VFAST Transactions on Software Engineering
https://vfast.org/journals/index.php/VTSE@ 2025, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 13, Number 4, September-December 2025 pp:137-148

The Secure GPS Tracking Data for transportation in Distributed Environments

Fida Hussain Khoso [1*], **Muhammad Asad Abbasi** [2], **Muhammad Shahid** [3], **Zahoor Hussain** [5], **Shafique Ahmed Awan** [6], **Muhammad Ahsan Riaz** [1]

[1]Department of Computer Science, Electronics Engineering, Dawood University of Engineering & Technology, Karachi, Pakistan; [2]Department of Computer Science and IT, Benazir Bhutto Shaheed University Lyari, Karachi Sindh, Pakistan; [3]Department of Computer Science, Iqra University, Karachi, Pakistan; [5]Department of Computer Science, Iqra University, Karachi, Pakistan; [6]Department of Computer Science, BBSUL, Karachi, Pakistan

Keywords: *Artificial Intelligence (AI), GPS Tracking, Secure Tracking Transport Method (SSTM), Smart Transportation, Distributed Systems, Encryption, Real-Time Monitoring, Karachi.*
Journal Info:
Submitted: November 10, 2025
Accepted: December 05, 2025
Published: December 31, 2025

Abstract
Efficient and secure tracking of the public transport has become a critical issue in smart cities, especially in urban areas like Karachi which has high population. Current systems are usually characterized by slowness, unreliability in data transmission and susceptibility to security risks including services by unauthorized users and information alteration. In an attempt to solve these problems, this paper introduces a Secure Transporting Tracking Method one of the (SSTM) which is a new AI-based system of real-time GPS tracking of the public transport in the distributed environment. The SSTM incorporates a secure transmit of GPS data, AIs, location prediction, and efficient vehicle-passenger matching to make it more accurate, less time-consuming, and data integrity-insured. The model has been applied to and tested in several places all around Karachi, such as Gulberg, Nipa, Gulshan-e-Iqbal and Sachal. The outcomes of simulations prove that SSTM is faster than the current approaches, such as the traditional GPS and chatbot-enhanced ones, in processing, encryption/decryption, and tracking. The research brings in a safe, large scale, and smart transport tracking platform specific to smart cities with the possible use in real-time fleet management and secure passenger safety systems.

***Correspondence author email address:** fidahussain.khoso@duet.edu.pk
DOI: 10.21015/vtse.v13i4.2271

1 Introduction
During the past years, the high population growth and fast urbanization of metropolitan cities like Karachi has given a lot of strain on its transport system. The problems that the passengers often face are unpredictable bus arrival time, poor management of routes and lack of real-time tracking options available to the passengers[1–4]. The conventional transport systems that in most cases are manualized in terms of scheduling and paper-based ticketing cannot handle

the dynamic needs of the smart cities. Furthermore, transportation data, such as GPS position and passenger data, vehicle status, etc., security is an acute issue, and threats of unauthorized access, data manipulations, and cyber-attacks are serious threats to the integrity of the system and the privacy of users.

In order to overcome these challenges, Artificial intelligence (AI) and Global Positioning System (GPS) application have been proposed as a solution to the development of intelligent transportation systems (ITS). The algorithms of AI can contribute to the optimization of the forecasting of routes, scheduling, and the accuracy of the real-time monitoring process, whereas GPS will provide an opportunity to monitor the position of public transport vehicles on the map in an accurate way. Nonetheless, the current GPS-based tracking systems are usually characterized by high latency, low scalability and weak security provisions, especially when in distributed and heterogeneous systems (5). Such constraints, in turn, stop the effective deployment and use of smart transport solutions in cities such as Karachi where internet connectivity is not always stable, and server failures are a common occurrence.

This paper expounds on a Secure Tracking Transport Method (SSTM), a new framework, aimed at offering a secure, efficient, and real time GPS tracking on transportation to the general population in distributed smart city settings. The suggested system uses the models of AI-based prediction, secure encryption, and optimized scheduling to reduce the processing time, improve the accuracy of the data, and provide the system with strong security in the context of possible threats. The framework has been applied and put into practice in different towns in Karachi such as Gulberg, Nipa, Gulistan-e-Johar, Saadi, Gulshan-e-Iqbal and Sachal and has been proven to be practical in real life urban environment.

The key contributions of this paper are as follows:

- **Design of a Secure Tracking Framework:** We propose a novel architecture for secure GPS data transmission and processing in distributed transportation networks, integrating encryption, AI-based prediction, and real-time monitoring.
- **Development of the SSTM Algorithm:** We introduce a computationally efficient algorithm that optimizes vehicle-passenger matching, minimizes processing and encryption/decryption time, and improves tracking accuracy.
- **Implementation and Validation:** We develop the framework on the basis of the latest technological tools (React.js, Node.js, MongoDB, and Google Maps API) and test its functionality with the help of the large-scale simulations on the real GPS data of Karachi.
- **Performance Benchmarking:** We compare SSTM with existing methods (standard GPS and chatbot-assisted systems) across multiple metrics, including processing time, accuracy, encryption overhead, and system reliability.

The rest of this paper is structured as follows: Section 2 conducts a literature review of related literature in GPS tracking, AI in transportation and secure distributed systems. Section 3 explains the suggested SSTM architecture, mathematical model and algorithm. Section 4 is the system implementation and experimental configuration. In section 5, we speak about performance evaluation and simulation results. Lastly, Section 6 gives a conclusion to the paper and plans the future research.

2 Related Work

This essay is the report to calculate the arrival time to the bus station of the people through the use of GPS-based information management systems. It aims at minimizing the waiting time of passengers by monitoring the bus locations in real-time [1]. Mobile computing is used to predict the arrival-time, and it requires constant monitoring and updates done centrally. The article has pointed out that passengers usually have to endure long queues and lack of transport information because bus routes are not clearly marked in their transport systems thus rely on informal sources of information at bus terminals [2–4].

To overcome these shortcomings, the authors suggest that RFID technology should be combined with the wireless sensor network (WSN) to have a superior bus management that will expand the RFID read ranges and optimize the performance of readers and utilization of power. The system is made up of three modules, which are; Passenger Module, Server Module, and Bus Device Module. Analytic prediction of bus arrival times and distances is made possible[5–7].

Subsequent researches reveal that most of the past bus-tracking and breakdown-management systems were not efficient. In Sri Lanka, a real-time arrival prediction system was created based on the GPS positioning to enhance punctuality and passenger experience [8]. It consists of a client application, a server application and a GPS module which includes the hardware implementation, installation and testing of a GPS module [9].

Additionally, chatbots have demonstrated potential for enhancing public transportation information, particularly in developing nations. Factors such as user trust and anthropomorphism are crucial for effective chatbot adoption; however, prior studies have been limited in scope [10]. An IoT-enabled mobility system for public bus tracking using Bluetooth beacons was also proposed [11]. Mobility-aware blockchain-enabled vehicular fog offloading and scheduling models have been introduced for intelligent transport systems [12], along with federated ACNN-integrated IoT workload offloading models [13]. Evolutionary metaheuristic scheduling methods for cyber-physical systems were presented in this study [14]. A cost-efficient mobility-aware microservice offloading scheme for IoVT fog-cloud networks was proposed in [15].

A transient-fault-aware application partitioning mechanism for mobile cloudlets was introduced in [16]. In contrast, a secure, fault-tolerant, delay-optimal workload assignment framework for Internet-of-Drone applications was proposed in [17]. A homomorphic blockchain-based secure intelligent transport service architecture was presented in [18]. An energy-efficient federated learning scheduling system for blockchain-integrated healthcare networks was developed in [19], and a metaverse-assisted body sensor network architecture was introduced in [20].

Delay-optimal IoT workload scheduling approaches for heterogeneous edge–cloud systems were proposed in [21]. Dynamic content- and failure-aware task offloading schemes for mobile cloud networks were developed in [22], and deadline-aware energy-efficient scheduling algorithms were proposed in [23]. Multi-layer latency-aware workload assignment models for e-transport IoT applications were introduced in [24]. A sustainable AI-assisted energy-efficient edge–cloud system for industrial IoT workloads was presented in [25]. A deep reinforcement learning- and blockchain-enabled IoMT workflow optimization model was proposed in [26].

Federated learning-based green edge computing frameworks for IIoT systems were introduced in [28]. Federated learning was also used to enhance intelligent reflecting surfaces in fog–cloud cellular environments [29]. A hybrid workload-aware secure healthcare monitoring framework for distributed fog–cloud systems was proposed in [30]. A lightweight secure adaptive scheme for IMoT applications was introduced in [31]. A digital-twin-enabled secure IoHT architecture integrated with federated fog–cloud systems was developed in [32]. A fuzzy decision-based sustainable transport model for fog networks was proposed in [33]. Deep neural network–based hospital workflow partitioning and scheduling schemes were introduced in [34]. An RBM-assisted secure serverless mobile edge computing system for IoMT analytics was proposed in [35]. Hybrid runtime offloading and resource allocation strategies for mobile cloudlets were designed in [36]. Secure dynamic application partitioning and biosensor workload scheduling methods were presented in [37]. A multi-objective federated learning blockchain-enabled IIoT architecture was developed in [38]. A blockchain-based energy-efficient IoE data scheduling system for smart home and vehicle environments was introduced in [39]. A sustainable and secure blockchain-assisted AIoT supply chain system was proposed in [40].

3 Proposed Methodology

This section details the proposed Secure Tracking Transport Method (SSTM), a novel framework for secure and intelligent real-time GPS tracking in smart city transportation systems. The methodology is structured into five core components:

1. **System Architecture Overview**
2. **Mathematical Modeling**
3. **Secure Transmission Protocol**
4. **AI-Based Prediction Model**
5. **Optimization and Vehicle-Passenger Matching**
6. **SSTM Algorithm Description**
7. **System Implementation**

Each component is designed to address the key challenges of real-time tracking, data security, computational efficiency, and scalability in distributed urban transport environments.

3.1 System Architecture Overview

The architecture will entail incorporation of real-time GPS survey, safe encryption/decryption modules, AI-based prediction engines, and a mobile application that would be linked with the passenger through Bluetooth and GPS. The system acquires the transport and passenger data in distributed servers that guarantee that the network has low latency and high reliability even when the network is unstable as illustrated in Figure 1.

The SSTM framework follows a layered, modular architecture that supports scalability, security, and real-time processing. The architecture is divided into four interconnected layers:

- **Data Acquisition Layer:** This layer is made up of public transport vehicles that are GPS equipped and which periodically transfer location coordinates. All cars have an onboard GPS module and a communications unit that provides the support of encrypted data transmission through the 4G/5G, Wi-Fi, or Bluetooth connection.
- **Processing & Prediction Layer:** Incoming GPS data streams are processed in real-time using AI-based predictive models. This layer is deployed on edge servers located at key transit hubs within Karachi to minimize latency. Historical and real-time data are used to train and update prediction models dynamically.
- **Security & Encryption Layer:** All GPS packets are encrypted before transmission using AES-256 symmetric encryption. Decryption occurs at the edge or cloud server after secure receipt. This layer also handles key management, authentication, and intrusion detection to prevent unauthorized access and data tampering.
- **Application & Interface Layer:** This layer provides user-facing interfaces via mobile and web applications. Passengers can track nearby vehicles in real-time, receive arrival predictions, and get alerts. Fleet managers access a centralized dashboard for monitoring, scheduling, and security auditing.

The architecture is designed to operate in both online and offline modes. In cases of network instability, the system temporarily stores data locally using the browser's IndexedDB or vehicle-side storage, synchronizing once connectivity is restored.

3.2 Mathematical Modeling

Let $V = \{v_1, v_2, \ldots, v_m\}$ be the set of m public transport vehicles operating within Karachi, each equipped with a GPS transceiver. Let $L = \{l_1, l_2, \ldots, l_n\}$ represent n passenger locations or predefined landmarks such as bus stops, universities, and residential towns (e.g., Gulberg, Nipa, Saadi, Sachal).

Each vehicle v_i transmits its geographic coordinates at discrete time intervals:

$$p_i(t) = \big(x_i(t), y_i(t)\big), \quad t \in \mathbb{R}^+,$$

where $x_i(t)$ and $y_i(t)$ denote latitude and longitude, respectively.

For a passenger situated at location l_j, the Euclidean distance to vehicle v_i at time t is computed as:

$$d_{ij}(t) = \|p_i(t) - l_j\|_2.$$

This distance metric serves as the basis for proximity-based vehicle recommendations.

3.3 Secure Transmission Protocol

To mitigate risks of data interception and manipulation, each GPS packet is secured using a lightweight

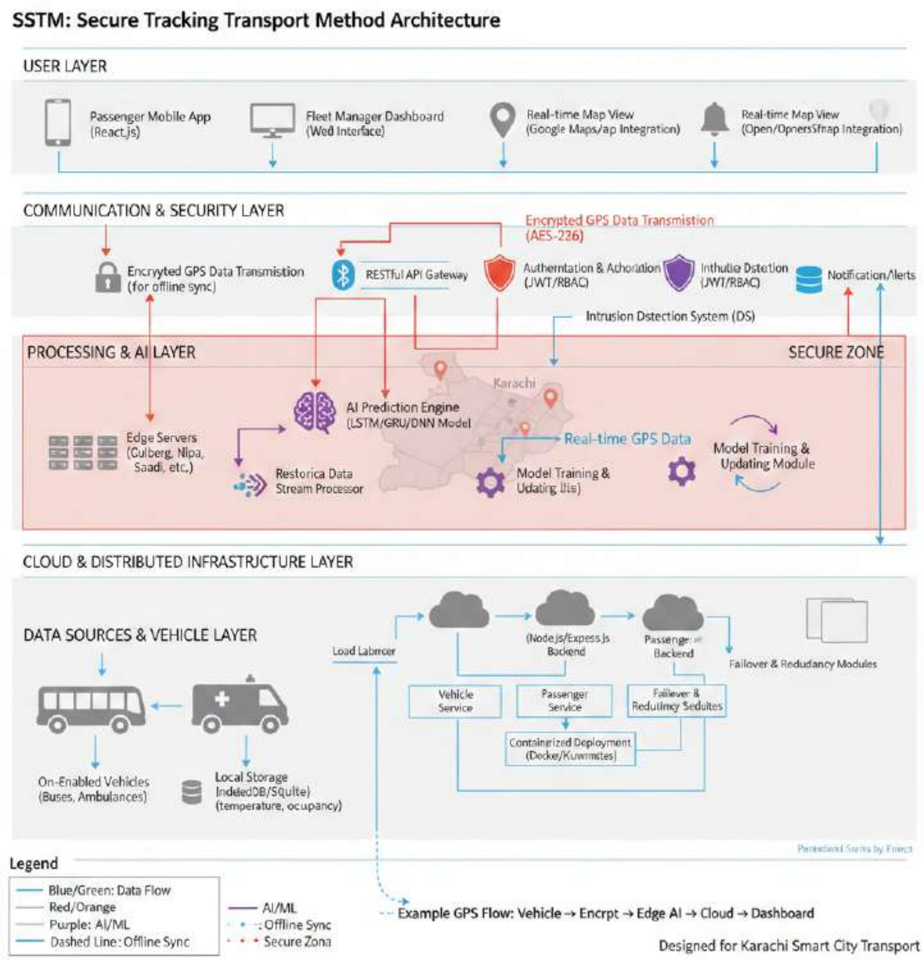

Figure 1. Proposed Secure Tracking Transport Framework

cryptographic protocol. Before transmission, the raw coordinates $p_i(t)$ are encrypted as:

$$C_i(t) = \mathrm{Enc}_K(p_i(t)),$$

where Enc_K denotes encryption under a symmetric key K.

The total secure transmission latency for vehicle v_i is modeled as:

$$T_i^{\mathrm{sec}}(t) = T_i^{\mathrm{enc}} + T_i^{\mathrm{net}}(t) + T_i^{\mathrm{dec}},$$

where:

- T_i^{enc}: Time required to encrypt the GPS packet,
- $T_i^{\mathrm{net}}(t)$: Network latency, which varies with congestion and signal strength,
- T_i^{dec}: Time required to decrypt the packet upon receipt.

Encryption and decryption times are modeled as functions of packet size and computational capacity of the transmitting/receiving node.

3.4 AI-Based Predictive Model

The SSTM employs a machine learning model f_θ to forecast vehicle trajectories and estimate arrival times. The model is trained offline on historical GPS trajectories collected from Karachi's transport network and fine-tuned online using real-time data.

Let $\hat{p}_i(t+1)$ denote the predicted position of vehicle v_i at time $t+1$:

$$\hat{p}_i(t+1) = f_\theta\big(p_i(t), p_i(t-1), \ldots, p_i(t-k)\big),$$

where k is the historical window length.

Prediction error is quantified as the Euclidean distance between actual and predicted positions:

$$\epsilon_i(t) = \|p_i(t+1) - \hat{p}_i(t+1)\|_2.$$

The accuracy of the prediction for vehicle v_i at time t is defined as:

$$A_i(t) = 1 - \frac{\epsilon_i(t)}{\epsilon_{\max}},$$

where $\epsilon_{\max}$ is a system-defined maximum tolerable error (e.g., 500 meters).

3.5 Optimization Formulation

The core objective of SSTM is to minimize secure transmission latency while maximizing prediction accuracy. This is formulated as a constrained optimization problem:

$$\min_{\theta} J = \alpha \cdot \frac{1}{m} \sum_{i=1}^{m} T_i^{\text{sec}}(t) \; - \; \beta \cdot \frac{1}{m} \sum_{i=1}^{m} A_i(t),$$

subject to:

$$T_i^{\text{sec}}(t) \leq T_{\max}, \quad A_i(t) \geq A_{\min},$$

where:

- α and β are weighting coefficients that balance the trade-off between latency and accuracy,
- $T_{\max}$ is the maximum allowable transmission delay (e.g., 2 seconds),
- $A_{\min}$ is the minimum acceptable prediction accuracy (e.g., 0.85).

3.6 Vehicle-Passenger Matching

Given a passenger at location l_j, the system selects the optimal vehicle $v_j^*(t)$ by solving:

$$v_j^*(t) = \arg\min_{v_i \in V} \left[d_{ij}(t) + \lambda \cdot T_i^{\text{sec}}(t) \right],$$

where λ is a tuning parameter that controls the relative importance of distance versus secure transmission delay. This formulation ensures that vehicles are selected not only based on proximity but also on communication reliability and security overhead.

3.7 Detailed SSTM Algorithm

The Secure Tracking Transport Method (SSTM) is implemented via a structured algorithmic pipeline designed to process real-time GPS data, apply encryption, perform AI-based predictions, and schedule transport resources efficiently. The complete algorithm is described step-by-step below.

Algorithm 1. Secure Tracking Transport Method (SSTM)

```
Input:
Methods = {SSTM, GPS, Chatbot}
RecordSizes = [1000, 2000, . . . , 10000]
Passenger locations L, Vehicle set V, GPS streams p_i(t)
Output:
Performance metrics: Processing Time, Encryption Time, Decryption Time,
  Accuracy, Security Score
Optimal vehicle–passenger assignments
Step 1: Initialization
Initialize data structures for passengers, vehicles, and location mappings.
Apply convex and concave optimization models to preprocess passenger,
  transport, and location data.
Load encryption/decryption modules and accuracy benchmarks from his-
  torical datasets.
Step 2: Metric Simulation
for each method M ∈ Methods do
    Compute processing time T^M_proc for transport search based on GPS lo-
  cations.
    Set accuracy A^M as a predefined constant derived from empirical bar
  graphs.
    Calculate encryption time T^M_enc and decryption time T^M_dec for passenger
  and vehicle data.
    Simulate network latency T^M_net(t) under distributed conditions.
end for
Step 3: Metric Analysis
Compare T_proc, T_enc, T_dec across all methods.
Evaluate accuracy–latency trade-offs for SSTM, GPS, and Chatbot-based
  tracking.
Implement GPS and Chatbot baseline methods for benchmarking.
Step 4: Passenger–Transport Tracking & Scheduling
for each passenger l_j do
    Apply job-shop scheduling to assign optimal vehicle v*_j(t).
    Use the cost function d_ij(t) + λT^sec_i(t) for vehicle selection.
    Update real-time tracking dashboard and notify passengers via mobile
  interface.
end for
Repeat until all passenger requests are satisfied or system timeout.
Step 5: Performance Logging & Output
Record all timing and accuracy metrics.
Generate comparative analysis reports.
Output optimal schedules and security audit logs.
Complexity Analysis:
The time complexity of the SSTM algorithm is O(log M × N), where M is the
  number of workloads and N is the number of scheduling nodes.
```

3.8 System Implementation

The SSTM framework has been implemented as a full-stack web and mobile system named **Transit Link**. Key technologies include:

- **Frontend:** React.js for dynamic, real-time user interfaces
- **Backend:** Node.js with Express.js for API services and business logic
- **Database:** MongoDB for persistent storage of vehicle, passenger, and trip data
- **Mapping:** Google Maps API and OpenStreetMap for geospatial rendering
- **Communication:** RESTful APIs for data exchange; Bluetooth for offline synchronization
- **Security:** AES-256 encryption, JWT-based authentication, and role-based access control

The system supports the following core functionalities:

- Real-time vehicle tracking with live GPS updates
- Route monitoring and historical trajectory analysis
- Automated alerts for route deviations or unauthorized access
- Secure passenger registration and trip logging

The implementation is publicly accessible via a hosted dashboard and open-source repository (see Section 4).

The given methodology offers a holistic, repeatable, and scalable architecture of secure GPS tracking in urban transport systems with smart cities. The combination of AI-predictive, cryptographic protection, and multi-objective optimization makes SSTM the most advanced tracking systems compared to those used in the classical field, as shown in the next section.

4 Performance Evaluation

The experiment involves a simulation that was carried out as part of the experiment and offers an overall assessment of the Secure Tracking Transport Method (SSTM) in a passenger transport management system. The main purpose of the simulation is to create a model city transport which will optimize efficiency, precision, and security of GPS-based, SSTM, and chatbot-assisted tracking systems. The simulation includes both the computational and operational elements of the transport framework, which allows conducting stringent performance analysis.

Its implementation takes advantage of Python and Java in order to perform complex computations, data processing, and algorithms. Python was used in data analysis, model optimization, and machine learning activities whereas Java offered a good platform to simulate, integrate, and interface with the mobile devices. A combination of these programming languages guarantees the efficiency of calculating computations and the testing of scheduling and tracking algorithms in different conditions.

Among the technologies used in the simulation, there are GPS to obtain the precise location and Bluetooth to support short-range communication between transport nodes and passengers. GPS provides proper real time coordinates of vehicles and pickup point, and Bluetooth is compatible with the scheduling of real time updates in a distributed environment. This combination of technologies allows a simulation model that can capture short-range as well as long-range interactions in an urban transport ecosystem.

The experiments were carried out based on real GPS data of Karachi, which represents a variety of traffic load, passenger loads, and vehicle supply. The simulation used 100000 transport vehicles to serve 100000 passengers in Karachi city transport network. To maintain statistical reliability, as well as the effect of variability of traffic patterns and passenger behavior, this experiment was repeated 25 times. Smartphones were used to simulate passenger devices to emulate real-life scenarios of smart cities usage.

Table the simulation parameters are presented in Table 1 and record the programming languages, system set-up, type of experiment, the number of vehicles, and passengers, number of repeats, and the type of devices.This structured recording ensures transparency and reproducibility of the simulation process.

The simulation results, illustrated in Figure 2, provide a comparative analysis of SSTM, conventional GPS, and chatbot-assisted tracking methods under varying operational constraints.

- **Processing Time Comparison:** SSTM demonstrated the shortest processing time, while the

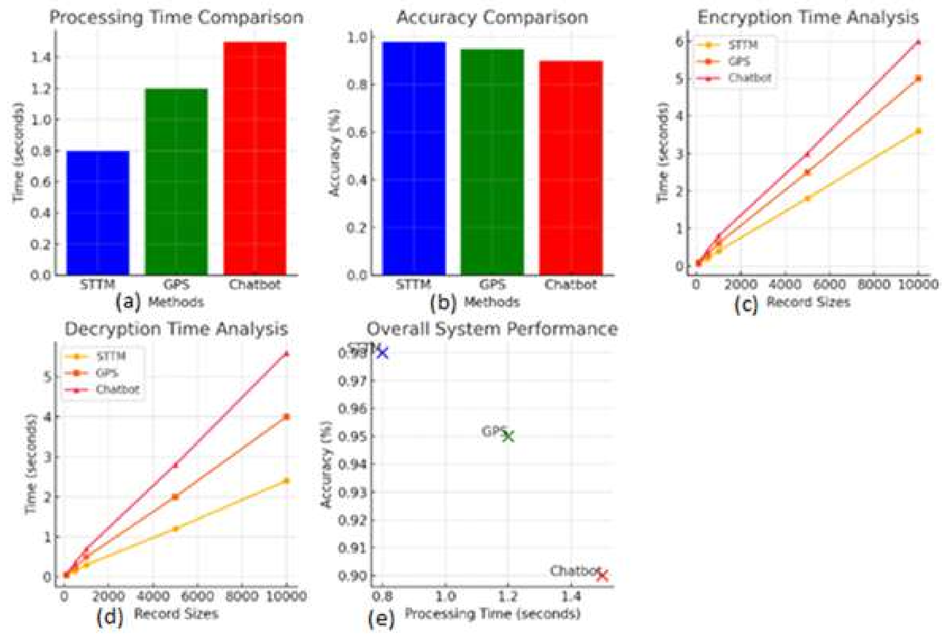

Figure 2. Result Analysis of Different Workload Assignments and Scheduling on Different Nodes.

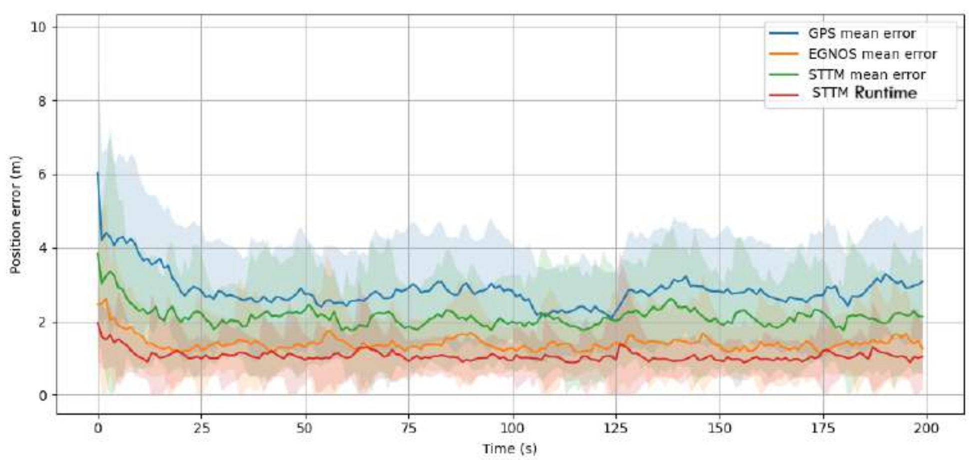

Figure 3. Runtime Error Finding of the Methods for the Correct Location.

chatbot method exhibited the longest. This improvement is attributed to the distributed workload allocation across local, edge, and cloud nodes.

- **Accuracy Comparison:** All three methods achieved high accuracy, with SSTM showing superior correctness in location prediction and vehicle selection.
- **Encryption Time Analysis:** As record sizes increased, encryption time rose across all methods. The chatbot method recorded the highest encryption overhead, followed by GPS, with SSTM being the fastest.
- **Decryption Time Analysis:** Similar trends were observed for decryption time, where SSTM consistently achieved the lowest latency, and the chatbot method the highest.
- **Overall System Performance:** The analy-

Table 1. Simulation Parameters

Simulation Variable	Values
Languages	Python and Java
Configuration	GPS, Bluetooth
Experiments	Karachi GPS Data, Navigation
Transport Vehicles	10,000
Passengers	100,000
Experiment Repetitions	25
Passenger Device Type	Mobile

sis reveals a trade-off between accuracy and processing time. SSTM provided the optimal balance, offering high accuracy with minimal latency. The GPS method yielded moderate results, whereas the chatbot method suffered from slower processing and marginally lower accuracy.

Figure 2(a) illustrates the execution time across tasks, confirming that SSTM achieves shorter processing times than existing methods. Figure 2(b) highlights the improved accuracy of SSTM in service selection and location search, minimizing errors during execution. Figure 2(c) demonstrates the scheduling efficiency of SSTM under scalable task loads. Figure 2(d) and 2(e) further validate that SSTM requires less time for encryption and decryption while maintaining higher accuracy compared to baseline approaches.

The principal contribution of this work is the optimization of secure GPS tracking data for transportation in distributed environments, supporting scalable and secure distributed applications. A noted limitation is that the proposed method does not currently defend against zero-day attacks.

Additional simulation results, including runtime error analysis and location search efficiency under static and dynamic conditions, further confirm that SSTM outperforms baseline methods in both accuracy and operational efficiency, particularly when vehicles rely on GPS for real-time applications.

The main finding of this paper is to optimize the secure GPS Tracking Data for transportation in Distributed Environments and support the distributed applications into system. The main limitation is that, the zero day attacks are not supported by the proposed methods.

We have generated many other results with the simulation to find the correct location and have minimum error rates at the runtime of scheduling.

Figure 3 shows the the propsoed methods STTM has higher location searaching with both static and runtime analysis as compared to baseline methods while vehicles used the GPS for the applications.

5 Conclusion and Future Work

This paper presented a secure global positioning system (GPS)-based framework for tracking public transportation in Karachi's smart city. The proposed Secure Tracking Transport Method (SSTM) ensures passenger safety and convenience by leveraging AI algorithms for GPS location and navigation data. The framework successfully implemented GPS-enabled transport tracking applications for passengers in various areas of Karachi, including Gulberg, NIPA, Saadi, and Sachal. Through simulation, the SSTM algorithm demonstrated optimal performance in processing time, encryption, decryption, and accuracy compared to existing methods. Such findings highlight the usefulness and the credibility of the suggested method to improve the smart transportation systems in smart cities. The upcoming studies will aim at broadening the framework to include real-time data analytics and dynamic route optimization as a way of enhancing the efficiency of the public transportation systems. Further attempts will be implemented to incorporate blockchain technology hence enhancing data security and privacy. Moreover, the system could be projected to other smart cities across the world in line with various urban transportation infrastructural arrangements. Focus will also be done on making lightweight AI algorithms on resource-constrained environments to enhance the scalability and usability of the system.

Code Availability

Please find the GitHub repository link for the project below.The repository contains the complete project script, source code, and related files. The user name and password are given below so you may access the software and code easily and also check its performance and accuracy.**GitHub Link:** https://github.com/mohdtaharana/Transit-link **Vercel Link:** https://transit-link.vercel.app/

Author Contributions

Fida Hussain Khoso conceptualized the study and prepared the original draft. **Muhammad Asad Abbasi** and **Muhammad Shahid** contributed to methodology design and experimental analysis, **Zahoor Hussain**assisted with data collection, simulation, and result validation.**Shafique Ahmed Awan** contributed to technical review and manuscript revision. **Muhammad Ahsan Riaz** supported literature review and formatting. All authors reviewed and approved the final manuscript.

Compliance with Ethical Standards

It is declared that all authors don't have any conflict of interest. It is also declare that this article does not contain any studies with human participants or animals performed by any of the authors. Furthermore, informed consent was obtained from all individual participants included in the study.

References

[1] O. D. Jimoh *et al.*, "A vehicle tracking system using greedy forwarding algorithms for public transportation in urban arterial," *IEEE Access*, vol. 8, pp. 191706–191725, 2020.

[2] Z. Ning *et al.*, "Blockchain-enabled intelligent transportation systems: A distributed crowdsensing framework," *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4201–4217, 2021.

[3] R. Kumar *et al.*, "A privacy-preserving secure framework using blockchain-enabled deep learning in cooperative intelligent transport systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 16492–16503, 2021.

[4] J. Srinivas *et al.*, "Designing secure user authentication protocol for big data collection in IoT-based intelligent transportation systems," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7727–7744, 2020.

[5] O. D. Jimoh *et al.*, "A vehicle tracking system using greedy forwarding algorithms for public transportation in urban arterial," *IEEE Access*, vol. 8, pp. 191706–191725, 2020.

[6] R. Verma, B. K. Singh, and F. Zahidi, "Management of GPS tracking systems in transportation," in *Intelligent Transportation System and Advanced Technology*. Singapore: Springer, 2024, pp. 251–263.

[7] P. Sadeghian, J. Håkansson, and X. Zhao, "Review and evaluation of methods in transport mode detection based on GPS tracking data," *Journal of Traffic and Transportation Engineering*, vol. 8, no. 4, pp. 467–482, 2021.

[8] V. Patil, S. B. Parikh, and P. K. Atrey, "GeoSecure-O: A method for secure distance calculation for travel mode detection using outsourced GPS trajectory data," in *Proc. IEEE International Conference on Multimedia Big Data (BigMM)*, 2019.

[9] M. Abinaya and R. U. Devi, "Intelligent vehicle control using wireless embedded systems based on GSM and GPS technology," vol. 3, pp. 244–258, 2014.

[10] M. N. M. Bhutta and M. Ahmad, "Secure identification, traceability, and real-time tracking of agricultural food supply during transportation using the Internet of Things," *IEEE Access*, vol. 9, pp. 65660–65675, 2021.

[11] O. Elijah *et al.*, "Transforming urban mobility with Internet of Things: Public bus fleet tracking using proximity-based Bluetooth beacons," *Frontiers in the Internet of Things*, vol. 2, p. 1255995, 2023.

[12] M. Ahmad, M. Bilal, A. Jolfaei, and R. M. Mehmood, "Mobility-aware blockchain-enabled offloading and scheduling in vehicular fog cloud computing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4212–4223, 2021.

[13] T. M. Grønli *et al.*, "IoT workload offloading efficient intelligent transport systems in federated ACNN-integrated edge–cloud networks," *Journal of Cloud Computing*, vol. 13, no. 1, p. 79, 2024.

[14] T. M. Grønli, G. Muhammad, and P. Tiwari, "Evolutionary meta-heuristic offloading and scheduling schemes for industrial cyber–physical systems," *IEEE Systems Journal*, vol. 18, no. 2, pp. 826–835, 2024.

[15] M. S. Memon *et al.*, "Cost-efficient mobility offloading and task scheduling for microservices IoVT applications in container-based fog–cloud networks," *Cluster Computing*, vol. 25, no. 3, pp. 2061–2083, 2022.

[16] X. Li, "Transient fault-aware application partitioning and computational offloading in microservices-based mobile cloudlet networks," *Computing*, vol. 102, no. 1, pp. 105–139, 2020.

[17] M. Elhoseny, M. A. Mohammed, and M. M. Jaber, "SFDWA: Secure and fault-tolerant delay-aware workload assignment in edge computing for Internet of Drone Things," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–12, 2022.

[18] T. M. Grønli, H. Wu, M. Younas, and G. Ghinea, "A novel homomorphic blockchain scheme for intelligent transport services in fog/cloud and IoT networks," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[19] A. Mohammed *et al.*, "Energy-efficient distributed federated learning offloading and scheduling for healthcare systems in blockchain-based networks," *Internet of Things*, vol. 22, p. 100815, 2023.

[20] A. Lakhan, "Metaverse-assisted healthcare body sensor network architecture," in *Proc. IEEE International Conference on Body Sensor Networks (BSN)*, Oct. 2024, pp. 1–4.

[21] M. A. Mohammed *et al.*, "Delay-optimal schemes for Internet of Things applications in heterogeneous edge–cloud computing networks," *Sensors*, vol. 22, no. 16, p. 5937, 2022.

[22] F. A. Khan and Q. H. Abbasi, "Dynamic content- and failure-aware task offloading in heterogeneous mobile cloud networks," in *Proc. Advances in Emerging Computing Technologies (AECT)*, 2020, pp. 1–6.

[23] M. A. Mohammed, A. N. Rashid, S. Kadry, and K. H. Abdulkareem, "Deadline-aware and energy-efficient scheduling for fine-grained tasks in mobile edge computing," *Int. J. Web and Grid Services*, vol. 18, no. 2, pp. 168–193, 2022.

[24] M. A. Dootio *et al.*, "Multi-layer latency-aware workload assignment for e-transport IoT applications in mobile sensor–cloudlet–cloud networks," *Electronics*, vol. 10, no. 14, p. 1719, 2021.

[25] A. Lakhan and T. M. Grønli, "Sustainable AI-assisted energy-efficient edge–cloud systems for industrial IoT applications," in *Intelligent Urban Mobility*. Springer, 2025, pp. 315–326.

[26] M. A. Mohammed, S. Kozlov, and J. J. P. C. Rodrigues, "Mobile–fog–cloud-assisted deep reinforcement learning and blockchain-enabled IoMT system for healthcare workflows," *Trans. Emerging Telecommunications Technologies*, vol. 35, no. 4, p. e4363, 2024.

[27] M. A. Mohammed *et al.*, "Efficient deep reinforcement learning-aware resource allocation in SDN-enabled fog paradigms," *Automated Software Engineering*, vol. 29, no. 1, p. 20, 2022.

[28] T. M. Grønli, A. Lakhan, and M. Younas, "Federated learning-enabled green edge computing systems for IIoT applications," in *Proc. Mobile Web and Intelligent Information Systems*, 2024, pp. 19–31.

[29] M. A. Mohammed, S. Kadry, *et al.*, "Federated learning-enabled intelligent reflecting surfaces in fog–cloud cellular networks," *PeerJ Computer Science*, vol. 7, p. e758, 2021.

[30] Q. U. A. Mastoi *et al.*, "Hybrid workload-enabled secure healthcare monitoring in distributed fog–cloud networks," *Electronics*, vol. 10, no. 16, p. 1974, 2021.

[31] A. H. Sodhro *et al.*, "A lightweight secure adaptive approach for Internet of Medical Things healthcare applications in edge–cloud networks," *Sensors*, vol. 22, no. 6, p. 2379, 2022.

[32] A. A. A. Lateef *et al.*, "Computer and information sciences," *Journal of King Saud University–Computer and Information Sciences*, vol. 35, p. 101747, 2023.

[33] M. A. Mohammed *et al.*, "Fuzzy decision-based energy evolutionary systems for sustainable transport in ubiquitous fog networks," *Human-centric Computing and Information Sciences*, vol. 13, 2023.

[34] Q. U. A. Mastoi *et al.*, "Deep neural network-based application partitioning and scheduling for IoT-assisted mobile fog–cloud healthcare systems," *Enterprise Information Systems*, vol. 16, no. 7, p. 1883122, 2022.

[35] M. A. Mohammed *et al.*, "Restricted Boltzmann machine-assisted secure serverless edge systems for the Internet of Medical Things," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 673–683, 2022.

[36] F. H. Khoso *et al.*, “Hybrid runtime offloading and resource allocation in mobile-assisted cloudlet-based cloud networks,” *Journal of Information & Communication Technology*, vol. 14, no. 2, 2021.

[37] A. Lakhan *et al.*, “Dynamic application partitioning and secure task scheduling for biosensor healthcare workloads in mobile edge clouds,” *Electronics*, vol. 10, no. 22, p. 2797, 2021.

[38] M. A. Mohammed *et al.*, “Federated learning-aware multi-objective modeling and blockchain-enabled systems for IIoT applications,” *Computers and Electrical Engineering*, vol. 100, p. 107839, 2022.

[39] M. A. Mohammed *et al.*, “BEDS: Blockchain energy-efficient IoE sensor data scheduling for smart home and vehicular applications,” *Applied Energy*, vol. 369, p. 123535, 2024.

[40] Z. A. A. Alyasseri *et al.*, “Sustainable secure blockchain-assisted AIoT and green multi-constraint supply chain systems,” *IEEE Internet of Things Journal*, 2025.

[41] A. Lakhan *et al.*, “Cost-efficient service selection and execution in blockchain-enabled serverless networks for Internet of Medical Things,” 2021.

[42] A. A. A. Lateef *et al.*, “Secure fault-tolerant industrial IoHT frameworks based on digital twin federated fog–cloud networks,” *Journal of King Saud University–Computer and Information Sciences*, vol. 35, no. 9, p. 101747, 2023.

[43] M. Elhoseny *et al.*, “Underwater sensor multi-parameter scheduling for heterogeneous computing nodes,” *ACM Transactions on Sensor Networks*, vol. 18, no. 3, pp. 1–23, 2022.