Interaction Vertex Identification Using Convolutional Neural Network

Heng-Ye Liao
ProtoDUNE-SP Analysis Meeting
Dec 12, 2019

Introduction

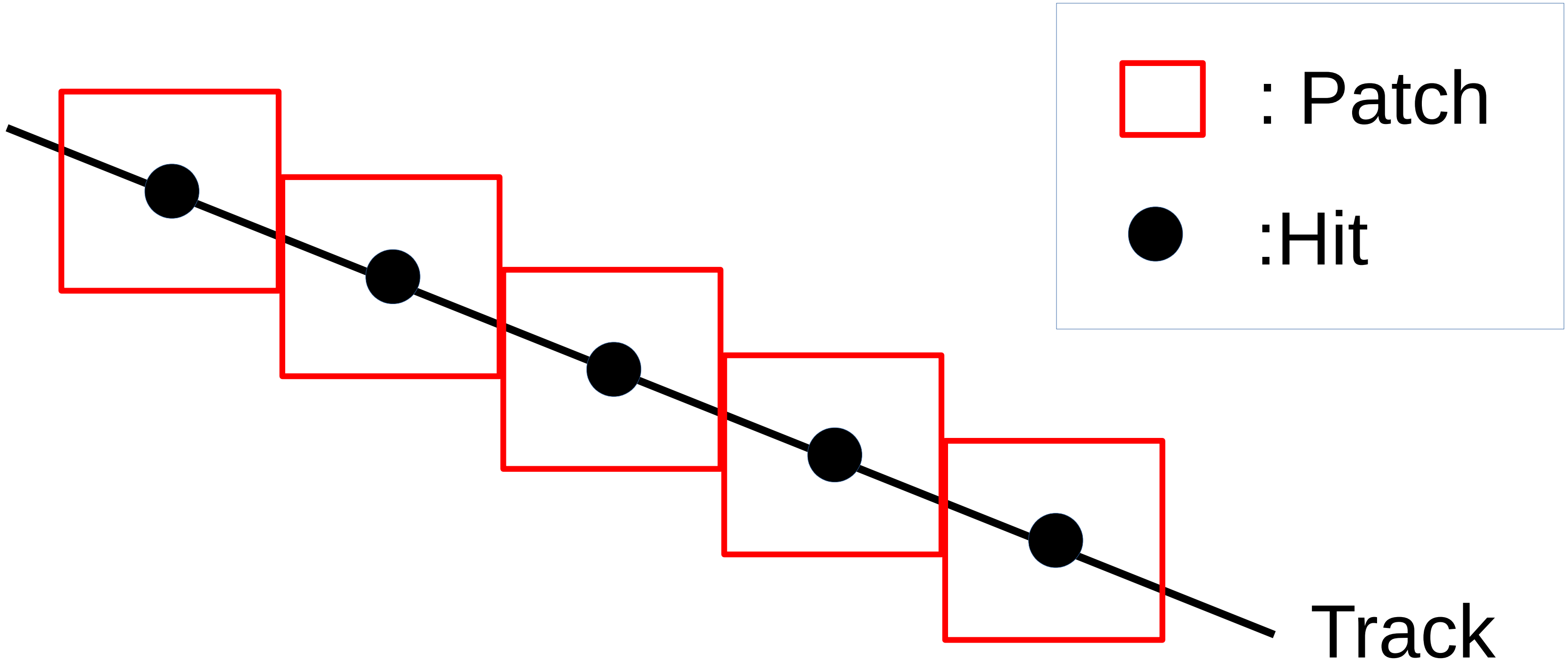
- Use Convolutional Neural Network for interaction vertex finding
- Modifying the current existing network of finding neutrino vertices original developed by Robert Sulej and Dorota Stefan
- Give each hit in a track the scores of elastic and inelastic scattering
  → The initial idea is triggered by Leigh
- Structure of Convolutional Neural Network
  → Tingjun and Aidan made major contribution for building the framework

Basics of Convolutional Neural Network

CNN is composed of three parts:

1. **Convolution layer** (extract feature of input image)
2. **Pooling layer** (dimensionality reduction, prevent over-fitting)
3. **Fully-connected layer (dense layer)** (Output result)

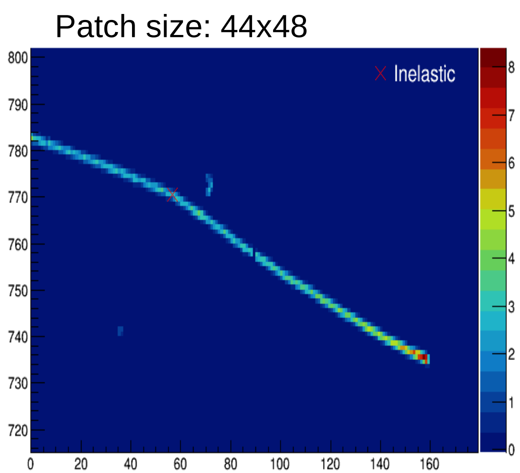
Patches

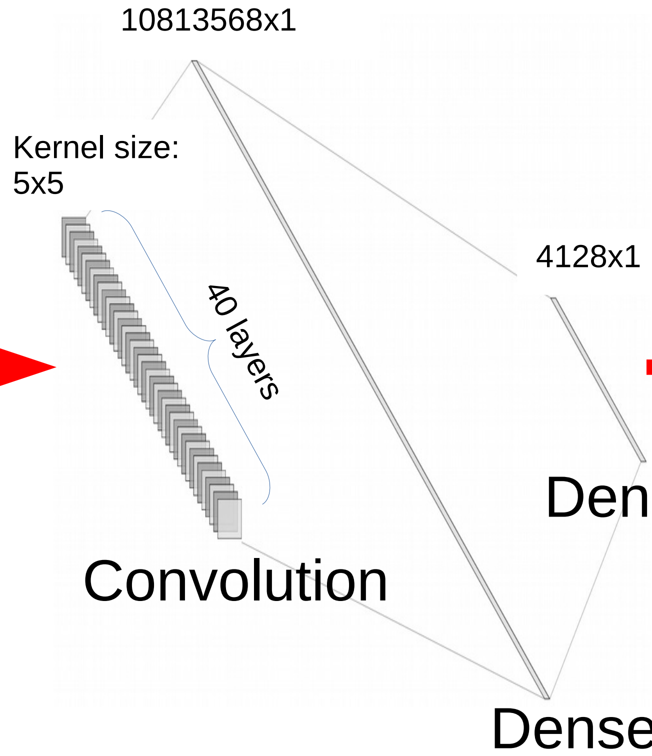


- Network training not using the entire image but using patches instead
- Advantages of using patches:
  Save memory, save time, and make the training possible

Training Sample & Procedure

- Training sample: Single proton MC, SCE OFF
- Procedure:



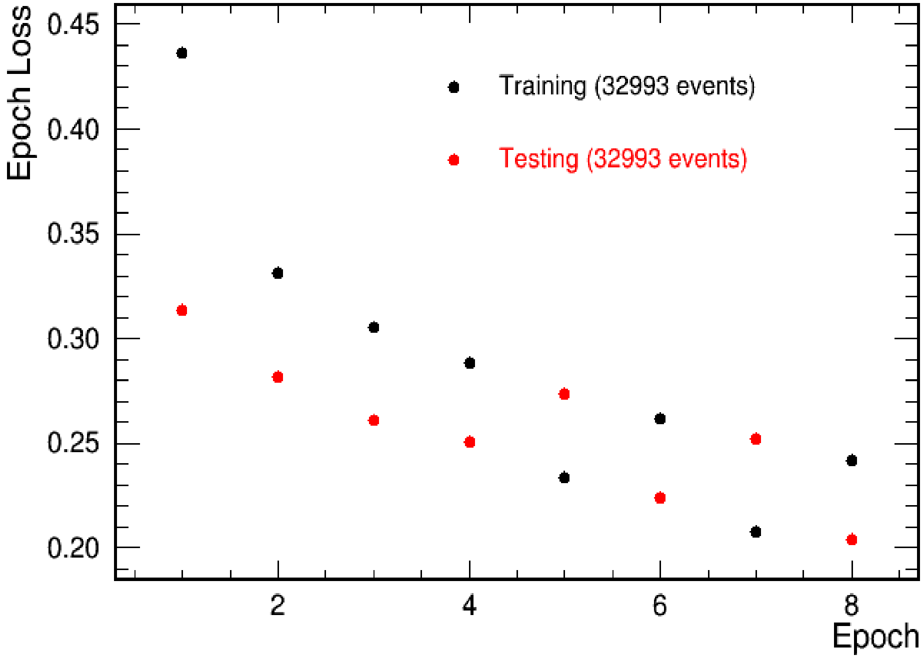


Output:
Elastic, Inelastic & none scattering score [0-1] per pixel

Input:
Image with labels (elastic/inelastic/none)

[Only collection plane Images]

Loss Function



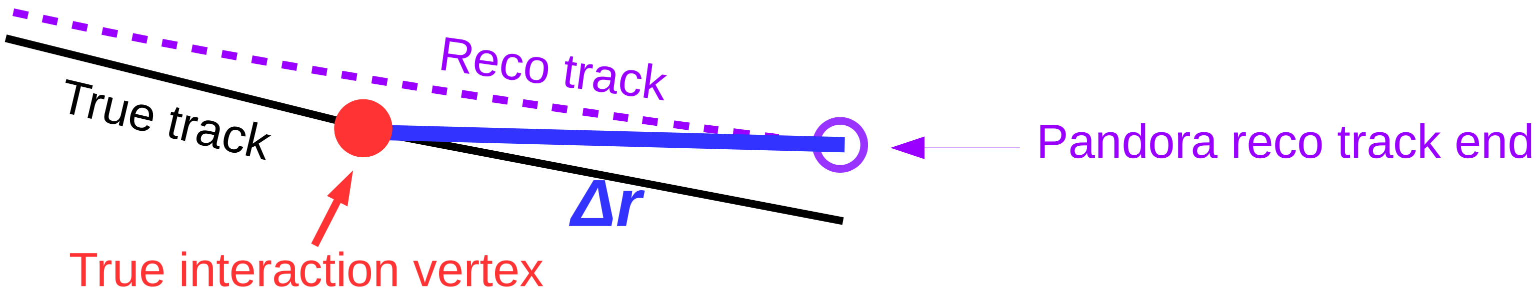
- Loss function (brief description):
  A metric to measure "distance" between the ground truth and the network outputs (the lower, the better)

- Loss function of use:
  Categorical cross-entropy

Epoch:
The process of updating network parameters after forward+backward propagations

Definition of Δr

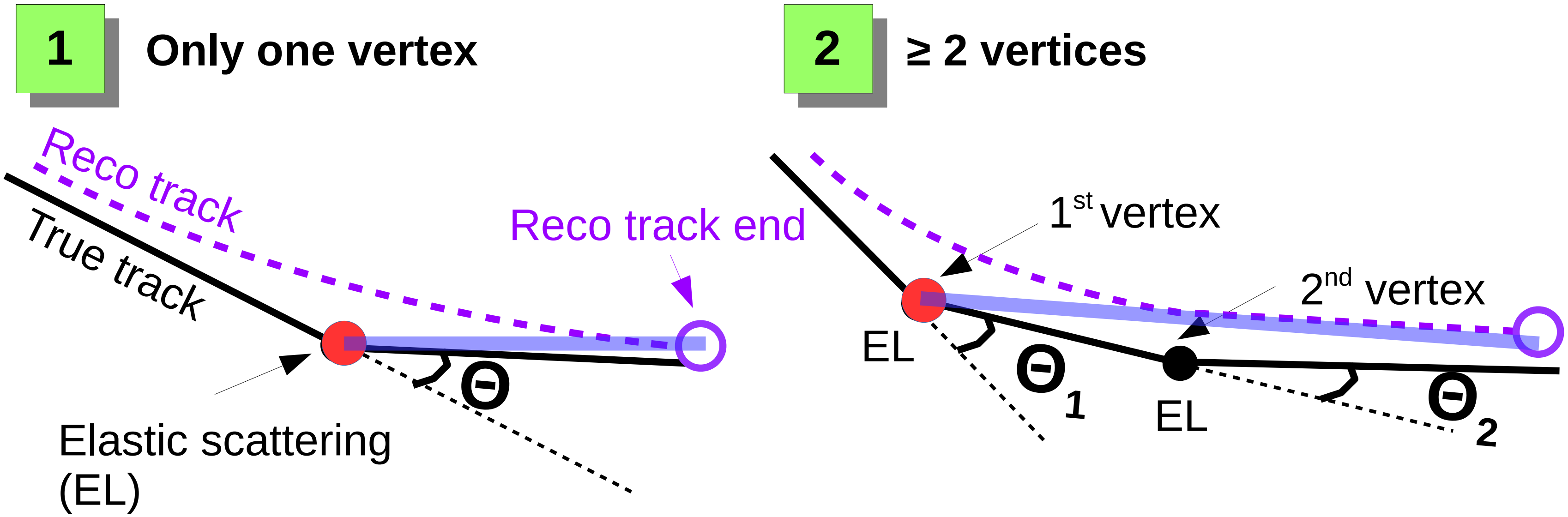
- Δr := Distance between true vertex and reco vertex



- Pandora & CNN vertex identification:
  - Pandora: Assume reconstructed primary track end has interaction
  - CNN: Choose the highest CNN score as interaction vertex (elastic / inelastic score)

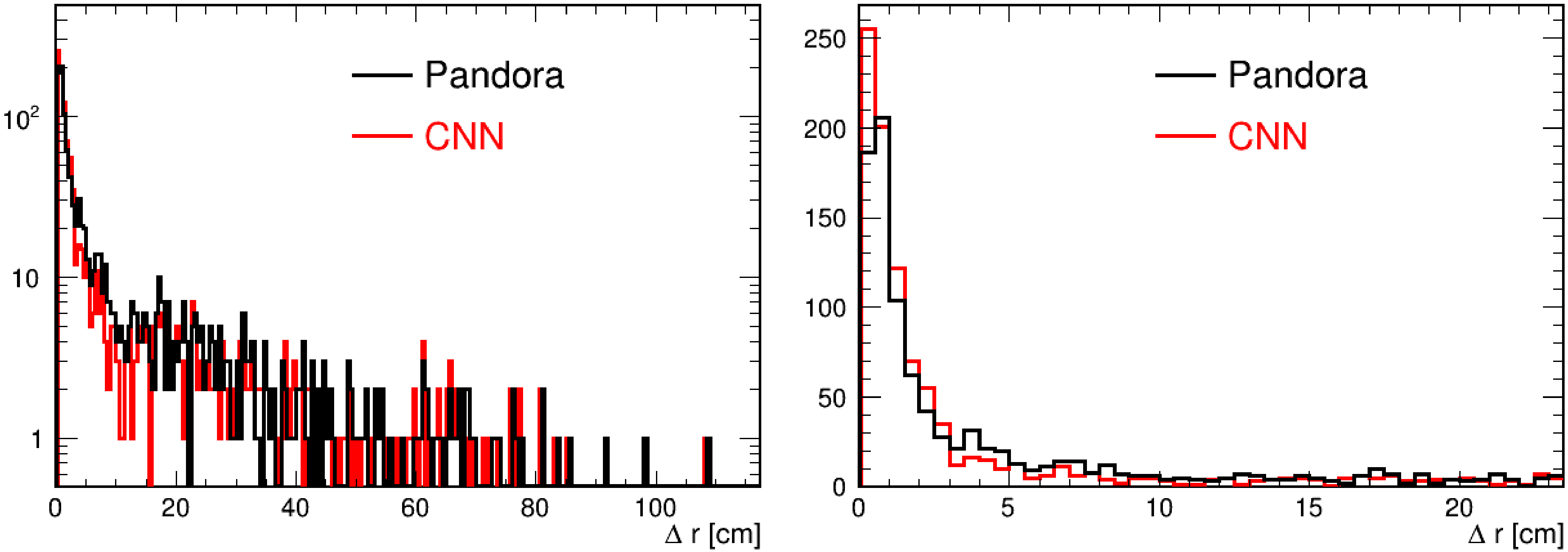
  p.s. Sometimes the beginning of track gets a higher score, skipping the scores of first few wires (≥3 cm in Z) for the entire analysis

Elastic/Inelastic scattering



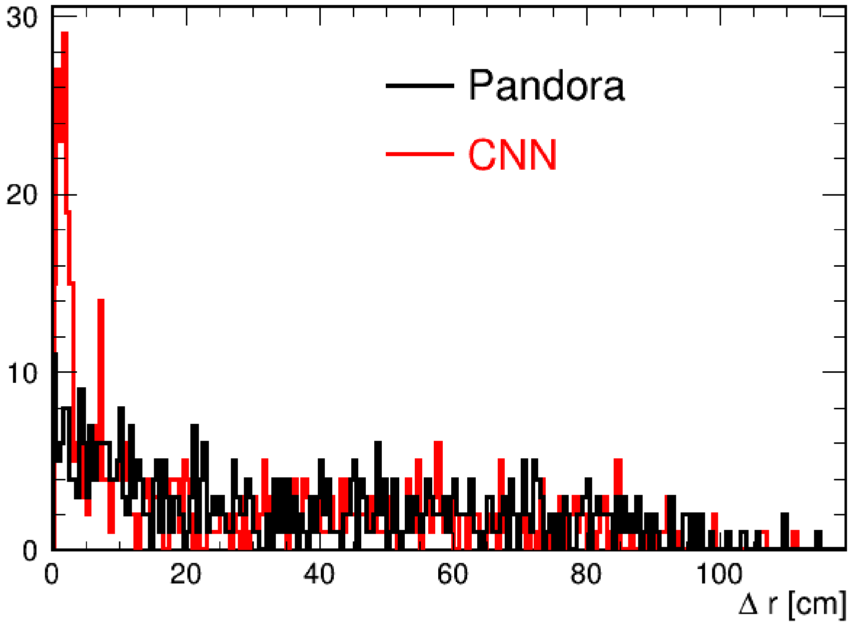
- For the inclusive XS study, we only care about where the interaction takes place → Looking at the first interaction vertex and see how well we can identify it
  Δr

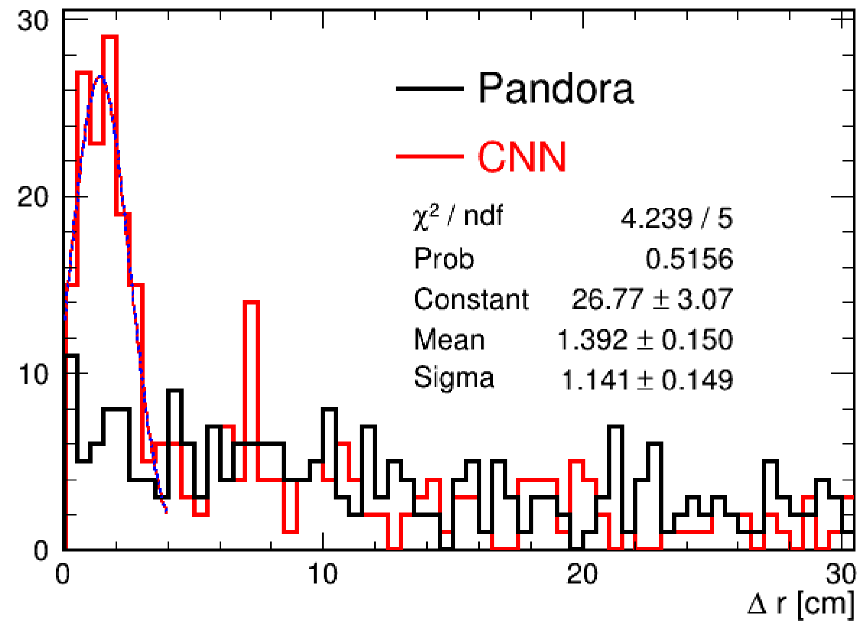
Δr Distributions-Inelastic Scattering



- Pandora does a great job on inelastic scattering

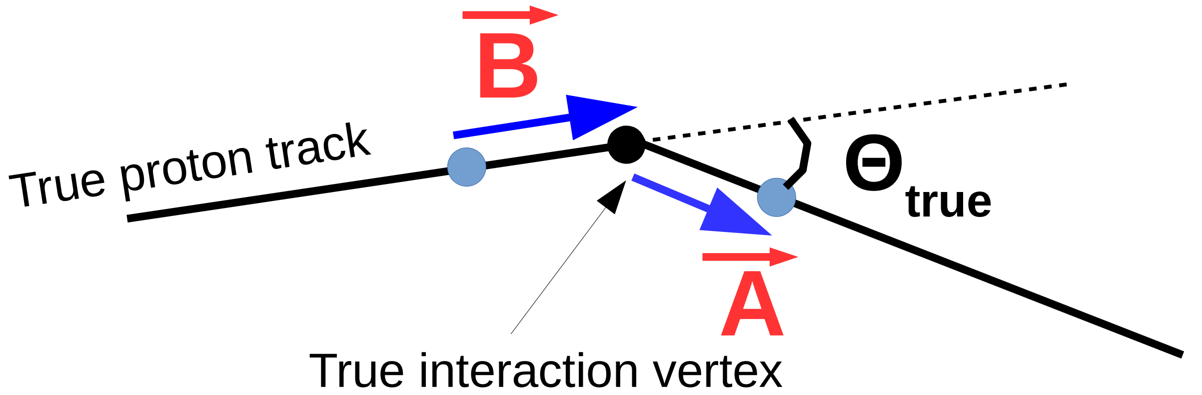
Δr Distributions – Elastic Scattering





- CNN peaking at ~1.4 cm in Δr

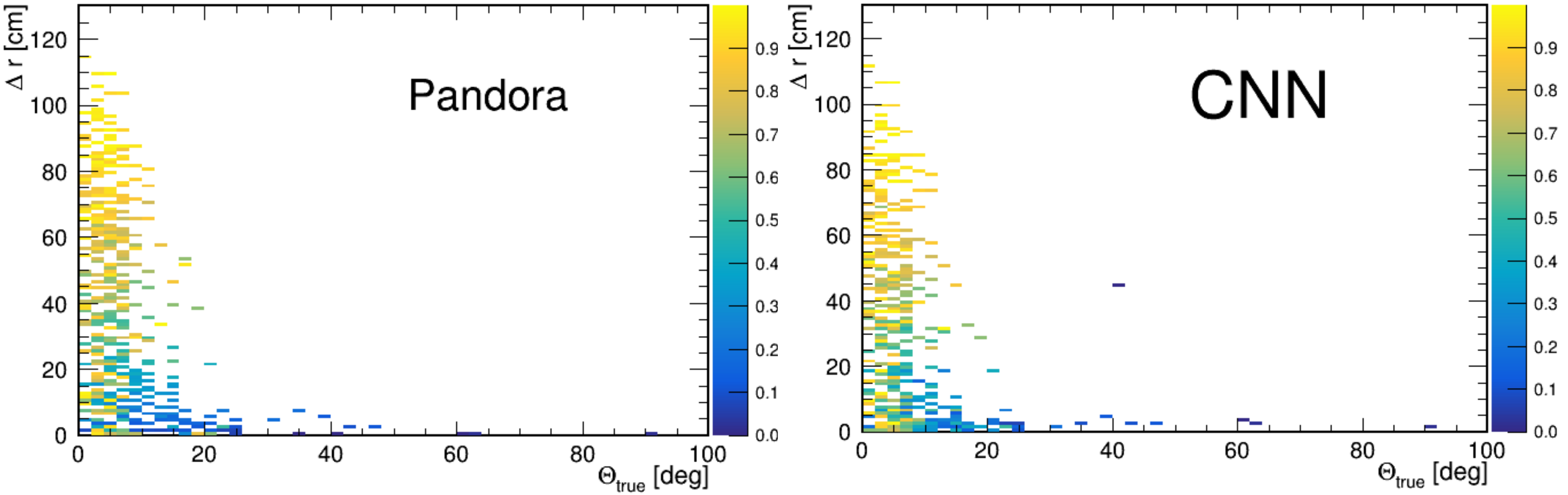
Angle Calculation



- Angle calculation for true interaction vertices:
  - Use the hits before and after the true interaction vertices
  - Definition of angle:

$$\Theta_{true} = \cos^{-1}\left(\frac{\vec{A}\cdot\vec{B}}{|\vec{A}||\vec{B}|}\right)$$

Δr v.s. Θ [Elastic Scattering]

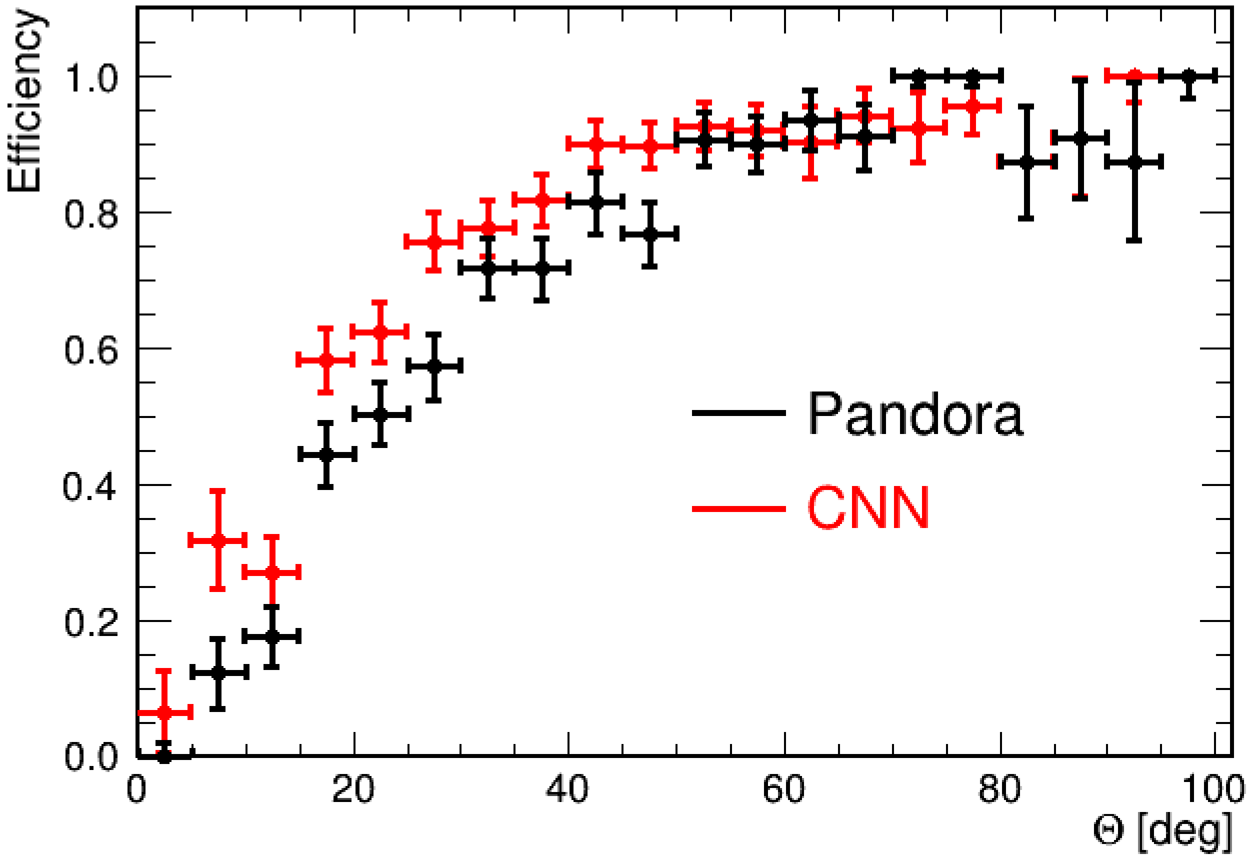


- Color code in the plots: Fraction of remaining KE before interaction vertex

$$f^{RE} = 1 - \frac{\sum_{j=1}^{vertex} \Delta E_j}{KE^{ff}}$$

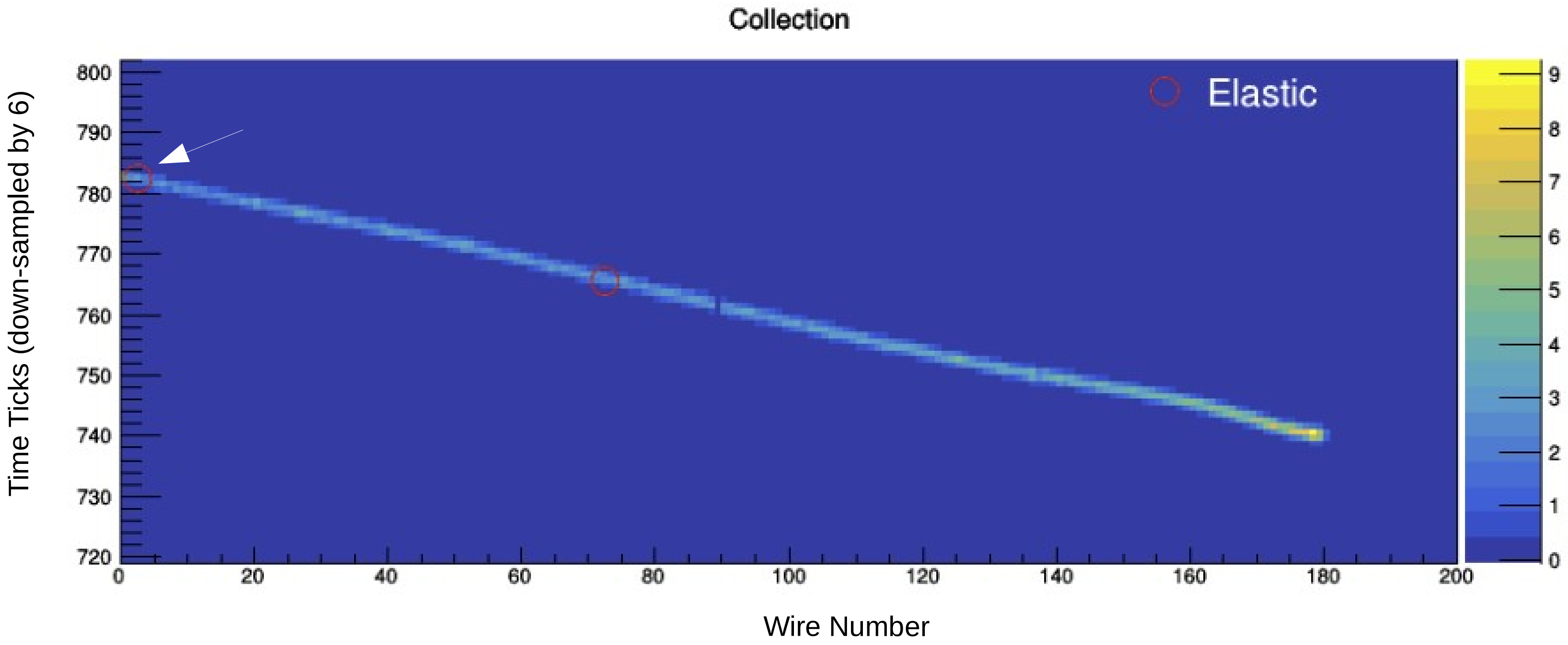
ΔE_j : Energy deposition of each hit

Efficiency v.s. Θ [Elastic Scattering]



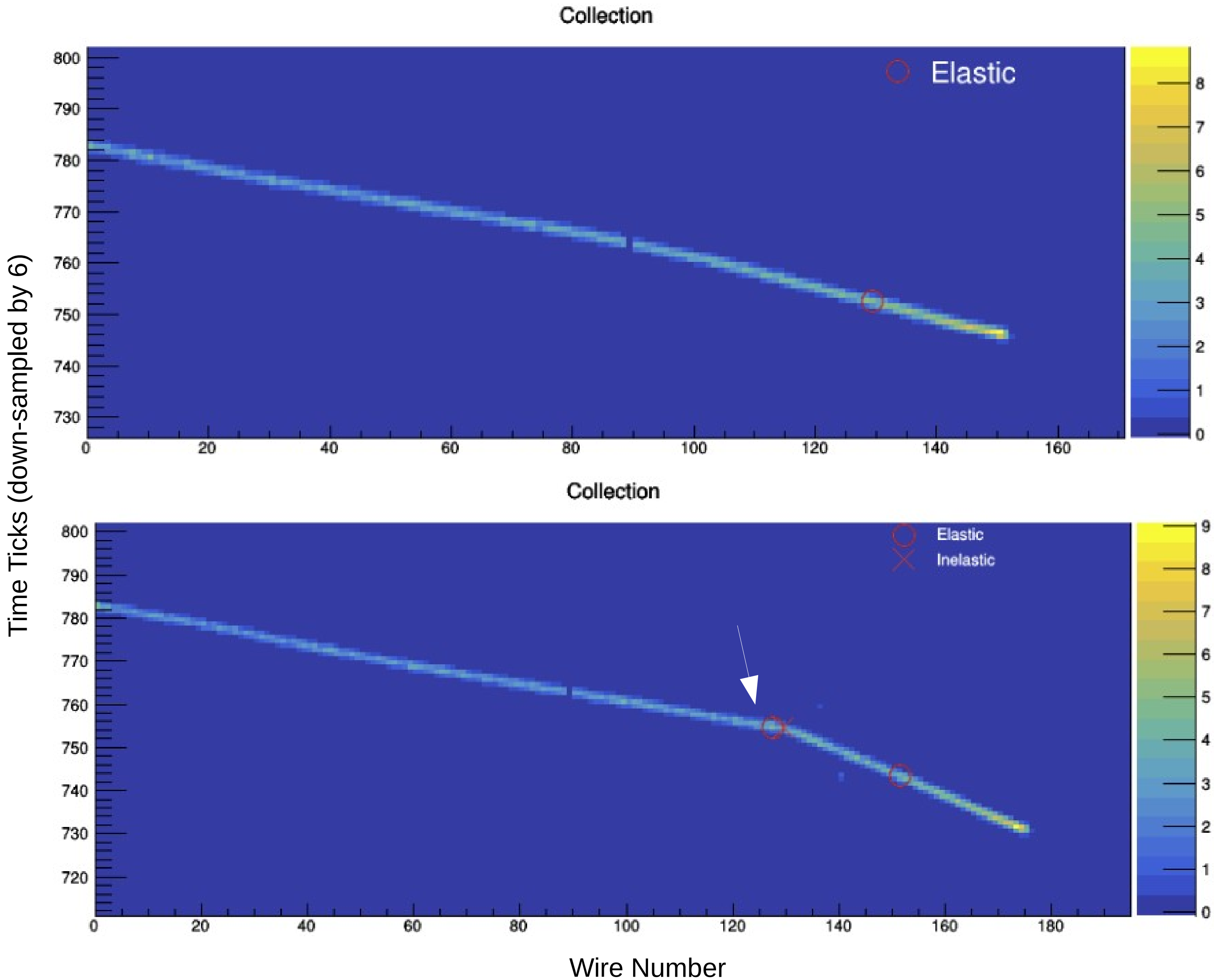
- Δr<=5 cm : Good reco. events

Hard Cases for Reconstruction [Elastic Scattering]



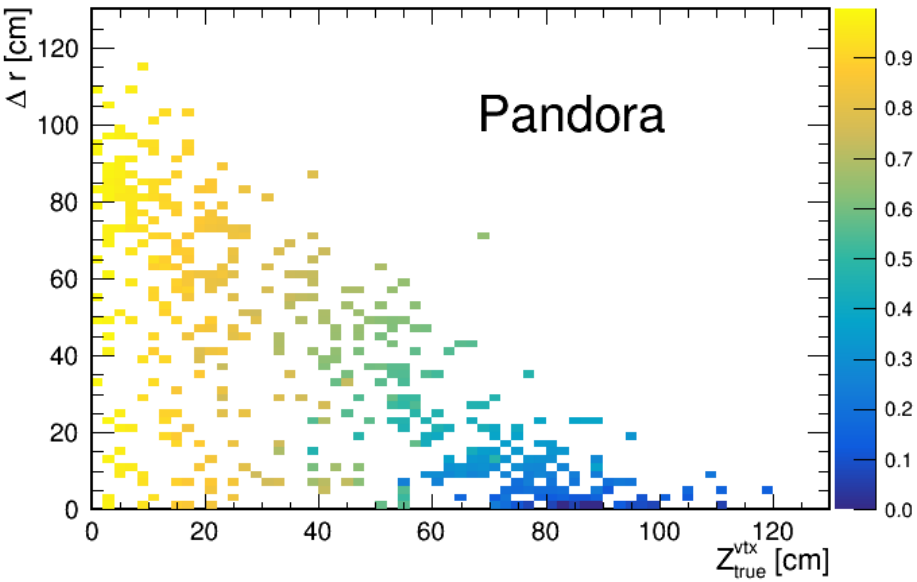
* True vertex showing in the figure, not reco vertex

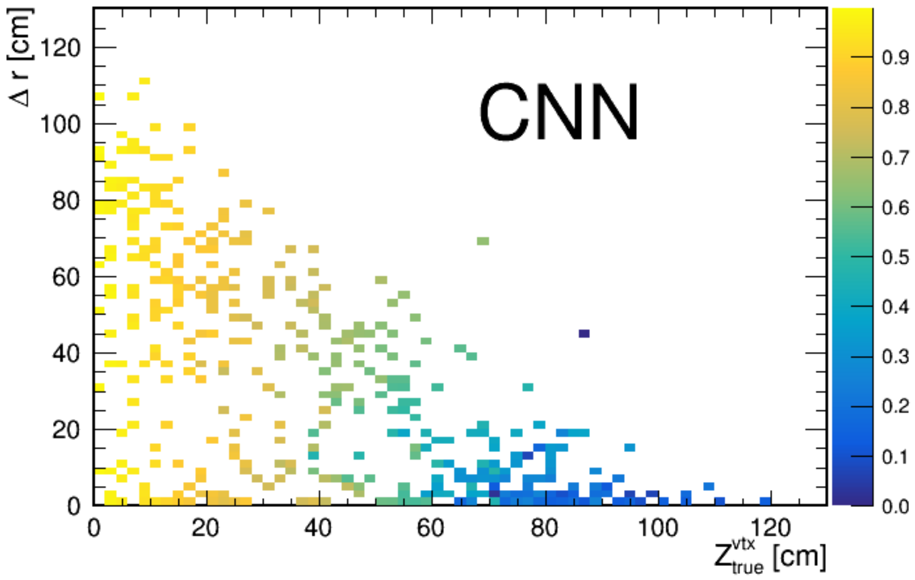
Hard Cases for Reconstruction [Elastic Scattering]



* True vertex showing in the figure, not reco vertex

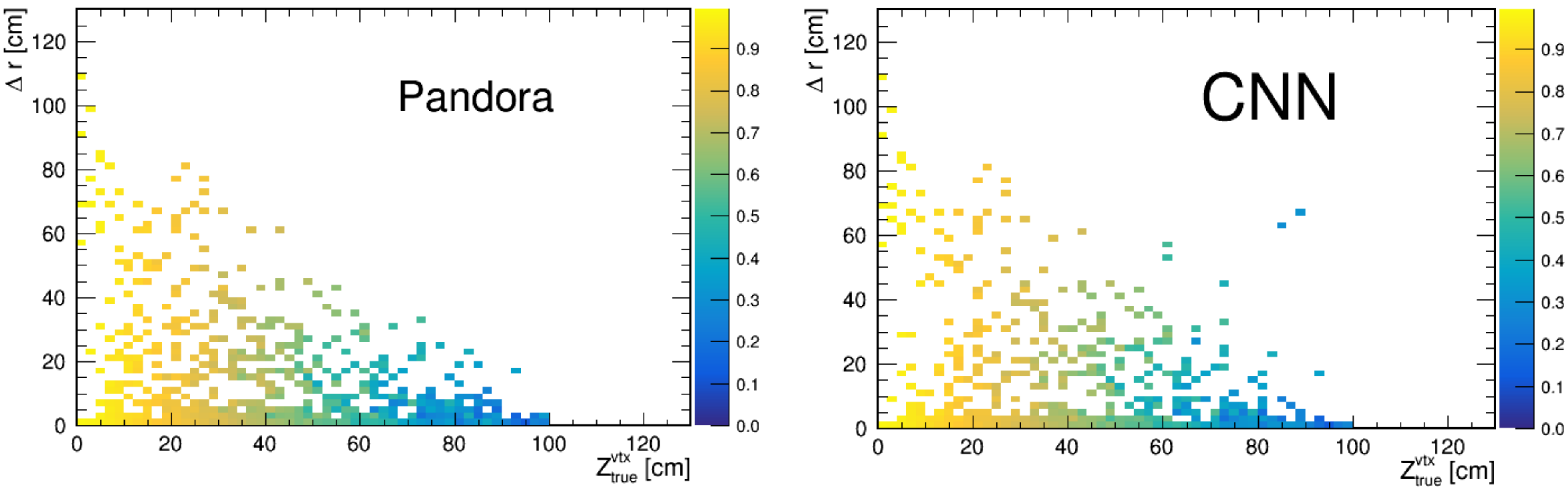
Position Dependency [Elastic]





- If the interaction vertex happens in the beginning of track (~< 20 cm)
  → Not easy to identify

Position Dependency [Inelastic]



- More uniform in Δr v.s. Z^{vtx}_{true}
- Better vertex identification for both Pandora and CNN

Summary & Outlook

- Improvement on elastic scattering using CNN
- Pandora does a great job on inelastic scattering

Next step

- Focus on inelastic XS [MC & Data]

- Keep improving the vertex identification
  - Add 2 induction views for training
  - Try other networks