network

MDPI

Article

Forecasting-Aware Digital Twin Calibration for Reliable Multi-Horizon Traffic Prediction

Zeyad AlJundi [1], Taqwa A. Alhaj [1,*], Fatin A. Elhaj [2], Inshirah Idris [3] and Tasneem Darwish [4]

[1] Centre of Excellence in Cybercrime and Digital Forensics, Naif Arab University for Security Sciences, Riyadh 11452, Saudi Arabia; zaljundi@nauss.edu.sa
[2] College of Computing & Intelligent Systems, University of Khorfakkan, Sharjah 18119, United Arab Emirates; fatin.mohamed@ukf.ac.ae
[3] Computer Science and Information Technology College, Wadmedani Ahlia University, Wad Medani P.O. Box 402, Sudan; inshirah.idris@gmail.com
[4] Department of Computer Science, St. Francis Xavier University, Antigonish, NS B2G 2W5, Canada; tdarwish@stfx.ca
* Correspondence: talhaj@nauss.edu.sa

Abstract

Digital twin systems are becoming an important tool in intelligent transportation management, as they provide simulation-based environments for monitoring, analyzing, and predicting traffic behavior. However, the predictive performance of traffic digital twins is often limited by the quality and temporal consistency of sensor-level data generated from microscopic simulations. Most current calibration methods focus mainly on matching macroscopic traffic indicators, such as vehicle count and speed, without explicitly addressing the requirements of multi-horizon forecasting. This creates a gap between achieving realistic simulations and building reliable predictive models. This research proposes a forecasting-aware digital traffic twin framework that integrates microscopic SUMO simulation, controlled sensor-level observation modeling through geometric misalignment and noise injection, behavioral calibration, and deep temporal forecasting within a unified end-to-end structure. Unlike traditional calibration approaches, the proposed Genetic Algorithm (GA) reformulates calibration as a multi-step predictive optimization task. Simulation parameters are optimized by minimizing forecasting error produced by a lightweight proxy sequence model embedded within the calibration loop. In this way, calibration moves beyond simple statistical matching and instead emphasizes temporal learnability and forecasting stability, enabling the digital twin to generate traffic patterns more suitable for long-term prediction. Based on the calibrated traffic time series, both convolutional and recurrent deep learning models are evaluated under single-step and multi-step forecasting scenarios. To further examine generalizability, external validation is performed using the real-world PEMS-BAY dataset. The experimental findings demonstrate that forecasting-aware calibration reduces macroscopic traffic signal errors by around 50% for vehicle count and around 40% for average speed, improves temporal stability, and significantly enhances forecasting accuracy across both short-term and long-term horizons.

Keywords: intelligent transportation; SUMO; digital twin; traffic forecasting; deep learning; genetic algorithm; data calibration

Check for updates

Academic Editors: Manuel Jesus-Azabal, Vasco N. G. J. Soares and Adriano Raposo

Received: 5 January 2026
Revised: 15 February 2026
Accepted: 22 February 2026
Published: 6 March 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Transportation infrastructure represents a fundamental pillar of economic development and the functioning of modern societies. Rapid urban expansion, the continuous

growth in vehicle ownership, and increasing sustainability requirements have significantly amplified the complexity of managing transportation systems. For instance, in Saudi Arabia, the national road network spans approximately 268,000 km, while the railway system extends nearly 5330 km across the Kingdom. This rapidly expanding network must accommodate accelerating urbanization, rising mobility demand, and evolving transportation patterns, thereby encountering pressures similar to those experienced by large-scale systems [1]. This immense system faces persistent challenges, including severe traffic congestion, disparities in infrastructure quality, and escalating maintenance costs. Moreover, sharp increases in population and vehicle ownership—particularly in urban areas—have placed substantial pressure on existing infrastructure systems [2].

Despite the growing demands placed on transportation networks, conventional infrastructure management strategies remain largely reactive. In most cases, roadway deterioration or congestion must reach critical levels before corrective interventions are initiated. Such delayed responses are both costly and inefficient and may negatively affect the operational lifespan and safety of infrastructure assets [3].

In this context, the concept of Digital Twins (DTs), particularly within urban environments, has gained substantial attention due to its transformative potential in smart city planning and infrastructure management [2]. A digital twin is defined as a real-time virtual replica of a physical system, including its surrounding environment and operational processes. Through continuous data synchronization between the physical and virtual domains, DTs enable deeper insight into system behavior and enhance decision-making in complex urban ecosystems. The rapid growth of Urban Digital Twins (UDTs) is closely linked to advancements in smart city technologies, positioning DTs as a foundational tool for detailed modeling and analysis of urban physical systems [4]. Recent developments have further improved the efficiency and data-driven management capabilities of these technologies [5].

Within the transportation sector, simulation-based digital twin platforms such as SUMO (Simulation of Urban Mobility) have emerged as powerful and flexible tools for generating traffic data, analyzing network behavior, and assessing mobility strategies [6]. SUMO allows the construction of realistic and fully controllable virtual road networks equipped with virtual sensors capable of capturing traffic flow/count and speed. Although such platforms have significantly improved the visibility and understanding of traffic systems, their practical effectiveness is highly dependent on the availability of accurate and reliable traffic forecasts. High-quality forecasting is essential for proactive traffic control, congestion mitigation, resource allocation, and infrastructure management [7].

Traffic flow prediction involves extracting meaningful information from historical data using computational techniques to estimate the most likely traffic conditions in the future [8]. More details on the methods of historical traffic collection and estimation are presented in [9]. Fundamentally, traffic forecasting belongs to the broader class of time-series modeling problems [8]. However, traffic dynamics exhibit strong temporal dependencies arising from recurrent daily patterns (e.g., peak-hour congestion), short-term disturbances (e.g., sudden surges in demand), and long-term evolving trends (e.g., progressive congestion accumulation). Consequently, the ability to capture both localized temporal variations and long-range temporal dependencies is essential for reliable forecasting [10]. Local temporal patterns describe immediate changes in recent data, whereas long-term dependencies represent sustained interactions that develop over extended periods. Traditional statistical approaches and basic machine learning models often struggle to capture such multi-scale temporal structures, leading to unstable and inaccurate predictions, particularly for longer forecasting horizons [11].

The rapid advancement of deep learning has introduced powerful tools for feature extraction and pattern recognition, demonstrating exceptional potential in traffic prediction tasks. Deep learning architectures—including deep neural networks, recurrent neural networks, and long short-term memory (LSTM) networks—are capable of modeling complex spatiotemporal relationships within traffic data, achieving substantial performance gains over conventional methods. Moreover, their adaptive learning capabilities allow them to maintain robust prediction accuracy even in the presence of noise and high-dimensional input features [11].

Despite the increasing adoption of Digital Twins and SUMO-based simulation environments for traffic analysis, most existing forecasting studies assume that sensor-derived time-series data are inherently reliable and immediately suitable for modeling. In practice, SUMO virtual sensors—similar to real-world sensors—can produce noisy, fluctuating, or inconsistent measurements of flow, speed, and occupancy, thus degrading prediction performance, especially for long-term forecasts [11]. To obtain a close match between the observed and simulated traffic measurements, one has to perform a proper calibration of microscopic traffic simulation model parameters. Because there are a large number of unknown parameters involved, the calibration process can be a time-consuming and complex task. As a result, such a calibration process has been formulated as an optimization model in which a huge search space exists due to a wide range of each relevant model parameter [12]. Among these algorithms, Genetic Algorithm (GA) has been widely used due to its easy implementation and good performance in calibration and optimization [12]. Although genetic algorithm–based calibration can effectively align simulated traffic statistics with real-world observations, most existing methods remain largely descriptive. They focus on improving how realistic the simulation appears, without directly accounting for the needs of temporal forecasting models. As a result, the link between calibration accuracy and the reliability of predictions derived from digital-twin traffic time series is still not well understood [12,13]. Similarly, recent calibration frameworks built around SUMO focus primarily on reproducing realistic traffic patterns and matching observed traffic trends. Their main objective is to improve the visual and statistical similarity between simulation and reality, but they do not sufficiently evaluate how calibration quality affects predictive model performance or the temporal stability of traffic forecasts. Because of this, there is still a clear methodological gap between traffic simulation calibration and predictive traffic intelligence, especially in transportation systems that rely on digital twin technologies.

To address this issue, this study proposes a forecasting-aware digital traffic twin framework. The framework integrates microscopic traffic simulation; sensor-level observation modeling, behavioral calibration using GA; and deep temporal forecasting within a single end-to-end pipeline. Unlike traditional calibration methods that aim only to improve descriptive similarity, the proposed approach reformulates GA optimization as a multi-step predictive objective. This allows the digital twin to generate traffic dynamics that are physically realistic, temporally consistent, and more suitable for training forecasting models.

The main contributions of this study can be summarized as follows:

1. End-to-End Digital Traffic Twin Architecture A unified methodological framework that combines SUMO-based microscopic simulation, synthetic LiDAR sensing, GA-based behavioral calibration, and deep sequence forecasting into one integrated system.
2. Forecasting-Oriented GA Calibration Framework A novel calibration strategy that optimizes simulation parameters based on multi-step forecasting error rather than traditional traffic statistics. This shifts calibration from simple descriptive matching toward improving predictive reliability and robustness.

3. Systematic Temporal Forecasting Assessment A structured comparative evaluation of convolutional and recurrent deep learning architectures across single-step and multi-step prediction horizons, highlighting the critical role of long-term temporal dependency modeling in achieving stable and reliable traffic forecasts.
4. External Validation Using Real-World Traffic Data A benchmark-based validation conducted on the PEMS-BAY dataset, demonstrating that the predictive patterns learned within the calibrated digital twin align with actual traffic dynamics, thereby reinforcing the external validity and practical applicability of the proposed framework.

2. Literature Review

Transportation infrastructure is essential for economic growth and the everyday existence of individuals, as it is central to economic growth and civic life [2]. Traffic flow prediction is one of the fundamental issues in an Intelligent Transportation System (ITS) [8]. Traffic flow prediction can alleviate urban traffic congestion and improve safety and efficiency on a road section [8].

A thorough analysis of previous research is required to position the contributions of this study. In this literature review, three interrelated domains are summarized: microscopic simulation for the production of reliable traffic data, calibration methods for ensuring physical plausibility, and the development of deep learning architectures for temporal forecasting. Through an examination of the intersections and constraints between deep learning models and genetic algorithm-based calibration, this review identifies the particular niche addressed by the suggested framework.

2.1. Traffic Forecasting Data

Maintaining consistently high-quality data inputs remains a persistent challenge, particularly in large and dynamic urban environments [2]. Sufficient traffic data is the basis of accurate traffic forecasts models [14]. Unreliable or noisy data may degrade the performance of forecasting models and lead to poor decision-making by traffic managers. Acquisition of high-fidelity real-world traffic data presents substantial challenges, as it is dependent on vast amounts of data from diverse sources, such as infrastructure-embedded sensors, IoT devices, traffic cameras, and GPS systems [10]. Integrating these data poses significant challenges due to differences in formats, protocols, and update frequencies between various systems [2]. Furthermore, the continuous development of deep learning algorithms that significantly assist in building prediction models, these models require a large scale of training data to improve accuracy, which makes collecting real data difficult. In addition, some approaches in transportation systems require evaluation. Therefore, in order to simulate the real environment, non-realistic data are used due to the lack of various traffic data [10]. To mitigate these challenges, microscopic traffic simulation has emerged as a tool for generating traffic datasets. The simulation environments offer a controlled platform to generate urban traffic data. The Simulation of Urban Mobility (SUMO) is a widely used open-source traffic simulation suite used in academia and industry research [15]. For instance, SUMO's controllability and fidelity have been highlighted in recent studies [16,17] that synthesize urban mobility patterns for autonomous vehicle testing. However, in most existing studies, simulation-generated datasets are calibrated mainly to match observed traffic statistics or overall macroscopic realism, while limited attention is given to their suitability for downstream multi-horizon forecasting. As a result, the predictive stability of these datasets for temporal learning tasks remains uncertain [18,19].

2.2. Data Calibration

While microscopic simulation provides a powerful technique for traffic dataset generation, their output metrics statistically may not align with observed real-world data. Therefore, model calibration is important to minimize the difference between the simulation results and corresponding field measurements [12]. To obtain a close match between the observed and simulated traffic measurements, one has to perform a proper calibration of microscopic traffic simulation model parameters. Because there are a large number of unknown parameters involved, the calibration process can be a time-consuming and complex task. As a result, such a calibration process has been formulated as an optimization model in which a huge search space exists due to a wide range of each relevant model parameter [12]. Among these algorithms, Genetic Algorithm (GA) has been widely used due to its easy implementation and good performance in calibration and optimization [12]. Although genetic algorithm-based calibration can effectively align simulated traffic statistics with real-world observations, most existing methods remain largely descriptive. They focus on improving how realistic the simulation appears, without directly accounting for the needs of temporal forecasting models. As a result, the link between calibration accuracy and the reliability of predictions derived from digital-twin traffic time series is still not well understood [12,13].

2.3. Deep Learning Traffic-Forecasting Models

The earliest phase of forecasting relied on mathematical and statistical models [14], often referred to as parametric approaches, such as Autoregressive Integrated Moving Average (ARIMA). However, parametric approaches can achieve good performance when traffic shows regular variations, but the forecast error is obvious when the traffic shows irregular variations.

Given the extensive traffic data in contemporary ITS, advanced techniques for big data analysis are highly sought after. With the development of artificial intelligence, deep learning approaches have emerged vigorously. Traffic forecasting has been gradually based on deep learning approaches, which have become a new trend [14]. Early deep learning models employed Multi-Layer Perceptron (MLP). Researchers utilize CNN deep learning architecture to extract spatiotemporal traffic features for the speed prediction method [20]. To capture the temporal and spatial evolution of traffic flow Recurrent Neural Networks (RNNs) have been used. However, previous studies proved that RNNs faced the crucial limitation of vanishing gradient and exploding gradient problems, making them difficult to train when dealing with long-term dependencies (beyond 5–10 min lags) [14]. To address this limitation, Long Short-Term Memory (LSTM) model was implemented. Compared with conventional RNNs, LSTM networks are able to capture the features of time series within longer time spans. Therefore, the traffic forecast can achieve a better performance by using the LSTM network [14]. Nevertheless, the performance of deep temporal models is highly sensitive to the statistical consistency and temporal stability of the underlying traffic data, indicating that improvements in data calibration may play a critical role in enhancing forecasting reliability.

2.4. Synthesis and Research Gap Identification

Deep learning has shown strong potential for traffic forecasting, but its success not only depends on the model architecture, but also relies on whether simulation-generated traffic data are calibrated in a way that preserves meaningful temporal patterns for prediction. Current GA-based calibration methods mainly focus on matching the statistical properties of real traffic, leaving an important gap in calibration strategies designed to support reliable multi-step forecasting.

3. Benchmark Comparison with Prior Studies

Recent work increasingly incorporates SUMO into digital-twin and simulation-driven traffic management systems. Kušić et al. [21] present a motorway digital twin in which SUMO is closely linked to real traffic-counter data, enabling continuous adjustment of microscopic simulation parameters so the model stays aligned with actual road conditions. Their study highlights SUMO's usefulness as a high-fidelity engine for digital twins, though it does not explore calibration at the data-series level or the use of deep learning for forecasting. Similarly, Schaffland et al. [22] expand SUMO's role toward city-scale digital twins by integrating diverse sensor inputs to study commuting patterns and evaluate incentive mechanisms. The priority in their framework is simulation accuracy and behavioral analysis rather than predictive modeling or the correction of multivariate time-series.

A smaller set of studies attempts to link SUMO outputs with prediction or control tasks, but important limitations remain. Nautiyal et al. [23], for example, use SUMO to generate vehicle-count datasets for an intelligent traffic-signal management system. They train a series of machine learning models for short-term traffic prediction as part of an adaptive light-control strategy. However, their system is built on classical ML models, does not explicitly incorporate multivariate indicators and lacks mechanisms for capturing temporal dynamics over different memory horizons.

In contrast, this study positions SUMO as the core of a fully integrated, end-to-end digital-twin framework designed specifically for traffic forecasting. Virtual sensors within SUMO are first configured to generate multivariate traffic time series—such as vehicle flow and speed—structured explicitly for temporal prediction tasks. Rather than relying on conventional calibration approaches that primarily enhance macroscopic realism, we propose a forecasting-aware Genetic Algorithm calibration strategy in which simulation parameters are optimized directly with respect to multi-step prediction error. This perspective reframes calibration from a purely descriptive process toward one that prioritizes temporal learnability and predictive reliability, allowing the digital twin to generate traffic dynamics that are inherently more suitable for downstream forecasting.

To examine the predictive implications of this calibration strategy, we conduct a structured deep-learning benchmark across two representative model families: 1D-CNN, which captures local temporal patterns, and LSTM, which models long-term temporal dependencies. Both single-step and multi-step forecasting settings are evaluated, enabling a systematic analysis of how calibration quality and temporal memory capacity interact to influence prediction stability.

Finally, to assess external validity, the same forecasting and evaluation protocol is applied to the real-world PEMS-BAY traffic dataset. This enables a direct comparison between the predictive behavior observed in the calibrated digital twin and that of real traffic dynamics.

Table 1 compares related SUMO-based digital twin and traffic forecasting frameworks and highlights the contributions of the proposed framework.

Table 1. Comparison of SUMO-based digital twin and forecasting frameworks.

Feature/Capability	Kušić et al. [21]	Schaffland et al. [22]	Nautiyal et al. [23]	Proposed Framework
Uses SUMO	✓ Microscopic simulator for motorway DT	✓ City-scale DT in SUMO	✓ SUMO to generate traffic data for adaptive signals	✓ SUMO-based simulation

Table 1. *Cont.*

Feature/Capability	Kušić et al. [21]	Schaffland et al. [22]	Nautiyal et al. [23]	Proposed Framework
Digital Twin Focus	✓ Real-time synchronized digital twin of motorway	✓ Urban DT with multiple city sensors	× Primarily control/ML for TL management	✓ Simulation environment aligned with DT concept
Virtual Sensors	✓ Uses traffic counters	✓ Simulated sensors	✓ Integrates various sensor types into SUMO	✓ Vehicle counts per phase/hour; flow, speed, and occupancy from virtual detectors
Data-Level Calibration (Time-Series Rectification)	× Calibration at simulator/parameter level, not sensor time-series	× Focus on data integration, not time-series rectification	× No explicit rectification of time-series	✓ GA-based calibration of sensor time-series
Forecasting Models	× No DL forecasting; mainly DT consistency	× No explicit forecasting; focus on behavior analysis	✓ Classical ML models for vehicle-count prediction	✓ Deep learning models (MLP, 1D-CNN, LSTM)
Temporal Dependency Analysis	× Not focused on temporal model comparison	× Not focused on temporal modeling	× Uses ML models but no temporal vs non-temporal analysis	✓ Explicit comparison: non-temporal vs local vs long-term temporal models
Multi-Step Forecasting	×	×	× Short-term prediction only (reported)	✓ Single-step and multi-step horizon evaluation
Multivariate Time-Series (Flow + Speed + Occupancy)	× Primarily flow/volume from counters	× Focus on sensor availability and data fusion	× Mainly vehicle counts	✓ Joint modeling of flow, speed, and occupancy

Note: ✓ indicates that the feature is supported or implemented; × indicates that the feature is not supported or not reported.

4. Methodology

The methodological framework of the proposed digital traffic twin is structured as a sequential, end-to-end pipeline that links microscopic traffic simulation, sensor-level data generation, behavioral calibration, and time-series forecasting within a single coherent workflow. The core idea underpinning this framework is that forecasting accuracy is not determined by learning models alone, but is strongly influenced by how traffic dynamics are generated and calibrated prior to prediction. Accordingly, the methodology progresses through five clearly defined stages: microscopic simulation and data extraction, synthetic LiDAR sampling, genetic algorithm–based calibration of traffic behavior, deep sequence learning for forecasting, and performance evaluation.

To confirm that the calibration process produces traffic dynamics that are realistic and consistent with real-world behavior, an external benchmark dataset is employed. The same forecasting and evaluation protocol is applied to both the calibrated digital twin data and real-world traffic measurements, allowing the validity of the calibration to be assessed through comparative predictive performance Figure 1.

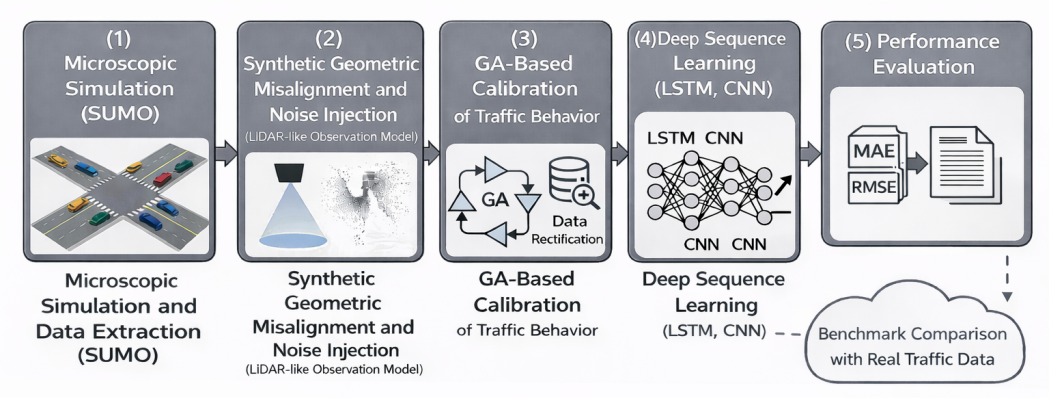


Figure 1. Schematic representation of the proposed end-to-end digital traffic twin methodology. The pipeline consists of five core stages: (1) microscopic simulation and data extraction using SUMO, (2) Synthetic Geometric Misalignment and Noise Injection (3) genetic algorithm-based calibration of traffic behavior, and (4) deep sequence learning for traffic forecasting using LSTM and CNN models. Forecasting performance is evaluated on the calibrated digital twin outputs, while an external benchmark dataset is employed for validation by applying the same forecasting and evaluation protocol to real-world traffic data.

4.1. Microscopic Simulation and Data Extraction in SUMO

Microscopic traffic dynamics are generated using the Simulation of Urban Mobility (SUMO) platform, which provides fine-grained control over individual vehicle behavior. This component employs SUMO to generate realistic vehicle trajectories around the targeted intersection near the Naif Arab University for Security Sciences (NAUSS), Figure 2. The simulation is executed with a fixed temporal resolution of $\Delta t = 0.1$ s to ensure smooth and accurate vehicle trajectories. Runtime interaction with the simulator is achieved through the Traffic Control Interface (TraCI), enabling continuous access to vehicle positions and speeds during execution.



Figure 2. Geospatial selection and validation of the study intersection within the surrounding area of Naif Arab University for Security Sciences (NAUSS) using real-world map data. The figure demonstrates the spatial alignment between the SUMO-generated road network and the corresponding OpenStreetMap representation, confirming the structural and geographical consistency of the selected intersection and its immediate campus vicinity for traffic digital twin implementation. The OpenStreetMap view includes the university name displayed in Arabic, corresponding to Naif Arab University for Security Sciences (NAUSS).

A fixed network node (ID: 9582306895) is selected as the reference point for a virtual roadside LiDAR sensor Figure 3. This node defines the origin of a local sensor-centric

coordinate system. For each vehicle with global SUMO coordinates $(x_{\text{sim}}, y_{\text{sim}})$ and sensor reference coordinates $(x_{\text{ref}}, y_{\text{ref}})$, local offsets are computed as:

$$dx = x_{\text{sim}} - x_{\text{ref}}, \qquad dy = y_{\text{sim}} - y_{\text{ref}}.$$

To focus on traffic conditions relevant to the target intersection, only vehicles located within a circular region of interest (ROI) of radius $R = 250$ meters are retained:

$$\sqrt{dx^2 + dy^2} \leq R.$$

This spatial constraint captures all incoming and outgoing approaches of the intersection while excluding distant traffic that does not meaningfully influence local traffic states.

Although the SUMO simulation operates at a 0.1 s resolution, vehicle data are recorded according to the sampling frequency of the virtual LiDAR sensor. Specifically, observations are logged every 0.2 s, corresponding to a sampling rate of 5 Hz. At each LiDAR timestamp, all vehicles within the ROI are iterated, and their local coordinates and instantaneous speeds are recorded. This process produces a detailed vehicle-level dataset that reflects the underlying microscopic traffic dynamics.

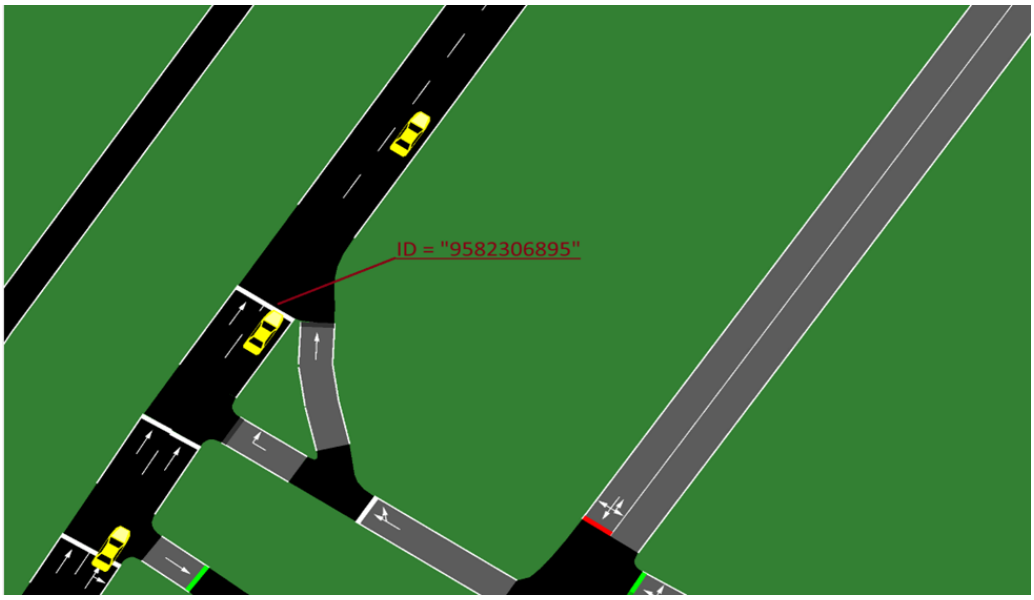


Figure 3. Microscopic traffic network preparation in SUMO, illustrating the geometric modeling of road segments, lane configurations, connectors, and turning movements. The red and green lines correspond to the default SUMO visualization. The figure highlights the preprocessing and validation of the target intersection (ID = 9582306895) to ensure realistic vehicle interactions and traffic flow behavior prior to simulation and data generation.

4.2. Synthetic Geometric Misalignment and Noise Injection

To emulate realistic sensing imperfections in a controlled manner, the local SUMO coordinates (dx, dy) are converted into a LiDAR-like observation stream via geometric misalignment and additive noise injection. The transformation remains fully traceable to the SUMO reference frame, enabling rigorous calibration analysis without implying full physical LiDAR simulation.

The geometric misalignment is defined by the parameter vector:

$$\mathbf{p}_{\text{mis}} = [s_x, s_y, \theta, t_x, t_y],$$

where s_x and s_y represent anisotropic scaling factors, θ denotes a planar rotation angle, and (t_x, t_y) correspond to translational offsets. In this study, these parameters are fixed to:

$$s_x = 1.02, \quad s_y = 0.98, \quad \theta = 12^\circ, \quad t_x = 10\,\text{m}, \quad t_y = -5\,\text{m}.$$

Given the SUMO local offsets (dx, dy) relative to the reference node, the LiDAR-like coordinates are computed by first compensating for translation and scaling:

$$x_0 = \frac{dx - t_x}{s_x}, \qquad y_0 = \frac{dy - t_y}{s_y},$$

followed by a planar rotation:

$$x_{\text{lidar}} = x_0 \cos\theta + y_0 \sin\theta, \qquad y_{\text{lidar}} = -x_0 \sin\theta + y_0 \cos\theta.$$

To account for measurement uncertainty, zero-mean Gaussian noise with standard deviation $\sigma_{\text{noise}} = 0.5$ m is added independently to each coordinate:

$$x_{\text{obs}} = x_{\text{lidar}} + \mathcal{N}(0, \sigma^2_{\text{noise}}), \qquad y_{\text{obs}} = y_{\text{lidar}} + \mathcal{N}(0, \sigma^2_{\text{noise}}).$$

The resulting $(x_{\text{obs}}, y_{\text{obs}})$ represent LiDAR-like perturbed measurements that incorporate systematic geometric bias and stochastic noise while remaining mathematically linked to the original SUMO vehicle positions.

4.3. GA-Based Calibration of Traffic Behavior

The Genetic Algorithm (GA) operates directly in the traffic behavior parameter space, aiming to identify parameter configurations that yield temporally stable and predictable macroscopic traffic signals.

Traffic behavior is parameterized by the vector:

$$\boldsymbol{\theta} = [\tau, a, d, \sigma, q, plc],$$

where τ denotes driver reaction time, a and d represent acceleration and deceleration capabilities, σ controls stochasticity in driver behavior, q scales global traffic demand, and plc governs lane-change cooperativeness.

Each GA individual corresponds to a candidate parameter vector $\boldsymbol{\theta}$. For a given individual, the pipeline executes a full SUMO simulation, generates synthetic LiDAR observations, and aggregates the resulting vehicle-level data into a macroscopic time series:

$$X(t) = [N(t), \bar{v}(t)],$$

where $N(t)$ denotes the number of vehicles within the region of interest (ROI) at time t, and $\bar{v}(t)$ represents their average speed. To ensure temporal continuity, missing values in the speed signal are handled using linear interpolation followed by forward and backward filling, resulting in a uniformly sampled time series.

To guide the GA search toward traffic dynamics that are not only physically plausible but also temporally learnable, a forecasting-aware fitness function is employed. The aggregated time series is split chronologically into training (80%) and testing (20%) segments and standardized using statistics computed exclusively from the training portion:

$$X_{\text{norm}}(t) = \frac{X(t) - \mu_{\text{train}}}{\sigma_{\text{train}}}.$$

Temporal predictability is assessed through a lightweight multi-step forecasting task embedded within the GA loop. A proxy Long Short-Term Memory (LSTM) model is used to evaluate each candidate solution. The proxy LSTM is trained using a fixed input window of $W = 120$ time steps and a multi-step forecasting horizon of $H = 20$ steps. To maintain computational efficiency and avoid overfitting during optimization, the proxy model is trained for a limited number of epochs (10) with a batch size of 64 and a fixed learning rate of 10^{-3}. The mean squared error (MSE) computed on the held-out test segment is adopted as the GA fitness measure.

After GA, the optimal parameter vector $\boldsymbol{\theta}^*$ is used to generate a single final calibrated dataset, denoted as $X_{\text{fa_timeseries}}$. This dataset constitutes the sole output of the calibration stage and serves as the input to the final forecasting experiments described in the subsequent sections.

4.4. Deep Sequence Learning for Traffic Forecasting

Forecasting is performed exclusively on the calibrated time series $X_{\text{fa_timeseries}}$. Two forecasting tasks are considered: single-step forecasting, where the next time step is predicted, and multi-step forecasting, where a sequence of future states over a fixed horizon is estimated.

4.4.1. Sequence Construction

The time series is split chronologically into training (80%) and testing (20%) subsets. Standardization is applied using statistics computed from the training set only:

$$X_{\text{norm}}(t) = \frac{X(t) - \mu_{\text{train}}}{\sigma_{\text{train}}}.$$

Input sequences are constructed using a sliding window of length $W = 120$ time steps. In the single-step setting, the target corresponds to the next time step. In the multi-step setting, the target is a sequence of $H = 20$ future time steps.

4.4.2. Forecasting Models and Training Configuration

Two deep learning architectures are evaluated under identical data and training conditions to assess the impact of temporal modeling capacity:

- **Convolutional Neural Network (CNN):** The CNN applies one-dimensional convolutional filters along the temporal axis to capture local temporal patterns, such as short-term fluctuations and recurring variations. This architecture efficiently extracts localized features but has limited capacity to model long-range dependencies.
- **Long Short-Term Memory (LSTM):** The LSTM is designed to model sequential dependencies explicitly. Stacked LSTM layers with a hidden dimension of 64 are used to capture longer-term temporal relationships, allowing relevant information to be retained across multiple time steps.

All models are trained using the Adam optimizer and the mean squared error (MSE) loss:

$$\text{MSE} = \frac{1}{M}\sum_{i=1}^{M}(p_i - t_i)^2.$$

Training is performed for a fixed number of epochs (100) with a batch size of 64. These values are selected to balance convergence stability and computational efficiency. A learning rate of 10^{-3} is used, which provides stable training behavior across all evaluated architectures.

4.5. Performance Evaluation Metrics

To quantitatively assess forecasting performance, two standard error metrics are computed in the original (denormalized) physical scale of the target variables for both Vehicle Count and Average Speed, across single-step and multi-step forecasting horizons.

Model training is performed on z-score normalized data, where the mean (μ_{train}) and standard deviation (σ_{train}) are computed exclusively from the training set to avoid data leakage. During evaluation, both predicted and ground-truth values are inverse-transformed to the original physical scale prior to metric computation according to:

$$x_{\text{original}} = x_{\text{normalized}} \cdot \sigma_{\text{train}} + \mu_{\text{train}}$$

Therefore, all reported MAE and RMSE values are expressed in real-world units (vehicles for count and meters per second for speed).

1. **Mean Absolute Error (MAE):** MAE measures the average magnitude of prediction errors while preserving the original physical units of the target variable:

$$\text{MAE} = \frac{1}{M}\sum_{i=1}^{M}|p_i - t_i|$$

2. **Root Mean Square Error (RMSE):** RMSE represents the square root of the mean squared prediction error and is more sensitive to large deviations, making it a robust indicator of prediction stability:

$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(p_i - t_i)^2}$$

Here, M denotes the total number of prediction samples, p_i is the predicted value, and t_i is the corresponding ground-truth value.

5. Results

This section presents an empirical evaluation of the proposed digital twin framework. We first analyze the impact of the forecasting-aware GA calibration on traffic signal stability, and then we assess the predictive performance of the deep learning models under both single-step and multi-step forecasting settings.

5.1. Effect of GA Calibration

This section presents the quantitative effect of the Genetic Algorithm (GA) calibration by jointly analyzing changes in traffic behavior parameters and their impact on macroscopic traffic signals. Two datasets are compared: the standard uncalibrated dataset (STD), generated directly from the microscopic simulation, and the forecasting-aware calibrated dataset (FA), obtained after GA optimization.

At the behavioral level, GA calibration produces a clear and physically meaningful shift in traffic parameters. Table 2 reports the optimized parameter values. Compared to the STD configuration, the calibrated FA parameters exhibit a substantial reduction in stochastic driving noise, with the randomness parameter σ decreasing from 0.895 to 0.128, corresponding to an 85.7% reduction. This reduction directly suppresses unrealistic fluctuations in vehicle behavior. In parallel, the reaction time τ increases from 0.50 to 0.64, and the acceleration parameter a increases from 0.97 to 2.78, indicating smoother and more realistic car-following dynamics. The demand scaling factor q increases from 0.67 to 1.11, restoring balanced traffic loading, while cooperative lane-changing probability plc increases from 0.35 to 0.55, improving interaction consistency among vehicles.

Table 2. Traffic behavior parameters before (STD) and after forecasting-aware GA calibration (FA). Values are rounded for clarity.

Parameter	STD	FA
Reaction time τ [s]	0.500	0.638
Acceleration a [m/s^2]	0.971	2.782
Deceleration d [m/s^2]	4.249	4.829
Randomness σ	0.895	0.128
Demand scaling q	0.675	1.108
Lane-change cooperation plc	0.354	0.550

These parameter-level corrections propagate directly to macroscopic traffic signals. Table 3 summarizes the Root Mean Square Error (RMSE) of vehicle count and average speed. For vehicle count, the RMSE is reduced from 43.41 vehicles in the STD dataset to 21.41 vehicles after calibration, representing a reduction of approximately 50.7%. For average speed, the RMSE decreases from 1.39 m/s to 0.83 m/s, corresponding to a 40.5% improvement. These reductions quantitatively confirm that GA calibration significantly stabilizes traffic dynamics.

Table 3. Macroscopic signal-level RMSE comparison between the standard (STD) dataset and the forecasting-aware calibrated (FA) dataset. The reported values measure direct signal discrepancies prior to predictive modeling. GA calibration reduces vehicle count RMSE by approximately 50.7% (43.41 $\rightarrow$ 21.41) and average speed RMSE by approximately 40.5% (1.39 $\rightarrow$ 0.83 m/s), indicating substantial improvement in macroscopic signal consistency before forecasting.

Metric (Signal-Level)	STD	FA
RMSE (Vehicle Count)	43.41	21.41
RMSE (Average Speed) [m/s]	1.39	0.83

The statistical impact of calibration is further illustrated through distributional analysis. Figure 4 shows that the STD vehicle count distribution is highly dispersed with long tails, indicating unstable vehicle accumulation and frequent extreme deviations. After GA calibration, the FA distribution becomes markedly more compact, with reduced variance and a higher concentration in the medium-to-high count range, reflecting improved spatial consistency.

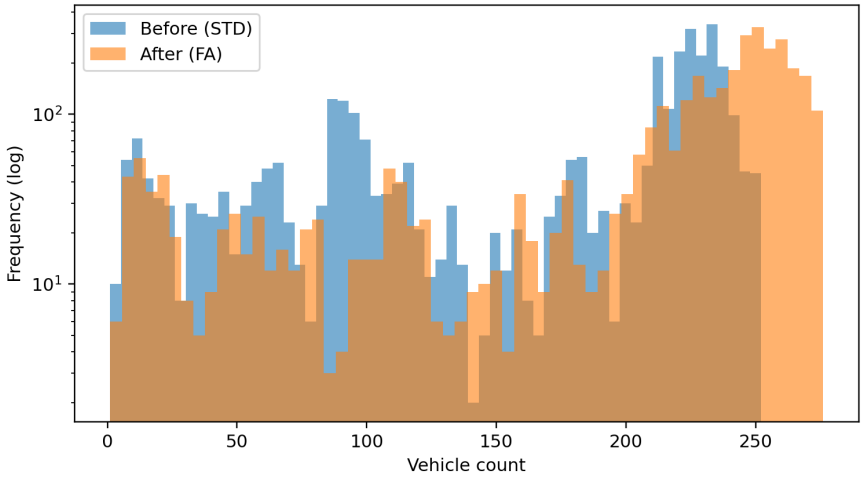


Figure 4. Distribution of vehicle count before (STD) and after GA calibration (FA). The FA distribution exhibits reduced dispersion and improved stability of vehicle accumulation.

A similar pattern is observed in the traffic density distribution (Figure 5). The STD data show strong dispersion, particularly at low-to-medium density levels, whereas the FA distribution is smoother and more concentrated. This behavior indicates that GA calibration suppresses noise-induced density fluctuations while preserving the overall structure of traffic flow.

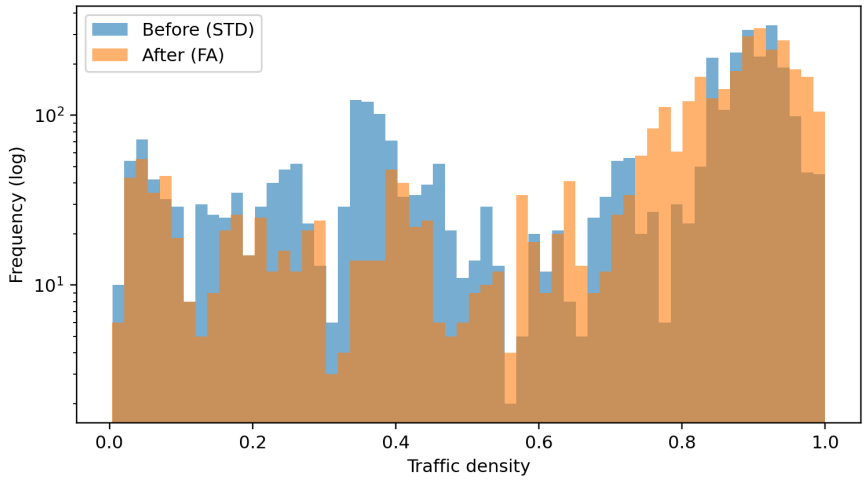


Figure 5. Distribution of estimated traffic density before (STD) and after forecasting-aware Genetic Algorithm (GA) calibration (FA). The calibrated FA distribution exhibits a narrower spread and reduced tail dispersion compared to the STD case, indicating lower variability and improved spatial consistency of density estimation after calibration.

Finally, Figure 6 illustrates the effect of calibration on average speed. While speed signals are inherently smoother than vehicle count, the STD distribution still exhibits noticeable variability. After calibration, the FA distribution shows reduced variance and fewer outliers, indicating enhanced temporal coherence of vehicle motion without altering realistic speed ranges.

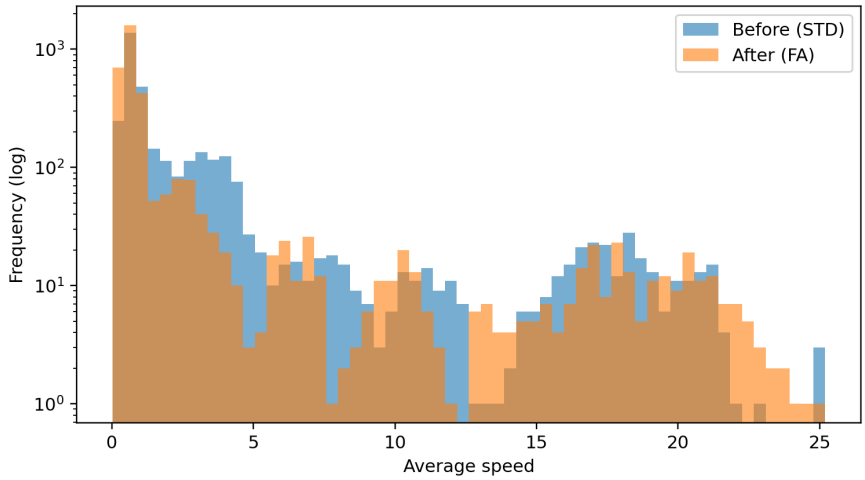


Figure 6. Distribution of average speed before (STD) and after forecasting-aware Genetic Algorithm (GA) calibration (FA). GA calibration reduces noise-induced variability and suppresses extreme speed deviations, resulting in a more compact and temporally consistent speed distribution.

These results demonstrate that GA calibration produces consistent improvements across behavioral parameters, macroscopic error metrics, and statistical distributions. By reducing randomness by over 85% and macroscopic errors by up to 50%, the calibra-

tion process transforms the simulated traffic data into a stable and physically coherent representation, establishing a reliable foundation for subsequent forecasting experiments.

5.2. Single-Step Forecasting Performance

The Single-Step forecasting task evaluates the stability of the models in predicting the immediate next traffic state $(t + 1)$. This setting directly reflects the ability of each architecture to capture short-term temporal dependencies in macroscopic traffic signals. Table 4 summarizes the quantitative performance of the CNN and LSTM models using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for both Vehicle Count and Average Speed.

Table 4. Single-Step forecasting results $(H = 1)$ for Vehicle Count and Average Speed.

Model	Count MAE	Count RMSE	Speed MAE (m/s)	Speed RMSE (m/s)
CNN	2.7435	3.3353	0.1173	0.1424
LSTM	0.6534	1.1373	0.0257	0.0339

The results clearly establish a superior performance for the LSTM architecture in the short-term prediction context (Figures 7 and 8). For the Vehicle Count feature, LSTM achieves the lowest error rates with an RMSE of 1.1373, outperforming the CNN model (RMSE = 3.3353) by approximately 65.9%. This substantial reduction highlights the effectiveness of recurrent temporal modeling in capturing short-term fluctuations in discrete traffic volume.

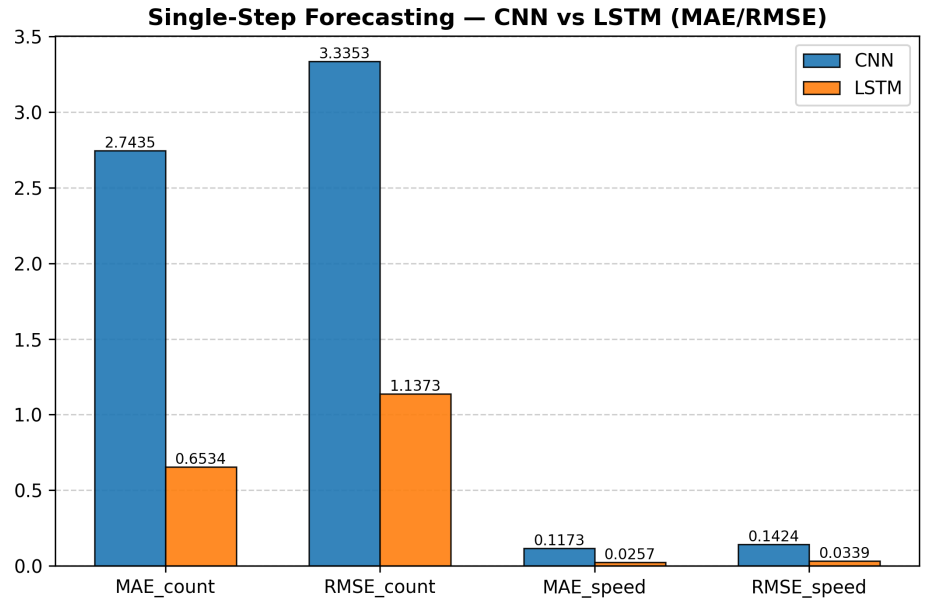


Figure 7. Single-Step forecasting performance comparison $(H = 1)$ between CNN and LSTM models for Vehicle Count and Average Speed. For vehicle count, the LSTM model achieves an RMSE of 1.1373, reducing the error by approximately 65.9% compared to CNN (RMSE = 3.3353). For average speed, LSTM further demonstrates superior accuracy with an RMSE of 0.0339 m/s, corresponding to a 76.2% improvement over CNN (RMSE = 0.1424 m/s). These results indicate that recurrent temporal modeling enables LSTM to more effectively capture short-term traffic dynamics than convolution-based architectures.

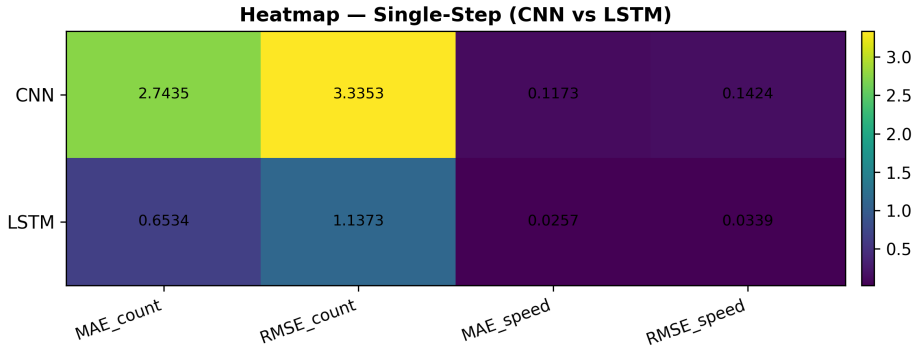


Figure 8. Heatmap visualization of Single-Step forecasting errors ($H = 1$) for CNN and LSTM models. The color intensity represents the magnitude of MAE and RMSE for both Vehicle Count and Average Speed. Lower error intensities are consistently observed for the LSTM model across all evaluated metrics, indicating more accurate and stable short-term predictions compared to the CNN model.

A similar trend is observed for the Average Speed feature. The LSTM model records an RMSE of 0.0339 m/s, representing a 76.2% improvement compared to the CNN (RMSE = 0.1424 m/s). The overall error magnitudes for speed prediction remain lower than those observed for vehicle count, which is consistent with the smoother temporal evolution of speed compared to the inherently more volatile count signal.

5.3. Multi-Step Forecasting Performance

The Multi-Step forecasting task evaluates the models' capability to predict an extended sequence of future traffic states over a fixed horizon ($H = 20$). Unlike single-step prediction, this setting is more demanding because forecasting errors accumulate across the prediction window. Table 5 reports the overall Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) for the CNN and LSTM models on both Vehicle Count and Average Speed.

Table 5. Multi-Step forecasting results ($H = 20$) for Vehicle Count and Average Speed.

Model	Count MAE	Count RMSE	Speed MAE (m/s)	Speed RMSE (m/s)
CNN	3.3196	3.9794	0.1487	0.1928
LSTM	0.9663	1.2680	0.0523	0.0697

The results show that LSTM maintains a substantially stronger performance under the multi-horizon forecasting setting (Figures 9 and 10). For the Vehicle Count feature, LSTM reduces RMSE from 3.9794 (CNN) to 1.2680, corresponding to an improvement of approximately 68.1%. A similar advantage is observed in MAE, where LSTM lowers the error from 3.3196 to 0.9663 (about 70.9% reduction). These margins indicate that the LSTM model provides more stable cumulative predictions across the full 20-step horizon.

For Average Speed, LSTM also yields consistently lower errors. The RMSE decreases from 0.1928 m/s (CNN) to 0.0697 m/s (about 63.8% improvement), while MAE drops from 0.1487 m/s to 0.0523 m/s (approximately 64.8% reduction). Overall, the multi-step results confirm that LSTM produces more accurate forecasts across both traffic indicators when prediction errors are aggregated over longer horizons.

5.4. Benchmark Evaluation on Real-World Traffic Data (PEMS-BAY)

To further evaluate the robustness and external validity of the proposed forecasting framework, an additional benchmark experiment was conducted using the real-world PEMS-BAY traffic dataset. Unlike the simulated digital twin data, which provide both

vehicle count and speed measurements, the PEMS-BAY dataset contains only traffic speed observations collected from highway loop detectors. Consequently, this benchmark evaluation focuses exclusively on average speed forecasting, enabling an assessment of model performance under real traffic dynamics without reliance on simulated counts.

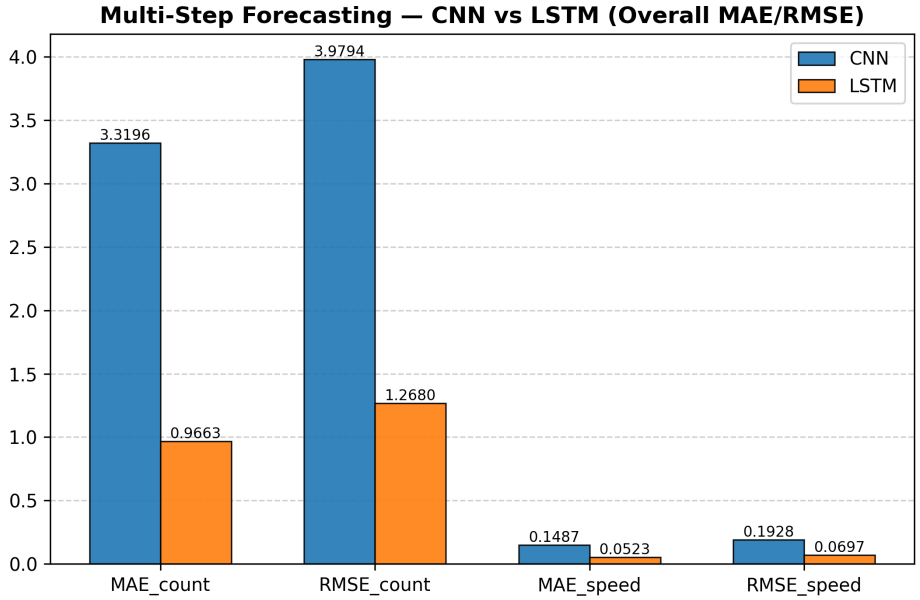


Figure 9. Multi-Step forecasting performance comparison ($H = 20$) between CNN and LSTM models using overall MAE and RMSE for Vehicle Count and Average Speed. For vehicle count, LSTM achieves a lower RMSE of 1.2680 compared to CNN (3.9794), corresponding to an improvement of about 68.1%. For average speed, LSTM records an RMSE of 0.0697 m/s, improving upon CNN (0.1928 m/s) by approximately 63.8%.

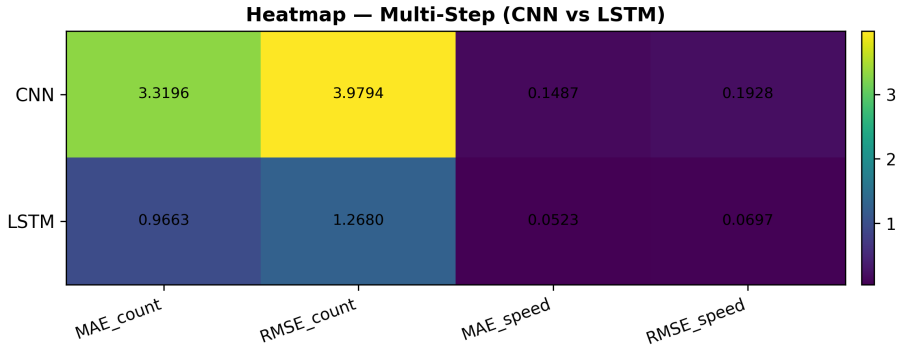


Figure 10. Heatmap visualization of Multi-Step forecasting errors ($H = 20$) for CNN and LSTM models. The color intensity represents the magnitude of MAE and RMSE for both Vehicle Count and Average Speed. Lower error intensities are consistently observed for the LSTM model across all metrics, indicating improved cumulative forecasting accuracy over the 20-step horizon.

To ensure methodological consistency, the same forecasting architectures (LSTM and CNN), preprocessing procedures, sliding window construction, and evaluation metrics employed in the digital twin experiments were retained. Both single-step ($H = 1$) and multi-step ($H = 20$) forecasting settings were evaluated in order to examine short-term predictive accuracy as well as error accumulation over extended prediction horizons.

Table 6 reports the overall Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) obtained for average speed prediction on the PEMS-BAY dataset.

Table 6. Benchmark forecasting results on the PEMS-BAY dataset (average speed only).

Model/ Horizon	MAE (m/s)	RMSE (m/s)
CNN (Single-Step)	0.2360	0.6051
LSTM (Single-Step)	0.1321	0.4760
CNN (Multi-Step)	1.1517	1.8810
LSTM (Multi-Step)	0.6876	1.2442

In the single-step forecasting task, the LSTM model demonstrates clearly superior performance compared to the CNN architecture. Specifically, LSTM reduces the MAE by approximately 44.0% and the RMSE by 21.3% relative to CNN, indicating a more accurate representation of short-term temporal dependencies in real-world traffic speed signals.

For the multi-step forecasting task, prediction errors increase for both models, reflecting the inherent difficulty of longer-horizon forecasting under real traffic variability. Nevertheless, LSTM consistently outperforms CNN, achieving reductions of approximately 40.3% in MAE and 33.9% in RMSE. These results indicate that recurrent memory mechanisms not only enhance short-term accuracy but also mitigate error accumulation over extended prediction horizons.

Figures 11 and 12 present heatmap visualizations of MAE and RMSE for single-step and multi-step forecasting, respectively. In both cases, lower error intensities are consistently observed for the LSTM model, confirming its superior predictive stability.

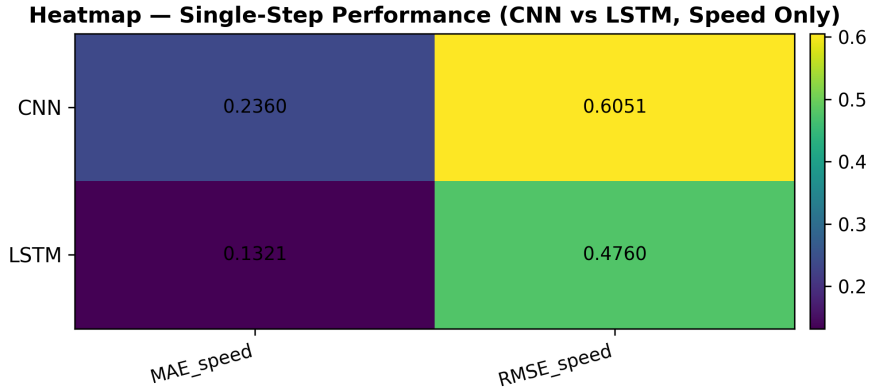


Figure 11. Heatmap comparison of single-step forecasting performance on the PEMS-BAY dataset (average speed only). Lower error values for LSTM indicate improved short-term predictive accuracy compared to CNN.

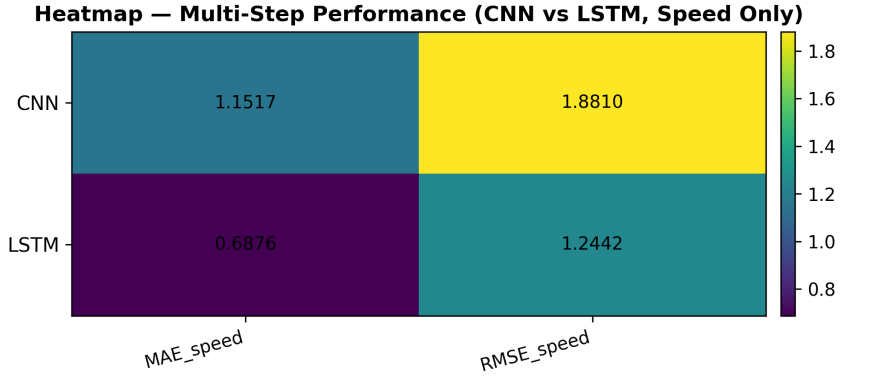


Figure 12. Heatmap comparison of multi-step forecasting performance on the PEMS-BAY dataset (average speed only). LSTM exhibits reduced error accumulation and improved stability across extended prediction horizons.

Complementary bar-chart comparisons are shown in Figures 13 and 14, further illustrating the consistent performance gap between CNN and LSTM across both forecasting horizons.

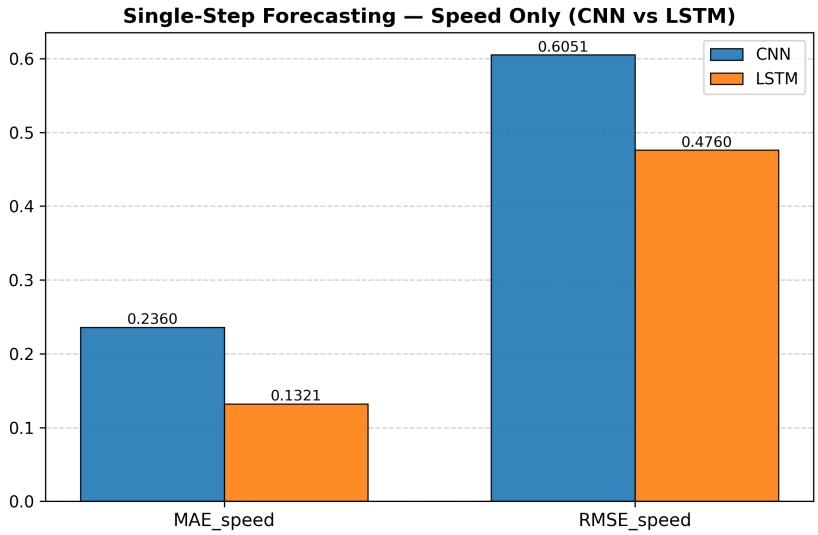


Figure 13. Single-step forecasting performance comparison on the PEMS-BAY dataset (average speed only). LSTM achieves lower MAE and RMSE than CNN, highlighting the benefits of recurrent temporal modeling.

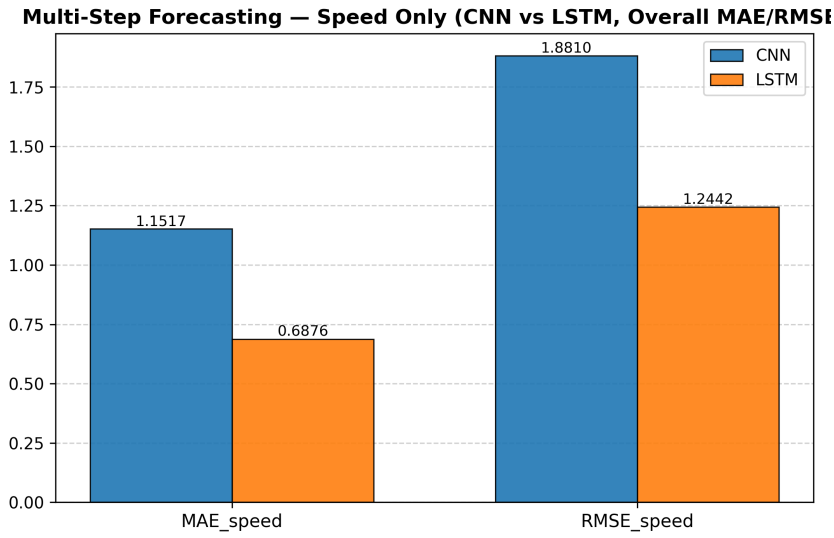


Figure 14. Multi-step forecasting performance comparison on the PEMS-BAY dataset (average speed only). The widening error gap emphasizes LSTM's robustness under longer forecasting horizons.

6. Discussion

The results of this study provide clear and consistent evidence that Genetic Algorithm (GA)-based calibration plays a central role in improving both the physical realism of simulated traffic data and its suitability for downstream forecasting tasks. Prior to calibration, the synthetic traffic signals exhibited substantial macroscopic discrepancies, reflected by high Root Mean Square Error (RMSE) values for both vehicle count and average speed. After calibration, vehicle count RMSE was reduced from 43.41 to 21.41, corresponding to an improvement of approximately 50.7%, while average speed RMSE decreased from 1.39 m/s to 0.83 m/s, representing an improvement of approximately 40.3%. These reductions confirm that the calibration process effectively corrects systematic distortions introduced

during synthetic data generation, resulting in traffic signals that are both statistically more stable and physically more coherent.

The impact of GA calibration extends beyond macroscopic error reduction. Statistical analyses of traffic density and speed distributions show that the calibrated data exhibit reduced variance and fewer extreme deviations, while preserving the overall temporal structure of traffic flow. This behavior is critical for learning-based models, as excessive noise can obscure meaningful temporal patterns, whereas overly smoothed data may lead to artificially optimistic forecasting performance. The calibrated dataset achieves a balance between stability and realism, providing a reliable foundation for predictive modeling.

These improvements in data quality are directly reflected in the forecasting results obtained on the calibrated digital twin dataset. In the single-step forecasting task, the LSTM model substantially outperformed the CNN model. For average speed prediction, LSTM achieved an RMSE of 0.0339 compared to 0.1424 for CNN, corresponding to a relative error reduction of approximately 76.2%. A similar pattern is observed in the multi-step forecasting task, where LSTM reduced RMSE from 0.1928 (CNN) to 0.0697, an improvement of approximately 63.8%. These results indicate that recurrent architectures benefit strongly from the improved temporal coherence introduced by GA calibration, particularly when forecasting over extended horizons where error accumulation becomes significant.

To assess whether these improvements are limited to the simulated environment or reflect more generalizable traffic dynamics, a benchmark evaluation was conducted using the real-world PEMS-BAY dataset. Unlike the simulated data, PEMS-BAY provides only speed measurements, allowing an isolated evaluation of speed forecasting under real traffic conditions. Although absolute error values increase on the benchmark dataset—as expected due to higher variability and uncontrolled noise—the relative performance between models remains consistent. In the single-step benchmark task, LSTM achieved an RMSE of 0.4760, outperforming CNN's RMSE of 0.6051 by approximately 21.3%. In the multi-step setting, LSTM reduced RMSE from 1.8810 to 1.2442, corresponding to an improvement of approximately 33.9%.

A consolidated comparison between the calibrated digital twin dataset and the PEMS-BAY benchmark is presented in Tables 7 and 8. The table highlights three key observations. First, LSTM consistently outperforms CNN across both datasets and forecasting horizons. Second, the transition from single-step to multi-step forecasting leads to predictable error growth in both datasets, reflecting increasing temporal uncertainty. Third, the relative scaling of errors between simulated and real-world data remains comparable, suggesting that the calibrated digital twin captures temporal dynamics that are structurally similar to those observed in real traffic systems.

Table 7. Multi-step forecasting performance ($H = 20$) for average speed on Our Dataset (Calibrated) versus the PEMS-BAY benchmark. The best RMSE within each dataset block is marked.

Dataset	Model	MAE Speed (m/s)	RMSE Speed (m/s)
Our Dataset (Calibrated)	LSTM (best)	0.0523	**0.0697**
	CNN	0.1487	0.1928
PEMS-BAY Benchmark	LSTM (best)	0.6876	**1.2442**
	CNN	1.1517	1.8810

Taken together, these findings demonstrate that GA-based calibration is not merely a preprocessing step, but a fundamental enabler of reliable traffic forecasting within digital twin frameworks. By reducing macroscopic errors, stabilizing statistical distributions, and enhancing temporal predictability, the calibration process produces simulated traffic data

whose forecasting behavior closely mirrors that of real-world traffic. The strong alignment between results obtained on the calibrated digital twin and the PEMS-BAY benchmark provides compelling evidence that the proposed methodology yields traffic representations that are both internally consistent and externally valid, supporting their use in predictive traffic management and decision-support applications.

Table 8. Single-step forecasting performance ($H = 1$) for average speed on Our Dataset (Calibrated) versus the PEMS-BAY benchmark. The best RMSE within each dataset block is marked.

Dataset	Model	MAE Speed (m/s)	RMSE Speed (m/s)
Our Dataset (Calibrated)	LSTM (best)	0.0257	**0.0339**
	CNN	0.1173	0.1424
PEMS-BAY Benchmark	LSTM (best)	0.1321	**0.4760**
	CNN	0.2360	0.6051

7. Conclusions

This study presented a forecasting-aware digital traffic twin framework that integrates microscopic SUMO simulation, controlled sensor-level observation modeling, Genetic Algorithm (GA)-based behavioral calibration, and deep temporal forecasting within a unified end-to-end pipeline. Rather than treating calibration as a purely descriptive process aimed at matching macroscopic traffic statistics, the proposed approach reformulates calibration as a predictive optimization task. By embedding a lightweight multi-step forecasting model inside the GA loop, simulation parameters are optimized with respect to temporal learnability and forecasting stability, thereby directly linking simulation behavior to downstream predictive performance.

The empirical results demonstrate that forecasting-aware calibration produces clear and measurable improvements in macroscopic traffic signals. Vehicle count RMSE was reduced from 43.41 to 21.41, corresponding to an improvement of approximately 50.7%, while average speed RMSE decreased from 1.39 m/s to 0.83 m/s, representing an improvement of approximately 40.3%. Beyond these quantitative gains, the calibrated dataset exhibited smoother statistical distributions and reduced noise-induced variability, resulting in more temporally stable and physically coherent traffic dynamics. These structural improvements translated directly into enhanced forecasting performance, particularly for recurrent architectures capable of modeling long-term dependencies.

Across both single-step and multi-step forecasting settings, the LSTM model consistently outperformed convolutional alternatives, highlighting the importance of long-term temporal memory when predicting traffic behavior over extended horizons. Importantly, the external benchmark evaluation using the real-world PEMS-BAY dataset confirmed that the predictive patterns observed in the calibrated digital twin remain structurally consistent with real traffic dynamics, supporting the external validity of the proposed methodology.

The findings indicate that reliable traffic forecasting in digital twin environments depends not only on the selection of deep learning architecture, but also on how simulation data are generated, structured, and calibrated prior to modeling. By explicitly linking calibration with forecasting objectives, this work establishes a methodological bridge between traffic simulation and predictive intelligence, contributing toward more stable, robust, and practically deployable traffic digital twin systems.

Author Contributions: Conceptualization, Z.A. and T.A.A.; methodology, Z.A.and T.A.A.; software, Z.A.; validation, Z.A. and T.A.A.; formal analysis, Z.A.; investigation, Z.A.; data curation, Z.A.; writing—original draft preparation, Z.A.and T.A.A.; writing—review and editing, T.A.A., F.A.E., I.I.

and T.D.; supervision, T.A.A.; funding acquisition, T.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Naif Arab University for Security Sciences under grant No. NAUSS-25-R01.

Data Availability Statement: The simulation-based traffic time-series data used in this study were generated using the SUMO microscopic traffic simulator under the specific network configuration and calibrated parameter settings described in the Methodology section. Because these datasets are scenario-specific and were produced within a controlled simulation environment, they are not publicly archived but are available from the corresponding author upon reasonable request. For external validation, we used the publicly available PEMS-BAY traffic dataset (traffic speed measurements), which was accessed via Kaggle. The PEMS-BAY dataset is openly accessible for research purposes through the Kaggle platform.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. SaudiPedia. Transport in Saudi Arabia. SaudiPedia 2024. Available online: https://saudipedia.com/en/transport-in-saudi-arabia (accessed on 21 February 2026).
2. Wu, D.; Zheng, A.; Yu, W.; Cao, H.; Ling, Q.; Liu, J.; Zhou, D. Digital twin technology in transportation infrastructure: A comprehensive survey of current applications, challenges, and future directions. *Appl. Sci.* **2025**, *15*, 1911. [CrossRef]
3. Kang, J.S.; Chung, K.; Hong, E.J. Multimedia knowledge-based bridge health monitoring using digital twin. *Multimed. Tools Appl.* **2021**, *80*, 34609–34624. [CrossRef]
4. Mazzetto, S. A review of urban digital twins integration, challenges, and future directions in smart city development. *Sustainability* **2024**, *16*, 8337. [CrossRef]
5. Grieves, M.; Vickers, J. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 85–113.
6. Krajzewicz, D.; Erdmann, J.; Behrisch, M.; Bieker, L. Recent development and applications of SUMO-Simulation of Urban MObility. *Int. J. Adv. Syst. Meas.* **2012**, *5*, 128–138.
7. Harrou, F.; Zeroual, A.; Kadri, F.; Sun, Y. Enhancing road traffic flow prediction with improved deep learning using wavelet transforms. *Results Eng.* **2024**, *23*, 102342. [CrossRef]
8. Wen, Y.; Xu, P.; Li, Z.; Xu, W.; Wang, X. RPConvformer: A novel Transformer-based deep neural networks for traffic flow prediction. *Expert Syst. Appl.* **2023**, *218*, 119587. [CrossRef]
9. Darwish, T.; Abu Bakar, K. Traffic density estimation in vehicular ad hoc networks: A review. *Ad Hoc Netw.* **2015**, *24*, 337–351. [CrossRef]
10. Afandizadeh, S.; Abdolahi, S.; Mirzahossein, H. Deep learning algorithms for traffic forecasting: A comprehensive review and comparison with classical ones. *J. Adv. Transp.* **2024**, *2024*, 9981657. [CrossRef]
11. Liu, R.; Shin, S.Y. A Review of Traffic Flow Prediction Methods in Intelligent Transportation System Construction. *Appl. Sci.* **2025**, *15*, 3866. [CrossRef]
12. Yu, M.; Fan, W.D. Calibration of microscopic traffic simulation models using metaheuristic algorithms. *Int. J. Transp. Sci. Technol.* **2017**, *6*, 63–77. [CrossRef]
13. Chiappone, S.; Giuffrè, O.; Granà, A.; Mauro, R.; Sferlazza, A. Traffic simulation models calibration using speed–density relationship: An automated procedure based on genetic algorithm. *Expert Syst. Appl.* **2016**, *44*, 147–155. [CrossRef]
14. Zhao, Z.; Chen, W.; Wu, X.; Chen, P.C.; Liu, J. LSTM network: A deep learning approach for short-term traffic forecast. *IET Intell. Transp. Syst.* **2017**, *11*, 68–75. [CrossRef]
15. Agrawal, D.; Vinod, D.; Rastogi, J. A SUMO Based Simulation Framework for Intelligent Traffic Management System. *SSRN Electron. J.* **2024**. [CrossRef]
16. Pechinger, M.; Lindner, J. Sumonity: Bridging sumo and unity for enhanced traffic simulation experiences. *Sumo Conf. Proc.* **2024**, *5*, 163–177. [CrossRef]
17. Manglano-Redondo, P.; Paricio-Garcia, A.; Lopez-Carmona, M.A. Spatio-Temporal AI Modeling for Urban Traffic Calibration: A SUMO-Based Approach. *Sumo Conf. Proc.* **2025**, *6*, 33–50. [CrossRef]
18. Punzo, V.; Ciuffo, B.; Montanino, M. Can results of car-following model calibration based on trajectory data be trusted? *Transp. Res. Rec.* **2012**, *2315*, 11–24. [CrossRef]
19. Afshari, A.; Lee, J.; Besenski, D.; Dimitrijevic, B.; Spasovic, L. Calibrating Microscopic Traffic Simulation Model Using Connected Vehicle Data and Genetic Algorithm. *Appl. Sci.* **2025**, *15*, 1496. [CrossRef]

20. Wang, K.; Ma, C.; Qiao, Y.; Lu, X.; Hao, W.; Dong, S. A hybrid deep learning model with 1DCNN-LSTM-Attention networks for short-term traffic flow prediction. *Phys. A Stat. Mech. Its Appl.* **2021**, *583*, 126293. [CrossRef]
21. Kušić, K.; Schumann, R.; Ivanjko, E. A digital twin in transportation: Real-time synergy of traffic data streams and simulation for virtualizing motorway dynamics. *Adv. Eng. Inform.* **2023**, *55*, 101858. [CrossRef]
22. Schaffland, A.; Nelson, J.; Schöning, J. Simulating Traffic Networks: Driving SUMO Towards Digital Twins. *Sumo Conf. Proc.* **2024**, *5*, 113–125. [CrossRef]
23. Nautiyal, K.; Gangodkar, D.; Diwakar, M.; Singh, P.; Bijalwan, A. Intelligent traffic light management using predictive and dynamic traffic flow analysis. *Sci. Rep.* **2025**, *15*, 37188. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.