Towards Physics-informed Diffusion for Anomaly Detection in Trajectories: A Summary of Results

Arun Sharma
University of Minnesota
Minneapolis, USA
arunshar@umn.edu

Mingzhou Yang
University of Minnesota
Minneapolis, USA
yang7492@umn.edu

Majid Farhadloo
University of Minnesota
Minneapolis, USA
farha043@umn.edu

Subhankar Ghosh
University of Minnesota
Minneapolis, USA
ghosh117@umn.edu

Bharat Jayaprakash
University of Minnesota
Minneapolis, USA
jayap015@umn.edu

Shashi Shekhar
University of Minnesota
Minneapolis, USA
shekhar@umn.edu

Abstract

Given a dataset of moving object trajectories, a domain-specific study area, and a user-defined error threshold, we aim to identify anomalous trajectories indicative of possible GPS spoofing (e.g., broadcasting fake signals). The problem is societally important to curb illegal activities such as unauthorized fishing and illicit oil transfers in international waters. The problem is challenging due to advances in AI-generated deep fakes (e.g., additive noise, fake trajectories) and the scarcity of labeled samples for ground-truth verification. Current state-of-the-art methods ignore fine-scale spatiotemporal dependencies and prior physical knowledge, resulting in lower accuracy. In this paper, we propose a physics-informed anomaly detection framework based on an encoder-decoder architecture that incorporates kinematic constraints to identify trajectories that violate physical laws. Experimental results on maritime and urban domains demonstrate that the proposed approach yields higher solution quality and lower estimation error for anomaly detection and trajectory reconstruction tasks, respectively.

CCS Concepts

• **Physics-informed Neural Network**; • **Trajectory Reconstruction**; • **Spatiotemporal Data Mining**; • **Anomaly Detection**;

Keywords

Physics-informed Neural Network, Trajectory Reconstruction, Spatiotemporal Data Mining, Anomaly Detection

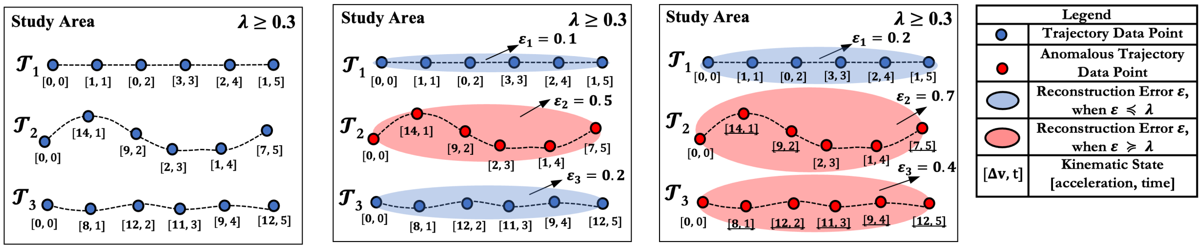
ACM Reference Format:
Arun Sharma, Mingzhou Yang, Majid Farhadloo, Subhankar Ghosh, Bharat Jayaprakash, and Shashi Shekhar. 2025. Towards Physics-informed Diffusion for Anomaly Detection in Trajectories: A Summary of Results. In *The 2nd ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection (GeoAnomalies '25), November 3–6, 2025, Minneapolis, MN, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3764914.3770595

GeoAnomalies '25, November 3–6, 2025, Minneapolis, MN, USA
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2260-8/2025/11.
https://doi.org/10.1145/3764914.3770595

1 Introduction

Given a moving object trajectory dataset, a domain-specific study area (e.g., road network, maritime waters), and a user-defined reconstruction threshold (λ), our aim is to identify anomalous trajectories indicative of possible GPS spoofing (e.g., broadcasting fake trajectory signals). A ***moving object*** is defined as an entity that displays realistic motion characteristics in urban (such as road networks) and maritime mobility datasets. ***GPS spoofing*** is a type of anomalous behavior that occurs when an object intentionally broadcasts spurious or fake locations to conceal its movements for potential illicit activities and deceive the end-user. These false signals do not conform to the historical mobility patterns of nearby entities and they are not physically feasible, resulting in higher ***reconstruction errors*** ε (i.e., divergence between the generated and original input trajectories) as compared to the normal pattern (e.g., the shortest or most frequent path). For instance, Figure 1 (a) shows trajectories $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$ where each point within a trajectory $\mathcal{T}_i$ has a ***kinematic state*** **[Δv, t]** where Δv denotes acceleration (or change in velocity v) and t as timestamp (where $t \in [0, .., 5]$). Figure 1 (b) shows that trajectory $\mathcal{T}_2$ exhibits significantly different mobility behavior, followed by $\mathcal{T}_3$ and $\mathcal{T}_1$ in terms of their spatial deviation from the shortest path. Hence, $\mathcal{T}_2$'s behavior is flagged as anomalous with a reconstruction error of ε_2, exceeding the threshold of **0.3**. In contrast, given the same threshold, Figure 1 (c) shows $\mathcal{T}_2$ and $\mathcal{T}_3$ as anomalous because we are considered their kinematical state [Δv, t] based additional physical attributes (i.e., acceleration or change in velocity) at each timestamp t along with spatial-deviation from the shortest path, resulting in more accurate reconstruction error $\varepsilon_2 = 0.7$ and $\varepsilon_2 = 0.4$ respectively. In this paper, we investigate a physics-informed anomaly detection framework based on a physics-informed diffusion probabilistic model (Pi-DPM), which effectively analyzes spatiotemporal and physical characteristics that violate laws of physics.

GPS spoofing has significant implications for homeland security, public health and safety, enforcement of UN sanctions, and maritime regulations. An example of anomalous activity that put the public at risk occurred in 2020, when a Berlin artist used 99 phones to manipulate Google Maps and reroute traffic from a residential street [30]. In another case [71], an oil tanker named Cathay Phoenix was reported to be broadcasting fake location signals near the Sea of Japan, suggesting it was making frequent abrupt changes in direction (or bearing) and moving in an anticlockwise pattern.



(a) **Input:** Trajectories $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$ with **kinematic state** $[\Delta \mathbf{v}, \mathbf{t}]$ where $\lambda \geq 0.3$.

(b) **Related Work Output:** Anomalous Trajectory $\mathcal{T}_2$ when only considering the **spatial deviation** (shortest-path).

(c) **Proposed Work Output:** Anomalous Trajectories $\mathcal{T}_2$ and $\mathcal{T}_3$ when considering **both spatial deviation** and **kinematic state** $[\Delta \mathbf{v}, \mathbf{t}]$.

Figure 1: An illustration example of a problem statement with input and output comparison for both related work and the proposed approach (Best in Color).

Typically, the bearing change distribution for tankers follows a straight-line path with almost no variation (i.e., the green box). In contrast, the Cathay Phoenix exhibited sudden directional shifts and a high rate of turns, as shown by its bearing change distribution (i.e., the red box). Such a high rate of turns is not physically plausible for a vessel as large as an oil tanker to execute within a short period of time in a confined geographic region [71]. It was later revealed by nearby cargo vessels that the ship's actual location at the time was a Russian port, where it was loading oil for shipment to China to evade U.S. sanctions.

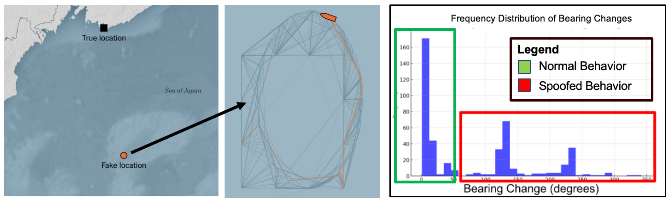


Figure 2: An oil tanker, Cathay Phoenix, whose bearing distribution (red box) is significantly different from normal behavior (green box) (Best in Color).

Challenges: Identifying anomalous trajectories arising from possible GPS spoofing is challenging due to advances in AI-generated deep fakes (e.g., additive noise, fake trajectories). Moreover, the scarcity of ground-truth data on known GPS spoofing behavior hinders the development of accurate and physically plausible anomaly detection methods. Finally, machine learning models provide statistical estimations and fail to integrate real physical constraints when generating trajectories. This limits their interpretability and explainability, making it hard for human analysts to manually inspect them for illegal activities [71].

Related Work: Traditional literature on anomaly detection methods for trajectories leverages spatial and temporal feature representations for classifying anomalous trajectories. However, these methods are limited to capturing anomalous trajectories at a coarser level for a given time interval (e.g., aggregated by asynchronous routes or trips) [9, 85], which restricts their ability to learn real-time spatiotemporal dependencies at a finer granularity and to investigate complex behavioral mobility patterns. Current data-driven approaches [22, 28, 43, 48, 52, 74] do consider such synchronous spatial or temporal characteristics while capturing trajectory anomalies. However, none of them account for physics-based parameters (e.g., rate of turn, acceleration), making it challenging to investigate real-time mobility dynamics, which are crucial for detecting rare but significant anomalous behaviors such as the GPS spoofing [71]. In this work, we propose a diffusion-based probabilistic model that incorporates real-time spatiotemporal dependencies and physics-based parameters [56] to enforce kinematic constraints at a finer granularity. This ensures physically feasible behaviors.

Our contributions are as follows:

- We introduce a physics-informed anomaly detection framework that builds upon an encoder-decoder architecture for more accurate error reconstruction estimation, followed by threshold-based anomaly detection.
- We propose a physics-informed diffusion probabilistic model (Pi-DPM) as an encoder-decoder architecture that leverages fine-scale spatiotemporal dependencies with physical constraints to more accurately estimate maneuvers and complex motion dynamics in the sampling phase.
- Thorough extensive experiments on real-world trajectory datasets, we demonstrate that Pi-DPM outperforms state-of-the-art methods in both trajectory reconstruction and anomaly detection.

GeoAnomalies Workshop Relevance: This work is relevant to the GeoAnomalies workshop for the following reasons:

- It builds on recent advances in geo-anomaly detection, including synthetic mobility datasets with injected anomalies [2, 70], context-aware detection [34], and LLM-based trajectory pattern mining [88].
- While these approaches improve robustness and consider contextual information (e.g., POIs), they often overlook enforcement of physics at a finer granularity [36].
- The proposed framework aligns with the workshop's focus on geo-spatial anomaly detection using generative models.

Scope: This paper only considers one type of geometric spoofing behavior (i.e., circular based on [71]). It does not address other spoofing behavior prevalent in maritime and urban settings (e.g., record and reply, false AIS identification, etc.). The paper also does not consider spatial and temporal attention separately to identify which one plays a more significant role in the proposed work. Finally, the paper does not consider real-world incidents of the spoofing [71], which remains a direction for future work.

Organization: The rest of the paper is organized as follows. Section 2 introduces key concepts and formally defines the problem. Section 3 describes the proposed model in detail. In Section 4, we evaluate the proposed model experimentally based on anomaly detection, trajectory reconstruction, and sensitivity analysis. A detailed review of related work is given in Section 5. Finally, Section 6 concludes the paper and outlines future work.

2 Problem Formulation

2.1 Notations and Definitions

Definition 2.1. A **trajectory** $\mathcal{T}$ is a collection of location traces generated by a moving object (e.g., vehicle, ship), typically represented as a chronologically ordered sequence of points where each point consists of object's coordinates $(lat, long)$ and its kinematic state at time t. For instance, in Figure 1, the trajectories $\mathcal{T}_1, \mathcal{T}_2$, and $\mathcal{T}_3$ consist of ordered location traces and collection of such trajectories forms a trajectory dataset: $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_N$ where N is the number of temporal observations in the trajectory.

Definition 2.2. The **kinematic state** of each point can be derived via computing travel-time, speed, angle, from consecutive points $\in [lon_t, lat_t, v_t, \delta_t, a_t, \dot{\delta}_t]^T$, where v_t is velocity, a_t is acceleration (equivalent to Δv in Figure 1), δ_t is bearing and $\dot{\delta}$ as rate of turn.

Definition 2.3. An **anomalous trajectory** deviates from typical patterns in aspects such as position, orientation, and speed. On roadways, such deviations often arise from violations of traffic rules or physical driving constraints, such as sudden or extreme changes in movement due to complex or unpredictable conditions. In maritime settings, common examples include drifting off, prolonged stops, or sudden changes in a vessel's bearing. Due to data scarcity, this paper synthetically generated anomalies by injecting normal trajectories with noise characteristics, including sudden acceleration, long-term stops, frequent lane shifting, side-to-side swinging, reverse driving, sudden stops, deviations beyond road boundaries, and violent turning.

Definition 2.4. A **reconstruction error** ε is the error margin derived from comparing the divergence between the generated and original input trajectories using any generative model (such as a diffusion model). The reconstruction loss used during training captures the cumulative squared error across all trajectory points:

$$\mathcal{L}_{\text{Rec}} = \sum_{t=1}^{T} \left\| \begin{pmatrix} \hat{lon}_t \\ \hat{lat}_t \end{pmatrix} - \begin{pmatrix} lon_t \\ lat_t \end{pmatrix} \right\|^2 \tag{1}$$

In Figure 1 (b) and (c), the reconstruction errors of $\mathcal{T}_3$ are 0.2 and 0.4, respectively. In addition, $\mathcal{T}_3$ is labeled as **anomalous** if its reconstruction error is $\geq \lambda$ (i.e., 0.3), indicating notable deviations in latitude, longitude, speed, or heading from neighboring trajectories where λ is defined as the **reconstruction threshold**.

Definition 2.5. A **diffusion probabilistic model (DPM)** is a generative framework that learns to reverse a noise-adding diffusion process, typically modeled with Gaussian perturbations, so that samples drawn from the reverse Markov chain approximate the actual data distribution.

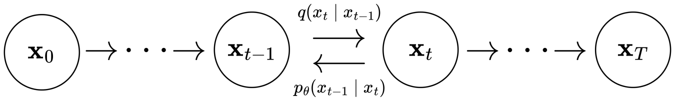


Figure 3: Forward and reverse diffusion processes

Figure 3 presents a detailed view of a diffusion model, where in the forward process, isotropic Gaussian noise is progressively added to a signal through a predetermined Markov chain, denoted by $q(\mathbf{x}_t \mid \mathbf{x}_{t-1})$, generating a series of intermediate noise variances referred as *forward* diffusion. Conversely, in *reverse* diffusion, the process begins with a complete random noise, $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The model then incrementally refine through successive iterations $(\mathbf{x}_{T-1}, \mathbf{x}_{T-2}, \ldots, \mathbf{x}_0)$ using **learned transition distributions** $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$, aiming for $\mathbf{x}_0 \sim p(\mathbf{x})$ and reconstructing the signal from noise using a backward Markov chain.

Definition 2.6. A **forward process** progressively introduces (q) Gaussian noise to $\mathbf{x}_0$ over T iterations [32] as described by:

$$q(\mathbf{x}_{1:T} \mid \mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \,, \tag{2}$$

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t \mid \sqrt{\alpha_t}\, \mathbf{x}_{t-1}, (1-\alpha_t)\mathbf{I}) \,, \tag{3}$$

Here, the hyper-parameters $\alpha_{1:T}$ are constrained between 0 and 1, representing the noise variance introduced at each iteration. The variable $\mathbf{x}_{t-1}$ is scaled down by a factor of $\sqrt{\alpha_t}$ to keep the variance of the random variables finite. Additionally, deriving $\mathbf{x}_t$ from $\mathbf{x}_0$ can be streamlined using Equation 4.:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) \;=\; \mathcal{N}(\mathbf{x}_t \mid \sqrt{\bar{\alpha}_t}\, \mathbf{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}) \,, \tag{4}$$

where $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. In addition, one can derive the posterior distribution of $\mathbf{x}_{t-1}$ given $(\mathbf{x}_0, \mathbf{x}_t)$ as

$$\begin{aligned} q(\mathbf{x}_{t-1} \mid \mathbf{x}_0, \mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1} \mid \boldsymbol{\mu}, \sigma^2 \boldsymbol{I}) \\ \boldsymbol{\mu} &= \frac{\sqrt{\bar{\alpha}_{t-1}}\,(1-\alpha_t)}{1-\bar{\alpha}_t}\,\mathbf{x}_0 + \frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\,\mathbf{x}_t \\ \sigma^2 &= \frac{(1-\bar{\alpha}_{t-1})(1-\alpha_t)}{1-\bar{\alpha}_t} \,. \end{aligned} \tag{5}$$

Definition 2.7. A **reverse process** (p_θ) is the *reverse* of a forward diffusion process in that it begins with a completely noisy state, $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. The model then incrementally refines the trajectories through successive iterations $(\mathbf{x}_{T-1}, \mathbf{x}_{T-2}, \ldots, \mathbf{x}_0)$. In contrast to the forward process q, p_θ goes in the reverse direction starting from Gaussian noise $\boldsymbol{x}_T$:

$$p_\theta(\mathbf{x}_{0:T}) \;=\; p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \tag{6}$$

$$p(\mathbf{x}_T) \;=\; \mathcal{N}(\mathbf{x}_T \mid \mathbf{0}, \boldsymbol{I}), \tag{7}$$

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \;=\; \mathcal{N}(\mathbf{x}_{t-1} \mid \mu_\theta(\mathbf{x}_t, \alpha_t), \sigma_t^2 \boldsymbol{I}) \,, \tag{8}$$

where $p_\theta(\mathbf{x}_{0:T})$ is the joint probability distribution of $(\mathbf{x}_0, \ldots, \mathbf{x}_T)$ and $p(\mathbf{x}_T)$ is a Gaussian distribution. The mean $\mu_\theta(\mathbf{x}_t, t)$ accepts $\mathbf{x}_t$

and t as input and is obtained by learning while the variance term $\sigma_t^2 \boldsymbol{I}$ follows the inference process defined using isotropic Gaussian distributions $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$, which are learned. If the noise variance in the forward process steps is minimized, *i.e.*, $\alpha_{1:T} \approx 1$, the resulting optimal reverse process in Equation 8, $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ will closely approximate a Gaussian distribution [68]. Moreover, it is necessary for $1 - \alpha_T$ to be large enough to ensure that the distribution of $\mathbf{x}_T$ closely aligns with the prior distribution $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T|\mathbf{0}, \mathbf{I})$, which is a standard Gaussian with mean zero and identity covariance matrix. Here we designed f_θ to predict $\boldsymbol{\epsilon}$, including x_t. Consequently, we can estimate $\mathbf{x}_0$ by rearranging terms as shown in Equation 9:

$$\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\, f_\theta(\mathbf{x}_t, \bar{\alpha}_t) \right) . \tag{9}$$

Finally, we insert $\hat{\mathbf{x}}_0$ into the posterior distribution of $q(\mathbf{x}_{t-1}|\mathbf{x}_0, \mathbf{x}_t)$, which parameterizes the mean of $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ in Equation 10. We set the variance of $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ to $(1 - \alpha_t)$, which is the default variance as determined by the variance of the forward process [33]:

$$\mu_\theta(\mathbf{x}_t, \bar{\alpha}_t) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} f_\theta(\mathbf{x}_t, \bar{\alpha}_t) \right) . \tag{10}$$

Additionally, we carry out refinement at each iteration as described in Equation 11 (where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, 1)$):

$$\mathbf{x}_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} f_\theta(\mathbf{x}_t, \bar{\alpha}_t) \right) + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t . \tag{11}$$

For consistency, we provide a summary of the mathematical symbols employed in this paper. Table 1 consolidates the key notations used in the proposed framework.

Table 1: Summary of Mathematical Notations and Symbols

Notation	Description
p_t	Location sample at time t: $(lon_t, lat_t, v_t, \psi_t, a_t, \delta_t, \dot{\delta}_t)$
$lon_t,\ lat_t$	Longitude and latitude coordinates at time t
$v_t,\ \hat{v}_t$	True / reconstructed speed (m s^{-1})
$a_t,\ \hat{a}_t$	True / reconstructed longitudinal acceleration (m s^{-2})
$\psi_t,\ \hat{\psi}_t$	Heading (yaw) angle / reconstructed heading (rad)
δ_t	Steering (bearing/turning) angle in S-KBM (rad)
$\dot{\delta}_t$	Rate of turn (rad s^{-1})
L	Wheelbase or effective heading-base of the object (m)
$\kappa_t,\ \hat{\kappa}_t$	Curvature, $\kappa_t = \tan(\delta_t)/L$ / reconstructed curvature
R_t	Turning radius, $R_t = L/\tan(\delta_t)$
ε_t	Reconstruction error at time t
λ	Reconstruction-error threshold for anomaly detection
$\mathcal{T}$	A trajectory: ordered sequence of spatiotemporal states
$z = [x, y, \psi]^\top$	Object state in S-KBM (position and heading)
$u = [v, \delta]^\top$	Control input in S-KBM (velocity and steering)
T	Total number of diffusion steps ($T = 1000$ in our experiments)
$q(\mathbf{x}_t\|\mathbf{x}_{t-1})$	Forward diffusion distribution (Gaussian)
$p_\theta(\mathbf{x}_{t-1}\|\mathbf{x}_t)$	Reverse denoising distribution parameterized by θ
α_t	Forward noise–scaling coefficient at step t
$\bar{\alpha}_t$	Cumulative product $\prod_{i=1}^{t} \alpha_i$
β_t	Forward variance increment, $\beta_t = 1 - \alpha_t$
σ_t	Variance in reverse step $p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$
$\mathbf{x}_0,\ \hat{\mathbf{x}}_0$	Original trajectory / reconstructed trajectory
$\boldsymbol{\epsilon},\ \epsilon_t$	Standard Gaussian noise vector / noise sample at step t
f_θ	Neural network predicting $\boldsymbol{\epsilon}$ or μ_θ in reverse process
$\mathbf{h}_{enc}$	Context-informed encoder embedding (spatiotemporal features)
$\mathbb{E}\,[[\,]\cdot],\ \mathbb{E}_q\,[\cdot]$	Expectation operator (unconditional / under distribution q)
$D_{\mathrm{KL}}(p\\|q)$	Kullback–Leibler divergence between distributions p and q
$\mathcal{L}_{\mathrm{VLB}}$	Variational lower-bound loss
$\mathcal{L}_{\mathrm{Rec}}$	Reconstruction loss (Eq. 1)
$\mathcal{L}_{\mathrm{Phy}}$	Physics-based regularization loss (S-KBM constraints, Eq. 20)
$\mathcal{L}^{\text{Pi-DPM}}$	Total training loss: $\gamma_1 \mathcal{L}_{\mathrm{VLB}} + \gamma_2 \mathcal{L}_{\mathrm{Phy}}$

2.2 Problem Statement

Input:

- Trajectory Dataset: $\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_N$
- Kinematic State: $[lon_t, lat_t, v_t, \delta_t, a_t, \dot{\delta}_t]^T$
- Reconstruction Threshold: λ

Output: Anomalous Trajectories where $\varepsilon_i \geq \lambda$
Objective: Solution Quality
Constraints: Data Availability, Ground truth

Figure 1 (a) shows three trajectories $\mathcal{T}_1$, $\mathcal{T}_2$, and $\mathcal{T}_3$, where trajectory $\mathcal{T}_2$ exhibits significantly different mobility behavior compared to $\mathcal{T}_1$ and $\mathcal{T}_3$ if only spatial deviation is considered. Hence, Figure 1 (b) shows $\mathcal{T}_2$ estimate higher reconstruction error ε_2 and exceeding reconstruction threshold $\lambda \geq 0.3$. Figure 1 (c) considered ***both*** spatial deviation and kinematic state $\Delta v\ \forall\ t \in [t_1, .., t_n]$ resulting in higher reconstruction error both exceeding $\lambda \geq 0.3$.

Framework: The proposed framework builds on physics-informed trajectory reconstruction to detect anomalous trajectories by leveraging both contextual and prior physical knowledge. The process begins with trajectory GPS signals, which are used to generate trajectories while computing the corresponding reconstruction error (ε_i). A reconstruction threshold (λ) is applied to these errors to identify candidate anomalous trajectories where $\varepsilon_i \geq \lambda$. These candidates are then examined in conjunction with satellite aerial imagery during an inspection by a human analyst. The analyst's review aids in forming a possible anomaly hypothesis, ensuring that the final detection is both data-driven and contextually validated. In this paper, we utilize a physics-informed diffusion probabilistic model (Pi-DPM) as an encoder-decoder architecture to generate the reconstruction error ε, which is discussed in Section 3.

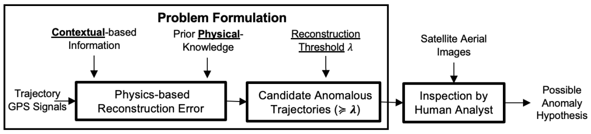

Problem Formulation
Contextual-based Information
Prior Physical-Knowledge
Reconstruction Threshold λ
Satellite Aerial Images
Trajectory GPS Signals
Physics-based Reconstruction Error
Candidate Anomalous Trajectories (⩾ λ)
Inspection by Human Analyst
Possible Anomaly Hypothesis

Figure 4: Physics-informed Anomaly Detection Framework

3 Physics-informed Diffusion Model

Preserving inter-agent dependencies in multi-agent systems is crucial for accurately modeling and detecting abnormal behaviors. To this end, we introduce an encoder-decoder architecture with two main components: a ***context-informed encoder*** to capture spatial-temporal synchronous dependencies while generating embeddings, and a ***physics-informed decoder*** to generate physically plausible trajectories and sample for estimating reconstruction error.

3.1 Proposed Architecture

Our proposed model leverages the original denoising diffusion probabilistic model (DDPM) [32] by considering local spatio-temporal dependencies derived from neighboring trajectories, conditioning the reverse process. The *forward* process gradually corrupts the reconstruction error of the input trajectory by adding Gaussian noise until it converges to complete distortion at $\boldsymbol{x}_T$. In the *reverse* process, the model de-noises $\boldsymbol{x}_T$ step-by-step to approximate the original trajectory $\tilde{x}_0$, where at each denoising step, physics-informed

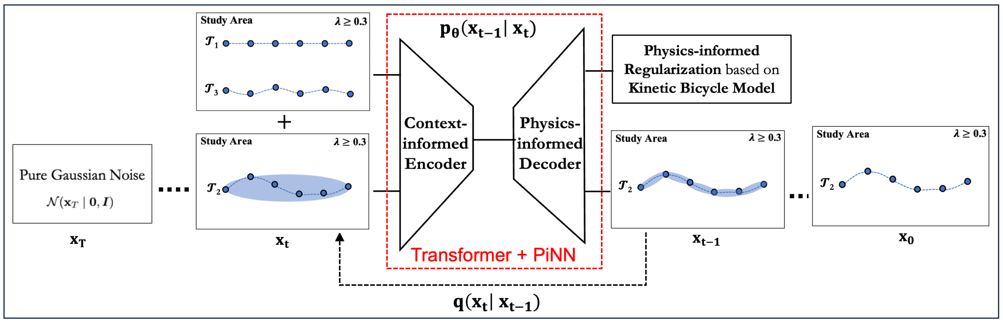
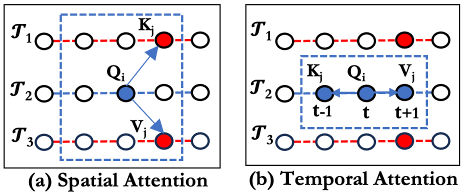


Figure 5: Physics-informed Diffusion Probabilistic Model (Pi-DPM) Architecture

decoder parameters are learned to enforce kinematic constraints, while conditioning on surrounding trajectories to enhance spatio-temporal contextuality. Figure 6 (b) shows an example of a typical transition between states $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$. The forward transition follows $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, while the reverse denoising step is modeled by $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$. At each timestep t, we concatenate the current noisy trajectory $\boldsymbol{x}_t$ with its spatially proximate neighbors and feed them into an encoder-decoder architecture. The encoder leverages spatio-temporal attention, and the decoder outputs a denoised trajectory $\boldsymbol{x}_{t-1}$. Iterating this reverse process ultimately yields a reconstructed trajectory. For training, we adopt a piecewise noise scheduling [31], defined as $p(\bar{\alpha}) = \sum_{t=1}^{T} \frac{1}{T} U(\bar{\alpha}_{t-1}, \bar{\alpha}_t)$.

3.2 Context-informed Encoder

To model contextual dependencies, Pi-DPM embeds object positions and applies spatial-temporal attention [73]. Figure 6 shows how the Contextual-Informed Encoder (CIE) captures spatial and temporal dependencies via neighborhood information.



(a) Spatial Attention (b) Temporal Attention

Figure 6: Context-informed Encoder

(1) Spatial Attention: We use a sliding window at time t, where the target object's trajectory features form a query Q, and its neighbors contribute keys K and values V. Attention weights, which are computed via the softmax-scaled dot products of Q and K, highlight the relevant neighbors. In Figure 6(a), object $\mathcal{T}_2$ attends to nearby objects $\mathcal{T}_1$ and $\mathcal{T}_3$. The multi-head attention formulation is:

$$\mathrm{MH}(Q_t^i, K_t^j, V_t^j) = \left[\sum \mathrm{softmax}\left(\frac{Q_t^i (K_t^j)^t}{\sqrt{d_k}} \right) V_t^j \right]_{m=1}^{n}, \qquad (12)$$

where Q_t^i, K_t^j, and V_t^j are linear projections for object i and its neighbors j at time t.

(2) Temporal Attention: After obtaining spatial information, we apply a **temporal convolution** operation on temporal edges in the spatio-temporal graph to capture dynamics over time. For the temporal modeling, we consider each object independently and employ a canonical transformer to extract the temporal dependencies. Within the temporal attention mechanism, queries Q, keys K, and values V represent the embeddings of all objects at a fixed time step and pertaining to the distinct time steps of a specific object. This design enables each object to leverage information from its neighboring objects, thereby recovering valuable spatial insights.

Figure 6 (b) shows an example of temporal convolution at time t, where $Q_i \in t$ and K_j and V_j are assigned to neighborhood points at time $t-1$ and $t+1$ respectively for the same trajectory $\mathcal{T}_2$ which uses sliding windows (e.g., $t-1$ to $t+1$) to model stepwise dependencies. Here, we leverage *co-linear attention* where Q_i, K_j, and V_j are linear projections which are learned independently from input embeddings for a given trajectory T_i.

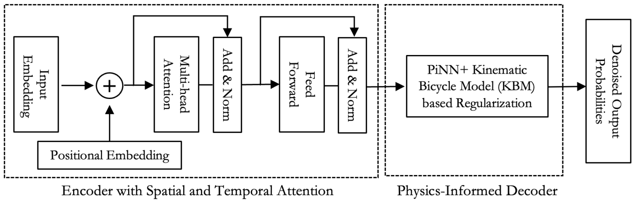

Input Embedding
Positional Embedding
Multi-head Attention
Add & Norm
Feed Forward
Add & Norm
PiNN+ Kinematic Bicycle Model (KBM) based Regularization
Denoised Output Probabilities
Encoder with Spatial and Temporal Attention
Physics-Informed Decoder

Figure 7: Proposed Encoder-Decoder Model

Figure 7 shows the components within the Context-Informed Encoder, where the entire spatial-temporal attention mechanism is performed in a multi-head attention module. The module is later learned by a feedforward layer via multiple skip connections and layer normalization, i.e., LayerNorm(x+Sublayer(x)), to ensure stability. The final learned representation is sent to a physics-informed decoder, which utilizes a PiNN architecture with a fully connected layer (FC) and incorporates a simplified version of the kinematic bicycle model as an integral part of a physics-informed regularizer.

3.3 Physics-informed Decoder

Data-driven generative models capture complex patterns but often overlook physical constraints, thereby limiting their robustness in noisy or unseen scenarios. The principle of physics-informed neural networks (PiNNs) is to train a neural network to learn from observed data and satisfy underlying physical equations by minimizing a loss function, which is a weighted sum of the standard reconstruction loss, the variational lower bound (VLB), and the physics-based loss. The ***variational lower bound*** ensures the model's predictions align with observed trajectory data. In contrast, the *physics-based loss* enforces adherence to the underlying kinematic constraints governing physical laws. Hence, this dual objective complements data-driven approaches by incorporating physical consistency into their predictions.

Kinematic Bicycle Model (KBM): This is a 3-degrees of freedom (DoF) formulation of x and y coordinate, and heading angle δ given a constant velocity v, shown in Figure 8, where x and y represent the coordinates of the rear wheel in the map, v denotes the velocity of the model at the rear wheel, L is the wheelbase of the object, δ is the heading angle, and ψ is the yaw angle of the object.

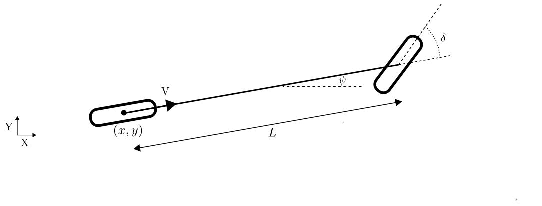


Figure 8: The kinematic bicycle model

However, due to dataset limitations, the physics-based information and object dimensions were absent in current datasets, which are required to calculate the complex kinematic state of the object or agent (e.g., slip angle). To address this, we integrate a ***simplified-version*** of the kinematic bicycle model that reduces the dual-axle model to a ***single-axle***, and compute physical parameters by considering consecutive kinematic states at a time.

Single-Axle Bicycle Model (S-KBM): At low speeds, KBM assumes negligible slip, slope, and inertial effects. Wheels are lumped into a single axis, and pitch, roll, and vertical dynamics are ignored, with motion governed by curvature or steering inputs. In this work, we employ a reduced form where the slip angle (between velocity and heading) is set to zero, yielding a computationally efficient yet physically grounded model [25, 57].

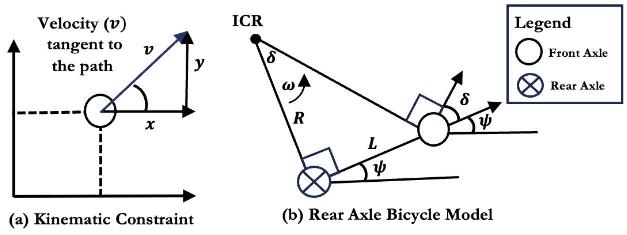


Figure 9: Simplified kinematic bicycle model (S-KBM)

Figure 9 (a) shows the simplified version of the kinematic-bicycle model commonly used for normal (i.e., smooth, non-high speed) mobility conditions. To ensure smooth and realistic predictions, we regularize trajectory reconstruction [56], which is sufficient for datasets without high-speed maneuvers. The S-KBM model's object state is defined by $z = [x, y, \psi]^\top$, and its input is defined by $u = [v, \delta]$, where (x, y) is the position, ψ is the heading, and v is the velocity. Substituting $x = v\cos(\psi)$, $y = v\sin(\psi)$, and $\psi = \omega$ using the rear axle formulation depicted in Figure 9 (b), the model's state evolution is then governed by:

$$\dot{z} = f(z, u) = \begin{bmatrix} v\cos(\psi) & v\sin(\psi) & \frac{v\tan\delta}{L} \end{bmatrix}^\top, \tag{13}$$

where ψ is the heading angle (also $\psi = \omega$), and $\omega = v/R$ is the rotation rate with turning radius $R = L/\tan(\delta)$. The substitution thus yields $\dot{\psi} = v \cdot \tan(\delta)/L$. The rate of heading change $\dot{\psi}$ is driven by the curvature input κ, defined at the front axle center. The S-KBM maps control inputs to state derivatives using object-specific parameters, such as wheelbase. However, since our real-world dataset lacks such details (e.g., wheelbase, steering angle), we use curvature κ as the control input. This enables an object-agnostic formulation of the motion model based on the state $\mathbf{x} = (x, y, \psi, v)$:

$$\frac{d\mathbf{x}}{dt} = \frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \\ v \end{bmatrix} = \begin{bmatrix} v\cos(\psi) \\ v\sin(\psi) \\ v\kappa \\ a \end{bmatrix} \tag{14}$$

3.4 Training and Inference

Variational Lower Bound: To effectively train the diffusion model, we derive its training objective by minimizing the negative log-likelihood of the model's predicted probability distribution and the true distribution of the data. We leverage Jensen's inequality to derive a variational lower bound to facilitate optimization:

$$\begin{aligned} \mathcal{L} &= -\mathbb{E}_{q(x_{0:T})} \log p_\theta(x_0) \\ &\leq -\mathbb{E}_{q(x_{0:T})} \log \frac{p_\theta(x_{0:T})}{q(x_{1:T} \mid x_0)} \\ &= \mathbb{E}_{q(x_{0:T})} \left[\log \frac{q(x_{1:T} \mid x_0)}{p_\theta(x_{0:T})} \right] \\ &= \mathcal{L}_{\text{VLB}} \end{aligned} \tag{15}$$

This step reformulates the problem into minimizing the KL-divergence between the two distributions. Further derivation results in a decomposition of the loss into entropy and multiple KL-divergence terms:

$$\mathcal{L}_{\text{VLB}} = \mathbb{E}_q \left[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t \mid x_{t-1})}{p_\theta(x_{t-1} \mid x_t)} \right] \tag{16}$$

This loss is further decomposed as:

$$\begin{aligned} \mathcal{L}_{\text{VLB}} = \mathbb{E}_q \Big[&-\log p_\theta(x_0 \mid x_1) + \mathrm{D}_{\text{KL}} \left(q(x_T \mid x_0) \parallel p_\theta(x_T) \right) \\ &+ \sum_{t=2}^{T} \mathrm{D}_{\text{KL}} \left(q(x_{t-1} \mid x_t, x_0) \parallel p_\theta(x_{t-1} \mid x_t) \right) \Big] \end{aligned} \tag{17}$$

To simplify this, we assume both q and p_θ are Gaussian distributions and we leverage the denoising process of DDPM to train our model. Hence, the KL divergence between them becomes:

$$\mathcal{L}_{\text{VLB}} = \mathbb{E}_q\left[\frac{1}{2}\left\|\Sigma_\theta(x_t,t)^{-1/2}\left(\tilde{\mu}_t(x_t,x_0)-\mu_\theta(x_t,t)\right)\right\|^2\right]+C \quad (18)$$

Once the diffusion model is trained, we can generate x_{t-1} from x_t according to Equation 19, iteratively, until the original sample x_0 is reconstructed. Then, anomaly detection is performed by comparing x_0 with the reconstructed trajectory $\hat{x}_0$. From the reparameterization trick and Equation 18, the sampling process to obtain x_{t-1} is given by:

$$x_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\left(\frac{x_t-\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t,t)}{\sqrt{\bar{\alpha}_t}}\right)+\sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\cdot\epsilon_\theta(x_t,t)+\sigma_t\epsilon_t, \quad (19)$$

where $\epsilon_t \sim \mathcal{N}(0,\mathbf{I})$ and ϵ_θ is the trained neural network. To control the stochasticity of the sampling, we introduce a hyperparameter η and define $\sigma_t(\eta)^2 = \eta \cdot \tilde{\beta}_t$, where $\eta \in \mathbb{R}^+$.

Physics-based Regularization: To guide learning with physical plausibility, we incorporate the simplified-kinematic bicycle model (S-KBM) as a prior within the diffusion framework. The overall loss combines the variational lower bound and a physics loss that penalizes deviations from S-KBM dynamics, including high acceleration and sharp curvature, promoting smooth and realistic trajectories.

$$\mathcal{L}_{\text{Phy}} = w_1\left\|\frac{d\hat{x}}{dt}-\hat{v}\cos\hat{\psi}\right\|^2 + w_2\left\|\frac{d\hat{y}}{dt}-\hat{v}\sin\hat{\psi}\right\|^2 + w_3\left\|\frac{d\hat{\theta}}{dt}-\hat{v}\hat{\kappa}\right\|^2 + w_4\left\|\frac{d\hat{v}}{dt}-\hat{a}\right\|^2 \quad (20)$$

As shown in Equation 20, the physics loss captures violations of S-KBM differential constraints. It acts as a regularizer against physically implausible motion, enforcing low-level kinematic constraints that reflect geometric and control limitations. The overall training objective combines variational and physics-based losses:

$$\mathcal{L}_{\text{Rec}}^{\text{Pi-DPM}} = \gamma_1\mathcal{L}_{\text{VLB}} + \gamma_2\mathcal{L}_{\text{Phy}} \quad (21)$$

4 Experiment Evaluation

We conducted experiments on real-world data to compare Pi-DPM with state-of-the-art trajectory reconstruction and anomaly detection methods. Ablation studies were also included as well as a sensitivity analysis on the reconstruction threshold λ. Figure 10 summarizes the overall design. The experiments addressed four research questions (RQs) as follows:

- **RQ1**: How does the proposed method with/without KBM and CIE compare with SOTA anomaly detection methods?
- **RQ2**: How does the proposed method with/without KBM and CIE compare with trajectory reconstruction methods?
- **RQ3**: How sensitive are the proposed methods to changes in λ threshold as compare with anomaly detection methods?
- **RQ4**: How generalizable are the proposed methods as we change the mobility datasets?

4.1 Experimental Design

Dataset: We evaluated our Pi-DPM data model on three real-world datasets spanning diverse spatial and temporal domains, namely: (1) Geolife [90], (2) MarineCadastre [6], and (3) Danish Maritime Authority [3]. These datasets vary significantly in scale and geographic coverage. For example, MarineCadastre includes over 1.1 million maritime trajectories within a global bounding box [−180°, 180°], [−90°, 90°], while the Danish Maritime dataset contains 132,135 coastal vessel trajectories within a bounded region around Denmark. In contrast, the Geolife dataset offers 17,621 urban object trajectories collected within a narrow spatial range in Beijing [116.25°, 116.55°], [39.75°, 40.05°]. The datasets also differ in trajectory granularity, with average durations ranging from 32.76 minutes (Geolife) to 394.4 minutes (MarineCadastre), and average distances spanning from 5.91 km to over 559 km.

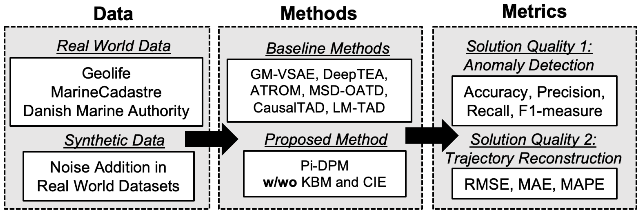

Data
Methods
Metrics
Real World Data
Geolife
MarineCadastre
Danish Marine Authority
Synthetic Data
Noise Addition in Real World Datasets
Baseline Methods
GM-VSAE, DeepTEA, ATROM, MSD-OATD, CausalTAD, LM-TAD
Proposed Method
Pi-DPM w/wo KBM and CIE
Solution Quality 1: Anomaly Detection
Accuracy, Precision, Recall, F1-measure
Solution Quality 2: Trajectory Reconstruction
RMSE, MAE, MAPE

Figure 10: Experiment Design

Synthetic Data Generation: Due to the scarcity of annotated anomalous trajectory data across all three datasets, we injected context-aware anomalies derived from first and second-order kinematic measures, including abrupt speed changes, bearing discontinuities, acceleration spikes, and high-jerk events. These perturbations emulate realistic irregularities such as unauthorized turns, erratic motion, and violations of navigational norms in both Geolife and the maritime datasets. The procedure is fully reproducible, with all thresholds, window sizes, and random seeds explicitly specified. The detailed methods for generating the synthetic anomalies we used for ground-truth verification are provided below.

(1) Geolife: First, we extracted sub-trajectory sequences from raw GPS trajectories using a fixed-seed deterministic pipeline. Data were de-duplicated, temporally ordered, and split if inter-point gaps exceeded 10 seconds. Segments lower or greater than 100 points or 120 seconds were discarded, respectively. For each segment, great-circle distance, speed, acceleration, jerk, and heading were computed, with speed/heading median-filtered for a fixed size window W=5. Segments were retained if median speed ≥ 6,m/s, 95th-percentile speed ≤ 45,m/s, median |acceleration| ≤ 3,m/s^2, and median turn rate ≤ 15°/s. Consecutive segments were stitched together if within 60,s, 50,m, and 20°. Anomalies were injected into $p \in$ 5%, 10%, 20% of subsequences via temporal compression ($\alpha \in [0.5, 0.9]$), bearing jumps (90°–150°), or lateral drift (30–120,m), with parameters resampled if bounds were violated.

(2) MarineCadastre and Danish Maritime Authority (DMA): We removed duplicate time-stamps, sorted chronologically, and discarded segments less than 50 points or greater than 60s. Kinematic states (i.e., acceleration, jerk, and turning rate) were derived from travel time and distance between two consecutive points

Table 2: Pi-DPM with kinematic constraints detects anomalous trajectories more accurately than baselines (5% anomalies)

Methods	Geolife				MarineCadastre				Danish Maritime Authority			
	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)
iBAT [85]	0.620	0.600	0.620	0.610	0.610	0.590	0.610	0.600	0.620	0.600	0.620	0.610
iBOAT [9]	0.670	0.650	0.670	0.660	0.660	0.640	0.660	0.650	0.670	0.650	0.670	0.660
GM-VSAE [48]	0.720	0.700	0.720	0.710	0.710	0.690	0.710	0.700	0.720	0.700	0.720	0.710
DeepTEA [28]	0.770	0.750	0.770	0.760	0.760	0.740	0.760	0.750	0.770	0.750	0.770	0.760
ATROM [22]	0.820	0.800	0.820	0.810	0.810	0.790	0.810	0.800	0.820	0.800	0.820	0.810
CausalTAD [43]	0.870	0.850	0.870	0.860	0.860	0.840	0.860	0.850	0.870	0.850	0.870	0.860
MSD-OATD [74]	0.900	0.880	0.900	0.890	0.890	0.870	0.890	0.880	0.900	0.880	0.900	0.890
LM-TAD [52]	0.920	0.900	0.920	0.910	0.910	0.890	0.910	0.900	0.920	0.900	0.920	0.910
Pi-DPM w/o KBM (Ours)	0.940	0.930	0.940	0.940	0.930	0.920	0.930	0.930	0.940	0.930	0.940	0.940
Pi-DPM w/o CIE (Ours)	0.960	0.950	0.960	0.960	0.950	0.940	0.950	0.950	0.960	0.950	0.960	0.960
Pi-DPM (Ours)	**0.980**	**0.970**	**0.990**	**0.980**	**0.970**	**0.960**	**0.980**	**0.970**	**0.980**	**0.970**	**0.990**	**0.980**

Bold marks statistically best results ($p < 0.05$) over the best baseline. ↑: higher is better. Values are means.

in a shipping lane using a median-filtered window size of 5. Using NOAA's Electronic Nautical Charts (ENC's) and Automatic Identification Systems (AIS) density maps to delineate shipping lanes, harbor approaches, and waypoints, we computed empirical speed over ground (SOG), turning rate, and lane-adherence profiles and then injected 5%, 10%, 20% anomalies via speed profile and turning-rate manipulation from ranges $[5.0, 9.0]$ knots degree within a *bearing* $\in [20^\circ, 60^\circ]$ points. Course deviation (30–200 m with lateral drift over *bearing* $\in [60^\circ, 180^\circ]$, outside NOAA shipping lanes, extracted from ENCs). We also included scenarios with waypoints skipping or unscheduled turns (90°–150° jump over *bearing* $\in [10, 30]$ points, cosine-tapered). For post-perturbation, kinematics were recomputed and anomalies violating IMO bounds were resampled, yielding reproducible, physically plausible maritime anomalies.

Task 1: Anomaly Detection: We first compared Pi-DPM against two traditional methods, iBAT [85] and iBOAT [9], which leverage the concept of degree of isolation to identify trajectories with anomalies. We also tested ML-based methods: GM-VSAE [48] employs a Gaussian mixture model to represent trajectory features in latent space. In contrast, ATROM [22] identifies a specific type of anomaly a trajectory exhibits. Other methods: DeepTEA [28] and CausalTAD [43], integrate mobility patterns with time-dependent trajectory patterns, while MSD-OATD [74] and LM-TAD [52] incorporate spatiotemporal dependencies to a certain extent.

Task 2: Trajectory Reconstruction: As Pi-DPM operates as a generative model followed by threshold-driven anomaly detection, its performance hinges on accurately representing what defines *normal* trajectories. Hence, we evaluated Pi-DPM against six generative models: VAE [80], TrajGAN [47], and DP-TrajGAN [87], DiffWave [37], DiffTraj [91], and ControlTraj [92]. More detailed descriptions for all baseline methods are in Appendix A and B.

Experimental Setup: We implemented the Pi-DPM framework using PyTorch 2.4 and Python 3.12 and executed all experiments on a single NVIDIA A100 40GB GPU. The test set was generated by selecting 20% of the training data, and the remaining 80% of the trips were split randomly into 60% training and 20% validation subsets over ten trials. We used a fixed number of diffusion steps $T = 1000$, consistent with prior work [31, 68, 69], and adopted a linear noise schedule with $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$, ensuring a low signal-to-noise ratio at $\mathbf{x}_T$ while preserving alignment between the forward and reverse processes. Our implementation is available in the official GitHub repository[1] and hyperparameter ranges are reported in Table 3.

Table 3: Hyperparameter Settings for Pi-DPM

Hyperparameter	Setting Value	Reference Range
Diffusion Steps	1000	200 ∼ 500
Skip Steps	4	1 ∼ 8
Guidance Scale	2.5	0.5 ∼ 8
β (linear)	0.0002 ∼ 0.04	–
Batch Size	512	≥ 128
Sampling Blocks	3	≥ 2
ResNet Blocks	3	≥ 2
Input Length	180	100 ∼ 200

Evaluation Metrics: To comprehensively assess the performance of learning-based methods for anomaly detection, we evaluated their effectiveness using a suite of standard metrics: accuracy, precision, recall, and F-measure. These metrics offer a comprehensive view of model performance, encompassing both the accuracy of predictions and the balance between false positives and false negatives. We further compared these methods with a trajectory reconstruction approach, where the primary objective was to synthesize trajectories that closely resemble real-world movement patterns. To quantify error estimation, we adopted established methodologies [15, 91], namely root mean square error (RMSE), mean absolute error (MAE), and mean average precision error (MAPE), for both comparison and ablation studies (more details in Appendix C).

Ablation Study: To assess the contribution of Pi-DPM's components, we compared baseline models, the full Pi-DPM, and two variants: (i) Pi-DPM w/o CIE, which removed the spatiotemporal encoder but retained the KBM regularizer, and (ii) Pi-DPM w/o KBM, which excluded the bicycle model while using spatiotemporal dependencies. The comparative results were integrated and shown with other baseline methods in Table 4, Table 5, and Table 6.

[1] https://github.com/arunshar/Physics-Informed-Diffusion-Probabilistic-Model.git

Table 4: Pi-DPM generates trajectories with the lowest error rate.

Methods	Geolife [90]			MarineCadastre [6]			Danish Maritime Authority [3]		
	RMSE (↓)	MAE (↓)	MAPE (↓)	RMSE (↓)	MAE (↓)	MAPE (↓)	RMSE (↓)	MAE (↓)	MAPE (↓)
VAE [80]	247.18 (3.25)	333.30 (4.10)	74.3 (0.85)	313.51 (3.12)	309.14 (3.45)	77.2 (0.92)	239.26 (2.90)	342.34 (3.78)	73.7 (0.81)
TrajGAN [58]	216.44 (2.98)	336.23 (3.84)	69.8 (0.79)	251.25 (2.75)	269.85 (3.21)	72.7 (0.84)	213.76 (2.66)	345.63 (3.69)	71.3 (0.80)
DP-TrajGAN [87]	208.02 (2.85)	284.26 (3.42)	66.2 (0.75)	258.69 (2.89)	232.87 (3.05)	68.2 (0.78)	207.12 (2.73)	283.68 (3.28)	66.9 (0.76)
GM-VSAE [48]	290.12 (3.55)	330.45 (4.05)	74.1 (0.85)	310.32 (3.42)	305.56 (3.82)	76.9 (0.90)	255.48 (3.11)	340.21 (3.97)	74.0 (0.83)
DeepTEA [28]	275.34 (3.40)	325.12 (3.95)	73.0 (0.84)	305.78 (3.31)	300.43 (3.76)	75.8 (0.88)	250.23 (3.05)	335.10 (3.88)	73.2 (0.82)
ATROM [22]	260.25 (3.26)	320.03 (3.84)	72.0 (0.82)	300.12 (3.18)	295.34 (3.65)	74.7 (0.86)	245.36 (2.96)	330.22 (3.78)	72.1 (0.81)
CausalTAD [43]	245.18 (3.10)	310.78 (3.70)	71.0 (0.81)	290.44 (3.05)	285.67 (3.54)	73.6 (0.85)	240.15 (2.85)	325.01 (3.65)	71.2 (0.80)
MSD-OATD [74]	235.40 (2.95)	305.50 (3.62)	70.0 (0.80)	285.23 (2.92)	280.34 (3.49)	72.5 (0.84)	235.02 (2.78)	320.34 (3.59)	70.5 (0.79)
LM-TAD [52]	225.67 (2.88)	300.12 (3.54)	69.0 (0.79)	280.45 (2.87)	275.78 (3.45)	71.5 (0.83)	230.14 (2.70)	315.23 (3.51)	69.7 (0.78)
Diffwave [37]	250.68 (3.20)	333.53 (3.99)	74.7 (0.86)	313.79 (3.15)	309.60 (3.72)	77.9 (0.91)	261.56 (3.00)	343.46 (3.89)	75.3 (0.84)
DiffTraj [91]	225.34 (2.86)	345.13 (4.08)	72.3 (0.82)	252.73 (2.80)	272.38 (3.38)	73.1 (0.85)	215.21 (2.65)	364.66 (4.15)	71.7 (0.81)
ControlTraj [92]	211.50 (2.78)	284.85 (3.42)	69.1 (0.80)	261.87 (2.75)	235.45 (3.20)	68.5 (0.79)	208.53 (2.62)	298.60 (3.35)	69.3 (0.80)
Pi-DPM w/o KBM (Ours)	195.00 (2.50)	205.00 (3.00)	66.0 (0.77)	240.00 (2.60)	233.00 (3.05)	71.0 (0.79)	212.00 (2.40)	255.00 (3.10)	62.0 (0.75)
Pi-DPM w/o CIE (Ours)	165.00 (2.10)	180.00 (2.80)	63.5 (0.74)	230.00 (2.50)	231.00 (2.95)	60.0 (0.72)	190.00 (2.20)	230.00 (2.85)	60.5 (0.73)
Pi-DPM (Ours)	**143.33 (1.95)**	**160.70 (2.65)**	**61.3 (0.71)**	**224.15 (2.30)**	**229.51 (2.80)**	**57.0 (0.70)**	**179.54 (2.05)**	**216.29 (2.70)**	**59.7 (0.72)**

Bold marks statistically best results ($p < 0.05$) over the best baseline. ↓: lower is better. Values are mean (standard deviation).

4.2 RQ1: Anomaly Detection

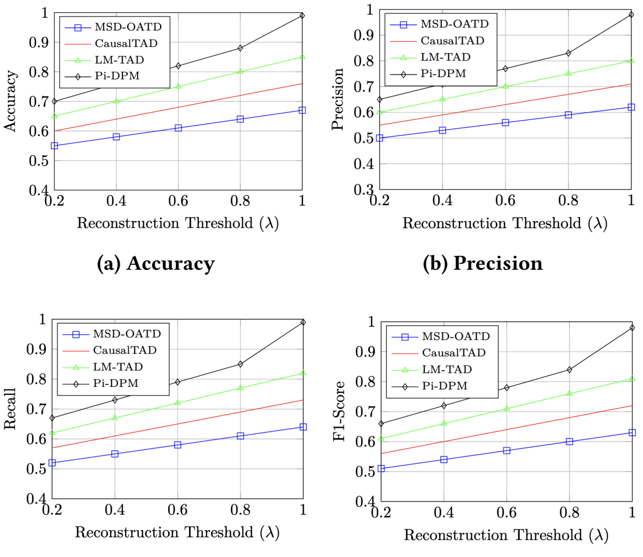
We assessed anomaly detection performance using accuracy, precision, recall, and F1-score, comparing Pi-DPM to baselines iBAT [85], iBOAT [9], GM-VSAE [48], DeepTEA [28], ATROM [22], CausalTAD [43], MSD-OATD [74], and LM-TAD [52]. As shown in Table 2, Pi-DPM with the kinematic bicycle model (KBM) consistently outperforms all baselines. On **Geolife**, it achieves 0.980 accuracy and 0.980 F1, surpassing CausalTAD at 0.860 F1 and LM-TAD at 0.910 F1, with similar gains on **MarineCadastre** (0.970 accuracy, 0.970 F1) and **Danish Maritime Authority** (0.980 accuracy, 0.990 recall). Among its variants, Pi-DPM w/o KBM yields the lowest performance (e.g., 0.940 F1 on Geolife), followed by Pi-DPM w/o CIE (0.960 F1 on Geolife), while full Pi-DPM demonstrates consistent superiority, confirming the utility of contextual knowledge and incorporating physics-informed motion constraints. These results, based on 5% injected anomalies and additional results for 10% and 20% anomalies, show similar trends, with Pi-DPM retaining the highest performance across all datasets. Due to space constraints, the results for 10% and 20% are shown in Tables 5 and 6 respectively.

4.3 RQ2: Trajectory Reconstruction

Since Pi-DPM applies a generative model and then detects anomalies using thresholds, its effectiveness depends on precisely modeling *normal* trajectory patterns. High-fidelity reconstruction ensures that deviations can be reliably identified as anomalies. As shown in Table 4, the full Pi-DPM achieves the lowest error rates across Geolife, Marine-Cadastre, and DMA, demonstrating the value of embedding physics-informed priors in a diffusion backbone. On Geolife, RMSE drops from 247.180 (VAE) to 143.330, with similar gains on the other datasets. Traditional baselines (VAE, TrajGAN, DP-TrajGAN) emphasize spatial patterns but model temporal correlations only weakly, yielding the highest MAPE (≥ 66%). Diffusion baselines (DiffWave, DiffTraj, ControlTraj) better capture stochastic dynamics yet remain motion-agnostic. Ablations confirm that ***both*** KBM and CIE are essential across all three datasets.

4.4 RQ3: Sensitivity Analysis

We assessed the effect of the reconstruction error threshold λ on model performance by varying λ from 0.2 to 1.0, and comparing Pi-DPM with MSD-OATD, CausalTAD, and LM-TAD. Figure 11 shows that all models improve with increasing lambda, but Pi-DPM consistently outperforms others across all metrics. Notably, Pi-DPM's accuracy and F1-score rise sharply to 0.99 and 0.98 at $\lambda = 1.0$, while MSD-OATD, CausalTAD, and LM-TAD reach only 0.67, 0.76, and 0.85 in accuracy, and 0.63, 0.72, and 0.81 in F1-score, respectively. Pi-DPM also excels in precision and recall, achieving scores of 0.98 and 0.99. This underscores Pi-DPM's robustness and anomaly detection capabilities, utilizing reconstruction-based criteria.



(a) Accuracy

(b) Precision

(c) Recall

(d) F-1 Score

Figure 11: Effect of increases in the λ threshold

4.5 RQ4: Transfer Learning

To evaluate both within and cross-domain generalizability, we pretrained Pi-DPM on one maritime region and then continued the training in another region using 5%–100% of domain data, comparing the results of Pi-DPM with only training on different percentages of data. As shown in Figure 12, *Origin* indicates that transfer learning is not applicable, while *Transfer* indicates applying transfer learning. Notably, even with just 5% of the data, the transfer learning model achieves a significantly lower error compared to the original one. As the percentage of data increases, the gap between the two models narrows. The results show that the PiDPM model exhibits strong adaptability and generalization capabilities when applied to different maritime domains. Beyond maritime-to-maritime transfer (within the domain), we further examined adaptation from MarineCadastre to both Geolife and DMA datasets (cross-domain) as shown in Figure 13. Despite the significant differences between urban mobility and maritime trajectories, the transfer setting shows only marginal loss relative to training from scratch, confirming that Pi-DPM captures transferable trajectory structures across heterogeneous domains. This robustness enables rapid adaptation in diverse applications such as maritime surveillance and urban planning.

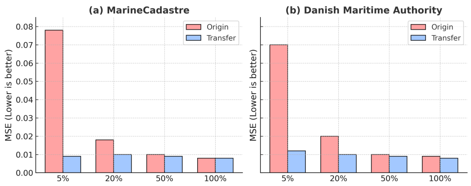


Figure 12: Transfer Learning within Maritime Domain

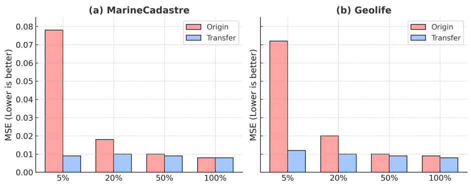


Figure 13: Transfer learning within/across Domains

5 Related Work

Trajectory anomaly detection literature [14, 55, 89] often relies on microscopic traffic simulators [20, 38, 70] and their city-scale extensions [44, 86], which embed car-following and lane-change heuristics [4] but struggle with the stochasticity of real mobility[23]. Similarity-based methods (e.g., edit distance, LCS, DTW, sliding windows) [5, 10, 11, 45, 82] have shown early promise. In contrast, simulation-based generators [67] and noise perturbations [84] still rely on historical assumptions that miss fine-scale spatiotemporal dependencies [40, 53, 81]. Grid-based approaches [9, 85] further simplify spatial interactions [17–19]. Generative models that consider spatiotemporal dependencies [22, 28, 43, 48, 74] often treat entire trips as independent, overlooking cross-route or asynchronous behaviors and remain limited in capturing complex mobility patterns [39, 59–66, 72, 72], including anomaly detection.

Deep generative models have introduced a new paradigm for trajectory synthesis, which can be classified into discrete and continuous methods. Discrete models generate coarse-grained outputs such as grid-cell sequences [21, 42, 49], pixel-based maps [7, 46], road segments [12, 35, 77], or POI transitions [13, 75, 78], favoring computational efficiency at the expense of spatial and temporal resolution. In contrast, continuous models offer finer fidelity, aligning more closely with raw GPS traces. In addition, VAEs and GANs have enabled modeling of complex trajectory distributions [29, 58, 80]. However, these models often convert trajectories into low-resolution representations such as grids [21, 54, 83] or images [27, 76], which degrade spatial-temporal accuracy [50]. While effective for group-level simulation, they fall short in reconstructing fine-grained, individual trajectories. Diffusion-based models [24, 32, 91] have recently emerged as a compelling alternative, offering continuous and high-fidelity reconstruction capabilities suitable for anomaly detection. However, purely data-driven methods often exhibit limitations in scientific domains due to their reliance on large-scale labeled data and inability to honor physical laws, resulting in implausible outputs in fields like climate science and biology [1, 16, 79]. Our proposed method addresses these gaps by embedding kinematic constraints from the bicycle model [56], ensuring that generated trajectories are not only statistically coherent but also physically feasible.

6 Conclusion and Future Work

We propose a Physics-Informed Diffusion Probabilistic Model (Pi-DPM) to capture GPS Spoofing behavior in maritime and urban domains, utilizing both contextual and physical parameters. To enhance reconstruction error estimation, we integrate synchronous nearby trajectory behavior in the encoder and the kinematic bicycle model [56] at the decoder stage, thereby leveraging physical parameters. Experimental results demonstrate that Pi-DPM outperforms state-of-the-art methods on both anomaly detection and trajectory reconstruction tasks. Additionally, the proposed model is more generalizable across various application domains.

Future Work: We plan to extend Pi-DPM with additional physical parameters (e.g., slip angle, jerk) and more sophisticated kinematic models, such as a 4-wheel formulation [26, 57] of KBM, for improved maneuvering accuracy. We can also evaluate Pi-DPM on richer modalities (e.g., on-board diagnostics) and integrate multimodal auxiliary datasets via contrastive learning [8, 41] to enhance robustness. Another direction is to optimize the computational efficiency for real-time deployment in safety-critical domains. In the long term, we envision physics-informed *foundation models* [51] to generalize and support a diverse range of anomaly detection tasks.

Acknowledgments

This material is based on work supported by the USDA under Grant No. 2023-67021-39829, the National Science Foundation under Grants No. 2118285, 2040459, 1737633, 1901099, and 1916518, the USDOE Office of Energy Efficiency and Renewable Energy under FOA No. DE-FOA0002044, USDA under Grant No. 2021-51181-35861, and USDOD under Grant No. HM04762010009. We would also like to thank Kim Koffolt and the Spatial Computing Research Group for their helpful comments and refinements.

References

[1] Mark Alber, Adrian Buganza Tepole, William R Cannon, Suvranu De, Salvador Dura-Bernal, Krishna Garikipati, George Karniadakis, William W Lytton, Paris Perdikaris, Linda Petzold, et al. 2019. Integrating machine learning and multiscale modeling—perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences. *NPJ digital medicine* 2, 1 (2019), 115.

[2] Hossein Amiri, Ruochen Kong, and Andreas Züfle. 2024. Urban Anomalies: A Simulated Human Mobility Dataset with Injected Anomalies. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection*. ACM, Atlanta, Georgia, 1–11.

[3] Danish Maritime Authority. 2024. Danish Maritime Authority. https://www.dma.dk/safety-at-sea/navigational-information/download-data.

[4] MF Aycin and RF Benekohal. 1999. Comparison of car-following models for simulation. *Transportation research record* 1678, 1 (1999), 116–127.

[5] Petko Bakalov, Marios Hadjieleftheriou, Eamonn Keogh, and Vassilis J Tsotras. 2005. Efficient trajectory joins using symbolic representations. In *Proceedings of the 6th international conference on Mobile data management*. IEEE, Ayia Napa Cyprus, 86–93.

[6] BOEM and NOAA. 2020. MarineCadastre. https://marinecadastre.gov/ais/.

[7] Chu Cao and Mo Li. 2021. Generating mobility trajectories with retained data utility. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. ACM, Singapore, 2610–2620.

[8] Yanchuan Chang, Jianzhong Qi, Yuxuan Liang, and Egemen Tanin. 2023. Contrastive trajectory similarity learning with dual-feature attention. In *2023 IEEE 39th International conference on data engineering (ICDE)*. IEEE, Anahein, CA, 2933–2945.

[9] Chao Chen, Daqing Zhang, Pablo Samuel Castro, Nan Li, Lin Sun, Shijian Li, and Zonghui Wang. 2013. iBOAT: Isolation-based online anomalous trajectory detection. *IEEE Transactions on Intelligent Transportation Systems* 14, 2 (2013), 806–818.

[10] Lei Chen et al. 2005. Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*. ACM, Baltimore, Maryland, 491–502.

[11] Yun Chen and Jignesh M Patel. 2009. Design and evaluation of trajectory join algorithms. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 1, 266–275.

[12] Seongjin Choi, Jiwon Kim, and Hwasoo Yeo. 2021. TrajGAIL: Generating urban vehicle trajectories using generative adversarial imitation learning. *Transportation Research Part C: Emerging Technologies* 128 (2021), 103091.

[13] Bangchao Deng, Xin Jing, Tianyue Yang, Bingqing Qu, Dingqi Yang, and Philippe Cudre-Mauroux. 2025. Revisiting Synthetic Human Trajectories: Imitative Generation and Benchmarks Beyond Datasaurus. *arXiv:2409.13790* 1, 1 (2025), 201–212.

[14] Somayeh Dodge et al. 2008. Towards a taxonomy of movement patterns. *Information visualization* 7, 3-4 (2008), 240–252.

[15] Yuntao Du, Yujia Hu, Zhikun Zhang, Ziquan Fang, Lu Chen, Baihua Zheng, and Yunjun Gao. 2023. Ldptrace: Locally differentially private trajectory synthesis. *Proceedings of the VLDB Endowment* 16, 8 (2023), 1897–1909.

[16] J H Faghmous and Vipin Kumar. 2014. A big data guide to understanding climate change: Case for theory-guided data science. *Big data* 2, 3 (2014), 155–163.

[17] Majid Farhadloo, Arun Sharma, Jayant Gupta, Alexey Leontovich, Svetomir N Markovic, and Shashi Shekhar. 2024. Towards spatially-lucid ai classification in non-euclidean space: An application for mxif oncology data. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*. SIAM, 616–624.

[18] Majid Farhadloo, Arun Sharma, Alexey Leontovich, Svetomir N Markovic, and Shashi Shekhar. 2025. Spatially-Delineated Domain-Adapted AI Classification: An Application for Oncology Data. In *Proceedings of the 2025 SIAM International Conference on Data Mining (SDM)*. SIAM, 487–496.

[19] Majid Farhadloo, Arun Sharma, Shashi Shekhar, and Svetomir Markovic. 2024. Spatial computing opportunities in biomedical decision support: The atlas-ehr vision. *ACM Transactions on Spatial Algorithms and Systems* 10, 3 (2024), 1–36.

[20] Martin Fellendorf and Peter Vortisch. 2010. Microscopic traffic flow simulator VISSIM. *Fundamentals of traffic simulation* 1, 1 (2010), 63–93.

[21] Jie Feng et al. 2020. Learning to simulate human mobility. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. ACM, New York, NY, USA, 3426–3433.

[22] Qiang Gao, Xiaohan Wang, Chaoran Liu, Goce Trajcevski, Li Huang, and Fan Zhou. 2023. Open anomalous trajectory recognition via probabilistic metric learning. In *IJCAI*. IJCAI, Macau, China, 121–132.

[23] Subhankar Ghosh, Jayant Gupta, Arun Sharma, Shuai An, and Shashi Shekhar. 2024. Reducing false discoveries in statistically-significant regional-colocation mining: A summary of results. *arXiv preprint arXiv:2407.02536* (2024).

[24] Subhankar Ghosh, Arun Sharma, Jayant Gupta, Aneesh Subramanian, and Shashi Shekhar. 2024. Towards kriging-informed conditional diffusion for regional sea-level data downscaling: A summary of results. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*. 372–383.

[25] Agapius Bou Ghosn, Philip Polack, and Arnaud de La Fortelle. 2022. Learning-based Observer Evaluated on the Kinematic Bicycle Model. In *2022 10th International Conference on Systems and Control (ICSC)*. IEEE, Egypt, 423–428.

[26] Thomas D. Gillespie. 1992. *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, Inc., Warrendale, PA, USA. https://doi.org/10.4271/R-114

[27] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Montreal, Canada, 2672–2680.

[28] Xiaolin Han, Reynold Cheng, Chenhao Ma, and Tobias Grubenmann. 2022. DeepTEA: Effective and efficient online time-dependent trajectory outlier detection. *Proceedings of the VLDB Endowment* 15, 7 (2022), 1493–1505.

[29] Nils Henke, Shimon Wonsak, Prasenjit Mitra, Michael Nolting, and Nicolas Tempelmeier. 2023. Condtraj-gan: Conditional sequential gan for generating synthetic vehicle trajectories. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, Springer, Cham, 79–91.

[30] Alex Hern. 2020. Berlin artist uses 99 phones to trick Google into traffic jam alert. *The Guardian* 3 (2020), 13–25.

[31] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Barcelona, Spain, 4565–4573.

[32] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, Vol. 33. Curran Associates, Inc., Red Hook, NY, USA, 6840–6851.

[33] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[34] Haoji Hu, Jina Kim, Jinwei Zhou, Sofia Kirsanova, JangHyeon Lee, and YaoYi Chiang. 2024. Context-Aware Trajectory Anomaly Detection. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection*. ACM, New York, NY, USA, 12–15.

[35] Wenjun Jiang, Wayne Xin Zhao, Jingyuan Wang, and Jiawei Jiang. 2023. Continuous trajectory generation based on two-stage GAN. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. AAAI Press, Palo Alto, CA, 4374–4382.

[36] Lance Kennedy and Andreas Züfle. 2024. Kinematic Detection of Anomalies in Human Trajectory Data. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection*. ACM, New York, NY, USA, 94–97.

[37] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2021. DiffWave: A Versatile Diffusion Model for Audio Synthesis. In *International Conference on Learning Representations*. ICLR, International Conference on Learning Representations, Online, xx–yy. https://openreview.net/forum?id=a-xFK8Ymz5J

[38] Daniel Krajzewicz, Jakob Erdmann, Michael Behrisch, and Laura Bieker. 2012. Recent development and applications of SUMO-Simulation of Urban MObility. *International journal on advances in systems and measurements* 5, 3&4 (2012), 1–10.

[39] K Vimal Kumar, Divakar Yadav, and Arun Sharma. 2015. Graph based technique for hindi text summarization. In *Information Systems Design and Intelligent Applications: Proceedings of Second International Conference INDIA 2015, Volume 1*. Springer, 301–310.

[40] Chaoneng Li, Guanwen Feng, Yunan Li, Ruyi Liu, Qiguang Miao, and Liang Chang. 2024. Difftad: Denoising diffusion probabilistic models for vehicle trajectory anomaly detection. *Knowledge-Based Systems* 286 (2024), 111387.

[41] Shuzhe Li, Wei Chen, Bingqi Yan, Zhen Li, Shunzhi Zhu, and Yanwei Yu. 2023. Self-supervised contrastive representation learning for large-scale trajectories. *Future Generation Computer Systems* 148 (2023), 357–366.

[42] Siyu Li, Toan Tran, Haowen Lin, John Krumm, Cyrus Shahabi, Lingyi Zhao, Khurram Shafique, and Li Xiong. 2024. Geo-Llama: Leveraging LLMs for Human Mobility Trajectory Generation with Spatiotemporal Constraints. *arXiv preprint arXiv:2408.13918* xx, yy (2024), xx–yy.

[43] Wenbin Li, Di Yao, Chang Gong, Xiaokai Chu, Quanliang Jing, Xiaolei Zhou, Yuxuan Zhang, Yunxia Fan, and Jingping Bi. 2024. Causaltad: Causal implicit generative model for debiased online trajectory anomaly detection. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, IEEE, San Diego, CA, USA, 4477–4490.

[44] Chumeng Liang, Zherui Huang, Yicheng Liu, Zhanyu Liu, Guanjie Zheng, Hanyuan Shi, Kan Wu, Yuhao Du, Fuliang Li, and Zhenhui Jessie Li. 2023. Cblab: Supporting the training of large-scale traffic control policies with scalable traffic simulation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 4449–4460.

[45] Rake& Agrawal King-lp Lin and HSSK Shim. 1995. Fast similarity search in the presence of noise, scaling, and translation in time-series databases. In *Proceeding of the 21th International Conference on Very Large Data Bases*. Citeseer, VLDB Endowment, Zurich, Switzerland, 490–501.

[46] Yan Lin, Huaiyu Wan, Jilin Hu, Shengnan Guo, Bin Yang, Youfang Lin, and Christian S Jensen. 2023. Origin-destination travel time oracle for map-based services. *Proceedings of the ACM on Management of Data* 1, 3 (2023), 1–27.

[47] Xi Liu, Hanzhou Chen, and Clio Andris. 2018. trajGANs: Using generative adversarial networks for geo-privacy protection of trajectory data (Vision paper).

In *Location privacy and security workshop*. N/A, N/A, 1–7.

[48] Yiding Liu, Kaiqi Zhao, Gao Cong, and Zhifeng Bao. 2020. Online anomalous trajectory detection with deep generative sequence modeling. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, IEEE, Dallas, TX, USA, 949–960.

[49] Qingyue Long, Huandong Wang, Tong Li, Lisi Huang, Kun Wang, Qiong Wu, Guangyu Li, Yanping Liang, Li Yu, and Yong Li. 2023. Practical Synthetic Human Trajectories Generation Based on Variational Point Processes. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, New York, NY, USA, 4561–4571.

[50] Massimiliano Luca, Gianni Barlacchi, Bruno Lepri, and Luca Pappalardo. 2021. A survey on deep learning for human mobility. *ACM Computing Surveys (CSUR)* 55, 1 (2021), 1–44.

[51] Gengchen Mai, Weiming Huang, Jin Sun, Suhang Song, Deepak Mishra, Ninghao Liu, Song Gao, Tianming Liu, Gao Cong, Yingjie Hu, Chris Cundy, Ziyuan Li, Rui Zhu, and Ni Lao. 2024. On the Opportunities and Challenges of Foundation Models for GeoAI (Vision Paper). *ACM Trans. Spatial Algorithms Syst.* 10, 2, Article 11 (July 2024), 46 pages. https://doi.org/10.1145/3653070

[52] Jonathan Kabala Mbuya, Dieter Pfoser, and Antonios Anastasopoulos. 2024. Trajectory Anomaly Detection with Language Models. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*. ACM, San Francisco, CA, USA, 208–219.

[53] Sandro Meloni, Nicola Perra, Alex Arenas, Sergio Gómez, Yamir Moreno, and Alessandro Vespignani. 2011. Modeling human mobility responses to the large-scale spreading of infectious diseases. *Scientific reports* 1, 1 (2011), 62.

[54] Kun Ouyang, Reza Shokri, David S Rosenblum, and Wenzhuo Yang. 2018. A non-parametric generative model for human trajectories.. In *IJCAI*, Vol. 18. International Joint Conferences on Artificial Intelligence, California, USA, 3812–3817.

[55] Dieter Pfoser and Yannis Theodoridis. 2003. Generating semantics-based trajectories of moving objects. *Computers, Environment and Urban Systems* 27, 3 (2003), 243–263.

[56] Philip Polack, Florent Altché, Brigitte d'Andréa Novel, and Arnaud de La Fortelle. 2017. The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?. In *2017 IEEE intelligent vehicles symposium (IV)*. IEEE, IEEE, Piscataway, NJ, USA, 812–818.

[57] Rajesh Rajamani. 2011. *Vehicle Dynamics and Control*. Springer, New York, NY, USA. https://doi.org/10.1007/978-1-4614-1433-9

[58] Jinmeng Rao, Song Gao, Yuhao Kang, and Qunying Huang. 2020. LSTM-TrajGAN: A deep learning approach to trajectory privacy protection. *arXiv preprint arXiv:2006.10521* 1, 1 (2020), –.

[59] Arun Sharma et al. 2022. Towards a tighter bound on possible-rendezvous areas: preliminary results. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*. 1–11.

[60] Arun Sharma et al. 2024. Physics-based abnormal trajectory gap detection. *ACM Transactions on Intelligent Systems and Technology* 15, 5 (2024), 1–31.

[61] Arun Sharma, Majid Farhadloo, Yan Li, Jayant Gupta, Aditya Kulkarni, and Shashi Shekhar. 2022. Understanding Covid-19 effects on mobility: A community-engaged approach. *AGILE: GIScience Series* 3 (2022), 14.

[62] Arun Sharma, Jayant Gupta, and Shashi Shekhar. 2022. Abnormal trajectory-gap detection: A summary (short paper). In *15th International Conference on Spatial Information Theory (COSIT 2022)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 26–1.

[63] Arun Sharma, Zhe Jiang, and Shashi Shekhar. 2022. Spatiotemporal data mining: A survey. *arXiv preprint arXiv:2206.12753* (2022).

[64] Arun Sharma and Shashi Shekhar. 2022. Analyzing trajectory gaps to find possible rendezvous region. *ACM Transactions on Intelligent Systems and Technology (TIST)* 13, 3 (2022), 1–23.

[65] Arun Sharma, Xun Tang, Jayant Gupta, Majid Farhadloo, and Shashi Shekhar. 2020. Analyzing trajectory gaps for possible rendezvous: A summary of results. In *11th International Conference on Geographic Information Science (GIScience 2021)-Part I (2020)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 13–1.

[66] Arun Sharma, Syed Mohammed Arshad Zaidi, Varun Chandola, Melissa R Allen, and Budhendra L Bhaduri. 2018. WebGIobe-A cloud-based geospatial analysis framework for interacting with climate data. In *Proceedings of the 7th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data*. 42–46.

[67] Filippo Simini, Gianni Barlacchi, Massimilano Luca, and Luca Pappalardo. 2021. A deep gravity model for mobility flows generation. *Nature communications* 12, 1 (2021), 6576.

[68] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, PMLR, N/A, 2256–2265.

[69] Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., Red Hook, NY, USA, 11895–11907.

[70] Chris Stanford, Suman Adari, Xishun Liao, Yueshuai He, Qinhua Jiang, Chenchen Kuai, Jiaqi Ma, Emmanuel Tung, Yinlong Qian, Lingyi Zhao, et al. 2024. NUMOSIM: A Synthetic Mobility Dataset with Anomaly Detection Benchmarks. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection*. ACM, New York, NY, USA, 68–78.

[71] Christiaan Triebert, Blacki Migliozzi, Alexander Cardia, Muyi Xiao, and David Botti. 2023. Fake Signals and American Insurance: How a Dark Fleet Moves Russian Oil. *International New York Times* 1 (2023), NA–NA.

[72] Pawan Kumar Upadhyay, Satish Chandra, and Arun Sharma. 2016. A novel approach of adaptive thresholding for image segmentation on GPU. In *2016 Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC)*. IEEE, 652–655.

[73] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017), 1–15.

[74] Chenhao Wang, Lisi Chen, Shuo Shang, Christian S Jensen, and Panos Kalnis. 2024. Multi-Scale Detection of Anomalous Spatio-Temporal Trajectories in Evolving Trajectory Datasets. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 2980–2990.

[75] Huandong Wang, Changzheng Gao, Yuchen Wu, Depeng Jin, Lina Yao, and Yong Li. 2023. PateGail: a privacy-preserving mobility trajectory generator with imitation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. AAAI Press, Washington, DC, USA, 14539–14547.

[76] Xingrui Wang, Xinyu Liu, Ziteng Lu, and Hanfang Yang. 2021. Large scale GPS trajectory generation using map based on two stage GAN. *Journal of Data Science* 19, 1 (2021), 126–141.

[77] Yong Wang, Guoliang Li, Kaiyu Li, and Haitao Yuan. 2022. A deep generative model for trajectory modeling and utilization. *Proceedings of the VLDB Endowment* 16, 4 (2022), 973–985.

[78] Yu Wang, Tongya Zheng, Shunyu Liu, Zunlei Feng, Kaixuan Chen, Yunzhi Hao, and Mingli Song. 2024. Spatiotemporal-augmented graph neural networks for human mobility simulation. *IEEE Transactions on Knowledge and Data Engineering* 1, 1 (2024), xx–yy.

[79] Jared Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. 2022. Integrating scientific knowledge with machine learning for engineering and environmental systems. *Comput. Surveys* 55, 4 (2022), 1–37.

[80] Tianqi Xia, Xuan Song, Zipei Fan, Hiroshi Kanasugi, QuanJun Chen, Renhe Jiang, and Ryosuke Shibasaki. 2018. Deeprailway: a deep learning system for forecasting railway traffic. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, IEEE, Piscataway, NJ, 51–56.

[81] Hao Xue, Flora Salim, Yongli Ren, and Nuria Oliver. 2021. MobTCast: Leveraging auxiliary trajectory forecasting for human mobility prediction. *Advances in Neural Information Processing Systems* 34 (2021), 30380–30391.

[82] Byoung-Kee Yi, Hosagrahar V Jagadish, and Christos Faloutsos. 1998. Efficient retrieval of similar time sequences under time warping. In *Proceedings 14th International Conference on Data Engineering*. IEEE, IEEE, Piscataway, NJ, 201–208.

[83] Yuan Yuan, Jingtao Ding, Huandong Wang, Depeng Jin, and Yong Li. 2022. Activity trajectory generation via modeling spatiotemporal dynamics. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 4752–4762.

[84] Paul A Zandbergen. 2014. Ensuring confidentiality of geocoded health data: Assessing geographic masking strategies for individual-level data. *Advances in medicine* 2014, 1 (2014), 567049.

[85] Daqing Zhang, Nan Li, Zhi-Hua Zhou, Chao Chen, Lin Sun, and Shijian Li. 2011. iBAT: detecting anomalous taxi trajectories from GPS traces. In *Proceedings of the 13th international conference on Ubiquitous computing*. ACM, Beijing, China, 99–108.

[86] Huichu Zhang, Siyuan Feng, Chang Liu, Yaoyao Ding, Yichen Zhu, Zihan Zhou, Weinan Zhang, Yong Yu, Haiming Jin, and Zhenhui Li. 2019. Cityflow: A multi-agent reinforcement learning environment for large scale city traffic scenario. In *The world wide web conference*. ACM, New York, NY, USA, 3620–3624.

[87] Jing Zhang, Qihan Huang, Yirui Huang, Qian Ding, and Pei-Wei Tsai. 2023. DP-TrajGAN: A privacy-aware trajectory generation model with differential privacy. *Future Generation Computer Systems* 142 (2023), 25–40.

[88] Zheng Zhang, Hossein Amiri, Zhenke Liu, Liang Zhao, and Andreas Züfle. 2024. Large language models for spatial trajectory patterns mining. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Geospatial Anomaly Detection*. ACM, New York, NY, USA, 52–55.

[89] Yu Zheng. 2015. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology (TIST)* 6, 3 (2015), 1–41.

[90] Yu Zheng, Xing Xie, and Wei-Ying Ma. 2010. GeoLife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.* 33, 2 (2010), 32–39.

[91] Yuanshao Zhu, Yongchao Ye, Shiyao Zhang, Xiangyu Zhao, and James Yu. 2023. Difftraj: Generating gps trajectory with diffusion probabilistic model. *Advances in Neural Information Processing Systems* 36 (2023), 65168–65188.

[92] Yuanshao Zhu, James Jianqiao Yu, Xiangyu Zhao, Qidong Liu, Yongchao Ye, Wei Chen, Zijian Zhang, Xuetao Wei, and Yuxuan Liang. 2024. Controltraj: Controllable trajectory generation with topology-constrained diffusion model. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 4676–4687.

A Baseline Anomaly Detection Methods

iBAT: Trajectories are first embedded in a spatio-temporal feature space (speed, heading change, stop duration, etc.), and random hyper-planes recursively split the space. Rare paths are isolated after only a few splits, so a short isolation depth directly yields a high anomaly score. [85].

iBOAT: iBOAT slides a fixed-length window along each route and, for every window, counts its k nearest neighbours in a large corpus of normal traffic. Windows with little neighbor support are deemed isolated and given high anomaly scores; the voyage's final score is the maximum window score, so even a brief deviation is enough to flag the whole track [9].

GM-VSAE: The Gaussian-Mixture Variational Sequence AutoEncoder learns rich spatio-temporal dependencies along a route, then clusters the resulting embeddings with a Gaussian-mixture prior to map each major "normal" route type into its own region of a continuous latent space. New trajectories are quickly projected into this space; those that cannot be well reconstructed or that fall outside the high-density mixture are flagged as anomalies [48].

DeepTA: DeepTA is a deep-probabilistic, time-aware anomaly detector that learns the evolving distribution of vehicle states, allowing it to flag outliers even under highly complex traffic patterns. To enable live monitoring, a lightweight, approximate version of the model trades a small amount of accuracy for the speed needed to surface abnormal behavior in real-time [28].

ATROM: The Anomaly Typology Recognition and Organization Model (ATROM) introduces a structured typology to classify anomalies across five dimensions: structure, distribution, context, semantics, and temporal patterns. This domain-independent framework improves interpretability, consistency, and adaptability across diverse spatiotemporal anomaly detection scenarios [22].

CausalTAD: CausalTAD integrates spatial clustering with temporal prediction for trajectory anomaly detection. It uses DBSCAN to form baseline clusters of normal behavior and trains a ProbSparse Transformer to model recent motion sequences. Discrepancies between predicted and clustered behavior—measured via dynamic time warping (DTW)—are used to flag anomalies. The method emphasizes causality and time-aware interpretation of deviation [43].

MSD-OATD: MST-OATD is a multi-scale system for real-time trajectory anomaly detection, capturing movement patterns at several spatial and temporal granularities. An integrated ranking mechanism continually refreshes its reference set with incoming data, allowing the model to adapt to shifting traffic behaviors while preserving detection accuracy [74].

LM-TAD: LM-TAD reinterprets a trajectory as a language-like token sequence and trains an autoregressive, causal-attention model to learn its joint probability distribution. User-specific tokens personalize the context, so points with low likelihood (high perplexity) are flagged as anomalies; this yields state-of-the-art results on the PoL dataset and competitive performance on the Porto taxi dataset, while supporting GPS, stay-poistrike a balance between activity tokens. An efficient key-value cache lets the model operate online without repeatedly recomputing attention, keeping latency low [52].

B Baseline Trajectory Reconstruction Methods

Variational Autoencoder (VAE): The VAE encodes trajectories into a lower-dimensional latent space using two convolutional layers followed by a fully connected layer. This compact representation captures key spatiotemporal features. A symmetric decoder reconstructs the trajectory using a fully connected layer followed by two deconvolutional layers. The model is trained by minimizing a combination of reconstruction loss and KL divergence, encouraging the latent space to follow a Gaussian prior [80].

Trajectory GAN (TrajGAN): TrajGAN synthesizes plausible trajectory data by training a generator–discriminator pair. The generator takes as input a noise vector concatenated with trajectory seeds, passing it through two fully connected layers followed by two convolutional layers to output synthetic trajectories. The discriminator evaluates the authenticity of the generated sequences via convolutional and dense layers. The adversarial training optimizes both networks to balance fidelity and diversity [58].

DP-TrajGAN: The DP-TrajGAN model addresses privacy-preserving trajectory synthesis by integrating an enhanced LSTM-based GAN with formal differential privacy (DP) mechanisms. The generator captures trajectory dynamics while preserving privacy by injecting calibrated noise into the training process. To optimize the privacy–utility trade-off, a Partially Observable Markov Decision Process (POMDP) is employed to allocate the privacy budget across training iterations adaptively. This results in synthetic trajectories that maintain statistical similarity to real-world patterns while safeguarding sensitive location information [87].

Diffwave: Originally designed for speech synthesis, Diffwave is a generative model based on the WaveNet architecture. It comprises 16 residual layers employing bidirectional dilated convolutions and skip connections to propagate contextual information across time steps. The output is transformed via tanh and sigmoid activations into a 1D convolutional decoder. Its autoregressive structure enables it to model long-range dependencies, making it suitable for sequential data reconstruction tasks [37].

DiffTraj: DiffTraj leverages diffusion processes to progressively refine noise into structured trajectories. Unlike GANs, it does not rely on adversarial training. The model incorporates conditioning via learned embeddings of start and end locations, thereby improving semantic coherence with respect to origin–destination constraints. This conditioning facilitates location-aware reconstruction without enforcing explicit topological priors [91].

ControlTraj: ControlTraj introduces a topology-constrained diffusion model for high-fidelity and geography-aware trajectory reconstruction. It incorporates a novel road segment autoencoder to learn fine-grained embeddings of underlying street networks. These embeddings, combined with trip-level attributes, guide a modified geographic denoising UNet architecture (GeoUNet), enabling the reconstruction of plausible trajectories conditioned on structural constraints. ControlTraj demonstrates adaptability across diverse spatial contexts and preserves geographical realism [92].

C Evaluation Metrics

Mean Squared Error (MSE): The MSE quantifies the average squared discrepancy between predicted values, $\hat{y}_i$, and observed values, y_i, for a dataset of size n. By squaring the errors, MSE

Table 5: Pi-DPM with kinematic constraints detects anomalous trajectories more accurately than baselines (10% anomalies)

Methods	Geolife				MarineCadastre				Danish Maritime Authority			
	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)
iBAT [85]	0.600	0.580	0.600	0.590	0.590	0.570	0.590	0.580	0.600	0.580	0.600	0.590
iBOAT [9]	0.650	0.630	0.650	0.640	0.640	0.620	0.640	0.630	0.650	0.630	0.650	0.640
GM-VSAE [48]	0.700	0.680	0.700	0.690	0.690	0.670	0.690	0.680	0.700	0.680	0.700	0.690
DeepTEA [28]	0.750	0.730	0.750	0.740	0.740	0.720	0.740	0.730	0.750	0.730	0.750	0.740
ATROM [22]	0.800	0.780	0.800	0.790	0.790	0.770	0.790	0.780	0.800	0.780	0.800	0.790
CausalTAD [43]	0.850	0.830	0.850	0.840	0.840	0.820	0.840	0.830	0.850	0.830	0.850	0.840
MSD-OATD [74]	0.880	0.860	0.880	0.870	0.870	0.850	0.870	0.860	0.880	0.860	0.880	0.870
LM-TAD [52]	0.900	0.880	0.900	0.890	0.890	0.870	0.890	0.880	0.900	0.880	0.900	0.890
Pi-DPM w/o KBM (Ours)	0.920	0.910	0.920	0.920	0.910	0.900	0.910	0.910	0.920	0.910	0.920	0.920
Pi-DPM w/o CIE (Ours)	0.940	0.930	0.940	0.940	0.930	0.920	0.930	0.930	0.940	0.930	0.940	0.940
Pi-DPM (Ours)	**0.960**	**0.950**	**0.970**	**0.960**	**0.950**	**0.940**	**0.960**	**0.950**	**0.960**	**0.950**	**0.970**	**0.960**

Bold marks statistically best results ($p < 0.05$) over the best baseline. ↑: higher is better. Values are means.

Table 6: Pi-DPM with kinematic constraints detects anomalous trajectories more accurately than baselines (20% anomalies)

Methods	Geolife				MarineCadastre				Danish Maritime Authority			
	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)	Acc. (↑)	Prec. (↑)	Rec. (↑)	F1 (↑)
iBAT [85]	0.580	0.560	0.580	0.570	0.570	0.550	0.570	0.560	0.580	0.560	0.580	0.570
iBOAT [9]	0.630	0.610	0.630	0.620	0.620	0.600	0.620	0.610	0.630	0.610	0.630	0.620
GM-VSAE [48]	0.680	0.660	0.680	0.670	0.670	0.650	0.670	0.660	0.680	0.660	0.680	0.670
DeepTEA [28]	0.730	0.710	0.730	0.720	0.720	0.700	0.720	0.710	0.730	0.710	0.730	0.720
ATROM [22]	0.780	0.760	0.780	0.770	0.770	0.750	0.770	0.760	0.780	0.760	0.780	0.770
CausalTAD [43]	0.830	0.810	0.830	0.820	0.820	0.800	0.820	0.810	0.830	0.810	0.830	0.820
MSD-OATD [74]	0.860	0.840	0.860	0.850	0.850	0.830	0.850	0.840	0.860	0.840	0.860	0.850
LM-TAD [52]	0.880	0.860	0.880	0.870	0.870	0.850	0.870	0.860	0.880	0.860	0.880	0.870
Pi-DPM w/o KBM (Ours)	0.900	0.890	0.900	0.900	0.890	0.880	0.890	0.890	0.900	0.890	0.900	0.900
Pi-DPM w/o CIE (Ours)	0.920	0.910	0.920	0.920	0.910	0.900	0.910	0.910	0.920	0.910	0.920	0.920
Pi-DPM (Ours)	**0.940**	**0.930**	**0.950**	**0.940**	**0.930**	**0.920**	**0.940**	**0.930**	**0.940**	**0.930**	**0.950**	**0.940**

Bold marks statistically best results ($p < 0.05$) over the best baseline. ↑: higher is better. Values are means.

emphasizes larger deviations, making it particularly sensitive to outliers. A lower MSE indicates higher prediction accuracy.

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 . \tag{22}$$

Root Mean Squared Error (RMSE): The RMSE is the square root of the MSE, providing an error measure in the same units as the original data. RMSE facilitates intuitive interpretation of error magnitude while retaining sensitivity to significant deviations.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}. \tag{23}$$

Mean Absolute Error (MAE): The MAE computes the average absolute difference between predicted and observed values. Unlike MSE, MAE assigns equal weight to all errors, offering robustness against outliers and a straightforward measure of average error.

$$\mathrm{MAE} = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| . \tag{24}$$

Mean Absolute Percentage Error (MAPE): The MAPE expresses the error as a percentage of the observed values, enabling scale-independent comparisons. MAPE is valuable for cross-dataset comparisons but may become unstable when y_i approaches zero.

$$\mathrm{MAPE} = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100. \tag{25}$$

Binary classification: Predictions are evaluated against ground-truth labels (TP, TN, FP, FN) using standard metrics:

$$\text{Accuracy (ACC)} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{26}$$

$$\text{Precision (P)} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{27}$$

$$\text{Recall (R)} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{28}$$

$$\text{F1-score (F1)} = \frac{2PR}{P + R}. \tag{29}$$