LOOT results for primary vertex finding in BESIII inner detector

Ekaterina Rezvaya, *Pavel Goncharov, Yury Nefedov, Gennady Ososkov,*
Egor Schavelev, Aleksey Zhemchugov

As it was said in the report of Dr. A. Zhemchugov at the recent SPD meeting, the use of a special on-line filter based on deep neural networks is absolutely necessary due to the too high rate of receipt of event data in the SPD. Therefore, one of the important components of the online filter will be the LOOT program to determine the vertex of the event immediately BEFORE the search for tracks.

I have already talked about the structure and methods of training deep neural network LOOT, so now I will just remind you how the results of predicting the coordinates of the vertex were evaluated, what problems arose and how we managed to solve them.

Criteria for evaluating results:
Mean Absolute Error = MAE
Mean Square Error = MSE

First results and problems

- Cartesian coordinate system
- Not normalized

Initial data

Transform to cylindrical and normalize

$$Z = \sqrt{x^2 + y^2},$$
$$X = arctg\left(\frac{y}{x}\right),$$
$$Y = z$$

Train model

MAE = 0.009

MAE= $\frac{1}{n}\sum_{i=1}^{n}|x_i - y_i|^2$

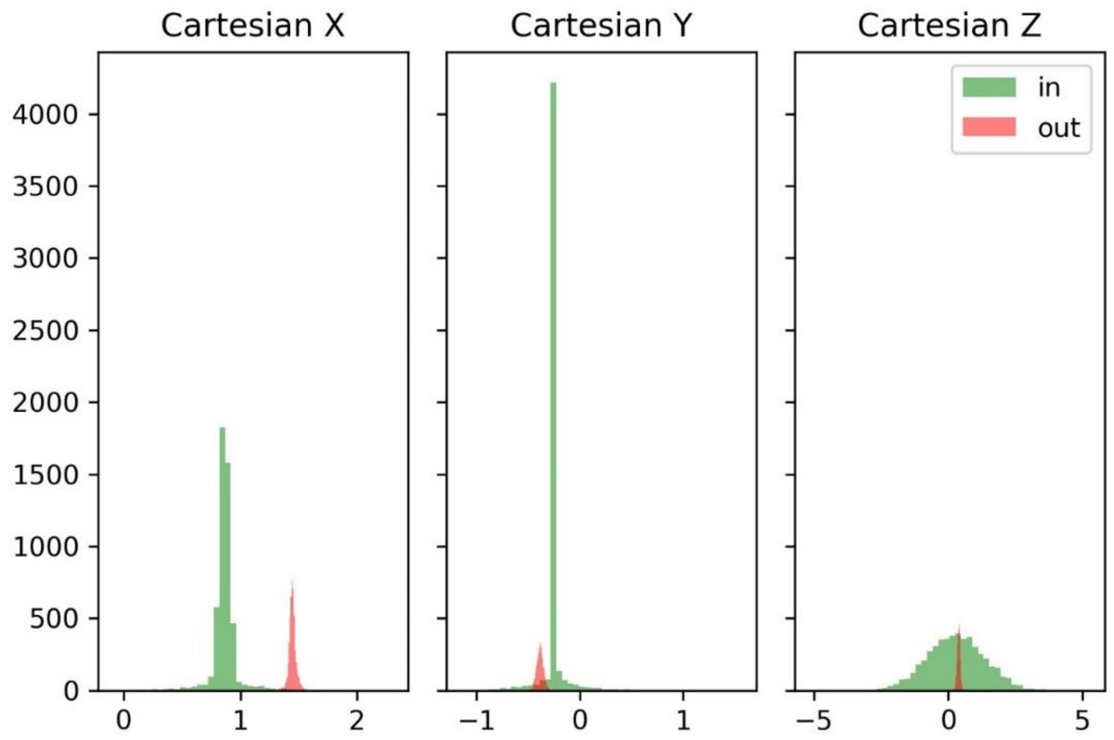
problematic to get the distance in the required units

The neural network model obtained at the previous stage predicted the primary event vertex with MAE = 0.009. The problem is that the error was calculated for normalized data, and it turned out to be quite problematic to get the distance in the required units.

Metrics and inverse transformation

To get the value of the metric in initial units, it is necessary that the metric accepts data in the same units as input. Therefore, to calculate MAE in cm, an inverse transformation was added when calculating metrics

The graph shows that the predicted vertex is practically one point with a small spread, moreover, with a large deviation from the real top. This shows that the trained model is not working correctly.



The MAE after recalculating the metric is 0.4 cm

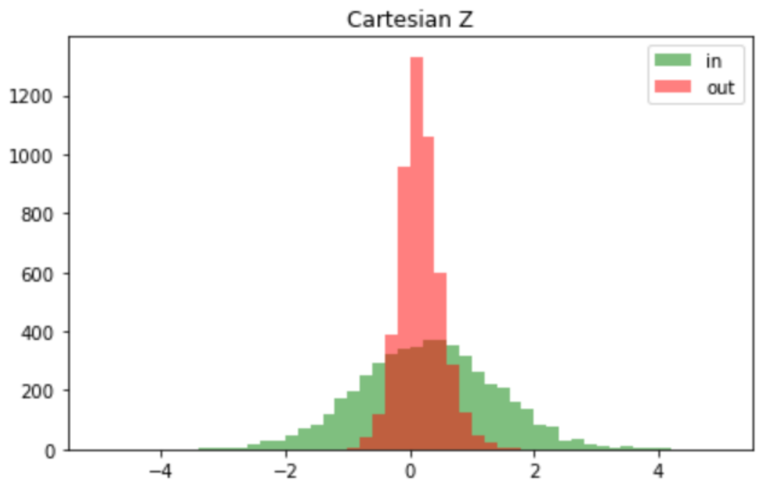
Loss-functions

Loss function takes two parameters as input - true and predicted values. The function shows how well the model performs in prediction. The higher its value, the worse the model works.

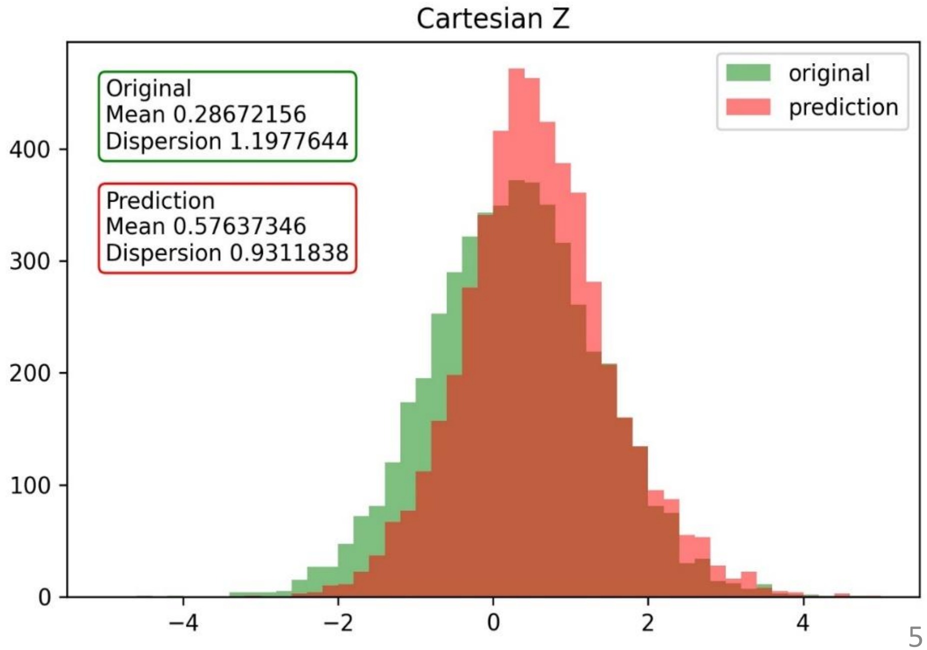
It is worth noting that due to the structural features of the BESIII collider, the X and Y coordinates are well known. Therefore, they can be fixed at the average value and make sure that the network finds Z more accurately.

However, shortening the coordinates for prediction didn't affect the quality of the result.

Further analysis of the data showed that all normalized coordinate values are close to zero. Squaring such small values in the loss function gives the function value close to zero. With such small values, the neural network easily falls into the local minimum during the optimization process. This prevents the network from making accurate predictions.



To solve this problem, it is necessary to give unnormalized values to the input of the loss function.



Ariadne: PyTorch Library for Particle Track Reconstruction Using Deep Learning

Ariadne – the first library for deep learning tracking on Python:

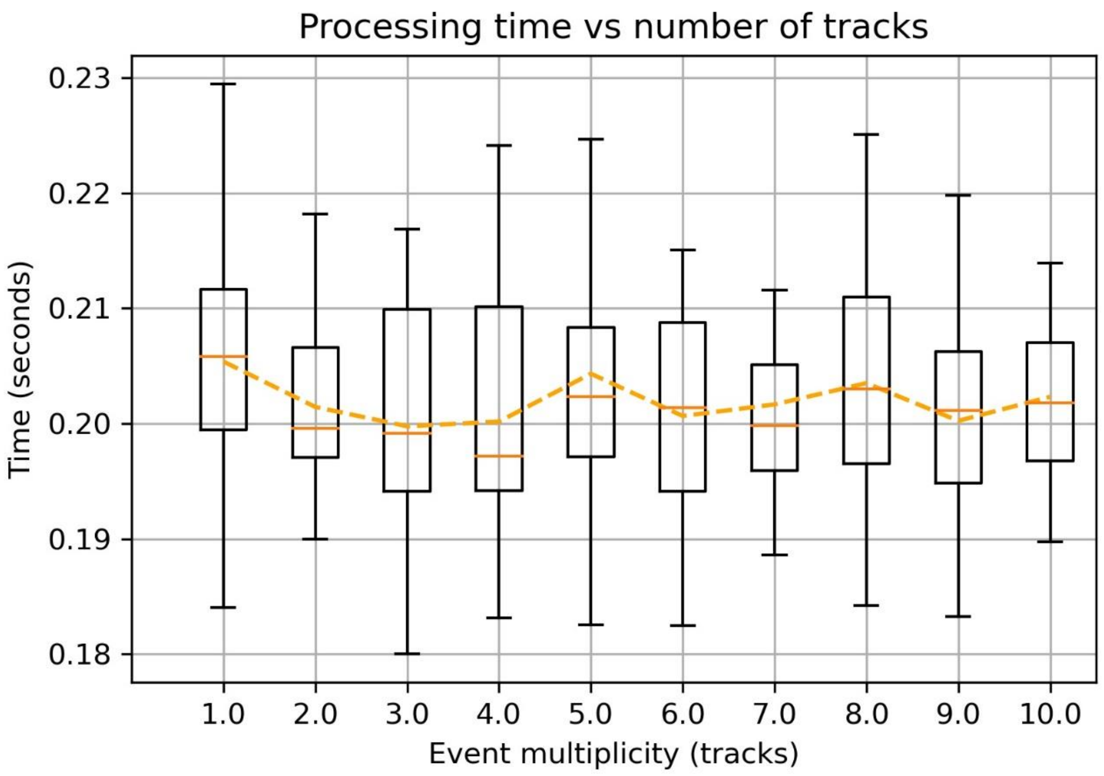
- ✓ any type of event data including collider and fixed-target experiments
- ✓ metrics logging, multiprocessing for data preparation, multi-GPU training
- ✓ open source and fully deterministic (https://github.com/t3hseus/ariadne)

Conclusion

As a result of the work, a model was trained that predicts Z coordinate of the primary vertex of the event with MAE = 1.15 cm

Running time of the trained model does not depend on the multiplicity of the event

The program code is included in the Ariadne library



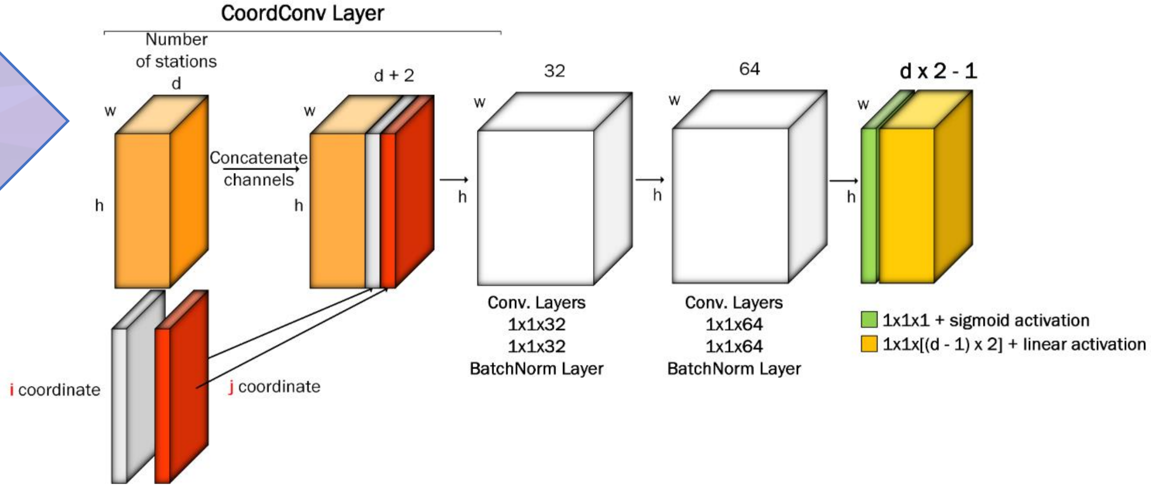
LOOT results for primary vertex finding in BESIII inner detector

Ekaterina Rezvaya, *Pavel Goncharov, Yury Nefedov, Gennady Ososkov, Egor Schavelev, Aleksey Zhemchugov*

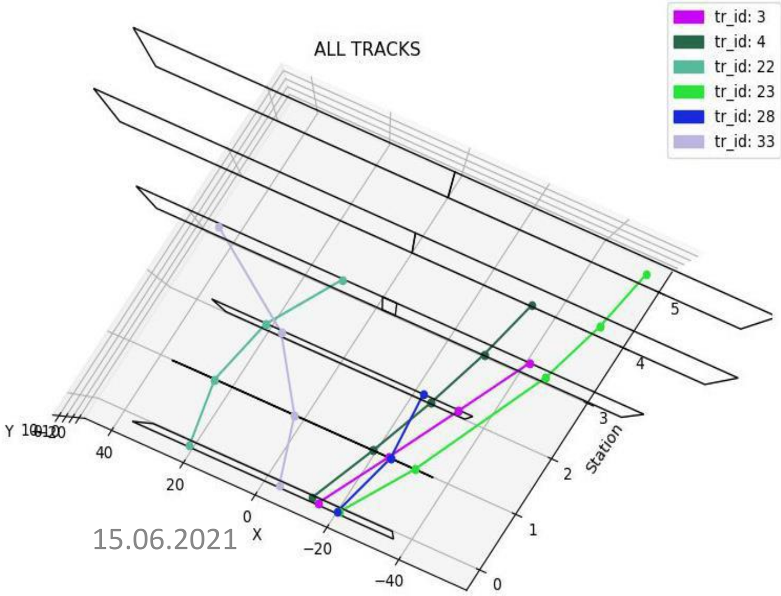
How to extend a convolutional neural network to represent a physical event

Using Look Once On Tracks (LOOT) model

Our main idea is to use OZ dimension instead of RGB channels – it's a radically new approach. Height and Width are the sizes of the largest station (most often the last).



See Goncharov et al http://ceur-ws.org/Vol-2507/130-134-paper-22.pdf



Event - image. Stations - color

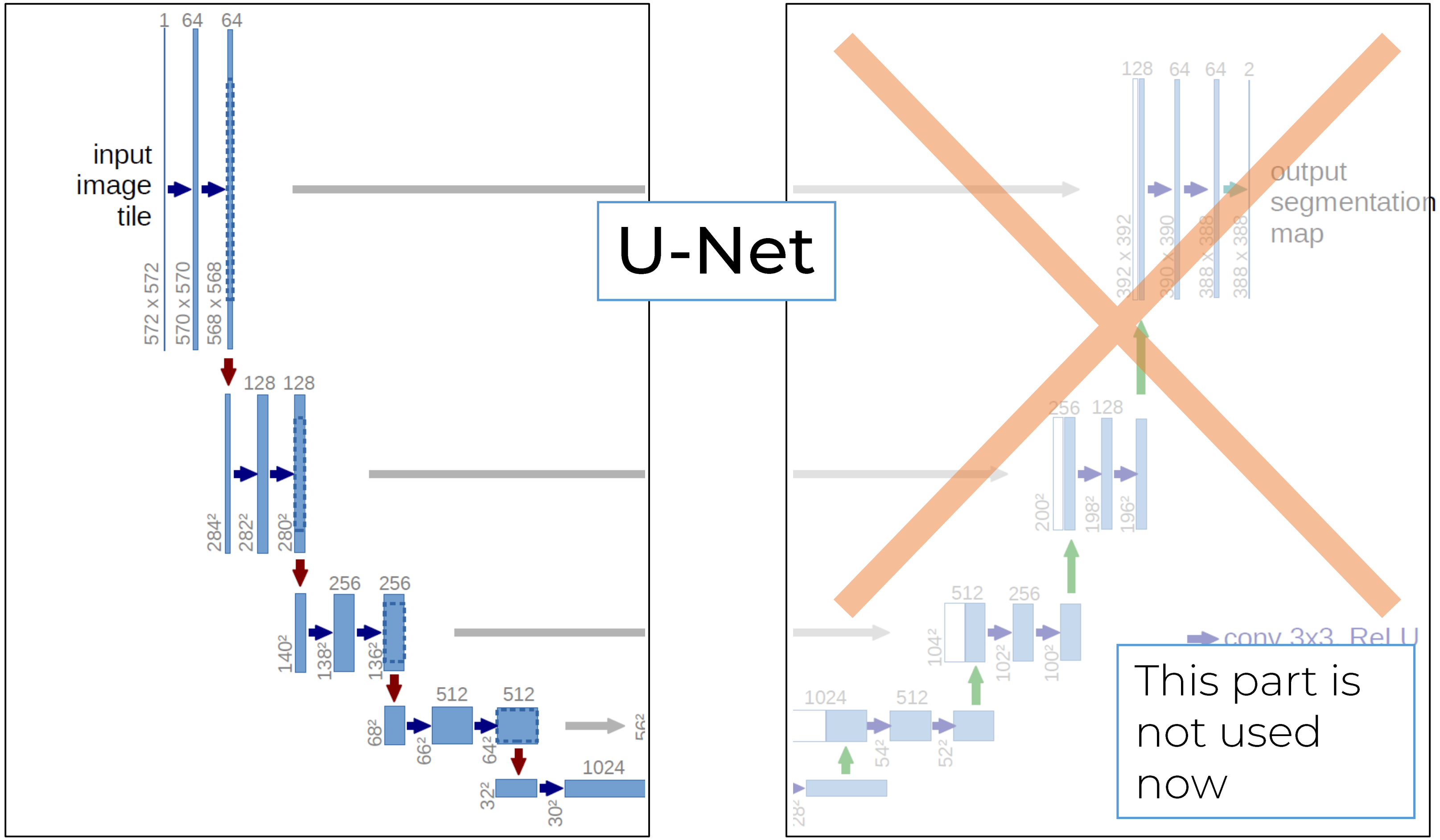
Images have 3D format: Height+Width+RGB

- ✓ Data from each station is a sparse matrix of zeros and ones, where ones indicate hits appearance
- ✓ Events have 3D format too: Height+Width+Stations

LOOT + U-net architecture for vertex prediction

U-Net is a convolutional neural network that was developed for biomedical image segmentation.

Network consists of a contracting path and an expansive path, which gives it the u-shaped architecture.



+