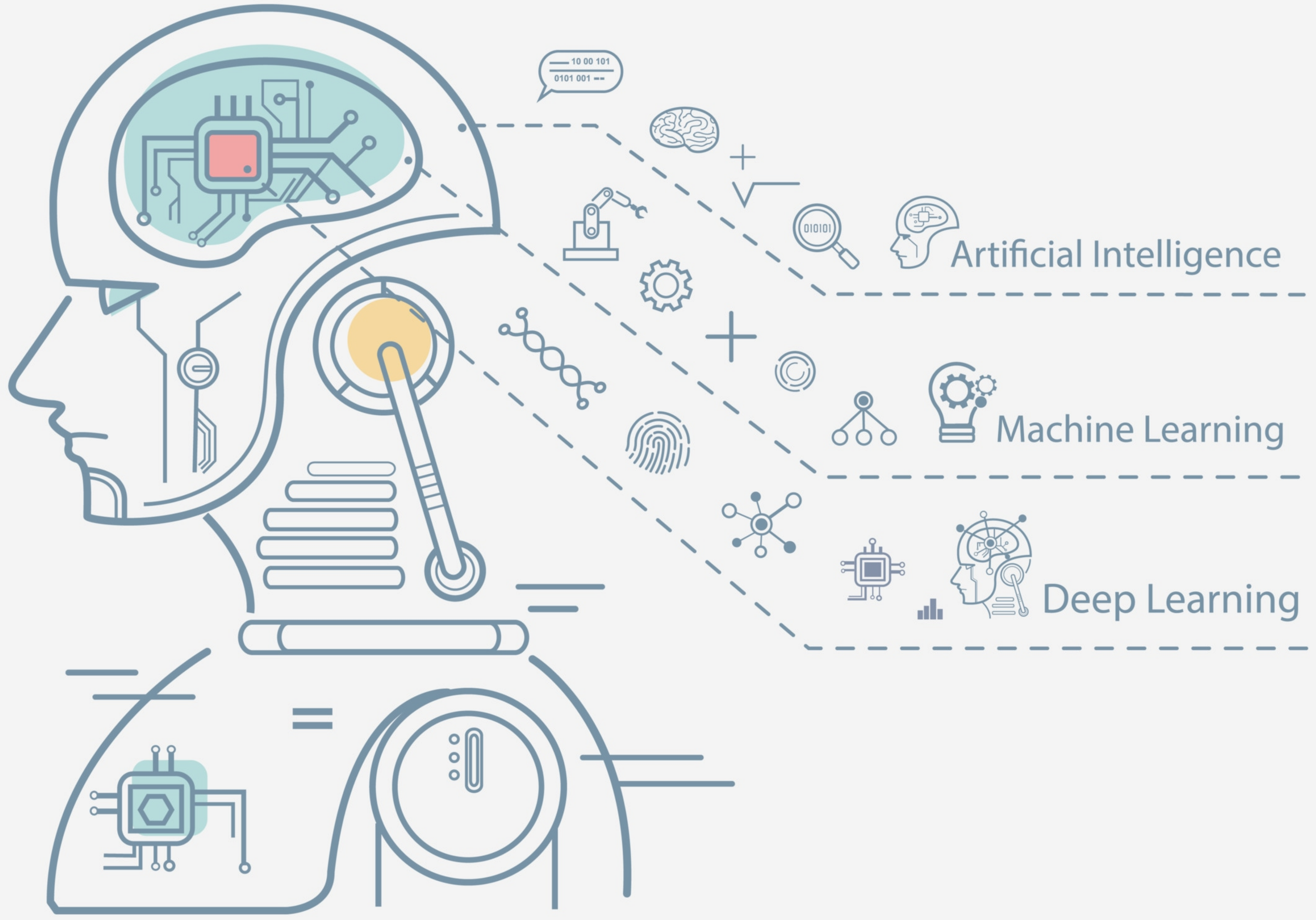
Tracking Particles using AI in CLAS12

Track reconstruction and identification with AI

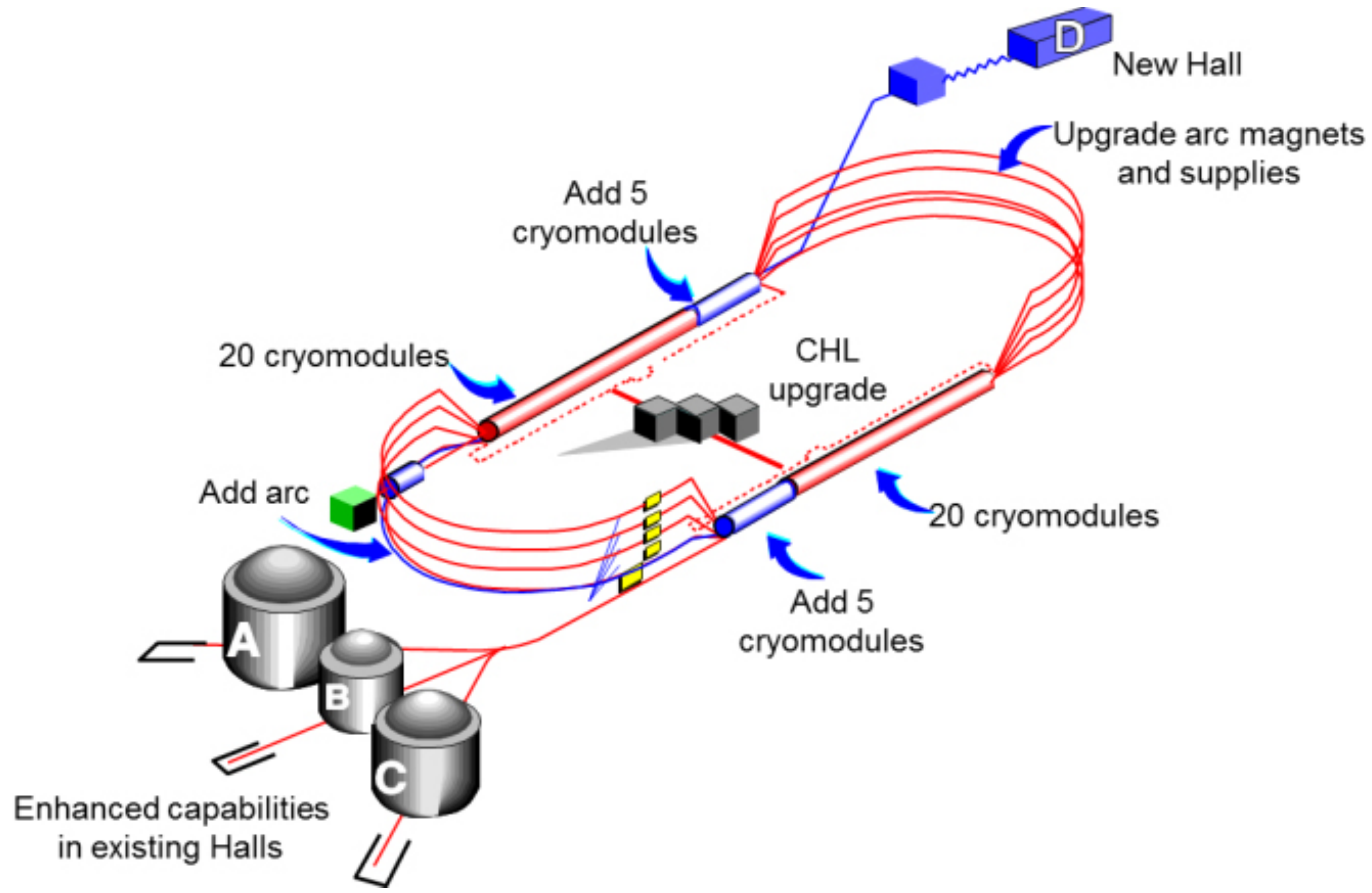
G.Gavalian (Jefferson Lab)



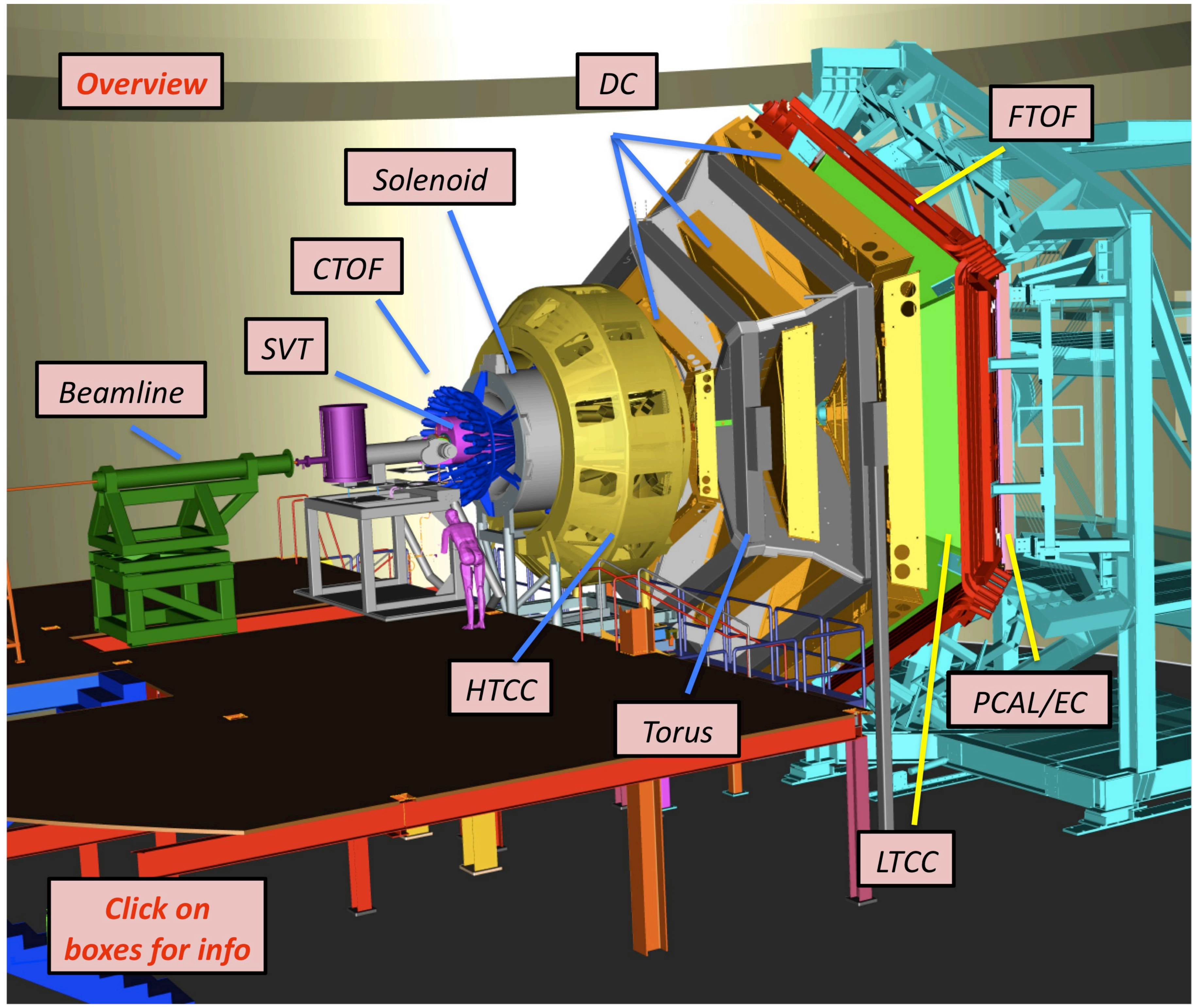
Angelos Angelopoulos (CRTC)
Polykarpos Thomadakis (CRTC),
Nikos Chrisochoides (CRTC)
Department of Computer Science,
Old Dominion University, Norfolk, VA, 23529

CLAS12 Detector

CLAS12 Tracking with Artificial Intelligence



Experimental Hall-B (CLAS12)

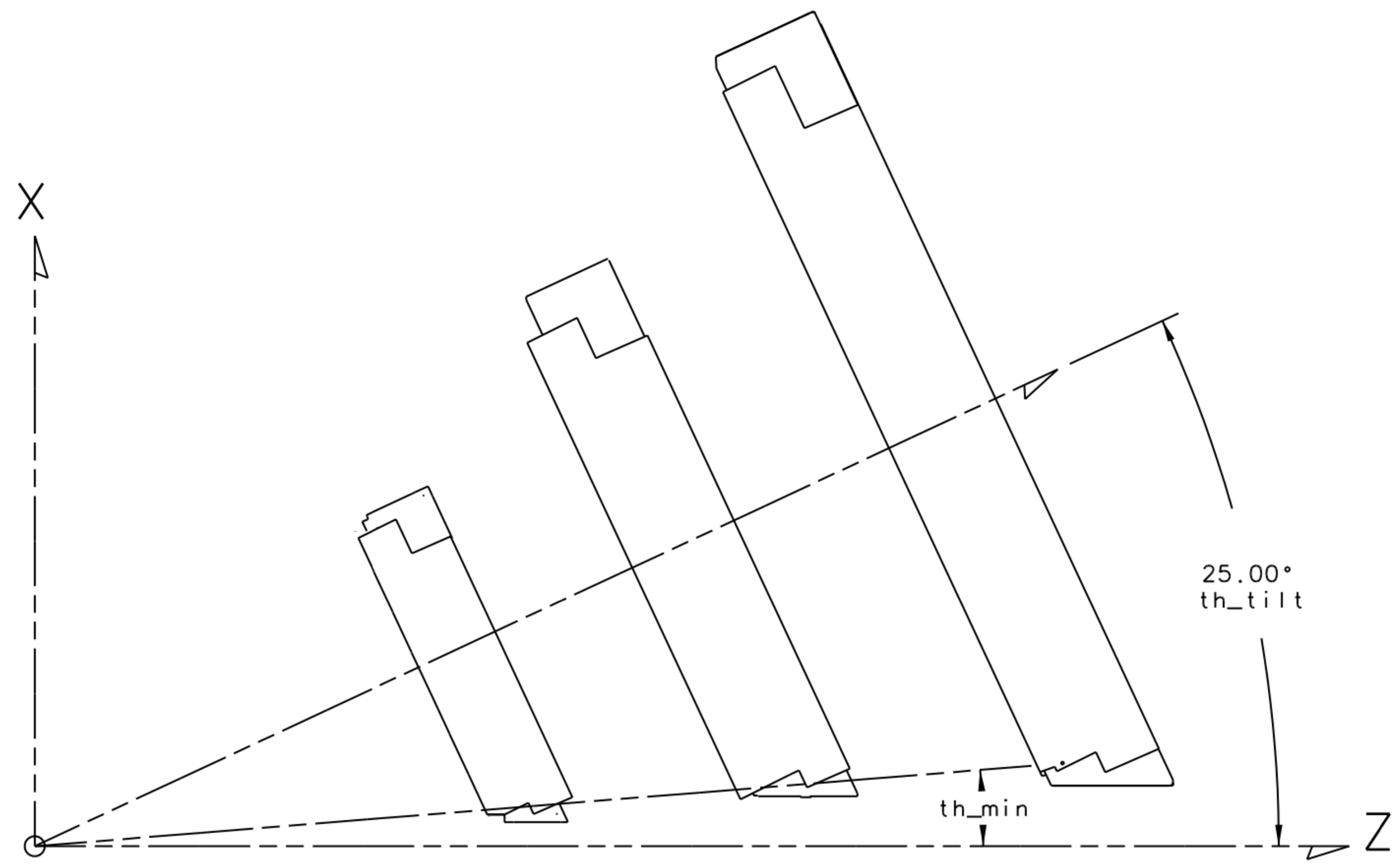


- Jefferson Lab accelerator recently upgraded to 12 GeV
- With addition of new experimental Hall (D)
- Provides beam for 4 experimental Halls

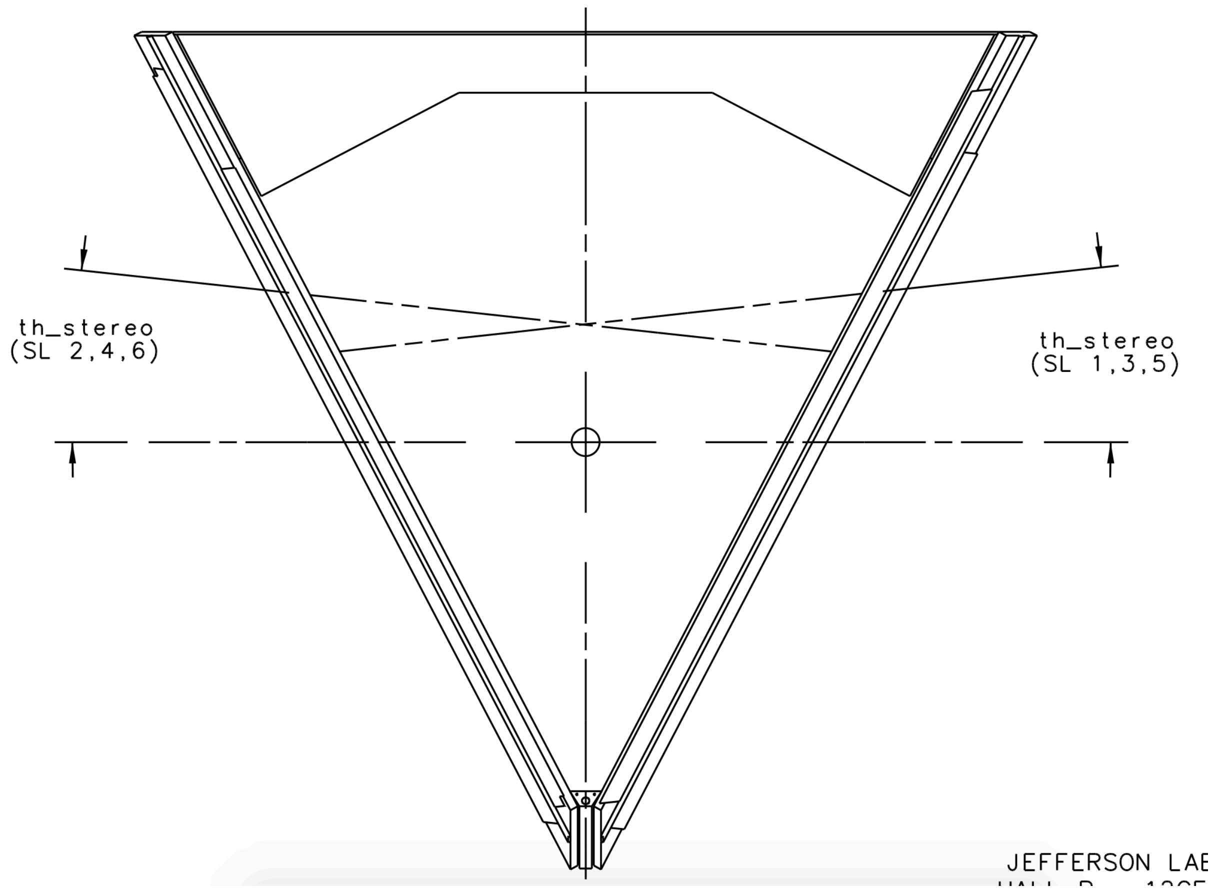
- 4 π detector consisting of 6 sectors
- Forward Drift Chambers are inside toroidal magnetic field

CLAS12 Drift Chambers

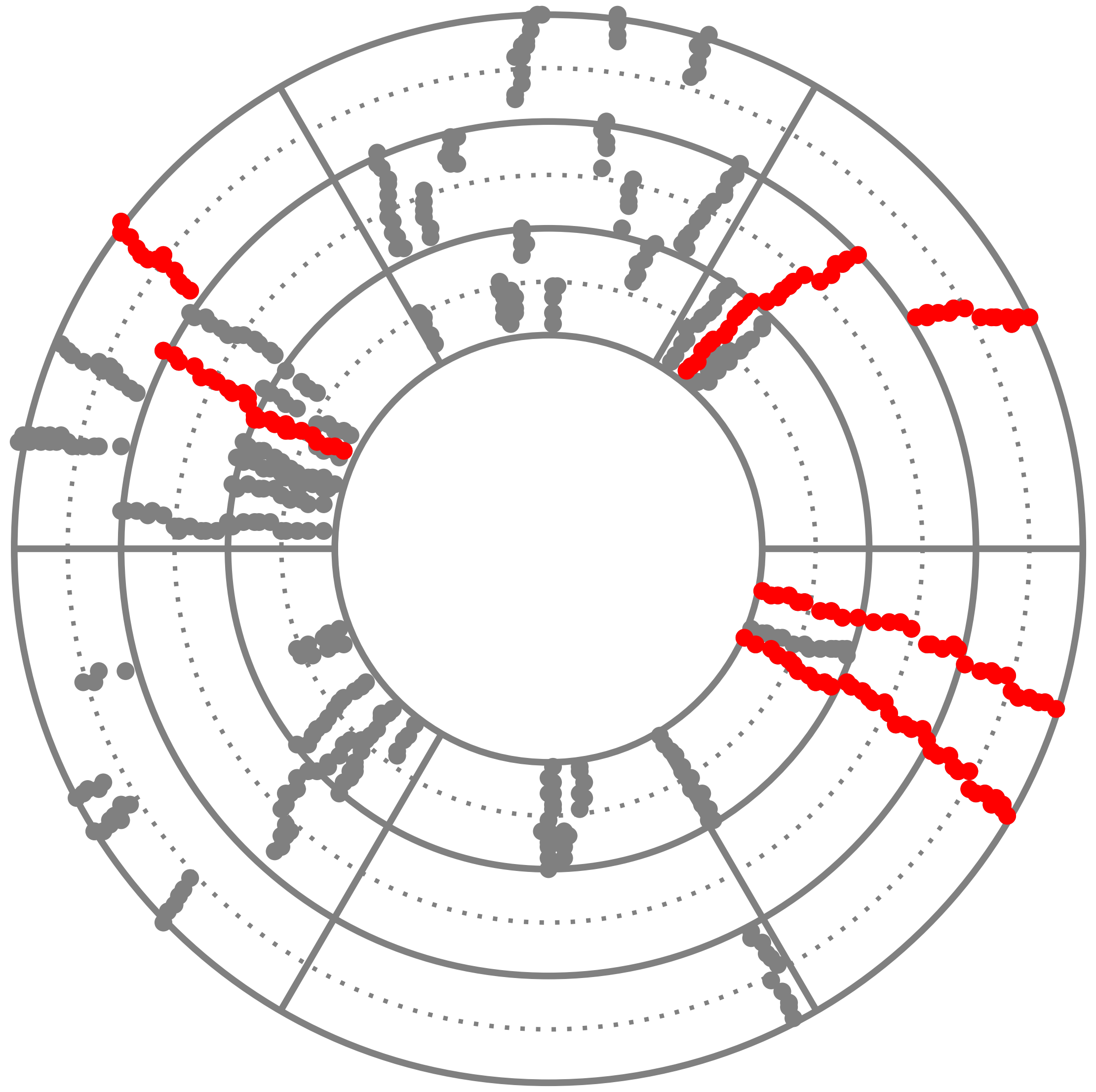
CLAS12 Tracking with Artificial Intelligence



- Drift Chambers consist of three regions (inside the magnetic field)
- Each region consists of two chambers (super-layers), each with 6 layer of wires
- Each layer in super-layers contains 112 wires.
- the wire planes in each super-layers in the region are tilted by 6 degrees relative to each other to detect position of interaction of particle in the cell.



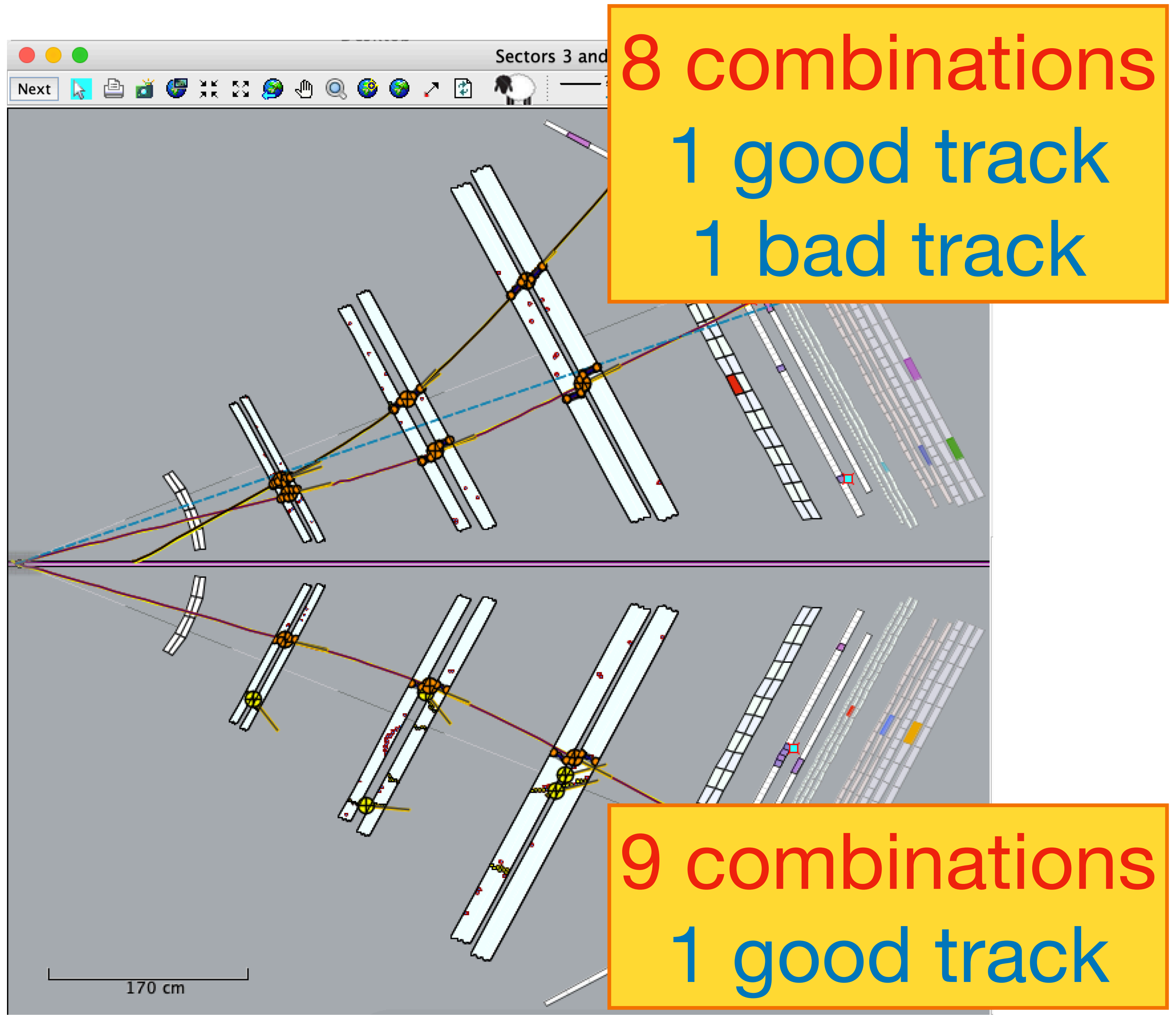
- Particle passing through drift chambers leaves signal in each layer
- There are background hits in the chambers that have to be considered by tracking algorithm.
- All combinations of segments in each super-layer have to be considered by tracking algorithm and fitted by Kalman filter to determine if it's a valid track.



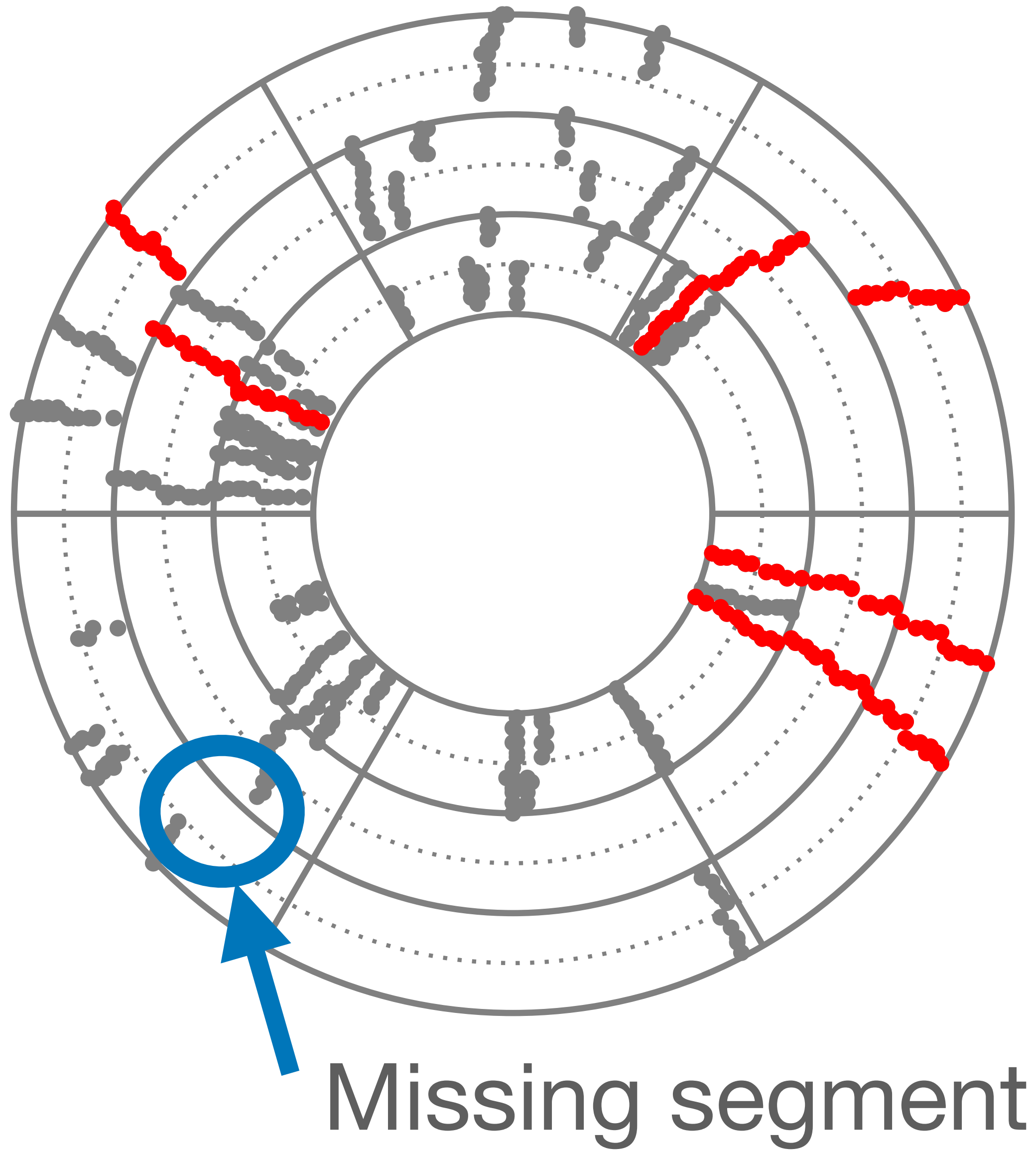
Tracking Challenges
CLAS12 Tracking with Artificial Intelligence

Combinatorics

- Tracking is computationally intensive (~94% reconstruction time)
- It relies on fitting tracks with Kalman-Filter
- Reduction of track candidates to fit can lead to significant speed up of the code.
- DC tracking with clusters:
  - Many combinations of clusters to form a track.
  - Many end up not as valid track, though time is spend on fitting them.
  - Even after fitting, some tracks are not traced to the target and have to be discarded.



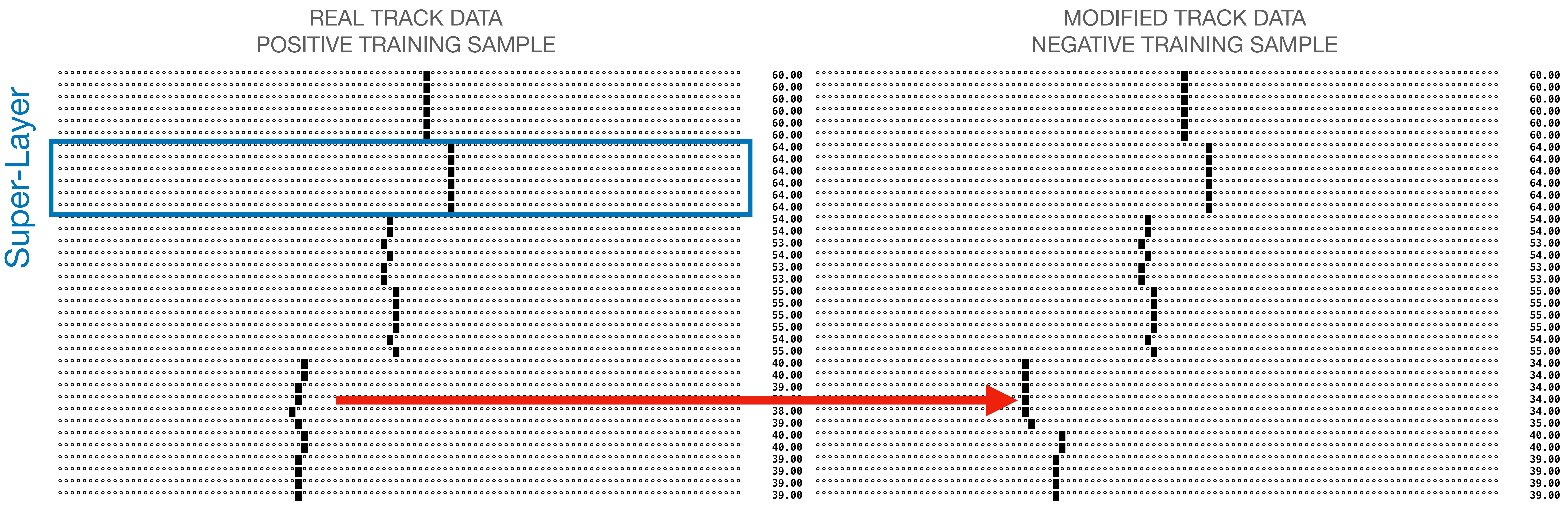
Missing Segments



- Inefficiencies and dead channels in drift chambers can leas to lost tracks
- Traditional tracking algorithm also considers 5 segment combinations and recovers tracks
- If track candidate detection is replaced with AI, a method has to be developed to address the missing segment issue.

Training Data

CLAS12 Tracking with Artificial Intelligence

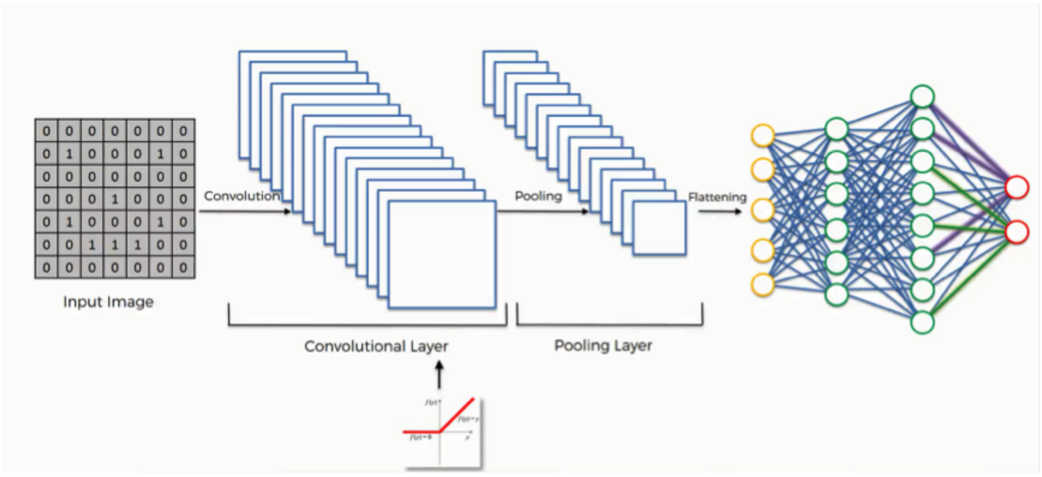


- Training sample is composed of real track data for positive sample and a modified track data where one of the segments is replaced with random segment in the drift chamber from the same event
- The segment is chosen to be closest to the track, since we found that network learns best when negative sample is very close to positive sample.
- For CNN an image with dimensions 36x112 was used, for ERT and MLP 6 features were used which are average wire position of the segment in each super-layer.
- (more details on how to chose training sample is in the published work, see Summary)

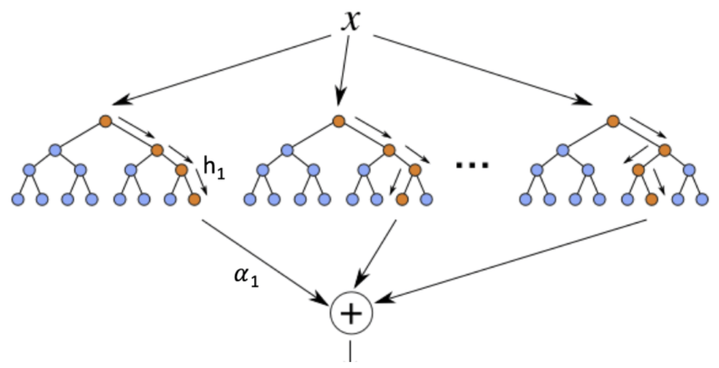
Training Data

CLAS12 Tracking with Artificial Intelligence

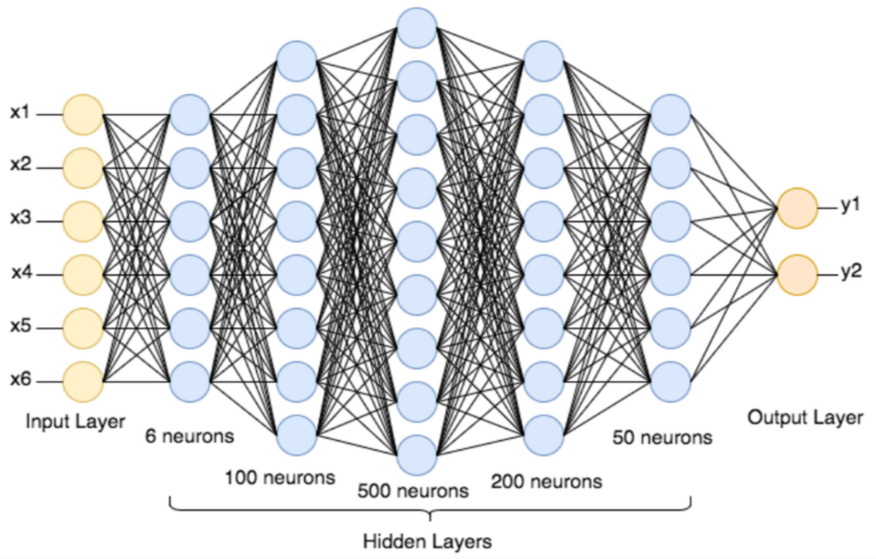
CNN



ERT

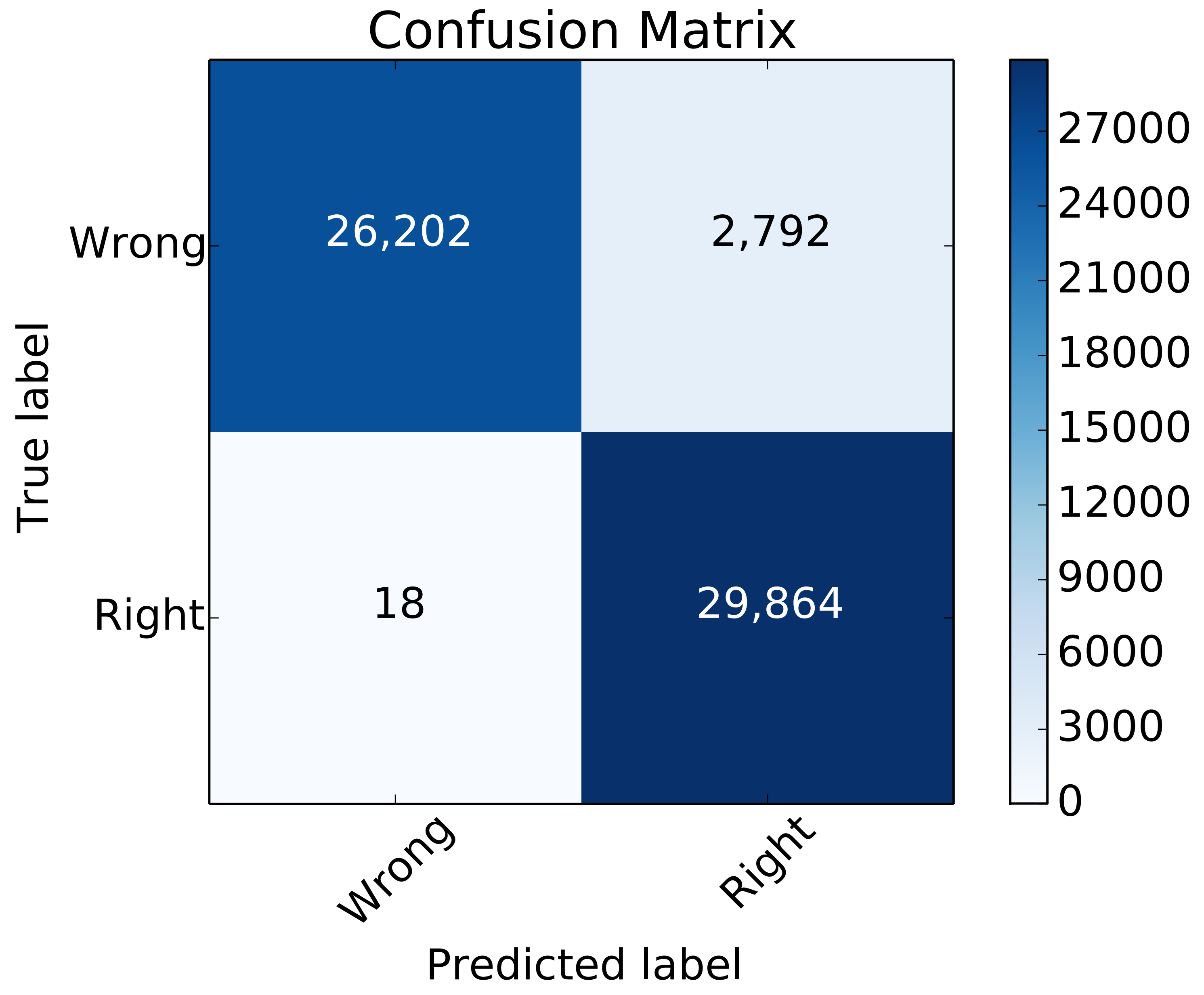


MLP



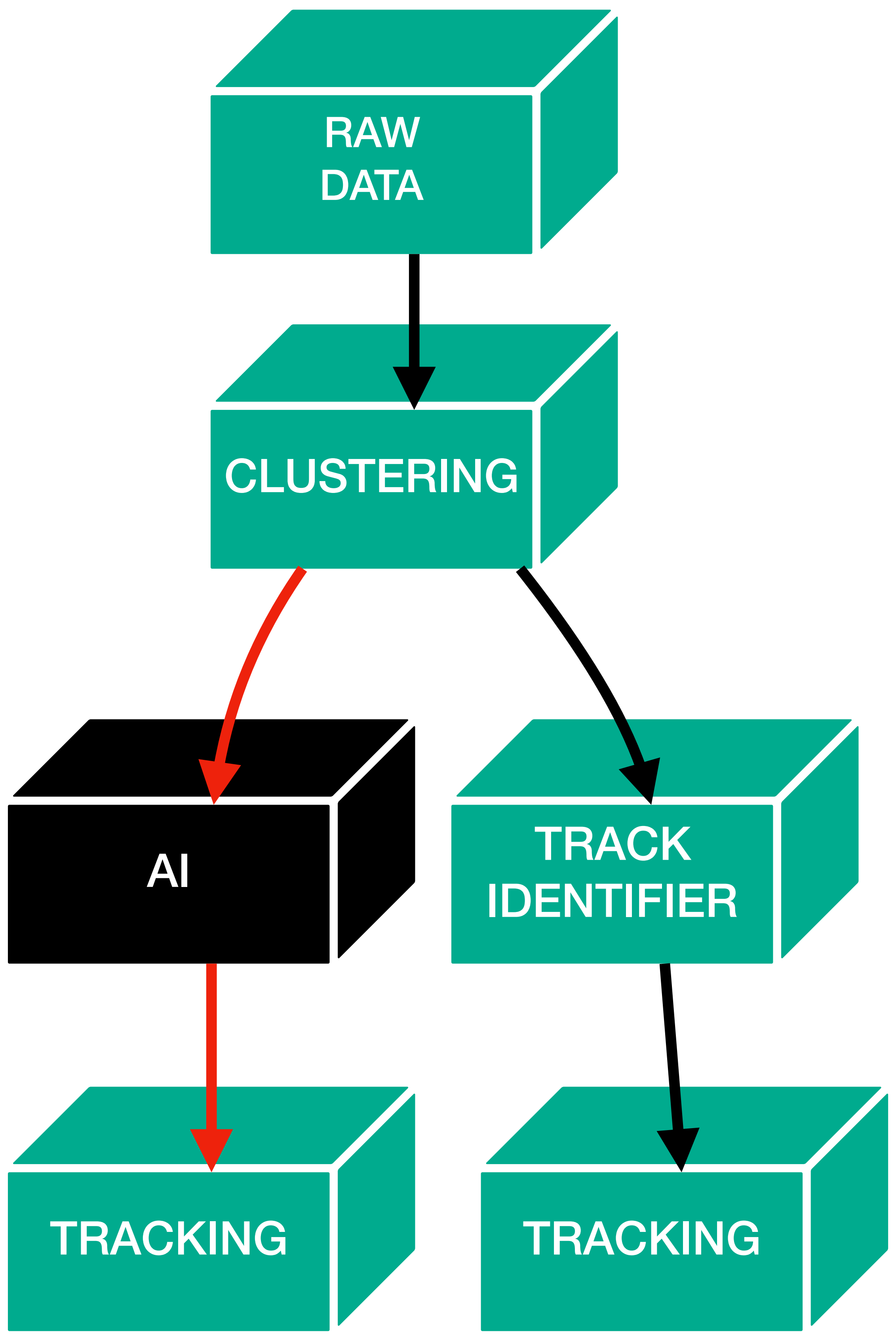
- Conventional metrics does not reflect true performance of the network.
- Training accuracy also reflects false positives, which are not crucial for our case as long as true positive is given higher probability of being a good candidate.
- New metrics was introduced to measure network performance on event-by-event basis (not track by track)
- Ah is the percentage of the "true" tracks in the event identified by network as good candidates with higher probability than other candidates in the same event.
- If two candidates have sharing segments the one with higher probability is taken.

Model Type	*A1* Metric	*Ac* Metric	*Ah* Metric	*Af* Metric	Training Accuracy	Time to Train	Time to Predict / sample
MLP	96.5%	20.2%	98.7%	1.3%	94.7%	252 *sec*	4 *µs*
ERT	93.3%	19.9%	91.9%	6.6%	99.9%	1.7 *sec*	5 *µs*
CNN	96.4%	30.1%	89.4%	3.5%	93.4%	457 *sec*	1.2 *ms*

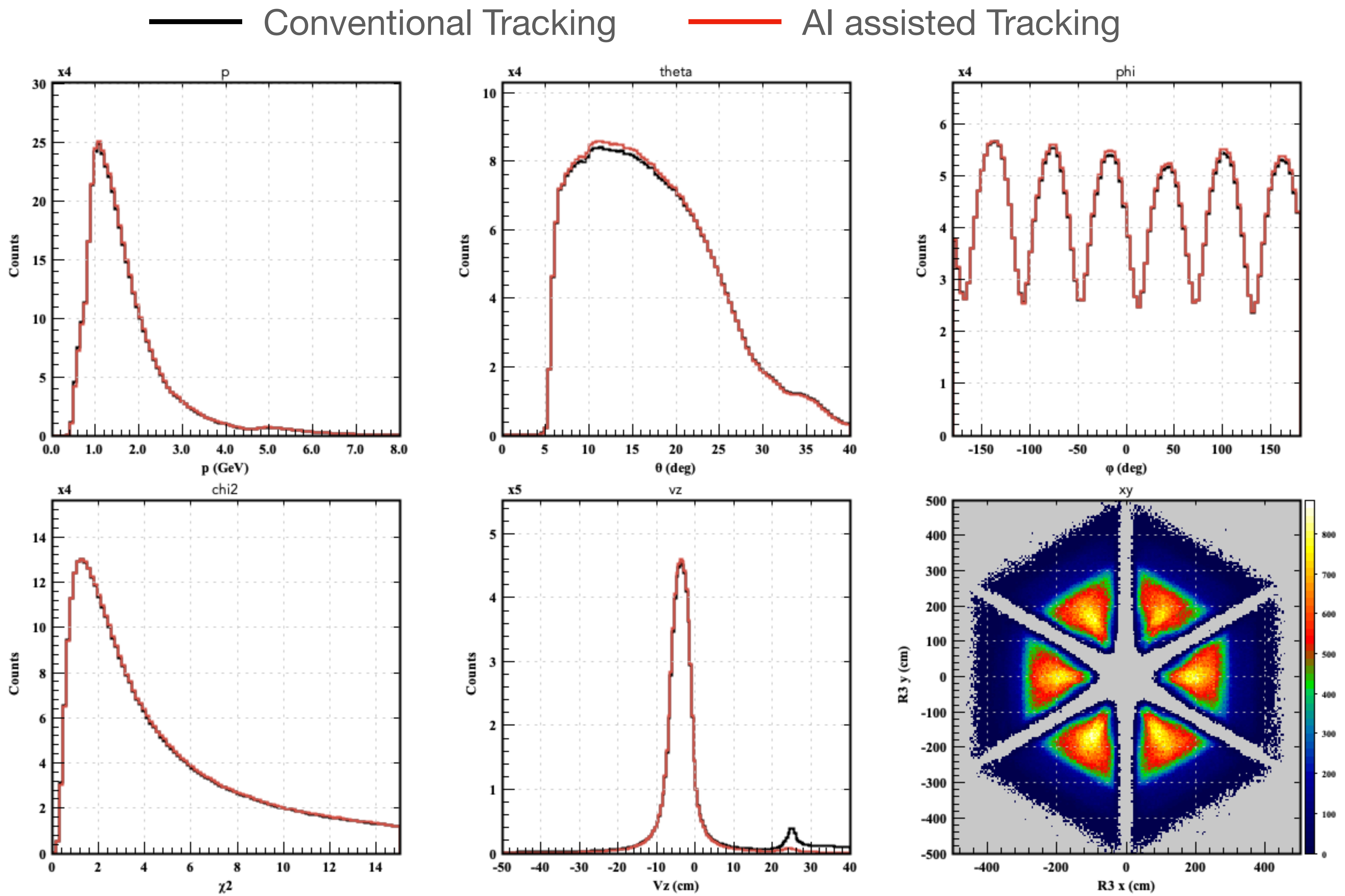


Results (Comparison of conventional and AI tracking)

CLAS12 Tracking with Artificial Intelligence

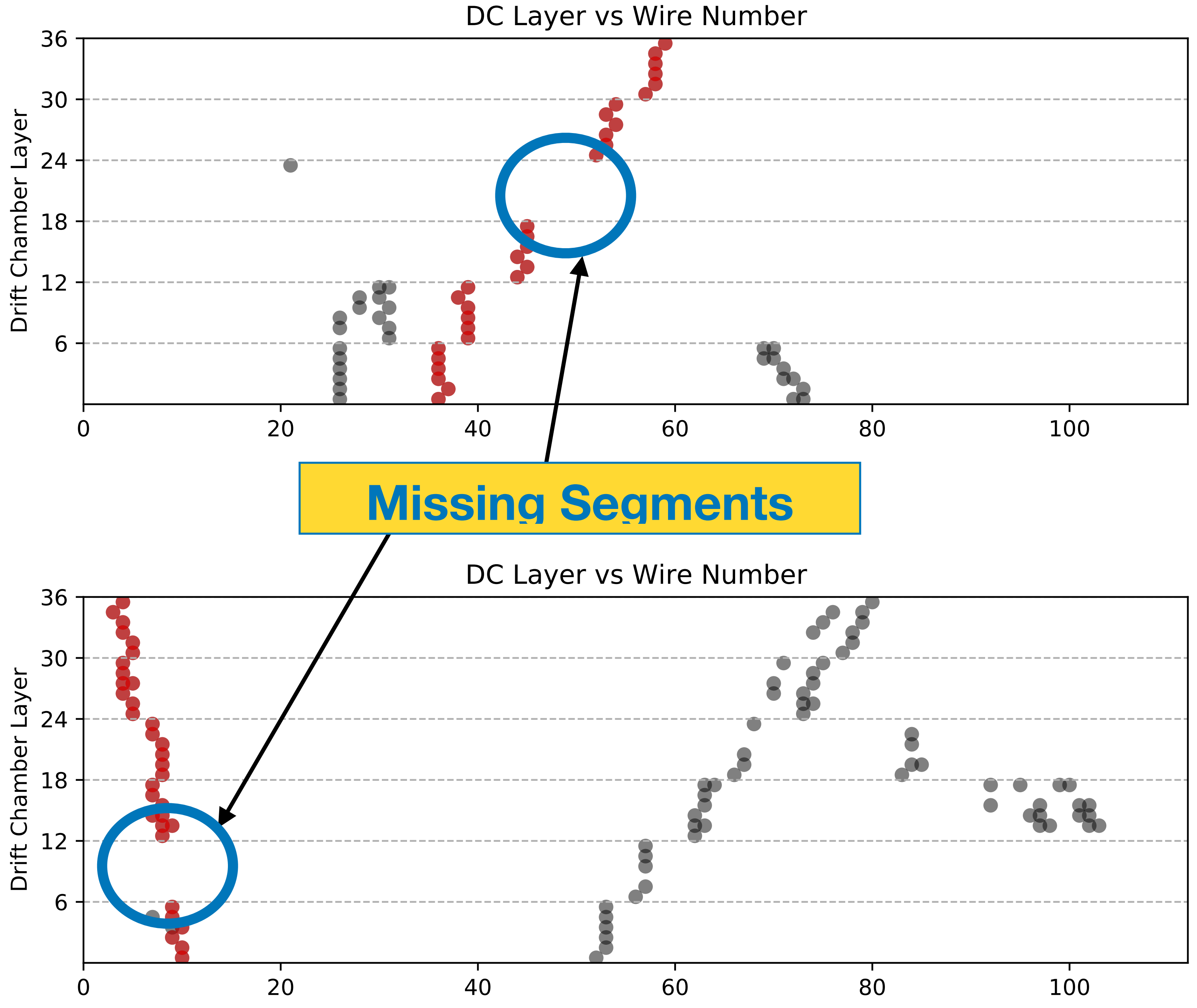


x4 Speed up

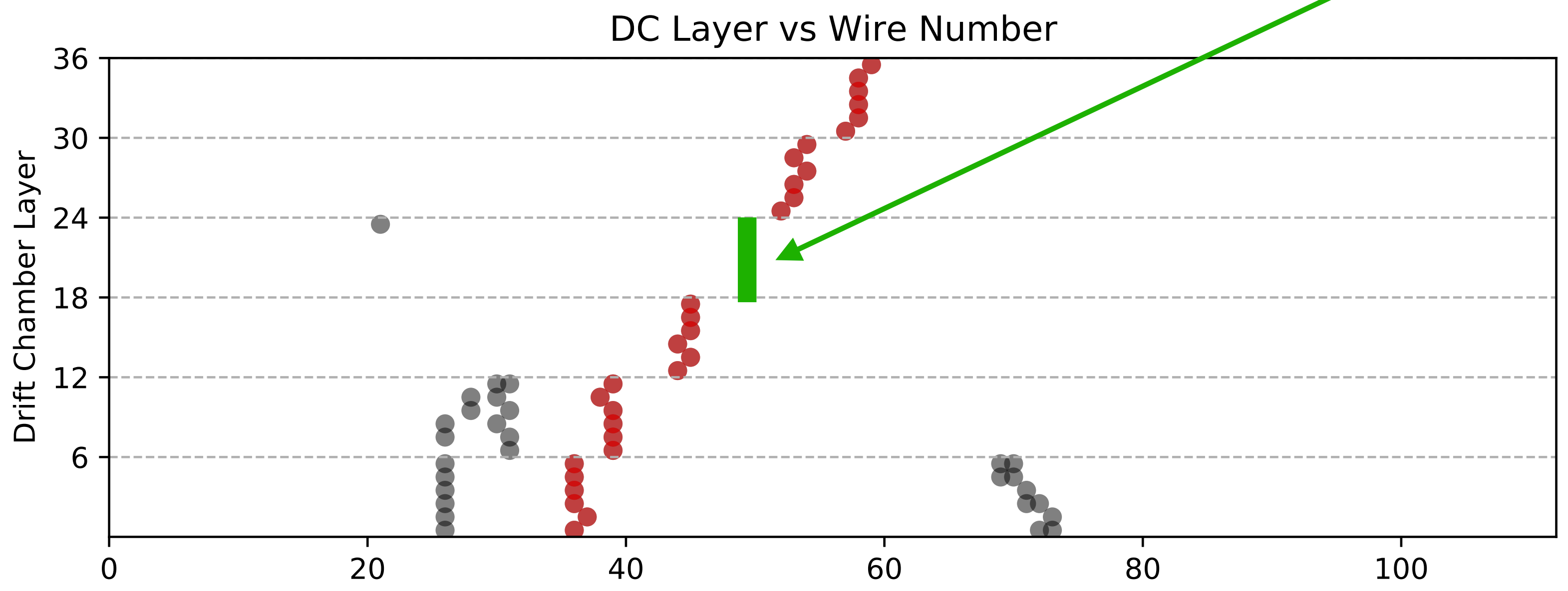


AI Tracking

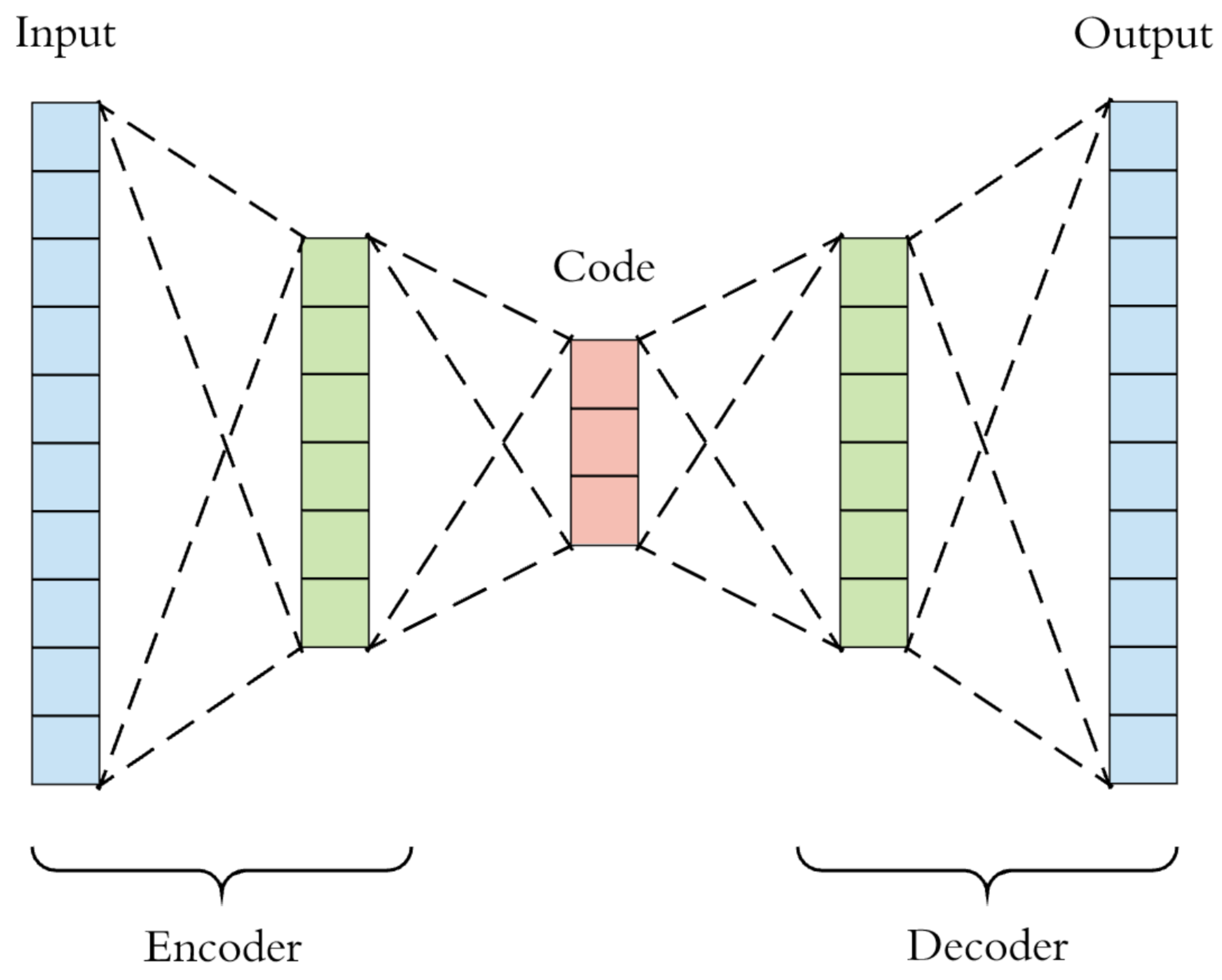
CLAS12 Tracking with Artificial Intelligence



$$X(x_1, x_2, x_3, 0, x_5, x_6) => Y(x_1, x_2, x_3, x_4, x_5, x_6)$$



- **Auto-encoders:**
  - An **auto-encoder** is a type of artificial neural network used to learn efficient data codings in an unsupervised manner. The aim of an **auto-encoder** is to learn a representation (encoding) for a set of data, typically for dimensionality reduction, by training the network to ignore signal “noise”
  - The input and output of encoder is vector of the same size. And it learns the output vector even if there is a corruption in the input.



- **What we want Network to do:**
  - Given combinations of 5 segment data network should predict the position of 6th segment
  - The 6 segment track candidate can be validated with track classifier.

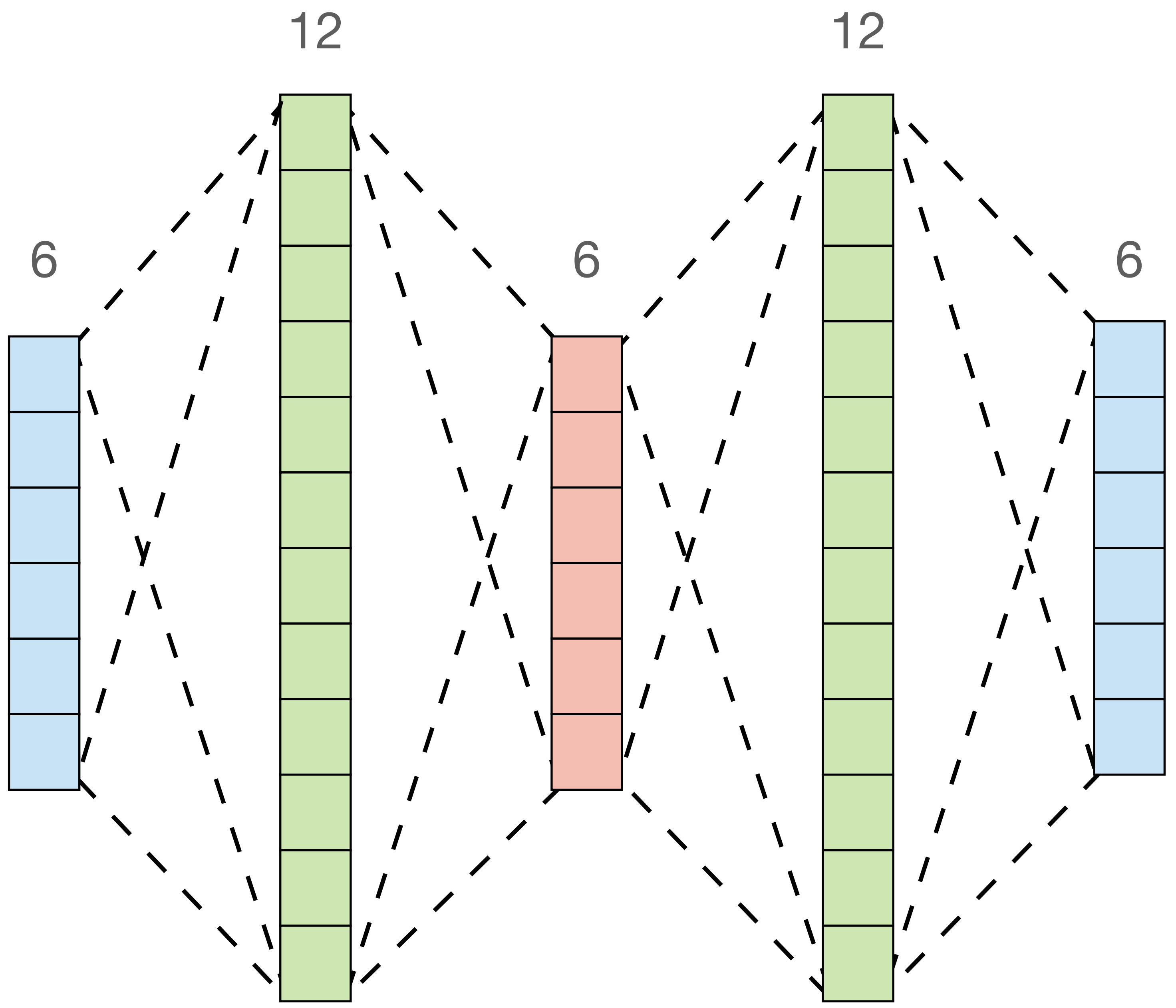
AI Tracking

CLAS12 Tracking with Artificial Intelligence

Introduction of Corruption into data set

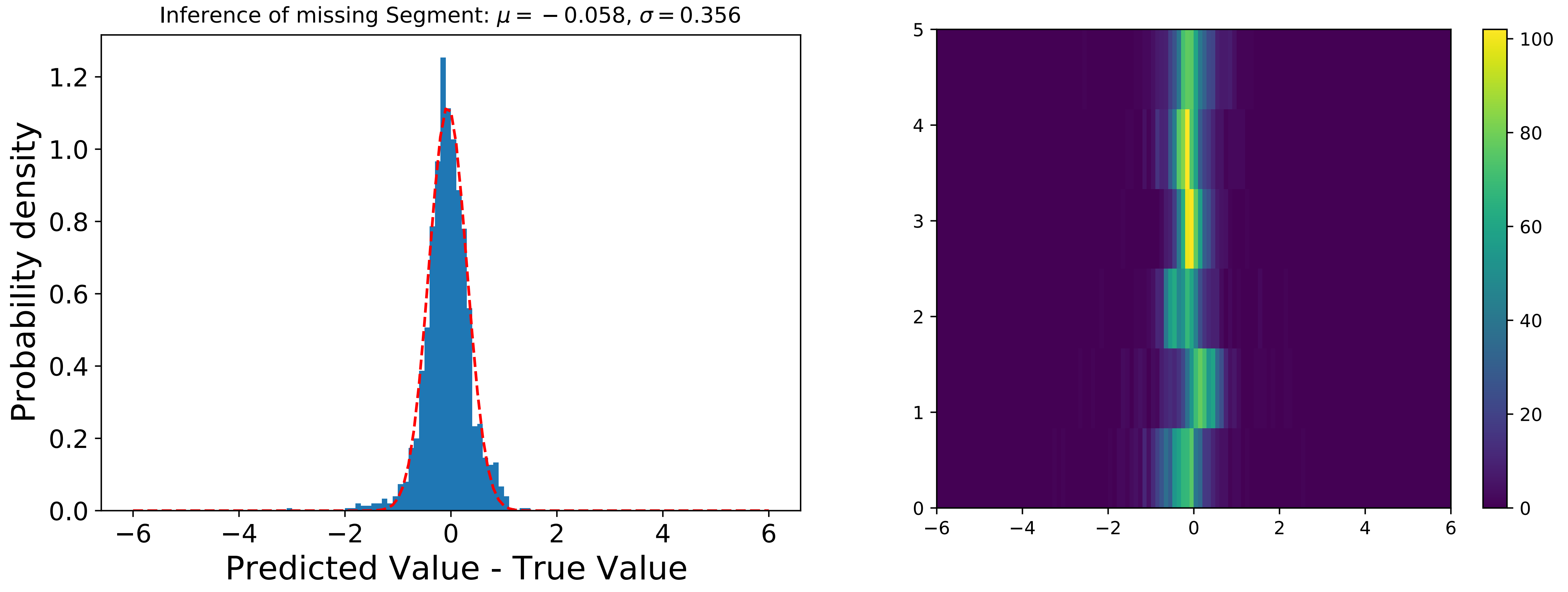
$$(x_1,x_2,x_3,x_4,x_5,x_6)\begin{cases} X(\underline{0.0,}\,x_2,x_3,x_4,x_5,x_6)\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \\ X(x_1,\underline{0.0,}\,x_3,x_4,x_5,x_6)\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \\ X(x_1,x_2,\underline{0.0,}\,x_4,x_5,x_6)\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \\ X(x_1,x_2,x_3,\underline{0.0,}\,x_5,x_6)\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \\ X(x_1,x_2,x_3,x_4,\underline{0.0},x_6)\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \\ X(x_1,x_2,x_3,x_4,x_5,\underline{0.0)}\rightarrow & Y(x_1,x_2,x_3,x_4,x_5,x_6) \end{cases} \tag{7}$$

Auto-Encoder Architecture



- Starting from fully reconstructed tracks (with 6 segments), we introduced corruption into training data set.
- xi represents the average wire position in the segment.
- The network was trained with corrupted data as input and original data as output
- The network was tested on 6 segment data by corrupting one of the segments and having network to reconstruct missing segment information.
- The average position of segment was reconstructed with accuracy of 0.36 wires.

AI Inference Results



Summary

CLAS12 Tracking with Artificial Intelligence

- Neural Network track candidate classification was developed and implemented into CLAS12 reconstruction code.
- Provides 99.7% track identification accuracy and results in x4 code speed up.
- Neural Network based on Auto-Encoders successfully corrects for inefficiencies in Drift chambers (recovers 99.8% of the tracks with missing segments)
- Track Classification using AI:
  - https://arxiv.org/abs/2008.12860
- Auto-Encoders for track reconstruction:
  - https://arxiv.org/abs/2009.05144
- There are ongoing work on denoting Neural Networks for Drift chambers using auto-encoders and LSTM for track segment reconstruction, soon to be published
- CLAS12 reconstruction is in JAVA, present work was done using DL4J libraries:
  - https://deeplearning4j.org

BACKUP SLIDES