PROCEEDINGS OF SCIENCE

Efficient Transformers for Charged Particle Tracking at the HL-LHC

Zef Wolffs,[a,b,*] **Viacheslav Pshenov**[a,b] **and Sascha Caron**[a,c]

[a] *Nikhef, National Institute for Subatomic Physics,*
Science Park 105, 1098 XG Amsterdam, The Netherlands
[b] *Institute of Physics, University of Amsterdam,*
Science Park 904, 1098 XH Amsterdam, The Netherlands
[c] *Institute for Mathematics, Astrophysics and Particle Physics (IMAPP),*
Radboud University, Heyendaalseweg 135, 6525 AJ Nijmegen, The Netherlands

E-mail: f.author@inst.edu, s.author@univ.country

The High-Luminosity LHC (HL-LHC) will operate with unprecedented detector occupancies, raising a significant computational challenge in the tracking of charged particles. Transformer architectures may provide a solution in this challenge due to their ability to learn global relationships and their excellent parallel scaling on GPU hardware. However, the quadratic cost of the self-attention mechanism prevents straightforward application to events containing the hit multiplicities expected at the HL-LHC. This article presents an efficient transformer model for track pattern recognition that overcomes this limitation by introducing a geometry-driven sparse attention scheme based on FlexAttention. Locality is imposed by clustering hits in projected detector surfaces, reducing the number of computed attention interactions by more than two orders of magnitude. Evaluated on the pixel-detector portion of the TrackML dataset, the model reaches a double-majority efficiency of 90% for tracks with $p_{\mathrm{T}} > 0.9$ GeV, while achieving per-event inference latencies of under 100 ms on an NVIDIA H100 GPU. These results demonstrate that transformer architectures equipped with sparse locality-aware attention are a promising direction for fast, scalable charged particle tracking at the HL-LHC.

The European Physical Society Conference on High Energy Physics (EPS-HEP2025)
7–11 July 2025
Marseille, France

*Speaker

 ISSN 1824-8039 . Published by SISSA Medialab.

PoS(EPS-HEP2025)666

1. Introduction

Charged particle tracking plays a central role in event reconstruction at the LHC. In addition to providing the primary inputs for reconstructing charged particles themselves, it is used in precise vertex determination, jet reconstruction, and measurements of missing transverse momentum. At the HL-LHC, the average number of interactions per bunch crossing is expected to reach roughly 200, dramatically increasing the hit occupancy in the silicon detectors and pushing current reconstruction algorithms toward computational limits. Studies of the baseline ATLAS software stack indicate that, under simulated HL-LHC conditions, tracking accounts for a disproportionately large fraction of total CPU usage [1], making it a clear target for optimization.

Machine learning (ML) approaches have gained attention as potential replacements or augmentations to parts of the tracking pipeline. Graph Neural Networks (GNNs) have shown strong performance on HL-LHC-like datasets, but graph construction and message passing are computationally expensive [2, 3]. Transformers [4], by contrast, take advantage of highly optimized matrix multiplication on modern accelerators and have proven exceptionally scalable in other fields. Their key limitation is the quadratic cost of pairwise attention.

This work introduces a transformer-based tracking approach that preserves the global relational modeling capability of transformers while eliminating the quadratic bottleneck. By enforcing locality through hit clustering and employing FlexAttention [5] to mask attention interactions, the method performs end-to-end inference with dramatically reduced computational overhead. Combined with a contrastive learning scheme for track formation, the resulting model achieves competitive reconstruction efficiency with state of the art inference latency.

2. TrackML dataset

The TrackML dataset [6] has become a widely accepted benchmark in the literature for evaluating machine learning-based track reconstruction algorithms for the HL-LHC. It provides simulated pp collisions at $\sqrt{s} = 14$ TeV with ~ 200 pile-up interactions, modeled via Pythia 8. The detector comprises multiple concentric cylindrical and disk-shaped sensor layers embedded in a magnetic field. Each collision event produces a set of three-dimensional hit coordinates from particles traversing these sensor layers. In addition to hit-level data, the dataset includes truth-level information for each simulated particle.

On average, an event contains $O(11{,}000)$ particles and $O(20{,}000)$ noise hits. Following common practice in ML-based tracking studies, the method proposed here is trained and evaluated solely on hits from the innermost (pixel) part of the detector, reducing the complexity while retaining the relevant physics challenge.

3. Transformer architecture with locality-aware sparse attention

The transformer model used in this work is designed to capture global relationships between detector hits while remaining computationally tractable at HL-LHC occupancies. To achieve this, we begin with an encoder-only transformer architecture and introduce a series of modifications that exploit the geometric structure of particle trajectories. These additions constrain the attention

mechanism to local, physically motivated neighborhoods and replace discrete track classification with a contrastive embedding approach. The main components of the architecture are described in the sections below.

3.1 The baseline: encoder-only transformer

The starting point for this work is the encoder-only transformer introduced in Ref. [7]. In its original form, the model—referred to as the "encoder-classifier" (EncCla)—treats track finding as a multi-class classification problem. Each hit is assigned to one of 5400 discretized track "roads", defined by quantile-binned combinations of parameters such as the particle charge, momentum, and angular direction. The input hit coordinates (x, y, z) are first embedded into a latent vector space and then processed by a stack of transformer encoder layers, each consisting of multi-head self-attention followed by a feed-forward network. A final softmax layer outputs a probability distribution over all track classes for every hit.

Although conceptually simple and effective on reduced datasets, this baseline architecture suffers from a fundamental scalability limitation. The self-attention mechanism computes pairwise interactions between all n hits in an event, producing $n \times n$ attention matrices. For HL-LHC–like events with $n \sim 10^5$ pixel hits, a single attention matrix contains 10^{10} elements. At `float32` precision, storing this matrix requires roughly 40 GB of memory per head, far exceeding the capacity of modern GPUs and making full attention computationally infeasible.

This limitation motivates the key innovation of this work: introducing physics-motivated locality constraints that drastically reduce the number of attention interactions without degrading tracking performance.

3.2 Detector-driven locality via hit projection and clustering

The locality constraint is motivated by the observation that transformer attention weights in the baseline model concentrate almost exclusively on hits lying within narrow neighborhoods in (η, ϕ), consistent with the geometric structure of real particle trajectories. To exploit this, we define locality explicitly by projecting hits onto simplified detector surfaces that approximate the pixel geometry:

- a cylindrical surface at fixed radius for the barrel region,
- two planar surfaces at fixed z for the forward endcaps.

Hits are mapped to these surfaces along straight lines from the detector origin and then clustered using DBSCAN [8] in the projected (η, ϕ) coordinates. The clustering groups together hits that lie close in direction and are therefore likely to originate from the same physical trajectory, while separating geometrically distant hits. Importantly, more than 98% of particles with $p_T > 0.9$ GeV and with at least three pixel hits remain fully contained within a single cluster, ensuring that the locality constraint preserves most reconstructible tracks.

3.3 Sparse attention using FlexAttention

Once locality clusters are defined, the self-attention mechanism is restricted to operate only within them. This is implemented using FlexAttention, a recent attention primitive that supports

fine-grained, dynamically specified masking of query–key pairs. A binary mask M encodes which hit pairs belong to the same cluster, and the attention becomes

$$\text{Attention}(Q, K, V) = \text{softmax}\left(M \odot \frac{QK^\top}{\sqrt{d_k}}\right) V.$$

Masked entries are completely excluded from both computation and memory, allowing the model to focus exclusively on physically relevant local interactions. For TrackML pixel events, this reduces the number of computed attention elements by more than 99.9%, corresponding to a reduction in effective attention complexity by over two orders of magnitude. Crucially, this sparsification preserves the central strength of transformers—the ability to learn relational patterns among hits—while aligning computation with the known geometric structure of charged particle trajectories.

3.4 Contrastive embedding learning for track formation

The original encoder-classifier model assigned each hit to one of thousands of discretized track "roads." While effective for low-occupancy events, this approach becomes limiting at HL-LHC densities, where road bins must become increasingly narrow to avoid merging unrelated tracks, making the classification task brittle and difficult to learn.

To overcome this, we replace discrete road classification with a contrastive embedding objective [9]. The transformer outputs a continuous embedding vector for each hit, and training encourages hits from the same particle to lie close together in this space while pushing apart hits from different particles in the same locality cluster. Pairwise similarity is computed using the cosine similarity, $\text{sim}(i, p) = \frac{q_i \cdot q_p}{\|q_i\| \, \|q_p\|}$, where q_i and q_p are the embedding vectors of hits i and p. The result is scaled by a temperature parameter T and passed through a softmax over all hits in the locality cluster $N(i)$:

$$\Pr(p|i) = \frac{\exp(\text{sim}(i, p)/T)}{\sum_{u \in N(i)} \exp(\text{sim}(i, u)/T)}.$$

The contrastive loss then maximizes the likelihood of selecting hits that belong to the same particle,

$$\mathcal{L} = -\frac{1}{|\mathcal{J}|} \sum_{i \in \mathcal{J}} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \Pr(p|i),$$

where $\mathcal{J}$ is the set of hits from particles with $p_\text{T} > 0.9$ GeV and $P(i)$ is the set of hits sharing the same particle as hit i. Minimizing this loss encourages coherent "track manifolds" in embedding space that reflect true track membership.

At inference time, track candidates are formed by grouping, for each hit, its nearest neighbors in the learned embedding space. The number of neighbors is tuned to reflect the number of expected pixel layers, yielding track segments directly from the learned geometry of the embedding.

4. Results

The performance of the proposed transformer-based tracking approach is evaluated along two complementary dimensions: reconstruction accuracy and computational efficiency. The following subsections summarize the key results.

4.1 Tracking accuracy

Tracking performance is assessed using the TrackML double-majority (DM) efficiency. A predicted track must contain at least half of the true hits of a particle, and at least half of its own hits must belong to that particle [6]. Only particles with $p_T > 0.9$ GeV are included.

The proposed method achieves:

DM efficiency = 90%

across the full pixel detector. Efficiency remains above 90% for most p_T ranges, with a mild decrease only for very low momentum tracks. The result is comparable to leading GNN-based approaches applied to similar TrackML subsets, and moderately below a recent MaskFormer-based transformer pipeline [10], which reaches $\sim$ 94% using a two-stage architecture.

4.2 Inference latency

Latency tests were conducted on an NVIDIA H100 GPU. The full pipeline includes CPU-based DBSCAN clustering, transformer encoder inference, and embedding-based track formation. Results are summarized in Table 1.

Table 1: Average per-event inference time on an NVIDIA H100 GPU. Multi-core CPU values assume parallelized DBSCAN on 36 cores.

Pipeline component	**Time / event (ms)**
Hit clustering & mask building (multi-core CPU)	6
Transformer encoder forward pass (GPU)	20
Track candidate formation (GPU)	47
Total latency	$\sim$ 73 **ms**

The total per-event latency of $\sim$ 70 ms makes transformer-based tracking attractive for online applications. It is moreover faster than published ML-based tracking results: GNN4ITk reports $\sim$ 500 ms for full ITk events [3], and the recent MaskFormer-based method achieves $\sim$ 100 ms on TrackML events [10].

5. Conclusion

This work presents a computationally efficient transformer architecture for charged particle tracking that leverages detector geometry to impose sparse attention via FlexAttention. Combined with a contrastive learning objective, this enables high-quality track pattern recognition at HL-LHC occupancies with efficient compute.

The model achieves a double-majority efficiency of 90% on the TrackML pixel dataset while achieving under 100 ms inference times per event on an H100 GPU. These results demonstrate that locality-aware transformers are a highly promising candidate for future low-latency tracking pipelines, including real-time applications in trigger systems at the LHC.

Future developments include training on larger datasets, incorporating full detector geometries and realistic non-idealities, and exploring deeper or wider transformer configurations to exploit the scaling behavior characteristic of these models.

References

[1] ATLAS Collaboration, *Fast Track Reconstruction for HL-LHC*, ATL-PHYS-PUB-2019-041 (2019), https://cds.cern.ch/record/2693670.

[2] J. Shlomi, P. Battaglia and J.-R. Vlimant, *Graph neural networks in particle physics*, Mach. Learn. Sci. Technol. **2** (2021) 021001, doi:10.1088/2632-2153/abbf9a, arXiv:2007.13681 [hep-ex].

[3] S. Caillou, P. Calafiura, X. Ju, D. Murnane, T. Pham, C. Rougier, J. Stark and A. Vallier, *Physics performance of the ATLAS GNN4ITk track reconstruction chain*, EPJ Web Conf. **295** (2024) 03030, doi:10.1051/epjconf/202429503030.

[4] A. Vaswani et al., *Attention Is All You Need*, NeurIPS 2017.

[5] J. Dong, B. Feng, D. Guessous, Y. Liang and H. He, *FlexAttention: A programming model for generating optimized attention kernels*, arXiv:2412.05496 [cs.LG] (2024), doi:10.48550/arxiv.2412.05496.

[6] M. Kiehn *et al.*, *The TrackML High-Energy Physics Tracking Challenge on Kaggle*, EPJ Web Conf. **214** (2019) 06037, doi:10.1051/epjconf/201921406037.

[7] S. Caron, N. Dobreva, A. Ferrer Sánchez, J. D. Martín-Guerrero, U. Odyurt, R. Ruiz de Austri Bazan, Z. Wolffs and Y. Zhao, *Trackformers: in search of transformer-based particle tracking for the high-luminosity LHC era*, Eur. Phys. J. C **85** (2025) 460, doi:10.1140/epjc/s10052-025-14156-3, arXiv:2407.07179 [hep-ex].

[8] J. Sander, M. Ester, H.-P. Kriegel and X. Xu, *Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications*, Data Min. Knowl. Disc. **2** (1998) 169–194, doi:10.1023/A:1009745219419.

[9] T. Chen, S. Kornblith, M. Norouzi and G. Hinton, *A simple framework for contrastive learning of visual representations*, in *Proceedings of the 37th International Conference on Machine Learning (ICML 2020)*, PMLR **119** (2020) 1597–1607, arXiv:2002.05709 [cs.LG].

[10] S. van Stroud, P. Duckett, M. Hart, N. Pond, S. Rettie, G. Facini and T. Scanlon, *Transformers for charged particle track reconstruction in high energy physics*, arXiv:2411.07149 [hep-ex] (2024), doi:10.48550/arxiv.2411.07149.