

# Object Condensation Tracking



Kilian Lieret<sup>1,2</sup>  
@klieret



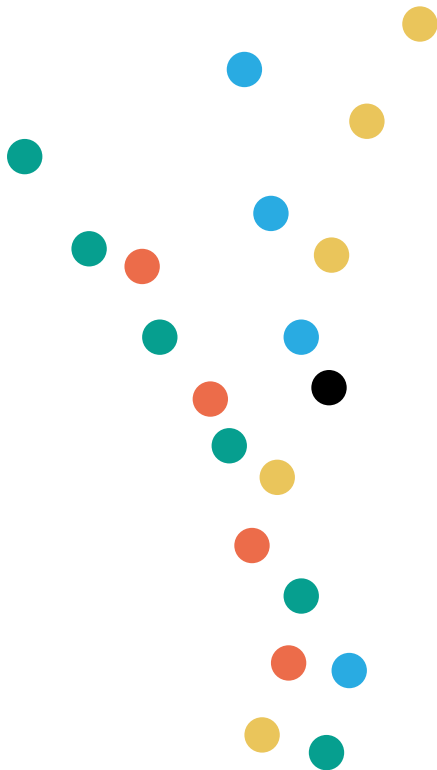
Gage DeZoot<sup>1</sup>  
@gageDeZoot

<sup>1</sup>Princeton University, <sup>2</sup>IRIS-HEP



# Vision: One-shot tracking

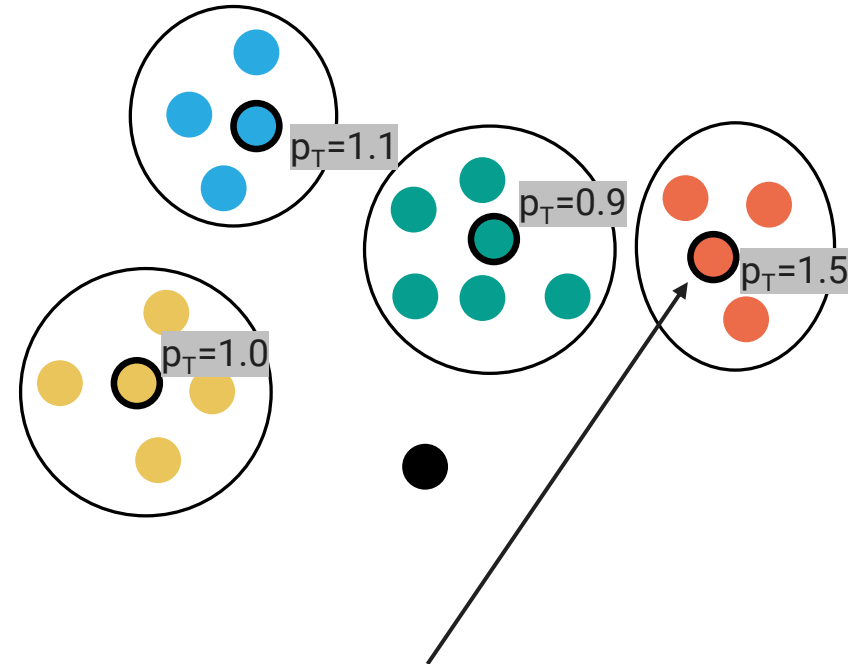
## Point cloud



**Repulsion & attraction**  
of points in latent space

## Learnt latent space

Hits already clustered by particle;  
Clusters can be collected trivially

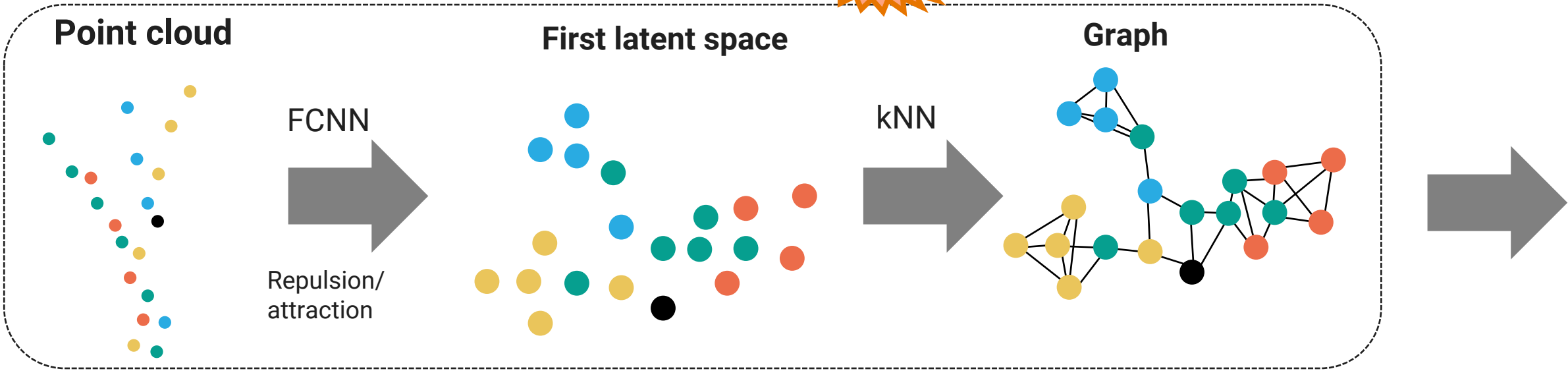


## Condensation point

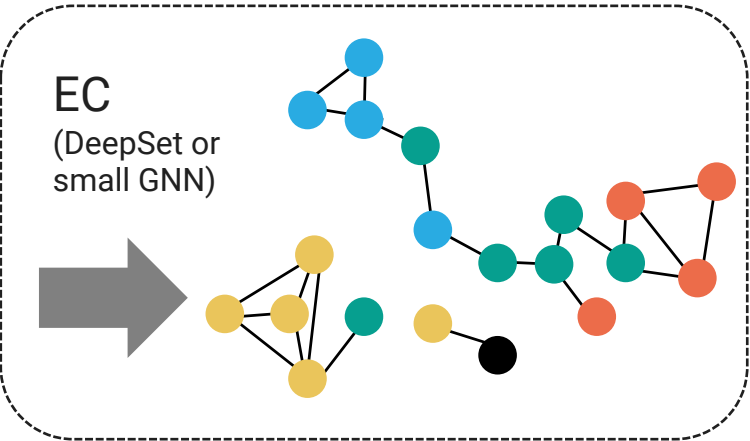
Represents the track, can  
learn track parameters like  $p_T$   
(WIP)

# Current pipeline

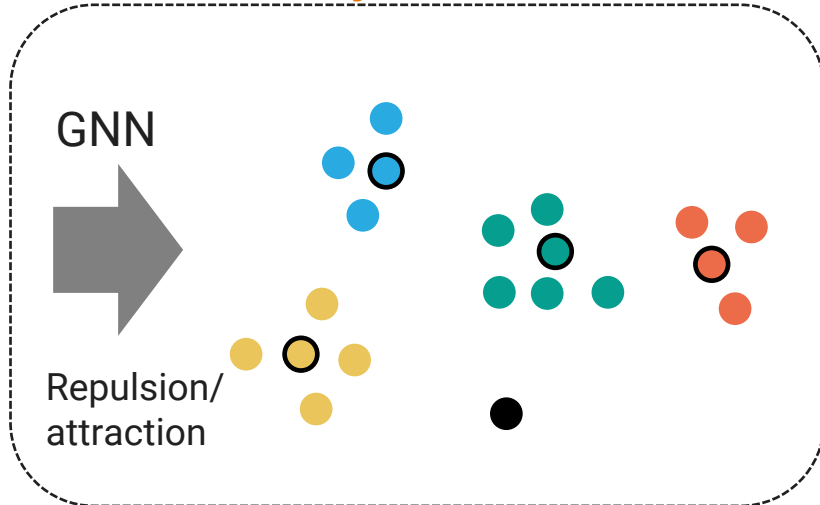
## STAGE 1: Graph building



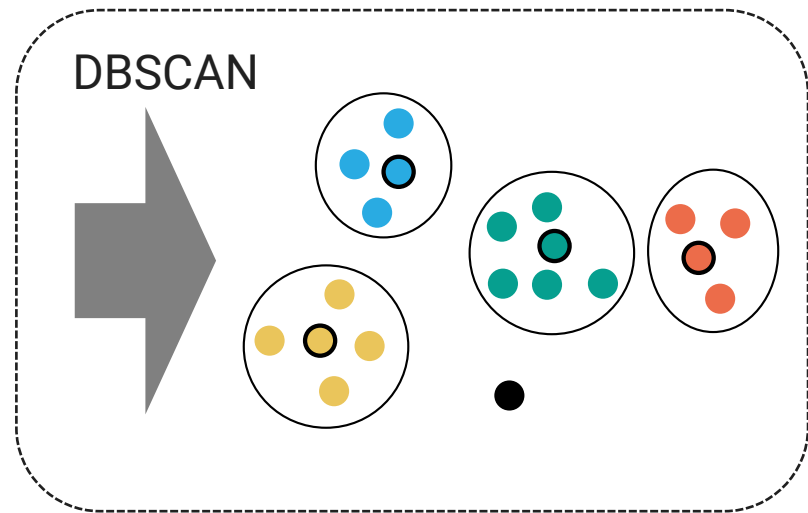
## (Optional STAGE 1a: Graph refinement)



## STAGE 2: Object condensation



## STAGE 3: Collect clusters



# People

## Core team



**Kilian Lieret**  
(Princeton)  
Current development  
& training



**Gage DeZoort**  
(Princeton)  
Ideas, GNN know-how,  
original codebase

## Transformer exploration

CS focused;  
Tracking as application/benchmarking



**Siqi Miao**  
(Georgia Tech)

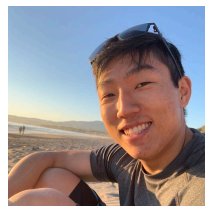


**Pan Li**  
(Georgia Tech)

Actively looking for more collaborators, especially additional core team members (e.g., Ph.D. student)

## CMS LST Liaison

Application to CMS data, possibly combined with early LST stages



**Jonathan Guiang**  
(UCSD)

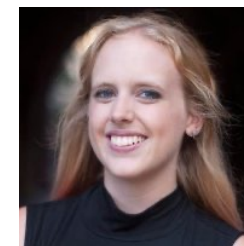


**Philip Chang**  
(Florida)

## Feedback & intellectual support



**Javier Duarte**  
(UCSD)



**Savannah Thais**  
(Columbia)

## Summer fellows



**Jian Park**  
(Chicago)



**Devdoot Chatterjee**  
(Delhi Tech U)



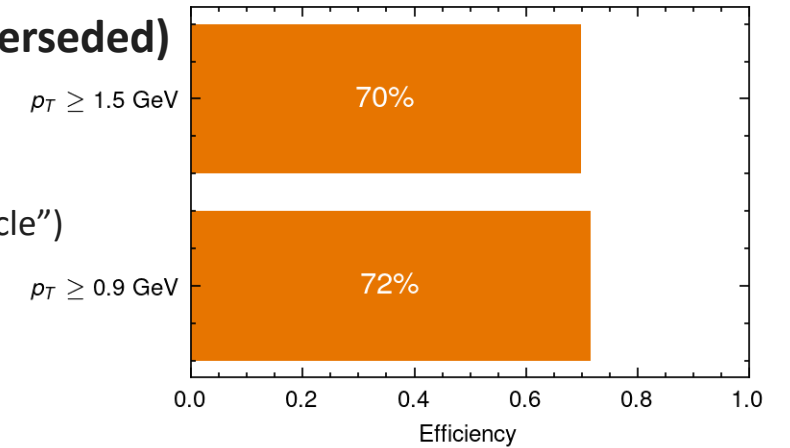
**Refilwe Bua**  
(Brown)

# Status

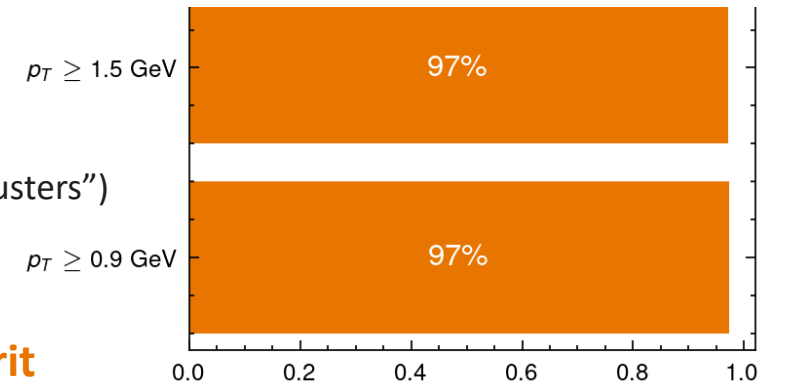
- So far running on **trackML dataset pixel layers**
- **Results presented at CHEP:**
  - First results **without truth cuts**
  - OC results seem to **match/outperform comparable EC GNN pipeline** trained on same data when using connected components for evaluation
- **Significantly improved pipeline since CHEP:**
  - Using embedding + kNN for graph construction
  - Performance improved while reducing memory consumption
- **In progress:**
  - CHEP proceedings
  - Paper with results of new pipeline

## CHEP results (superseded)

**Perfect**  
("cluster = particle")

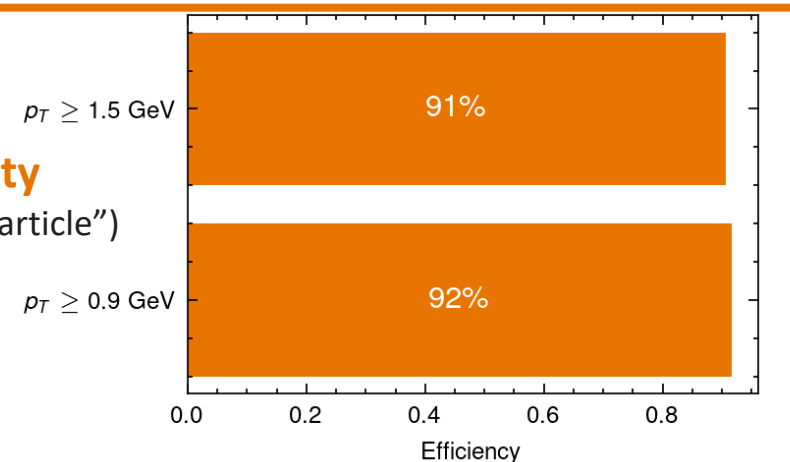


**LHC**  
("homogeneous clusters")



## Main Figure of Merit

**Double Majority**  
("1:1 match cluster <> particle")



# Plans

- **Core team is only ~1 FTE**



- enough to sustain **exploration and R&D**
- unlikely to bring this towards production fast (ExaTrkx has many times our resources and is only converging there)

- **2023 Milestones:**

- **Publishing** current research
- Running on **full-detector graphs** → This will result in more useful benchmarks & 🍏-🍎 **comparisons**

- **2023 Forks in the road**



- Does our approach **match/outperform ExaTrkx's** EC-based model on the full detector?
- Can we apply OC tracking to MDs from **LST project**? Do we see significantly better features in **CMS data**?
- Will **local transformers** be competitive in our approach?
- Does it make sense to merge frameworks with **ExaTrkx**?

- **2024 Milestones:**

- Adding **track parameter prediction**
- Training on **CMS data** and getting first benchmarks/results
- First **performance tests for speed**

- **2025 Milestones:**

- **Performance optimization** with specific accelerators
- **CMSSW** integration
- **Physics** studies with CMS data

- **End of project (3-5y):**

- OC Tracking is used/tested in **production**

# Thanks!

Find us on github!

<https://github.com/gnn-tracking>

**gnn\_tracking** Public

Edit Pins Unwatch 5 Fork 10 Starred 19

main 5 branches 6 tags

Charged particle tracking with graph neural networks

[gnn-tracking.rtf.io/](https://github.com/gnn-tracking)

tracking ai clustering ml hep hep-ex tracking-algorithm trackml gnn gnn-model

MIT license Cite this repository Activity 19 stars 5 watching 10 forks

**v23.09.0** Latest 2 weeks ago

**Contributors** 7

Python 99.0% Shell 1.0%

File	Description	Last Commit
<b>pre-commit-ci[bot]</b> [pre-commit.ci] pre-commit autoupdate (#424) ...		79e943c last week 1,153 commits
.github	Add ExpandWandbConfig callback	2 months ago
docs	Add links to source code in sphinx	2 months ago
environments	Use pyg implementation of edge_subgraph (#401)	last month
readme_assets	Add banner image	last year
src/gnn_tracking	[pre-commit.ci] pre-commit autoupdate (#424)	last week
tests	Fix test and test ECCut	last week
.all-contributorsrc	Add Jian as contributor	4 months ago
.flake8	Ignore studies submodule for flake8	last year
.gitignore	[Lightning WIP] Training loop working	3 months ago
.pre-commit-config.yaml	[pre-commit.ci] pre-commit autoupdate (#424)	last week
.readthedocs.yaml	Add RTD config file (#149)	last year
CHANGELOG.md	Release 23.09.0	2 weeks ago
CITATION.cff	Fix citation file format	last month
LICENSE.txt	Add packaging configuration	last year
README.md	Markdown link fix	last month

# Object condensation: Training losses

Latent space  
before training

GNN predicts **condensation likelihoods (CL)** for every hit.  
Hit with max CL for particle\* is **condensation point (CP)**

\*during inference: for cluster

**Attractive loss function**  
rewards hits close to their CP  
quadratic potential  
Attraction stronger if CP's CL is high

## Repulsive loss function

penalizes hits close to other CP  
hinge loss: no more repulsion after certain distance  
repulsion stronger for strong CP CLs

## Background loss function

noise hits should have low CL

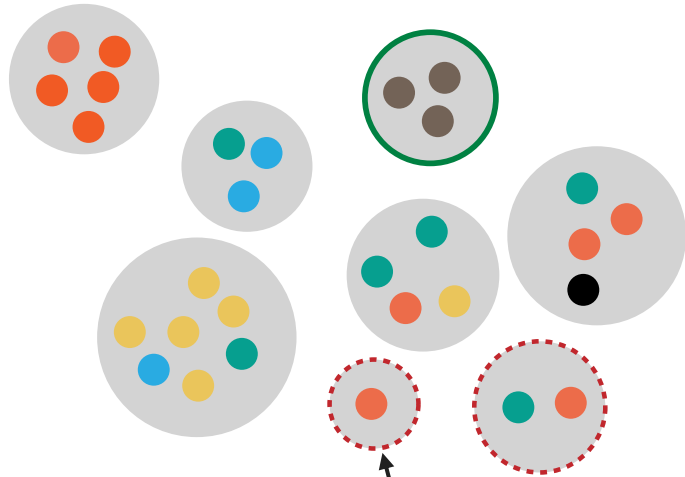
Loss functions implemented from  
Kieseler 2020 ([2002.03605](#))



# Metrics

## Perfect

Cluster contains only hits from one particle and no hits outside of cluster



Clusters with < 3 hits or non-reconstructable majority particle are discarded

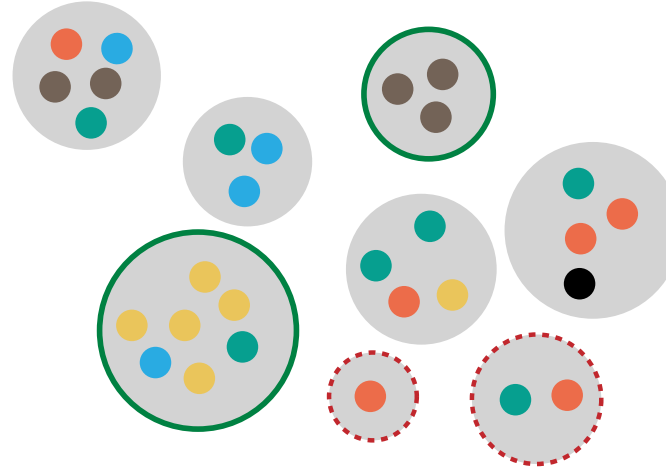
#reconstructable particles

Perfect efficiency =  $1/5$

Perfect fakes =  $5/5$

## LHC

Cluster contains  $\geq 75\%$  hits from one particle



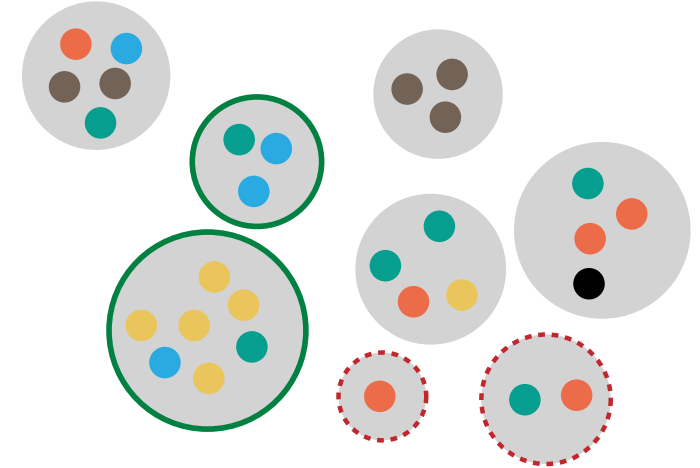
#clusters with  $\geq 3$  hits & majority particle reconstructable

LHC efficiency =  $2/5$

LHC fakes =  $4/6$

## Double Majority

Cluster contains  $\geq 50\%$  hits from one particle and This particle has < 50% of its hits outside



#reconstructable particles

DM efficiency =  $2/5$

DM fakes =  $4/5$

We also evaluate these **metrics at pT thresholds**: pT cut is applied to majority particle of cluster or particle (this is not a truth cut on the data, but simply a efficiency vs pT study)

Reconstructable:  $\geq 3$  hits