Attention-Bayesian Hybrid Approach to Modular Multiple Particle Tracking

Anonymous authors
Paper under double-blind review

Abstract

Tracking multiple particles in dense scenes remains challenging due to a combinatorial explosion of trajectory hypotheses, which scales super-exponentially with the number of frames. The transformer architecture has shown a significant improvement in robustness against this high combinatorial load. However, its performance still falls short of the conventional Bayesian filtering approaches in locally sparse scenarios presenting a reduced set of trajectory hypothesis. This suggests that while transformers excel at narrowing down possible associations, they are not able to reach the optimality of the Bayesian approach in locally sparse scenario. Hence, we introduce a hybrid tracking framework that combines the ability of self-attention to learn the underlying representation of particle behavior with the reliability and interpretability of Bayesian filtering. We perform trajectory-to-detection association by solving a label prediction problem, using a transformer encoder to infer soft associations between detections across frames. This prunes the hypothesis set, enabling efficient multiple-particle tracking in Bayesian filtering framework. Our approach demonstrates improved tracking accuracy and robustness against spurious detections. These results open the way to a solution for high-clutter multiple-particle tracking scenarios that takes advantage of the large context accessible to transformers, together with the interpretability and theoretical guarantees of Bayesian filtering techniques.

1 Introduction

Fluorescence imaging in live cells has uncovered environments in which molecules move through crowded and dynamic intracellular spaces (Stephens & Allan (2003); Chen et al. (2014a); Dean et al. (2016)). These images enable the study of how cellular behaviors emerge from dense populations of interacting molecules Chen et al. (2014b); David et al. (2019). However, tracking particles, even in limited sub-regions (Fig. 1), has remained a fundamental challenge due to the density of moving particles, the stochasticity of their motions, and the noisy nature of the photon-limited fluorescence signal. These challenges have hindered the understanding of the physiological and pathological functions of cells.

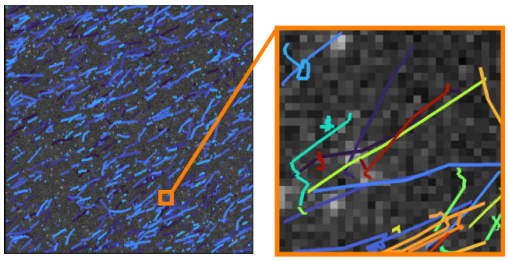

Figure 1: Ground-truth trajectories illustrating the complex motion patterns in viral dynamics simulations from the ISBI Particle Tracking Challenge.

Concretely, multiple-particle tracking consists in reconstructing trajectories from noisy, unlabeled detections observed in each frame. The difficulty arises from false positive and false negative detections in photon-limited imaging (Basset et al., 2015; Smal et al., 2009; Dominguez Mantes et al., 2025; Chenouard et al., 2014), limited temporal sampling, high particle density, and unpredictable transitions between motion regimes such as diffusion, directed transport, and confinement. Solving this problem requires a trajectory model that can evaluate candidate sequences of detections. In the multi-particle setting, however, such evaluation depends on resolving the association between detections and underlying trajectories. Since the number of possible trajectory-to-detection associations grows super-exponentially with the numbers of particles and frames, classical multiple-hypothesis tracking (MHT) methods (Fortmann et al., 1983; Reid, 1979) infer

approximate solutions that rely on various tree-pruning heuristics, later adapted to bioimaging (Jaqaman et al., 2008; Chenouard et al., 2013; 2014).

In practice, conventional reconstruction algorithms are temporally greedy. They alternate between evaluating the likelihood of all possible trajectory-to-measurement associations over the next few frames and a pruning step that uses linear programming to select a locally optimal, mutually compatible set of associations within a fixed temporal window (Chenouard et al., 2013; Liang et al., 2014; Smal & Meijering, 2015). In bioimaging, trajectory likelihood evaluation has largely relied on conventional Markov modeling, with most approaches based on Kalman filtering equipped with multiple dynamic models (Jaqaman et al., 2008; Roudot et al., 2017; Chenouard et al., 2013). Later, recurrent neural networks (RNNs) (Spilger et al., 2020; 2021) have been proposed to enable nonlinear modeling with efficient inference. However, while they have shown promising performance, especially at very low SNR and under extreme false negative rates, carefully engineered multiple-motion Markov models still perform best under typical experimental conditions (Roudot et al., 2023; Ritter et al., 2024).

Transformers (Vaswani et al., 2017) marked a major advance in word sequence modeling by exploiting broader context and affinities between input elements. In multiple particle tracking, this makes them attractive not only for evaluating sequences of detections (Zhang & Yang (2023)), much like Markov models and RNNs, but also for guiding the selection of mutually compatible trajectory-to-detection assignments. In a study simulating submarine tracking scenarios, Pinto et al. (2023) replaced the discrete optimization step with a transformer trained to infer trajectory-to-measurement associations directly, with gains becoming more apparent as the association problem grew more ambiguous. In other recent work on cell trackingGallusser & Weigert (2024), authors used a transformer to associate cell coordinates into a consistent lineage tree, achieving strong performance on cell tracking benchmarks Maka et al. (2023), especially when trained on large and diverse datasets. These studies suggest that transformers are useful not only for modeling trajectory structure, but also for pruning the space of plausible associations. This idea was examined more explicitly by Mishra & Roudot (2024), who showed that transformers are more robust to increasing combinatorial complexity than conventional multiple-hypothesis tracking, while scaling more favorably with the size of the temporal context. At the same time, they found that classical MHT is more robust when the hypothesis space is tractable. Taken together, these studies suggest that the strengths of transformers and Markov models are complementary: transformers are well suited to reducing ambiguous association hypotheses over longer contexts, whereas Markov models remain attractive for estimating the parameters and likelihood of individual trajectories. This is particularly true in our setting, where particle dynamics are well described by diffusion and motor-driven transport amenable to Markov modeling (Welte, 2004; Brangwynne et al., 2009). Additionally, physical interpretability and goodness-of-fit assessment are important for identifying tracking errors Mettlen & Danuser (2014); Kuhn et al. (2021). Despite these complementary strengths, existing approaches typically emphasize either global association modeling with transformers or local dynamical estimation with Markov models, rather than combining both within a single tracking framework.

Based on this insight, this paper makes three main contributions. First, we introduce an Attention-Bayesian Hybrid Approach for modular multiple-particle tracking (ABHA), in which a novel attention-based architecture infers an association matrix to prune the space of trajectory-to-detection hypotheses before Kalman filtering estimates the dynamical parameters of each trajectory from noisy measurements. Each component of our neural network design is supported by ablation experiments that demonstrate the importance of feed-forward depth, nonlinearity, and the attention layer. Second, through a controlled comparison with classical multiple-hypothesis tracking using the same Bayesian state-estimation backbone, we find strong performance as soon as background clutter is introduced, which is unavoidable in bioimaging. Third, we demonstrate the applicability of this hybrid strategy to random illumination microscopy, an imaging modality that is especially prone to clutter.

The manuscript is organized as follows. Section 2 presents the Bayesian framework, the transformer architecture, the MHT implementation, the performance metrics, and the experimental setup used for comparison. Section 3 reports the tracking results across different regimes, including the ablation studies and the application to random illumination microscopy. Finally, Section 4 summarizes the main findings, discusses limitations, and outlines directions for future work.

2 Methods

In this section, we first describe the Bayesian tracking framework along with the attention-based association module and the filtering module. We then present the baseline methods, evaluation metrics, and experimental setup used throughout the paper.

2.1 Proposed Methodology for ABHA

We formulate multiple-particle tracking as a posterior estimation problem over particle states, obtained by marginalizing over trajectory-to-detection associations. Let $\mathbb{T}$ denote the total number of particle trajectories present over the full sequence. For each time step t, let $\Omega_t \subseteq \{1, \ldots, \mathbb{T}\}$ denote the set of trajectory indices active at time t, and define the corresponding set of particle positions by $\mathcal{X}_t = \{\chi_t^i\}_{i \in \Omega_t}$, with $\chi_t^i \in \mathbb{R}^2$. Let $\mathbf{Z}_t = \{z_t^1, \ldots, z_t^{M_t}\}$ denote the set of detections at time t, where M_t is the number of detections. Because detections are unlabeled, tracking also requires inferring an association matrix A that links detections in $\mathbf{Z}_{1:T}$ to trajectory indices across the sequence. For Bayesian filtering, each active particle position χ_t^i is represented by an extended latent state $\mathbf{x}_t^i$, which augments position with the dynamical variables needed to preserve the Markov property.

From a Bayesian perspective, the posterior distribution over latent states is obtained by marginalizing over all admissible association hypotheses:

$$p(\mathbf{X}_{1:T} \mid \mathbf{Z}_{1:T}) = \sum_{\mathbf{A}} p(\mathbf{X}_{1:T} \mid \mathbf{A}, \mathbf{Z}_{1:T})\, p(\mathbf{A} \mid \mathbf{Z}_{1:T}). \tag{1}$$

This decomposition thus reflects the separation of the the tracking problem into a discrete association term, which quantifies the plausibility of competing trajectory-to-detection assignments, and a conditional state-estimation term, which evaluates the latent trajectories under a given association hypothesis.
In ABHA, the association term $p(\mathbf{A} \mid \mathbf{Z}_{1:T})$ is approximated by an attention-based module that uses the full detection sequence to predict a soft association structure. These scores are then used to prune the combinatorial space of admissible assignments. Conditional on the retained association hypotheses, the state term $p(\mathbf{X}_{1:T} \mid \mathbf{A}, \mathbf{Z}_{1:T})$ is approximated by Bayesian state estimation. Although Eq. 1 is written in smoothing form, this is implemented in practice through a filtering recursion.

2.1.1 A point-cloud association technique using self-attention as a way to prune trajectory hypotheses

The input to the transformer is built directly from the detection sequence $\mathbf{Z}_{1:T}$. We flatten this sequence into a single list of $n = \sum_{t=1}^{T} M_t$ detections while preserving the frame index of each detection. This yields two aligned arrays: a measurement matrix $\mathbf{z} \in \mathbb{R}^{n \times 2}$, whose rows are the coordinates of all detections in $Z_{1:T}$, and a time-index vector $\mathbf{t} \in \{1, \ldots, T\}^n$, whose p-th entry gives the frame from which the p-th detection was drawn. The association module therefore processes detections jointly across all frames in $\mathbf{Z}_{1:T}$, while retaining temporal information through $\mathbf{t}$. The output of the transformer is the association matrix $\mathbf{A} \in \mathbb{R}^{n \times B}$, with B chosen as an upper bound on the number of trajectories.

Each detection coordinate is first projected into an h-dimensional embedding space using a learned linear map $\mathbf{W} \in \mathbb{R}^{2 \times h}$:

$$\begin{aligned} \mathscr{P} &: \mathbb{R}^{n \times 2} \to \mathbb{R}^{n \times h} \\ \mathbf{z}' &= \mathscr{P}(\mathbf{z}) = \mathbf{z}\mathbf{W} \end{aligned} \tag{2}$$

The frame indices are encoded through a learned temporal embedding. Let $\mathscr{T}$ map $\mathbf{t}$ to the set of unique frame indices,

$$\mathscr{T} : \mathbb{R}^n \to \mathbb{S} \text{ where } \mathbb{S} = \{s | s \in \mathbf{t}\} \tag{3}$$

which is then projected into the same h-dimensional space,

$$\mathscr{L} : \mathbb{S} \to \mathbb{R}^{|\mathbb{S}| \times h} \tag{4}$$

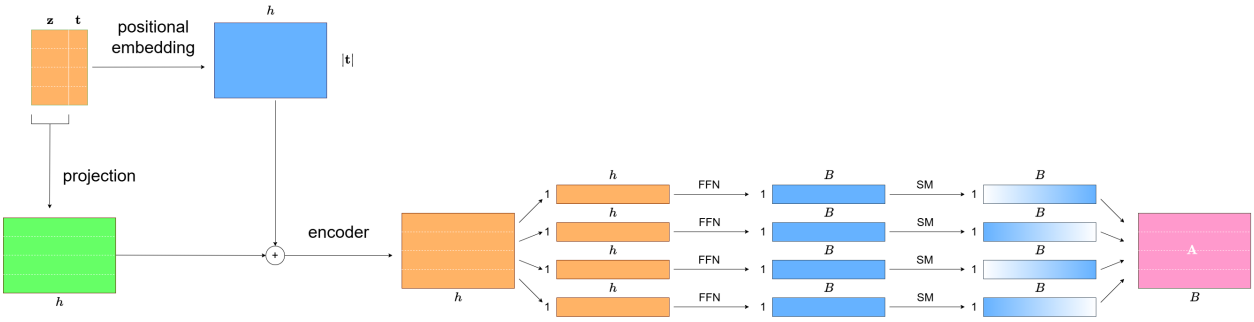

Figure 2: Schematic representation of the association logic. Two data structures $\mathbf{z}$ and $\mathbf{t}$ are processed by the transformer encoder to learn associations. Each row of the encoder output represents the latent space of the corresponding measurement, and undergoes feed-forward (FFN) and softmax (SM) layers to output probabilities of association of the corresponding measurement with a label ID. These rows are concatenated to get the association matrix $\mathbf{A}$.

The measurement and temporal embeddings are added elementwise to form the encoder input,

$$\begin{aligned} \oplus &: \mathbb{R}^{n\times h}, \mathbb{R}^{|\mathbb{S}|\times h} \to \mathbb{R}^{n\times h} \\ \mathbf{e}^0 &= \oplus(\mathbf{z}', \mathbf{t}) = \mathbf{z}' + \mathscr{L}(\mathscr{T}(\mathbf{t})) \end{aligned} \tag{5}$$

Thus, each row of $\mathbf{e}^0$ represents one detection from $Z_{1:T}$, enriched with the identity of the frame in which it was observed. This data structure is then passed to the encoder, as discussed in Appendix A.1. Consider $n_l \in \mathbb{N}$ to be the number of encoder layers such that,

$$\begin{aligned} \mathscr{E}_{n_l} &: \mathbb{R}^{n\times h} \to \mathbb{R}^{n\times h} \\ \mathbf{e} &= \mathscr{E}_{n_l}(\mathbf{e}^0) \end{aligned} \tag{6}$$

Each row of $\mathbf{e}$ is a latent representation of the corresponding detection. In order to translate it to a set of probabilities for each trajectory label, we pass it through a feed-forward network and a final softmax layer. Hence, each row of $\mathbf{e}$ goes through a 3-layer feedforward network for $\mathbf{W}_{f,1} \in \mathbb{R}^{1\times h}$, $\mathbf{W}_{f,2} \in \mathbb{R}^{h\times h}$ and $\mathbf{W}_{f,3} \in \mathbb{R}^{h\times B}$ such that

$$\begin{aligned} \mathscr{F} &: \mathbb{R}^{1\times h} \to \mathbb{R}^{1\times B} \\ \mathscr{F}(\mathbf{e}_p) &= \mathbf{e}_p.\mathbf{W}_{f,1}.\mathbf{W}_{f,2}.\mathbf{W}_{f,3}, \forall p \in [1, n] \end{aligned} \tag{7}$$

For brevity, we omit the representation of the ReLU activation function after each feedforward layer. And finally, we get the association matrix $\mathbf{A} \in \mathbb{R}^{n\times B}$,

$$\mathbf{A} = \text{concat}\left(\text{softmax}\left(\mathscr{F}(\mathbf{e}_p)\right), \forall p \in [1, n]\right) \tag{8}$$

The entire logic is schematized in Fig. 2.

2.1.2 Association Loss Calculation

We assume that we have a ground truth association matrix $\mathbf{A}^* \in \mathbb{R}^{n\times\mathbb{T}}$ each of whose columns is representative of a unique trajectory label and each row is a one-hot vector. However, it is likely that the columns of predicted $\mathbf{A}$ and known $\mathbf{A}^*$ are not matched, more so if $B \neq \mathbb{T}$. We want to match the ground truth trajectory to the predicted trajectory which has the highest number of measurements that match it according to $\mathbf{A}$.

In order to do that, we pad the smaller of the ground truth or prediction matrices with zeros such that $\mathbf{A}, \mathbf{A}^* \in \mathbb{R}^{n \times \max(B,\mathbb{T})}$. We set a cost matrix $\mathbf{C} \in \mathbb{R}^{\max(B,\mathbb{T}) \times \max(B,\mathbb{T})}$ such that,

$$\mathbf{C}_{p,q} = -\sum_{r=1}^{\max(B,\mathbb{T})} \mathbf{A}_{r,p}\mathbf{A}^*_{r,q} \tag{9}$$

This gives a dissimilarity score between the p^{th} predicted trajectory and the q^{th} ground truth trajectory. A more negative dissimilarity score means that the two indices are less dissimilar. Using the Hungarian algorithm (Kuhn (1955)) on $\mathbf{C}$, we get the reordered column order and apply it to $\mathbf{A}^*$. Now, we calculate the cross-entropy loss between $\mathbf{A}$ and the reordered $\mathbf{A}^{*,o}$,

$$\mathcal{L}(\mathbf{A}, \mathbf{A}^{*,o}) = -\frac{1}{n}\sum_{p=1}^{n}\sum_{q=1}^{\max(B,\mathbb{T})} \log(\mathbf{A}_{p,q})\mathbf{A}^{*,o}_{p,q} \tag{10}$$

We thus, match the columns of the ground truth association matrix with those of the predicted association matrix to subsequently calculate cross-entropy between them as a metric of loss.

2.1.3 Assigning associations to their measurements

Each row of $\mathbf{A}$ contains the predicted probabilities that a given detection belongs to the candidate trajectory labels. We assign each detection to the label with highest probability,

$$i_p = \arg\max_q \mathbf{A}_{p,q}, \qquad p = 1, \ldots, n. \tag{11}$$

For each trajectory label i, we then collect the detections with $i_p = i$ into a subset $\mathbf{Z}_i$, which is treated as a candidate trajectory for subsequent Bayesian filtering.

2.1.4 Kalman filtering of a pruned tree of trajectory hypotheses

In this section, we adhere to the well-established Bayesian filtering technique introduced by Kalman (1960). Suppose that we are following a particle i that moves with a constant velocity and its state at time t is described as $\mathbf{x}^i_t = \begin{bmatrix} x^i_t & \dot{x}^i_t & y^i_t & \dot{y}^t_i \end{bmatrix}^T$ where x^i_t and y^i_t are its positions and $\dot{x}^i_t$ and $\dot{y}^i_t$ are its velocity components. Its next state is extrapolated using Kalman filtering with a state transition matrix $\mathbf{F} = \begin{bmatrix} 1 & dt & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and the noise covariance matrix of the process $\mathbf{Q} = \sigma_q^2 \begin{bmatrix} \frac{dt^3}{3} & \frac{dt^2}{2} & 0 & 0 \\ \frac{dt^2}{2} & dt & 0 & 0 \\ 0 & 0 & \frac{dt^3}{3} & \frac{dt^2}{2} \\ 0 & 0 & \frac{dt^2}{2} & dt \end{bmatrix}$ for some process noise parameter σ_q for a sampling time interval dt. See Appendix A.2 for the derivation of $\mathbf{Q}$. Subsequently, the next state and its uncertainty is updated using the observation matrix $\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ and the measurement noise covariance matrix $\mathbf{R} = \sigma_r^2\mathbf{I}_2$ for some measurement noise parameter σ_r. This results in the predicted state at the next time step, i.e., $\hat{\mathbf{x}}^i_{t+1}$ containing the position and speed.

2.2 Multiple Hypothesis Tracking

Here, we briefly discuss MHT as the strategy of comparison against ABHA. It uses the same Kalman filtering model as ABHA but uses a purely Bayesian combinatorial association strategy at each frame. At each time step t, a set of active hypotheses $\mathcal{H}_{t-1}$ is maintained, each representing a predicted object state. Each hypothesis is propagated forward using the Kalman filter as discussed in Section 2.1.4, yielding a predicted measurement. Now, to assign observations to hypotheses, a cost matrix $\mathbf{G}$ is defined each of whose elements contains the Euclidean distance between each predicted and observed measurement. The association is now formulated as a linear assignment problem solved using the Hungarian algorithm (Kuhn (1955)). All assignments beyond a gating threshold τ are discarded. Each matched hypothesis is updated using the same Kalman filter as described above. Unmatched hypotheses age incrementally on increasing time steps and are removed if they exceed a maximum allowed number of misses a (See Table 1). Unassigned measurements are used to initialize new hypotheses.

2.3 Evaluation

2.3.1 Data Description and Generation

We use 30×30 patches from the Viruses dataset of the ISBI Particle Tracking Challenge (Chenouard et al. (2014)) to compare our approach. This dataset reproduces the motion of diffraction-limited spots undergoing stop-and-go dynamics that mimic viral trafficking in vivo. For each ground-truth position χ_t^i, we generate an ideal noisy detection according to the Gaussian measurement model

$$\zeta_t^i = \chi_t^i + \omega_t^i, \tag{12}$$

where $\omega_t^i \sim \mathcal{N}(\mathbf{0}, \sigma_m^2 \mathbf{I}_2)$. We denote by $\mathcal{Z}_t = \{\zeta_t^i\}_{i \in \Omega_t}$ the set of noisy detections associated with the particles active at time t. False negatives are then simulated by removing each element of $\mathcal{Z}_t$ independently with probability p_{fn}. False positives are added independently at each time step according to a Poisson distribution with rate λ_{fp} per unit area, and their positions are sampled uniformly over the field of view. The resulting detection set at time t is thus the detection set $\mathbf{Z}_t$. For training, the identity of the ground-truth particle associated with each retained true detection is preserved in order to construct $\mathbf{A}^*$; for validation and inference, these labels are removed.

2.3.2 Performance Metrics

Point-congruence metrics: At each time step t, the ground-truth particle positions are given by $\mathcal{X}_t = \{\chi_t^i\}_{i \in \Omega_t}$, and the particle positions estimated by the Kalman filter are denoted by $\hat{\mathcal{X}}_t = \{\hat{\chi}_t^j\}_{j \in \{\hat{\Omega}_t\}}$, with $\hat{\Omega}_t \subseteq \{1, \ldots, B\}$ the estimated set of trajectories index estimated by the association strategy at time t. The cost of matching a ground-truth position χ_t^i to an estimated position $\hat{\chi}_t^j$ is defined by the Euclidean distance

$$d(\chi_t^i, \hat{\chi}_t^j) = \|\chi_t^i - \hat{\chi}_t^j\|_2. \tag{13}$$

This defines the cost matrix $\mathbf{D}_t$, with entries

$$(\mathbf{D}_t)_{i,j} = d(\chi_t^i, \hat{\chi}_t^j). \tag{14}$$

The Hungarian algorithm (Kuhn (1955)) is then applied to the Euclidean cost matrix $\mathbf{D}_t$, with matches restricted to pairs satisfying $(\mathbf{D}_t)_{i,j} \leq d_\phi$, to obtain the optimal one-to-one matching set $\mathcal{M}_t$ between ground-truth and estimated particle positions at time t. Thus, $\mathcal{M}_t$ contains the matched index pairs (i, j) such that χ_t^i is assigned to $\hat{\chi}_t^j$. We denote by $\mathcal{M} = \mathcal{M}_{1:T}$ the set of matches over the full sequence. Point precision P_P, recall R_P and Jaccard similarity coefficient, JSC_P are defined as follows:

$$P_P = \frac{|\mathcal{M}|}{|\hat{\mathcal{X}}|} \tag{15}$$

$$R_P = \frac{|\mathcal{M}|}{|\mathcal{X}|} \tag{16}$$

$$JSC_P = \frac{|\mathcal{M}|}{|\mathcal{X}| + |\hat{\mathcal{X}}| - |\mathcal{M}|} \tag{17}$$

Link-congruence metrics: A link is counted as correct when the detection in a ground-truth particle is matched to the same predicted trajectory in the predicted set in two consecutive frames, Let $\mathbf{L}$ denote the set of all such correct links over the full sequence. We then define link precision P_L, recall R_L, F1-score $F1_L$, and Jaccard similarity coefficient JSC_L by normalizing $|\mathbf{L}|$ by the total numbers of predicted and ground-truth consecutive links.

$$P_L = \frac{|\mathbf{L}|}{|\mathbf{L}_{\hat{\mathcal{X}}}|} \tag{18}$$

$$R_L = \frac{|\mathbf{L}|}{|\mathbf{L}_{\mathcal{X}}|} \tag{19}$$

Table 1: Dictionary of all parameters used for the methodology and evaluations

T	B	x_{lim}, y_{lim}	h	d_k	d_v	d_{ffn}	n_l	n_h	dt	σ_q	σ_r	τ	a
100	20	30	128	16	16	1024	6	8	10^{-1}	10^{-3}	10^{-3}	10	2

Table 2: Task initialization for comparison

	ϕ	A	B	C	1	2	3	4
σ_m	0	0	1	1	1	1	3	3
p_{fn}	0	10^{-1}	10^{-1}	0	10^{-1}	10^{-1}	10^{-1}	10^{-1}
λ_{fp}	0	5×10^{-5}	0	5×10^{-5}	5×10^{-5}	2.5×10^{-4}	2.5×10^{-4}	3.3×10^{-4}

$$F1_L = \frac{2P_L R_L}{P_L + R_L} \tag{20}$$

$$JSC_L = \frac{|\mathbf{L}|}{|\mathbf{L}_{\mathcal{X}}| + |\mathbf{L}_{\hat{\mathcal{X}}}| - |\mathbf{L}|} \tag{21}$$

TGOSPA: We use Trajectory Generalized Optimal Sub-Pattern Assignment, TGOSPA, (García-Fernández et al. (2020)) to evaluate the accuracy of predicted trajectories over time. TGOSPA extends the GOSPA (Rahmathullah et al. (2017)) metric by aggregating spatial and cardinality errors across all frames. For each frame t, GOSPA (Appendix A.3) is computed between the sets of ground truth and predicted positions, and TGOSPA is defined as

$$TGOSPA = \left(\frac{1}{T} \sum_{t=1}^{T} d_p^c(\mathcal{X}_t, \hat{\mathcal{X}}_t)^p \right)^{1/p} \tag{22}$$

where d_p^c is the GOSPA distance at time t, p is the order of the metric, c is the cutoff threshold. A lower TGOSPA indicates better spatial and temporal consistency in the predicted trajectories.

The dictionary of all parameters for the performance metrics can be found in Appendix Table 6.

2.3.3 Implementation Details

Method parameters and training details: We obtain the patches of (x_{lim}, y_{lim}) pixels, as defined in Table 1 by slicing the dataset with $SNR = 4$ for low and medium densities from Chenouard et al. (2014). The other parameters are also as mentioned in Table 1. The association strategy is trained with 8 such patches and validated on 2 patches. Training is done with a starting learning rate of 10^{-3} and the cyclical annealing strategy of the learning rate (Smith (2017)) is employed throughout training. Along with this learning rate annealing strategy, we also use a Jaccard similarity coefficient between $\mathbf{A}$ and $\mathbf{A}^{*,o}$. First, $\mathbf{A}$ is converted to a binary matrix $\mathbf{A}^b$ such that any value ≥ 0.5 becomes 1 and < 0.5 becomes 0. Then, the metric is calculated as:

$$JSC_A = \frac{|\mathbf{A}^b \cap \mathbf{A}^{*,\mathbf{o}}|}{|\mathbf{A}^b \cup \mathbf{A}^{*,\mathbf{o}}|} \tag{23}$$

While using cross-entropy as the metric for gradient descent helps in smooth loss reduction, it has certain limitations: (1) it is hard to know what a low cross-entropy loss is, and hence (2) knowing how long to train for is not straightforward. Having an upper-bound at 1, JSC_A helps bypass this problem. Training is done so that JSC_A reaches at least 0.8. As such, a training session takes ~ 21 hours. All implementation is done in Python version 3.8.10 and a GPU cuda version 11.2.

Data parameters and tasks: A patch of $x_{lim} \times y_{lim}$ pixels is taken, and corruptions are made to its ground truth to simulate measurements as described above. Different scenarios called tasks are defined containing the initializations of the parameters used for the said corruptions as described in Table 2. Task ϕ is a baseline task with no noise added to it. Tasks A, B and C introduce a moderate level of two of the three types of corruption. Tasks 1, 2, 3 and 4 are more realistic cases.

3 Results

We organize the presentation of our results into two main parts. First, in Section 3.1, our goal is to evaluate the improvement ABHA offers over the classical baseline introduced in Section 2.2. To do this, we assess ABHAs performances using a set of quantitative metrics, followed by qualitative examples to help interpreting the former. These analyses show that ABHA improves significantly over the baseline, thanks to its conservative tracking that focuses on robust high-confidence associations. We also examine how ABHA performs in challenging experimental data that simulate a non-uniform and stochastic illumination typically used for super-resolution microscopy, demonstrating how it can be used beyond conventional benchmark conditions. Second, in Section 3.2, we report a series of experiments and ablation studies that justify the parameterization of our network architecture and demonstrate the importance of the attention layer and additional non-linearities to reach performance goals.

3.1 Benchmarking

3.1.1 ABHA surpasses classical MHT in noisy tracking regimes and reduces tracking artifacts

We first examine[1] the TGOSPA scores for ABHA and MHT across all tasks in medium and low density scenarios in Table 3. TGOSPA captures location errors, missed and false detections, identity switches, and track fragmentation and lower scores indicate better performance. ABHA consistently achieves lower TGOSPA scores than MHT across most tasks, particularly under medium density conditions. This indicates that ABHA makes more accurate overall trajectory associations. The only exception to this pattern is observed in idealized baseline tasks, where MHT performs better. This is due to the simplicity of the setting that aligns perfectly with the a priori modeling and the greedy association strategy of MHT. This observation highlights that ABHA selects high-confidence associations. This conservative strategy results in fewer false positives, which are heavily penalized by TGOSPA, thus resulting in a better performance.

Table 3: TGOSPA comparison between ABHA and MHT across tasks in medium and low density settings (SNR = 4).

Task	**Description**			**Medium Density**		**Low Density**	
	σ_m	p_{fn}	λ_{fp}	**ABHA**	**MHT**	**ABHA**	**MHT**
ϕ	×	×	×	1.294 ± 0.114	0.837 ± 0.006	2.312 ± 0.012	1.283 ± 0.003
A	×	✓	✓	2.174 ± 0.017	2.085 ± 0.013	3.476 ± 0.102	3.109 ± 0.023
B	✓	✓	×	2.222 ± 0.103	2.296 ± 0.003	3.199 ± 0.130	1.632 ± 0.074
C	✓	×	✓	1.763 ± 0.011	1.815 ± 0.002	3.000 ± 0.115	2.155 ± 0.101
1	✓	✓	✓	2.111 ± 0.006	2.390 ± 0.034	2.105 ± 0.037	2.884 ± 0.006
2	✓	✓	✓+	2.612 ± 0.041	4.307 ± 0.101	2.040 ± 0.022	3.746 ± 0.041
3	✓+	✓	✓+	2.313 ± 0.088	4.528 ± 0.015	2.136 ± 0.103	5.154 ± 0.127
4	✓+	✓	✓++	2.116 ± 0.094	5.743 ± 0.103	2.114 ± 0.092	5.382 ± 0.133

For a more detailed understanding of the performance behavior (Tables 4 and 5), we compare the performance of both methods in terms of the other metrics defined in Section 2.3.2. ABHA generally scores higher in link precision P_L and comparable or slightly lower in link recall R_L relative to MHT. This indicates that while ABHA misses some true links, it is more accurate when it indeed decides to link. Similarly, its $F1_L$ and JSC_L scores are competitive, indicating that ABHA tries to construct viable trajectories and not just avoid errors.

To illustrate how the method behaves in practice, Fig. 3 shows the association map **A** alongside the time-accumulated tracking results for task 2 of the virus-trafficking dataset. Fig. 4 highlights a representative scenario where false positives and missed detections challenge data association. In this regime, our approach

[1]All scores in Tables 3, 4 and 5 are reported as mean of 3 runs. TGOSPA scores are reported as mean ± standard deviation of 3 runs. Other metrics showed negligible variability and are omitted for brevity.

Table 4: Comparison of ABHA and MHT on a patch of Viruses dataset for SNR = 4 and medium density.

Task	**Description**			**Method**	R_P	P_P	JSC_P	P_L	R_L	$F1_L$	JSC_L
	σ_m	p_{fn}	λ_{fp}								
ϕ	×	×	×	ABHA	0.974	0.974	0.948	0.957	0.908	0.932	0.466
				MHT	0.990	0.992	0.996	0.977	0.994	0.986	0.493
A	×	✓	✓	ABHA	0.912	0.999	0.853	0.990	0.798	0.896	0.448
				MHT	0.943	0.984	0.845	0.947	0.838	0.889	0.445
B	✓	✓	×	ABHA	0.907	0.994	0.823	0.772	0.643	0.832	0.411
				MHT	0.917	0.990	0.864	0.933	0.803	0.863	0.432
C	✓	×	✓	ABHA	0.872	0.988	0.994	0.983	0.754	0.943	0.564
				MHT	0.998	0.942	0.883	0.939	0.994	0.966	0.483
1	✓	✓	✓	ABHA	0.914	0.988	0.876	0.943	0.677	0.877	0.534
				MHT	0.928	0.989	0.855	0.900	0.786	0.839	0.419
2	✓	✓	✓+	ABHA	0.923	0.989	0.865	0.976	0.705	0.869	0.539
				MHT	0.990	0.873	0.804	0.815	0.867	0.840	0.420
3	✓+	✓	✓+	ABHA	0.884	0.935	0.823	0.882	0.564	0.773	0.530
				MHT	0.979	0.823	0.720	0.612	0.647	0.629	0.315
4	✓+	✓	✓++	ABHA	0.845	0.902	0.811	0.834	0.561	0.784	0.511
				MHT	0.979	0.812	0.681	0.544	0.532	0.546	0.278

Table 5: Comparison of ABHA and MHT on a patch of Viruses dataset for ($SNR = 4$) and low density.

Task	**Description**			**Method**	R_P	P_P	JSC_P	P_L	R_L	$F1_L$	JSC_L
	σ_m	p_{fn}	λ_{fp}								
ϕ	×	×	×	ABHA	0.902	0.944	0.820	0.811	0.748	0.778	0.389
				MHT	0.964	0.956	0.946	0.922	0.922	0.922	0.461
A	×	✓	✓	ABHA	0.839	0.847	0.757	0.790	0.621	0.696	0.348
				MHT	0.911	0.785	0.645	0.823	0.767	0.794	0.397
B	✓	✓	×	ABHA	0.813	0.930	0.690	0.770	0.553	0.644	0.322
				MHT	0.955	0.981	0.911	0.979	0.913	0.945	0.472
C	✓	×	✓	ABHA	0.875	0.875	0.766	0.774	0.699	0.735	0.367
				MHT	0.991	0.853	0.748	0.900	0.961	0.930	0.465
1	✓	✓	✓	ABHA	0.839	0.931	0.881	0.877	0.544	0.671	0.533
				MHT	0.973	0.858	0.772	0.848	0.815	0.831	0.416
2	✓	✓	✓+	ABHA	0.843	0.675	0.499	0.769	0.549	0.641	0.540
				MHT	0.973	0.661	0.483	0.619	0.709	0.661	0.330
3	✓+	✓	✓+	ABHA	0.874	0.843	0.885	0.812	0.534	0.644	0.381
				MHT	0.973	0.568	0.397	0.488	0.602	0.539	0.270
4	✓+	✓	✓++	ABHA	0.887	0.932	0.843	0.932	0.441	0.599	0.392
				MHT	0.964	0.532	0.358	0.357	0.437	0.393	0.196

remains robust to large apparent displacements, while MHT exhibits characteristic linking artifacts due to the permissive pruning strategy (pink arrows). At $t = 20$, a false detection initializes a spurious hypothesis. At $t = 21$, this hypothesis remains active despite the absence of a supporting detection, because the MHT formulation explicitly allows missed detections. In cluttered conditions, this measurement-less propagation

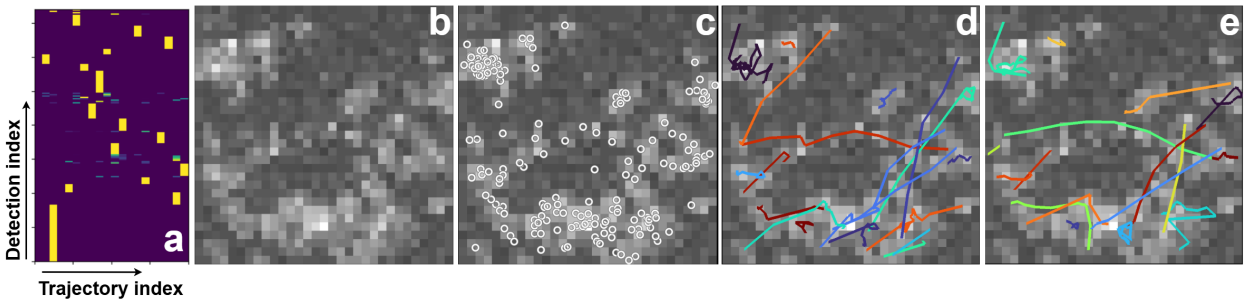

Figure 3: Visualization results of task 2 on medium density scenario. (a) predicted association matrix. (b) maximum intensity projection of time accumulated raw image. (c) time accumulated measurements. (d) time accoumulated ground-truth trajectories. (e) time accumulated predicted trajectories.

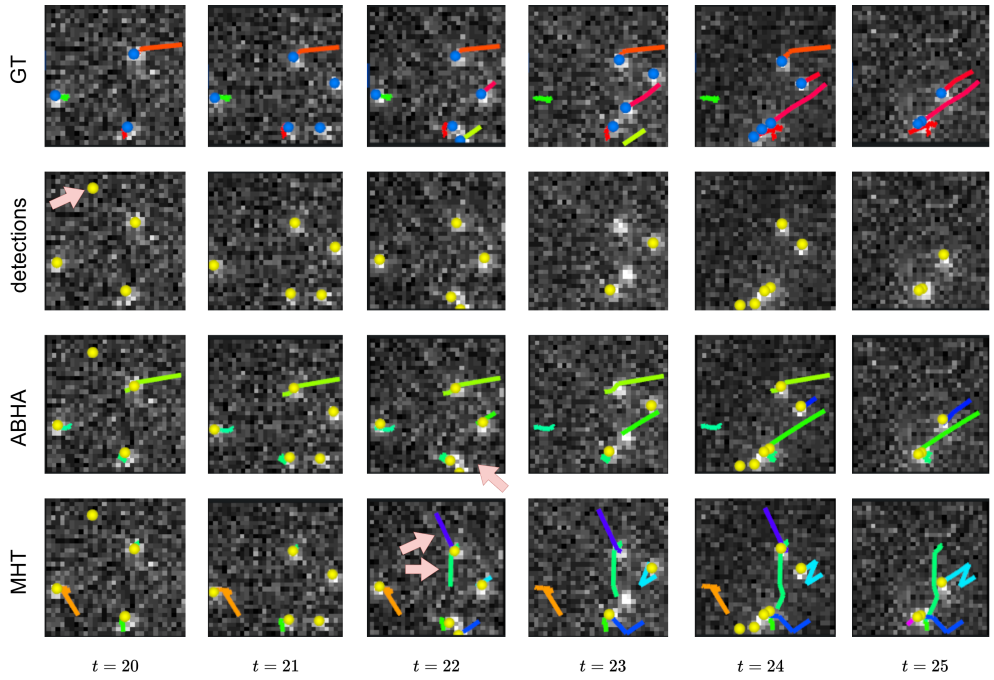

Figure 4: Frame-by-frame comparison of tracking behavior for virus trafficking under noise task 2 and $SNR = 4$ with ground truth (GT), raw detections, ABHA, and MHT. This example shows how an initial wrongful selection of hypotheses by MHT can self-reinforce into false tracks throughout. ABHA circumvents this problem by being less aggressive in its associations.

preserves unlikely branches that can later compete with valid ones during assignment (e.g., $t = 22$). Once such a branch is selected, subsequent missed detections within the gating threshold can reinforce the error, eventually suppressing the correct trajectory. Reducing the gating threshold could suppress some of these events, but our results show that this comes at the cost of degraded overall performance (Fig. 10). This illustrates how robustness to false negatives in MHT can also enable the persistence and self-reinforcement of artifactual tracks.

Fig. 5 illustrates a similar effect in a scenario where tracking is challenged by particle density and false positives. At $t = 84$, a false positive detection is incorrectly assigned by MHT as a valid continuation of an existing track. As in Fig. 4, this initial error propagates over time: the incorrect hypothesis is reinforced through subsequent associations, leading to additional erroneous links (pink arrows at $t = 85$) and ultimately to spurious trajectories. In contrast, ABHA maintains globally consistent and smooth trajectories through

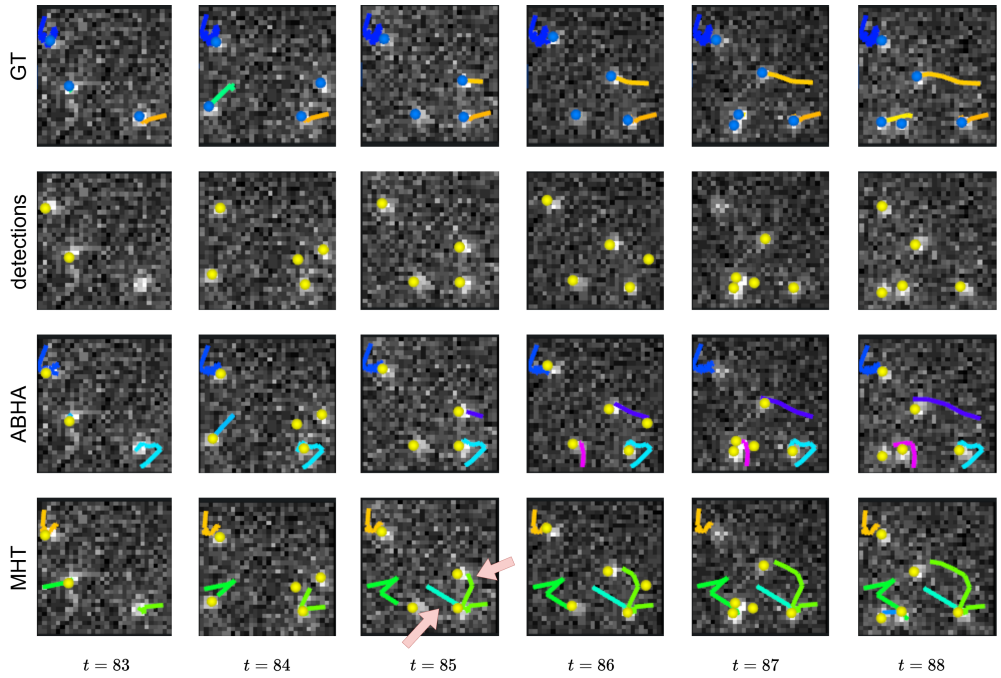

Figure 5: Frame-by-frame example illustrating trajectory merging for virus trafficking under noise task 2 with $SNR = 4$.

such ambiguous regions, as its associations rely on a combination of temporal consistency and geometric affinity.

3.1.2 Conservative associations improve precision at the cost of recall

Interestingly in Tables 4 and 5, we also observe that ABHA consistently exhibits lower recall, both in terms of its points (R_P) and its links (R_L). We hypothesize that this reduction primarily originates from the association stage, where the learned model deliberately avoids assigning uncertain detections to maintain hypothesis purity. The subsequent Kalman filtering stage then reinforces these selective associations rather than compensating for missing links, leading to a lower but more reliable set of trajectories. This trade-off, while resulting in fewer overall associations, proves advantageous in cluttered or noisy regimes, where aggressive linking can lead to compounding errors and identity fragmentation.

A representative example of this behavior is shown in Fig. 6. Here, the ground truth forms a continuous trajectory from $t = 71$ to $t = 73$, but ABHA intentionally leaves the ambiguous detection at $t = 72$ unlinked, creating a short gap that resolves itself once the observations regain clarity. This is precisely the trade-off behind ABHAs lower recall: it avoids uncertain detections to preserve track correctness. MHT, however, illustrates the opposite effect; its inclination to maintain continuity leads it, at $t = 70$, to connect two nearby tracks, emphasizing how aggressive linking can produce lasting identity errors.

Thus, ABHA is as a conservative yet accurate tracker particularly suited for cluttered regimes. ABHA shows the ability to discard detections that might be ambiguous while still reconstructing coherent tracks. This is because it keeps track of the probabilities of all association-links, no matter how uncertain. This helps us provide insight into how combining explicit filtering with a learned hypothesis-pruning strategy could offer better performance and adaptability in complex tracking scenarios.

Furthermore, our comparison of ABHA with Trackastra (Gallusser & Weigert, 2024) in Section A.5 shows that ABHA achieves better $TGOSPA$ scores and slightly lower recall than Trackastra, indicating that although Trackastra is more cautious in its associations than MHT, it does not match the performance of ABHA. However, these results should be interpreted with caution because of differences in scope between

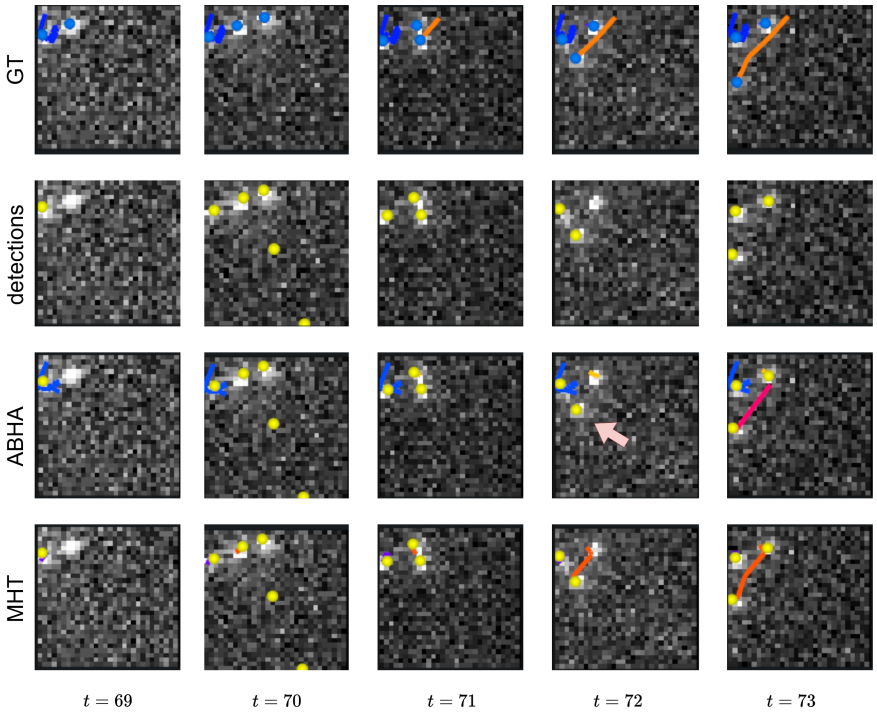

Figure 6: Frame-by-frame example illustrating the low recall tradeoff of ABHA for virus trafficking under noise task 2 with $SNR = 4$.

the two approaches: (1) Trackastra was designed for cell tracking and does not allow frames with zero detections, which are common in our data; (2) assessing how scaling down Trackastra to match the relatively lower number of trajectories in our study affects its scores is beyond the scope of this work; and (3) we use a pretrained Trackastra model, whereas ABHA is specifically trained for all the tasks presented.

3.1.3 Application to random illumination imaging

In this section, we apply ABHA to a challenging use-case simulating fluorescence microscopy under non-uniform and randomized illumination. We use speckle-illumination-structured pattern that is robust to aberration, and provides super-resolution and improved sensitivity after reconstruction on static sample (Mudry et al., 2012; Mangeat et al., 2021). The disadvantage of speckle microscopy on live sample comes the challenge in recovering both the temporal structure of the illumination and the particle motions. Indeed, the spotty nature of the illumination pattern presents a spatially varying field of bright and dark spot that is renewed randomly at each time-step. As such, the fluorescence of particles excited with this highly contrated illumination pattern can often fall below the detection threshold. This provide an compelling use-case for ABHA, considering its resistance to false negative. For each time step t, the sample excited by our speckle pattern at position x is

$$I_{ideal}(x,t) = \sum_{i=1}^{|\Omega_t|} A.\delta(x - \chi_t^i).I_E(t) \tag{24}$$

where δ is the dirac-delta function and $I_E(t)$ is the speckle pattern as defined in (Mudry et al., 2012). Subsequently, the signal emitted by sample is diffracted by the optical system as

$$I_{blurred}(x,t) = (I_{ideal} \star \mathcal{G}_{\sigma_{PSF}})(x,t), \tag{25}$$

where $\star$ is the convolution function operation and $\mathcal{G}$ is the response of the optical system to a punctual light source, characterized by its scale σ_{PSF}. We then simulate the typical noise footprint found in a microscopy

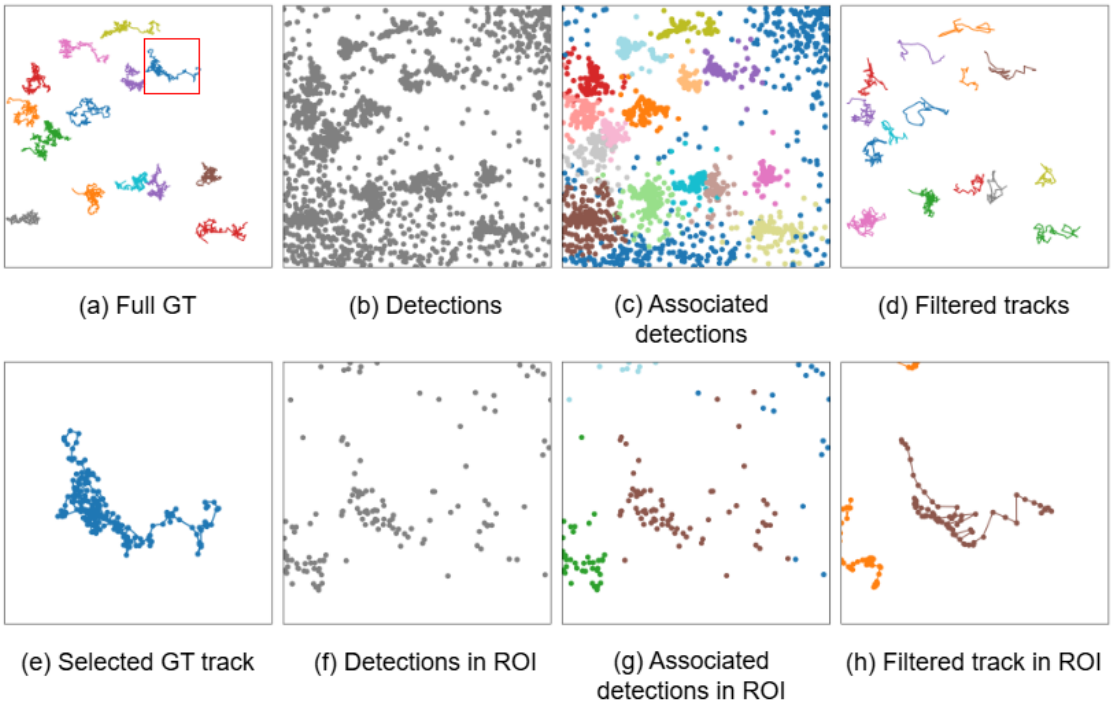

Figure 7: Qualitative results on simulated Brownian motion under speckle illumination. The figure shows (a) ground-truth trajectories, (b) the full detection cloud, (c) the output of the association module, and (d) the final reconstructed tracks, together with (e-h) zoomed views of the highlighted region.

image coming from Photon counting and camera offset and read-out noise (Vonesch et al., 2006):

$$I(x,t) = N(I_{blurred}(x,t)) + \epsilon \tag{26}$$

where $N(I_{blurred}) \sim \text{Poisson}(I_{blurred})$ and $\epsilon \sim \mathcal{N}(\mu_{cam}, \sigma^2_{cam})$. As such, this process simulates the excitation, tissue aberration, diffraction by the microscope objective lense, photon collection and the camera readout. To generate the measurements, we detect the points at each time-point using a maximum-likelihood estimator described in Aguet et al. (2013), resulting in a noisy detection set Z_t.

Fig. 7 illustrates the results of the proposed tracking framework on simulated detections of particles undergoing Brownian motion under speckle illumination described above. Panel (a) shows the true motion paths of all simulated particles over time which follow a Brownian motion. Panel (b) overlays all detections accumulated across time. Each gray point corresponds to a single noisy detection on the image. The stochastic nature of our illumination pattern combined with abberation and photon-limited rate produces a blinking pattern made from the high false positive and negative rates. Panel (c) shows the detection cloud after the application of the data association strategy of ABHA. It illustrates how this strategy successfully clusters detections into coherent tracks. Panel (d) shows the final trajectories obtained after tracking each cluster of associated detections using Kalman filtering. Compared to (a), the reconstructed paths follow the general structure of the ground truth but contain small deviations consistent with measurement noise and occasional association uncertainty. Panels (e-h) isolate this entire procedure to show the effect on one singular trajectory throughout time (highlighted in red in panel (a)). Panel (e) shows the real trajectory isolated from its neighbor and the blinking nature of detections can be observed in panel (f). Panels (h) shows how how the reconstructed path reproduce the general structure of the ground truth trajectory despite the high level of clutter and false negative detection.

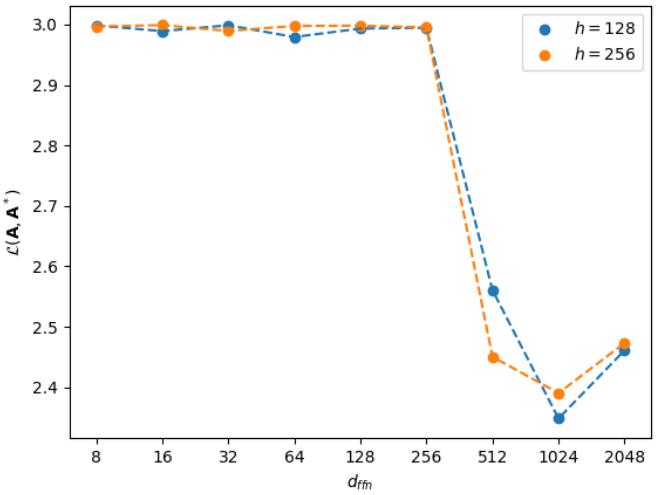

Figure 8: Loss of the association module as a function of the feedforward dimension d_{ffn}. Performance improves sharply up to $d_{\text{ffn}} = 1024$, after which gains saturate, with minimal sensitivity to the hidden size h.

3.2 Network parameterization and ablation experiments

3.2.1 Feed-forward depth and non-linearity govern association accuracy

In order to study the effect of feedforward network dimension on association performance, we train the association matrix described in Section 2.1 across several parameter configurations. We use a scenario of 10 batches of 20 particles undergoing Brownian motion for 100 time steps, where each batch corresponds to one of the four realistic tasks $1, 2, 3, 4$. Fig. 8 shows the training loss $\mathcal{L}(\mathbf{A}, \mathbf{A}^*)$ for two hidden sizes, $h = 128$ and $h = 256$, as a function of the feedforward dimension d_{ffn}.

For small feedforward sizes ($d_{\text{ffn}} \leq 256$), the loss remains approximately constant near 3.0. This behavior is consistent with the interpretation that the model capacity in this regime is primarily constrained by the feedforward sub-network. A clear decrease in loss appears once d_{ffn} exceeds 512, reaching a minimum around $d_{\text{ffn}} = 1024$. Beyond this scale, the loss does not further improve and shows slight variability, which may reflect diminishing returns or mild optimization sensitivity at larger model sizes. The two curves follow similar trends across all settings, suggesting that, for the range of configurations explored here, the feedforward dimension has a stronger influence on association accuracy than the choice of hidden size h.

Note that although only 10 simulated sequences are used per training run, each sequence contains 20 particles over 100 steps, yielding thousands of supervised association examples. This is in line with the current training practices (Spilger et al., 2021; Zhang & Yang, 2023).

We then sought to assess the necessity of the non-linearity introduced by the feedforward layer. Indeed, the RelU activation introduced in Eq. 7 function layers introduces additional non-linearity after the encoder. As such, it might not immediately be clear whether this extra layer is strictly necessary, since the transformer encoder itself captures non-linear relationships. To investigate this, we design two alternative versions of the association module.

1. `no_FFN`: The feedforward layer is completely removed, and the hidden size is adjusted as $h = B$ to maintain dimensional consistency.

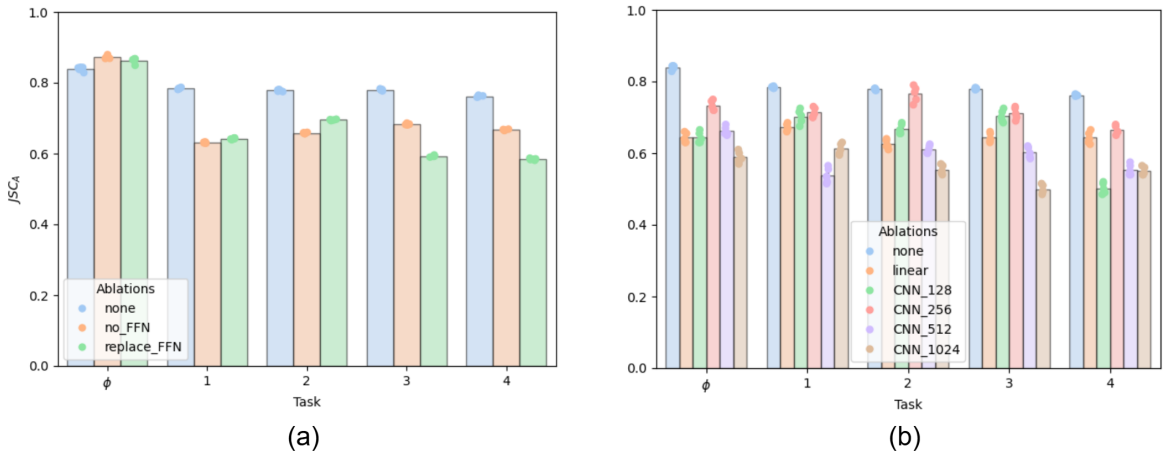

Figure 9: Bar plots showing the JSC_A scores of different versions of the association strategy without (a) eq. 7, and (b) eq. 6.

2. `replace_FFN`: The feedforward layer is replaced with a purely linear projection from the encoder output dimension to the output dimension, testing whether a simple linear mapping is sufficient without the non-linear activation.

Both versions of the association strategy are trained on the virus data set, and the corresponding JSC_A scores are recorded for tasks ϕ, 1, 2, 3, 4 (one clean baseline and 4 noisy and realistic tasks). All models are trained for 10^5 epochs in 10 batches containing information on 17.1 ± 1.889 particles on average. The recorded scores are the mean scores of the last five epochs before convergence.

The results in Fig. 9(a) show that the original pipeline consistently outperforms both alternative designs, particularly in realistic and noisy tasks. Removing the FFN or replacing it with a linear layer reduces performance, demonstrating that the additional non-linearity is beneficial for learning robust trajectory associations in challenging conditions. Thus, while the transformer encoder captures substantial non-linear dependencies, the post-encoder feedforward layer adds complementary non-linear capacity that improves association accuracy under noise and clutter, justifying its inclusion in the ABHA module.

3.2.2 Attention is essential for stable and context-aware association

In addition to the previous ablation study, we sought to test whether the transformer encoder is truly necessary for association in ABHA, or whether conventional architectures that do not have an attention layer could achieve comparable performance. To investigate this, we designed a set of experiments replacing the transformer encoder with alternative strategies:

1. Linear projection: A direct linear mapping of the input features to the output dimensions.

2. `CNN_128`: A convolutional network projecting the input to 128 channels and then to the original hidden dimensions. Note that the hidden dimension h is also 128, so in this case, there is no parametric expansion in the hidden layer of the convolutional network.

3. `CNN_256`: Similar to `CNN_128` but with 256 intermediate channels.

4. `CNN_512`: Similar to `CNN_128` but with 512 intermediate channels.

5. `CNN_1024`: Similar to `CNN_128` but with 1024 intermediate channels.

These alternatives were chosen to test whether linear or convolutional processing, with increasing representational capacity, could approximate the effect of the transformers attention mechanism. All architectures were trained on the virus dataset across the same set of tasks as in the previous ablation (one clean baseline and four noisy, realistic scenarios). Performance was measured using the JSC_A metric, averaged over the last five epochs before convergence.

Fig. 9(b) shows that the original ABHA pipeline with the transformer encoder consistently achieves higher scores across nearly all tasks. There is one edge case in task 2, where `CNN_256` appears to approach the transformers performance. However, statistical tests confirm the difference: a Students t-test yields a p-value of 0.0265, and Welchs t-test yields 0.0313, demonstrating significance at the 95% confidence level.

The wider variability observed across the CNN-based variants (Fig. 9(b)) compared to the FFN ablations (Fig. 9(a)) reflects the limitations of CNNs for an association task. Indeed, CNNs learn only local spatial patterns, so their performance depends strongly on incidental spatial arrangements within each task. This can explain both the occasional near-match to the transformer (as in task 2) and the sharp drops in other settings. In contrast, the transformer explicitly models interactions between all detections. These results suggest that the transformers advantage is not simply higher capacity, but its ability to capture the global, permutation-invariant structure of the association problem. Hence, the transformer encoder provides essential context-aware processing that cannot be fully replicated by linear or convolutional alternatives. This validates its inclusion in ABHA.

4 Conclusion, Limitations and Future Scope

We propose a hybrid multiple particle tracking framework, ABHA, that integrates the self-attention mechanism of the transformer architecture and Bayesian filtering. By casting data association as a label prediction problem, and hence using the transformer encoder to obtain association scores, we effectively prune the combinatorially large hypothesis space before applying Kalman filtering. This modularity of learning-based and model-based components allows our method to maintain robustness in noisy and cluttered environments while maintaining interpretability, say, in the form of the ability to evaluate the likelihood of a trajectory through the predicted association matrix. As such, we demonstrate that the proposed approach improves tracking accuracy according to several metrics, particularly in high-noise settings where traditional methods degrade. These results support the idea that attention contributes effectively at capturing relational structure of the data, making it well suited for complex association tasks that benefit from learned priors. Through several examples we show that ABHA is a conservative tracker. This is the result of a trade-off between precision and recall. As a consequence, ABHA is very cautious in linking detections, and would rather not link dubious detections instead of making a wrongful link. It compensates for this conservative behavior by considering trajectory-to-measurement associations that are transiently uncertain as gaps in a trajectory, and still assigned the precedent and future detection to the same trajectory. As such, ABHA increase robustness by reconstructing tracks correctly despite transient ambiguity.

The modular design of ABHA allows for weaker supervision than other tracking approaches based on neural networks. The only component that is learned is the association module, whose purpose is to link detections across frames into coherent trajectories. Crucially, training this module does not require access to the ground-truth physical states of the particles, unlike other end-to-end approaches based on neural networks (Spilger et al., 2021; Pinto et al., 2023; Gallusser & Weigert, 2024). Instead, it only relies on identity-level supervision, i.e., knowing which detections correspond to the same particle across time. This decoupling means that ABHA avoids the strong supervision requirements that are typical of end-to-end trackers, where the network must learn both the dynamics and the associations implicitly from fully annotated trajectories. As a result, ABHA can be trained efficiently on partially annotated data and is less sensitive to errors in motion-ground-truth generation. This property is particularly valuable in microscopy settings, where accurate physical state labels are costly or impossible to obtain.

More broadly, this work suggests that hybrid tracking strategies combining learned global association priors with interpretable Bayesian state estimation are a promising direction for multiple-particle tracking. In the future, ABHA could be extended to richer motion models, adaptive uncertainty estimation, and partially or weakly supervised training schemes that reduce the need for exhaustive trajectory annotations. Such

developments may improve robustness across imaging modalities and make the framework applicable to a broader range of dynamic intracellular processes.

Limitations of the study and perspectives: Our training and evaluation are limited to a single type of particle motion, a mix of Brownian and directed dynamics typical of virus trafficking. While this dynamic is of high biological relevance, it would be insightful to test our method on a broader set of motion types and mixtures. Moreover, due to computational constraints, we restrict training and evaluation to (30×30) patches of the full field of view, allowing the tracking of $\sim$ 20 particles at a time. This prevents the study from acting as a full benchmark but a comparison within a limited spatial and motion context. Nevertheless, it enables the precise study of the benefit of our hybrid approach when compared to conventional Bayesian techniques. The primary goal of the future scope of this study is to address these limitations. We will move towards a more generalizable training of this tracking strategy to infer on different types of biologically relevant movements. In addition, strategies will be implemented to work in a larger field of view for a more standardized comparison.

References

François Aguet, Costin N Antonescu, Marcel Mettlen, Sandra L Schmid, and Gaudenz Danuser. Advances in Analysis of Low Signal-to-Noise Images Link Dynamin and AP2 to the Functions of an Endocytic Checkpoint. *Developmental cell*, 26(3):279–291, 2013.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Antoine Basset, Jérôme Boulanger, Jean Salamero, Patrick Bouthemy, and Charles Kervrann. Adaptive spot detection with optimal scale selection in fluorescence microscopy images. *IEEE Transactions on Image Processing*, 24(11):4512–4527, 2015.

Clifford P. Brangwynne, Gijsje H. Koenderink, Frederick C. MacKintosh, and David A. Weitz. Intracellular transport by active diffusion. *Trends in Cell Biology*, 19(9):423–427, 2009.

Bi-Chang Chen, Wesley R. Legant, Kai Wang, Lin Shao, Daniel E. Milkie, Michael W. Davidson, Chris Janetopoulos, Xufeng S. Wu, John A. Hammer, Zhe Liu, Brian P. English, Yuko Mimori-Kiyosue, Daniel P. Romero, Alex T. Ritter, Jennifer Lippincott-Schwartz, Lillian Fritz-Laylin, R. Dyche Mullins, Diana M. Mitchell, Joshua N. Bembenek, Anne-Cecile Reymann, Ralph Böhme, Stephan W. Grill, Jennifer T. Wang, Geraldine Seydoux, U. Serdar Tulu, Daniel P. Kiehart, and Eric Betzig. Lattice light-sheet microscopy: Imaging molecules to embryos at high spatiotemporal resolution. *Science*, 346(6208):1257998, 2014a. ISSN 0036-8075, 1095-9203.

Jiji Chen, Zhengjian Zhang, Li Li, Bi-Chang Chen, Andrey Revyakin, Bassam Hajj, Wesley Legant, Maxime Dahan, Timothée Lionnet, Eric Betzig, et al. Single-molecule dynamics of enhanceosome assembly in embryonic stem cells. *Cell*, 156(6):1274–1285, 2014b.

Nicolas Chenouard, Isabelle Bloch, and Jean-Christophe Olivo-Marin. Multiple hypothesis tracking for cluttered biological image sequences. *IEEE transactions on pattern analysis and machine intelligence*, 35 (11):2736–3750, 2013.

Nicolas Chenouard, Ihor Smal, Fabrice De Chaumont, Martin Maška, Ivo F Sbalzarini, Yuanhao Gong, Janick Cardinale, Craig Carthel, Stefano Coraluppi, Mark Winter, et al. Objective comparison of particle tracking methods. *Nature methods*, 11(3):281–289, 2014.

Ana F David, Philippe Roudot, Wesley R Legant, Eric Betzig, Gaudenz Danuser, and Daniel W Gerlich. Augmin accumulation on long-lived microtubules drives amplification and kinetochore-directed growth. *Journal of Cell Biology*, 218(7):2150–2168, 2019.

Kevin M. Dean, Philippe Roudot, Carlos R. Reis, Erik S. Welf, Marcel Mettlen, and Reto Fiolka. Diagonally Scanned Light-Sheet Microscopy for Fast Volumetric Imaging of Adherent Cells. *Biophysical Journal*, 110 (6):1456–1465, 2016. ISSN 0006-3495.

Albert Dominguez Mantes, Antonio Herrera, Irina Khven, Anjalie Schlaeppi, Eftychia Kyriacou, Georgios Tsissios, Evangelia Skoufa, Luca Santangeli, Elena Buglakova, Emine Berna Durmus, Suliana Manley, Anna Kreshuk, Detlev Arendt, Can Aztekin, Joachim Lingner, Gioele La Manno, and Martin Weigert. Spotiflow: Accurate and efficient spot detection for fluorescence microscopy with deep stereographic flow regression. *Nature Methods*, 22(7):1495–1504, July 2025. ISSN 1548-7105.

Thomas Fortmann, Yaakov Bar-Shalom, and Molly Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE journal of Oceanic Engineering*, 8(3):173–184, 1983.

Benjamin Gallusser and Martin Weigert. Trackastra: Transformer-based cell tracking for live-cell microscopy. In *European Conference on Computer Vision*, pp. 467–484. Springer, 2024.

Ángel F García-Fernández, Abu Sajana Rahmathullah, and Lennart Svensson. A metric on the space of finite sets of trajectories for evaluation of multi-target tracking algorithms. *IEEE Transactions on Signal Processing*, 68:3917–3928, 2020.

Khuloud Jaqaman, Dinah Loerke, Marcel Mettlen, Hirotaka Kuwata, Sergio Grinstein, Sandra L Schmid, and Gaudenz Danuser. Robust single-particle tracking in live-cell time-lapse sequences. *Nature methods*, 5(8):695–702, 2008.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME - Journal of Basic Engineering*, 1960.

Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2 (1-2):83–97, 1955.

Timo Kuhn, Johannes Hettich, Rubina Davtyan, and J. Christof M. Gebhardt. Single molecule tracking and analysis framework including theory-predicted parameter settings. *Scientific Reports*, 11(1):9465, 2021. ISSN 2045-2322.

Liang Liang, Hongying Shen, Pietro De Camilli, and James S Duncan. A novel multiple hypothesis based particle tracking method for clathrin mediated endocytosis analysis using fluorescence microscopy. *IEEE transactions on image processing*, 23(4):1844–1857, 2014.

Thomas Mangeat, Simon Labouesse, Marc Allain, Awoke Negash, Emmanuel Martin, Aude Guénolé, Renaud Poincloux, Claire Estibal, Anaïs Bouissou, Sylvain Cantaloube, Elodie Vega, Tong Li, Christian Rouvière, Sophie Allart, Debora Keller, Valentin Debarnot, Xia Bo Wang, Grégoire Michaux, Mathieu Pinot, Roland Le Borgne, Sylvie Tournier, Magali Suzanne, Jérome Idier, and Anne Sentenac. Super-resolved live-cell imaging using random illumination microscopy. *Cell Reports Methods*, 1(1):100009, 2021. ISSN 2667-2375.

Martin Maka, Vladimír Ulman, Pablo Delgado-Rodriguez, Estibaliz Gómez-de Mariscal, Tereza Neasová, Fidel A. Guerrero Peña, Tsang Ing Ren, Elliot M. Meyerowitz, Tim Scherr, Katharina Löffler, Ralf Mikut, Tianqi Guo, Yin Wang, Jan P. Allebach, Rina Bao, Noor M. Al-Shakarji, Gani Rahmon, Imad Eddine Toubal, Kannappan Palaniappan, Filip Lux, Petr Matula, Ko Sugawara, Klas E. G. Magnusson, Layton Aho, Andrew R. Cohen, Assaf Arbelle, Tal Ben-Haim, Tammy Riklin Raviv, Fabian Isensee, Paul F. Jäger, Klaus H. Maier-Hein, Yanming Zhu, Cristina Ederra, Ainhoa Urbiola, Erik Meijering, Alexandre Cunha, Arrate Muñoz-Barrutia, Michal Kozubek, and Carlos Ortiz-de Solórzano. The Cell Tracking Challenge: 10 years of objective benchmarking. *Nature Methods*, 20(7):1010–1020, 2023. ISSN 1548-7105.

Marcel Mettlen and Gaudenz Danuser. Imaging and Modeling the Dynamics of Clathrin-Mediated Endocytosis. *Cold Spring Harbor Perspectives in Biology*, pp. a017038, 2014. ISSN , 1943-0264.

Piyush Mishra and Philippe Roudot. Comparative study of transformer robustness for multiple particle tracking without clutter. In *2024 32nd European Signal Processing Conference (EUSIPCO)*, pp. 571–575. IEEE, 2024.

E. Mudry, K. Belkebir, J. Girard, J. Savatier, E. Le Moal, C. Nicoletti, M. Allain, and A. Sentenac. Structured illumination microscopy using unknown speckle patterns. *Nature Photonics*, 6(5):312–315, 2012. ISSN 1749-4885, 1749-4893.

Juliano Pinto, Georg Hess, Yuxuan Xia, Henk Wymeersch, and Lennart Svensson. Transformer-based multi-object smoothing with decoupled data association and smoothing. *arXiv preprint arXiv:2312.17261*, 2023.

Abu Sajana Rahmathullah, Ángel F García-Fernández, and Lennart Svensson. Generalized optimal sub-pattern assignment metric. In *2017 20th International Conference on Information Fusion (Fusion)*, pp. 1–8. IEEE, 2017.

Donald Reid. An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control*, 24(6): 843–854, 1979.

C. Ritter, J. Y. Lee, M. T. Pham, M. K. Pabba, M. C. Cardoso, R. Bartenschlager, and K. Rohr. Multi-detector fusion and Bayesian smoothing for tracking viral and chromatin structures. *Medical Image Analysis*, 97:103227, 2024. ISSN 1361-8415.

Philippe Roudot, Liya Ding, Khuloud Jaqaman, Charles Kervrann, and Gaudenz Danuser. Piecewise-stationary motion modeling and iterative smoothing to track heterogeneous particle motions in dense environments. *IEEE Transactions on Image Processing*, 26(11):5395–5410, 2017.

Philippe Roudot, Wesley R Legant, Qiongjing Zou, Kevin M Dean, Tadamoto Isogai, Erik S Welf, Ana F David, Daniel W Gerlich, Reto Fiolka, Eric Betzig, et al. u-track3d: Measuring, navigating, and validating dense particle trajectories in three dimensions. *Cell Reports Methods*, 3(12), 2023.

Ihor Smal and Erik Meijering. Quantitative comparison of multiframe data association techniques for particle tracking in time-lapse fluorescence microscopy. *Medical image analysis*, 24(1):163–189, 2015.

Ihor Smal, Marco Loog, Wiro Niessen, and Erik Meijering. Quantitative comparison of spot detection methods in fluorescence microscopy. *IEEE transactions on medical imaging*, 29(2):282–301, 2009.

Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pp. 464–472. IEEE, 2017.

Roman Spilger, Andrea Imle, Ji-Young Lee, Barbara Mueller, Oliver T Fackler, Ralf Bartenschlager, and Karl Rohr. A recurrent neural network for particle tracking in microscopy images using future information, track hypotheses, and multiple detections. *IEEE Transactions on Image Processing*, 29:3681–3694, 2020.

Roman Spilger, Ji-Young Lee, Vadim O Chagin, Lothar Schermelleh, M Cristina Cardoso, Ralf Bartenschlager, and Karl Rohr. Deep probabilistic tracking of particles in fluorescence microscopy images. *Medical image analysis*, 72:102128, 2021.

David J. Stephens and Victoria J. Allan. Light Microscopy Techniques for Live Cell Imaging. *Science*, 300 (5616):82–86, 2003. ISSN 0036-8075, 1095-9203.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Cédric Vonesch, François Aguet, J-L Vonesch, and Michael Unser. The colored revolution of bioimaging. *Signal processing magazine, IEEE*, 23(3):20–31, 2006.

Michael A. Welte. Bidirectional transport along microtubules. *Current Biology*, 14(13):R525–R537, 2004.

Yudong Zhang and Ge Yang. A motion transformer for single particle tracking in fluorescence microscopy images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 503–513. Springer, 2023.

A Appendix

A.1 Prerequisites

A.1.1 Self-attention learns the relationship between each element of the input set

In this section we describe the theory behind self-attention, introduced by Vaswani et al. (2017). Consider $\mathbf{E} \in \mathbb{R}^{n \times h}$ to be the input data structure for some hidden dimension h. $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{n \times d_k}$ and $\mathbf{V} \in \mathbb{R}^{n \times d_v}$ are projections of $\mathbf{E}$ in different learned subspaces such that:

$$\begin{cases} \mathbf{Q} = \mathbf{E}.\mathbf{W}_q \\ \mathbf{K} = \mathbf{E}.\mathbf{W}_k \\ \mathbf{V} = \mathbf{E}.\mathbf{W}_v \end{cases} \tag{27}$$

where $\mathbf{W}_q, \mathbf{W}_k \in \mathbb{R}^{h \times d_k}$ and $\mathbf{W}_v \in \mathbb{R}^{h \times d_v}$ are learned weights. Two of these projections $\mathbf{Q}$ and $\mathbf{K}$ are used to calculate a score between each element of $\mathbf{E}$ and yet another projection $\mathbf{V}$ is used to carry the information from the input that would get mixed based on the aforementioned score. This is done by taking the softmax of the scaled-dot-product between $\mathbf{Q}$ and $\mathbf{K}^T$ ($\in \mathbb{R}^{n \times n}$) and multiplying it with $\mathbf{V}$, such as:

$$\text{attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}.\mathbf{K}^T}{\sqrt{d_k}}).\mathbf{V} \in \mathbb{R}^{n \times d_v} \tag{28}$$

In multi-head attention, we concatenate multiple attention heads before applying a final projection. Consider $n_h = \frac{h}{d_v} \in \mathbb{N}$ attention heads (which requires h to be divisible by d_v, and hence n_h), and the input projections of the j^{th} attention-head to be $\mathbf{Q}_j$, $\mathbf{K}_j$ and $\mathbf{V}_j$. Hence the output of this multi-head attentions is:

$$MHA = \text{concat}(\text{attention}(\mathbf{Q}_j, \mathbf{K}_j, \mathbf{V}_j), \forall j \in [1, n_h]) \in \mathbb{R}^{n \times h} \tag{29}$$

A.1.2 Transformer encoder is a neural network that uses self-attention

Suppose that the transformations undergone by the input $\mathbf{E}$ in the multi-head self-attention layer lead to an output $MHA(\mathbf{E}) \in \mathbb{R}^{n \times h}$. Subsequently, the following transformations take place in an encoder layer,

$$\mathbf{RO} = \mathbf{E} + MHA \in \mathbb{R}^{n \times h} \tag{30}$$

$$\mathbf{NO} = \text{LayerNorm}(\mathbf{RO}) \in \mathbb{R}^{n \times h} \tag{31}$$

$$\mathbf{FO} = \phi(\mathbf{NO}.\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \in \mathbb{R}^{n \times h} \tag{32}$$

where a non-linearity activation function ϕ is applied between the layers; $\mathbf{W}_1 \in \mathbb{R}^{h \times d_{ffn}}$, $\mathbf{W}_2 \in \mathbb{R}^{d_{ffn} \times h}$, $\mathbf{b}_1 \in \mathbb{R}^{n \times d_{ffn}}$ and $\mathbf{b}_2 \in \mathbb{R}^{n \times h}$. In eq. 31, layer normalization (Ba et al. (2016)) helps stabilize training.

$$\mathbf{RO}_{ffn} = \mathbf{NO} + \mathbf{FO} \in \mathbb{R}^{n \times h} \tag{33}$$

$$\text{Final output} = \text{LayerNorm}(\mathbf{RO}_{ffn}) \in \mathbb{R}^{n \times h} \tag{34}$$

If we combine all the transformations in an encoder layer to call it $\mathscr{E}$, the output of one encoder layer is $\mathscr{E}(\mathbf{E})$. In an encoder, there are multiple such encoder layers stacked one after the other. So, for an encoder with n_l encoder layers, the output can be called $\mathscr{E}_{n_l}(\mathbf{E}) \in \mathbb{R}^{n \times h}$.

A.2 Derivation of our process noise covariance matrix in Kalman filtering

For continuous time in one dimension, we can describe our state as a function of time as:

$$\mathbf{x}(t) = \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \tag{35}$$

The state transition equation can be written as:

$$\frac{d}{dt}\begin{bmatrix}x(t)\\v(t)\end{bmatrix} = \begin{bmatrix}0 & 1\\0 & 0\end{bmatrix}\begin{bmatrix}x(t)\\v(t)\end{bmatrix} + \begin{bmatrix}0\\1\end{bmatrix}a(t) \tag{36}$$

where $a(t)$ is the process noise that arises from unknown random acceleration. Let $a(t)$ be a zero-mean white noise process so $a(t)$ and $a(t')$ for different time-points t and t' are uncorrelated. Hence, noise covariance can be modeled using the Dirac-delta function:

$$\mathbb{E}[a(t)a(t')] = \sigma_q^2\delta(t-t') \tag{37}$$

where

$$\delta(t-t') = \begin{cases}\infty, & t = t'\\ 0, & \text{else}\end{cases} \tag{38}$$

So, when $t \neq t'$, noise autocorrelation is zero, meaning process noise at two different time steps is not correlated. This affects velocity and position differently

$$v(t) = v(0) + \int_0^t a(\tau)d\tau \tag{39}$$

$$x(t) = x(0) + v(0)t + \int_0^t\int_0^s a(\tau)d\tau ds \tag{40}$$

As such, the velocity-velocity covariance is determined as,

$$\mathbb{E}[v(t)v(t')] = \int_0^t\int_0^{t'}\mathbb{E}[a(\tau)a(\tau')]d\tau' d\tau \tag{41}$$

$$\mathbb{E}[v(t)v(t')] = \sigma_q^2\int_0^t\int_0^{t'}\delta(\tau-\tau')d\tau' d\tau \tag{42}$$

Now,

$$\int_0^{t'}\delta(\tau-\tau')d\tau' = \begin{cases}1, & \tau\in[0,t']\\ 0, & \text{else}\end{cases} = h(t'-\tau)\ \text{(let)} \tag{43}$$

So,

$$\mathbb{E}[v(t)v(t')] = \sigma_q^2\int_0^t h(t'-\tau)d\tau \tag{44}$$

$$\mathbb{E}[v(t)v(t')] = \sigma_q^2 min(t,t') \tag{45}$$

at $t = t'$,

$$\mathbb{E}[v^2(t)] = \sigma_q^2 t \tag{46}$$

Hence, the velocity-velocity covariance is affected such that a linear increase in time increases the covariance linearly. Similarly, position-velocity covariance has a quadratic dependence on time,

$$\mathbb{E}[x(t)v(t)] = \frac{\sigma_q^2 t^2}{2} \tag{47}$$

and position-position covariance has a cubic dependence on time.

$$\mathbb{E}[x^2(t)] = \frac{\sigma_q^2 t^3}{3} \tag{48}$$

Table 6: Dictionary of parameters used for performance metrics

d_ϕ	3
c	5
p	2

Consider, for the state $\mathbf{x}(t)$, a process noise covariance matrix denoted by $\mathbf{Q}(t) = \begin{bmatrix} q_{xx}(t) & q_{xv}(t) \\ q_{vx}(t) & q_{vv}(t) \end{bmatrix}$, where q_{xx} is the uncertainty of position with respect to itself, q_{vv} is the uncertainty of velocity with respect to itself, and q_{xv} and q_{vx} are uncertainties of position with respect to velocity and vice versa.

$$\mathbf{Q}(t) = \sigma_q^2 \begin{bmatrix} \frac{t^3}{3} & \frac{t^2}{2} \\ \frac{t^2}{2} & t \end{bmatrix} \tag{49}$$

Hence, for a discrete time sampling interval dt, the process noise matrix in 2-dimensions becomes,

$$\mathbf{Q} = \sigma_q^2 \begin{bmatrix} \frac{dt^3}{3} & \frac{dt^2}{2} & 0 & 0 \\ \frac{dt^2}{2} & dt & 0 & 0 \\ 0 & 0 & \frac{dt^3}{3} & \frac{dt^2}{2} \\ 0 & 0 & \frac{dt^2}{2} & dt \end{bmatrix} \tag{50}$$

Thus, we obtain the process noise covariance matrix used in our model.

A.3 Generalized Optimal Sub-Pattern Assignment

The GOSPA metric evaluates multi-object tracking accuracy at a single time step, incorporating both localization and cardinality errors. The GOSPA distance of order $p \geq 1$ with cutoff $c > 0$ is defined as,

$$d_p^c(\mathcal{X}_t, \hat{\mathcal{X}}_t) = \left(\min_{\pi \in \Pi} \sum_{i=1}^{k} \min \left(\|\chi_t^i - \hat{\chi}_t^{\pi(i)}\|^p, c^p \right) + \frac{c^p}{2} (|\mathcal{X}_t| + |\hat{\mathcal{X}}_t| - 2k) \right)^{1/p} \tag{51}$$

where, Π is the set of all injective assignments between the smaller of the two sets. Moreover, $\pi(i) = j$ means that the i^{th} index of $\mathcal{X}$ has matched with the j^{th} index of $\hat{\mathcal{X}}$. Further, $k = \min(|\mathcal{X}|, |\hat{\mathcal{X}}|)$. The first term measures the localization error between the matched points, while the second penalizes the unmatched ground-truth or predicted detections.

A.4 MHT performance approaches ground truth tracks only under specific permissive gating radii

In order to keep the comparison fair, we parameterize MHT only once. This is done such that in the case of low-density and no-noise, the predicted tracks would be as close to the ground truth tracks as possible. We demonstrate this procedure in Fig. 10 and 11. A gating threshold of 10 is hence chosen for all inferences that use this implementation of MHT.

A.5 Quantitative comparison with Trackastra baseline for virus trafficking

Here we present a quantitative comparison of our proposed approach with a pretrained version of Trackastra in the general 2D mode. A constraint is that Trackastra does not work if any frame has 0 detections. So just for Trackastra we remove the notion of false negatives and force one random detection for all frames containing zero detections. Note that these false negatives still exist for ABHA. Further, here we compare a pretrained model of Trackastra with a trained model of ABHA. In this regard, we see this comparison as not as fair as the comparison with classical MHT baseline. However, this comparison still provides meaningful insights.

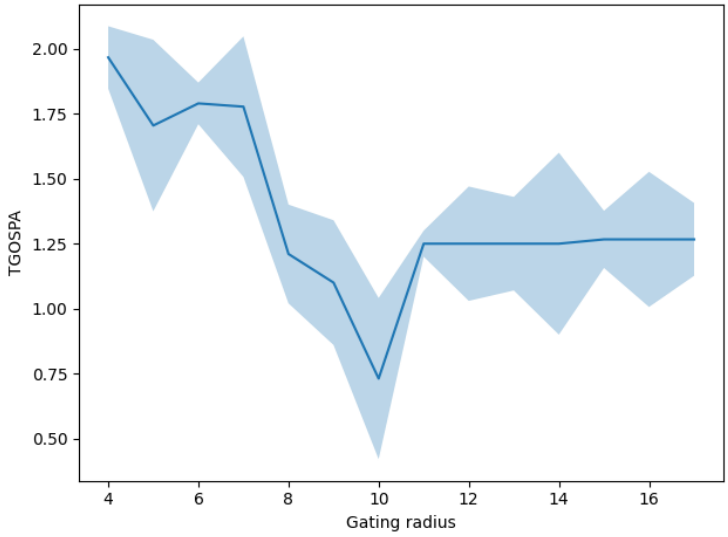

Figure 10: A gating radius of 10 achieves the lowest TGOSPA score for a set of 5 low-density-no-noise scenarios. The plot also shows a 95% confidence band.

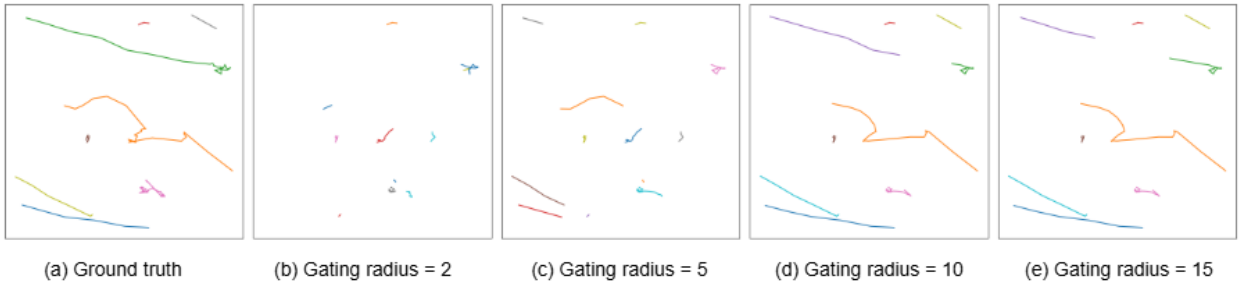

Figure 11: Relatively large gating radii for MHT result in tracks that are the closest to the ground truth tracks. A gating radius of 10 is chosen for all implementations.

Table 7: Comparison of ABHA and Trackastra on a patch of Viruses dataset for ($SNR = 4$) and medium density.

Task	**Method**	R_P	P_P	JSC_P	P_L	R_L	$F1_L$	JSC_L	TGOSPA
ϕ	ABHA	0.974	0.974	0.948	0.957	0.908	0.932	0.466	1.294 ± 0.114
	Trackastra	0.943	0.859	0.735	0.869	0.920	0.893	0.445	2.337 ± 0.132
A	ABHA	0.912	0.999	0.853	0.990	0.789	0.896	0.448	2.174 ± 0.017
	Trackastra	0.979	0.876	0.776	0.855	0.919	0.886	0.443	2.482 ± 0.225
B	ABHA	0.907	0.994	0.823	0.772	0.643	0.832	0.411	2.222 ± 0.103
	Trackastra	0.954	0.864	0.754	0.872	0.942	0.906	0.453	2.785 ± 0.111
C	ABHA	0.872	0.988	0.994	0.983	0.754	0.943	0.564	1.763 ± 0.011
	Trackastra	0.979	0.884	0.784	0.866	0.931	0.897	0.448	2.824 ± 0.121
1	ABHA	0.914	0.988	0.876	0.943	0.677	0.877	0.534	2.111 ± 0.006
	Trackastra	0.984	0.885	0.768	0.864	0.919	0.891	0.445	2.854 ± 0.243
2	ABHA	0.923	0.989	0.865	0.976	0.705	0.869	0.539	2.612 ± 0.041
	Trackastra	0.974	0.855	0.710	0.837	0.920	0.866	0.438	3.424 ± 0.244
3	ABHA	0.884	0.935	0.823	0.882	0.564	0.773	0.530	2.313 ± 0.088
	Trackastra	0.989	0.835	0.699	0.568	0.578	0.573	0.286	4.549 ± 0.143
4	ABHA	0.845	0.902	0.811	0.834	0.561	0.784	0.511	2.116 ± 0.094
	Trackastra	0.985	0.841	0.705	0.573	0.589	0.581	0.291	4.416 ± 0.123

Table 7 shows that ABHA consistently achieves higher point-based precision (P_P) and Jaccard similarity (JSC_P), indicating more accurate detection and tracking of individual particles. However, as with MHT, ABHA exhibits slightly lower recall compared to Trackastra, suggesting that the method occasionally misses detections, likely due to its conservative hypothesis pruning in highly cluttered scenarios. Despite this, ABHA demonstrates robust link-based performance ($F1_L$, JSC_L) in noisy and dense tasks (2, 3, 4), maintaining correct trajectory associations where Trackastra suffers from identity switches or missing links. The TGOSPA metric further reflects lower values for ABHA.