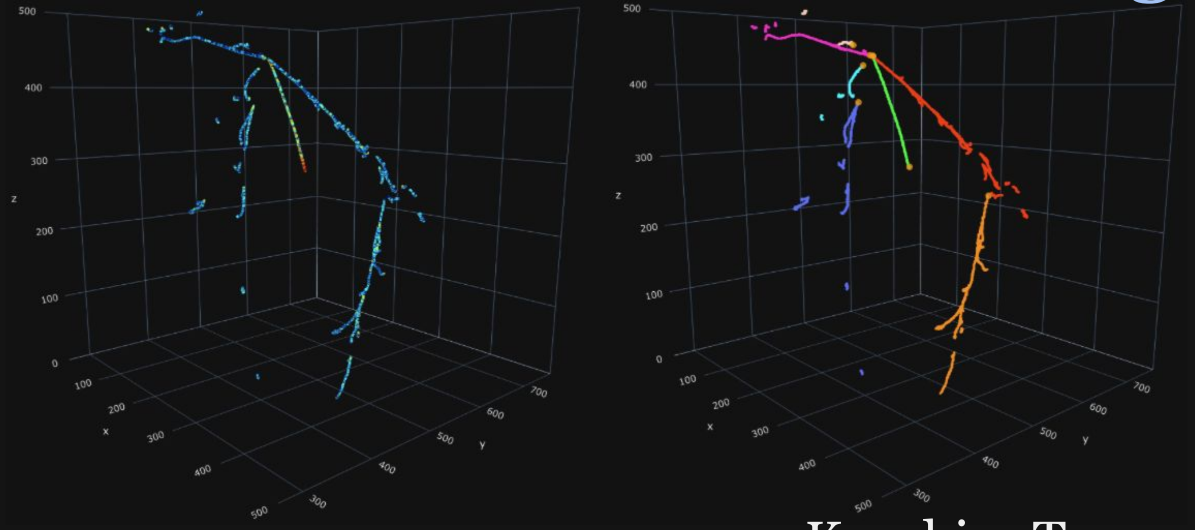
Data Reconstruction and Modeling of Particle Imaging Neutrino Detectors with Machine Learning

Original image credit: xkcd



Kazuhiro Terao
SLAC National Accelerator Laboratory

Jan. 13th 2023 @ IPMU Seminar Series

Happy New Year!

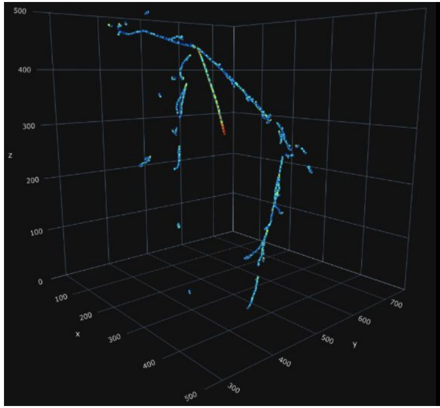
About myself:

- Experimental neutrino physicist
- AI/ML applications and research
    - Physics research and work unrelated (hobby!)
    - Love to discuss / learn about challenges outside my domain
    - Love to help a workshop, School (education), collaboration

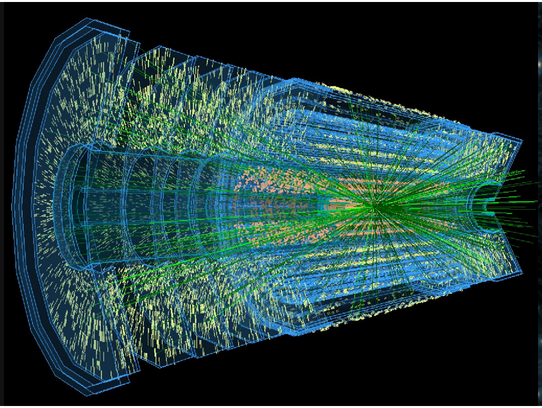
Interest: AI/ML for HEP and Beyond

ML is a "solution pattern" v.s. a domain-specific "hard-coded" solution.
It's **naturally reusable across domains including software tools**
supported by a large community of researchers.

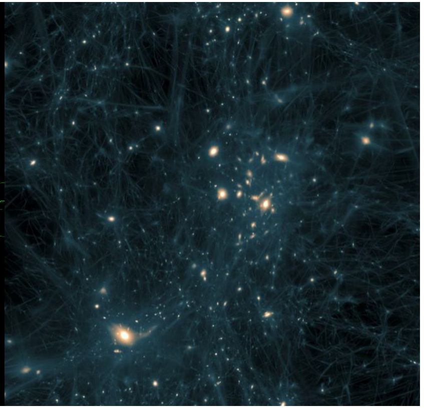
e.g.) physics inference on data from imaging detectors



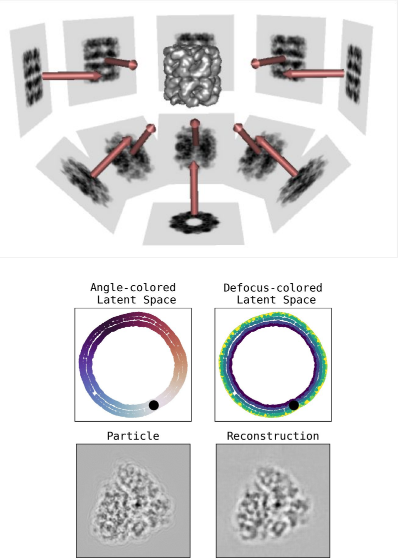
Intensity Frontier



Energy Frontier



Cosmic Frontier

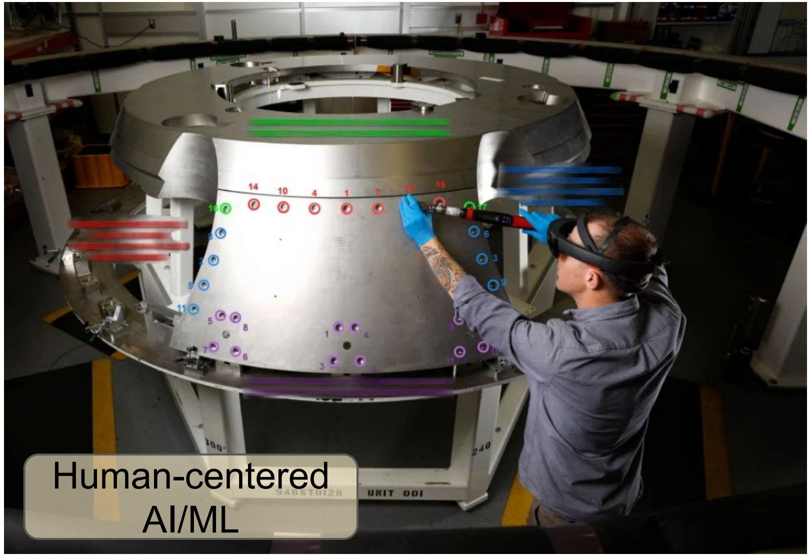


e.g.) Cryo-EM

Interest: AI/ML for HEP and Beyond

ML is a "solution pattern" v.s. a domain-specific "hard-coded" solution.
It's **naturally reusable across domains including software tools**
supported by a large community of researchers.

Even for hands-on work!

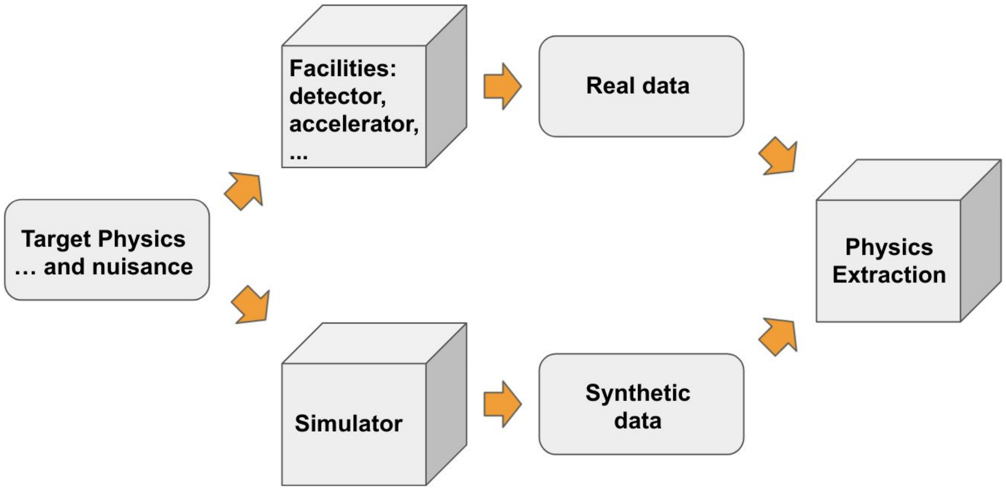


Interest: AI/ML for HEP and Beyond

ML is a “solution pattern” v.s. a domain-specific “hard-coded” solution. It’s **naturally reusable across domains including software tools** supported by a large community of researchers.

- Control/Operations Optimization
- Inverse Imaging
- Discriminative Physics Inference
- Fast/Differentiable Surrogates
- Uncertainty Quantification
- Design Optimization

ML applicable at all stages of an experiment

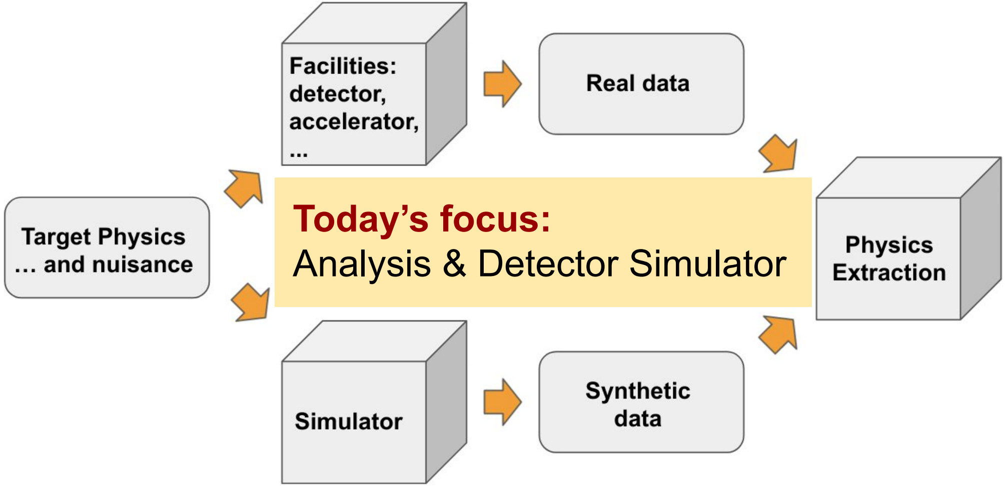


Interest: AI/ML for HEP and Beyond

ML is a "solution pattern" v.s. a domain-specific "hard-coded" solution.
It's **naturally reusable across domains including software tools** supported by a large community of researchers.

- Control/Operations Optimization
- Inverse Imaging
- Discriminative Physics Inference
- Fast/Differentiable Surrogates
- Uncertainty Quantification
- Design Optimization

ML applicable at all stages of an experiment

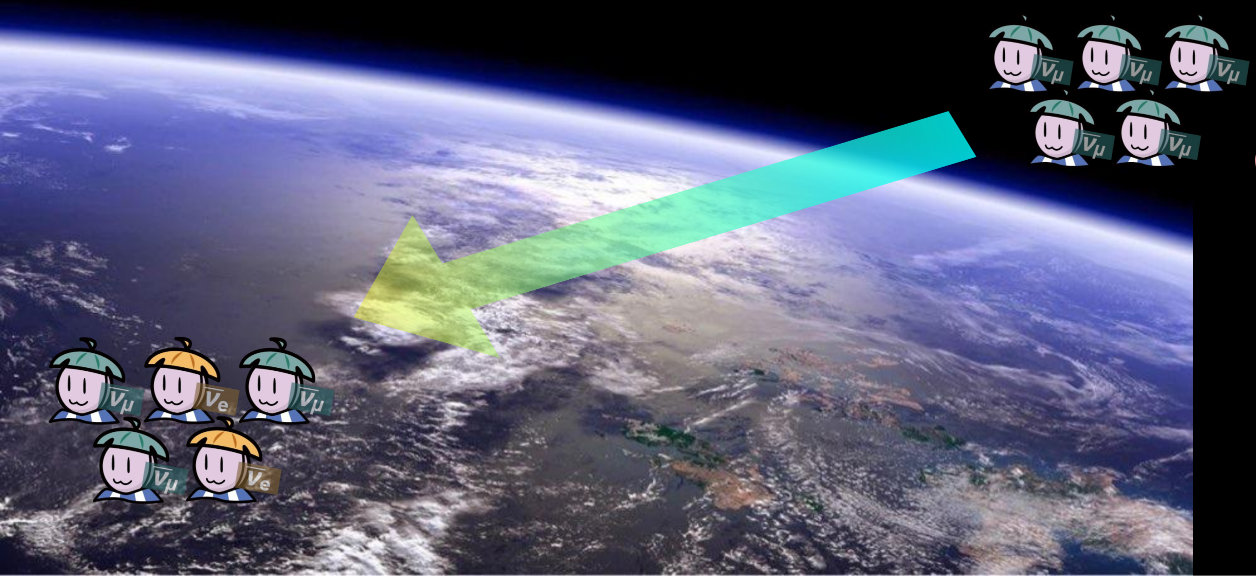


Oscillation Experiments and Big Imaging Detectors

ML for Analyzing Big Image Data in Neutrino Experiments

Neutrinos

Studying Neutrinos



change features

as they travel

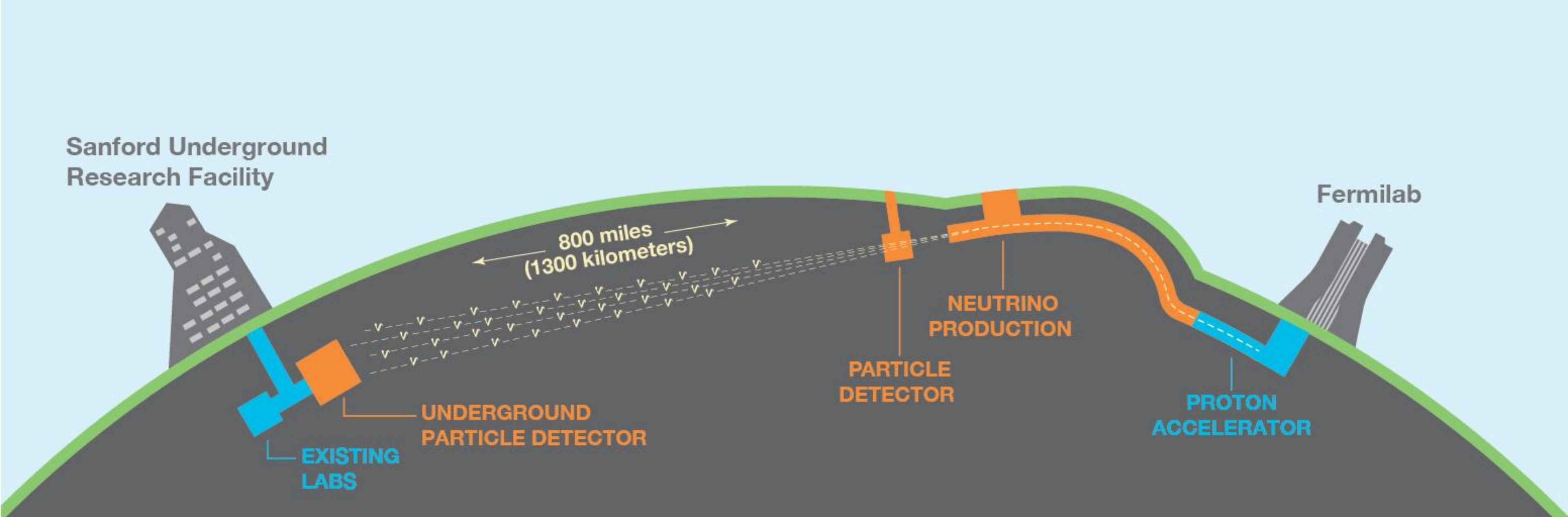
(nu oscillation)

ML for Analyzing Big Image Data in Neutrino Experiments

Neutrino Oscillation Experiments

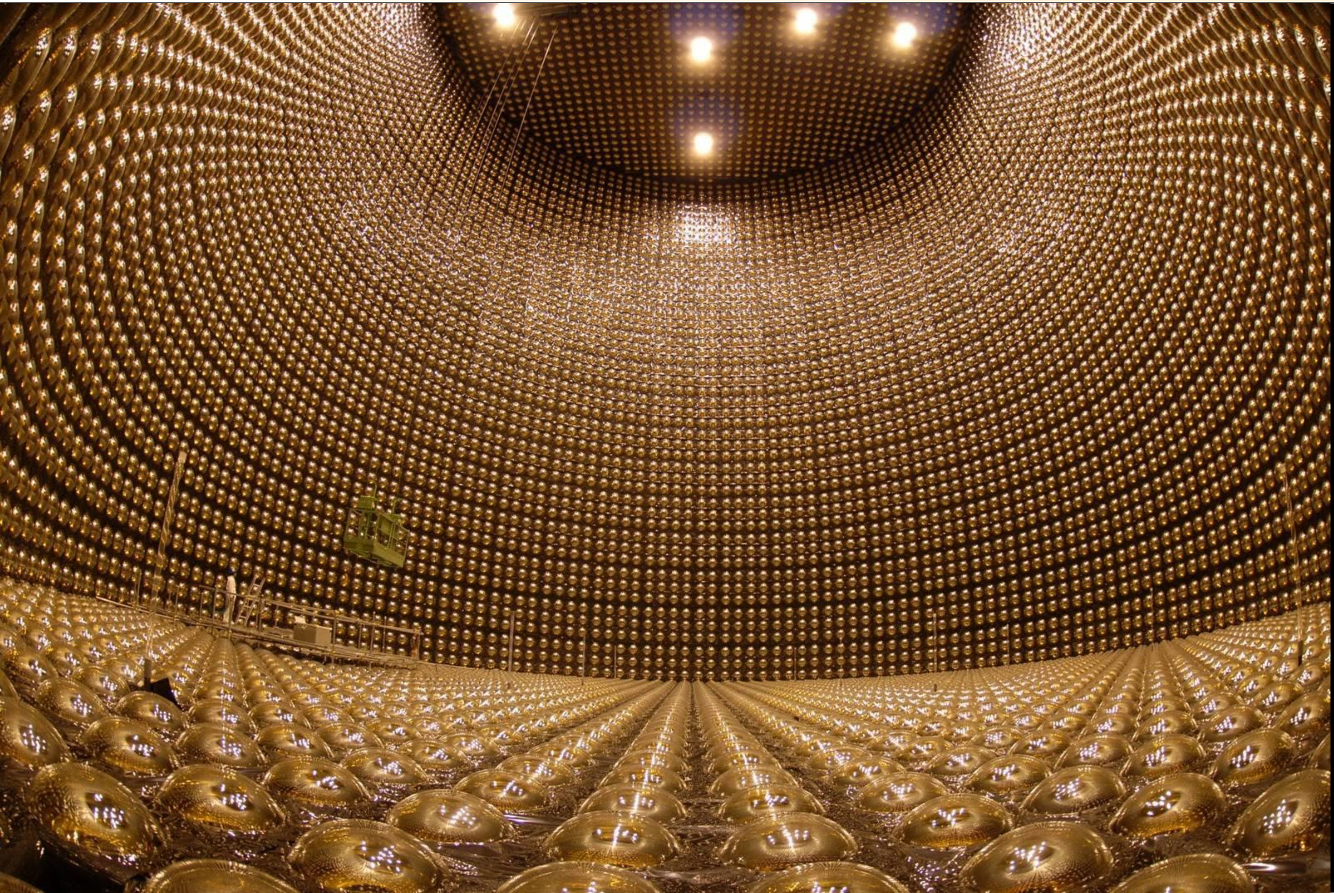
SLAC

Goal: shoot a beam of particles (neutrinos), detect at two locations, quantify the difference.



ML for Analyzing Big Image Data in Neutrino Experiments

Neutrino Oscillation Experiments

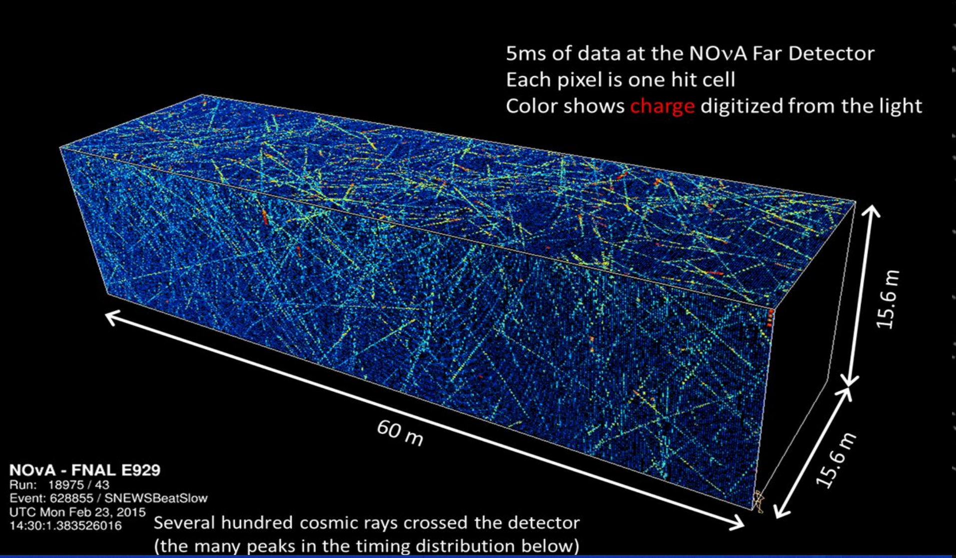


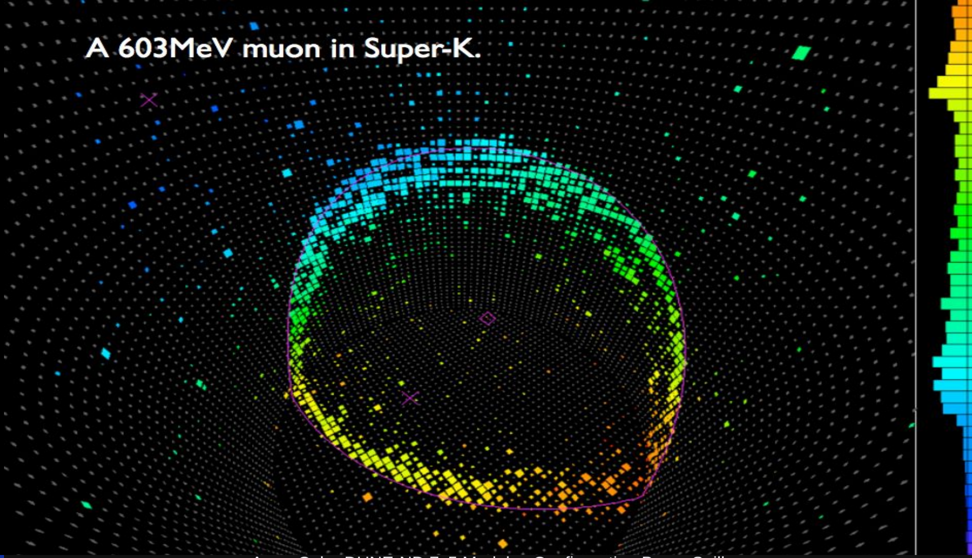
Detectors

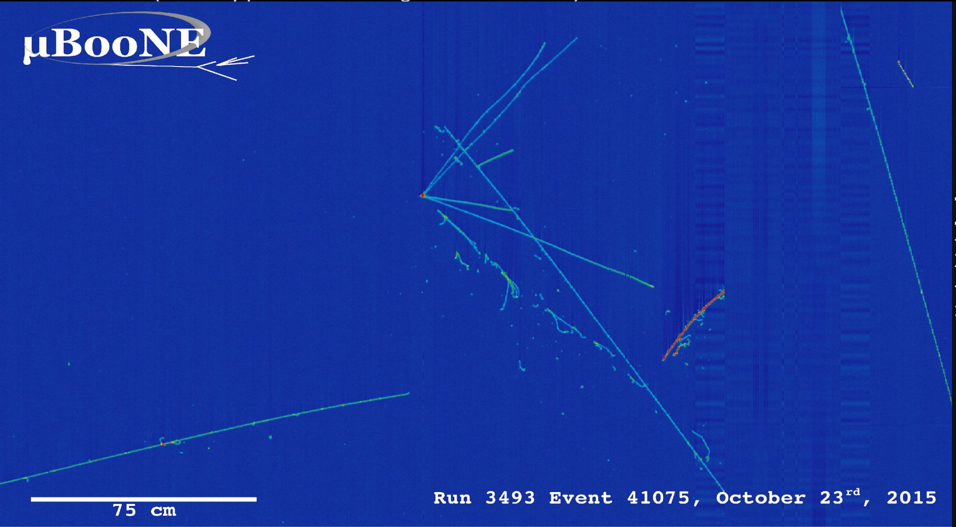
must be **BIG**

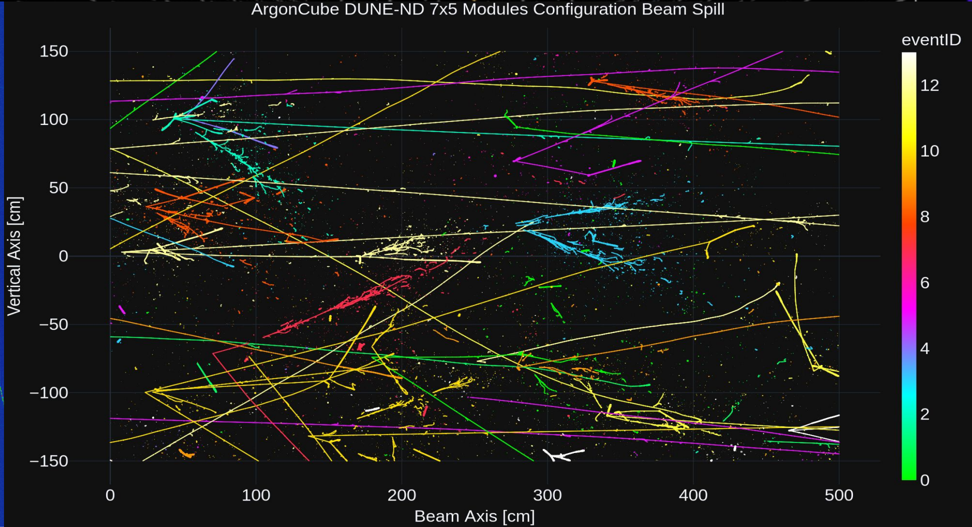
50,000 ton

ultra-pure water watched
by 11,000 PMTs in
Super-Kamiokande (1996)


5ms of data at the NOvA Far Detector
Each pixel is one hit cell
Color shows charge digitized from the light
60 m
15.6 m
15.6 m
NOvA - FNAL E929
Run: 18975 / 43
Event: 628855 / SNEWSBeatSlow
UTC Mon Feb 23, 2015
14:30:1.383526016
Several hundred cosmic rays crossed the detector
(the many peaks in the timing distribution below)

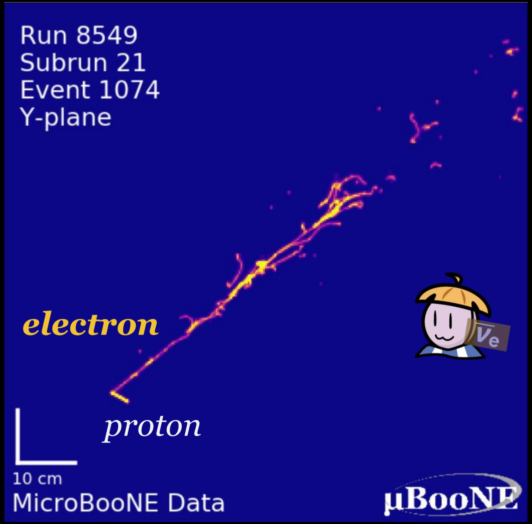

A 603MeV muon in Super-K.

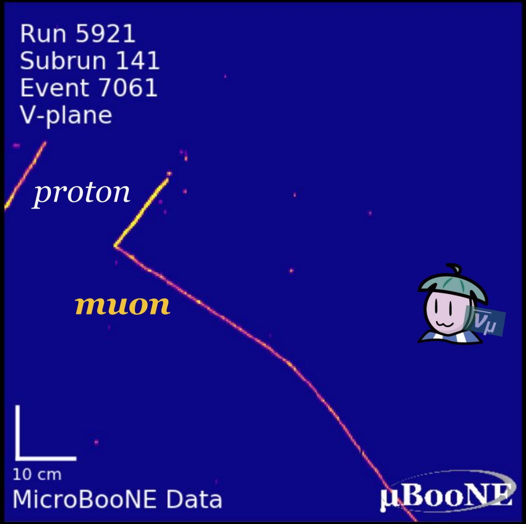

μBooNE
75 cm
Run 3493 Event 41075, October 23rd, 2015


ArgonCube DUNE-ND 7x5 Modules Configuration Beam Spill
Vertical Axis [cm]
Beam Axis [cm]
eventID
150
100
50
0
−50
−100
−150
0
100
200
300
400
500
12
10
8
6
4
2
0

ML for Analyzing Big Image Data in Neutrino Experiments

Neutrino Oscillation Experiments





ν_e creates electron (e)

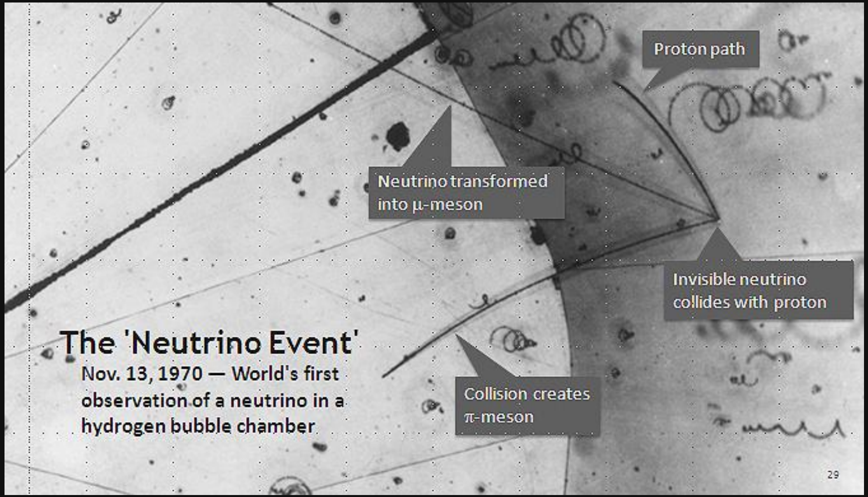
ν_μ creates muon (μ)

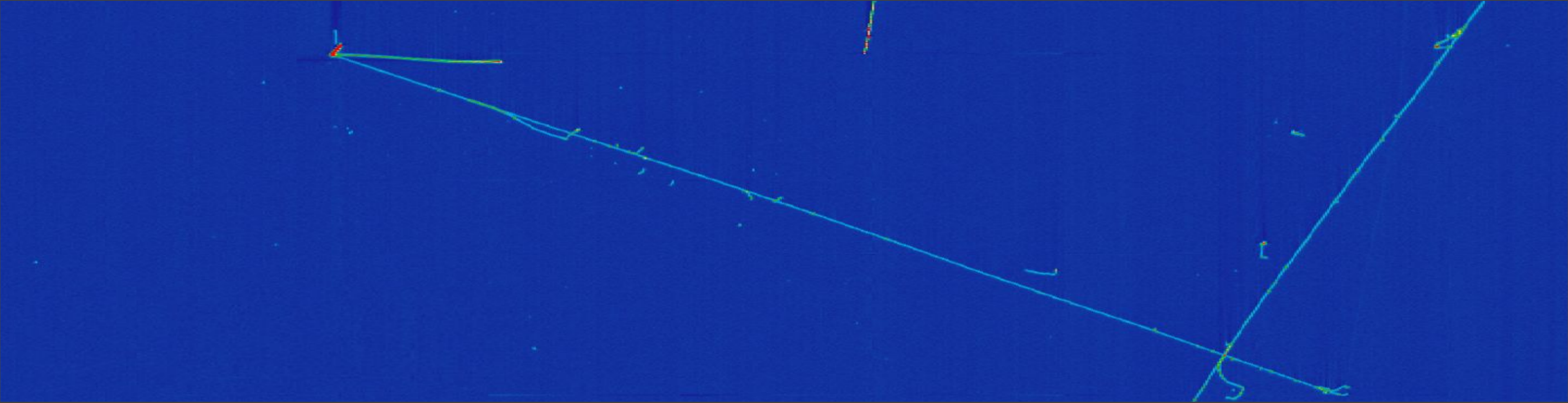
Detectors must be capable of measuring **type** & **energy**

Challenges

1. Lack of an automated, quality analysis methods for big image data
2. Manual ("by-hand") workflow for development & tuning
3. Imperfect physics modeling

Image "hand scanning"
by professionals was how neutrino data had been analyzed from imaging detectors for long time





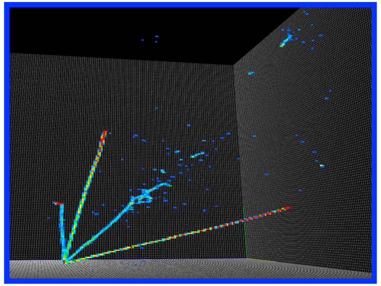
Outline

1. Introduction
2. ML-based data reconstruction
3. Differentiable simulation for detector physics modeling
4. Summary

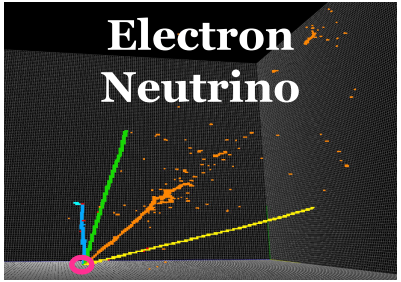
End-to-end data reconstruction using ML

Machine Learning for Neutrino Image Data Analysis

- **Goal**: particle-level type and energy reconstruction



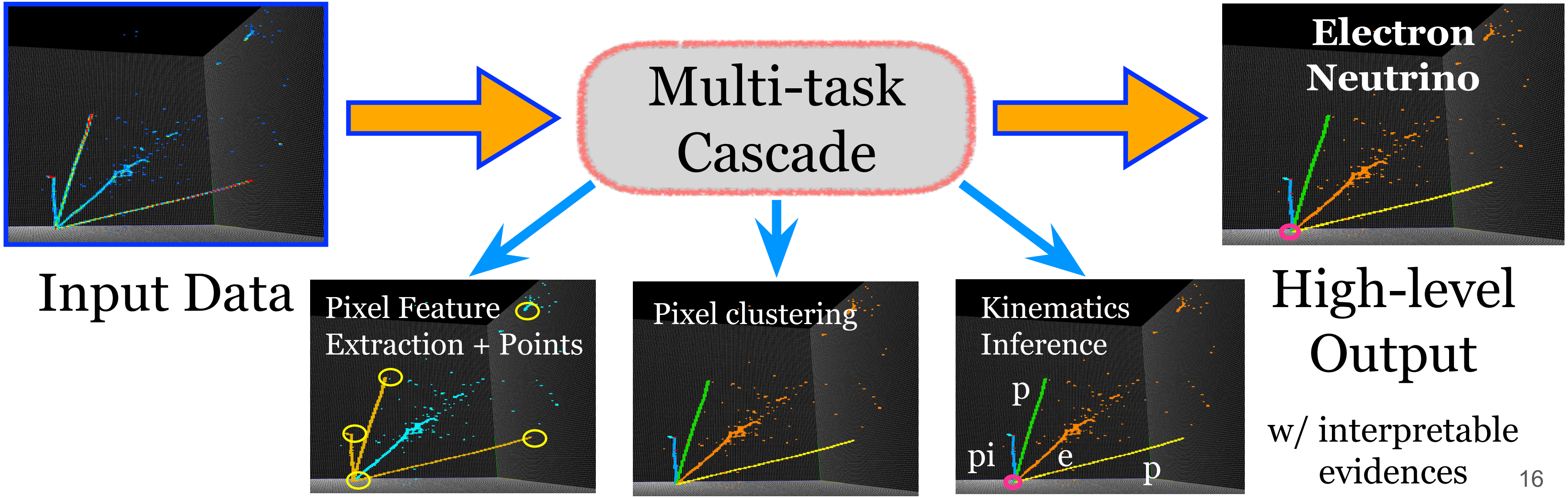
Input Data



High-level Output

Machine Learning for Neutrino Image Data Analysis

- **Goal**: particle-level type and energy reconstruction
- **How**: extract physically meaningful, hierarchical features (evidences) by chaining multiple ML models designed for each task

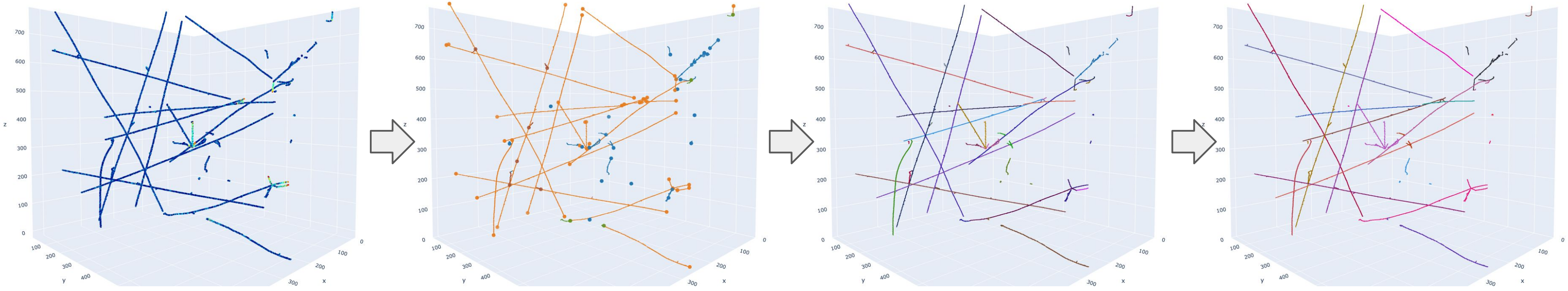


End-to-end data reconstruction using ML

Machine Learning for Neutrino Image Data Analysis

- **Goal**: particle-level type and energy reconstruction
- **How**: extract physically meaningful, hierarchical features (evidences) by chaining multiple ML models designed for each task

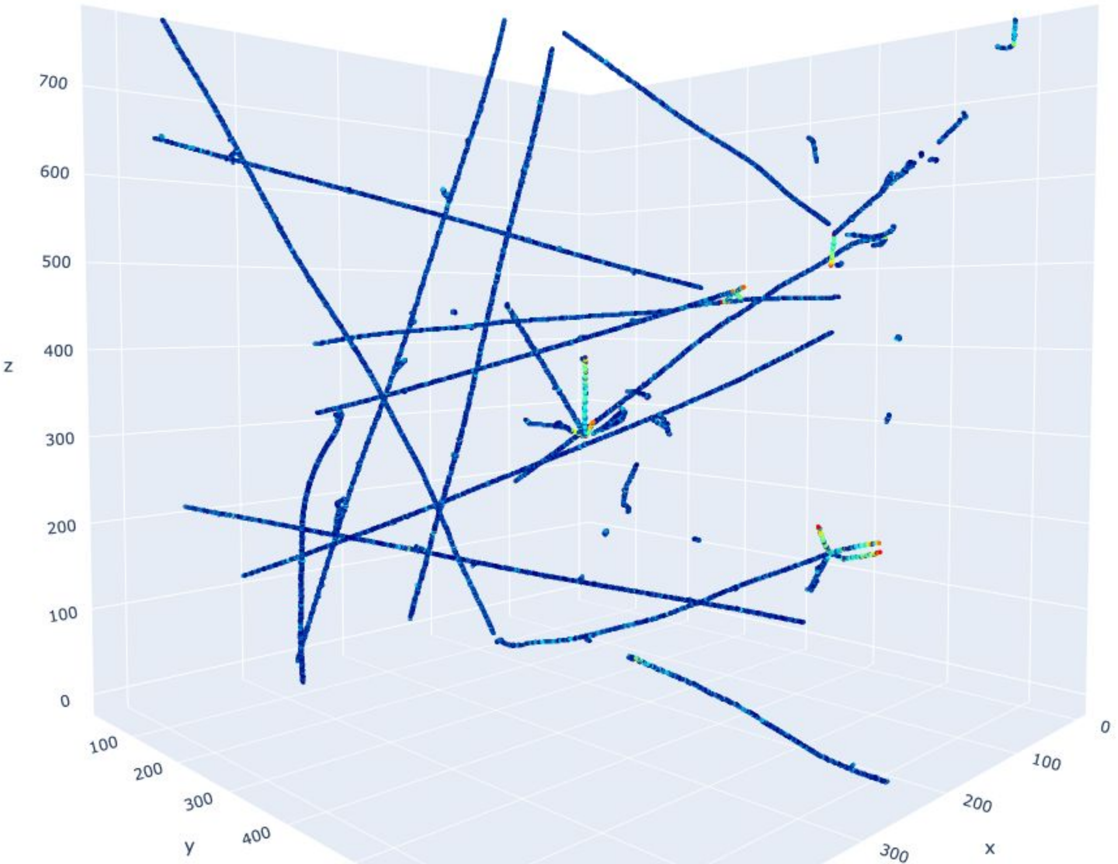
Input | Pixel Features | Pixel Clustering | Particle Clustering



Three major stages of reconstruction

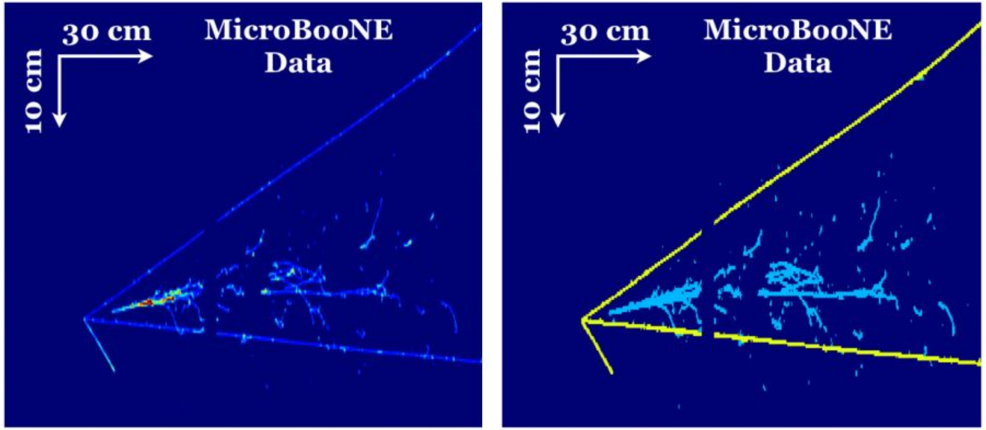
Machine Learning in Neutrino Physics & HEP

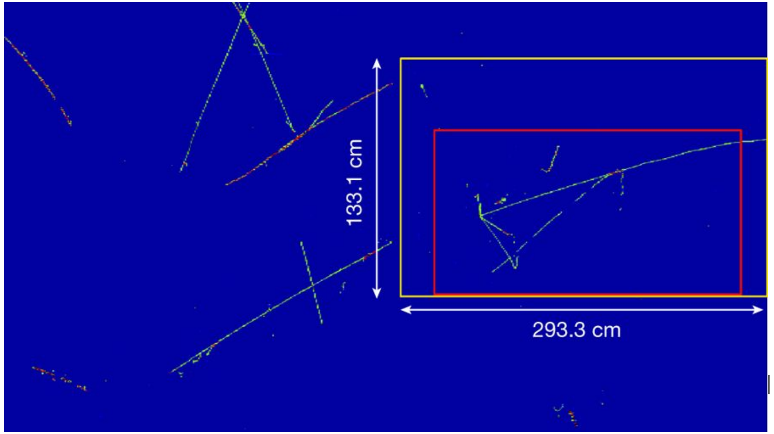
Deep Neural Network for Data Reconstruction



Convolutional neural network (CNN)

- Primarily aimed at image data
- Learns spatially local features of various size
- Translation invariant (target feature can be anywhere in image)
- Image/Pixel level classification/regression, object detection

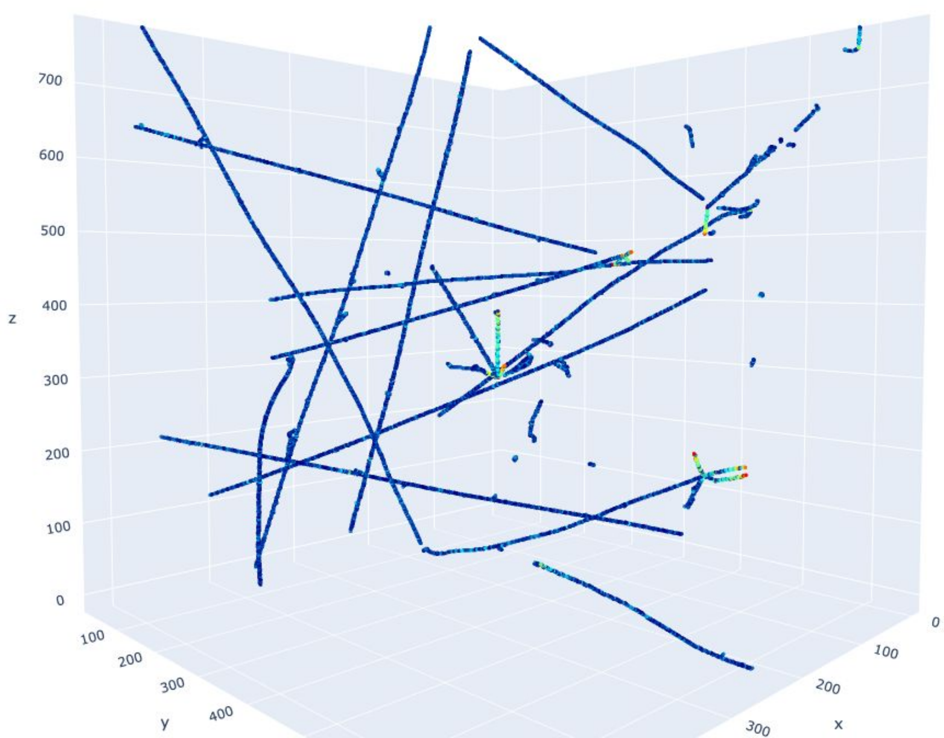


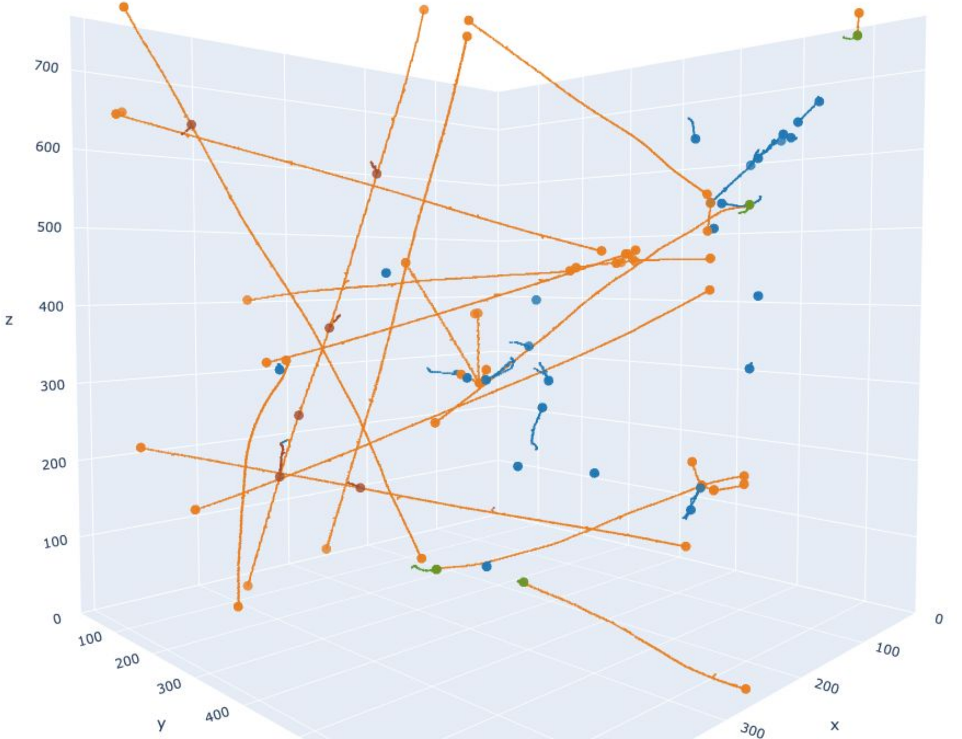


Deep Neural Network for Data Reconstruction

SLAC

Stage 1





Clustering in the embedding space

- Use CNN to learn a transformation function from the 3D voxels to the embedding space where clustering can be performed in a simple manner

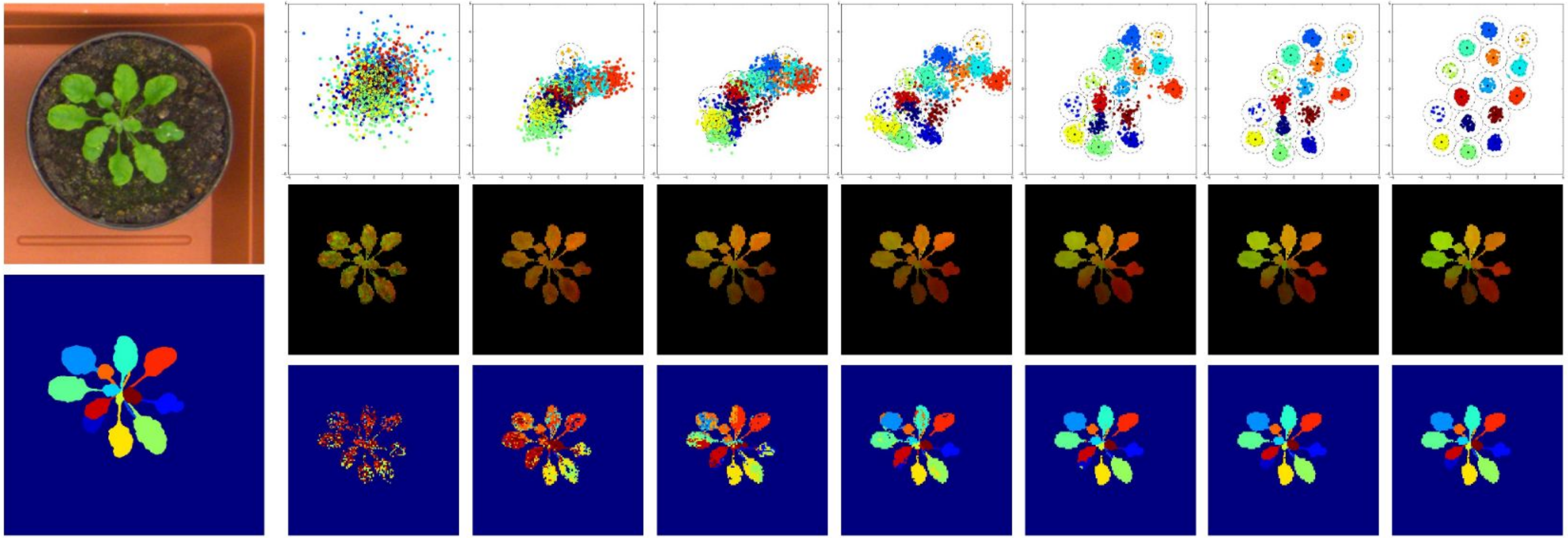
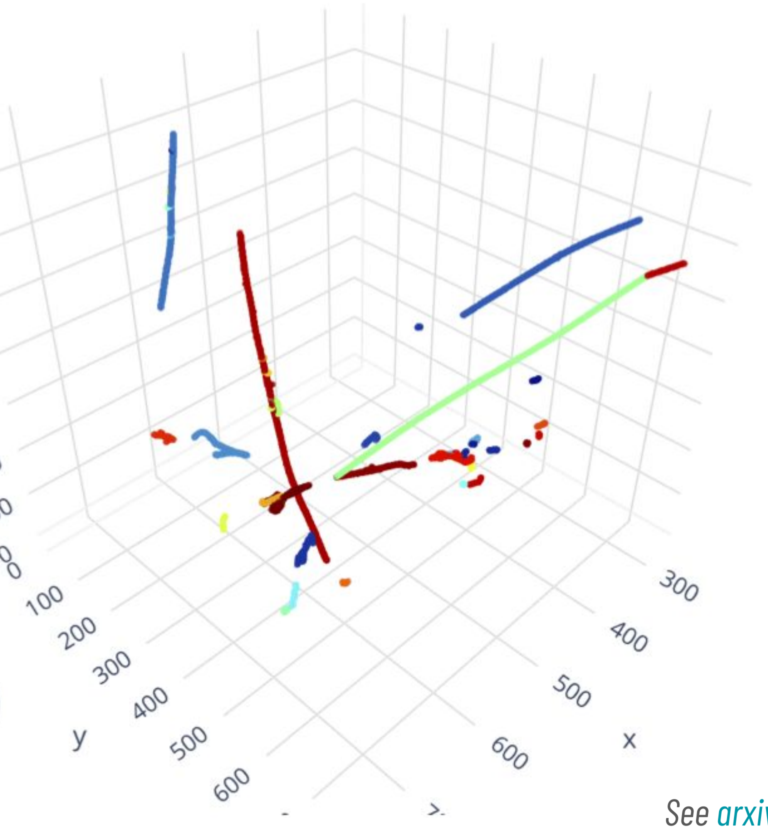
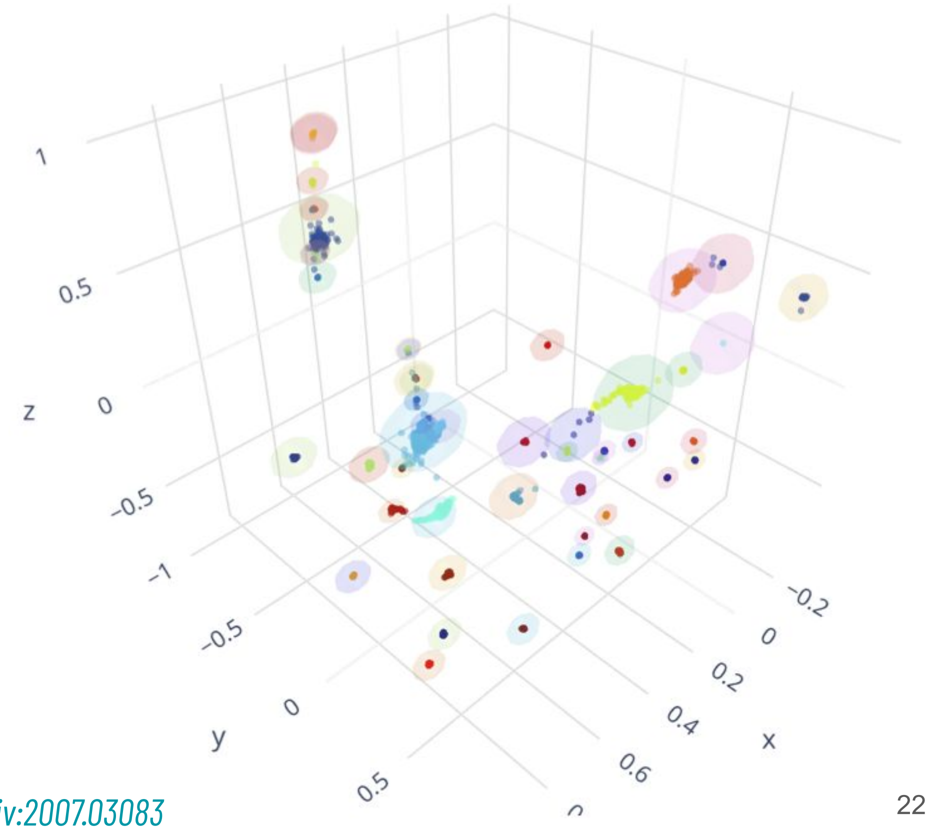


Image credit: arXiv 1708.02551

Stage 2-a: Dense Pixel Clustering





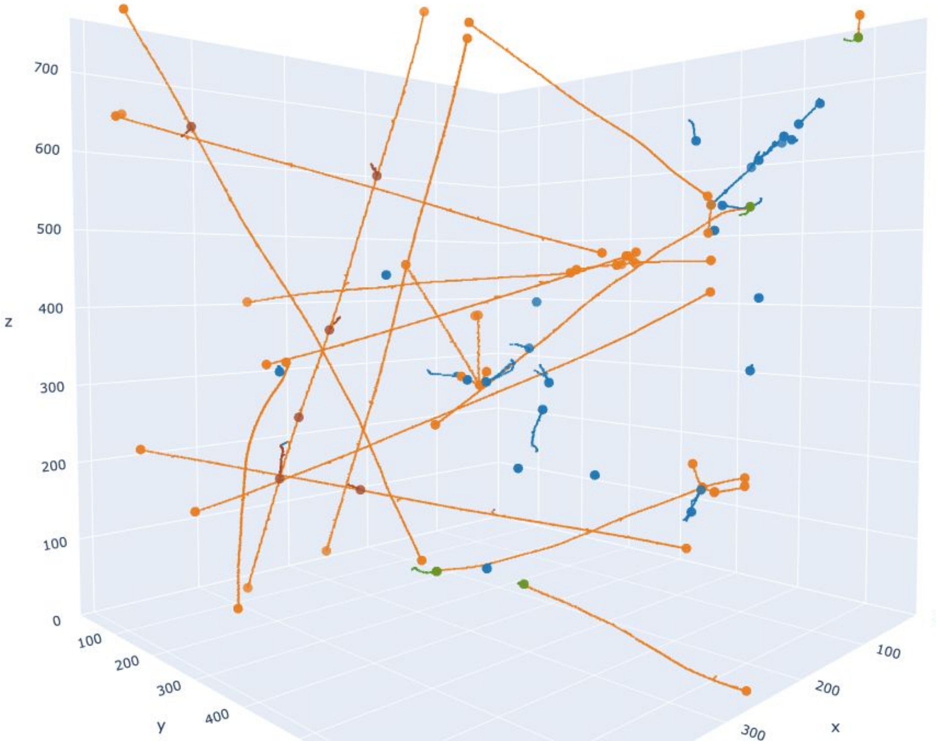
Work credit:
Dae Heun Koh (Stanford)

See [arxiv:2007.03083]

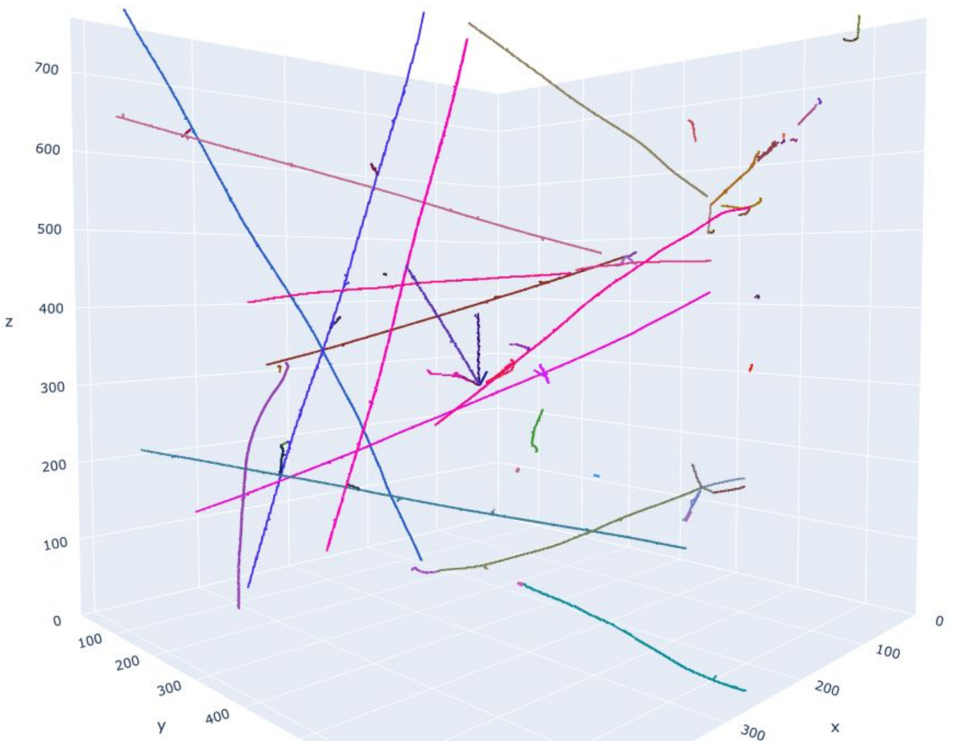
ML for Analyzing Big Image Data in Neutrino Experiments

Stage 2-a: input & output

Stage 2-a Input

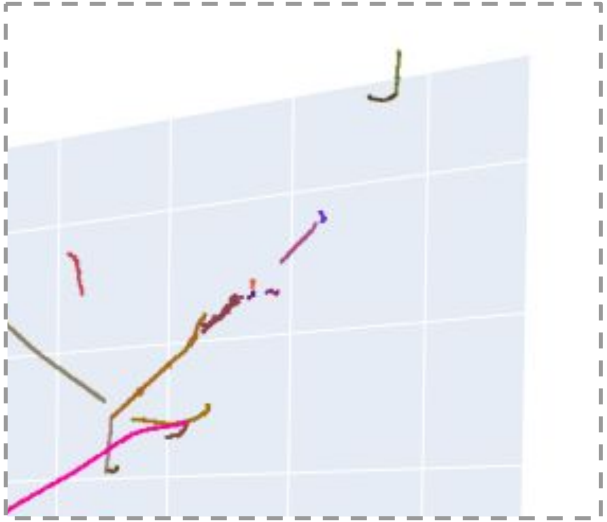


Stage 2-a Output

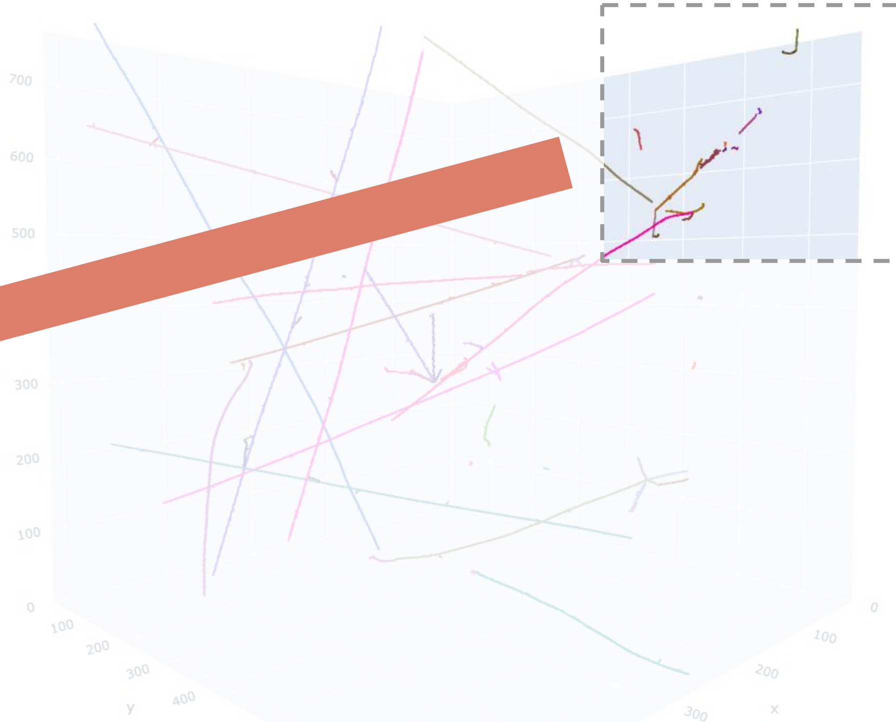


ML for Analyzing Big Image Data in Neutrino Experiments

Stage 2: grouping particles as a cluster



Stage 2-a Output

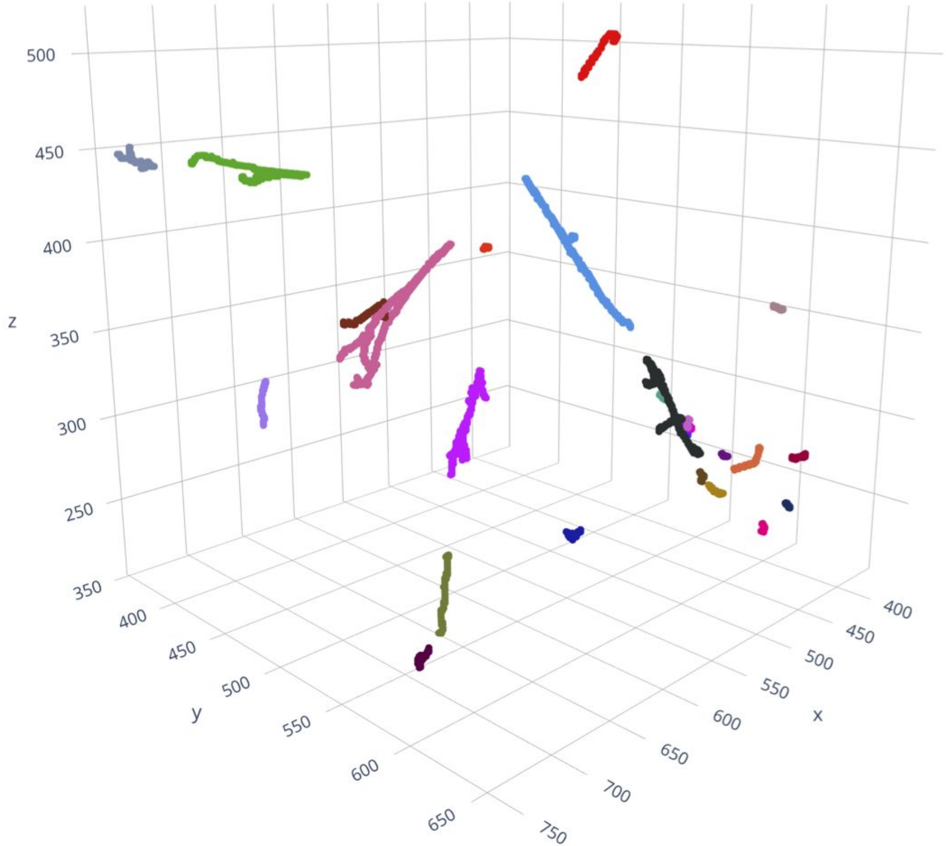


Stage 2-b: Sparse Fragment Clustering

Graph-NN for Particle Aggregation (GrapPA)

Input:

- Fragmented EM showers



See [Phys. Rev. D 104, 072004]

ML for Analyzing Big Image Data in Neutrino Experiments
Stage 2-b: Sparse Fragment Clustering

SLAC

Graph-NN for Particle Aggregation (GrapPA)

Input:

- Fragmented EM showers

Node features:

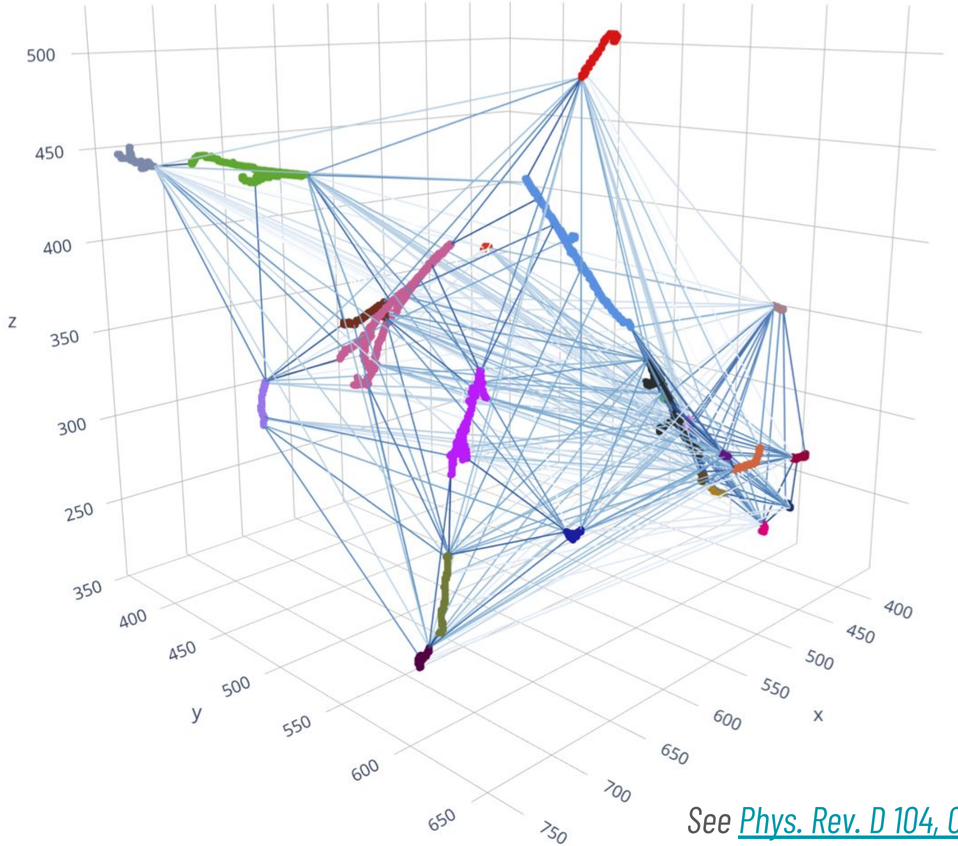
- Centroid, Covariance matrix, PCA
- Start point, direction (PPN)

Input graph:

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

Edge features:

- Displacement vector
- Closest points of approach

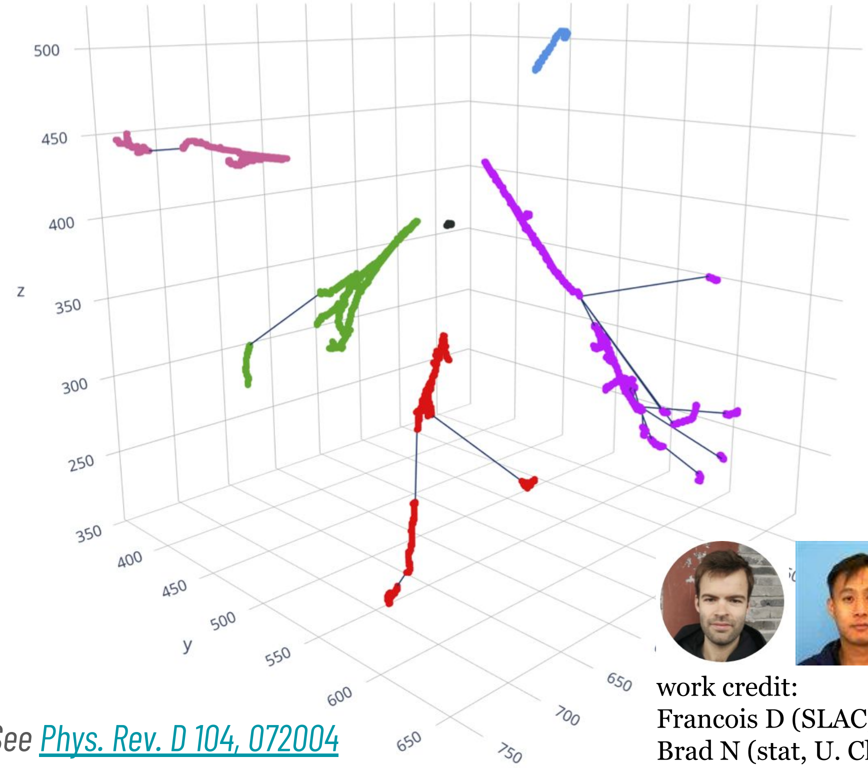


See Phys. Rev. D 104, 072004

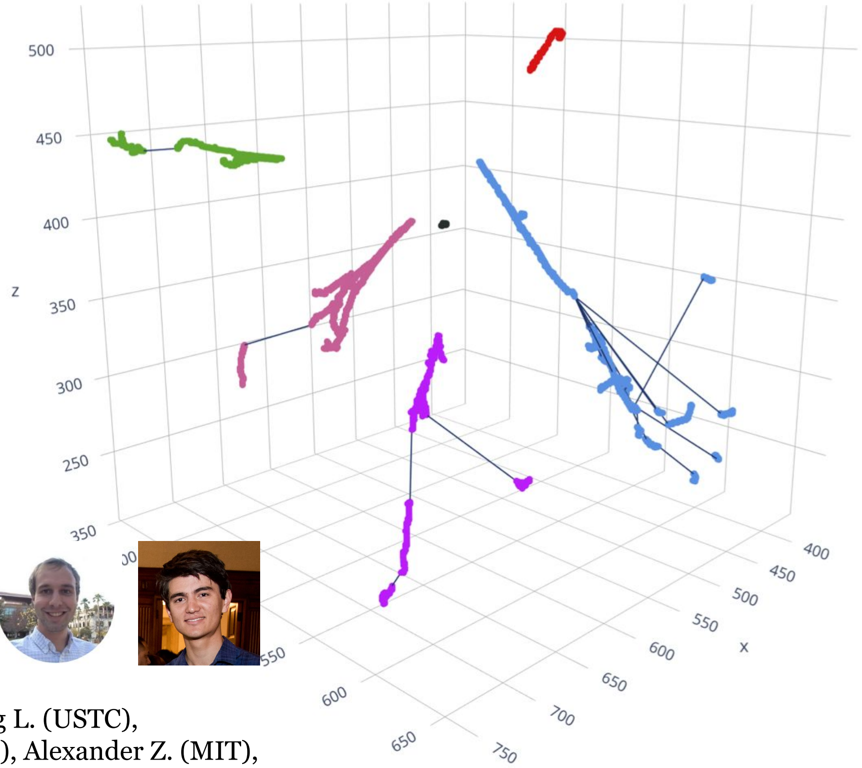
ML for Analyzing Big Image Data in Neutrino Experiments

Stage 2-b: Sparse Fragment Clustering

Target



Prediction



work credit:
Francois D (SLAC), Qing L. (USTC),
Brad N (stat, U. Chicago), Alexander Z. (MIT),

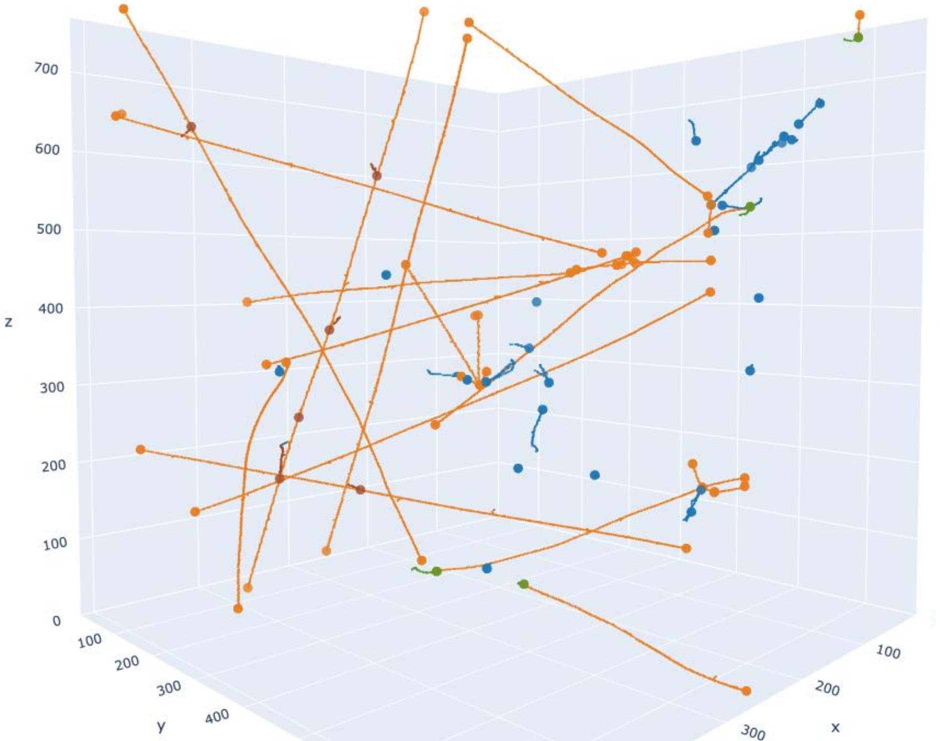
See [Phys. Rev. D 104, 072004]

ML for Analyzing Big Image Data in Neutrino Experiments

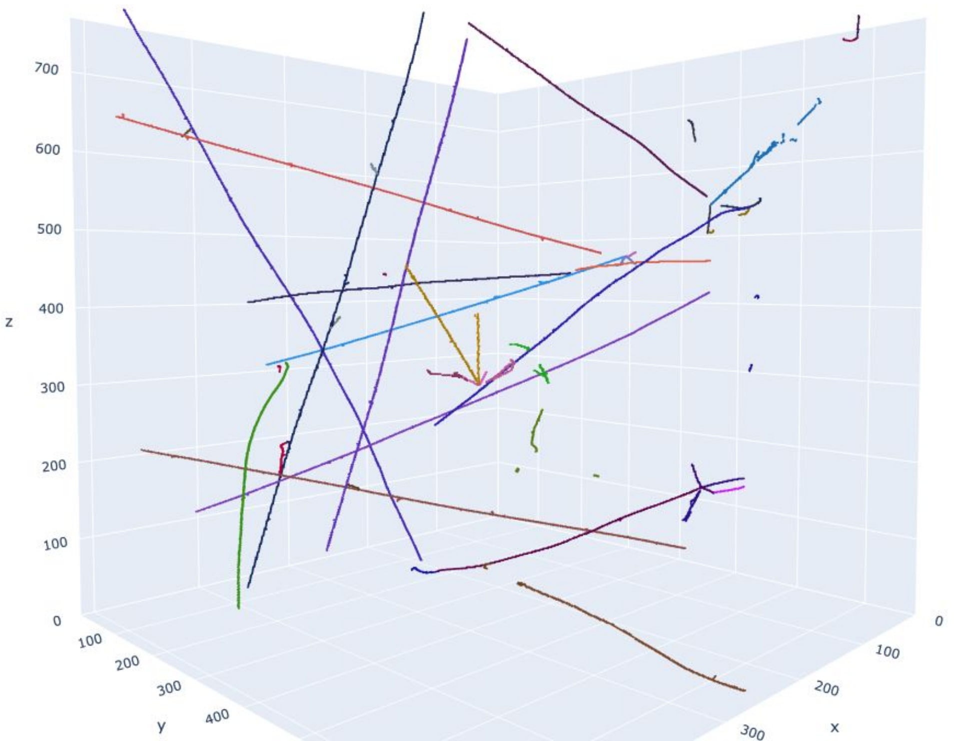
Stage 2: input & output

SLAC

Stage 2 Input

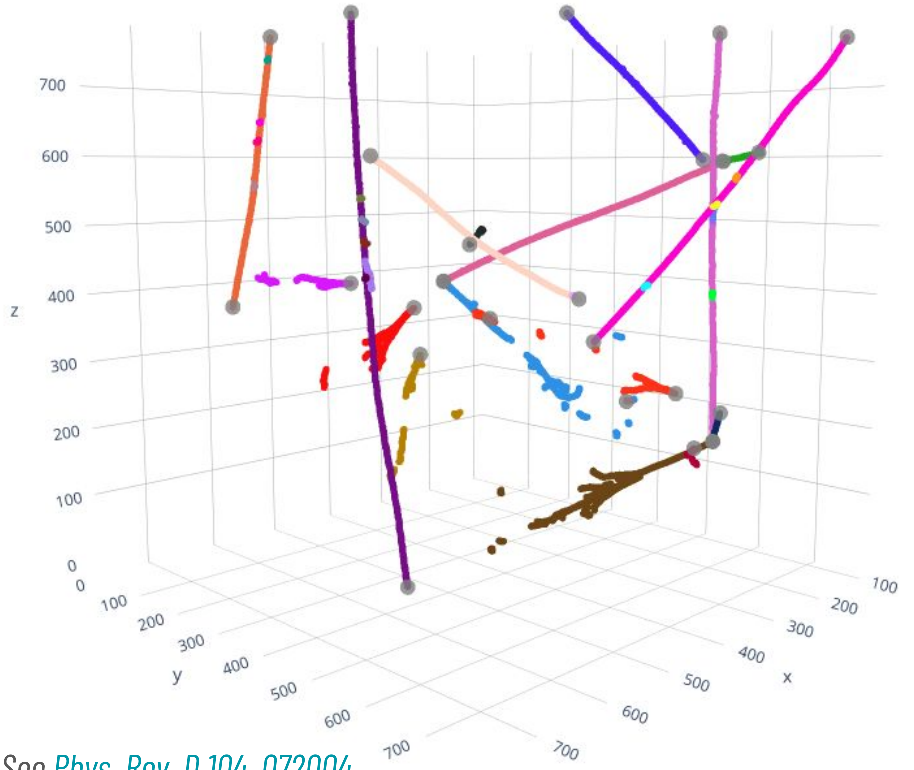


Stage 2 Output



ML for Analyzing Big Image Data in Neutrino Experiments

Stage 3: Interaction Clustering



Identifying Each Interaction?

Grouping task = re-use GrapPA!

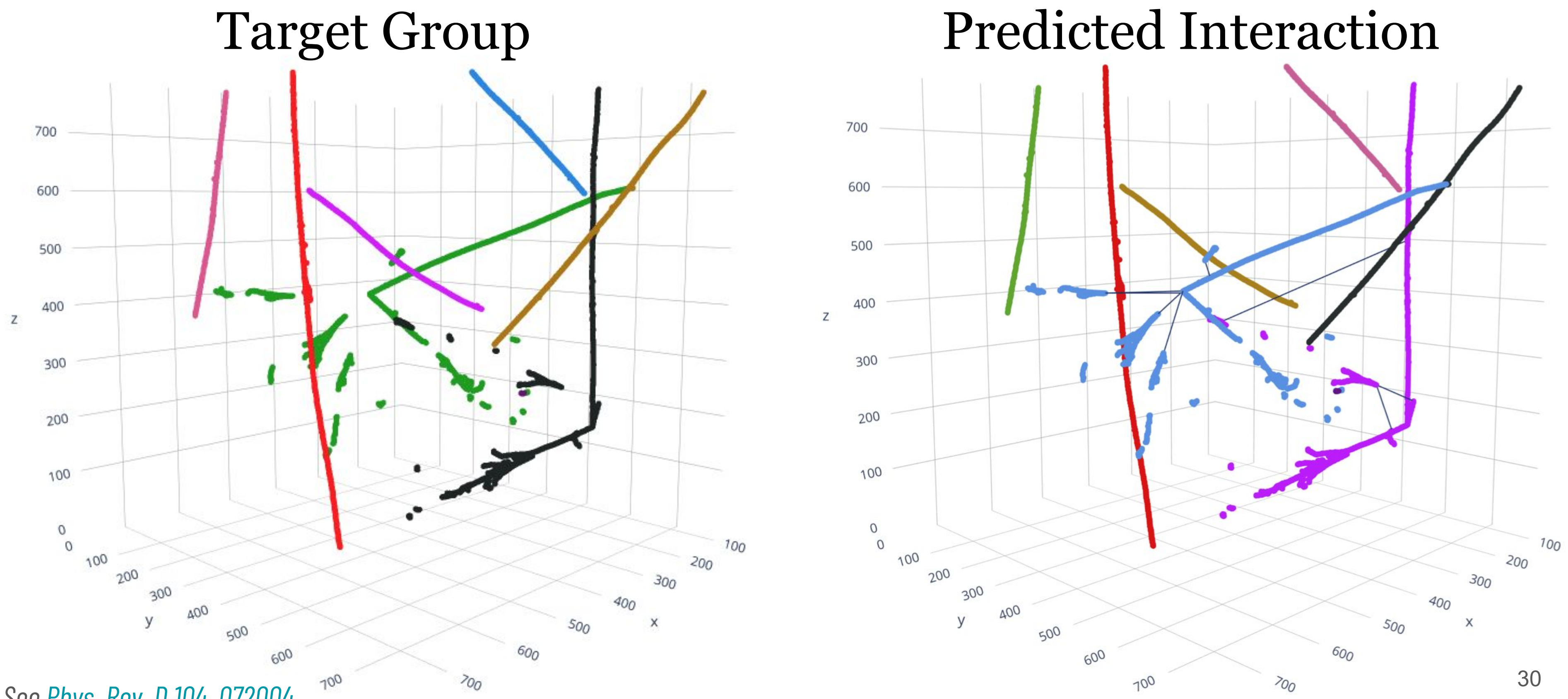
- Interaction = a group of particles that shared the same origin (i.e. neutrino interaction)
- Edge classification to identify an interaction
- Node classification for particle type ID

See Phys. Rev. D 104, 072004

ML for Analyzing Big Image Data in Neutrino Experiments

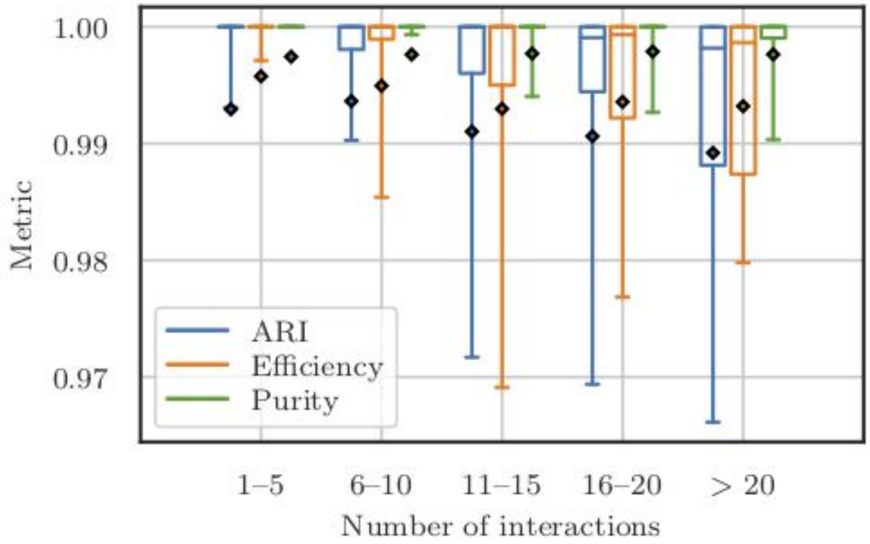
Stage 3: Interaction Clustering

SLAC

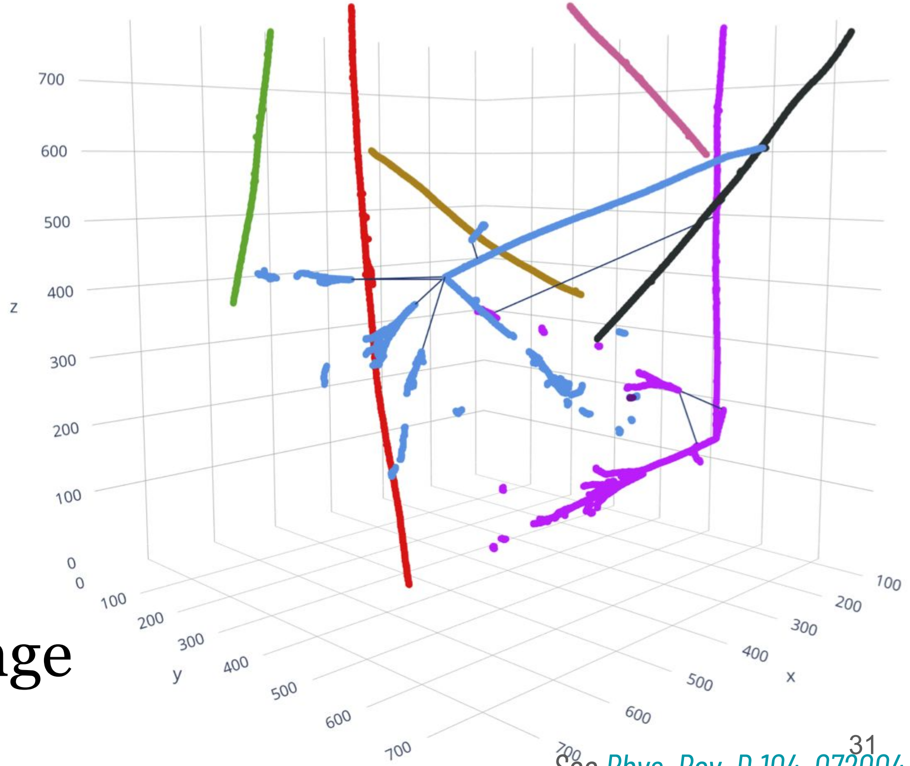


See [Phys. Rev. D 104, 072004]

ML for Analyzing Big Image Data in Neutrino Experiments
Stage 3: Interaction Clustering



Predicted Interaction

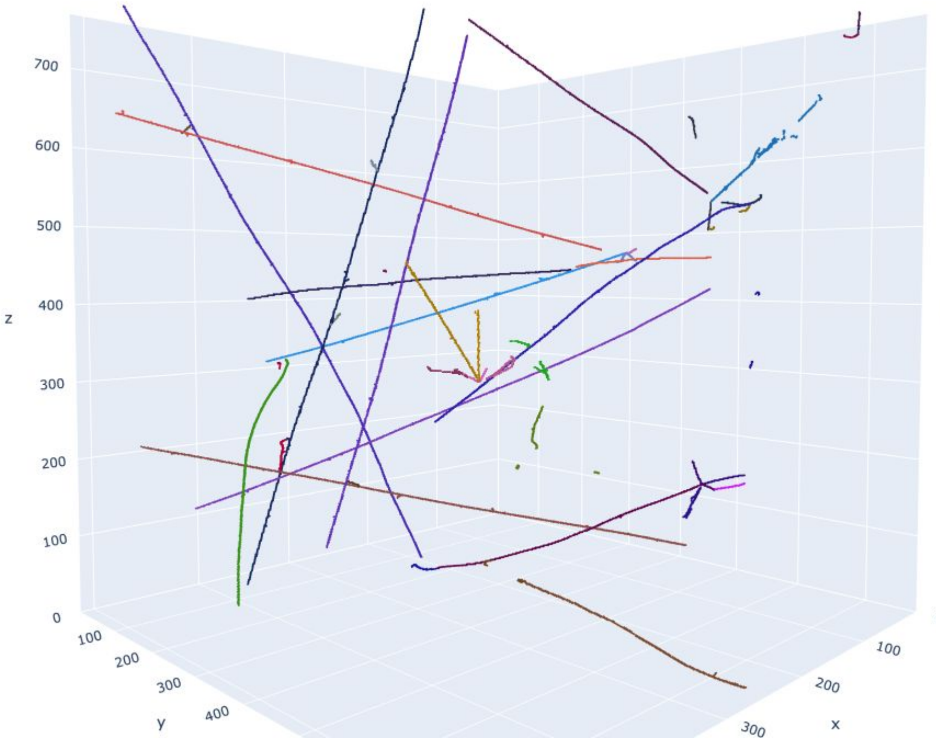


Promising result to address DUNE-ND reconstruction challenge (~20 neutrino pile-up)

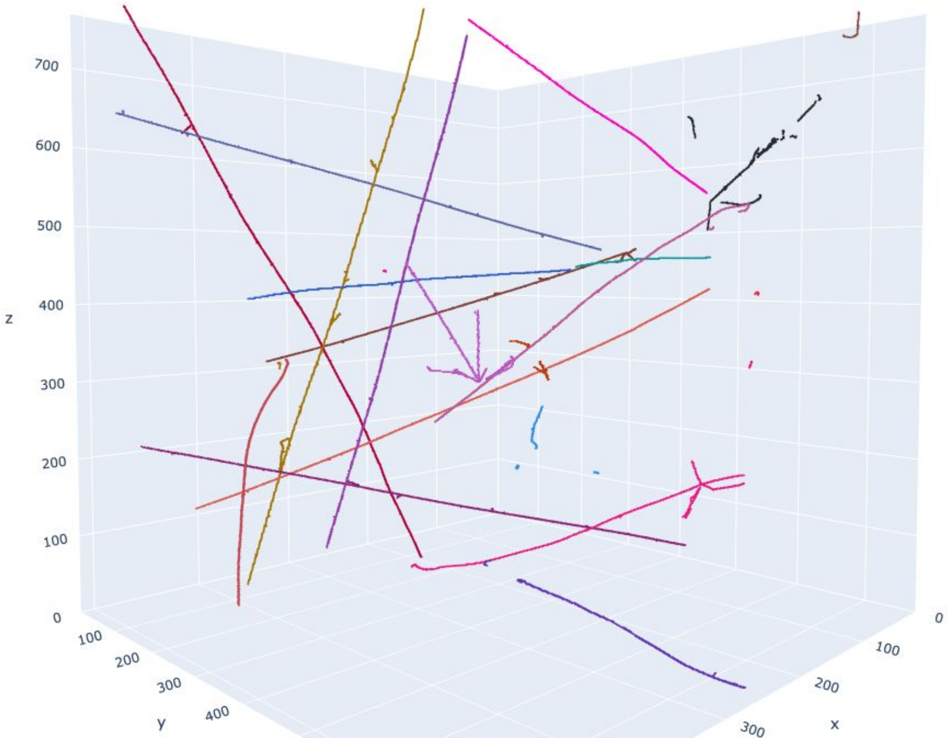
See Phys. Rev. D 104, 072004

ML for Analyzing Big Image Data in Neutrino Experiments
Stage 3: input & output

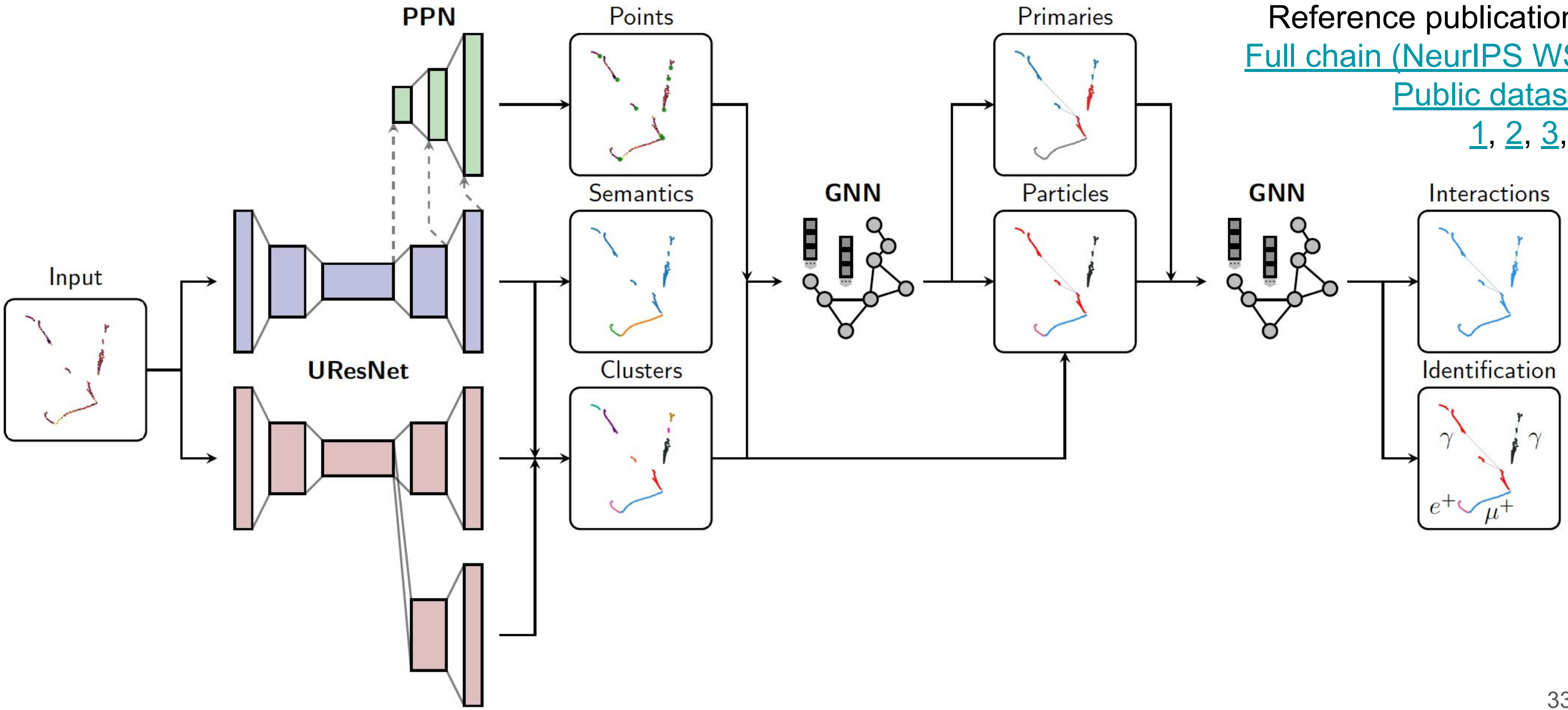
Stage 3 Input



Stage 3 Output



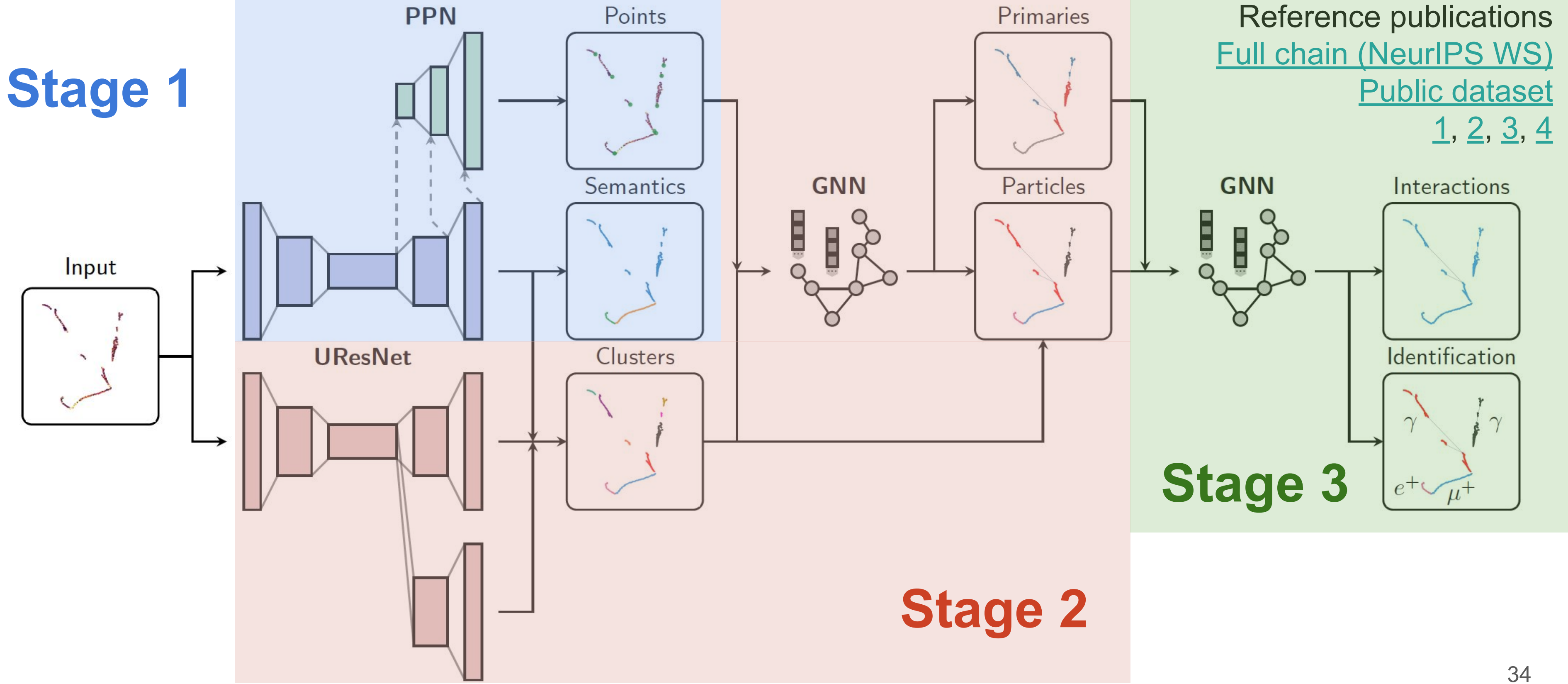
Machine Learning in Neutrino Physics & HEP
Deep Neural Network for Data Reconstruction



Reference publications
[Full chain (NeurIPS WS)]
[Public dataset]
1, 2, 3, 4

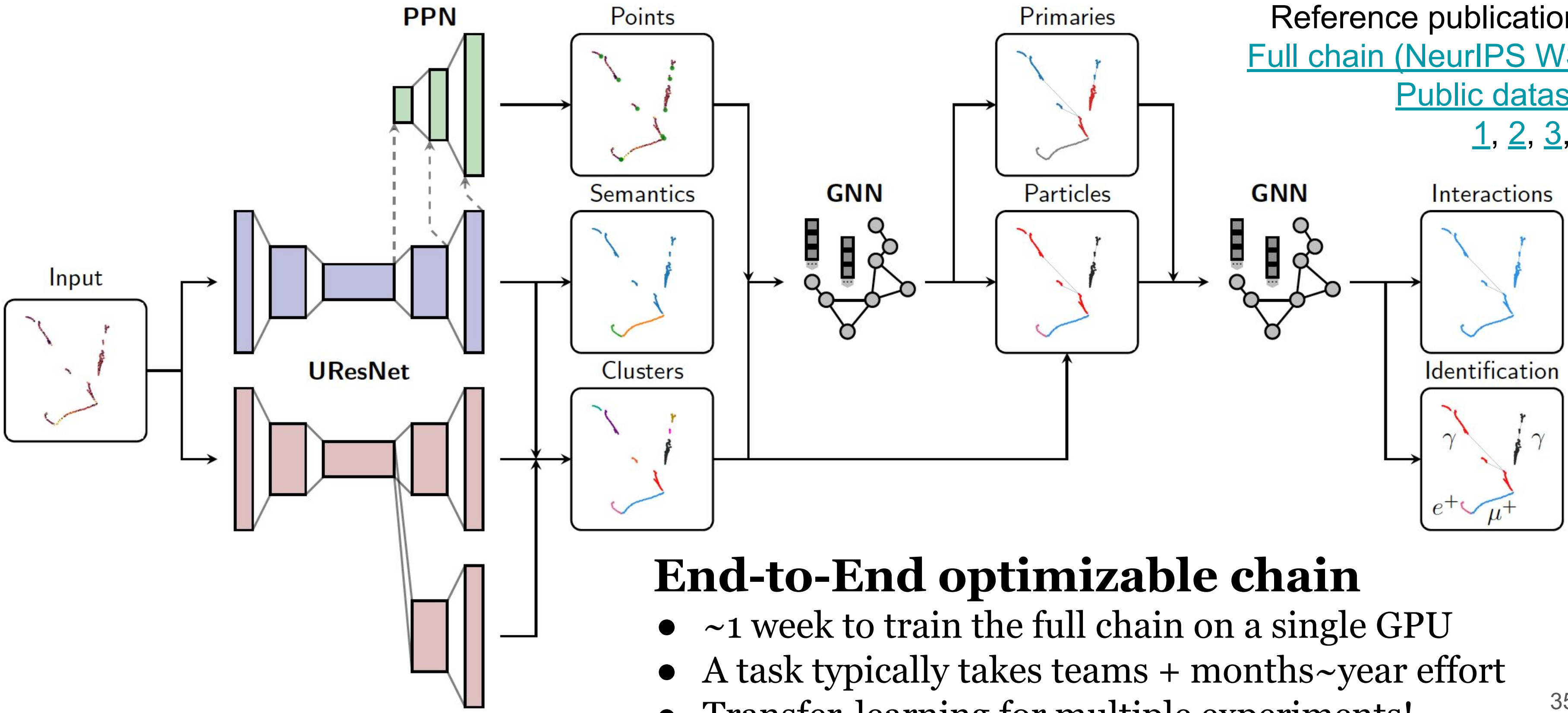
Machine Learning in Neutrino Physics & HEP
Deep Neural Network for Data Reconstruction

Reference publications
Full chain (NeurIPS WS)
Public dataset
1, 2, 3, 4



ML for Analyzing Big Image Data in Neutrino Experiments

Deep Neural Network for Data Reconstruction

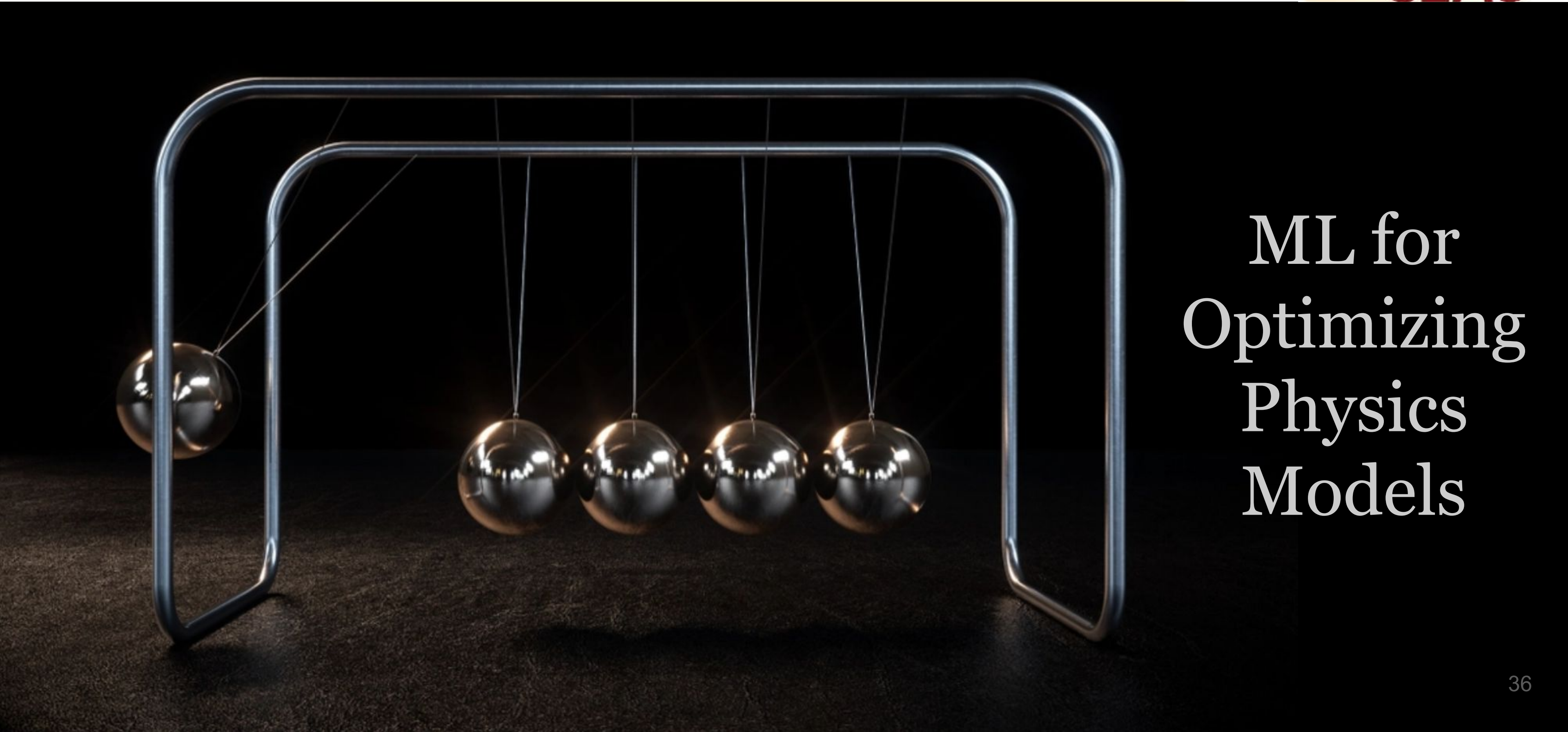


Reference publications
Full chain (NeurIPS WS)
Public dataset
1, 2, 3, 4

End-to-End optimizable chain

- ~1 week to train the full chain on a single GPU
- A task typically takes teams + months~year effort
- Transfer-learning for multiple experiments!

Automation of physics model tuning

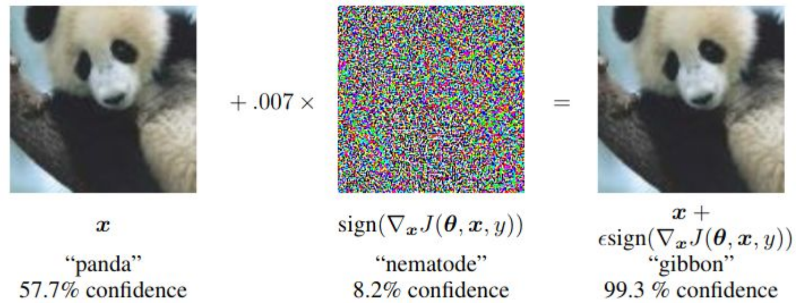


Automation of physics model tuning

Explaining and harnessing adversarial examples

The Catch

Supervised optimization with imperfect simulation may be vulnerable to **domain shift**.



Automation of physics model tuning

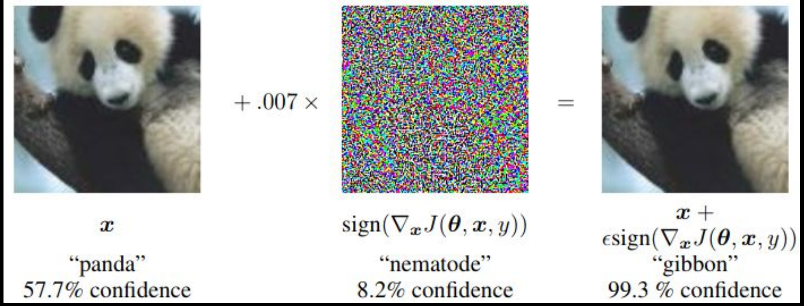
SLAC

Explaining and harnessing adversarial examples

The Catch

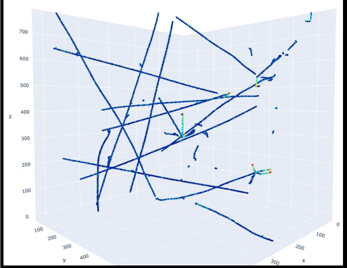
Supervised optimization with imperfect simulation may be vulnerable to **domain shift**.

Tuning of simulation: tricky & "by hand"



Detector physics knowledge applied in simulation

Detector physics knowledge extracted in reconstruction



Simulation Input (true dE/dX)

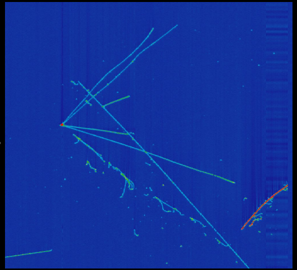
Detector Simulation

$$\mathcal{R}_{\text{ICARUS}} = \frac{A_B}{1 + k_B \cdot (dE/dx)/\mathcal{E}}$$

$$Q = Q_0 \exp(-v_{\text{drift}} t/\tau)$$

$$\sigma_t^2(t) \simeq \sigma_t^2(0) + \left(\frac{2D_L}{v_d^2}\right) t$$

…

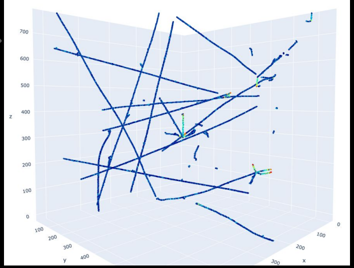


Detector Output (ADC)

Reconstruction

Detector physics model parameters

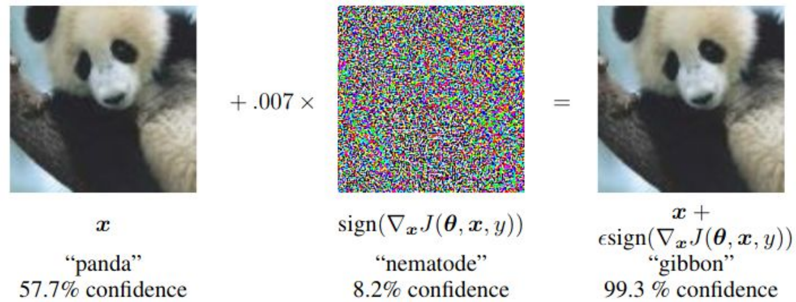
Calibration



Reconstructed (Calibrated dE/dX)

Automation of physics model tuning

Explaining and harnessing adversarial examples



The Catch

Supervised optimization with imperfect simulation may be vulnerable to **domain shift**.

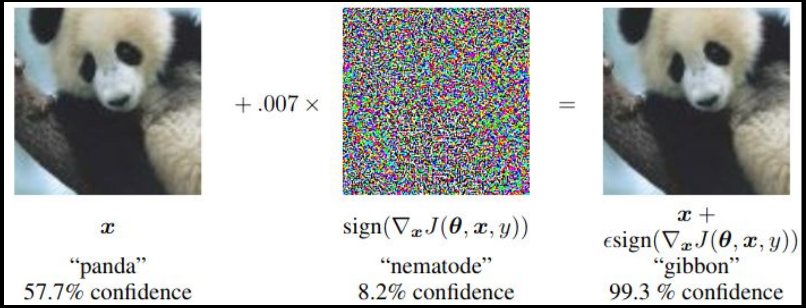
Tuning of simulation: tricky & "by hand"

Research directions

- Make the optimization of reco chain robust against domain shift
- Innovative simulator that can be automatically tuned with control dataset
- Learn data representations directly from data (+ use features to train reco chain)

ML for Detector Physics Modeling
Automation of physics model tuning

[Explaining and harnessing adversarial examples]



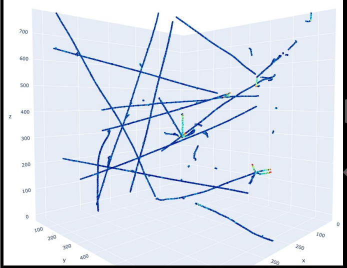
The Catch

Supervised optimization with imperfect simulation may be vulnerable to **domain shift**.

~~Tuning of simulation: tricky & "by hand"~~

Develop a simulator that can be tuned automatically on real data

Detector physics knowledge extracted in reconstruction



Simulation Input (true dE/dX)

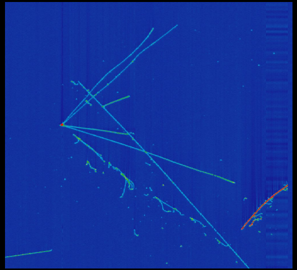
Detector Simulation

$$\mathscr{R}_{\mathrm{ICARUS}} = \frac{A_B}{1 + k_B \cdot (dE/dx)/\mathscr{E}}$$

$$Q = Q_0 \exp(-v_{\mathrm{drift}} t/\tau)$$

$$\sigma_t^2(t) \simeq \sigma_t^2(0) + \left(\frac{2D_L}{v_d^2}\right) t$$

...

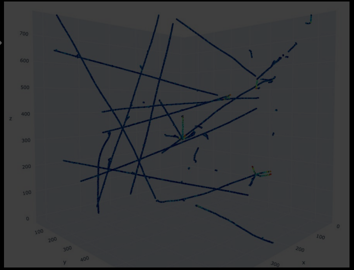


Detector Output (ADC)

Reconstruction

Detector physics model parameters

Calibration



Reconstructed (Calibrated dE/dX)

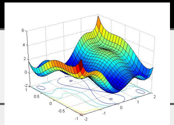
Gradient-based optimization

How: differentiable detector physics simulator

Yann LeCun
January 5, 2018

OK, Deep Learning has outlived its usefulness as a buzz-phrase.
Deep Learning est mort. Vive Differentiable Programming!

Approximated gradient



Input
x

physics model parameters
θ

Output
$F(x|\theta)$

Optimization target
$L(F(x|\theta), y)$

Exact gradient

Differentiable surrogate for optical photon transport

Optical Detector Simulation

ML for Detector Physics Modeling
LAr scintillator light detection

Photo-multiplier tubes (PMTs) detect scintillation photons

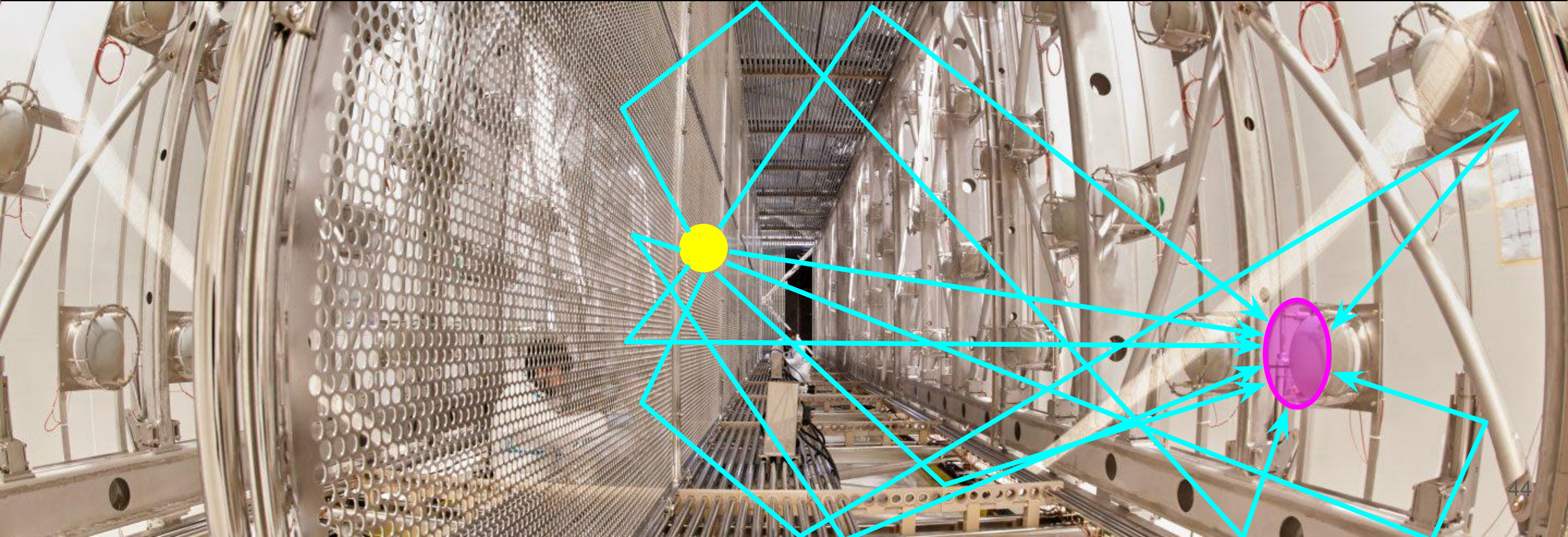
Optical Photon Transport

ML for Detector Physics Modeling
LAr scintillator light detection

Photo-multiplier tubes (PMTs) detect scintillation photons produced isotropically from an Argon atom

1 meter muon produces ~ **5M photons**

Optical Photon Transport

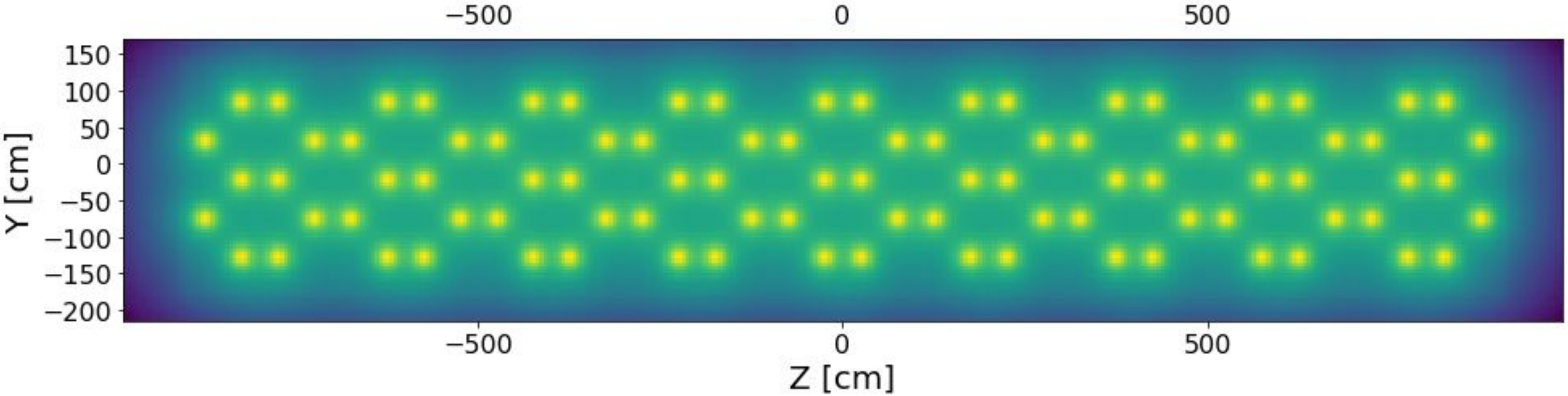


ML for Detector Physics Modeling
LAr scintillator light simulation

A marginalized "**Visibility Map**" for 3D voxelized volume used to estimate the mean photon count for each PMT

Issue: static and **not scalable**

Optical Photon Transport



Example: ICARUS detector, 2D slice of a 3D map

ML for Detector Physics Modeling
LAr scintillator light simulation

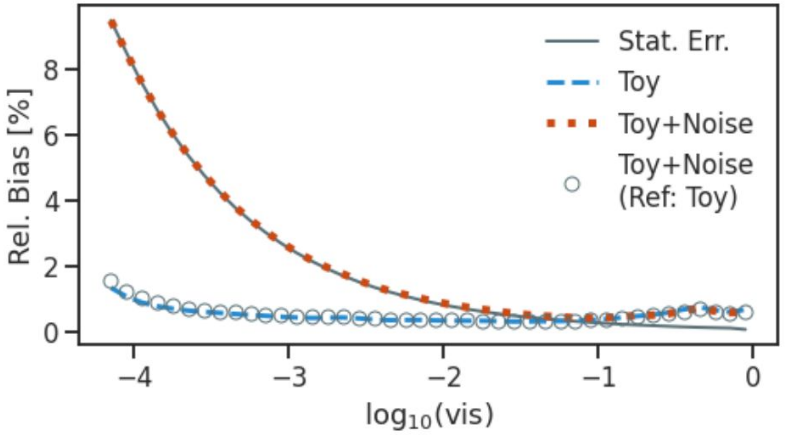
A marginalized "**Visibility Map**" for 3D voxelized volume used to estimate the mean photon count for each PMT

Issue: static and **not scalable**

Optical Photon Transport

$$\text{Relative Bias} = \frac{\left|\langle \text{P.E.} \rangle_{\text{true}} - \langle \text{P.E.} \rangle_{\text{pred.}}\right|}{\langle \text{P.E.} \rangle_{\text{true}} + \langle \text{P.E.} \rangle_{\text{pred.}}}$$

- Implicitly optimized based on simulation update (~2 weeks to produce each time)
- Limited scalability … ~1E9 voxels for ICARUS
  - Coarse voxel size (~5cm cubic)
  - Large statistical error (~30k photons/vox.)



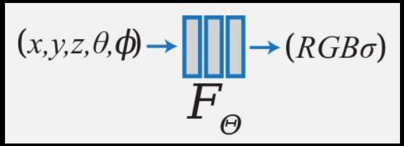
Difficult to scale full DUNE

Example: ICARUS detector, 2D slice of a 3D map

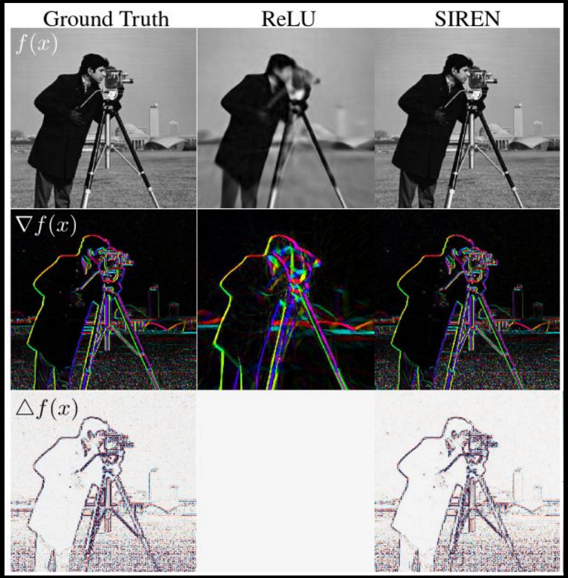
ML for Detector Physics Modeling

SIREN as a differentiable surrogate for optical detectors

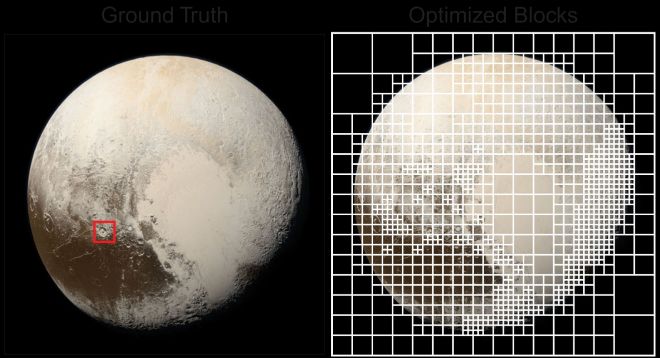
Differentiable Neural Scene Representation



NeRF breakthrough on high resolution image representation by a very simple nerual network



SIREN success of learning the 1st and 2nd order derivatives



ACORN scalable version of SIREN by adding spatial feature compression (essentially a learnable kd-tree)

… only a few examples

ML for Detector Physics Modeling

SIREN as a differentiable surrogate for optical detectors

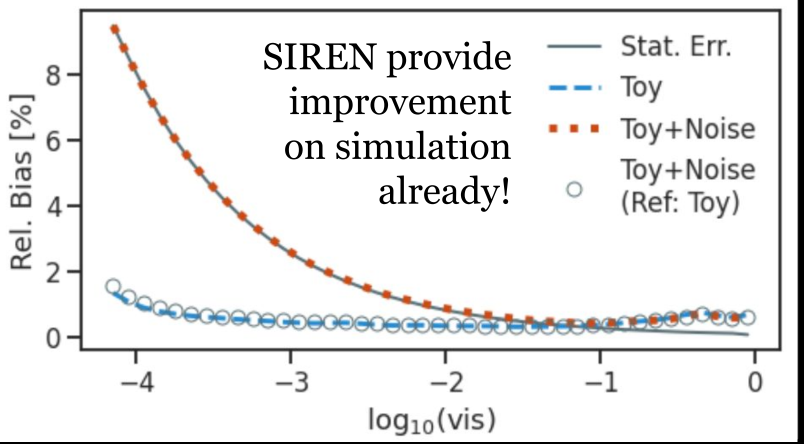
Differentiable Neural Scene Representation

SIREN for LArTPC detectors

- Designed as an implicit representation of a continuous function in space (suited to “visibility”, “E-field”, etc.)
  - Can be seen as a trade-off between an analytical function and a table
- “Differentiable” implies we can directly optimize against “data v.s. simulation discrepancy” given control samples

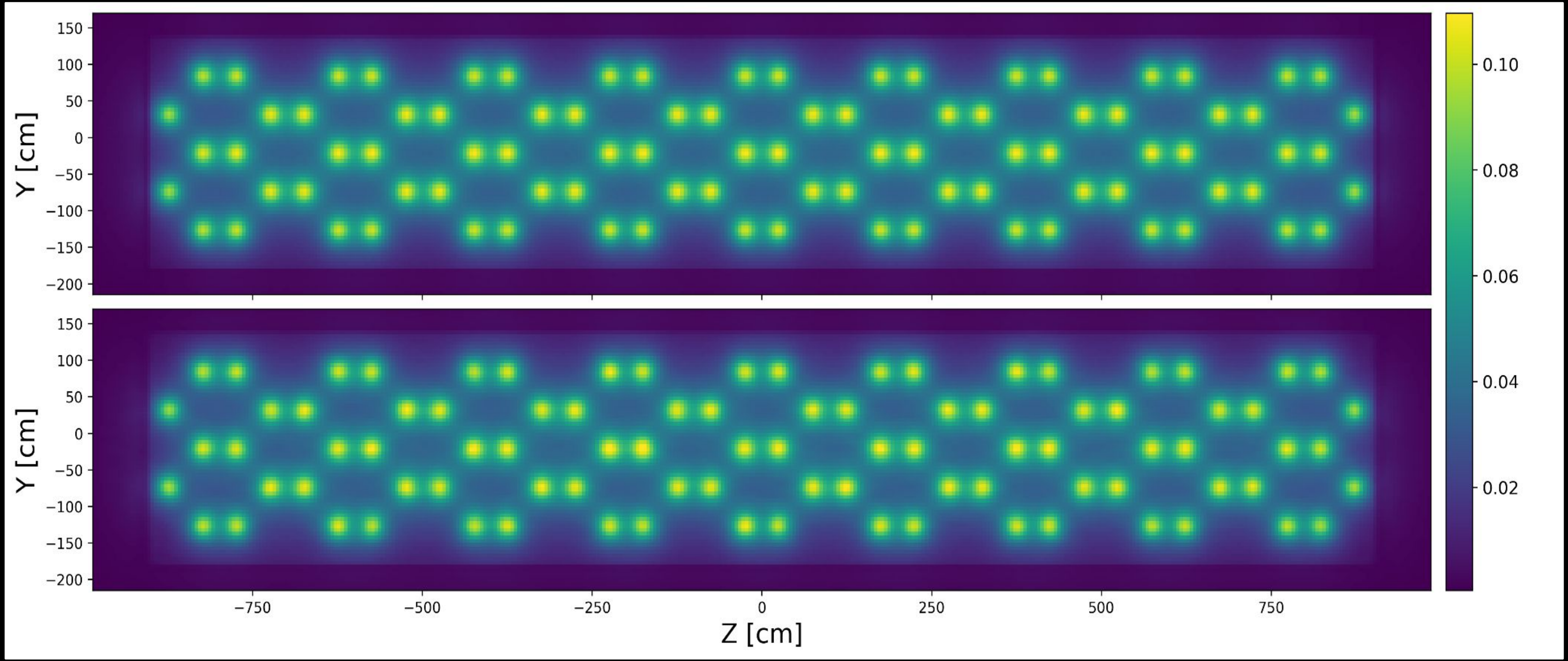
SIREN trained on “Toy + Noise” successfully learned the underlying analytical function shape (simulation)

$$\text{Relative Bias} = \frac{\left|\langle \text{P.E.} \rangle_{\text{true}} - \langle \text{P.E.} \rangle_{\text{pred.}}\right|}{\langle \text{P.E.} \rangle_{\text{true}} + \langle \text{P.E.} \rangle_{\text{pred.}}}$$



ML for Detector Physics Modeling
SIREN as a differentiable surrogate for optical detectors

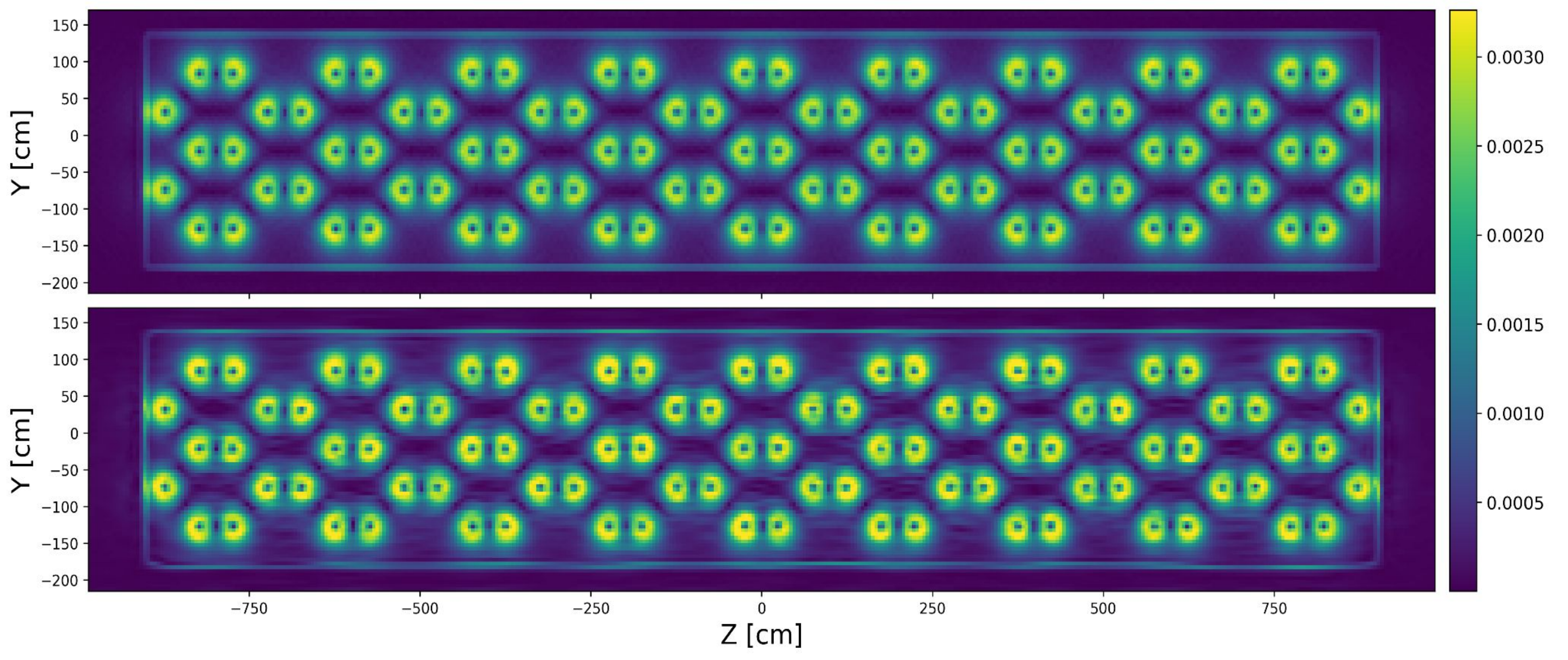
ICARUS: 2D slice, map (top) v.s. SIREN (bottom)



ML for Detector Physics Modeling

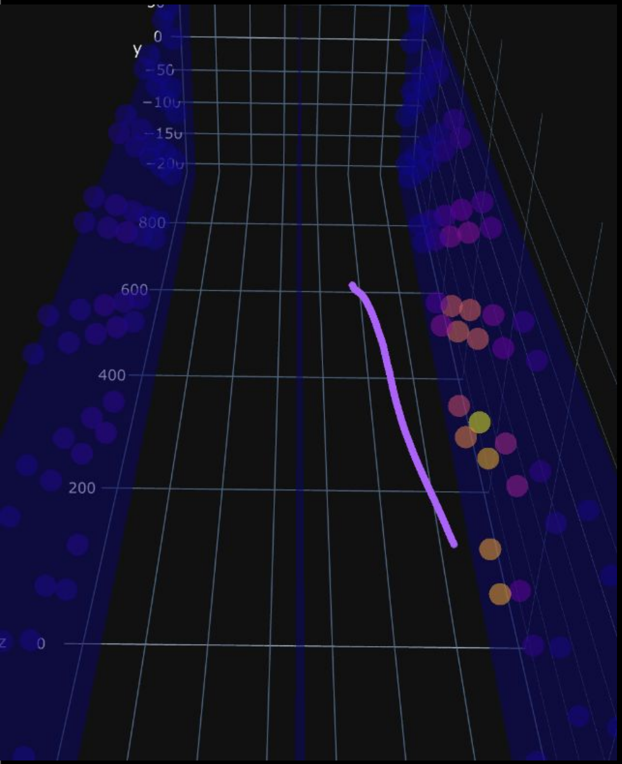
SIREN as a differentiable surrogate for optical detectors

ICARUS: 2D slice, map (top) v.s. SIREN (bottom)



SIREN as a differentiable surrogate for optical detectors

Training SIREN on real data



Control dataset: 3D TPC trajectory for which XYZ position of space-points are accurately measured

Predicted P.E. — Deposited charge at the point i — Quantum efficiency of the PMT j

$$P_j = \sum_i^{\text{all points}} C_i \times Y \times Q_j \times \Phi(\mathbf{r}_i)_j$$

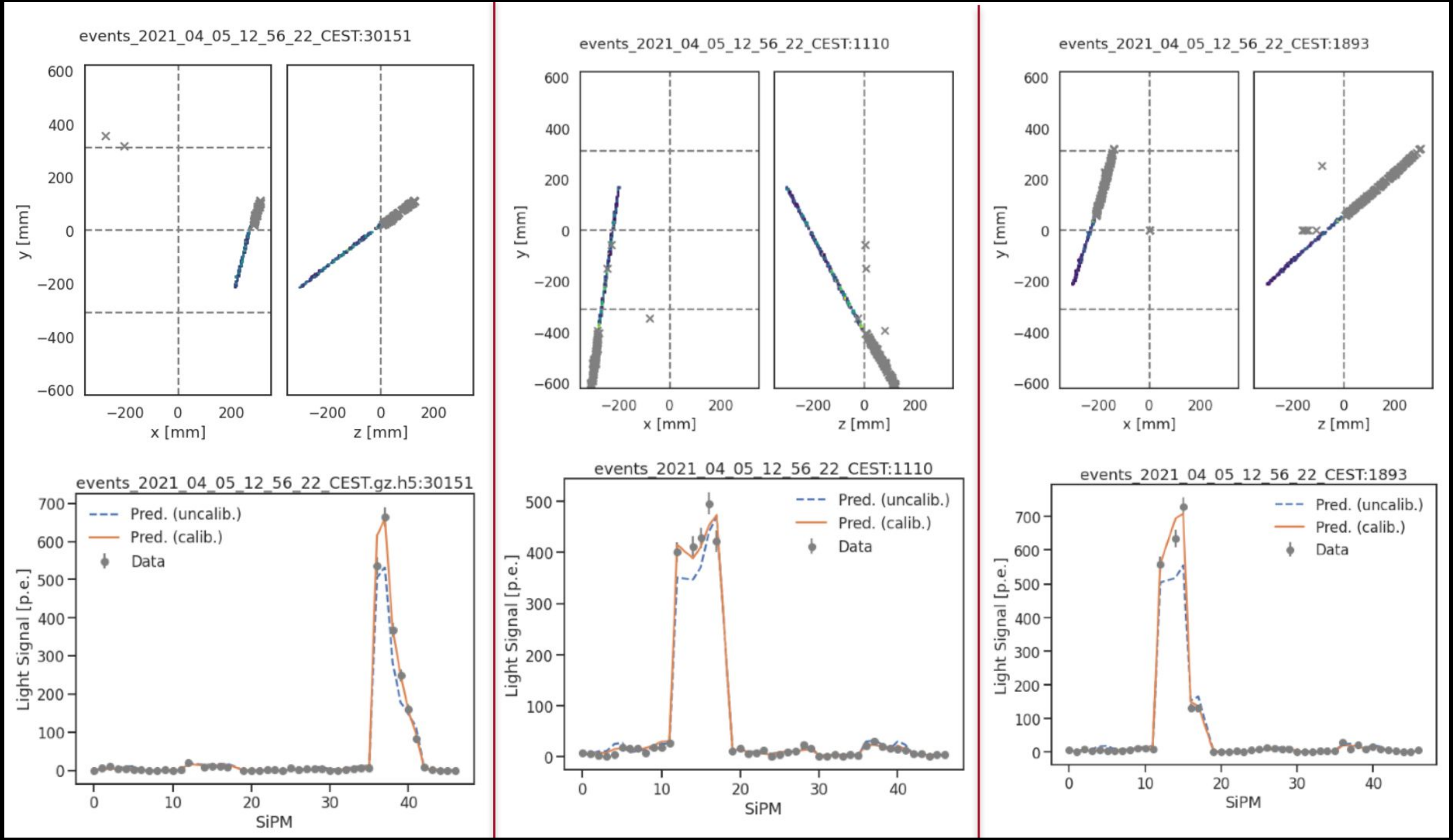
light yield — SIREN prediction for the point i at the PMT j

$$\text{Loss} = \sum_j^{\text{all PMTs}} \frac{(P_j - O_j)^2}{P_j^2 + \epsilon}$$

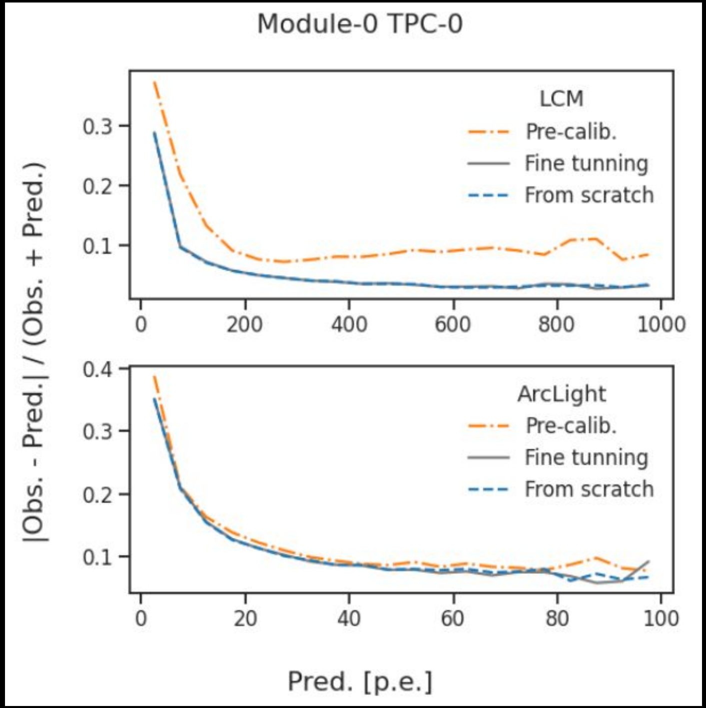
numerical stability, ~25 P.E.-squared

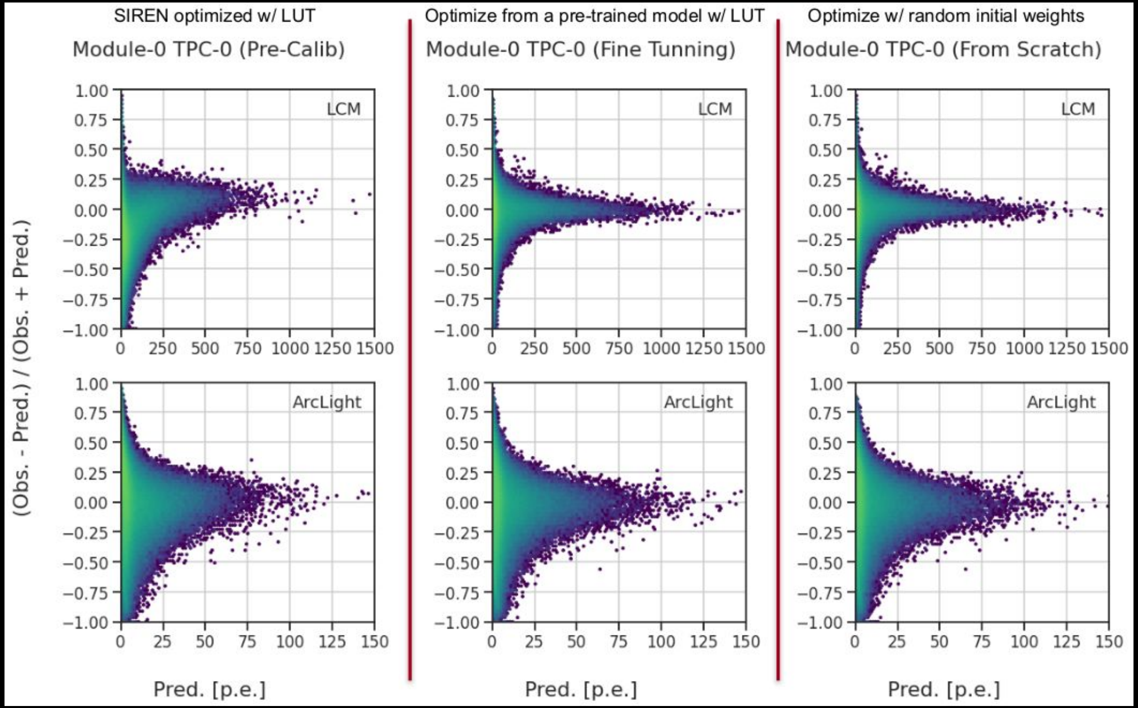
ML for Detector Physics Modeling

SIREN as a differentiable surrogate for optical detectors



Training SIREN on real data





ML for Detector Physics Modeling

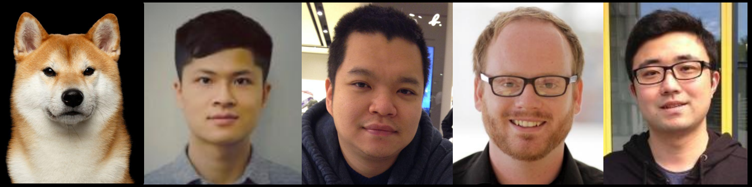
SIREN as a differentiable surrogate for optical detectors

Implicit Neural Representation as a Differentiable Surrogate for Photon Propagation in a Monolithic Neutrino Detector

Minjie Lei,[2,*] Ka Vang Tsang,[1,†] Sean Gasiorowski,[1] Chuan Li,[3] Youssef Nashed,[1] Gianluca Petrillo,[1] Olivia Piazza,[4] Daniel Ratner,[1] and Kazuhiro Terao[1]
(on behalf of the DeepLearnPhysics Collaboration)
[1]*SLAC National Accelerator Laboratory, Menlo Park, CA, 94025, USA*
[2]*Stanford University, Stanford, CA, 94305, USA*
[3]*Lambdalab Inc., San Francisco, CA, 94107, USA*
[4]*University of California, Berkeley, CA, 94720, USA*

Optical photons are used as signal in a wide variety of particle detectors. Modern neutrino experiments employ hundreds to tens of thousands of photon detectors to observe signal from millions to billions of scintillation photons produced from energy deposition of charged particles. These neutrino detectors are typically large, containing $\mathcal{O}(10^2 - 10^5)$ tons of target volume, and may consist of many materials with different optical properties. As a result, modeling individual photon propagation requires prohibitive computational resources. As an alternative to tracking individual photons, the experimental community has traditionally used a *look-up table*, which contains a mean probability of observing a photon per photon detector at each grid location in a uniformly voxelized detector volume. However, since the size of a table increases with detector volume for a fixed resolution, this method scales poorly for future larger detectors. Alternative approaches such as fitting a polynomial to the model could address the memory issue, but results in poorer performance. Furthermore, both look-up table and fitting approaches are prone to discrepancies between the detector simulation and the real-world detector response. We propose a new approach using SIREN, a implicit neural representation with periodic activation functions. In our approach, SIREN is used to model the look-up table as a "3D scene" and reproduces the acceptance map with high accuracy. The number of parameters in our SIREN model is orders of magnitude smaller than the number of voxels in the look-up table. As it models an underlying functional shape, SIREN is scalable to a larger detector. Furthermore, SIREN can successfully learn the spatial gradients of the photon library, providing additional information for downstream applications. Finally, as SIREN is a neural network representation, it is differentiable with respect to its parameters, and therefore tunable via gradient descent. We demonstrate the potential of optimizing SIREN directly on real data, which mitigates the concern of data vs. simulation discrepancies. We further present an application for data reconstruction where SIREN is used to form a likelihood function for photon statistics.

Preprint [arXiv:2210.01505](https://arxiv.org/abs/2210.01505)



Work credit (from left): Olivia P. (UC Berkeley), Minjie L. (SLAC), Patrick T. (SLAC), , Gordon W. (Stanford CS), Chuan L. (Lambda Labs)

Drift of Ionization Electrons for Imaging

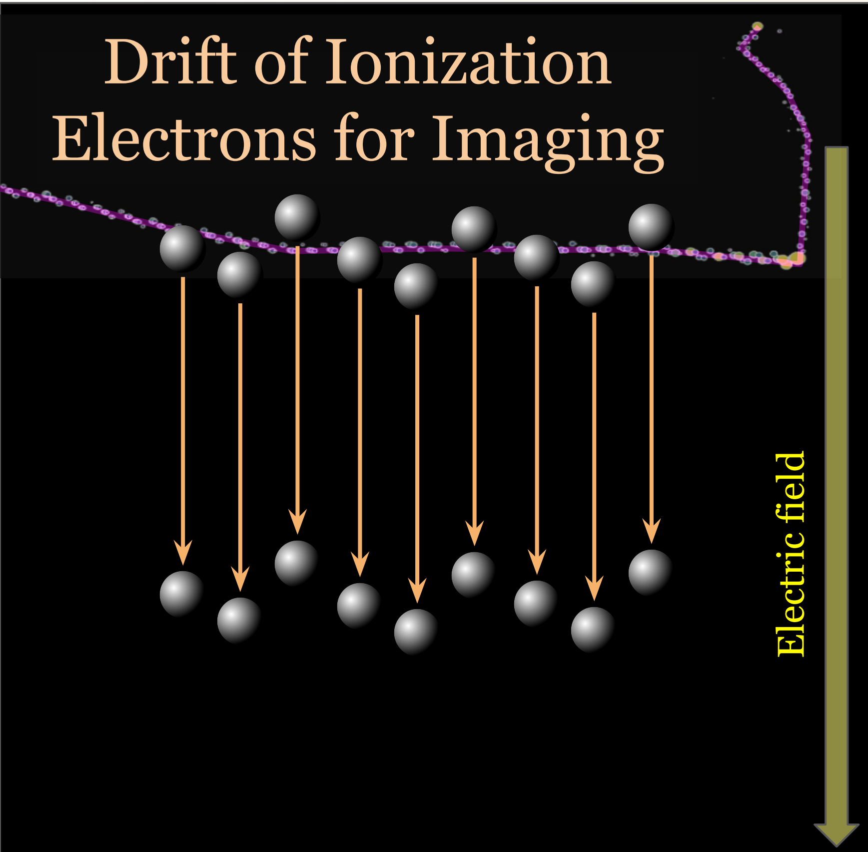
TPC Imaging Detector Simulation

Drift of Ionization Electrons for Imaging

Simulation steps:

1. Ionization of LAr from dE/dX

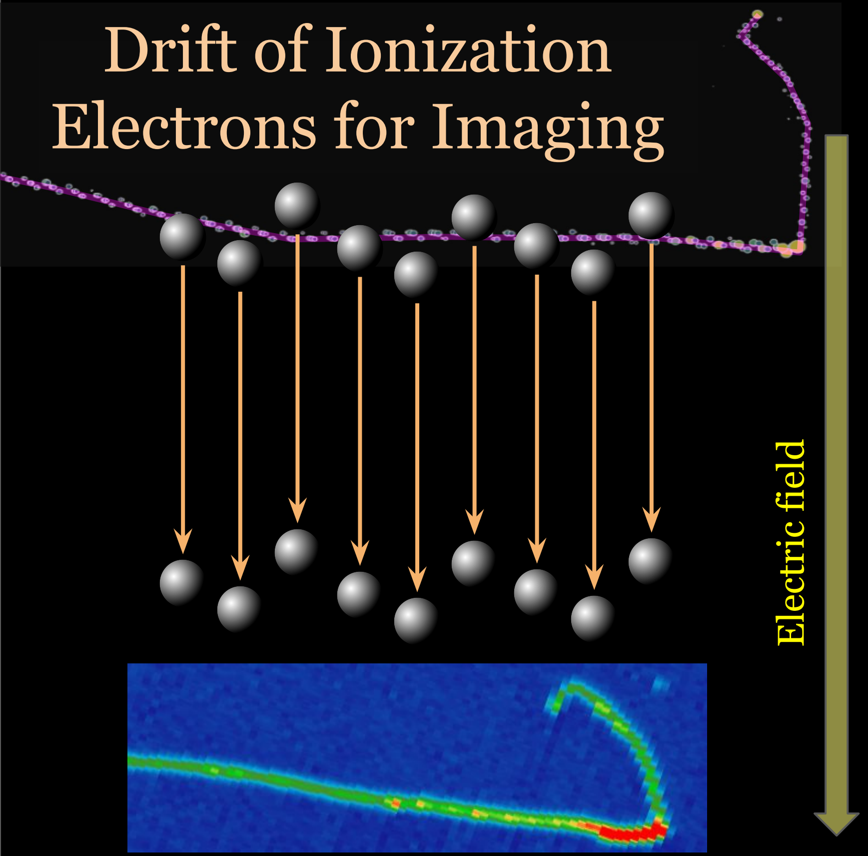
TPC Imaging Detector Simulation



Simulation steps:

1. Ionization of LAr from dE/dX
2. Ionization electron drift and diffuse in E-field at a “constant” velocity, some charge lost due to capture

ML for Detector Physics Modeling
TPC Imaging Detector Simulation



Simulation steps:

1. Ionization of LAr from dE/dX
2. Ionization electron drift and diffuse in E-field at a “constant” velocity, some charge lost due to capture
3. Imaging by charge-sensitive plane (detectors) at the anode

ML for Detector Physics Modeling

AD-enabled differentiable detector simulator

Drift of Ionization Electrons for Imaging

Detector Simulation

$$\mathscr{R}_{\mathrm{ICARUS}} = \frac{A_B}{1 + k_B \cdot (dE/dx)/\mathscr{E}}$$

$$Q = Q_0 \exp(-v_{\mathrm{drift}} t/\tau)$$

$$\sigma_t^2(t) \simeq \sigma_t^2(0) + \left(\frac{2D_L}{v_d^2}\right) t$$

…

PyTorch

Simulation steps:

1. Ionization of LAr from dE/dX
2. Ionization electron drift and diffuse in E-field at a “constant” velocity, some charge lost due to capture
3. Imaging by charge-sensitive plane (detectors) at the anode

A composite of a simple set of functions, and it’s parallelizable for many segments…

Differentiable programming FMWKs?

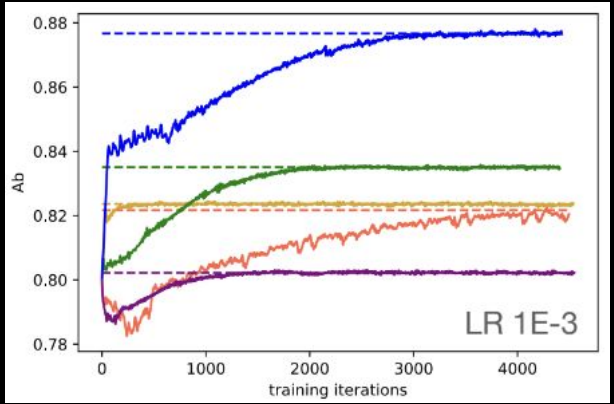
ML for Detector Physics Modeling
AD-enabled differentiable detector simulator

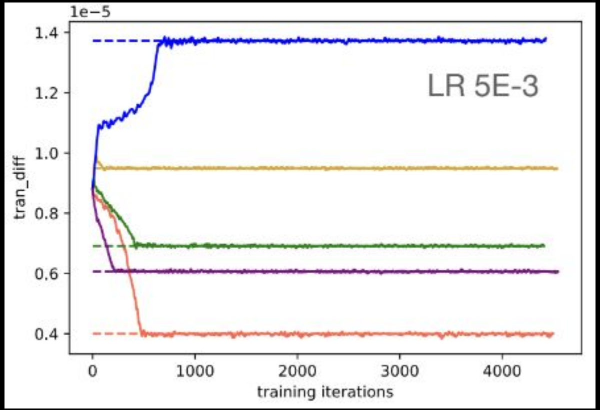
SLAC

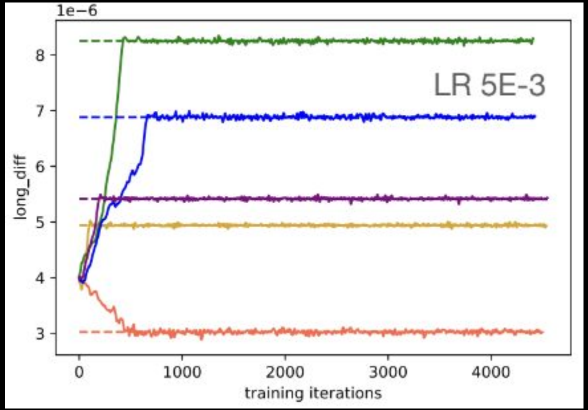
Optimization of TPC detector response

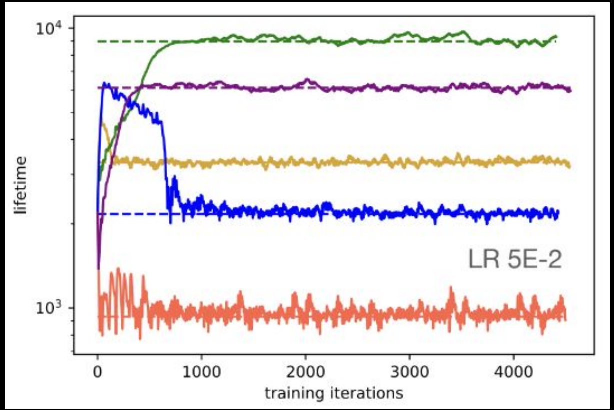
Much work in progress(!) = take it as a grain of salt

- Use contained proton tracks and MIP muons (true dE/dX can be well characterized)
- Simultaneous optimization of detector simulation parameters to minimize data/simulation shift









ML for Detector Physics Modeling
AD-enabled differentiable detector simulator

Optimization of TPC detector response

Much work in progress(!) = take it as a grain of salt

- Use contained proton tracks and MIP muons (true dE/dX can be well characterized)
- Simultaneous optimization of detector simulation parameters to minimize data/simulation shift

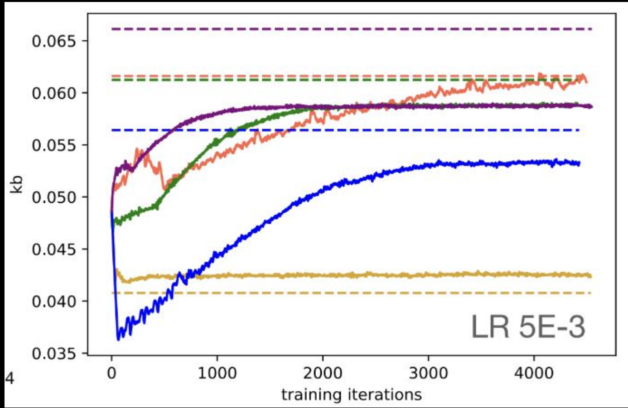
Detector Simulation

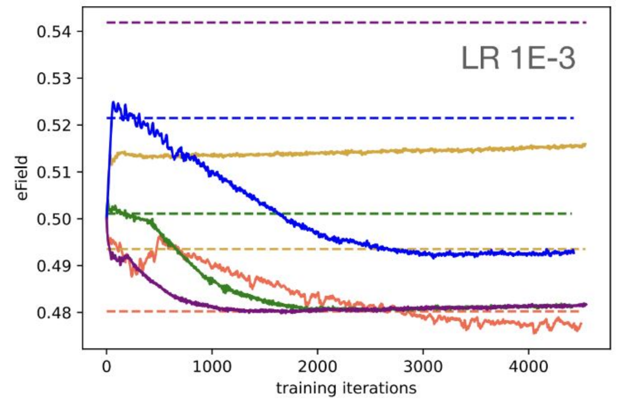
$$\boxed{\mathscr{R}_{\text{ICARUS}} = \frac{A_B}{1 + k_B \cdot (dE/dx)/\mathscr{E}}}$$

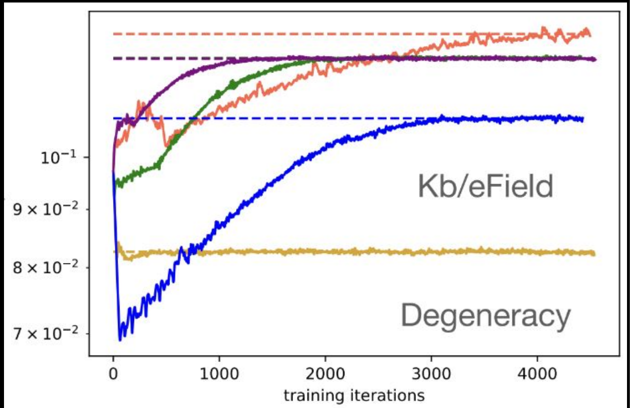
$$Q = Q_0 \exp(-v_{\text{drift}} t/\tau)$$

$$\sigma_t^2(t) \simeq \sigma_t^2(0) + \left(\frac{2D_L}{v_d^2}\right) t$$

...







Inverse imaging using a differentiable simulator

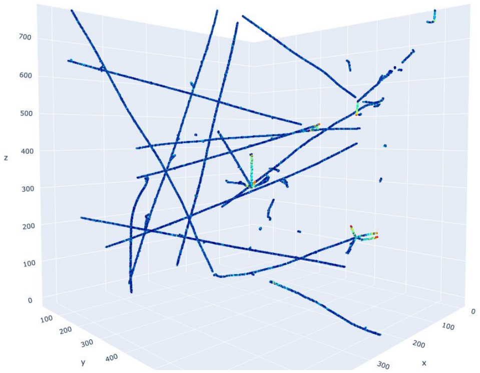
SLAC

E.g. use for optimizing an image inverse solver

$G\,(X|Y, \theta_G)$

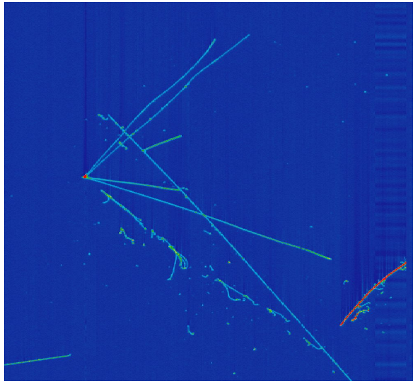
Inverse Image Solver

$$\boxed{\mathcal{L}_{\text{inv}} = |G(\mathbf{Y}) - \mathbf{X}|^2}$$



$\mathbf{X} \in \mathcal{D}_I$

Input domain of LArTPC simulator (inaccessible)



$\mathbf{Y} \in \mathcal{D}_O$

Output domain of LArTPC simulator (e.g. real data)

Inverse imaging using a differentiable simulator

SLAC

E.g. use for optimizing an image inverse solver

$G\ (X|Y, \theta_G)$

Inverse Image Solver

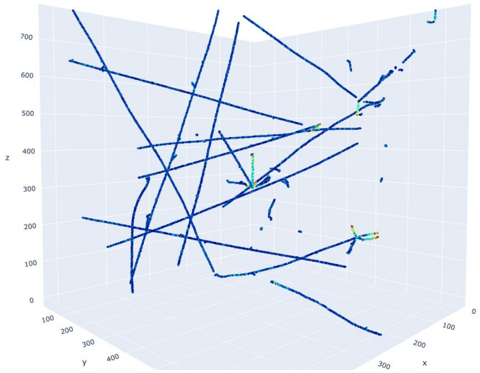
$$\mathcal{L}_{\text{inv}} = |G(\mathbf{Y}) - \mathbf{X}|^2$$

and / or

$$\mathcal{L}_{\text{cc}} = |F(G(\mathbf{Y})) - \mathbf{Y}|^2$$

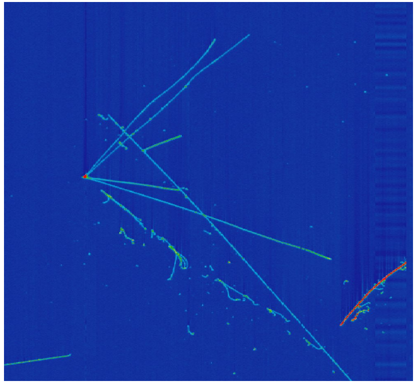
$F\ (Y|X, \theta_F)$

Differentiable LArTPC Simulator



$\mathbf{X} \in \mathcal{D}_I$

Input domain of LArTPC simulator (inaccessible)

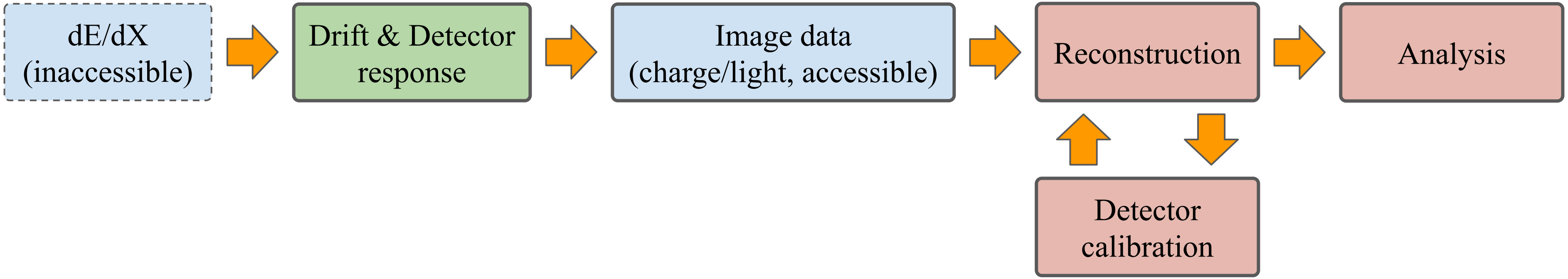


$\mathbf{Y} \in \mathcal{D}_O$

Output domain of LArTPC simulator (e.g. real data)

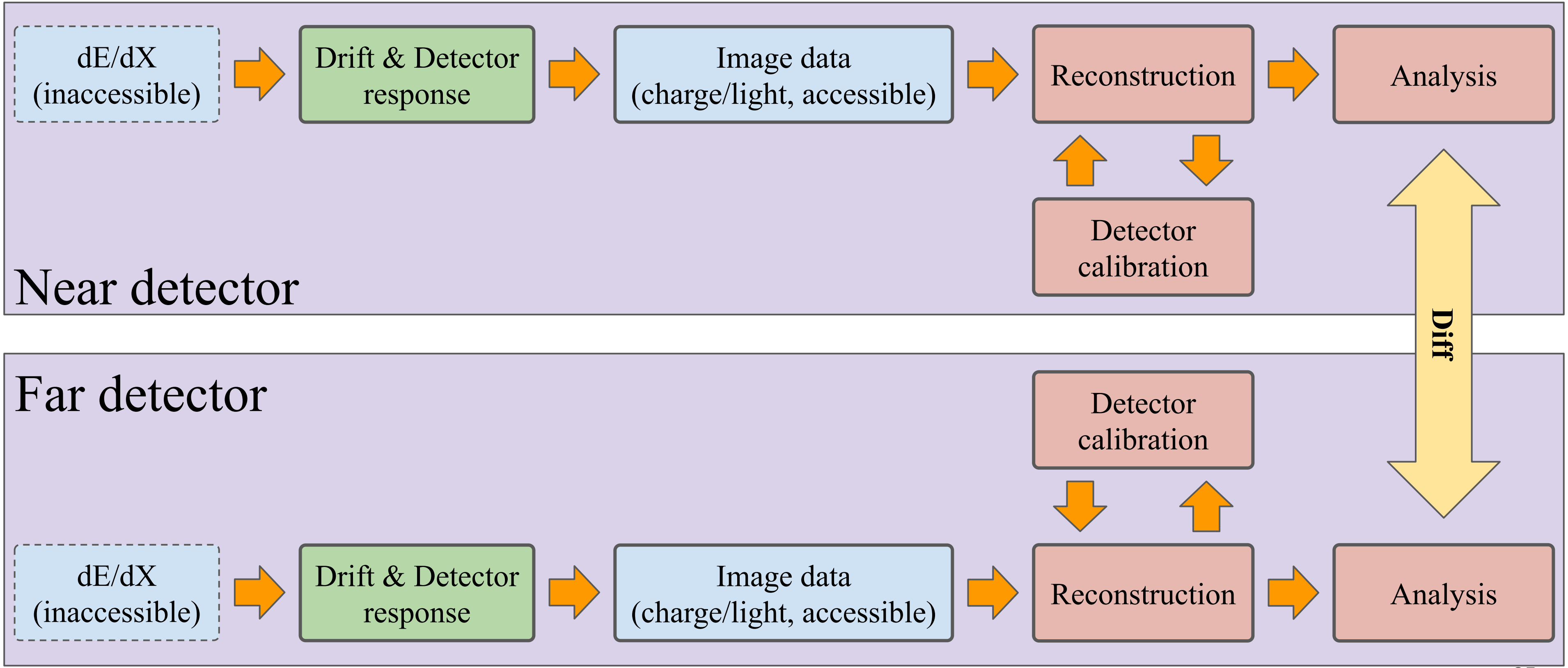
ML for Analyzing Big Image Data in Neutrino Experiments

Inverse imaging using a differentiable simulator

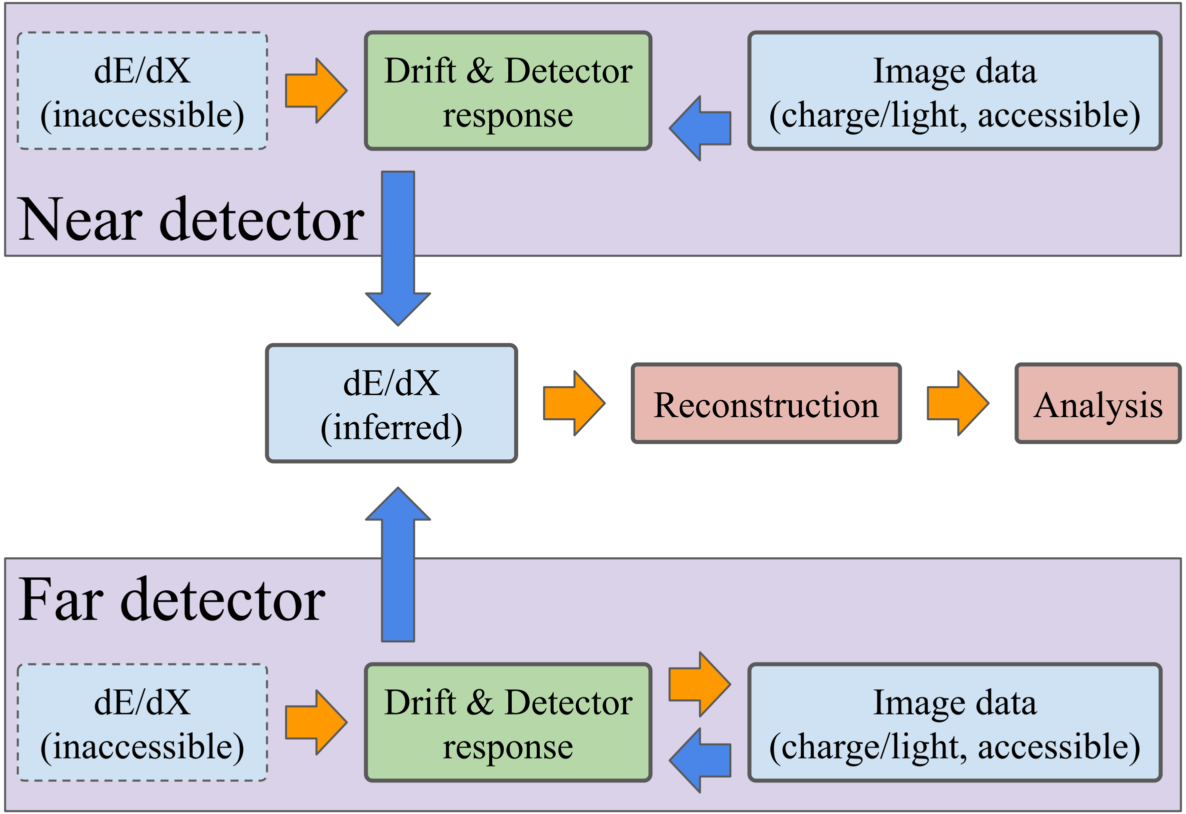


ML for Analyzing Big Image Data in Neutrino Experiments

Inverse imaging using a differentiable simulator



Inverse imaging using a differentiable simulator



Detector calibration can be automated

Reconstruction can be shared across detectors.

… wrapping up …

Wrapping-Up

AI/ML applications expanding in neutrino exp.!

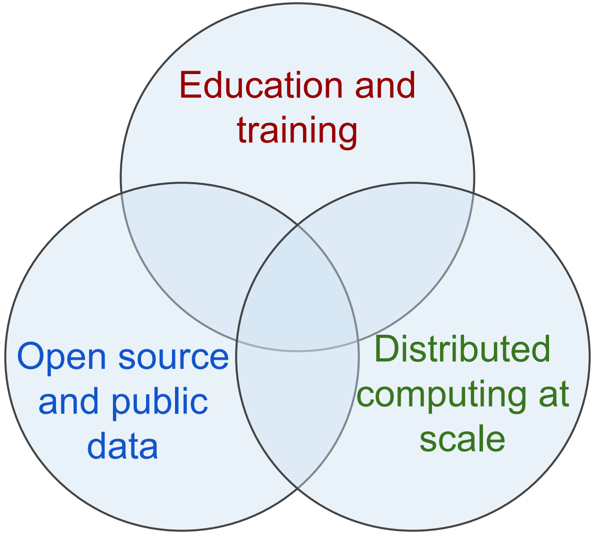
- End-to-end optimizable data reconstruction chain
- Differentiable simulator for detector physics model optimization
- **Exciting next stage**: inverse imaging and a full workflow automation

Topics not covered but I work on (let's discuss!):

- Uncertainty quantification for ML methods
- Foundation models for physics toward general AI R&D
  - Something much better than our "end-to-end" method will come out!

Cross-domain HEP AI ecosystem

ML is a "solution pattern" v.s. a domain-specific "hard-coded" solution. It's **naturally reusable across domains including software tools** supported by a large community of researchers.



HEP Ecosystem for AI research

- Accessible **education and training** at all levels
- **Reusable software tools** to unlock modern compute accelerators and networking (distributed ML)
- **Public datasets** with documentation and performance metrics for transparent, reproducible science
- Artificial Intelligence and Technology Office (AITO)
  - Federated, equitable, responsible, trustworthy AI
  - **AI is an accelerator**. It is coming. Don't avoid. **Participate to make sure the use is good.**