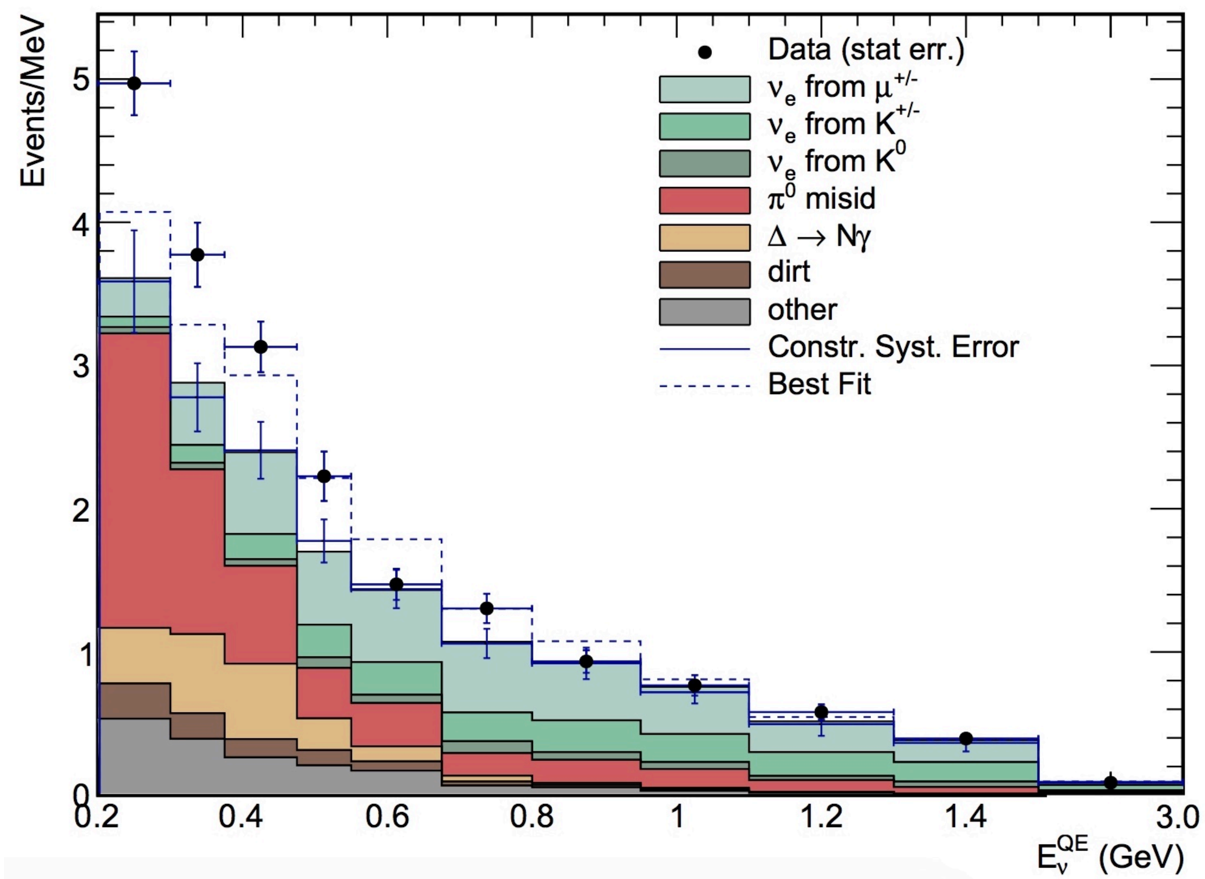
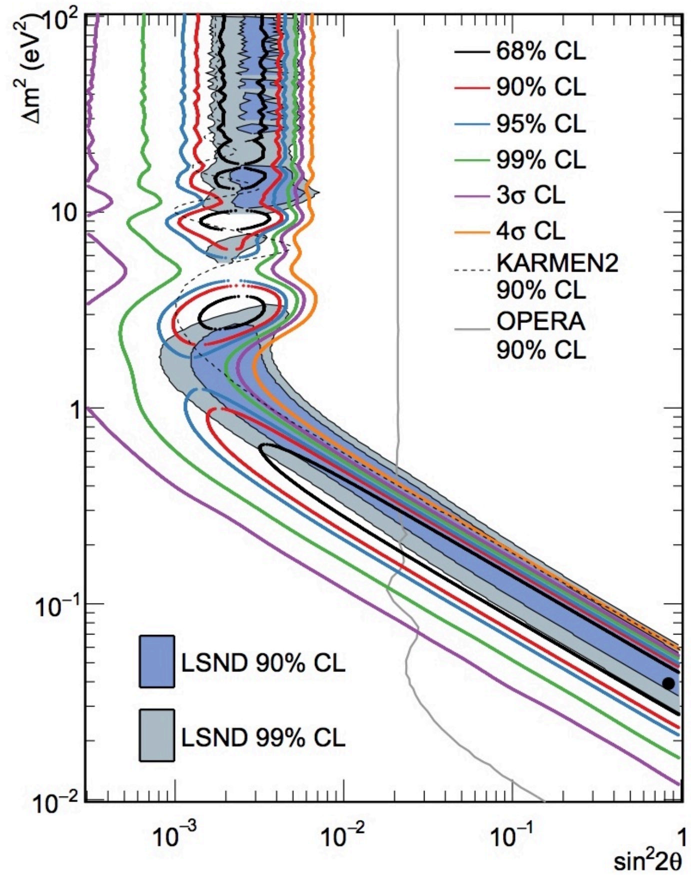
MicroBooNE Search for Low-Energy Excess Using Deep Learning Algorithms

Lauren Yates
Massachusetts Institute of Technology
On Behalf of the MicroBooNE Collaboration

NuFACT 2018

MiniBooNE Low-Energy Excess



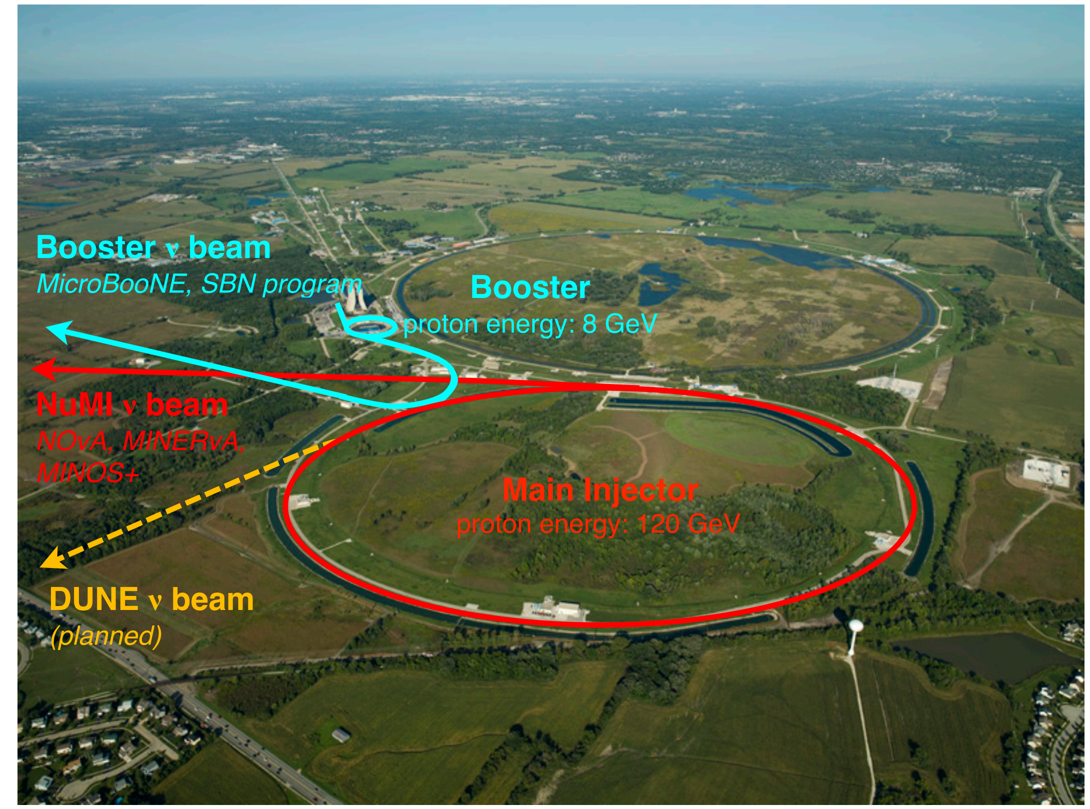


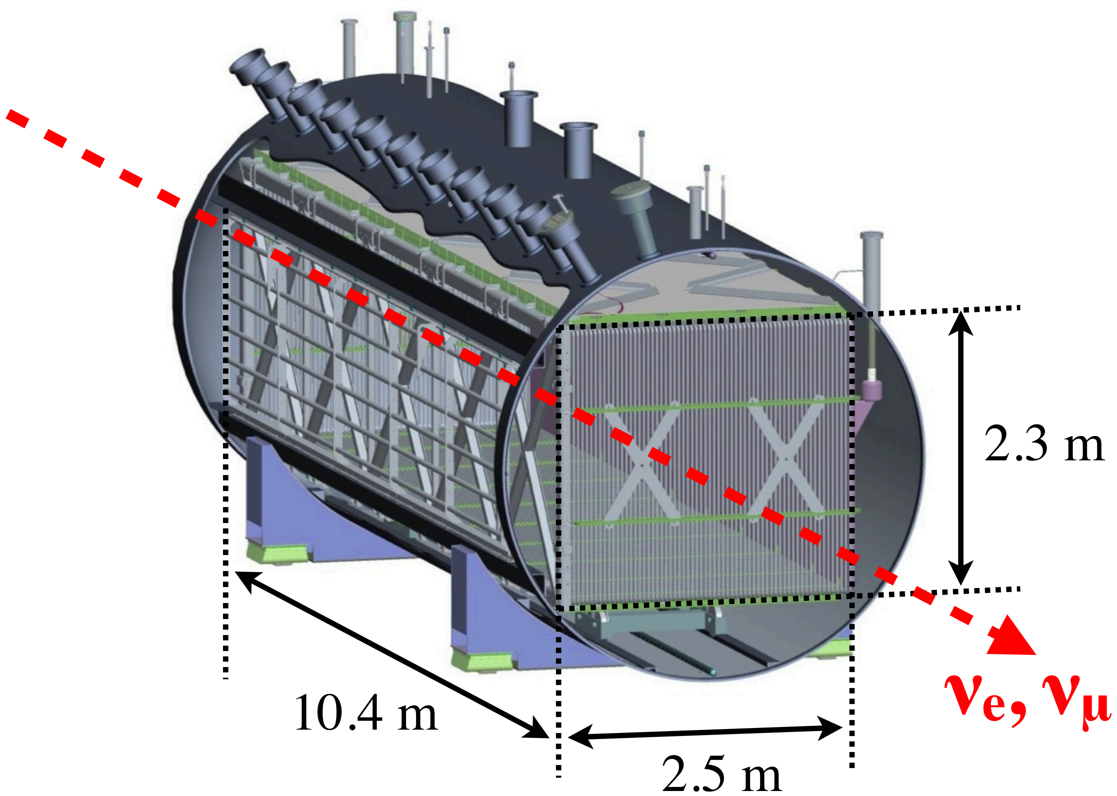
arXiv:1805.12028 [hep-ex]

- MiniBooNE sees a 4.5σ ν_e-like excess
- This result is in tension with global 3+1 model fit

- MiniBooNE
  - Mineral oil Cherenkov detector
  - Significant fraction of the background from γ/e^- mis-ID
- MicroBooNE
  - Same beam and similar baseline
  - LArTPC detector technology gives better γ/e^- separation power

The MicroBooNE Experiment

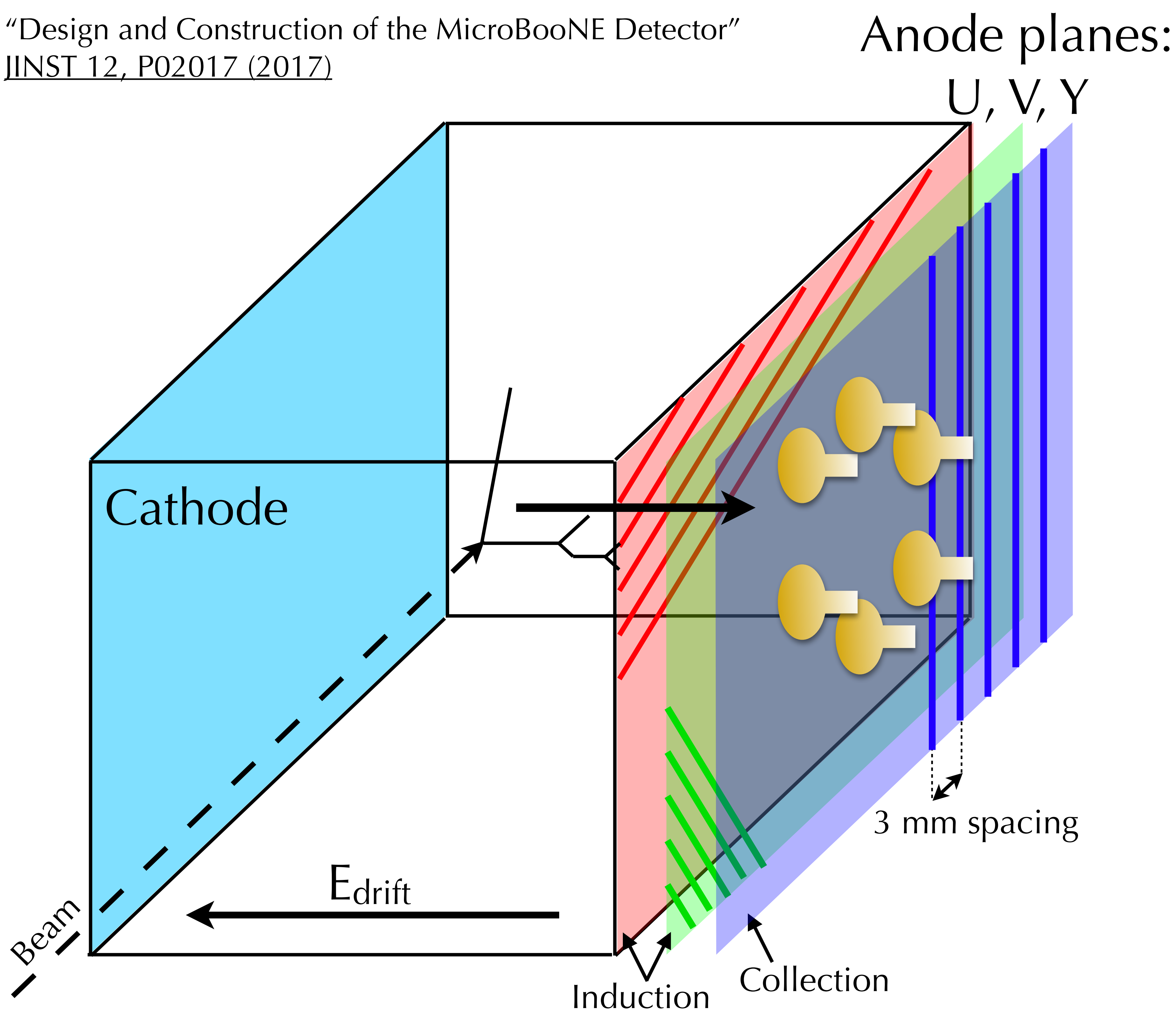


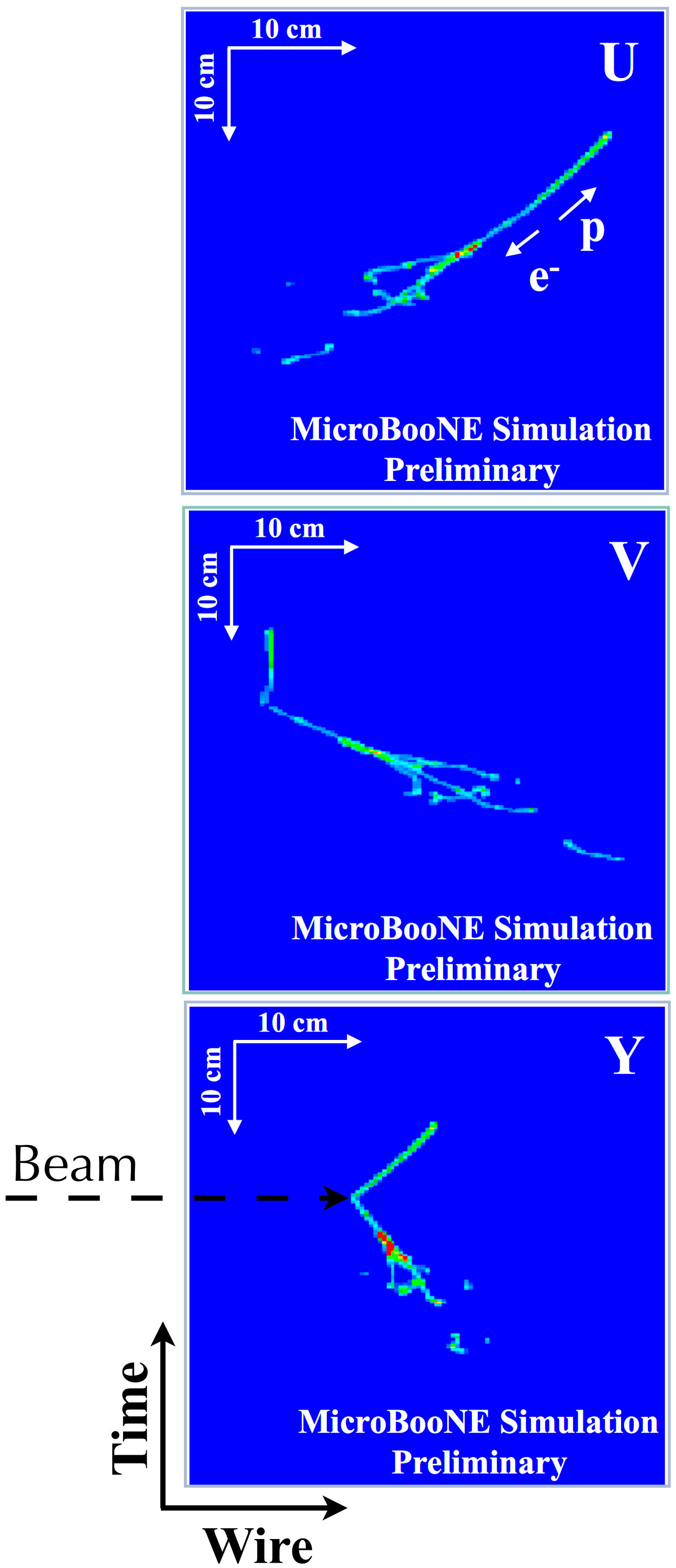


- **Micro Boo**ster **N**eutrino **E**xperiment
- 85 tonne **L**iquid **Ar**gon **T**ime **P**rojection **C**hamber (active mass)
- Located in the Fermilab **B**ooster **N**eutrino **B**eam
- $\nu_\mu \rightarrow \nu_e$ appearance experiment
- >95% detector uptime
- 9.6×10^{20} POT on tape to date

The MicroBooNE Detector

"Design and Construction of the MicroBooNE Detector"
JINST 12, P02017 (2017)



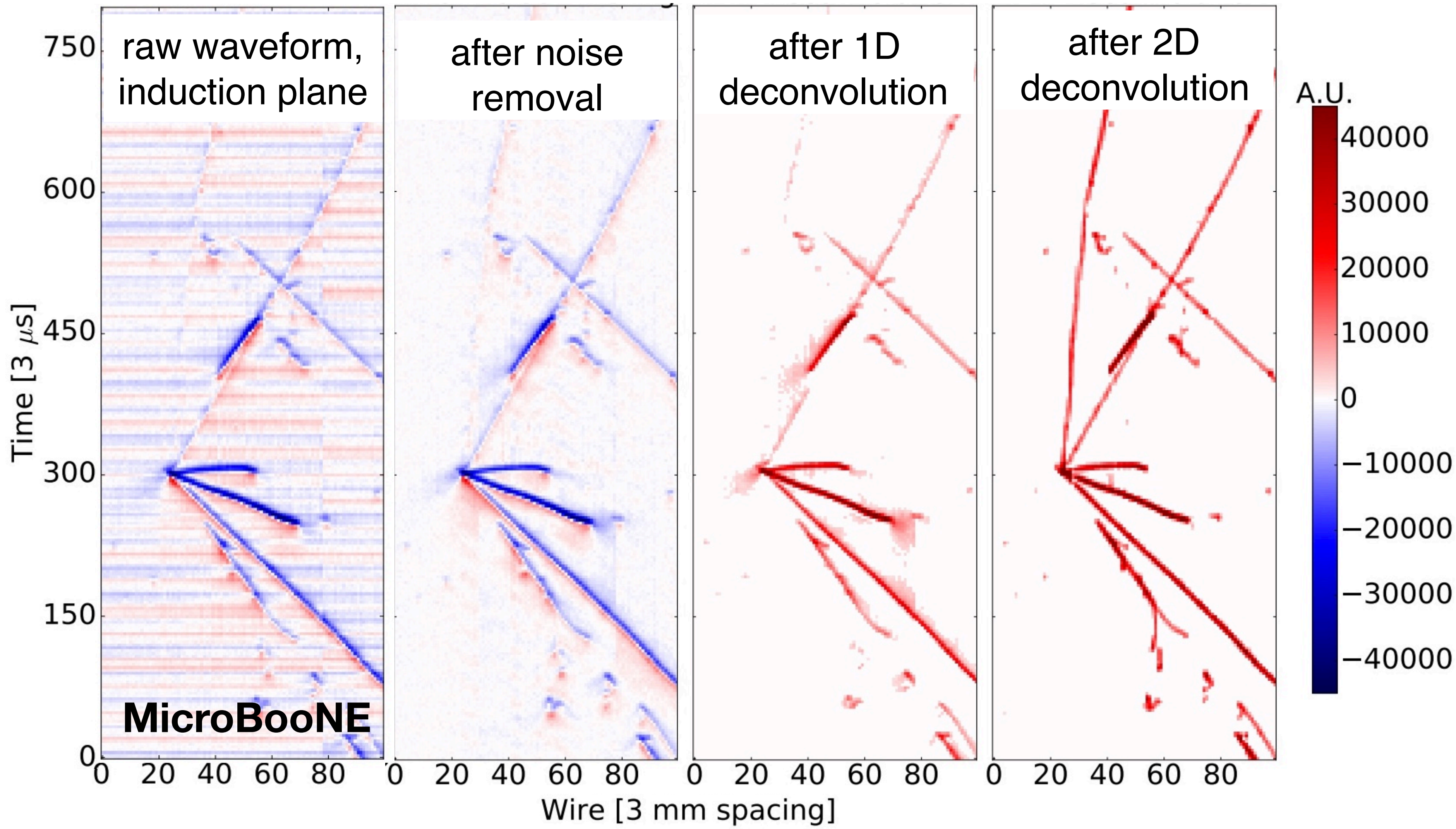


Understanding the Detector

- Detailed understanding of our detector is key to our physics goals and to R&D efforts for future LArTPC detectors
- Developed novel techniques for noise filtering and signal processing

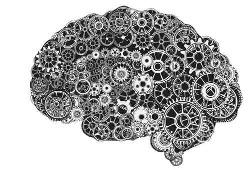
"Noise Characterization and Filtering in the MicroBooNE Liquid Argon TPC", JINST 12, P08003 (2017)

"Ionization Electron Signal Processing in Single Phase LArTPCs" Parts I & II, JINST 13, P07006 (2018) & JINST 13, P07007 (2018)

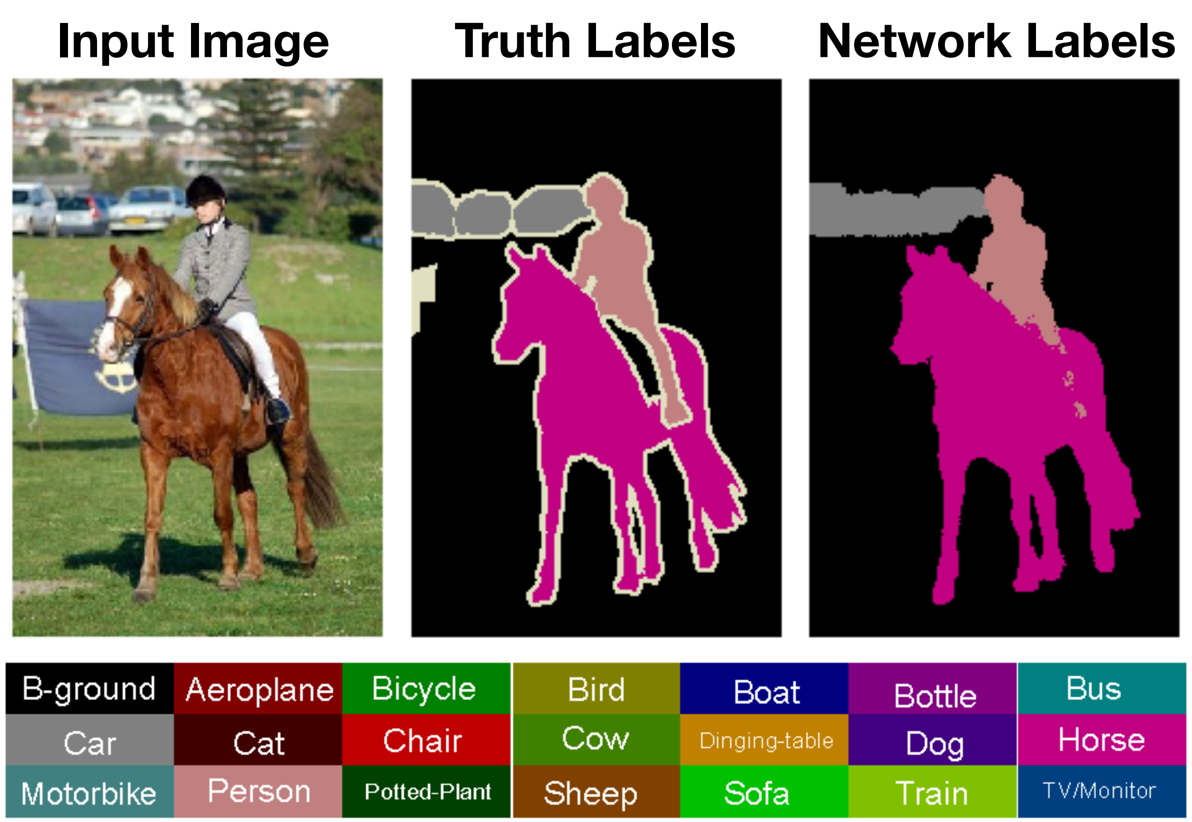


A Few Words About Deep Learning

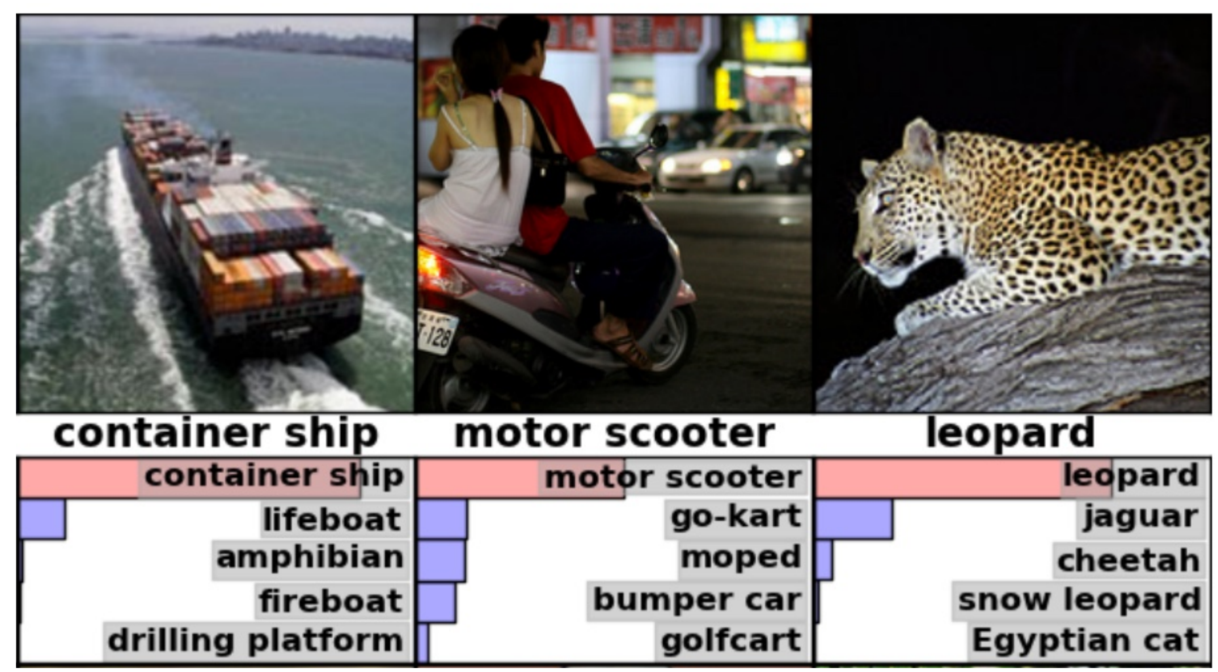
µBooNE



- Primarily use convolutional neural networks (CNNs)
- CNNs have been developed primarily for image analysis; we apply them to MicroBooNE event displays
- I will discuss two uses: semantic segmentation and classification



Example of semantic segmentation, from "Conditional Random Fields as Recurrent NNs", ICCV (2015)



Example of CNN classification, from "ImageNet Classification with Deep CNNs", NIPS (2012)

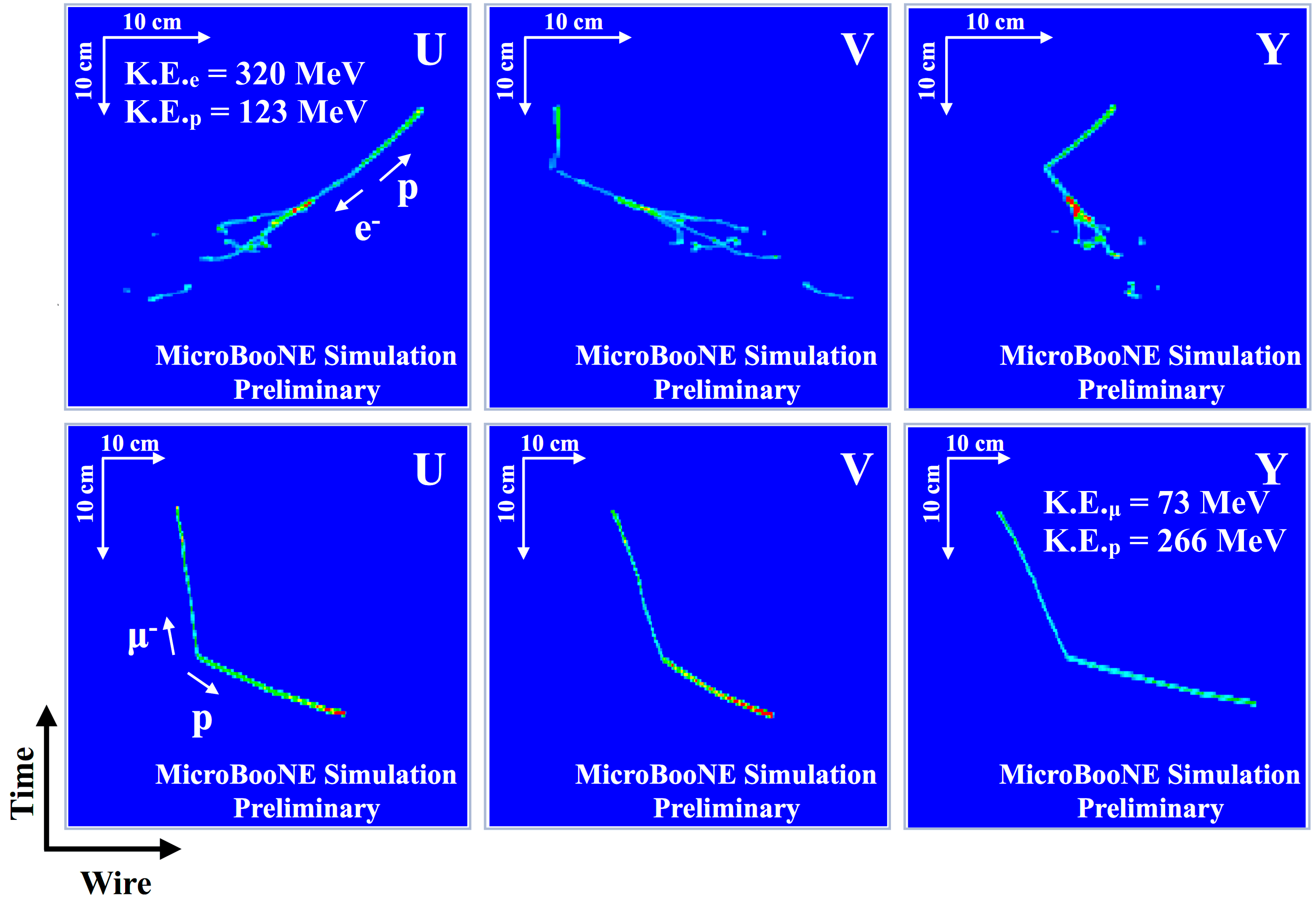
Definition of the Signal

μBooNE

- Define our signal to be charged current quasi-elastic events with one lepton and one proton (1l-1p) topology
  - Lepton (electron or muon) with kinetic energy >35 MeV
  - One proton with kinetic energy >60 MeV (possibly others below that energy threshold)
- Intrinsic ν_e backgrounds are constrained by ν_μ events

ν_e event: signal

ν_μ event: used to constrain the flux and cross-section systematics

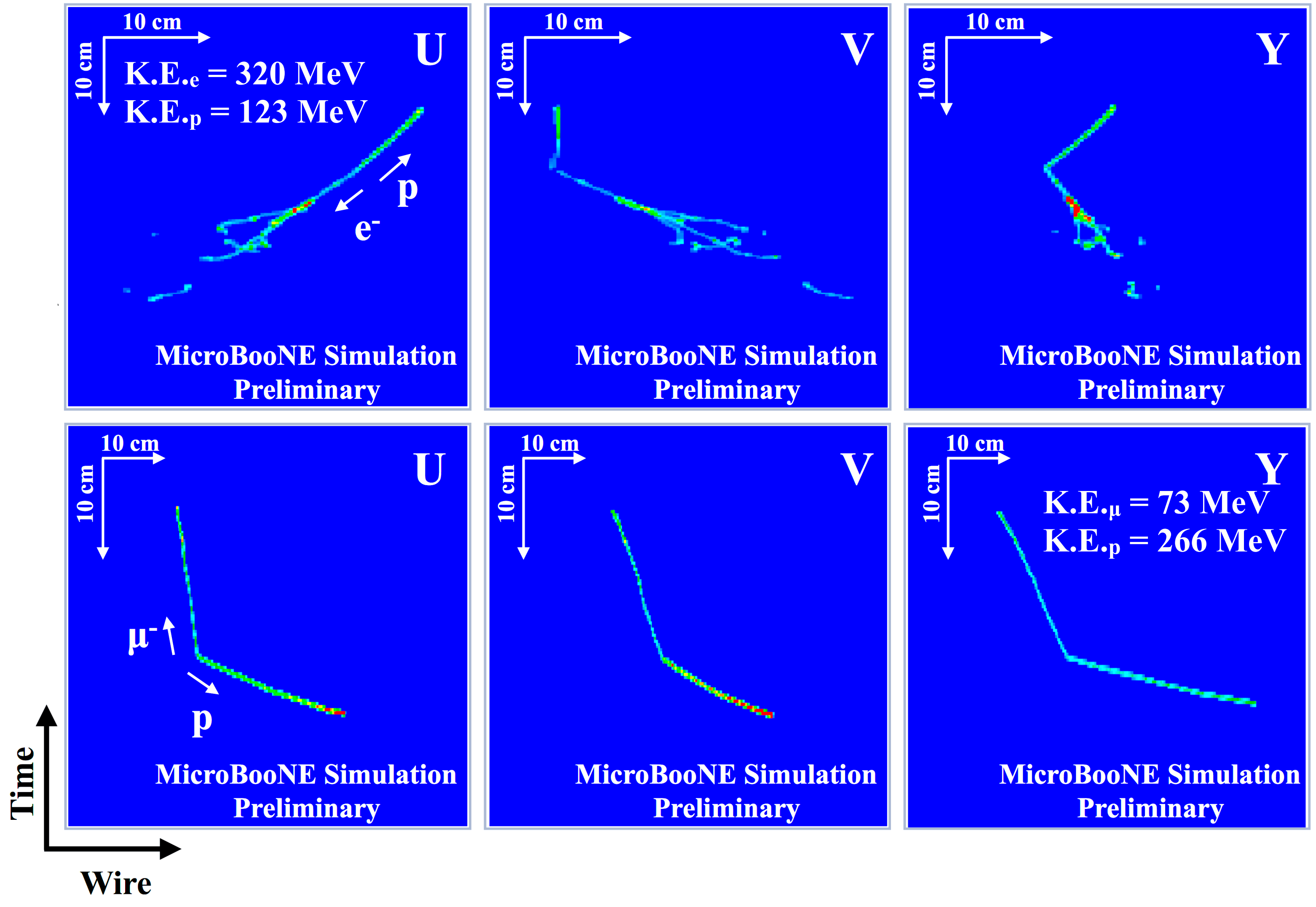


Definition of the Signal

- We are able to observe many event topologies in MicroBooNE, but we choose this relatively simple one
- Requiring a proton in the event reduces backgrounds from cosmic rays and single-photon events

ν_e event: signal

ν_μ event: used to constrain the flux and cross-section systematics



Overview of Reconstruction Chain

μBooNE

PMT Pre-Cuts

↓

Cosmic Tagging & ROI Finding

↓

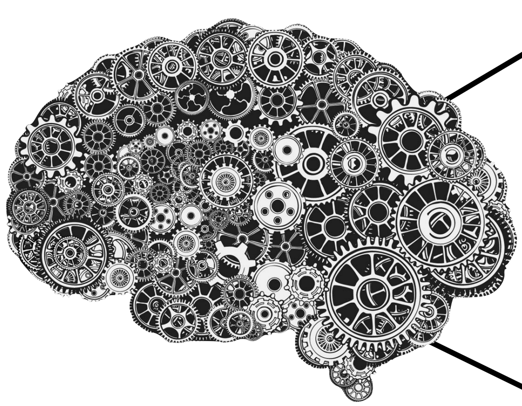
Track vs. Shower Pixel Labeling

↓

3D Reconstruction

↓

Particle ID



Reconstruction Chain

µBooNE

PMT Pre-Cuts

↓

Cosmic Tagging & ROI Finding

↓

Track vs. Shower Pixel Labeling

↓

3D Reconstruction

↓

Particle ID

- PMT pre-cuts reject low-energy noise and other backgrounds
- Keep >97% of neutrinos (based on simulations)
- Reject >75% of background (based off-beam data)

Reconstruction Chain

μBooNE

PMT Pre-Cuts

↓

Cosmic Tagging & ROI Finding

↓

Track vs. Shower Pixel Labeling

↓

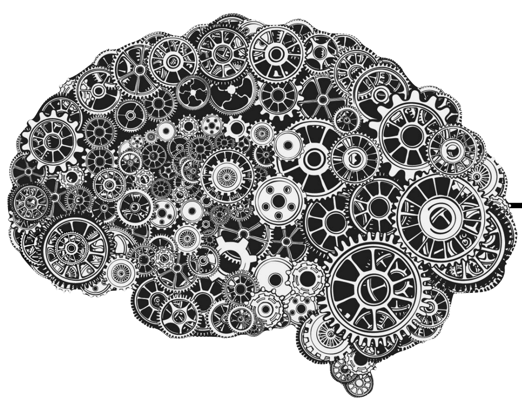
3D Reconstruction

↓

Particle ID

- Tag tracks that cross the TPC boundary
- Identify clusters of contained charge consistent with the in-time flash

Reconstruction Chain



PMT Pre-Cuts

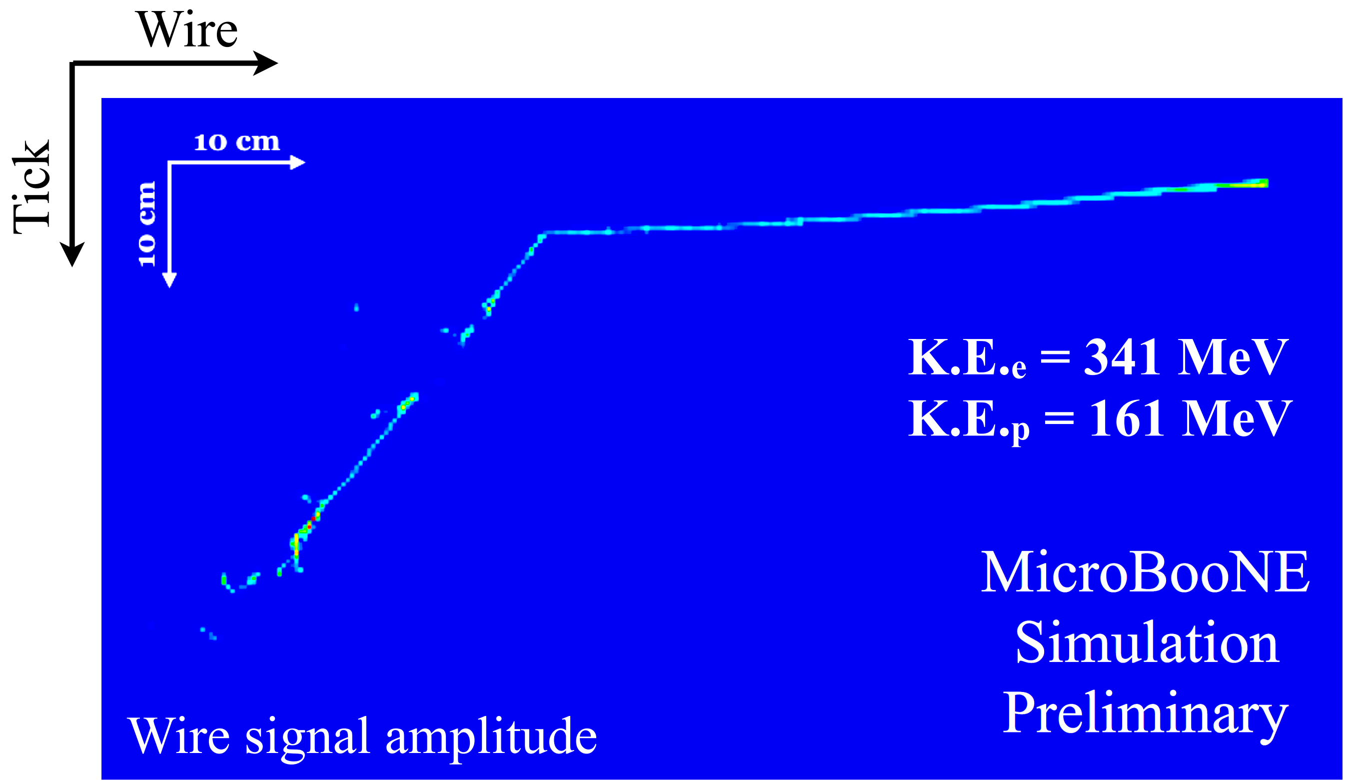
Cosmic Tagging & ROI Finding

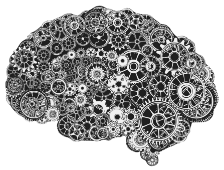
Track vs. Shower Pixel Labeling

3D Reconstruction

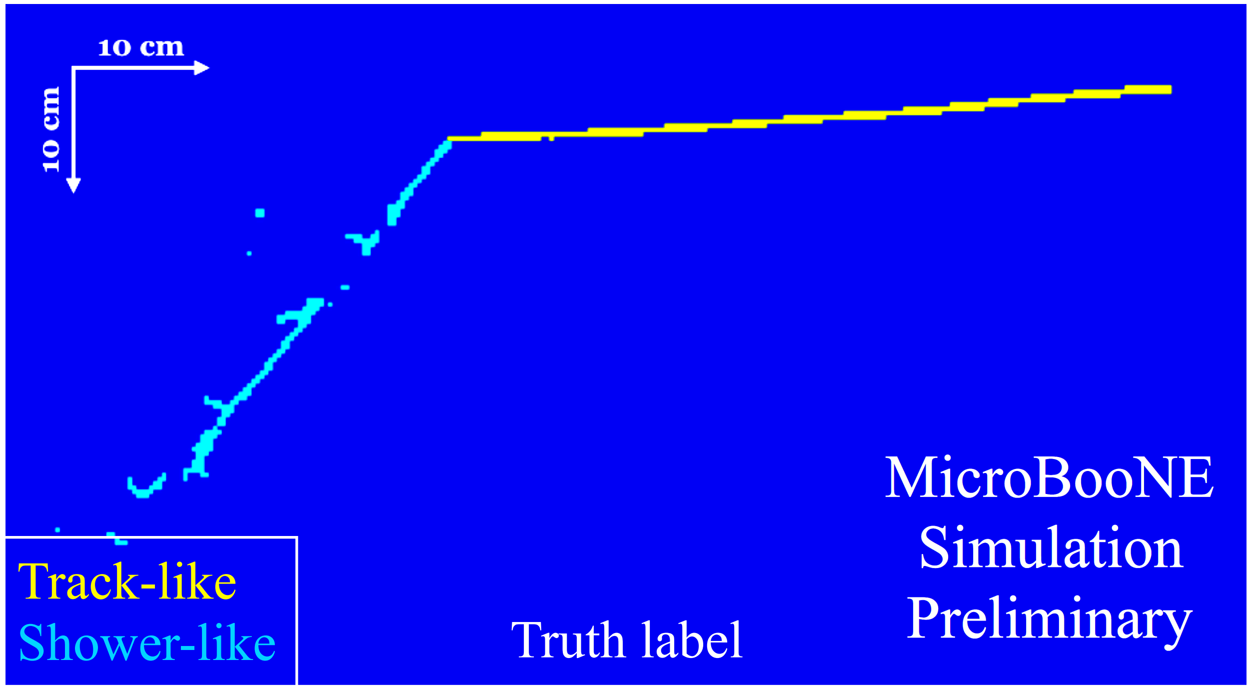
Particle ID

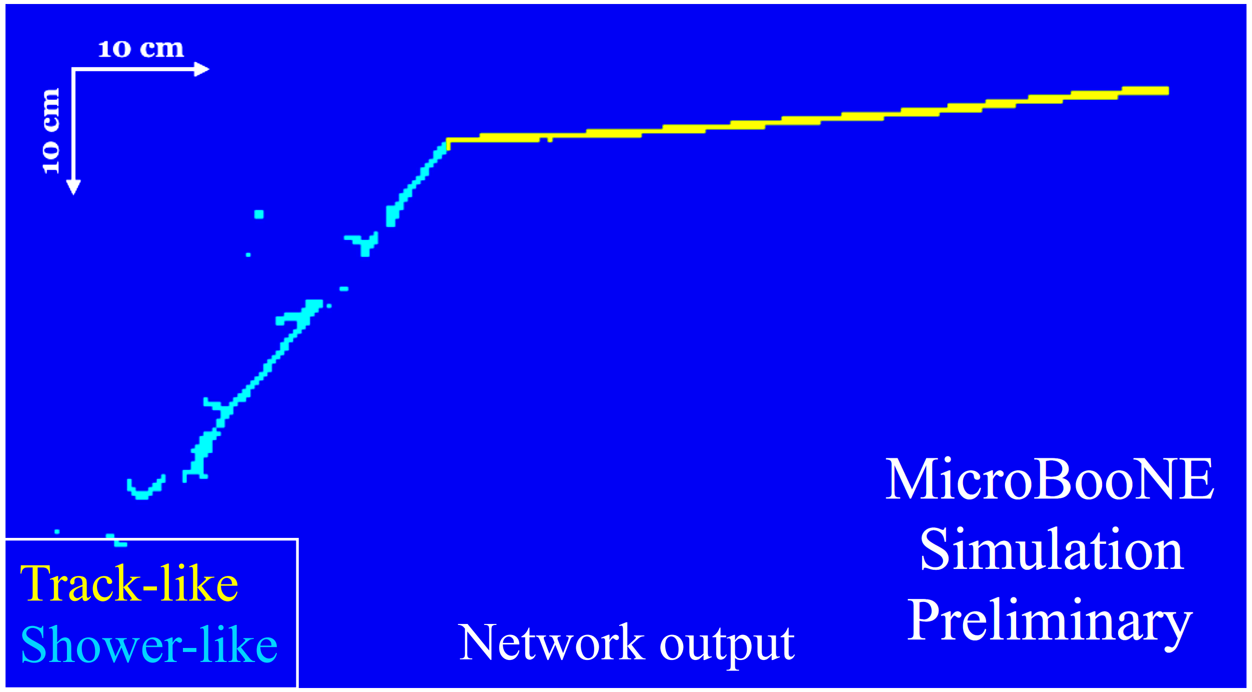
Track vs. Shower Pixel Labeling





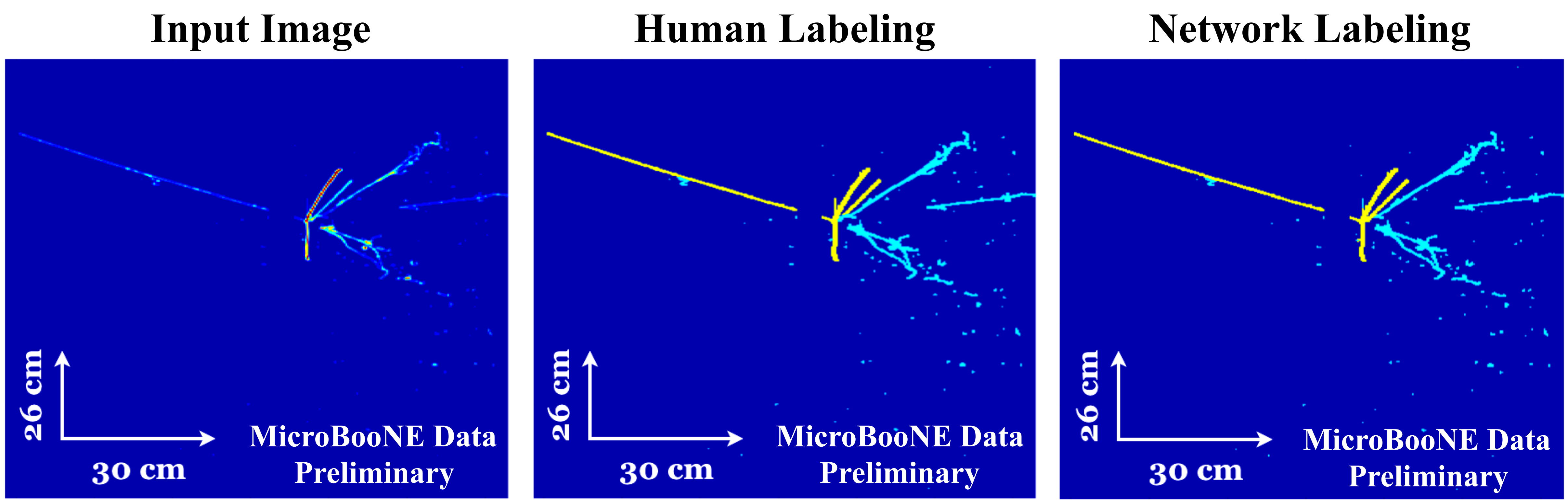
- Goal: separate tracks and showers to help provide vertex candidates
- Semantic segmentation network takes in the wire information and labels each pixel in the image as "track-like" (yellow) or "shower-like" (cyan)





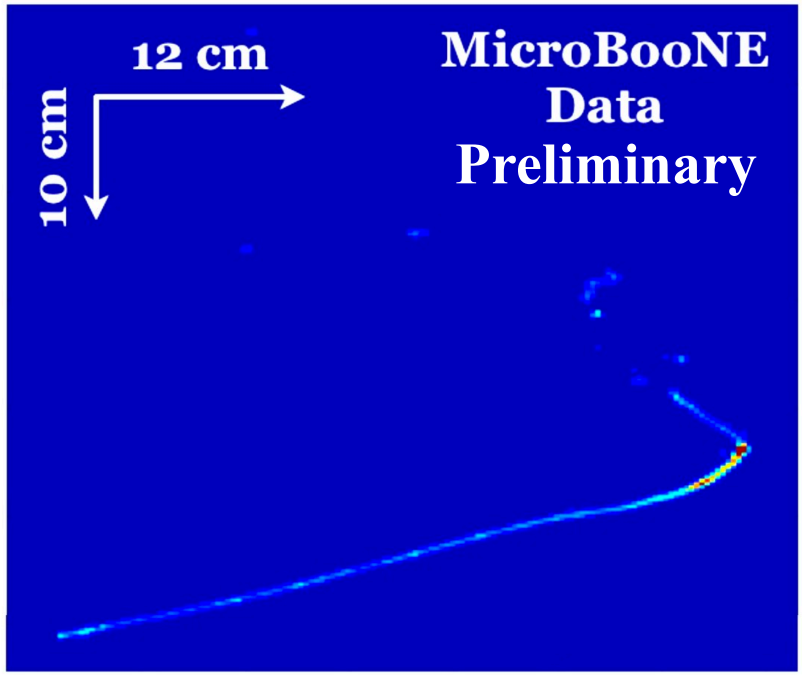
Performance on Data

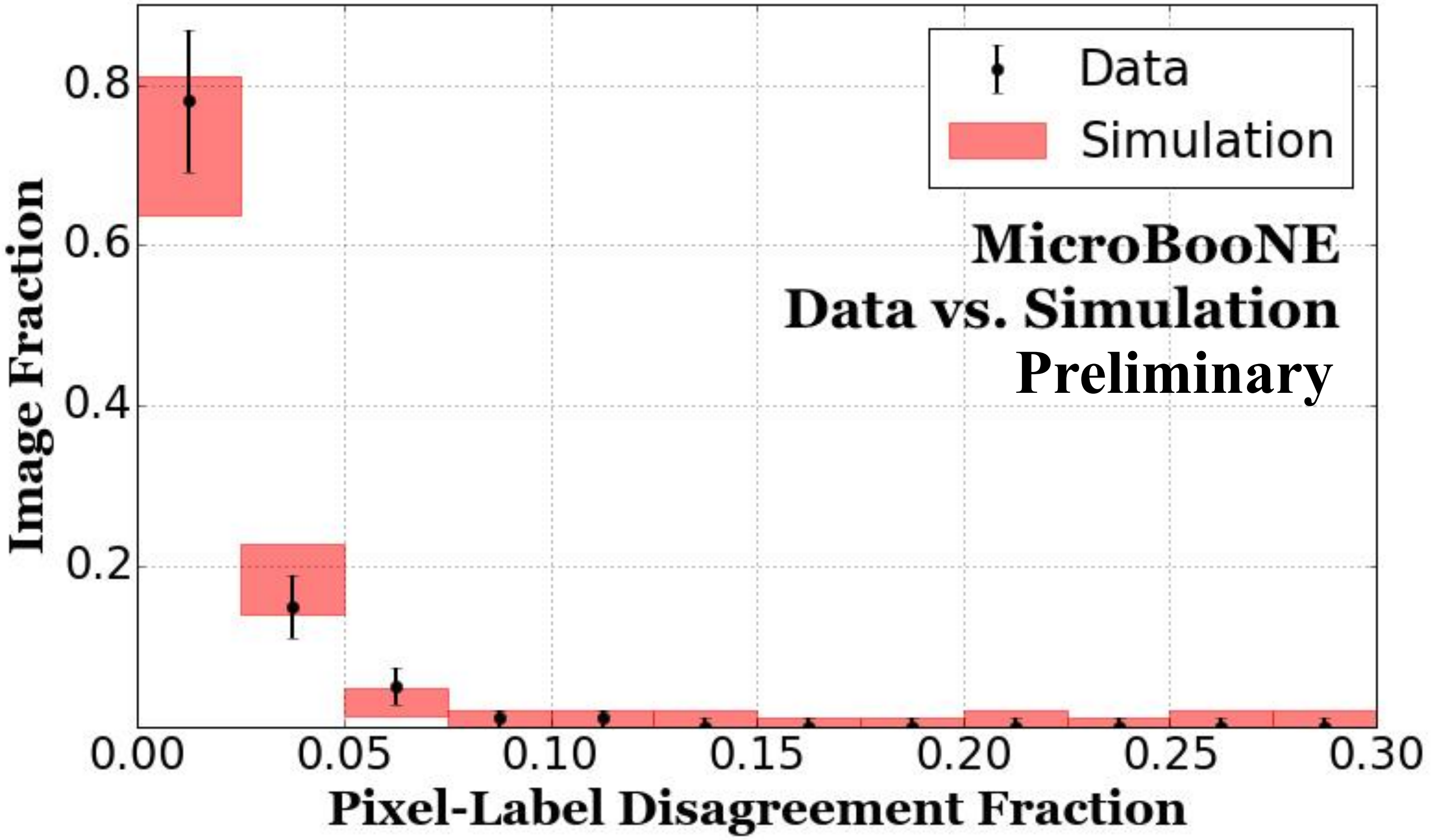
- Network shows very good performance on data, in spite of being trained on simulation
- Example: ν_μ charged current π^0 event
  - Outgoing muon and hadrons identified as track-like (yellow)
  - Showers resulting from π^0 decay identified as shower-like (cyan)



Performance on Data

- To quantify network's performance, look at level of disagreement in pixel labels between human and network over many images
  - In this case, looking at Michel electron events
- Disagreement is generally below 2.5% of non-empty pixels
- Level of agreement is consistent between data and simulation





Reconstruction Chain

PMT Pre-Cuts

↓

Cosmic Tagging & ROI Finding

↓

Track vs. Shower Pixel Labeling

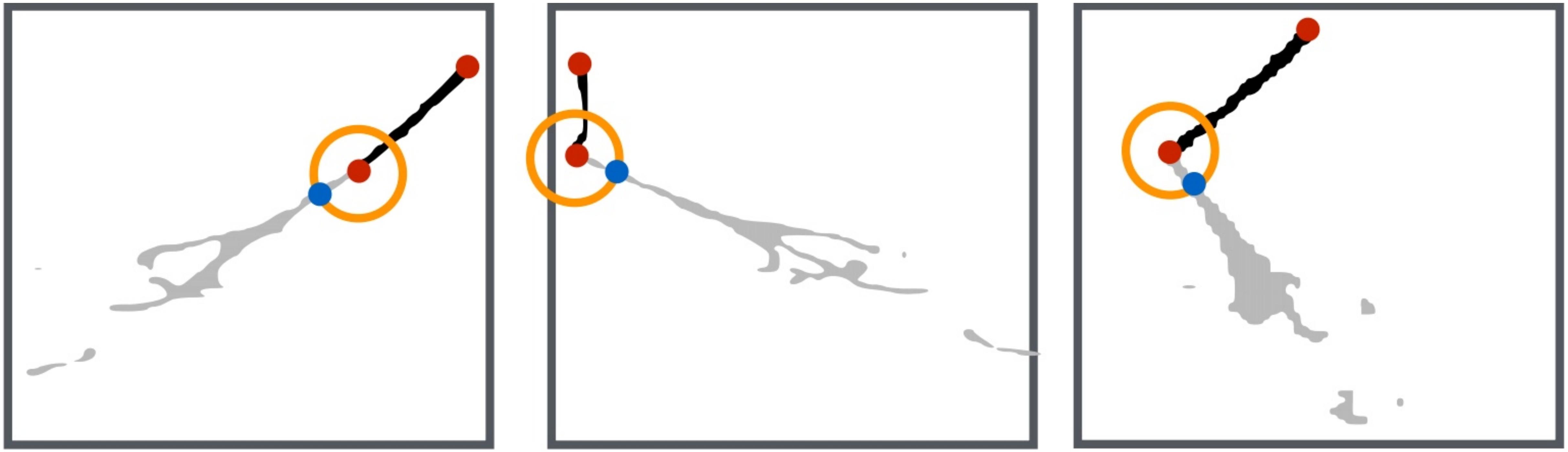
↓

3D Reconstruction

↓

Particle ID

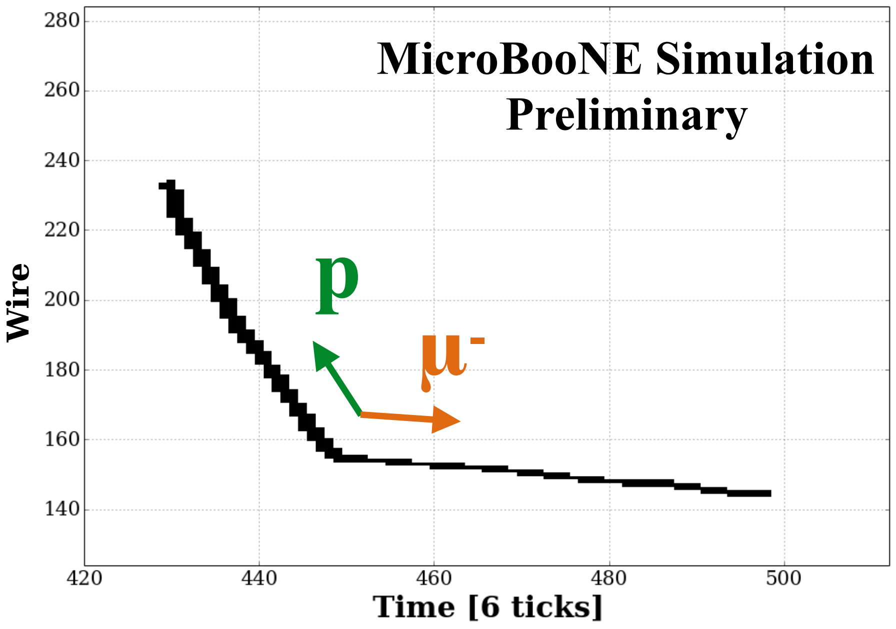
Vertex Reconstruction

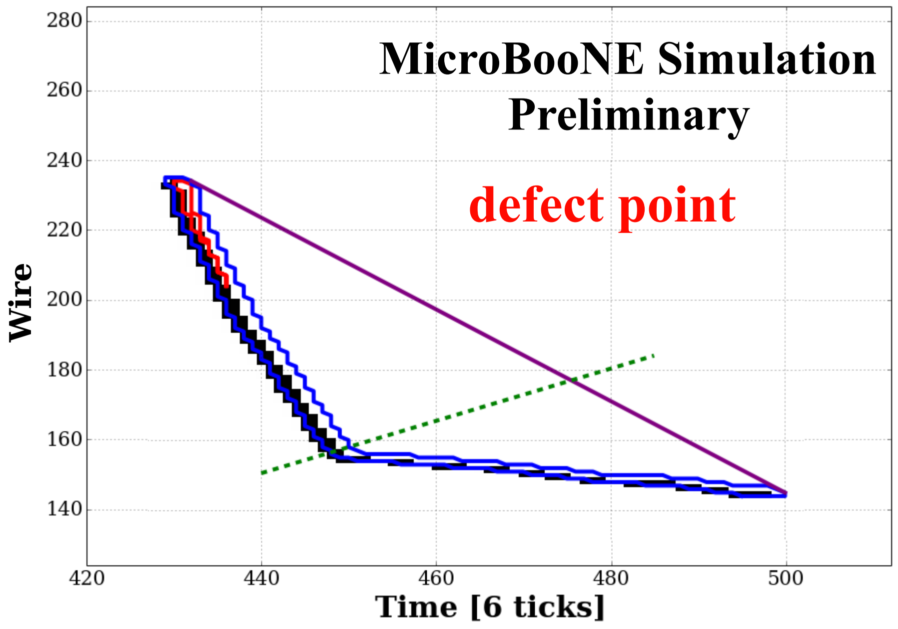


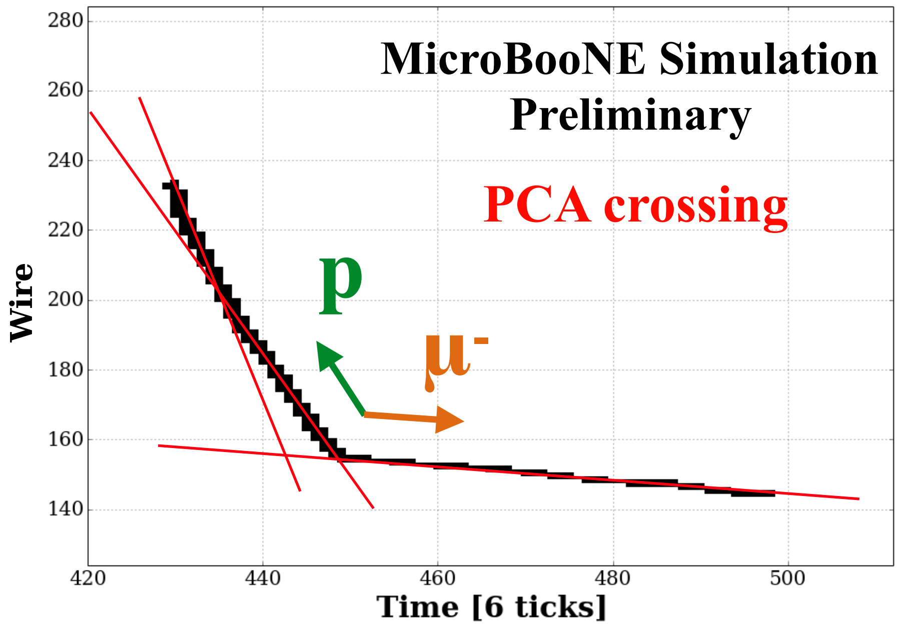
If both track-like and shower-like pixels are found (e.g., a ν_e event):

- For each plane: find endpoint of track where shower is attached
- Correlate these endpoints across planes to identify 3D region
- Scan 3D space around the candidate vertex
- Add a vertex at the 3D point that best matches where the track and shower meet across all three planes

Vertex Reconstruction





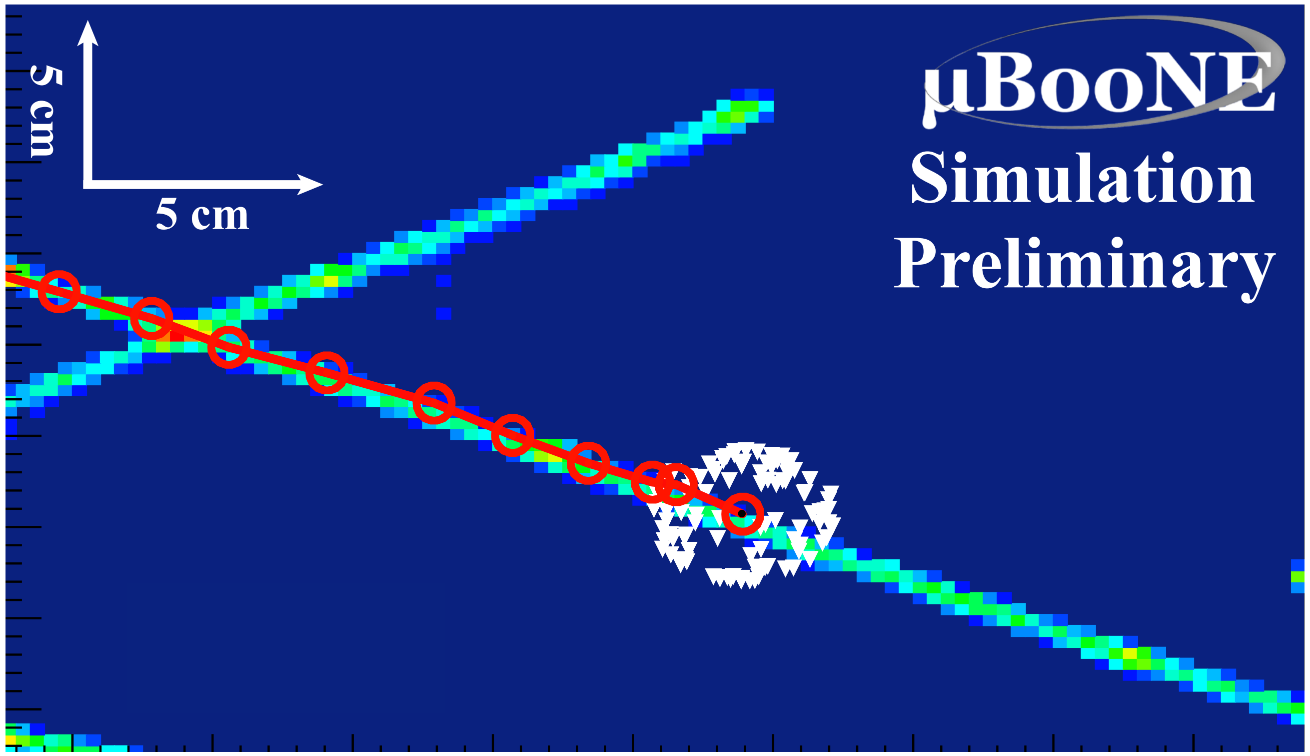


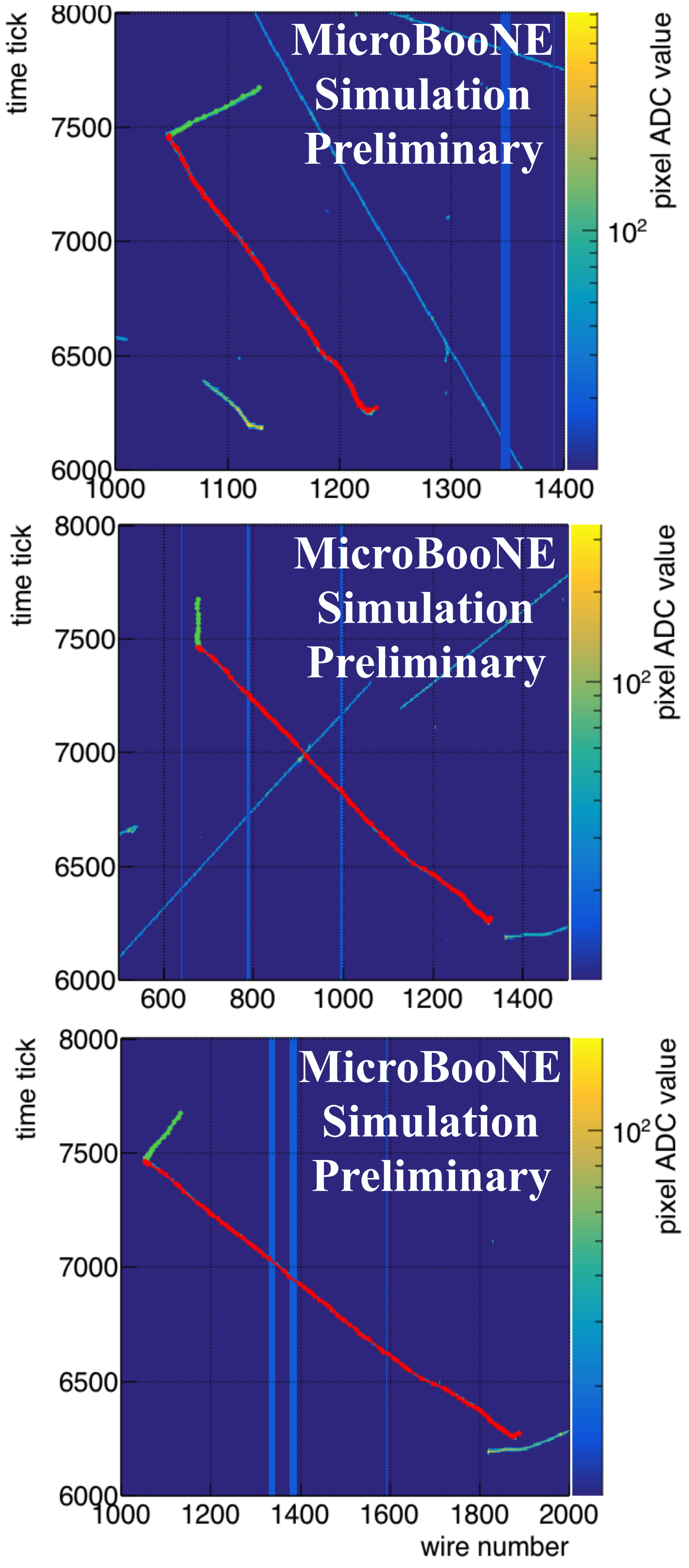
If there are only track-like pixels (e.g., ν_μ normalization sample):

- For each plane: create 2D vertex seeds at any kink points
- Scan space around each seed to find the best vertex point
- Combine information from all three planes
- If the best vertices from each plane are 3D-consistent, add a vertex at that 3D point

Track Reconstruction

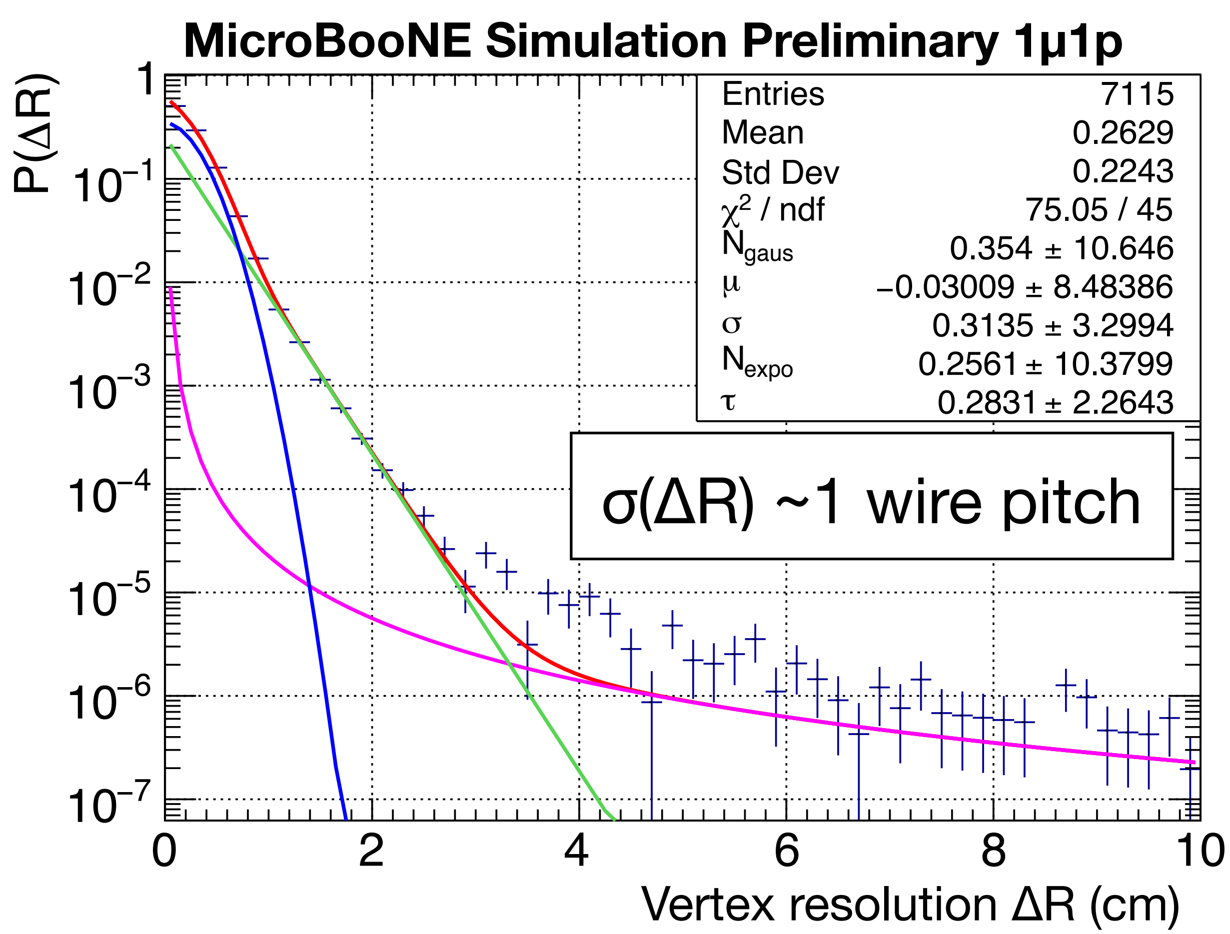
- Track reconstruction algorithm starts from the previously identified 3D vertex point
- Proceeds by stochastic search of nearby 3D space, with preference for continuing in forwards direction
- Once end of track is reached, mask pixels from that track and iterate search from vertex
- Self-diagnostic tool to identify failures

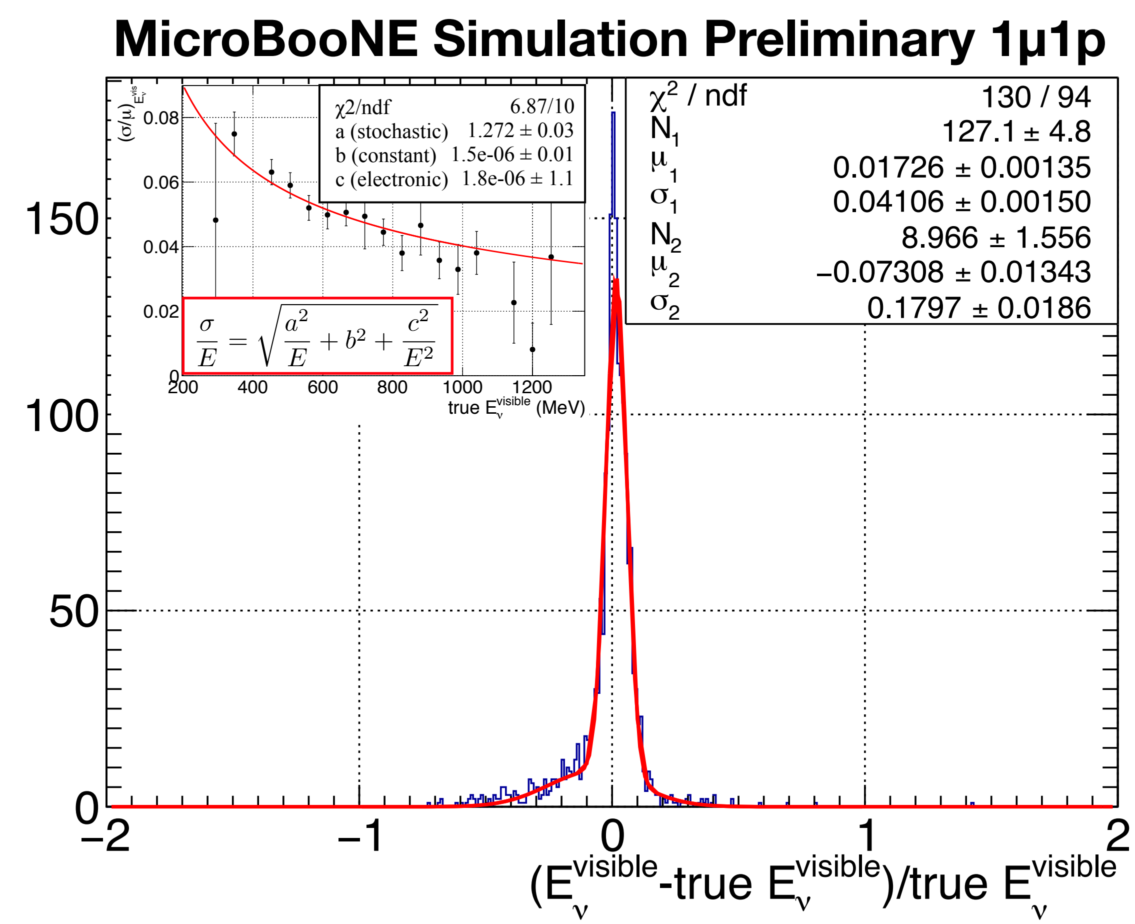




Reconstruction Performance

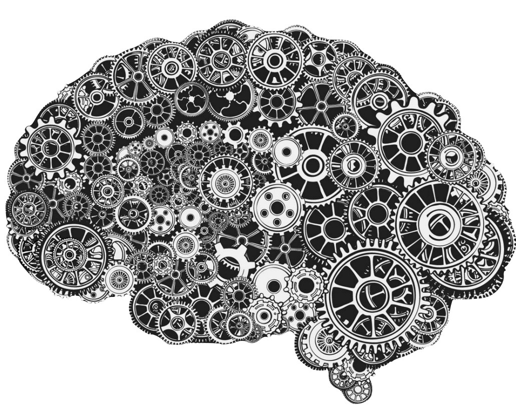
- Vertex spatial resolution is 0.3cm, equivalent to wire spacing
- Length-based energy based on reconstructed tracks achieves 4% resolution for 1μ1p events





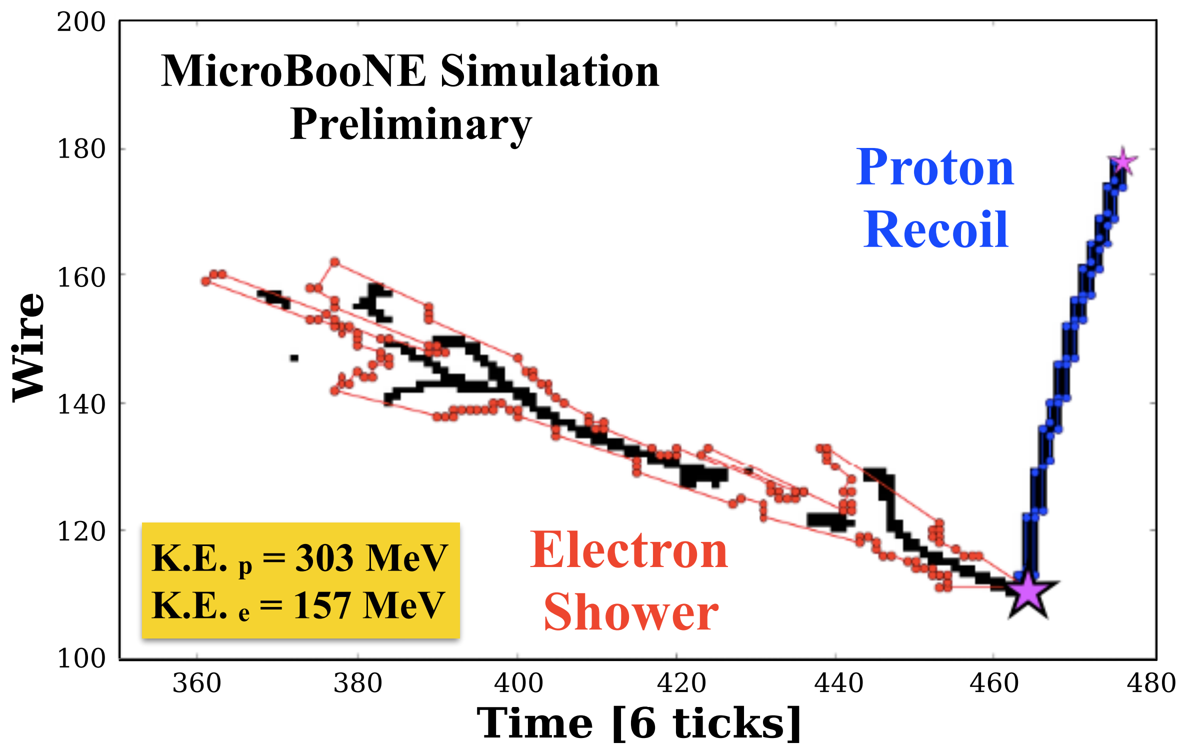
Reconstruction Chain

- PMT Pre-Cuts
- Cosmic Tagging & ROI Finding
- Track vs. Shower Pixel Labeling
- 3D Reconstruction
- Particle ID



Single Particle Identification

μBooNE

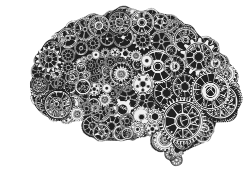


"Convolutional Neural Networks Applied to Neutrino Events in a LArTPC" JINST 12, P03011 (2017)

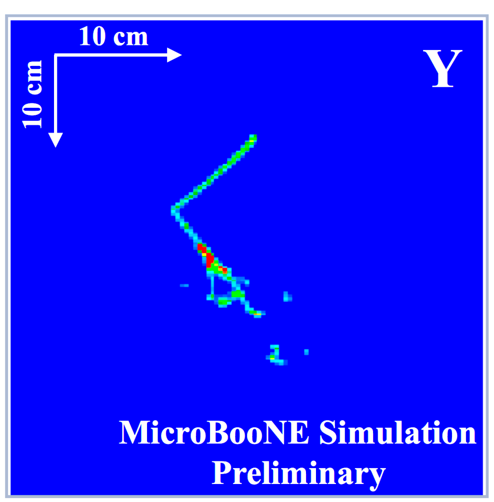
Particle	Correct ID
e^-	77.8 ± 0.7%
γ	83.4 ± 0.6%
μ^-	89.7 ± 0.5%
π^-	71.0 ± 0.7%
p	91.2 ± 0.5%

- Previous work on particle identification for single-particle clusters
  - After 3D vertex reconstruction, clustered pixels attributed to each single track or shower coming out of the vertex
  - Fed individual particle clusters into a CNN trained to do single-particle identification (HighRes GoogLeNet architecture)
- Achieved e^-/γ separation comparable to MicroBooNE design goals

Multi-PID Network



- Currently developing multiple-particle identification network
  - Removes the need to cluster particles coming out of the vertex
  - Provides the network with more context that it can use to make particle identification decisions
- Given an image, network provides the probability that the image contains each of the particles of interest: e^-, γ, μ^-, π^-, p
- Builds on previous single-particle identification network — uses much of the same architecture, just changing last few layers



High probability of e^-, p
Low probability of γ, μ^-, π^-

Reconstruction Chain

PMT Pre-Cuts

↓

Cosmic Tagging & ROI Finding

↓

Track vs. Shower Pixel Labeling

↓

3D Reconstruction

↓

Particle ID

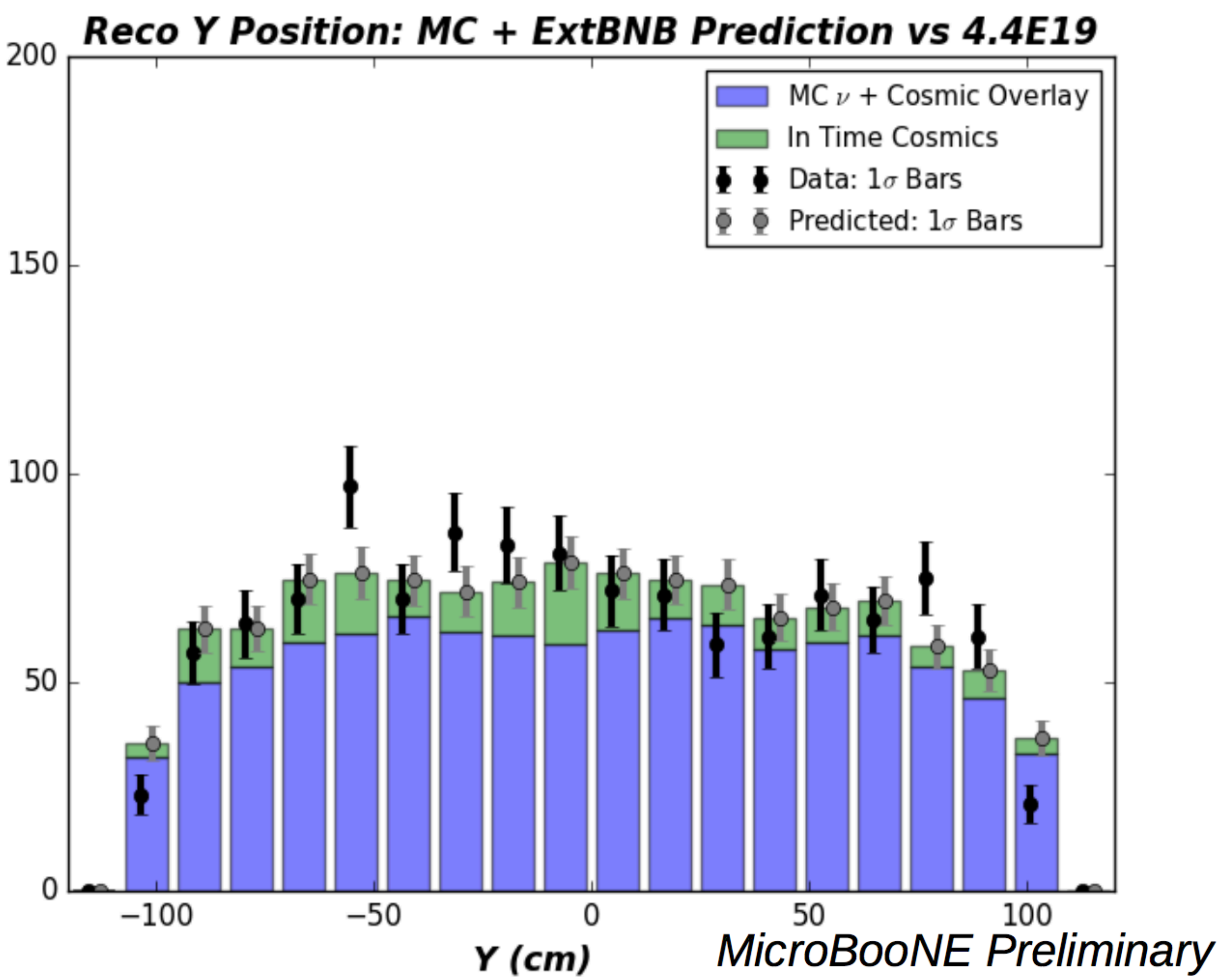
Next Steps: Event Selection For ν_e and ν_μ

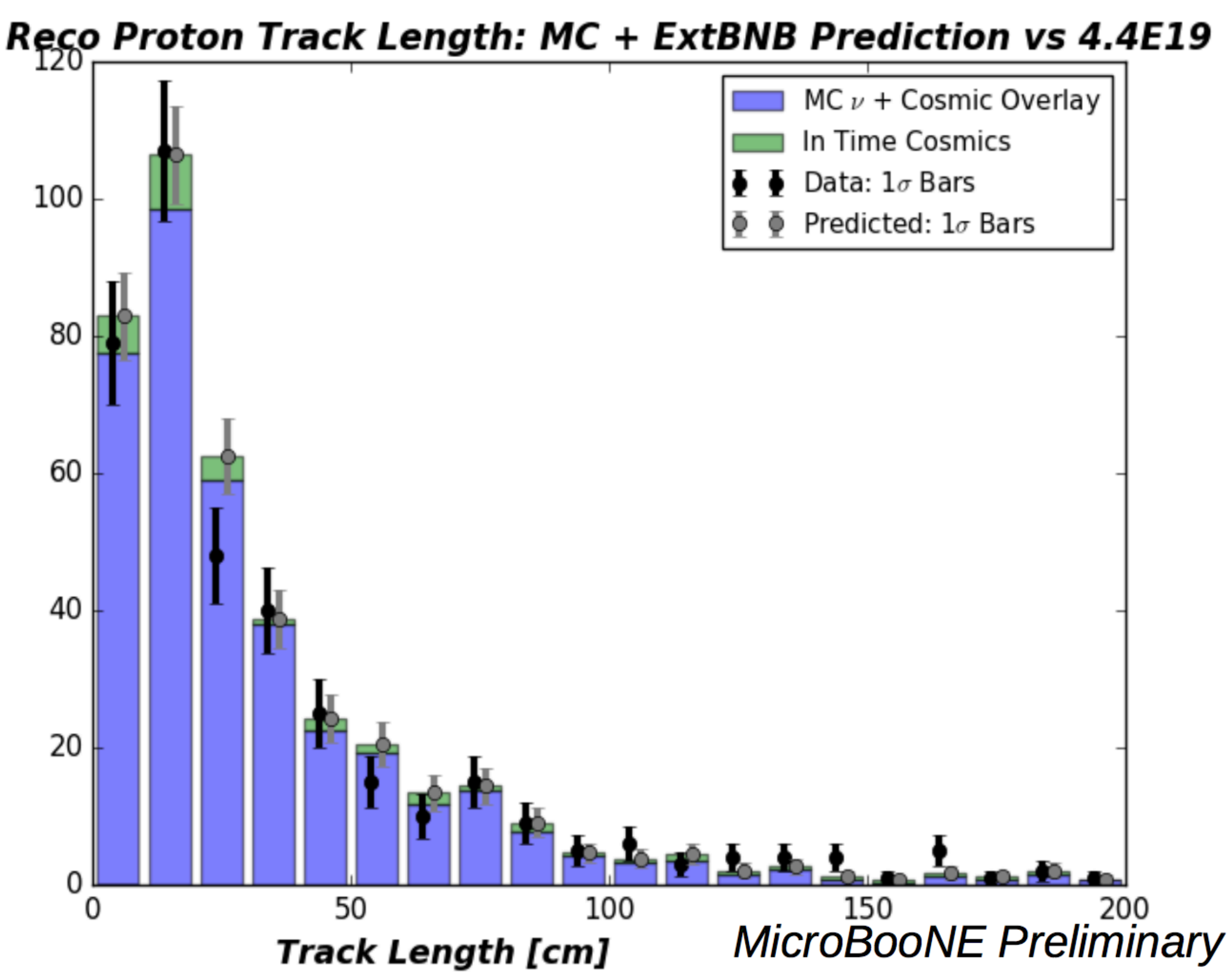
Neutrino Candidate Selection

- After we have reconstructed our events, need to select neutrino candidates for both ν_e and ν_μ
- Still have significant background from cosmic rays and from non-signal neutrino interactions, so selection must reject these
- Focus on the ν_μ selection
  - Exactly two 3D reconstructed tracks
  - Vertex inside the fiducial volume, >10cm from TPC boundary
  - Candidate must pass two likelihood cuts: one designed for cosmic rejection, other for neutrino background rejection
  - Likelihoods considers ionization difference between tracks, how close event is to TPC boundary, track angles relative to drift direction, track angles relative to beam direction

ν_μ Selection Performance

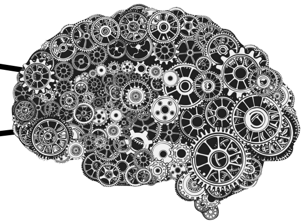
- Very successful at rejection cosmics, such that remaining backgrounds are dominated by neutrino events that do not meet signal definition
- Have achieved 18% efficiency, 47% purity for 1μ1p
- Optimized for low energy reconstruction relevant to MiniBooNE excess





Summary

- Making progress towards an analysis that can probe MiniBooNE low-energy excess anomaly in MicroBooNE
- Fully automated reconstruction chain for low-energy neutrino events, which includes traditional and deep learning algorithms
  - Reject cosmic backgrounds
  - Find the neutrino interaction within the event
  - Label pixels as tracks or showers
  - Reconstruct event in 3D
  - Identify particle species
  - Select ν_e and ν_μ events

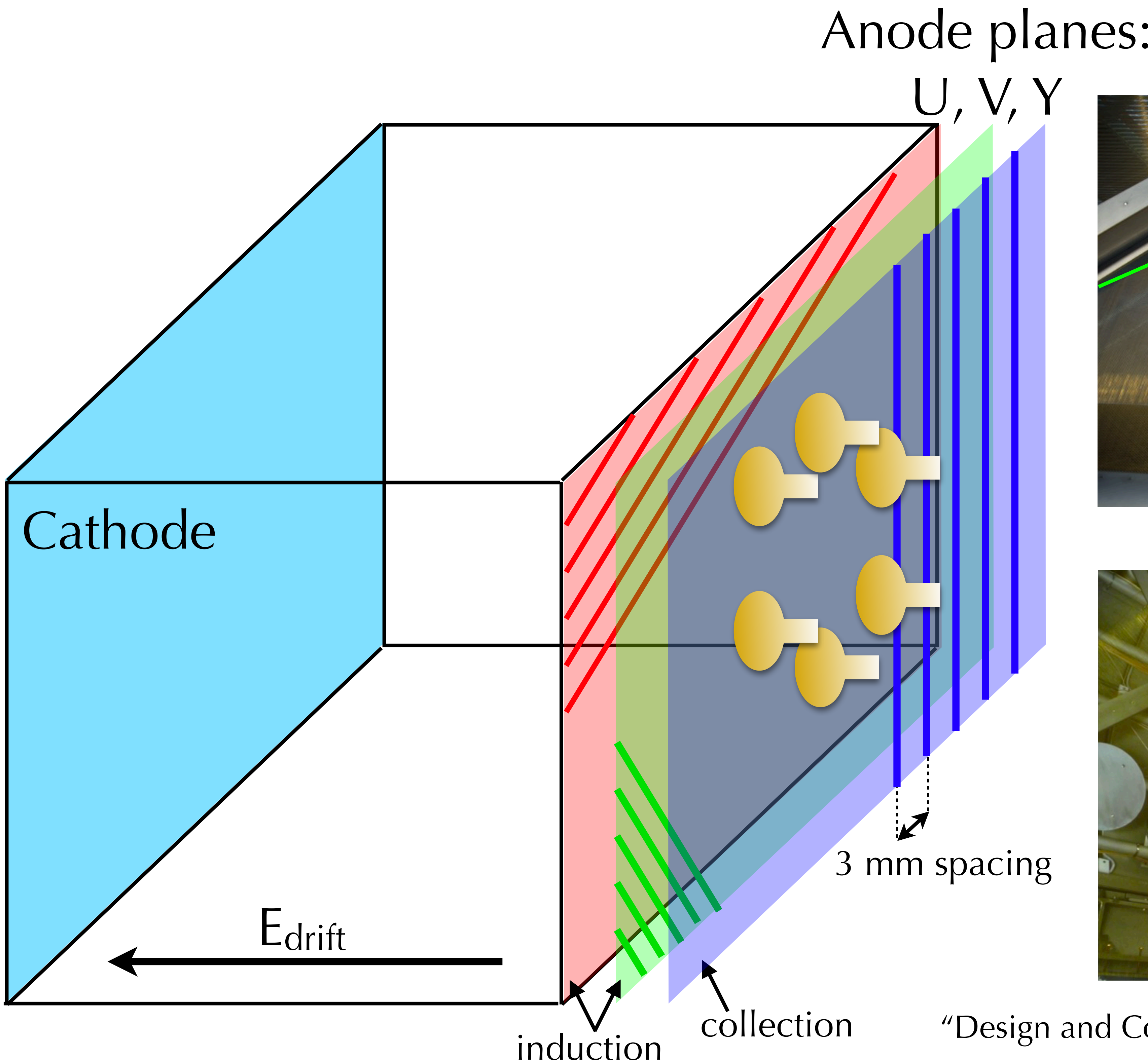


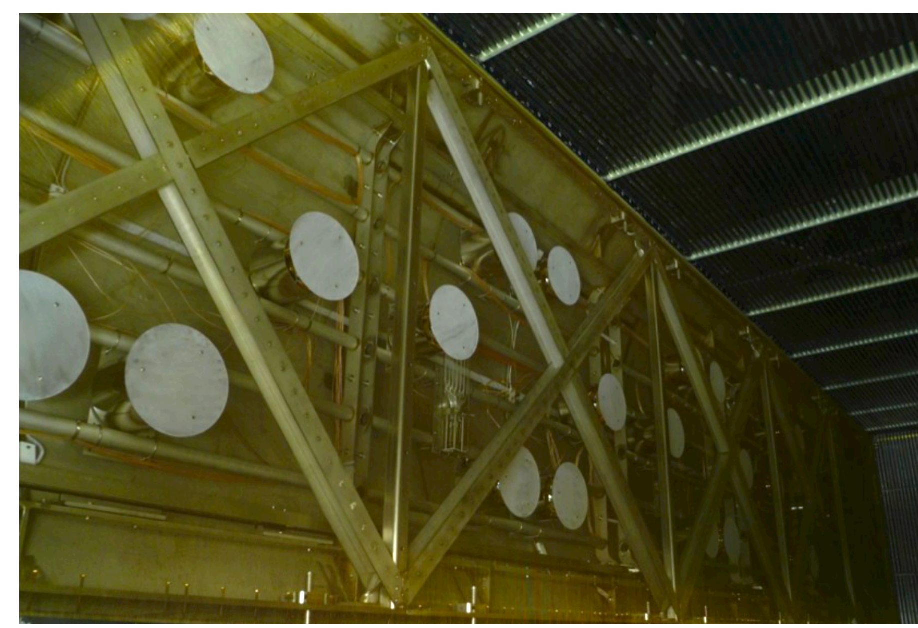
- Currently refining event selection algorithms and pursuing studies of flux, cross-section, and detector systematic uncertainties
- MicroBooNE is doing important development work for future LArTPC detector experiments

Thank you!

Backup Slides

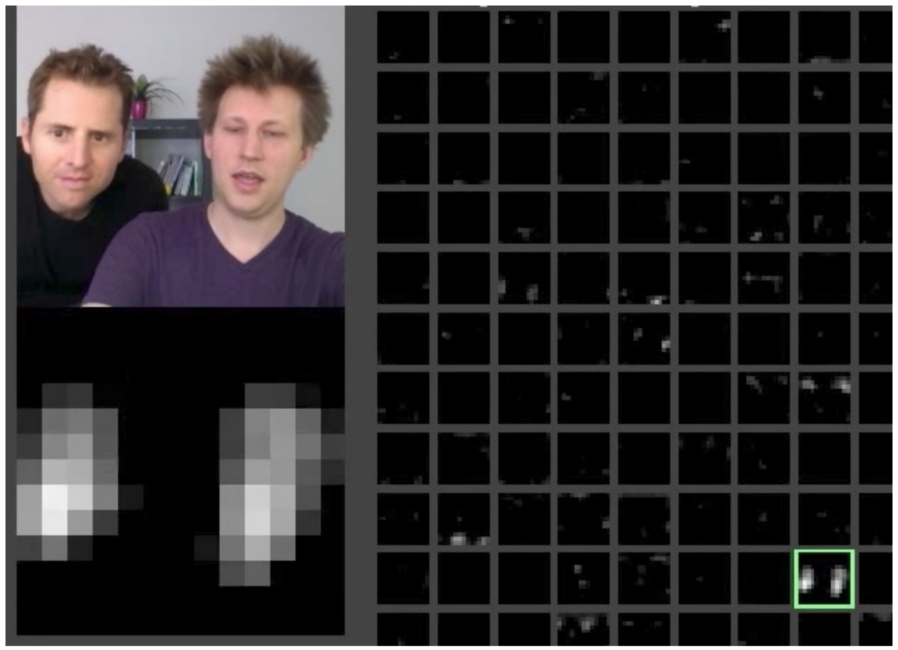
The MicroBooNE Detector





"Design and Construction of the MicroBooNE Detector"
JINST 12, P02017 (2017)

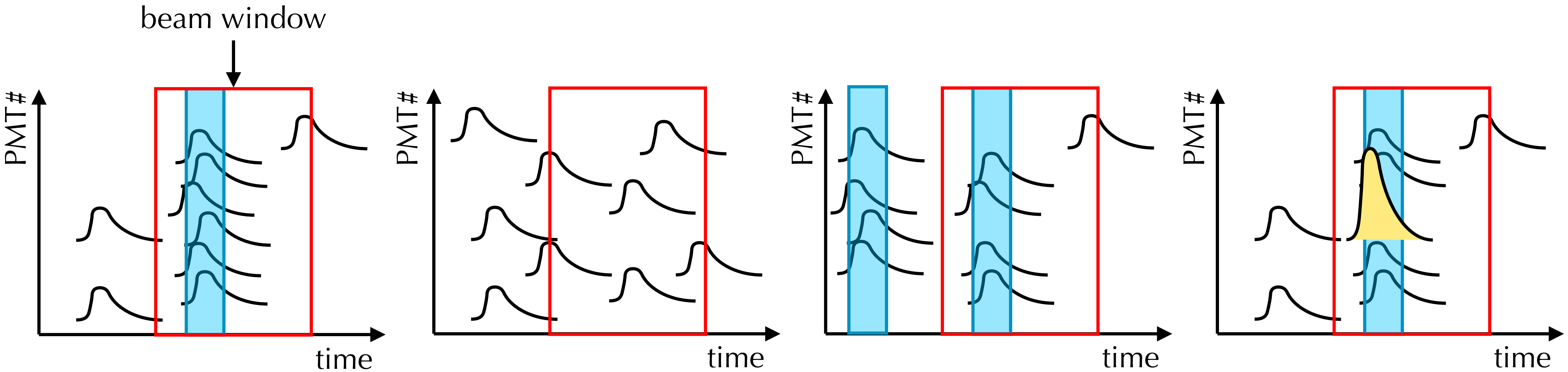
A Few Words About Deep Learning



https://www.youtube.com/watch?v=AgkfIQ4IGaM

- Convolutional neutral networks have several important properties
  - "Neurons" scan over the image looking at a limited set of pixels at each point
  - They "learn" local, translationally invariant features
  - Each layer of neurons builds on the features found by the previous ones to reach increasing levels of complexity/abstraction
- In the above, the black-and-white boxes show the "activation" of neurons in response to the images; the neuron highlighted on the right responds to faces, while the one on the left responds to text

PMT Pre-Cuts



Keep: All possible neutrino events

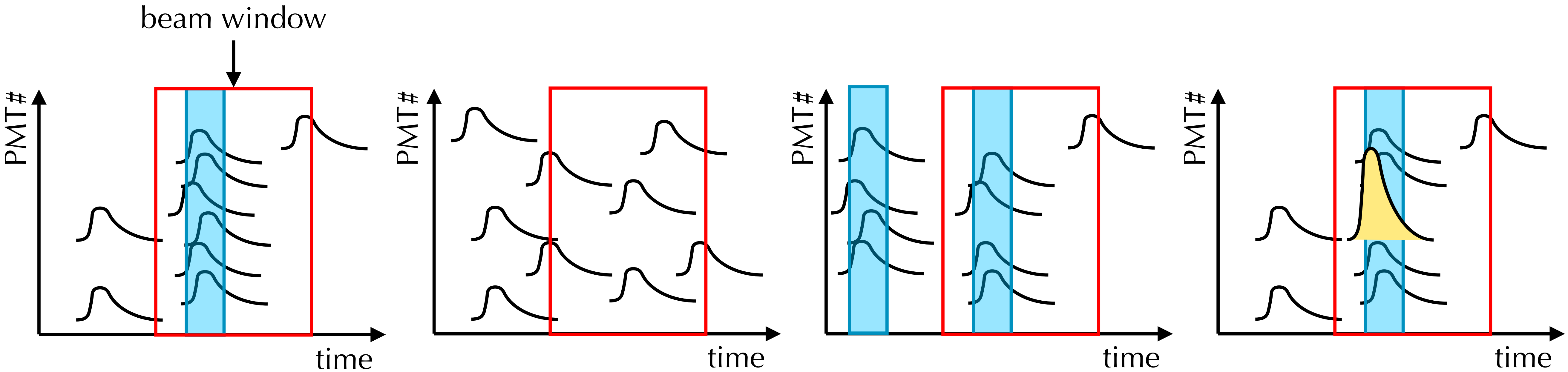
Reject: Random, single-photoelectron noise

Reject: In-time flash caused by Michel electron, from the decay of pre-beam cosmic muon

Reject: PMT-based noise

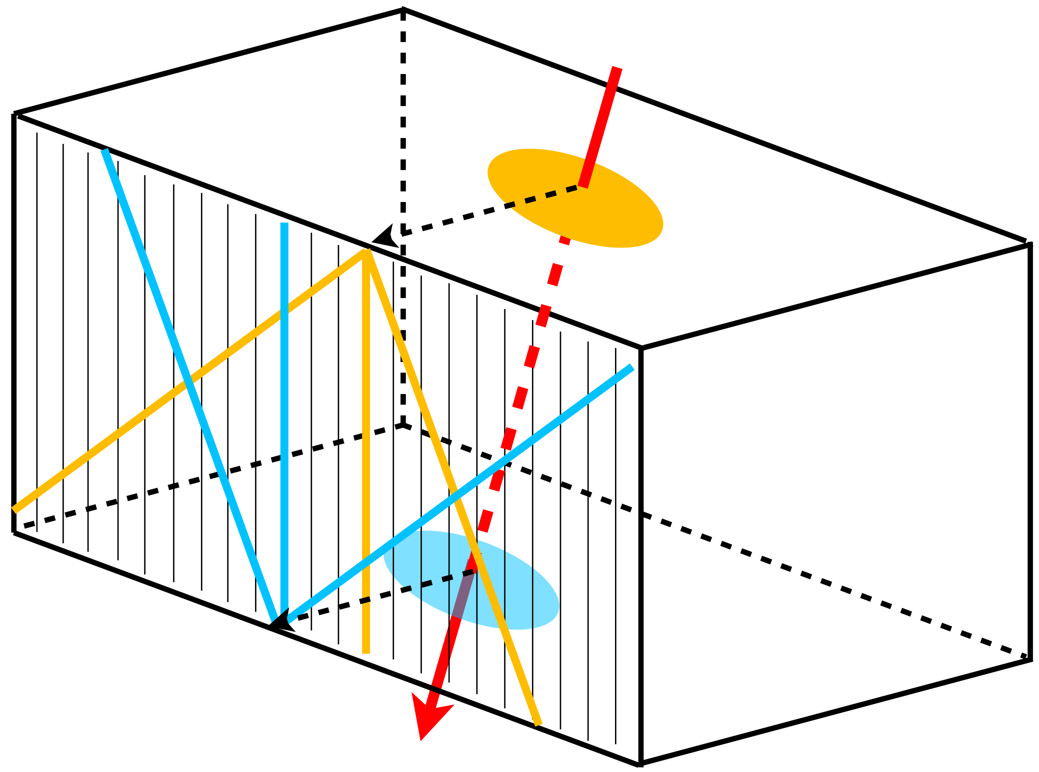
- Keep >96% of neutrinos (based on simulations)
- Reject >75% of background (based on rejection of off-beam data)

PMT Pre-Cuts



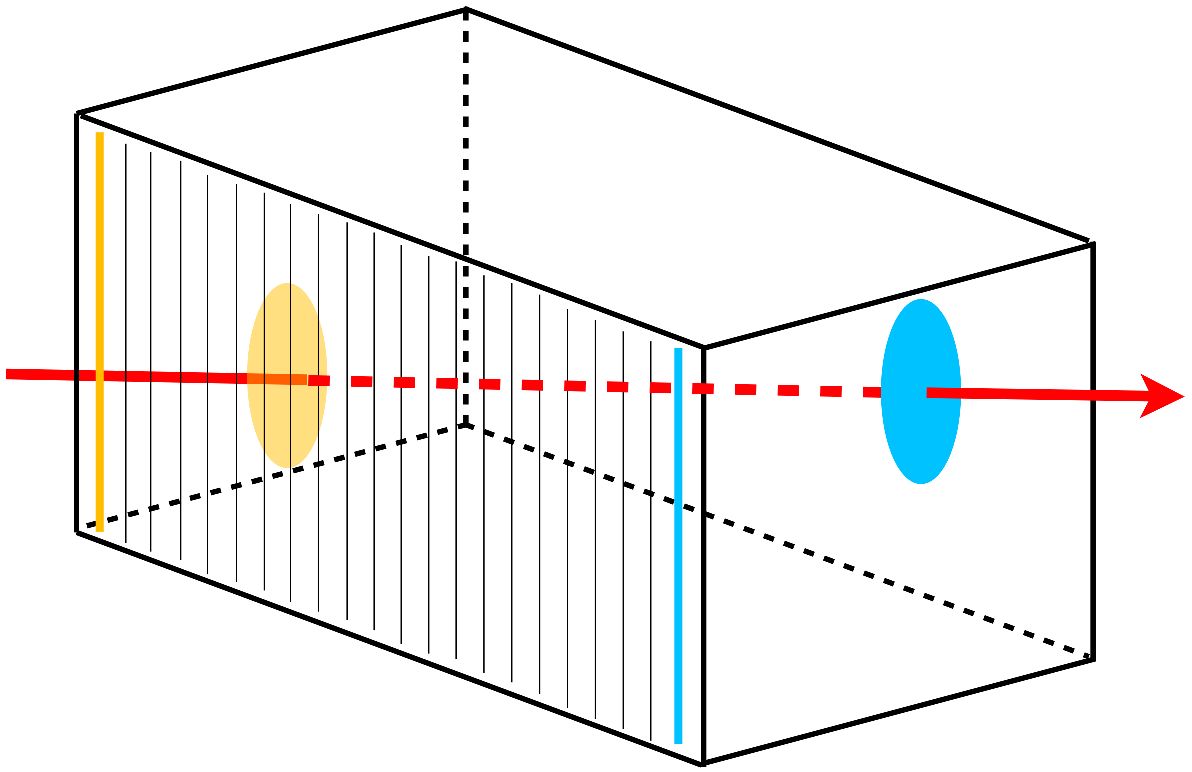
- Reject: Random, single-photoelectron noise (~200 kHz)
  - No time correlation between these single-photoelectron pulses
  - Require 20 photoelectrons in 93.75 ns — this becomes the definition of a "signal"
- Reject: In-time flash caused by Michel electron, from decay of a cosmic muon
  - Require no signal for 2 μs before the beam window
- Reject: PMT-based noise
  - Limit the total amount of the light collected by a single PMT to <60% of the total light
- Keep >96% of neutrinos (based on simulations)
- Reject >75% of background (based on rejection of off-beam data)

Cosmic Pixel Tagging



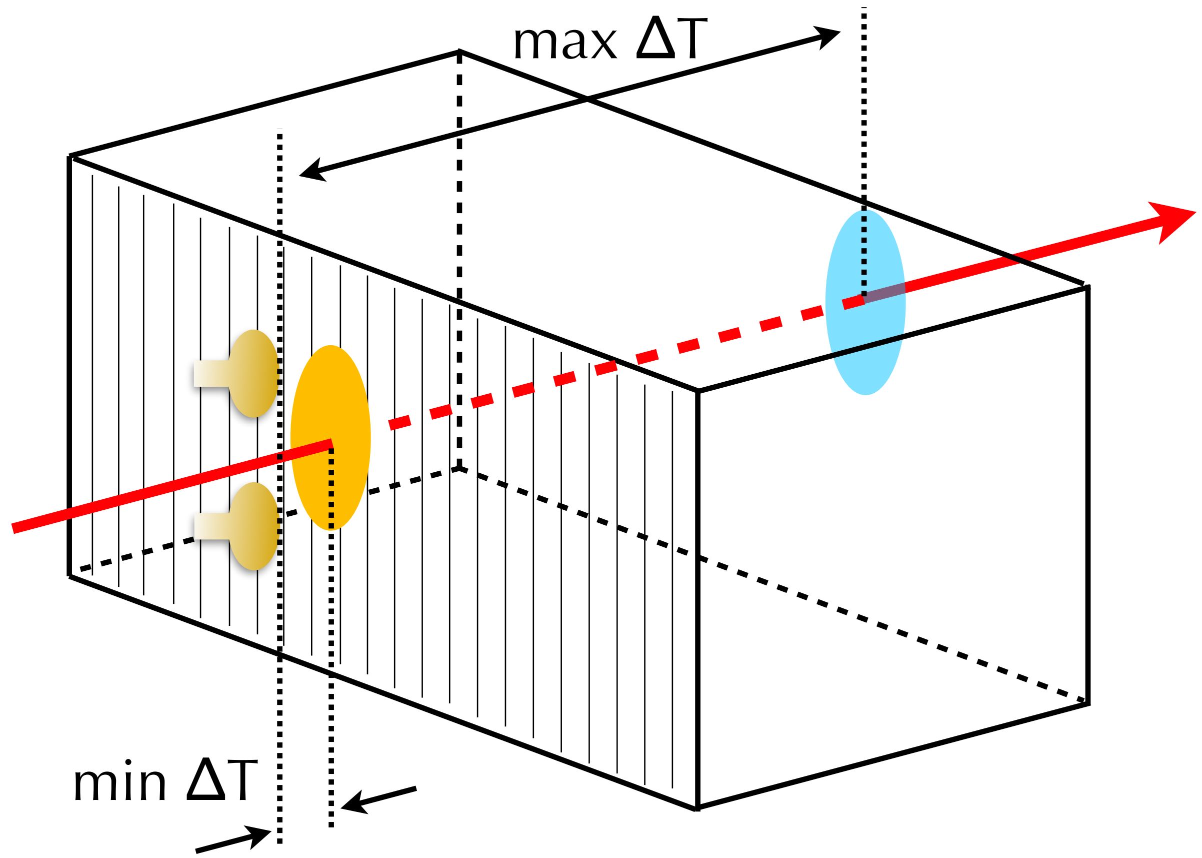
- Cosmic and other background tracks cross the TPC boundary
- Identify and tag these boundary crossing points
  - **Top/bottom**: crossings deposit charge on triplets of wires that meet at the boundary
  - Upsteam/downsteam: crossings deposit charge on the first/last wires on the Y plane
  - Anode/cathode: crossings have specific ΔT between PMT flash and wire signal
- Connect end points by following the charge using 3D path finding

Cosmic Pixel Tagging



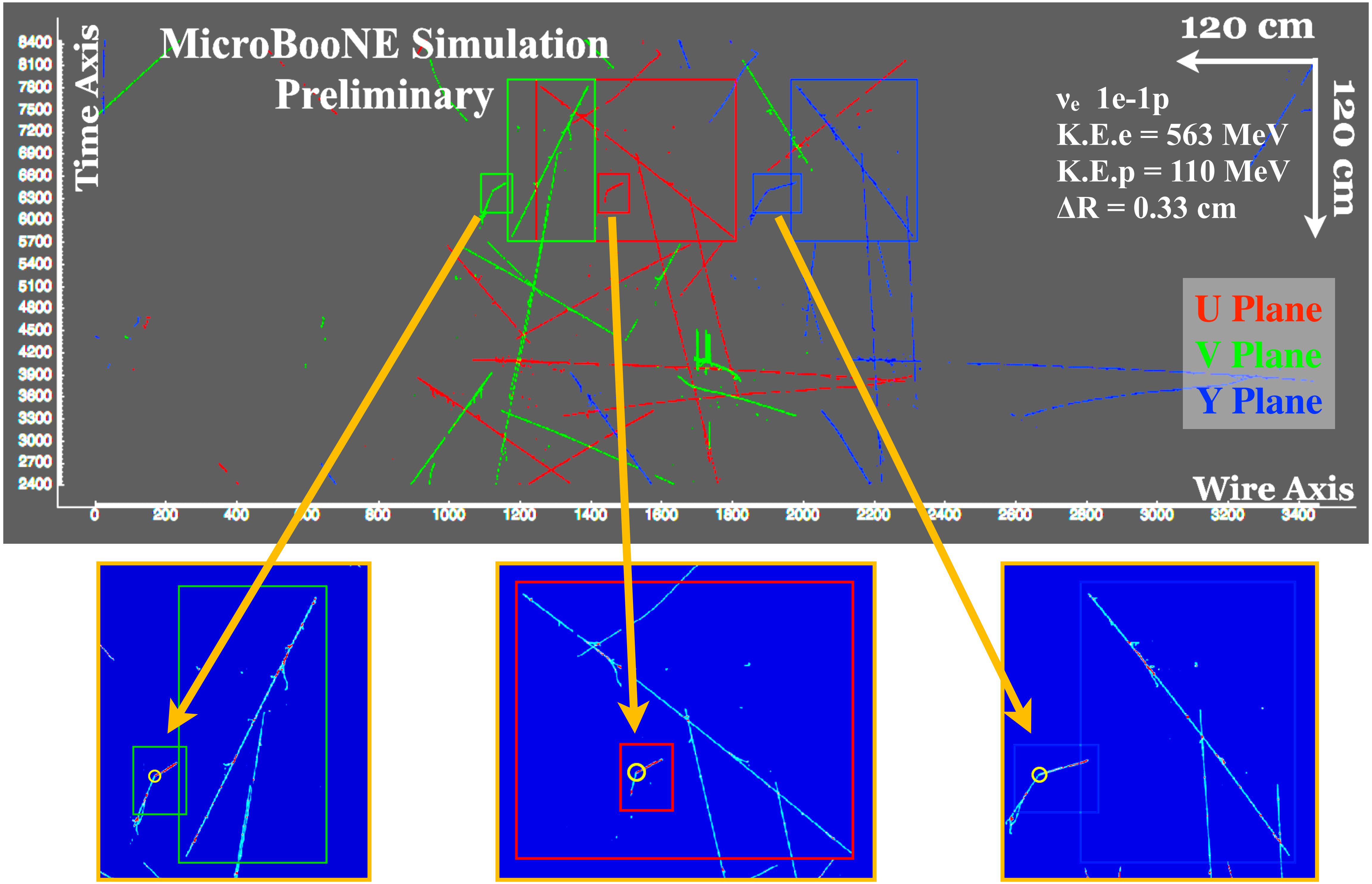
- Cosmic and other background tracks cross the TPC boundary
- Identify and tag these boundary crossing points
  - Top/bottom: crossings deposit charge on triplets of wires that meet at the boundary
  - **Upsteam/downsteam**: crossings deposit charge on the first/last wires on the Y plane
  - Anode/cathode: crossings have specific ΔT between PMT flash and wire signal
- Connect end points by following the charge using 3D path finding

Cosmic Pixel Tagging



- Cosmic and other background tracks cross the TPC boundary
- Identify and tag these boundary crossing points
  - Top/bottom: crossings deposit charge on triplets of wires that meet at the boundary
  - Upsteam/downsteam: crossings deposit charge on the first/last wires on the Y plane
  - **Anode/cathode**: crossings have specific ΔT between PMT flash and wire signal
- Connect end points by following the charge using 3D path finding

Region-of-Interest Finding



After tagging cosmic tracks, draw 3D region-of-interest (ROI) box around untagged pixels