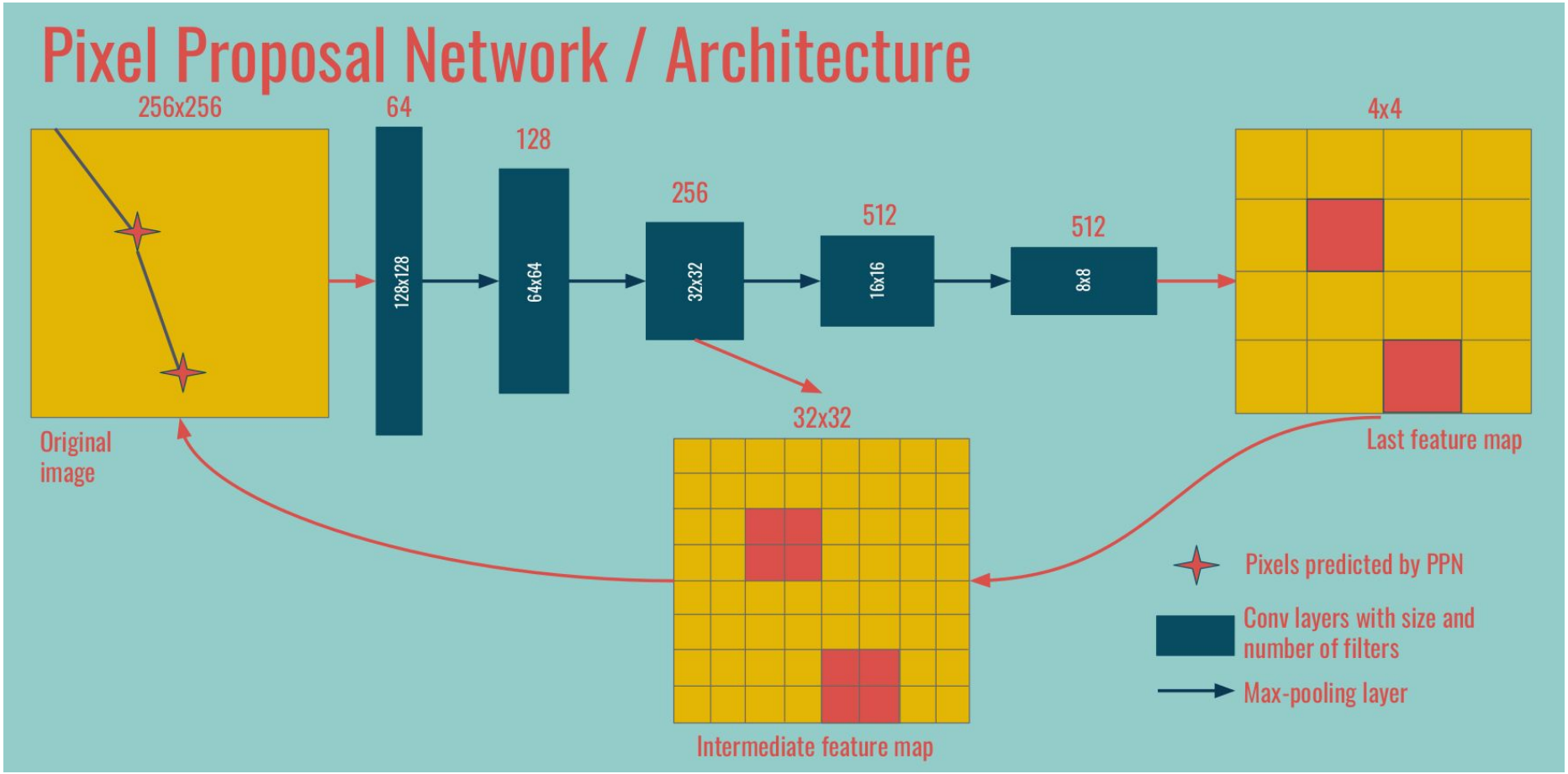
  - Computations expensive
  - Our features topology is different (track, shower)



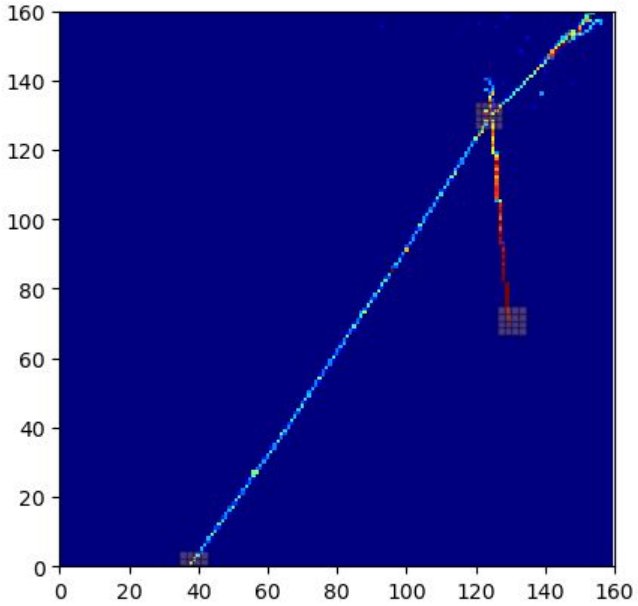


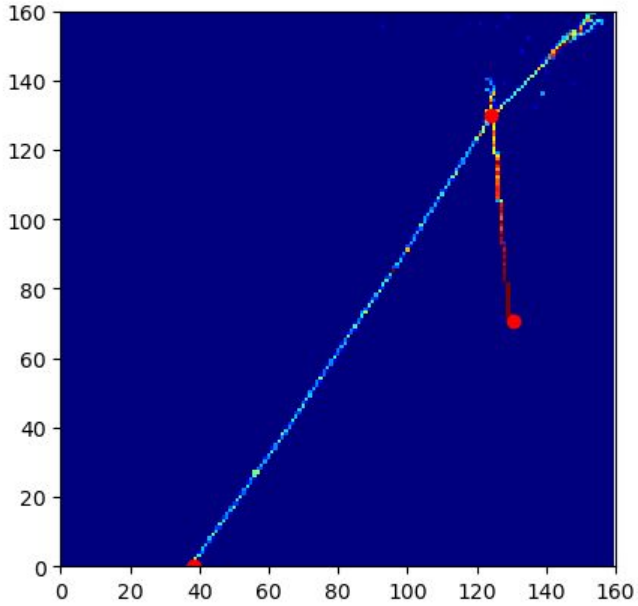




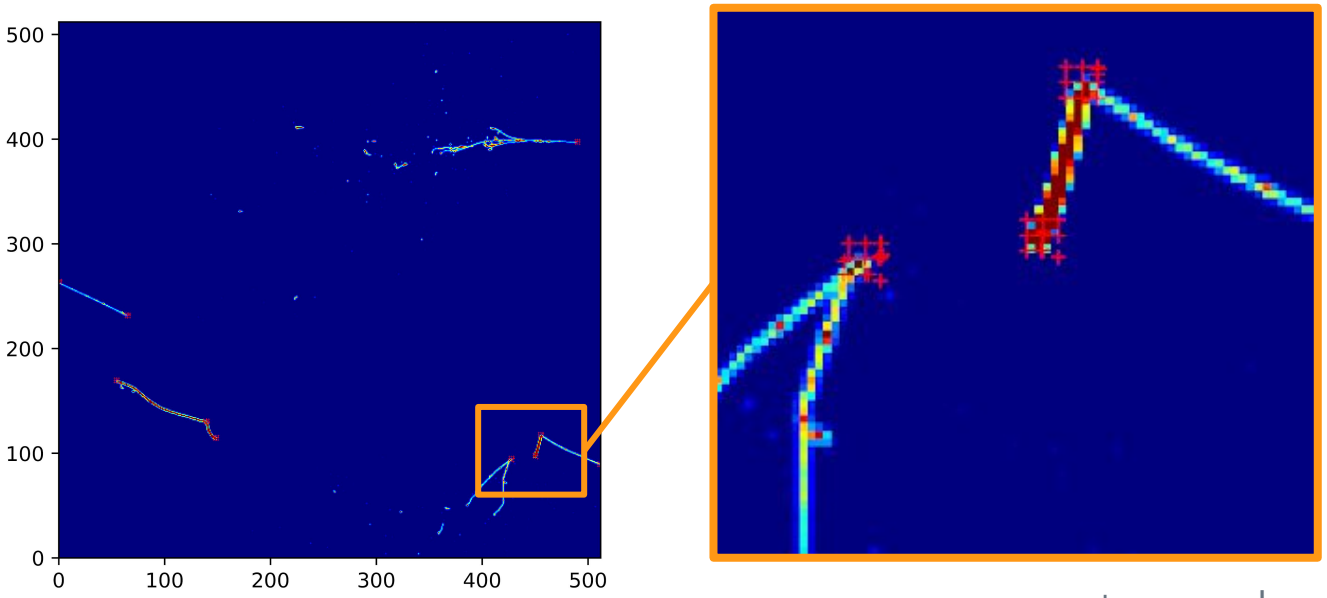

Pixel Proposal Network / Architecture
256x256
64
128
256
512
512
4x4
128x128
64x64
32x32
16x16
8x8
Original image
32x32
Last feature map
Intermediate feature map
Pixels predicted by PPN
Conv layers with size and number of filters
Max-pooling layer

PPN proposals





PPN needs post-processing

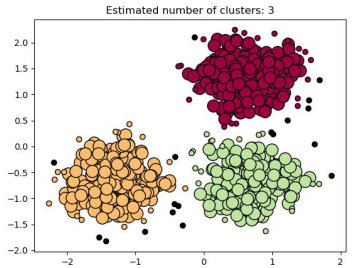


+ scores...!

PPN needs post-processing

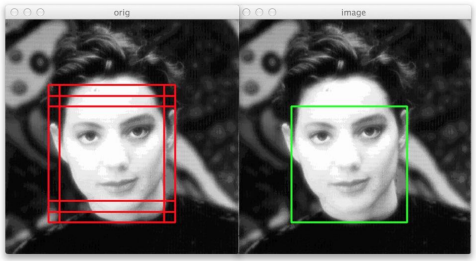
Option 1: DBSCAN

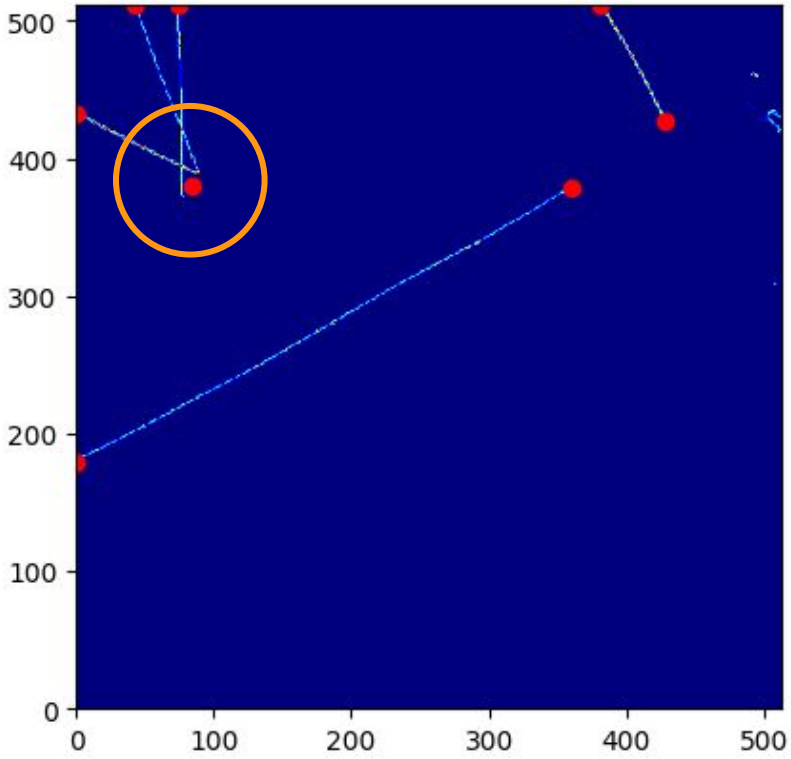
- Density estimation algorithm
- No prior on the number of clusters.

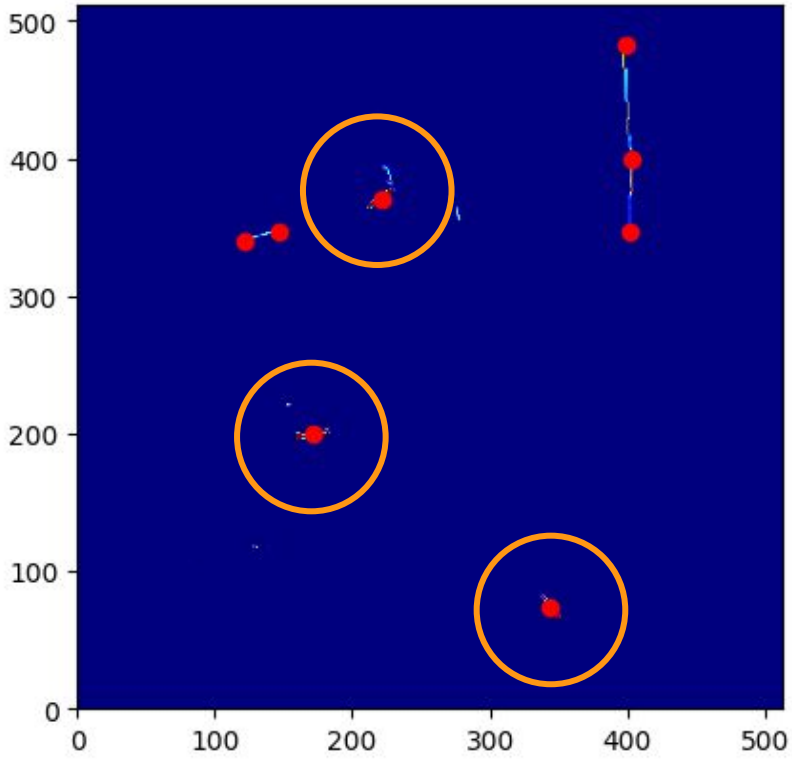


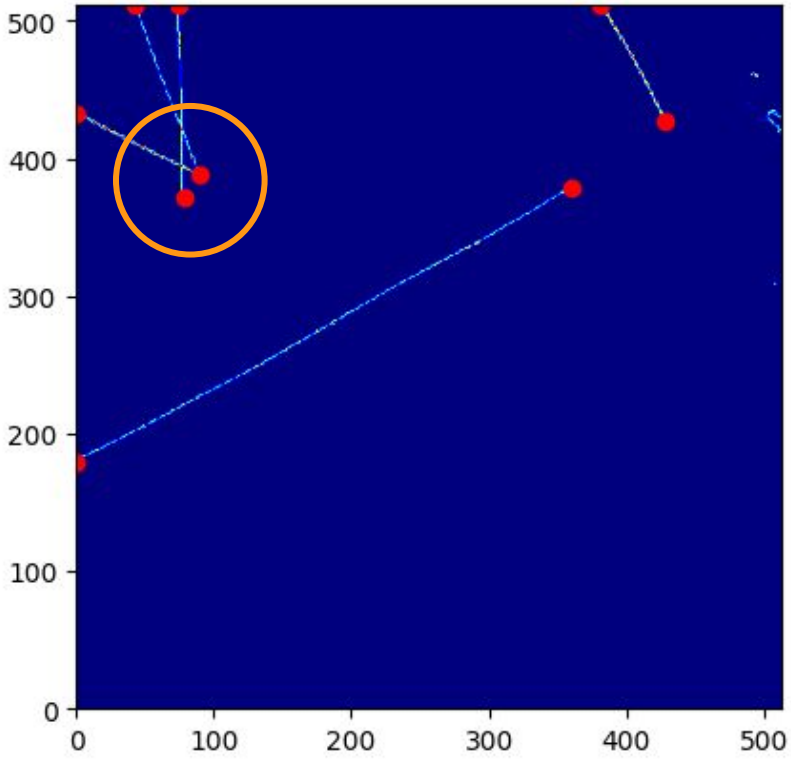
Option 2: NMS (Non-Maximal Suppression)

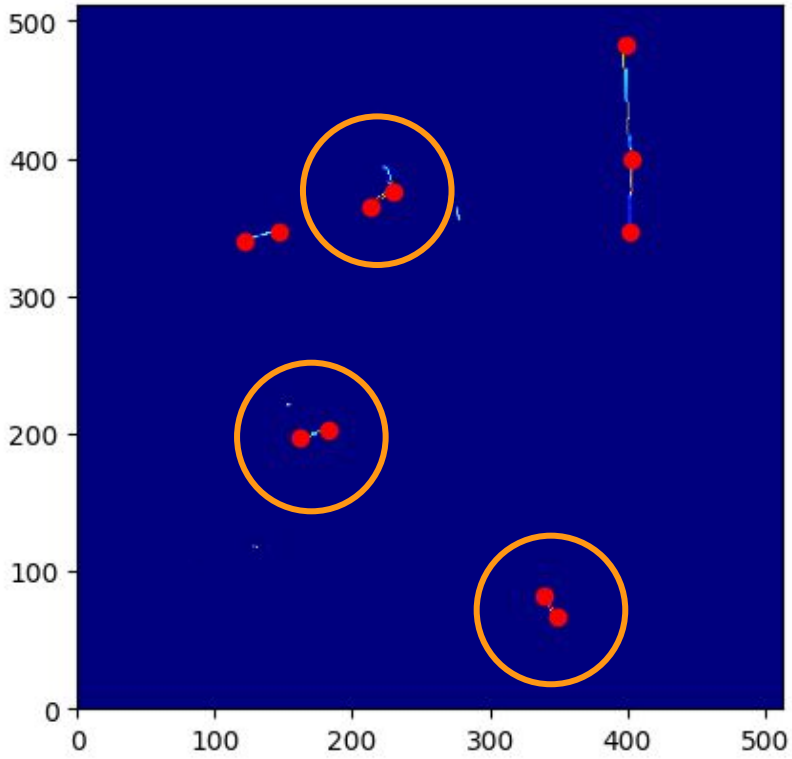
- Popular post-processing method for object detection
- Order by score and prune boxes with too much overlap

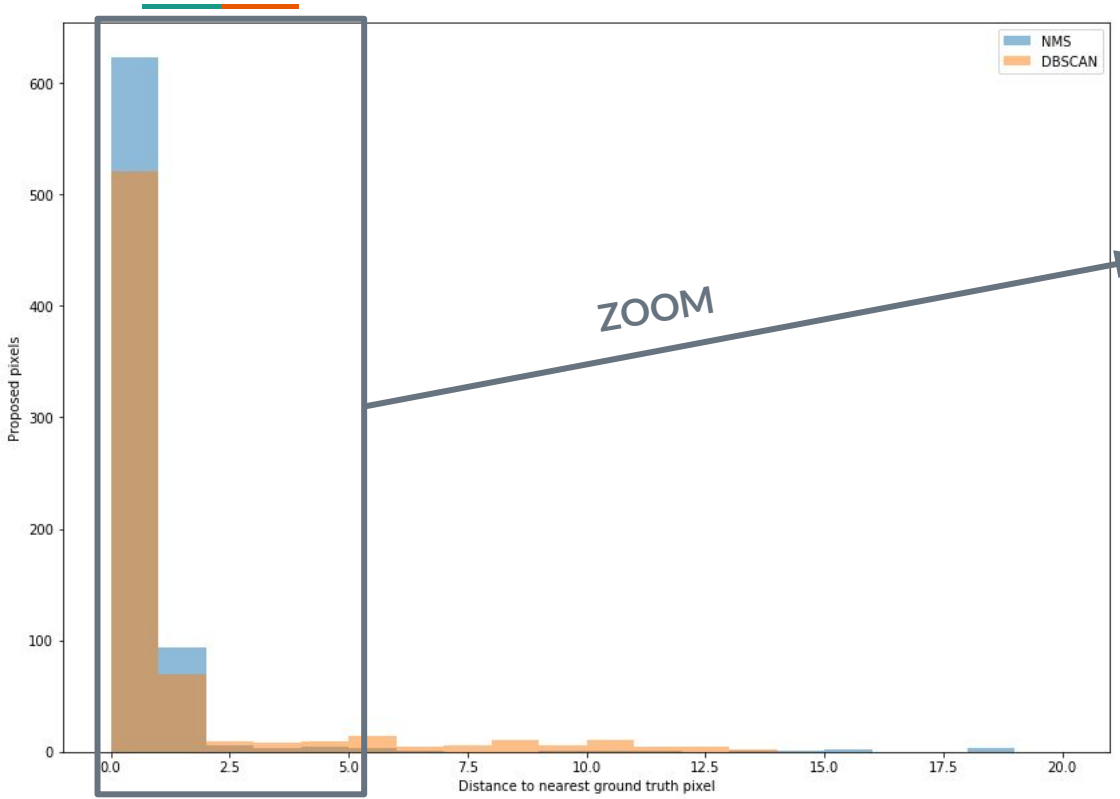


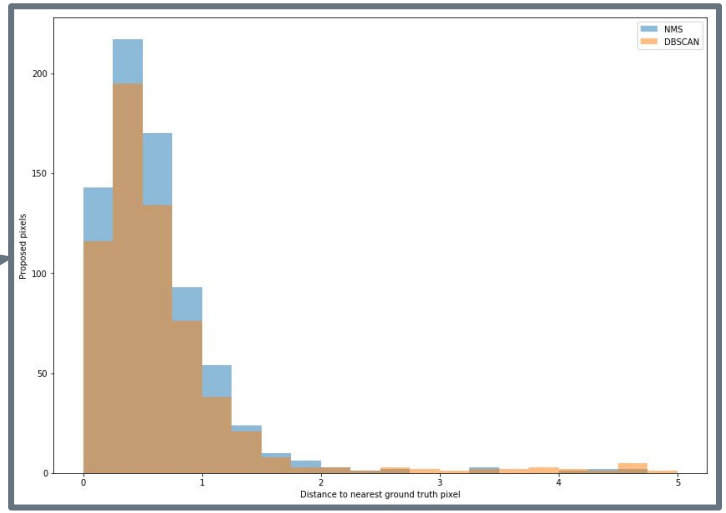

500
400
300
200
100
0
0
100
200
300
400
500


500
400
300
200
100
0
0
100
200
300
400
500


500
400
300
200
100
0
0
100
200
300
400
500

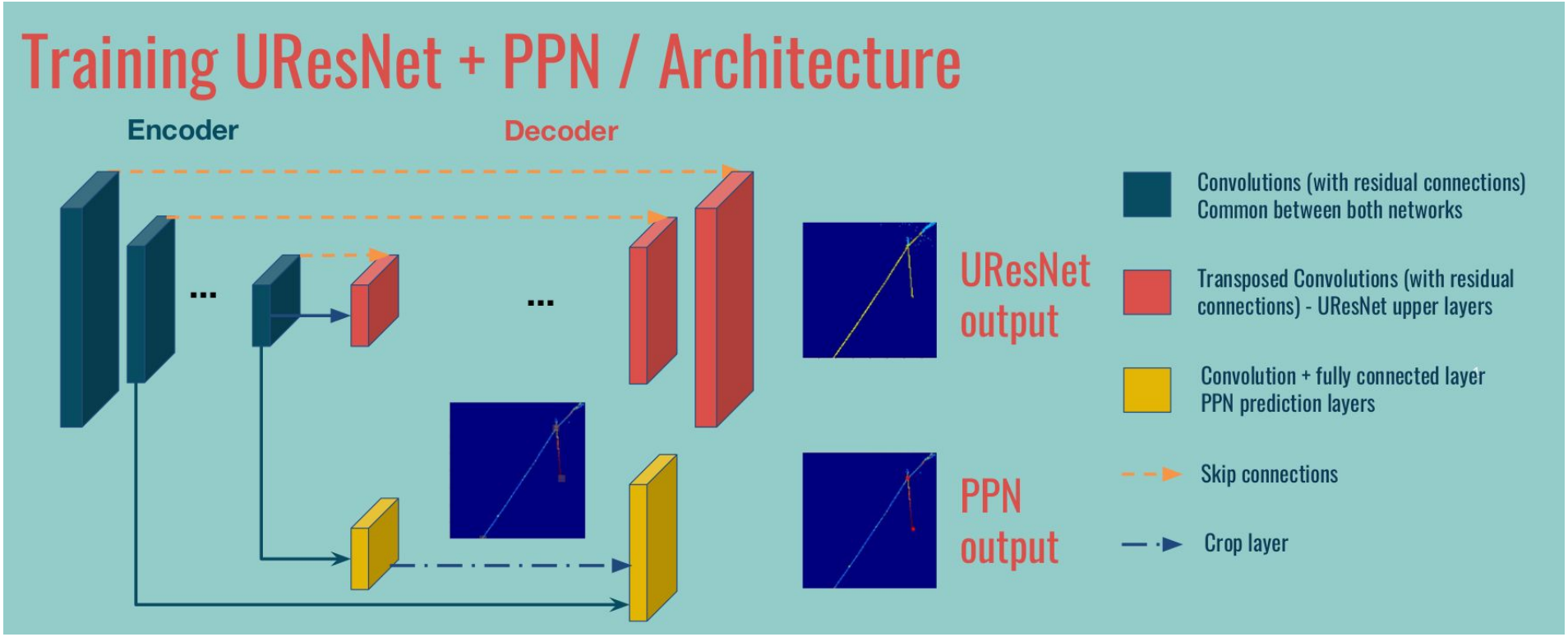

500
400
300
200
100
0
0
100
200
300
400
500



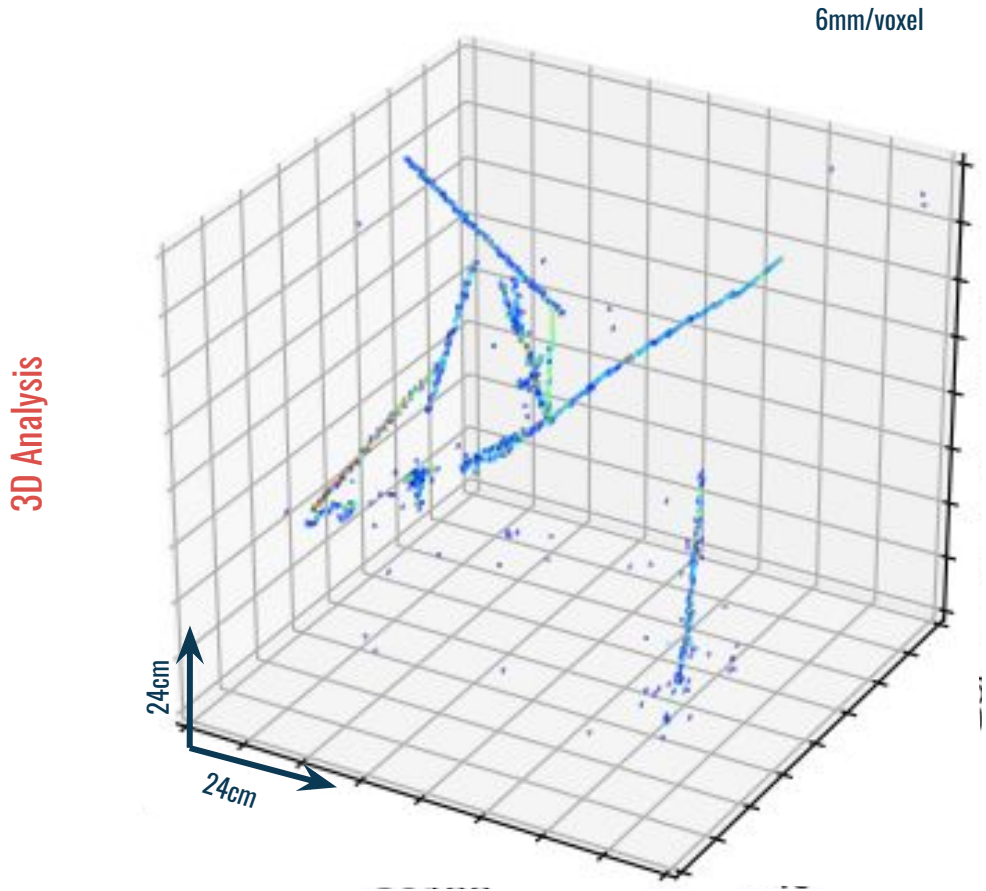


NB: independently of DBSCAN vs NMS, these plots also benefit from debugged ground truth pixels position.

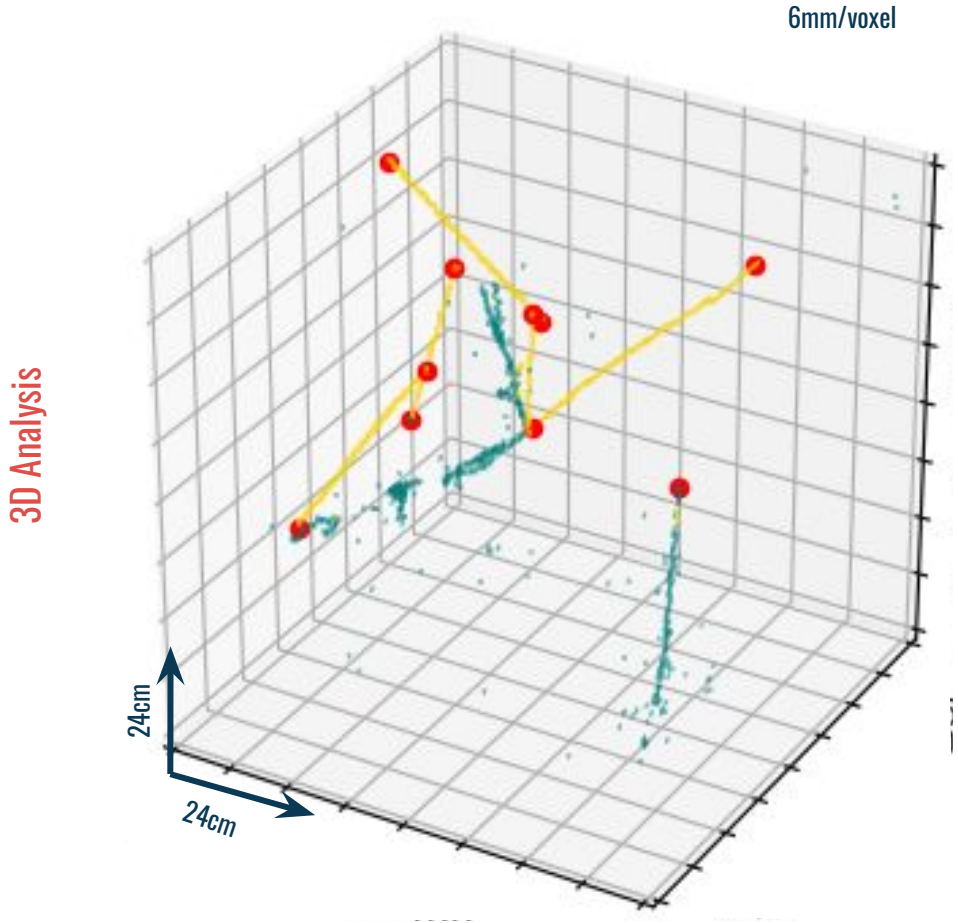
Training UResNet + PPN / Architecture



UResNet + PPN



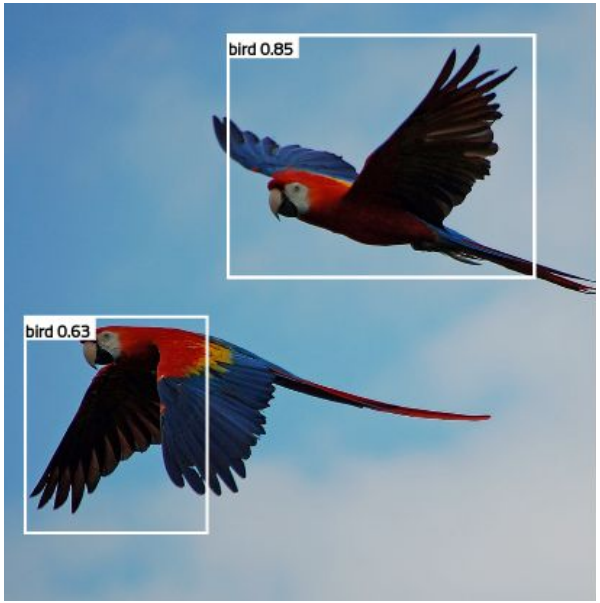
UResNet + PPN



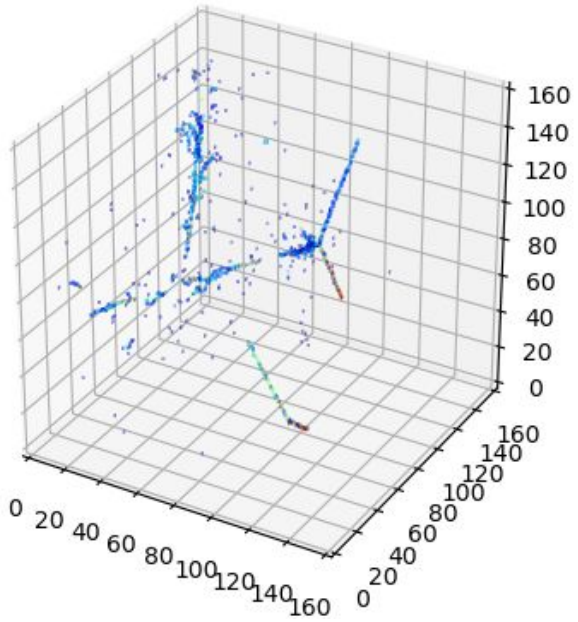
Sparse UResNet

How do we handle sparse data?

Dense



Sparse



Naive approach

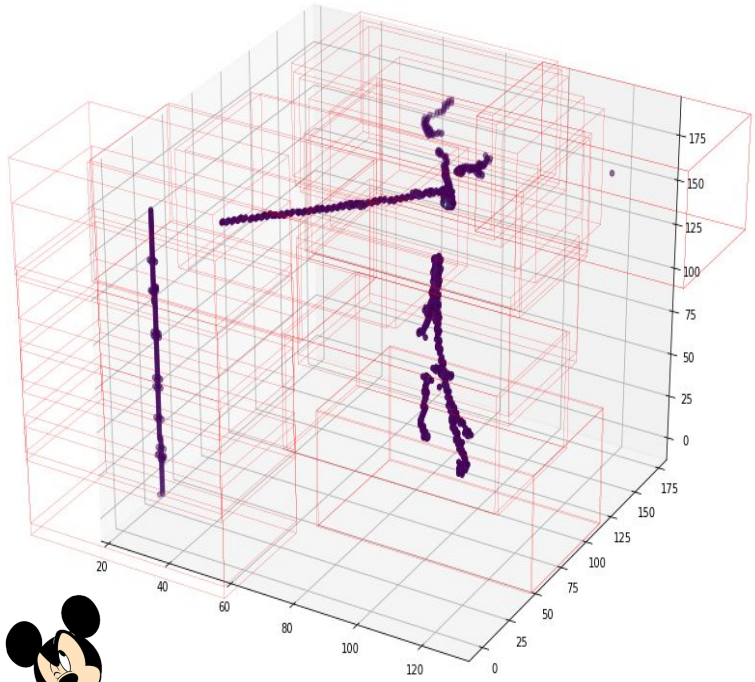
Input: dense 3D matrix of energy deposits.

- Crop your data
- Run the network on small cropped images
- Stitch together results

Many cropping algorithms possible

Compromises to make:

- Maximize the number of overlapping boxes (accuracy)
- Minimize the number of boxes (computation time)



Sparse Convolutions

Many possible definitions and implementations of ‘sparse convolutions’...

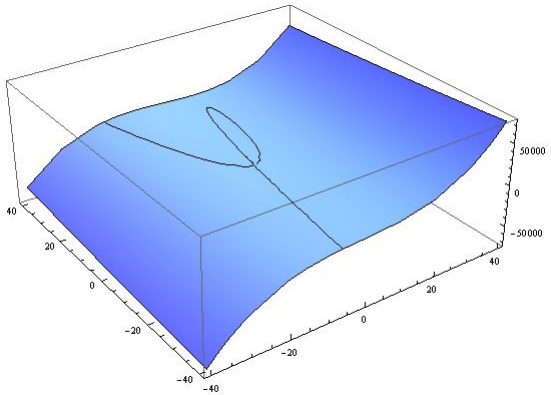
Submanifold Sparse Convolutions: https://github.com/facebookresearch/SparseConvNet

Submanifold?

“input data with lower effective dimension than the space in which it lives”

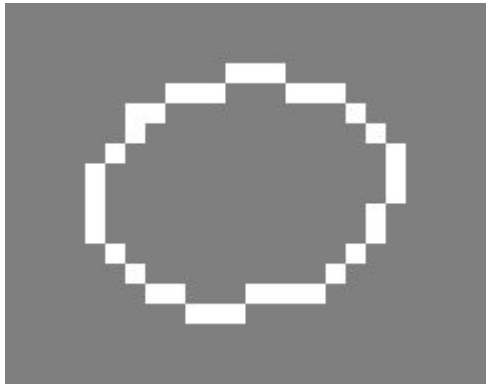
Ex: 1D curve in 2+D space, 2D surface in 3+D space

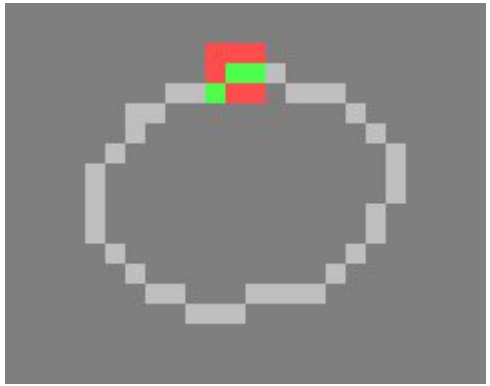
Our case: the worst! **1D curve in 3D space**...



Sparse Convolutions

Submanifold Sparse Convolutions: https://github.com/facebookresearch/SparseConvNet





3D Semantic Segmentation with Submanifold Sparse Convolutional Networks

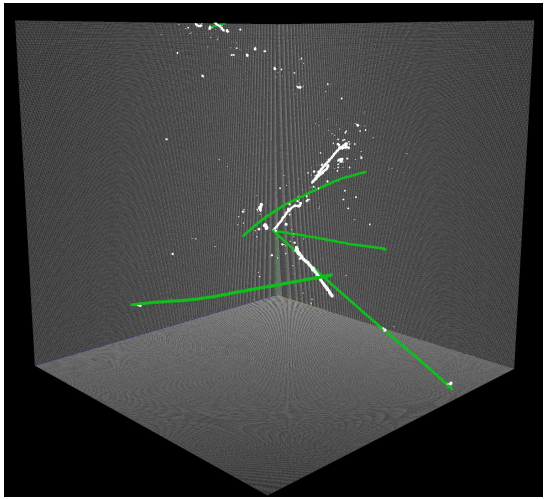
Sparse UResNet

Input: list of points coordinates and their features (e.g. energy deposition)

With UResNet architecture:

- >99.9% accuracy in 3D
- Faster training (less computations!): only a few hours
- Much lower memory usage

Example in larcv-viewer

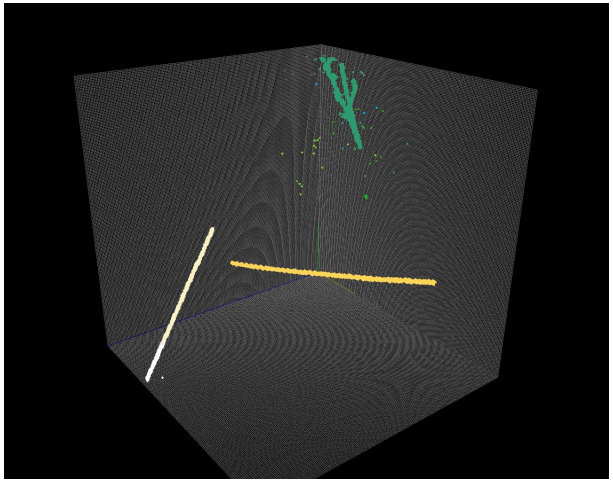


Summary

- Extract interesting / useful features with deep neural networks:
  - Points of interest with PPN
  - Pixel-wise classification track vs shower with UResNet
- Currently working on clustering and instance segmentation (particle type, particle instances)
- Sparse techniques are very exciting!

*Join **DeepLearnPhysics** group!*

- *Technical discussion on ML applied to experimental physics data*
- *Data + code sharing for reproducibility*



Thank you!

Backup slides

PPN Loss: details