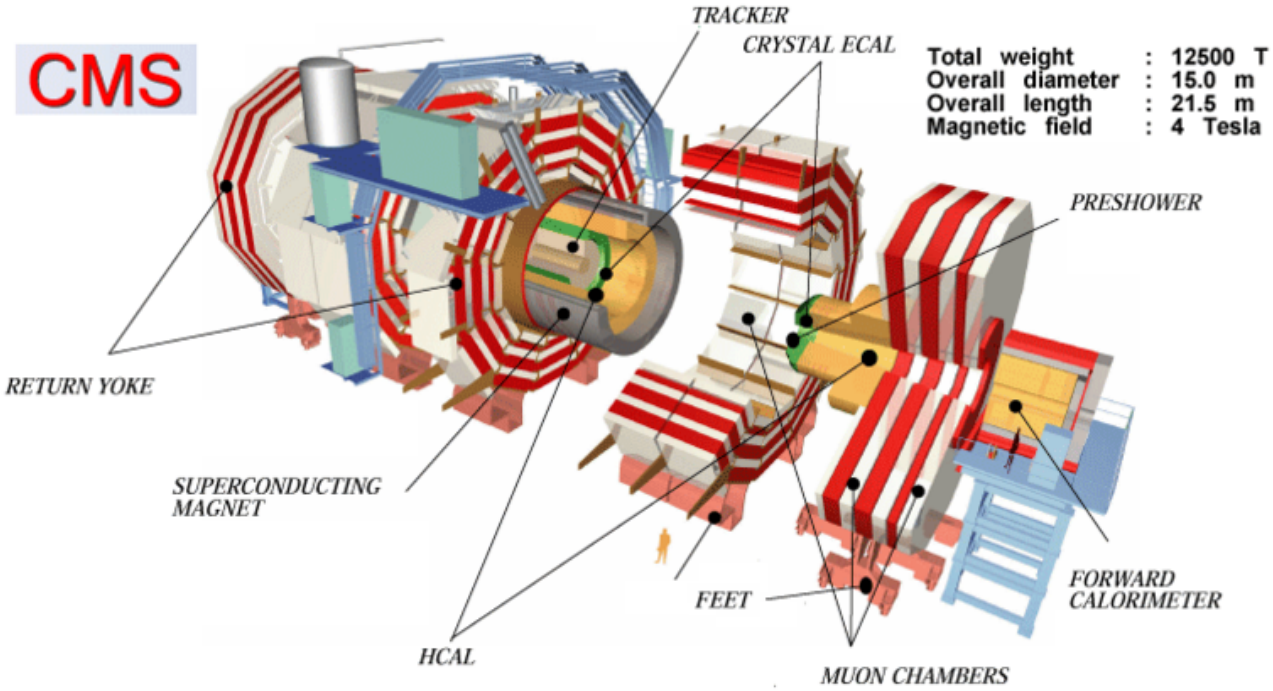
Particle tracking with graph neural networks

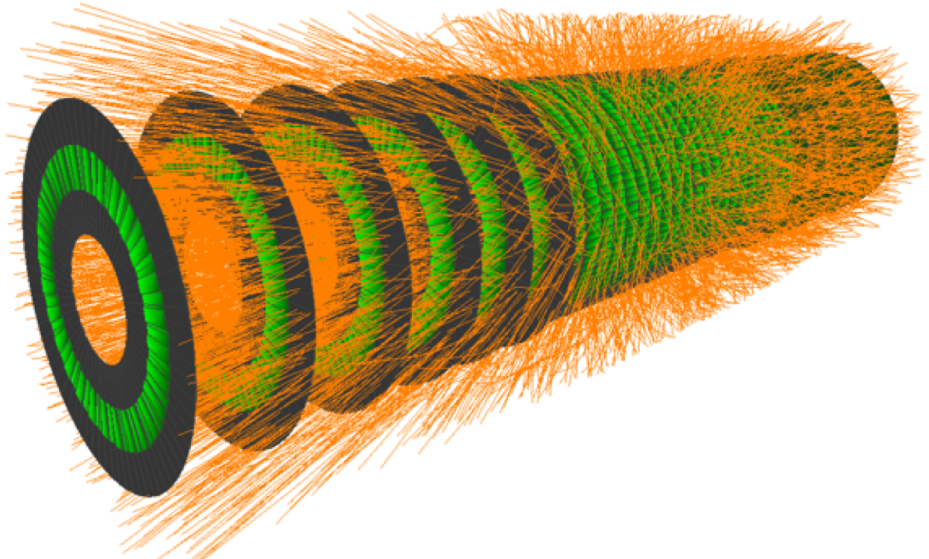
Shulgin Egor*, Ratnikov Fedor

"Mathematical methods of pattern recognition" Conference
Data Mining Section
Moscow, Russia

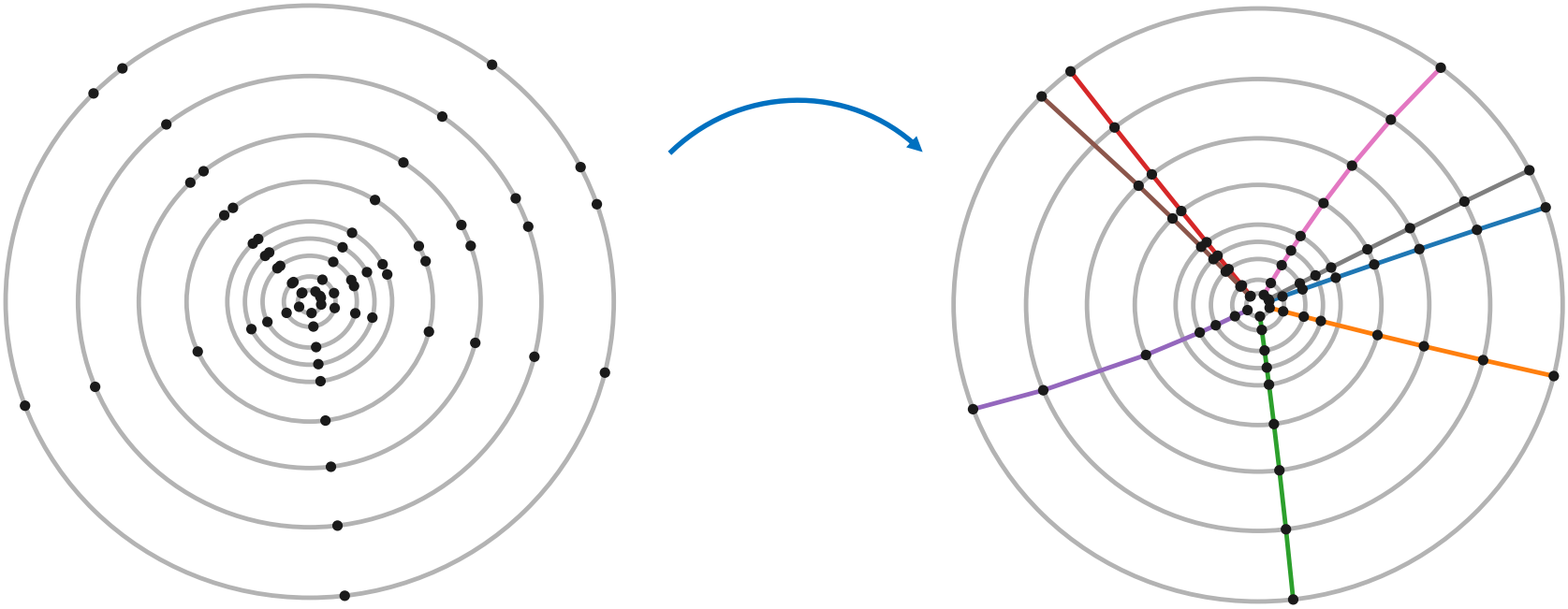
November 27, 2019

Introduction to the problem





Particle tracking problem



Machine learning approach proposal

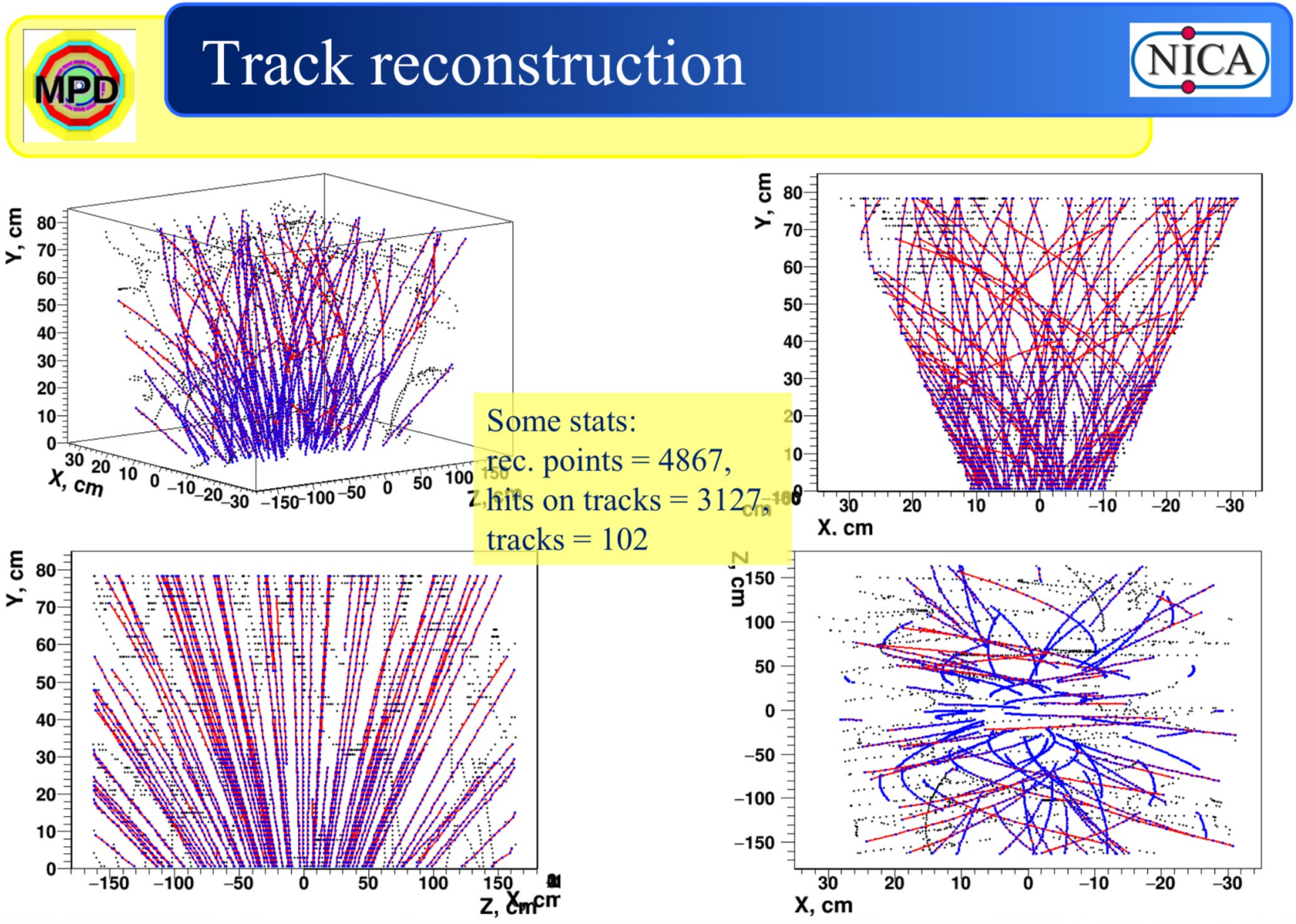
Treat this problem as a clusterization task.

Object space: hits $\{x_1, \ldots, x_n\} \in X \subset \mathbb{R}^3$

Clusters: tracks labels $Y \subset \mathbb{N}$

Required: reconstruct mapping $f : X \to Y$

Real data example

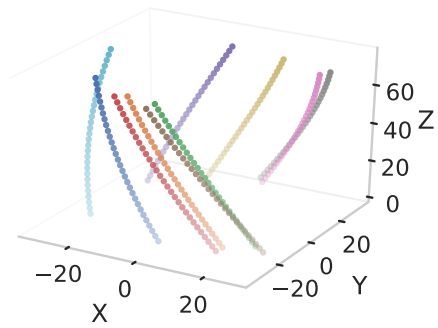


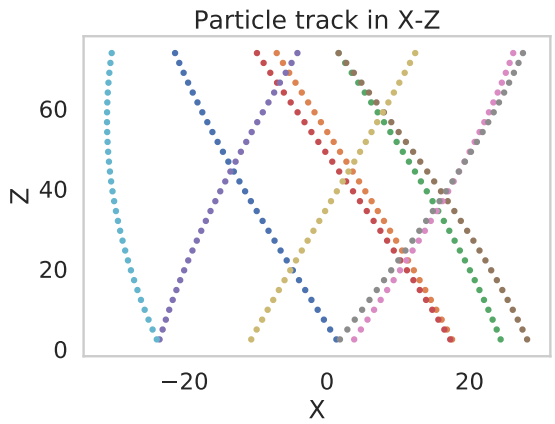
Synthetic data

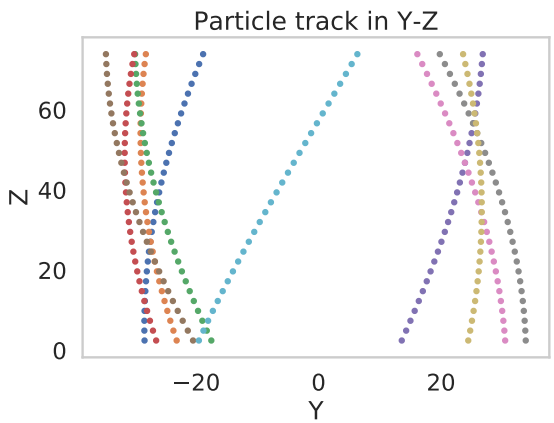
$$
\begin{cases}
t \sim \exp(\lambda) & & \\
x = x_0 + r \cdot \cos(a \cdot t + \phi_0) & + & \mathcal{N}(\mu_x, \sigma_x^2) \\
y = y_0 + r \cdot \sin(a \cdot t + \phi_0) & + & \mathcal{N}(\mu_y, \sigma_y^2) \\
z = z_0 + b \cdot t & + & \mathcal{N}(\mu_z, \sigma_z^2)
\end{cases}
$$

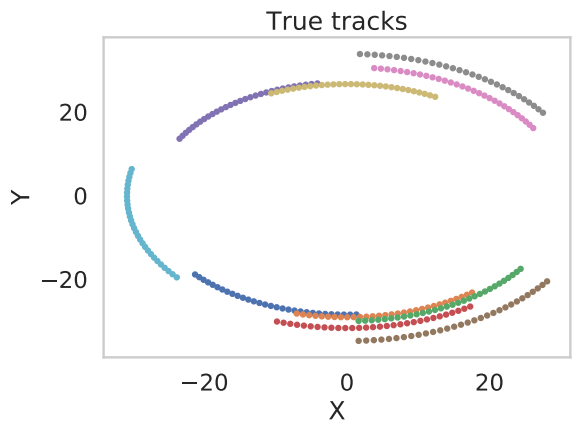
Synthetic data

$$
\begin{cases}
t \sim \exp(\lambda) & & \\
x = x_0 + r \cdot \cos(a \cdot t + \phi_0) & + & \mathcal{N}(\mu_x, \sigma_x^2) \\
y = y_0 + r \cdot \sin(a \cdot t + \phi_0) & + & \mathcal{N}(\mu_y, \sigma_y^2) \\
z = z_0 + b \cdot t & + & \mathcal{N}(\mu_z, \sigma_z^2)
\end{cases}
$$







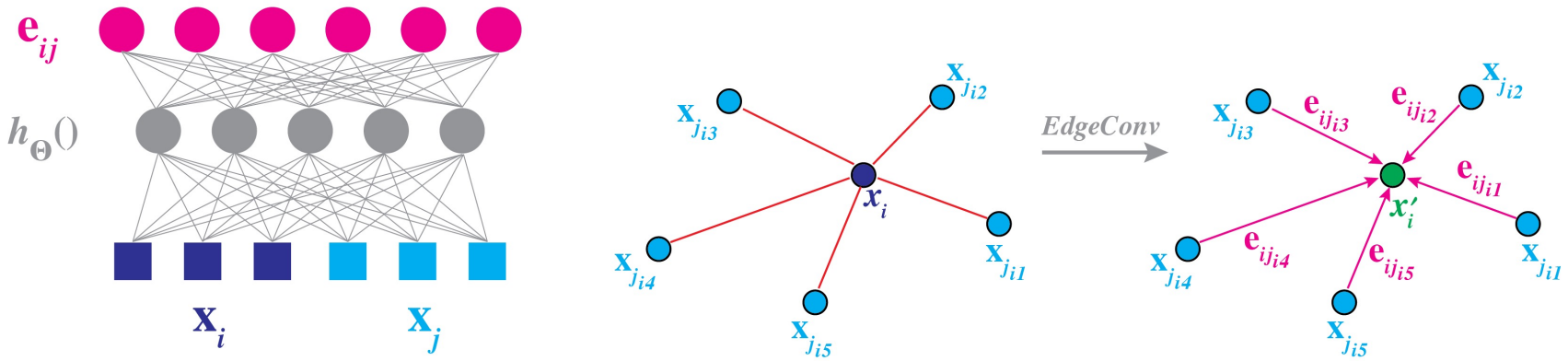


TrackML metric

Very short description:

It is the intersection between the reconstructed tracks and the ground truth particles, normalized to one for each event, and averaged on the events of the test set.

Edge Convolution



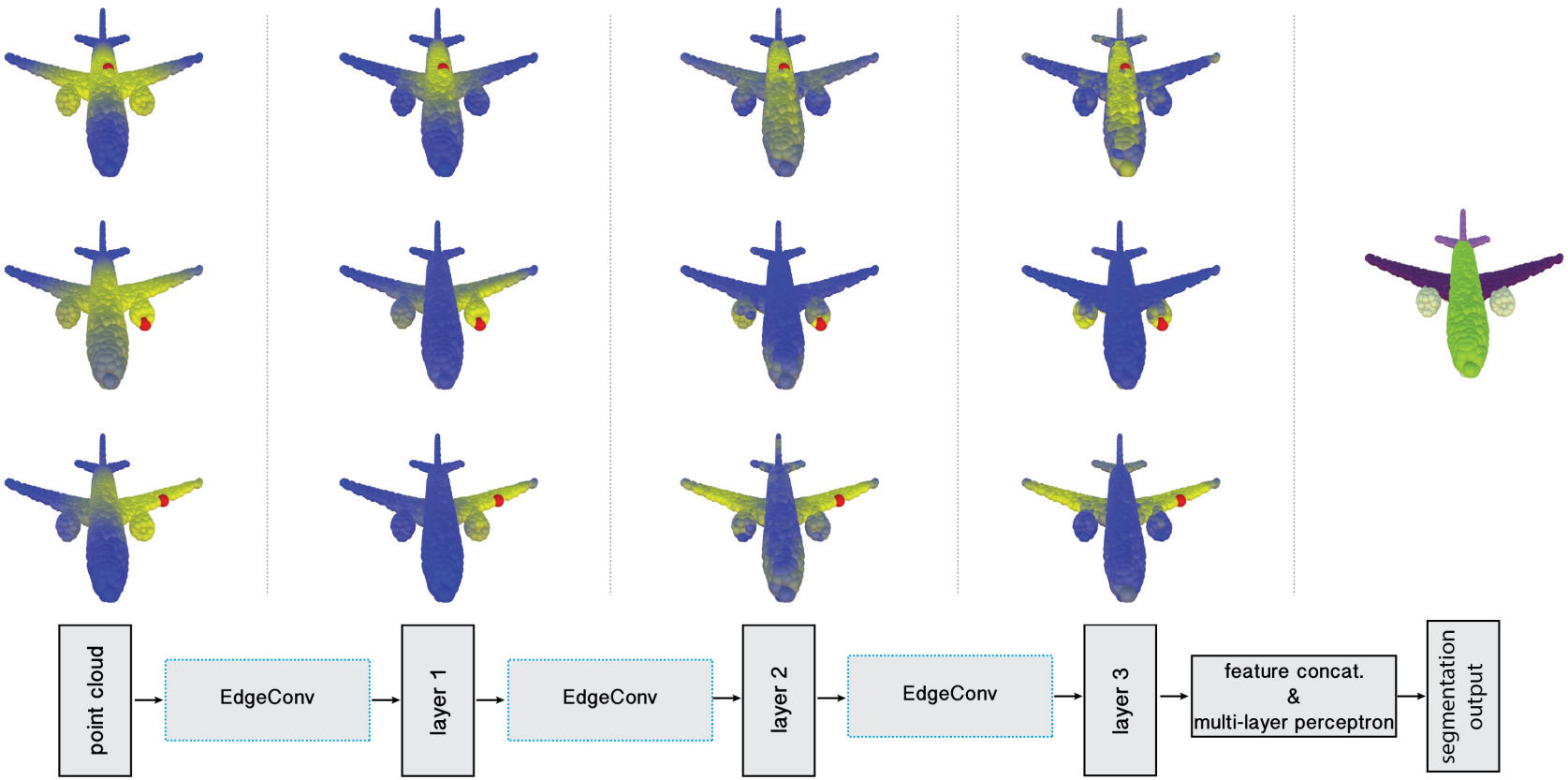
Point cloud: $X = \{x_1, \ldots, x_n\} \subset \mathbb{R}^F$
F represents the feature dimensionality of a given layer.

Edge features: $e_{ij} = h_\Theta(x_i, x_j)$, where
$h_\Theta : \mathbb{R}^F \times \mathbb{R}^F \to \mathbb{R}^{F'}$ – nonlinear function with a set of learnable parameters Θ. The output of **EdgeConv** at the i-th vertex:

$$x'_i = \square_{j:(i,j)\in\mathcal{E}} h_\Theta(x_i, x_j)$$

$\mathcal{E}$ – edges of the k-nearest neighbor (k-NN) graph of X in $\mathbb{R}^F$

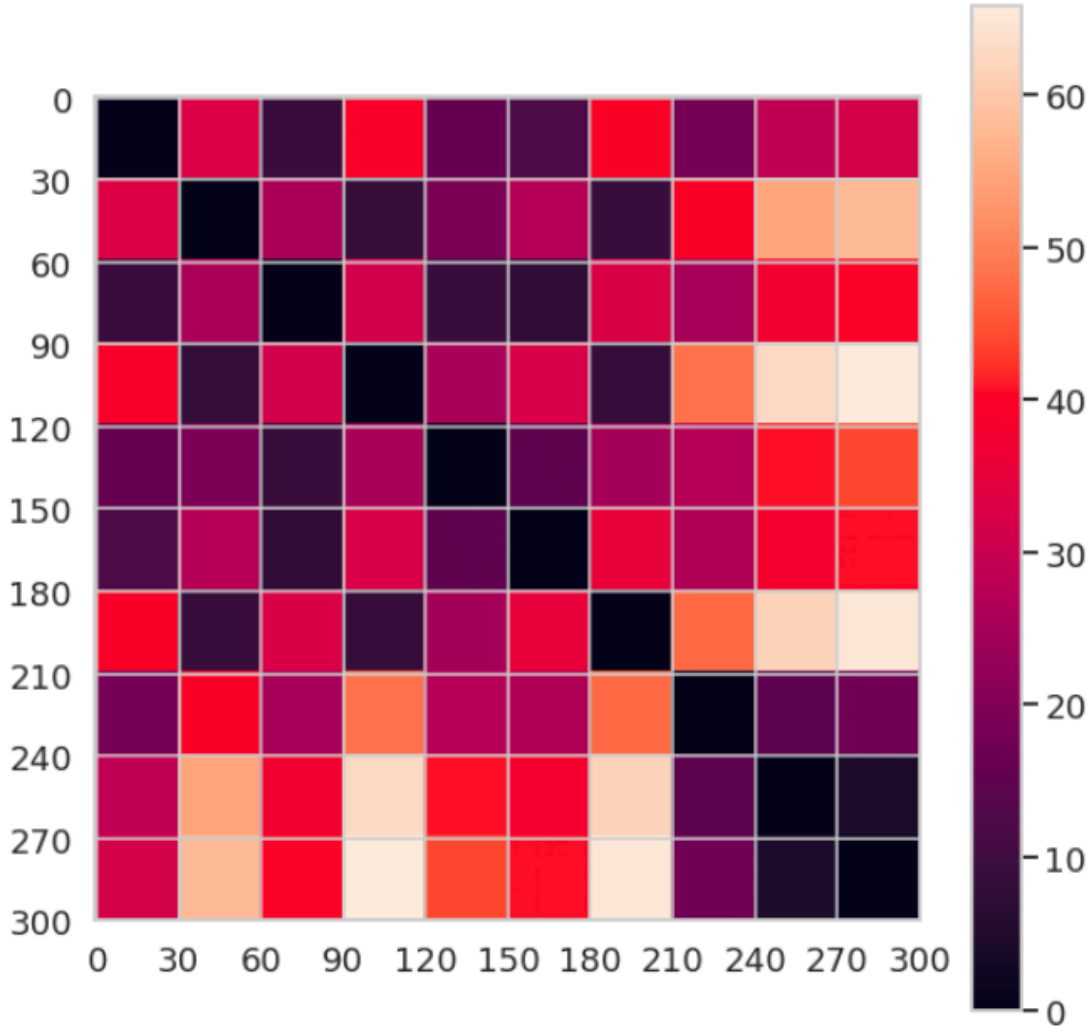
Dynamic Graph CNN[1]



[1]Wang et al. Dynamic Graph CNN for Learning on Point Clouds, 2019

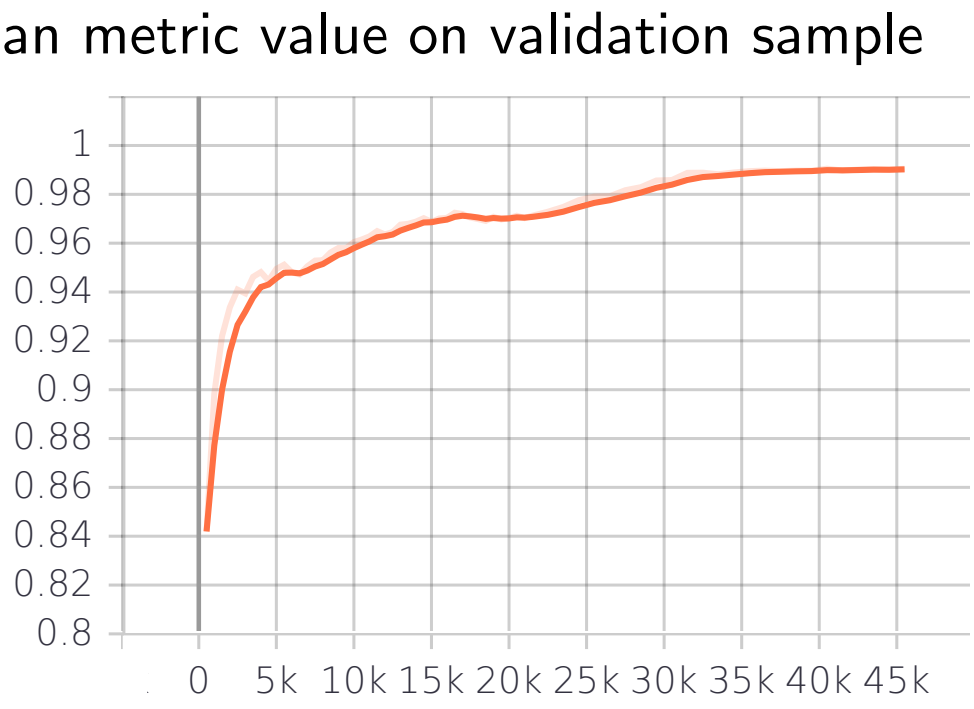
Proposed approach for tracking

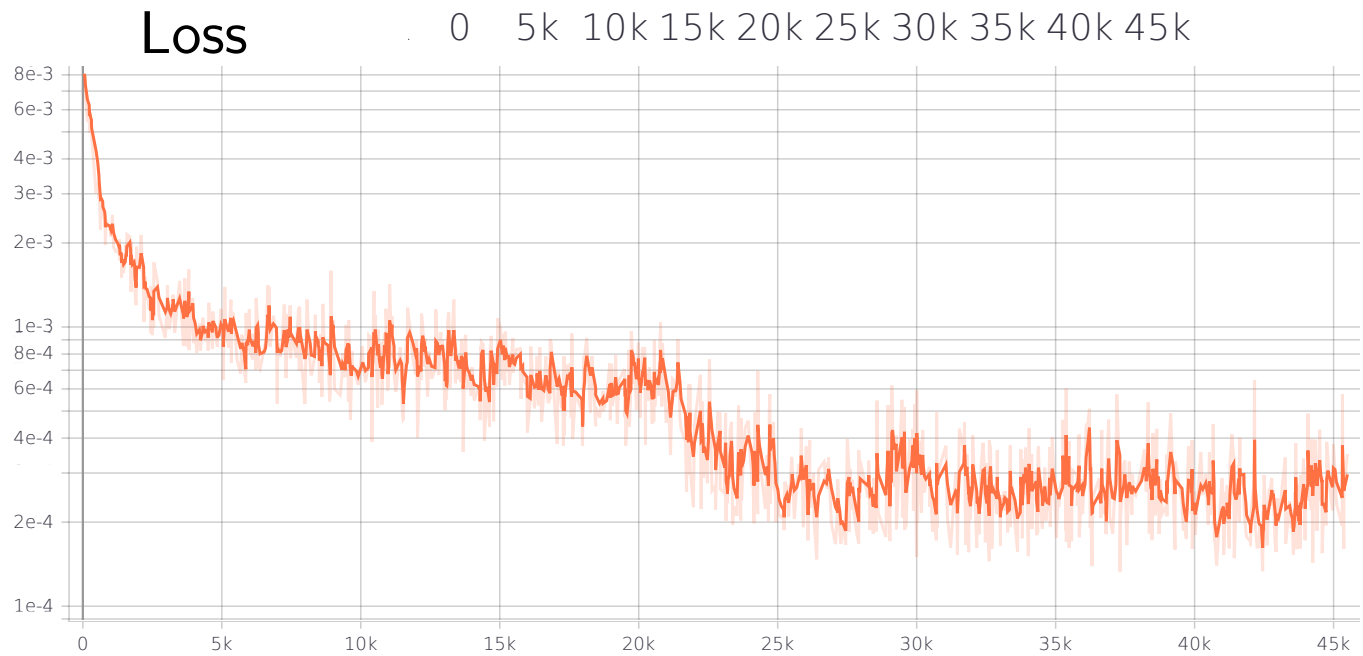
- DGCNN
- Calculation of the matrix of pairwise distances



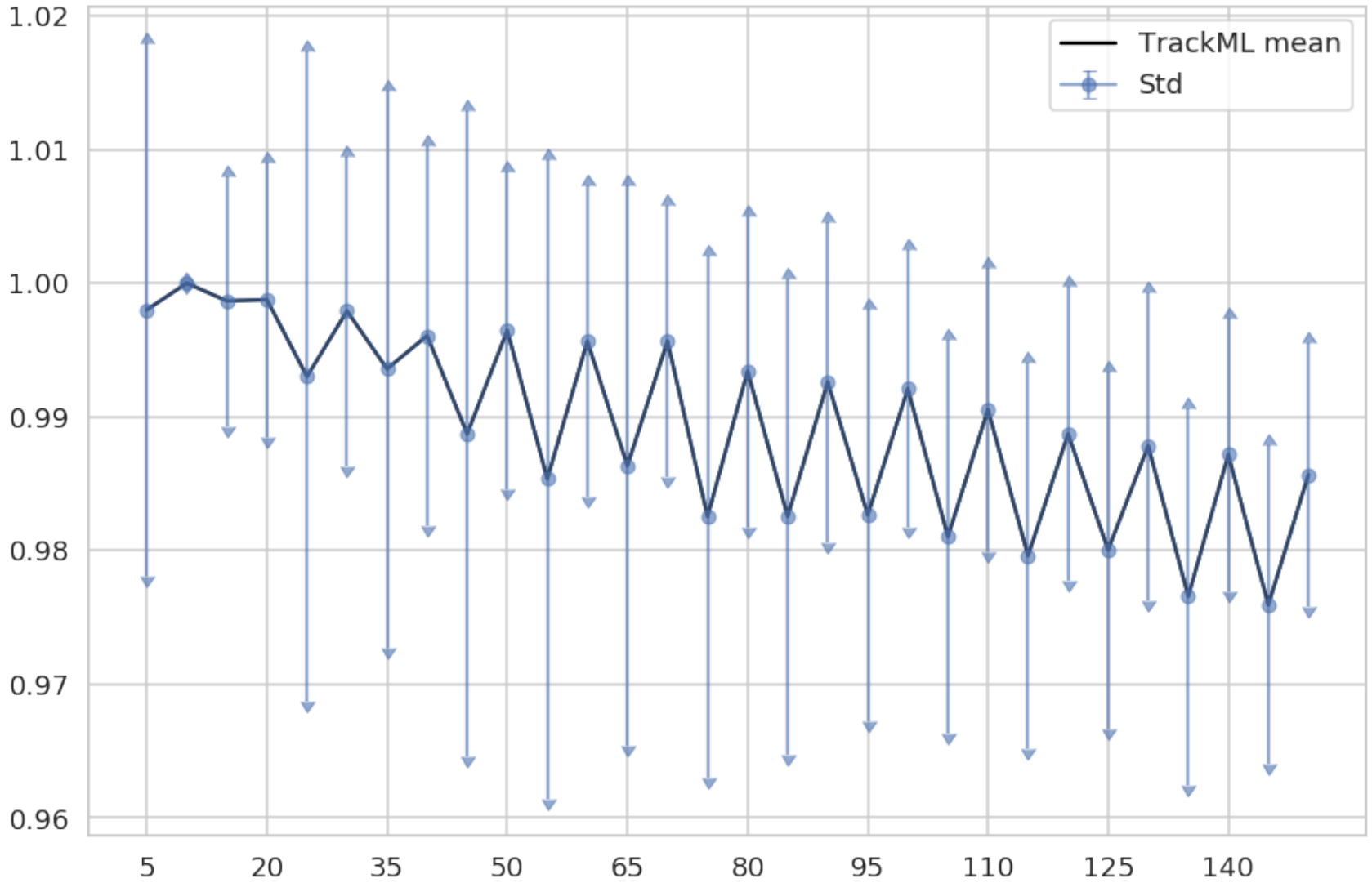
- Hierarchical agglomerative clustering

Model training





Metric value dependence on the number of tracks



Thank you for your attention.

Any questions?