International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/11 Volume 12, Issue 2, March – April (2025)

RESEARCH ARTICLE

PPFedSL: Privacy Preserving Split and Federated Learning Enabled Secure Data Sharing Model for Internet of Vehicles in Smart City

Komala Soares
Electronics and Communication Engineering, College of Engineering, Bharati Vidyapeeth (Deemed to be University), Dhankawadi, Pune, India.
✉ pccegoa@gmail.com

Arundhati A. Shinde
Electronics and Communication Engineering, Bharati Vidyapeeth (Deemed to be University), College of Engineering, Pune, India.
aashinde@bvucoep.edu.in

Mangal Patil
Electronics and Communication Engineering, Bharati Vidyapeeth (Deemed to be University), College of Engineering, Pune, India.
mvpatil@bvucoep.edu.in

Received: 04 December 2024 / Revised: 15 March 2025 / Accepted: 25 March 2025 / Published: 30 April 2025

Abstract – Recently, the Internet of Vehicles (IoV) technology has played a pivotal role in enhancing transportation efficiency and safety. In this context, high-density vehicles generate more sensitive heterogeneous data, increasing privacy concerns in secure IoV data sharing. Federated Learning (FL) and Split Learning (SL) are trending paradigms of collaborative learning that facilitate potential solutions to privacy and heterogeneous data concerns. Thus, developing a privacy-preserving collaborative learning strategy is crucial for improving performance. This paper introduces a novel Privacy-Preserving Federated and Adaptive Split Learning (PPFedSL) strategy, enabling secure and efficient data sharing for IoV in smart city environments. This model integrates adaptive SL and FL by establishing a dual-tier privacy-preserved data-sharing strategy. Exploiting lightweight and hybrid cryptographic algorithms across different tiers ensures security and efficiency in data sharing across edge and cloud infrastructures without imposing significant computational overhead. This approach has designed two phases: privacy-preserving SL-enabled vehicle-edge collaboration and privacy-preserving FL-enabled edge-cloud collaboration. The proposed strategy effectively addresses latency constraints by delegating emergency decision-making at the edge level, where data is processed close to the IoV devices. Edges can inspect and respond to pressing data streams in real-time and guarantee timely interventions for latency-sensitive traffic management and collision avoidance scenarios. Finally, the experimental results demonstrate the efficiency of this proposed PPFedSL. The PPFedSL enhances the robustness efficiency by 93.2% and learning accuracy by 98.4% with high privacy preservation and heterogeneous data handling.

Index Terms – Smart City, Internet of Vehicles (IoVs), Split Learning (SL), Federated Learning (FL), Dual-tier Security, Privacy Preservation, Cryptography Primitives, Secure Collaborative Data Sharing.

1. INTRODUCTION

The transformation of smart cities with technological breakthroughs plays a pivotal role in urban transportation development in which the Internet of Things (IoT) converges to create a more secure, efficient, and sustainable urban communication environment. Thus, the IoT is the key pillar of this transformation, enabling a promising revolution in transportation systems by enabling seamless communication and data sharing among network entities [1] [2]. However, as intelligent vehicles equipped with smart sensors, cameras, and other data-generating devices generate massive data, this abundance of sensitive data poses key challenges to security and privacy [3]. Effective collection and inspection of vehicle data holds immense potential in enhancing road safety and optimizing vehicle traffic flows, ultimately enhancing transportation efficiency.

However, it remains paramount that data security with user information privacy, such as vehicle location, driving patterns, and vehicle diagnostics, are critical for successful deployment [4]. Traditional secure data-sharing approaches often entail vehicle information in centralized cloud servers, leading to a single point of failure, potential security breaches, and privacy violations [5]. FL-based security solutions effectively address these challenges by contributing a decentralized solution that enhances data security and user privacy in IoV-enabled smart cities.

A special form of privacy-preserving machine learning model, FL is a promising approach to enable distributed customized learning models at edge vehicles without sharing the raw information of users to the central server [6]. Aggregating local model updates of different edges improves FL model accuracy and promotes collaborative learning through a global model that is retrained independently across edges and vehicles [7]. However, the FL poses significant challenges while implementing it with IoV secure data sharing [8]. Firstly, although the FL preserves the data privacy of IoVs by keeping the sensitive data localized on edge devices, there is still a risk of privacy breaches in FL model sharing, particularly when malicious nodes gain access to the aggregated model updates. Secondly, the heterogeneous vehicles and edges in IoV vary in software and hardware capabilities, resulting in limited learning performance. The heterogeneous vehicles and edges have different data quality, quantity, and distributions, and it is challenging to address the model disparities in secure FL data sharing, especially in Non-Independent and Identically distributed (non-IID) environments. Finally, the edges and FL aggregator are frequently involved in FL model update-related communication, which escalates the delay and communication overhead, particularly in resource-constrained IoV environments. Preserving raw data in vehicles is an appropriate solution to enhance the efficiency of privacy-preserving FL models.

Additionally, frequent communication among edges and the FL aggregator for updating the model resulted in higher latency and escalated communication overhead, which can strain IoVs with resource constraints. Retaining raw data at the vehicle level can shrink some of these difficulties by ensuring privacy and diminishing unnecessary data transfers. An alternative solution adapts SL for collaborative learning, dynamically retains raw data at the vehicle according to the resources, and splits the model process among vehicles and edges. SL effectively diminishes vehicle resource demands and alleviates potential privacy risks by only transmitting intermediate model activations like gradients. Moreover, a cohesive SL-FL paradigm is vital for secure, privacy-preserved, efficient data sharing in edge-enabled IoV within smart cities. By consolidating the privacy-preserving benefits of SL and the scalable distributed nature of FL, this proposed enhancement can effectively address the challenges of data heterogeneity, data security, user privacy, and limited model accuracy in IoV data sharing over large-scale IoV environments.

Therefore, this work presents a dual-tier approach wherein diversified learning methods and cryptography algorithms are implemented in each tier. This enhanced design enables PPFedSL to meet stringent performance demands, incorporating high learning accuracy, low latency, and lower computation costs through distributed intelligence while safeguarding data security and user privacy over an IoV-enabled smart city environment.

1.1. Contributions

Major contributions of the proposed PPFedSL are as follows.

- The primary intention of PPFedSL is to design a dual-tier approach for enabling secure and privacy-preserving model sharing by addressing strict performance requirements that are model accuracy, latency, and computation cost of heterogeneous resource-constrained large-scale IoVs.
- The initial phase presents privacy-preserving SL-enabled vehicle-edge collaboration wherein vehicles share the heterogeneous model parameters through the adaptive SL structure and lightweight differential privacy by keeping the data in vehicles. Dynamic split point assists in effectively offloading the computationally intensive layers to the edges and effectively handling the heterogeneous, resource-constrained concerns of IoV.
- The proposed PPFedSL strategy effectively meets latency constraints. It guarantees timely responses by delegating emergency decision-making to the edge level where processed vehicle data as SL gradients close to the IoV layer.
- The next phase enables privacy-preserving FL-enabled edge-cloud collaboration, where edges share only the SL model parameters as input to the FL, significantly reducing complexity and minimizing delays associated with raw data input while accomplishing improved model accuracy in demanding IoV applications.
- FL aggregation shrinks the number of model transmissions across edges and the server during learning, diminishing communication overhead. Share only FL-local model updates with the server at a lesser communication cost while ensuring privacy and maintaining aggregation accuracy.

1.2. Paper Organization

The remaining part of this work is organized as follows. Section 2 reviews the works related to FL-based secure data sharing in IoV. Section 3 provides preliminary information, such as problem formulation and system model. Section 4 gives the design overview of the proposed PPFedSL and explains its mechanisms in detail. Section 5 illustrates the simulation setting and performance results of the PPFedSL model. Finally, section 6 concludes the paper.

2. LITERATURE REVIEW

Although the data-sharing process maximizes the efficacy of IoV, it is more susceptible to different kinds of attacks, particularly eavesdropping-related threats. Therefore, a lot of privacy-preserving strategies have been introduced to defend such attacks [9] [10] [11]. However, the conventional strategies necessitate a centralized server, resulting in they may suffer from a single point of failure. The FL-based security is an effective solution to solve the single point of failure issues through its distributed learning nature while maintaining the sensitive information locally [12]. Instead of sharing raw information with the centralized server, the FL instructs the nodes to only share the model parameter updates to the server and assures a required level of privacy [13]. The work in [14] proposes an FL-enabled vehicular approach that preserves the privacy of sensitive information in the vehicular cyber-physical system. Unlike fundamental FL-based methods, the scheme abolishes the need for centralized managers. Further, it achieves the model update asynchronously according to the random sub-gossip update mechanism. Thus, it guarantees model training efficiency and rectifies the single point of failure problem. Also, it adds Laplace noise to distort model parameters against inference attacks. Meanwhile, the FL-enabled vehicular scheme trains a detection approach for collaborative data leakage.

The work in [15] introduces an FL-based collaborative authentication scheme for shared information. It provides anonymous authentication among the vehicles, RSUs, and servers and protects privacy by encrypting the model parameters. However, it aggregates the model parameters at the centralized server, which may lead to single-point-of-failure issues, especially in large-scale environments. Another work in [16] improves the communication performance between the vehicle and server by including a federated bidirectional connection broad learning method over the training of local datasets. Since the server gathers only the model parameters based on the federated broad learning mechanism and maximizes the model aggregation capability, it can enhance the data sharing efficiency and ensure shared data credibility. However, this scheme fails to consider the inferential attack possibilities on model parameters and incurs high latency and overhead in the network. To shrink the latency in privacy-preserved data sharing, the work in [17] includes a deep Q-network and FL in an intelligent collaborative information-sharing approach. It includes asymmetric encryption methods to protect the privacy of shared data. However, it still needs the assistance of a centralized cloud server to aggregate the global models, which might suffer from a high burden and server failure problems. The work in [18] introduces a secured data-sharing model incorporating a two-tier authenticated consortium blockchain strategy with machine learning. It accomplishes reliability in the data-sharing process through a one-time password-based reputation mechanism. Further, it chooses optimal data providers using metaheuristics such as the particle swarm optimization (PSO) algorithm and ensures shared data quality. Although it exploits a normal main storage authority to store the public-private keypairs of vehicles with high authentication, it results in private key disclosure that leads to high attack activities.

Albeit the FL is most popular due to its distributed learning structure, it has to learn sub-optimal strategies over heterogeneously distributed data, which is a key hallmark in many applications. To effectively address this problem, a novel variant of personalized FL has been proposed in [19]. It specializes in robust robot learning models on distinct distributions of users. Similarly, the work in [20] proposes the DFL with diversified Data Sources algorithm (DFL-DDS) to diversify the vehicle data resources. Particularly, each vehicle maintains a state vector in which the contribution model weight of each data source is recorded. The work in [21] presents a novel strategy that addresses the data diversity problem in FL by employing hypernetworks, referred to as pFedHN, for personalized Federated Hyper Networks. This model trains a central hypernetwork to produce a set of models in which each client consists of one model. Through this architecture, such work enables effective parameter sharing among various clients while keeping the capacity to produce unique and different personal models. However, this model is unable to transmit the hypernetwork parameters and decouples the communication cost from the trainable model size. The work in [22] adopts a privacy-preserving FL-based approach across a federation of independent DaaS providers in IoV applications, such as traffic prediction and car park occupancy management. However, this model gathers the trajectory information from vehicles in a centralized way, and it may escalate the transmission delay and cause privacy leakage issues at the driver level. The work in [23] presents an FL framework for Traffic state estimation (TSE), named FedTSE, that jointly considers the privacy preservation, accuracy, model computation ability, and transmission cost over the vehicular network. Further, it includes a deep reinforcement learning strategy to upload and download the model parameters and improve the accuracy of the local model estimation. Thus, it also balances the tradeoff between computation and communication costs.

The work in [24] proposed a heterogeneous FL strategy, FedVPS, that follows a distributed architecture among the three entities, such as vehicle terminals, edges, and a cloud in the IoV network. This work also designs a privacy protection strategy with the assistance of Secure Multi-Party Computation (SMPC). Thus, it assures that the terminals participating in FL can get precise calculation results without revealing useful information, resulting in local dataset privacy preservation. This model not only protects the vehicle privacy but also enhances the communication efficiency. It enables an

[25] enhances the learning performance efficiency at IoVs. It chooses appropriate nodes for aggregation by considering the computing capacity, network capacity, and learning value of training samples. Meanwhile, it includes a dynamic waiting time strategy that dynamically adjusts the server waiting time at each round, which makes the FL process more rapid and precise. The work in [26] presents an enabled privacy-preserving data-sharing model by participant collaboration that can resist gradient leakage and poisoning attacks.

The work in [27] and [28] employs a homomorphic encryption scheme to preserve the privacy of the FL model. The work in [29] preserves the privacy of FL without additional computation requirements by incorporating the concept of gradient encryption. It utilizes the computational power of edges to finetune the FL model and encrypt the data for privacy preservation while keeping the performance. The work in [30] includes the split learning concept to construct a privacy-preserving intrusion detection system while keeping the raw data at local devices. Thus, it improves the learning speed and preserves user privacy over resource-limited IoV accurate and efficient distributed learning approach across the IoV structure. However, sending raw data to the edges incurs high complexity and latency in model construction. A semi-synchronous federated learning (Semi-SynFed) protocol in

environments. However, no global knowledge about the environment decreases the efficiency of IoV. The work in [31] incorporates FSL that separates the deep learning model into two sub-models and trains the models in the vehicles and the roadside units, respectively. It effectively handles the complexity caused by raw and massive heterogeneous data collection. The work in [32] also includes the concept of split and federated learning to classify the traffic in a 5G-enabled autonomous vehicle environment. However, it fails to consider the massive data generation characteristics, which highly impact the strict IoV performances. The research in [33] presents a federated split learning strategy where training is performed in the vehicles and the roadside infrastructures. The vehicles securely share the intermediate models as local gradients to the roadside infrastructure using differential privacy. The works in [34] and [35] handle the heterogeneous massive data generation by introducing adaptive and parallel split learning for edge learning. Albeit these methods attempt to ensure user privacy, they are not effective against robust privacy attacks. Table 1 compares the existing works using different factors.

Table 1 Comparative Analysis of Existing Works

Works	Used Dataset	FL	SL	Personalization	FL aggregation	Cryptography primitives	Privacy Preservation	Security	Model accuracy	Handling heterogeneity	Complexity	Latency	Communication overhead	Performance efficiency	Application suitability	Convergence Rate
P.Zhao, et al., 2022, [15]	Real-world	√	X	X	FedAvg	√	X	√	√	X	√	X	√	√	√	X
Nakanoya et al, 2021, [19]	HRD	√	X	√	FedAvg	X	X	X	√	X	√	√	√	√	√	X
Hangdong et al, 2023, [24]	BIT vehicle	√	X	X	FedVPS	√	√	√	√	√	√	√	√	√	√	√
Jin et al, 2023, [27]	WikiText	√	X	X	FedML-HE	√	√	X	√	X	√	√	X	√	√	√

Fang et al, 2021, [28]	Two-real world	√	X	X	FedAvg	√	√	X	√	X	√	√	X	√	√	√
Parekhet al, 2023, [29]	GTSRB	√	X	X	FedAvg	√	√	X	√	X	√	√	X	√	√	X
Agbaje et al, 2023, [30]	Open source	X	√	X	FedAvg	√	√	X	√	X	√	√	√	√	√	X
Wu et al, 2023, [31]	Real-World	√	√	X	FedAvg	√	√	X	√	X	√	√	√	√	√	X
Padaria et al, 2023, [32]	GTSRB	√	√	X	FAF	√	√	X	√	X	√	√	X	√	√	√
Proposed	BIT vehicle	√	√	√	FedSGED	√	√	√	√	√	√	√	√	√	√	√

The review shows that most FL and SL-enabled privacy-preserving data-sharing paradigms only address isolated concerns, such as single points of failure, overhead, latency, communication costs, and privacy. However, these paradigms often ignore the inherent non-IID nature of vehicle data. They lack focus on stringent performance benchmarks, including higher accuracy, low latency, and lower computation costs, which are crucial for large-scale, heterogeneous IoVs. This gap underscores the need for a novel approach that guarantees security and efficiency in distributed intelligence sharing across end-edge-cloud structures. Such a solution should support robust privacy protections while shrinking latency to facilitate seamless, widespread IoV deployments.

3. PRELIMINARIES

This section provides the preliminary information on the problem statement and system architecture of the proposed PPFedSL. The symbols and descriptions used by PPFedSL are described in Table 2.

Table 2 Symbols and Descriptions Used in PPFedSL

Symbols	Descriptions
V	Number of smart vehicles
E	Number of edges
C	Cloud server
Vi	ith vehicle
Ej	jth Edge
Di	Local dataset of Vi
SL_m	local SL-model parameters
G	Gradients
ϖ	Privacy Loss of lightweight differential privacy
L	Loss
₱	Privacy Parameter

α	Tradeoff Parameter for privacy and model accuracy
DSP_t	Dynamic split point at time t
R_v	Resource capacity of vehicle
R_e	Resource capacity of Edge
C_x	Computation cost of ath layer
N	Total number of CNN layers
RC	Resource constraints
$\sum_{x=1}^{a} C_x \leq R_v(t)$	Resource constraint of the vehicle at time t
$\sum_{x=a+1}^{N} C_x \leq R_v(t)$	Resource constraint of Edge at time t
$SL_{vehicle}$	Vehicle side model of SL
SL_{Edge}	Edge side model of SL
$f_{vehicle}(D_i, CNN_{initial})$	SL function performed at the vehicle side
$CNN_{initial}$	CNN initial layers
η	Laplace Noise
$SL_{Vehicle}{}'$	Noise appending Vehicle side model of SL
$E_{SL_{Vehicle}}$	Encrypted Vehicle side model of SL
E_k	Encryption Key lightweight differential privacy
D_k	Decryption key of lightweight differential privacy
$D_{SL_{Vehicle}'}$	Decrypted Vehicle side model of SL
$SL_{Edge}(E_j)$	SL edge model construction at the jth edge device
$f_{edge}\left(\frac{1}{V}\sum_{i=1}^{V} SL_{vehicle}(V_i)\right)$	SL function performed at the Edge side
$SL_{Client}(V_i)$	i numbers of vehicle models constructed using SL
$X_i^{(l)}$ and $Y_i^{(l)}$	The input layer and output layer of the vision transformer
h	Hidden layer of vision transformer
$MHSAh_i^{l-1}$	Multi Head Self-Attention layer of vision transformer
$AF(h_i^{l-1})$	Activation function of vision transformer

3.1. Problem Definition

Most of the conventional FL-enabled privacy-preserving strategies utilize cryptography methods, such as homomorphic encryption and differential privacy, to preserve the privacy level of IoVs. However, these models fail to accomplish a better tradeoff between privacy and model accuracy. Notably, fulfilling the privacy requirements without causing high delay and complexity in a heterogeneous resource-limited edge-enabled IoV environment is crucial. The privacy and model accuracy tradeoff involves balancing to protect sensitive IoV information while maintaining the efficiency of learning strategies that exploit the heterogeneous IoV information. Lightweight differential privacy-enabled FSL is a highly suitable paradigm to rectify the outlined problem in a resource-constrained IoV environment. This paradigm facilitates effective distributed model training on IoV devices while keeping sensitive data and large-scale information secure on vehicles, protecting privacy without imposing overhead. This FSL-enabled proposed solution enhances user privacy without compromising data security and performance efficiency, even in resource-limited IoV environments. Consider an IoV system comprising V number of smart vehicles, E number of edge devices, and a cloud aggregation server C. This system can be referred to as a set N, where N∈ {V= {V_1, V_2,…, V_i}, E={E_1, E_2,…, E_j}, C}. Each vehicle V_i has a local dataset D_i, built from information gathered by distinct sensing devices such as cameras, sensors, and monitoring systems. Utilizing this dataset D_i, each vehicle V_i generates its local SL model parameters (SL_m). Instead of transmitting the raw dataset D_i to associated Edge E_j, the vehicle only forwards intermediate model outputs such as gradients (G), safeguarding data privacy while offering effective collaborative learning.

The local gradients are transferred by adding the noise for the lightweight differential privacy algorithm. Accordingly, the major Objective Function denoted as OF of the proposed PPFedSL is formulated using Equation (1).

$$\mathrm{OF} = \mathrm{min}_{\mathrm{L}} \sum_{\mathrm{i}=1}^{\mathrm{V}} \mathrm{SL}_{\mathrm{m}_{\mathrm{i}}} + \alpha * \varpi(\mathrm{L}, \mathrm{D}_{\mathrm{i}}, \text{₱}) \qquad (1)$$

In this, the term L is loss, and the term ₱ refers to the privacy parameter. The function, $\mathrm{min}_{L} \sum_{\mathrm{i}=1}^{\mathrm{V}} \mathrm{SL}_{\mathrm{m}_{\mathrm{i}}}$ shrinks the loss L for the client and edge model construction of SL, aiming to enhance accuracy. The next term $\alpha * \varpi(\mathrm{L}, \mathrm{D}_{\mathrm{i}}, \text{₱})$ represents the privacy-preserving function applied to gradients generated from dataset D_i that mainly depends on the information sensitivity and the privacy parameter. The tradeoff parameter α ensures the tradeoff between privacy and model accuracy. Moreover, this formulation shrinks the cumulative learning cost while guaranteeing user privacy using lightweight differential privacy algorithms, accomplishing an optimal balance between security and performance.

3.2. System Architecture

The FL and SL integrated IoV architecture within a smart city consists of three key layers: the IoV, edge, and cloud. The dual-tier structure is implemented for vehicle-to-edge and edge-to-cloud, respectively. Figure 1 illustrates the SL and FL-enabled, privacy-preserved, secured IoV data-sharing architecture tailored for a smart city environment. In this structure, multiple vehicles connect with a central cloud aggregation server through edges, as shown in Figure 1. Each vehicle constructs its local model locally during network initialization according to the real-time data. Hence, the model is inaugurated by the cloud aggregation server, guaranteeing that no local vehicle dataset is uploaded to either the edge or the cloud. Following a predefined number of epochs, the cloud server aggregates model parameters from the edge averages. It subsequently updates the global model reversely to the edges for model relearning. After completion of the initialization process, the time-critical decisions are taken at the edges, effectively reducing the latency impacts.

3.2.1. IoV or Data Layer

This layer comprises many smart individual vehicles equipped with intelligent onboard sensors, processors, and cameras. Vehicles are equipped with constrained computational power, memory, and energy resources. Each vehicle can gather information from its surroundings, like road conditions, parking availability, traffic density, and other safety and comfort parameters. Hence, vehicles are always connected with the corresponding Edge securely by exploiting a lightweight differential privacy algorithm that preserves data privacy. SL assures that only the intermediate models, rather than raw data, are shared with the edge layer. The IoV layer balances data usage and privacy in this SL-enabled setup. It enables vehicles to contribute valuable, real-time information across a wider network without sacrificing individual data privacy or overwhelming resources and achieving strict delay requirements.

3.2.2. Edge Layer

This layer comprises edge devices like road roadside units or towers. The edges are intermediate processing devices that take the cloud services closer to the vehicles. Edges have moderate processing and storage capabilities. It can perform computation tasks while shrinking latency and reducing bandwidth exploitation. Edge servers construct an FL model locally using SL parameters constructed with vehicle intermediate models. For this, each vehicle shares its SL intermediate model parameters as gradients to the Edge, which is used to build SL_m and integrates the SL_m into the construction of the FL model using a ViT for enhanced feature extraction and local learning. Further, it transfers the aggregated SL models of different vehicles as Local Model (LM) to the cloud server through the hybrid cryptographic algorithm-based privacy-preserving strategy. Moreover, by bringing computation services close to vehicles, the edge layer significantly handles latency constraints of critical data examinations and bandwidth load while enabling secure, privacy-preserved, efficient, timely data-sharing and model training across IoV devices.

3.2.3. Cloud Layer

This layer comprises a centralized cloud server responsible for storing and managing the massive heterogeneous IoV information. Cloud aggregation servers have high processing power, enabling them to handle complex computations and process large-scale data. It aggregates local models generated by the edges using Federated Stochastic Gradient Descent (FedSGD) and updates the global model by redistributing the global model to the corresponding edges. The global model is securely shared by using the hybrid cryptography algorithm to the edges without revealing the sensitive information of vehicle users.

3.2.4. Privacy Preservation for SL and FL

In the PPFedSL architecture, privacy-preserved SL and FL models are implemented through lightweight differential privacy and hybrid cryptography. This enhancement guarantees that raw data remains on vehicles and that unique data is shared as model parameters during SL and FL processes. SL model parameters are exchanged through a lightweight differential privacy algorithm, while FL model parameters are shared through hybrid cryptography. This intentional exploitation of SL and FL among IoV entities effectively manages the risk of exposing sensitive IoV data, fortifying data privacy at each tier without impacting efficiency. Moreover, integrating lightweight differential privacy and hybrid cryptography in vehicle-to-edge and edge-to-cloud collaborations can form a robust defense against privacy and security threats while ensuring stringent accuracy and meeting latency requirements across large-scale, heterogeneous IoV environments.

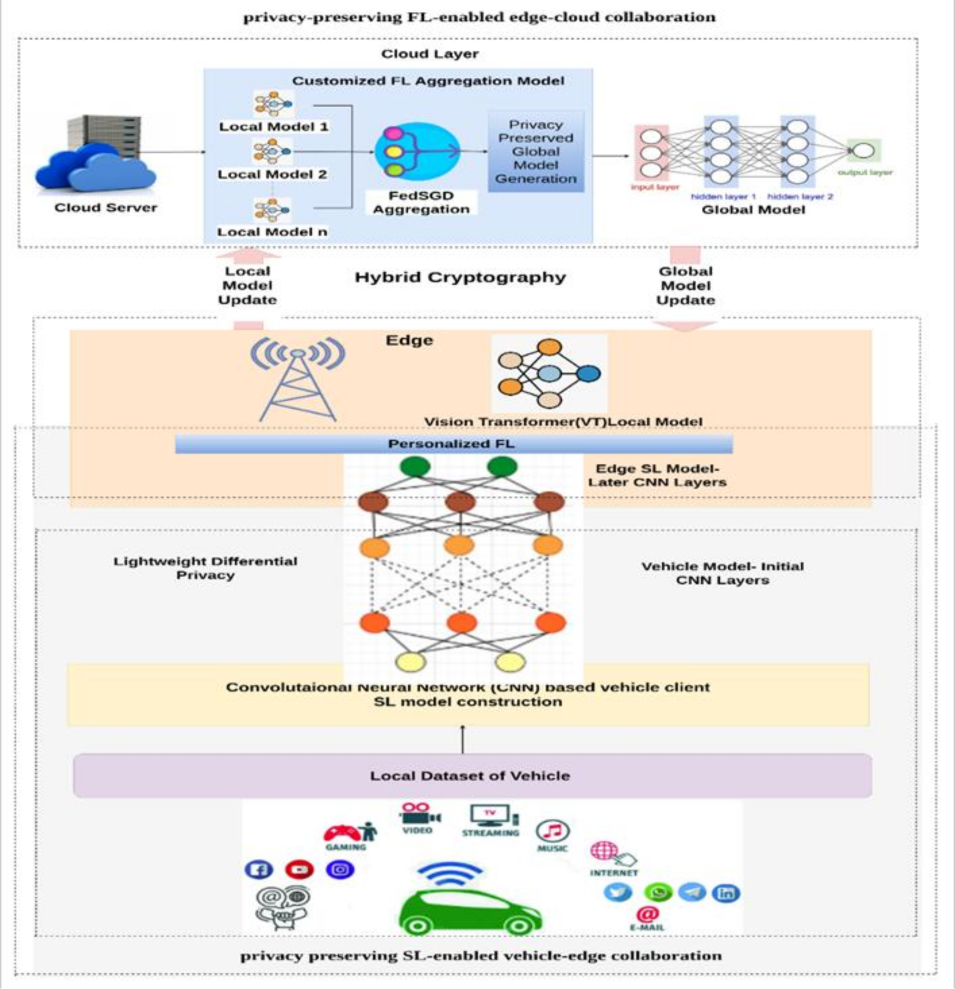

privacy-preserving FL-enabled edge-cloud collaboration
Cloud Layer
Customized FL Aggregation Model
Local Model 1
Local Model 2
Local Model n
Cloud Server
FedSGD Aggregation
Privacy Preserved Global Model Generation
input layer
hidden layer 1
hidden layer 2
output layer
Global Model
Local Model Update
Hybrid Cryptography
Global Model Update
Edge
Vision Transformer(VT)Local Model
Personalized FL
Edge SL Model-Later CNN Layers
Lightweight Differential Privacy
Vehicle Model- Initial CNN Layers
Convolutaional Neural Network (CNN) based vehicle client SL model construction
Local Dataset of Vehicle
GAMING
VIDEO
TV
STREAMING
MUSIC
INTERNET
E-MAIL
privacy preserving SL-enabled vehicle-edge collaboration

Figure 1 Architecture of the Proposed Model

4. PROPOSED WORK OVERVIEW

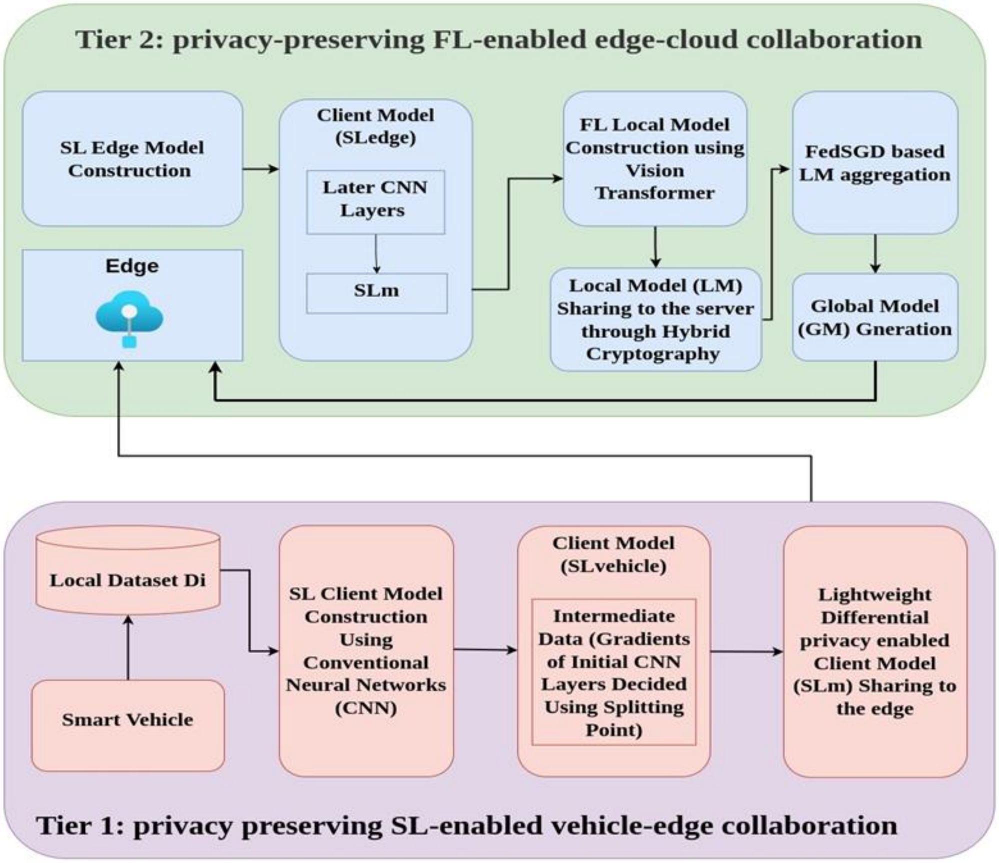

Tier 2: privacy-preserving FL-enabled edge-cloud collaboration
SL Edge Model Construction
Client Model (SLedge)
Later CNN Layers
SLm
FL Local Model Construction using Vision Transformer
Local Model (LM) Sharing to the server through Hybrid Cryptography
FedSGD based LM aggregation
Global Model (GM) Gneration
Edge
Local Dataset Di
Smart Vehicle
SL Client Model Construction Using Conventional Neural Networks (CNN)
Client Model (SLvehicle)
Intermediate Data (Gradients of Initial CNN Layers Decided Using Splitting Point)
Lightweight Differential privacy enabled Client Model (SLm) Sharing to the edge
Tier 1: privacy preserving SL-enabled vehicle-edge collaboration

Figure 2 Design Overview of Proposed Work

FL enables co-learning among entities without necessitating raw data. By deploying FL environments in IoVs, the intelligent services of vehicles, such as traffic management and route planning, are enhanced. However, the distributed storage of local datasets at edges or in the cloud makes traditional FL distributed learning systems more challenging. In an environment where the privacy of sensitive information is easily threatened, whether the local data of the vehicles is straightly gathered by the Edge or uploaded to the cloud, there is a risk of privacy leakages. Keeping raw data locally is the SL concept's optimal solution to reduce exposure.

The vehicles can adapt the SL model's initial layers based on the unique characteristics of the local dataset, which is a property crucial for non-IID handling. Therefore, the proposed paradigm incorporates the strengths of both SL and FL in its design for handling non-IID with high privacy and ensuring IoV performance requirements in distributed learning environments, respectively.

The design process of the proposed work is explained in Figure 2. The proposed PPFedSL comprises two phases.

- Tier 1

In this tier, the privacy-preserved SL model is implemented among vehicles and edges to keep the raw data on vehicles by only sharing the vehicle client models of SL constructed using Convolutional Neural Network (CNN) initial layers. This approach diminishes the risk of sensitive data exposure and follows privacy guidelines. This paradigm processes data near the vehicles by limiting the time-critical data examination at the edge level. It reduces any delay in critical decision-making, such as collision avoidance and traffic management of real-time applications. Shrinking latency in decision-making guarantees timely intervention in IoV safety-critical situations. Utilizing a lightweight differential privacy algorithm for model sharing at this tier enhances security and ensures privacy without imposing excessive computational overhead, which is vital for resource-constrained IoV devices. Notably, the lightweight differential privacy algorithm appends Laplace noise before vehicle client SL model parameter encryption and preserves privacy. Dynamic split point in this SL paradigm effectively offloads complex computational tasks of the CNN layers to edges and maintains fair resource handlings at IoV.

- Tier 2

In tier 2, the proposed model extends the capabilities of FL to facilitate secure collaboration among edges and the cloud aggregation server. By inputting SL_m constructed using a split model, the edges generated FL local models through the vision transformer. Implementing FL ensures aggregated model updates are shared between the edge and cloud, preserving the privacy of model parameters through a hybrid cryptography algorithm even at higher levels of collaboration. By leveraging FL, the cloud aggregation server refines the global model according to the local model contributions from multiple edges, guaranteeing that the system benefits from shared knowledge while ensuring privacy at each level of interaction. Thus, it enhances the privacy and security in both local and global model updates by safeguarding sensitive information while ensuring high-performance efficiency. By adopting a FedSGD-based secure model aggregation strategy in global model generation, the proposed work preserves the privacy level during global model aggregation and improves efficiency. Further, reversely shares the global models according to the principle of hybrid cryptography to the edges. By exploiting the globally shared parameters in the relearning process, the edges in the proposed work enhance the distributed learning efficiency and provide timely distributed knowledge about the driving environment across vehicles, thereby satisfying the strict performance requirements of IoV.

Moreover, administering emergency decisions at the edge meets latency requirements with non-IID handling. In contrast, managing more complex learning tasks in the cloud promises the performance efficiency of a large-scale heterogeneous IoV-enabled smart city environment.

4.1. Privacy Preserving SL-enabled Vehicle-Edge Collaboration

PPFedSL implements this approach among vehicles and edges to efficiently manage the heterogeneous, massive non-IID data produced by IoV systems while ensuring minimum latency. It incorporates adaptive privacy-preserving SL collaboration at the vehicle-edge level. Unlike FL and vertical FL, which process entire model updates across distributed nodes, the adaptive SL optimally splits learning tasks based on the dynamic splitting point, shrinking computational overhead in model construction.

Dynamic adoption of split point based on network conditions and vehicle resource availability, SL ensures seamless learning without high computation cost even in highly dynamic IoV environments. By retaining raw data locally on vehicles and sharing only the client-side SL model parameters with edges through lightweight differential privacy, this approach notably assures privacy and security without impacting data utility. This selective data-sharing paradigm shrinks communication overhead and diminishes latency in real-time critical IoV applications. Additionally, each SL vehicle model is perturbed with varied noise values, offering an added layer of privacy without impeding processing speed. This approach is essential for the responsiveness needed in large-scale IoVs. Therefore, it includes lightweight differential privacy to minimize distributed learning latency and burden at edges without sacrificing the model's accuracy and privacy.

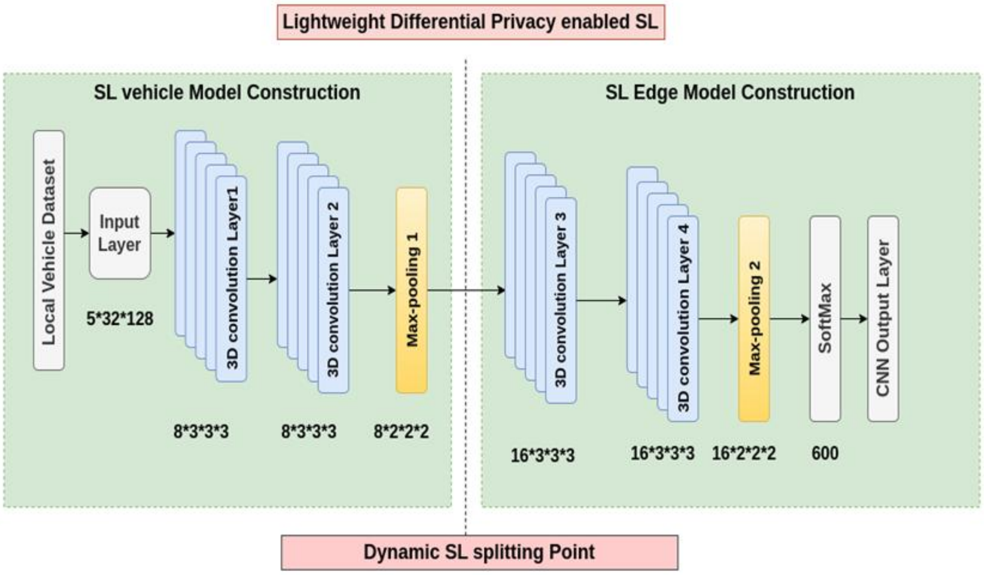


Figure 3 Construction of Splitting Among Vehicle and Edge

The proposed work retains raw information on local vehicles through an adaptive SL strategy, minimizing the computational burden and latency at the edges. The SL training process is strategically partitioned according to the dynamic splitting point decided based on vehicle resource level, as shown in Figure 3, where lightweight layers, specifically CNN layers 1 and 2, are processed on vehicles, while more computationally intensive layers, such as convolutional layers 3 and 4, are offloaded to the Edge. This dynamic split point-based partitioning effectively minimizes latency and ensures faster and more efficient data handling in IoV environments, which is crucial for time-critical decision-making. The dynamic split point is decided based on Equation (2).

$$\mathrm{DSP}_t = \arg\min_{a\in[1,N]}\left(\frac{\sum_{x=1}^{a} C_x}{R_v} + \delta\,\frac{\sum_{x=a+1}^{N} C_x}{R_e}\right) \qquad (2)$$

Where the term DSP_t is the dynamic split point at time t. Terms R_v and R_e are the resource capacity of the vehicle and edge, respectively. The term C_x refers to the computation cost of the ath layer, and N denotes the total number of CNN layers.

$$\mathrm{RC} = \begin{cases} \sum_{x=1}^{a} \mathrm{C_x} \le \mathrm{R_v(t)} \\ \sum_{x=a+1}^{N} \mathrm{C_x} \le \mathrm{R_v(t)} \end{cases} \qquad (3)$$

In Equation (3), RC is the resource constraint considered for dynamic split point construction. Where the vehicle and edge resource constraints should satisfy the functions $\sum_{x=1}^{a} \mathrm{C_x} \le \mathrm{R_v(t)}$ and $\sum_{x=a+1}^{N} \mathrm{C_x} \le \mathrm{R_v(t)}$ at time t respectively. In PPFedSL, the selection of dynamic split points assists in addressing issues like synchronization, model freshness, efficiency, and privacy-preserved learning in large-scale IoV environments. Conventional SL exploits synchronized updates. However, the dynamic split point selection in the proposed SL model permits mitigating vehicle-level network delays by offloading critical computations to the edge with adequate resources for vehicles to enable asynchronous learning. This dynamic split point estimation prevents stale updates, guaranteeing that recent and relevant data is exploited for learning while optimizing network and computational resources.

The main intention is to collaboratively train the vehicles and edges without sharing raw data and determine optimal SL local model parameters (SL_m) through the cooperation of vehicle clients and edges, fulfilling the minimization of the loss function as shown in Equation (1). The terms $\mathrm{SL_{vehicle}}$ and $\mathrm{SL_{Edge}}$ of equation (4), refer to the vehicle side and edge side models of the SL algorithm, respectively, and the terms W1 and W2 are the weighting factors, with W1+W2=1.

$$\mathrm{SL_m} = \mathrm{W1} * \mathrm{SL_{vehicle}} + \mathrm{W2} * \mathrm{SL_{Edge}} \qquad (4)$$

The process is divided into three main steps: SL-client model construction using CNN, vehicle model encryption and decryption using lightweight differential privacy, and SL-edge model construction using CNN.

4.1.1. SL-Client Model Construction Using CNN

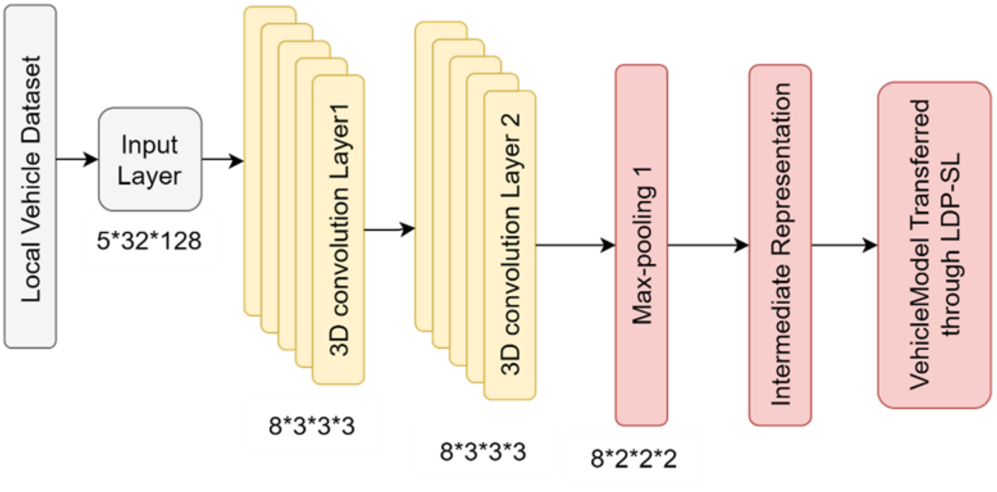


Figure 4 SL Vehicle Model Construction

CNN has powerful feature extraction capability with large-scale IoV datasets. The fully connected layers automatically learn unique features from the data and reduce manual feature extractions. It makes CNN especially effective for complex, high-dimensional dataset environments. Therefore, the PPFedSL includes the CNN algorithm and lightweight differential privacy to construct the SL-client model and share it with the edges in a privacy-preserving manner. SL prevents the edges from directly obtaining the vehicle's local data, protects the privacy of vehicles, and tries to ensure that the results of Edge local learning are close to the real-time environment. In other words, the SL model can collaboratively train the models across vehicles and edges without obtaining the raw data from the vehicle client. The SL training process is commonly divided into two segments: Vehicle-side model $SL_{Vehicle}$ and the Edge-side model SL_{Edge}. In this part, the $SL_{Vehicle}$ model is constructed based on the initial layers of CNN, as shown in Figure 4, and the intermediate representations are transferred to the edges using lightweight differential privacy. The vehicles offload heavy computational tasks to the edges through lightweight differential privacy-enabled intermediate representations to effectively accommodate the requirements for SL model training of resource-constrained vehicles. Implementation of SL among vehicles and clients neglects the necessity of sensitive data, consequently lowering the risk of data exposure in vehicles. Let us consider an edge-enabled IoV network in which each vehicle i has its private dataset, referred to as local dataset (Di), with several features. Each vehicle generates the $SL_{Vehicle}$ using the following Equation (5).

$$SL_{vehicle} = f_{vehicle}(D_i, CNN_{initial}) \qquad (5)$$

In Equation (5), $SL_{vehicle}$ denotes the model output of the vehicle clients. the term $f_{vehicle}(D_i, CNN_{initial})$ is the function performed at the vehicle side, and the terms D_i and $CNN_{initial}$ represents the local dataset of the ith vehicle and CNN initial layer model parameters, respectively. This phase dynamically decides a splitting point according to the vehicle resources, which separates the layers of CNN to the vehicle and edge sides, handling the resource scarcity. Further, the vehicles start to send the $SL_{Vehicle}$ parameters as intermediate representations to the edges.

4.1.2. Lightweight Differential Privacy for SL

Lightweight differential privacy is a powerful privacy-preserving technique that introduces noise to SL-vehicle client models before encryption, ensuring privacy protection. The FL applies lightweight differential privacy to the vehicle client model-sharing process among vehicles and edges, and it enables collaborative CNN model training without revealing sensitive information. Lightweight differential privacy is a variant of conventional differential privacy [36] and is highly adaptable to the resource-limited IoV environment. The steps of lightweight differential privacy used to share the SL client model from vehicles to edges are explained as follows.

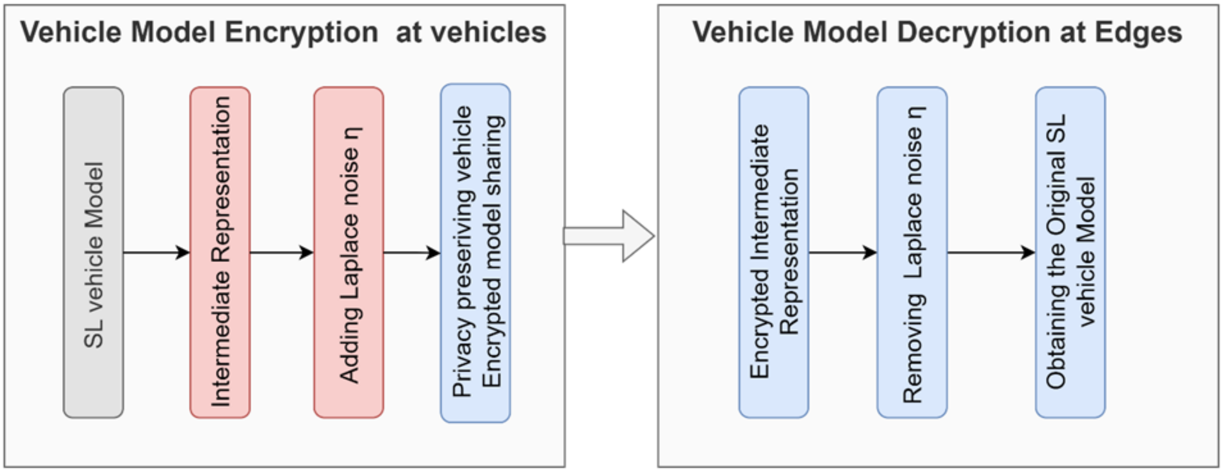

Vehicle Model Encryption at vehicles
SL vehicle Model
Intermediate Representation
Adding Laplace noise η
Privacy preserving vehicle Encrypted model sharing
Vehicle Model Decryption at Edges
Encrypted Intermediate Representation
Removing Laplace noise η
Obtaining the Original SL vehicle Model

Figure 5 SL Vehicle Model Encryption and Decryption Process

Model Partitioning: The SL dynamically partitions the CNN layers according to vehicle resource availability, expressed as follows.

Client or Vehicle Side Model: Responsible for the initial, convolutional, and pooling layers. For l=1,2,…, L, the output of each layer can be represented as:

$$Z^l = \left|f\left(X^{l-1} * K^l + b^l\right)\right|_R \qquad (6)$$

Where, X^{l-1} is the input layer. Further, it adds Laplace noise with the Z^l.

Laplace Noise Addition: According to the lightweight differential privacy principle [36], the Laplace noise η is added to vehicle model parameters ($SL_{vehicle}$) before encryption to accomplish differential privacy. This approach drowns the noise independently according to the Laplace distribution with appropriate scale parameter value b. Moreover, the amount of noise is appended with each $SL_{vehicle}$

based on the desired privacy level of the vehicle clients and the sensitivity of the client model parameters.

$$\eta = \mathrm{Lap}(0, b) \quad (7)$$

$$\mathrm{SL}_{\mathrm{Vehicle}}{}' = \mathrm{SL}_{\mathrm{vehicle}} * \eta \quad (8)$$

$\mathrm{SL}_{\mathrm{vehicle}}$ Encryption and Decryption: After adding noise η, the lightweight differential privacy encrypts the perturbed parameters by exploiting encrypted keys E_{k}. The encryption and decryption of the perturbed gradients of the $\mathrm{SL}_{\mathrm{vehicle}}$ is defined as follows.

$$\mathrm{E}_{\mathrm{SL}_{\mathrm{Vehicle}}} = \mathrm{E}_{\mathrm{k}}(\mathrm{SL}_{\mathrm{Vehicle}}{}') \quad (9)$$

$$\mathrm{D}_{\mathrm{SL}_{\mathrm{Vehicle}}}{}' = \mathrm{D}_{\mathrm{k}}\left(\mathrm{E}_{\mathrm{SL}_{\mathrm{Vehicle}}}\right) \quad (10)$$

In Equation (9) and (10), the terms E_{k} and D_{k} are encryption and decryption keys used by the lightweight differential privacy, respectively. The decryption is the reverse process of encryption. Vehicles perform the encryption process at the IoV layer, and noise is added to the SL vehicle model parameters. Edges perform the decryption process at the edge layer, removing the noise. Moreover, the SL vehicle model parameters are more precisely transferred to the edges in a privacy-preserved manner. This LDP fails to include formally defined parameters like privacy budget (ε), sensitivity (Δf), and noise scale (b) in its design, which is crucial for robust privacy guarantees. However, the LDP-enabled SL parameter sharing accomplishes privacy through Laplace noise addition (η) and encryption by exploiting keys (E_{k}). Perturbed model parameters are encrypted by vehicles at the IoV layer, guaranteeing security and privacy in transmission. At the edge layer, the process is reversed by decryption using a key (D_{k}), removing noise and retaining the original parameters. This method is more precise and enables privacy-preserved data transfer among vehicles and edges. While it is not formally parameterized, this method efficiently safeguards sensitive parameters in IoV scenarios.

4.1.3. SL-Edge Model Construction Using CNN

This part is executed on the edges where intermediate representations from the vehicle-side model are obtained to perform further processing with CNN later layers, as shown in Figure 3. Moreover, this process generates the final output of SL_{m} by retrieving the CNN edge model, $\mathrm{SL}_{\mathrm{Edge}}$ with model parameters.

$$\mathrm{SL}_{\mathrm{Edge}}\left(\mathrm{E}_{\mathrm{j}}\right) = \mathrm{f}_{\mathrm{edge}}\left(\frac{1}{\mathrm{V}}\sum_{\mathrm{i}=1}^{\mathrm{V}} \mathrm{SL}_{\mathrm{vehicle}}(\mathrm{V}_{\mathrm{i}})\right) * \varpi \quad (11)$$

```
\\ Lightweight Differential Privacy Preservation \\
Input: Non IID Local dataset of vehicles, CNN, and lightweight differential privacy;
Operations: SL-Vehicle model generation and sharing;
Output: SL model parameter, $SL_m$;
Initializes the network;
Each vehicle do {
Constructs its local dataset Di using the sensed information;
Inputs the raw dataset to the CNN;
CNN do {
Generates the $SL_{Vehicle}$ model parameters;
$SL_{vehicle} = f_{vehicle}(D_i, CNN_{initial})$            (5)
        }
Transfers the $SL_{vehicle}$using lightweight differential privacy;
Lightweight differential privacy algorithm do
{
Computes Laplace noise using equation (7);
Adds the η with $SL_{vehicle}$ using equation (8);
$SL_{Vehicle}' = SL_{vehicle} * \eta$
Performs encryption on $SL_{vehicle}'$using equation (9);
Consider $SL_{vehicle}'$ as intermediate representation of SL;
Shares the $SL_{vehicle}'$ to the corresponding edge;
}
};
Edge do {
Performs decryption on $SL_{vehicle}'$using equation (10);
Obtains the original $SL_{vehicle}$ by removing the η;
Computes the final SL model parameters $SL_m$ by computing $SL_{Edge}$;
};
```

Algorithm 1 Lightweight Differential Privacy Preservation of SL

In Equation (11), the term $\mathrm{SL}_{\mathrm{Edge}}\left(\mathrm{E}_{\mathrm{j}}\right)$ represents the SL edge model construction at the jth edge device with the V number of $\mathrm{SL}_{\mathrm{Client}}(\mathrm{V}_{\mathrm{i}})$, where the terms V_{i}. refer to the $\mathrm{SL}_{\mathrm{Vehicle}}$ output model parameters of the ith vehicle at the IoV layer. The term ϖ is the privacy loss of lightweight differential privacy. Instead of collecting raw data from vehicles in the IoV layer, the PPFedSL model only collects model parameters while ensuring privacy. It efficiently addresses the non-IID data heterogeneity of IoV data by leveraging the strengths of SL and lightweight differential privacy. Upon completion of this step, the edges initiate the construction and sharing of the FL model. The Edge should inspect the collected model parameters before model retraining through interaction with the cloud aggregation server. If the examined vehicle information is time-critical, the Edge promptly decides to handle the time-sensitive scenarios. This step is very crucial for real-time emergency scenarios.

4.2. Privacy-Preserving FL-enabled Edge-Cloud Collaboration

Though critical data decisions are made at edges, processing non-critical data at the cloud level globally enhances the learning ability of edges. Hence, the global parameters are generated based on the FL local parameters of different edges aggregated distributively to construct a unified model that upgrades the entire IoV performance while preserving data privacy. This section outlines the detailed FL local model construction processes, global model generation, and sharing within the proposed model. Figure 6 depicts the working process. The PPFedSL includes Vision Transformer (VT) and FedSGD at the edge and server levels to construct the local and global models. VT-enabled local model construction in FL over CNN-based alternatives delivers the superior capability to model and capture global dependencies, which is a unique demand for IoV scenarios. In contrast to CNNs relying on local receptive fields, VT exploits a self-attention mechanism to process entire feature maps, making it highly fit for handling heterogeneous FL model parameters across distributed edges. Also, the VT-based learning architecture scales better than CNNs when model complexity escalates, making it more suitable for IoV sensor data with high-resolution. Therefore, the proposed model selects VT instead of CNN models. This section is divided into three parts: LM construction using VT, GM Aggregation using FedSGD, and FL Model Sharing using hybrid cryptography.

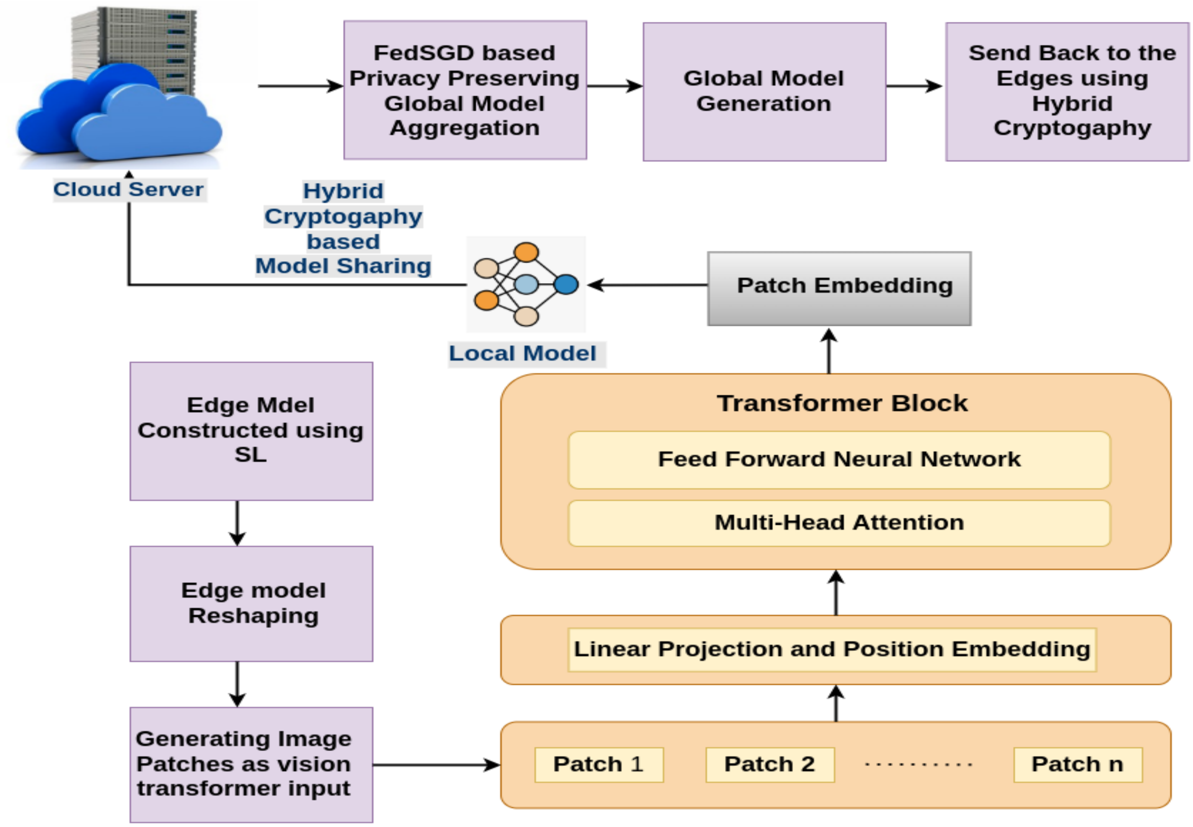

FedSGD based Privacy Preserving Global Model Aggregation
Global Model Generation
Send Back to the Edges using Hybrid Cryptogaphy
Cloud Server
Hybrid Cryptography based Model Sharing
Patch Embedding
Local Model
Edge Mdel Constructed using SL
Transformer Block
Feed Forward Neural Network
Multi-Head Attention
Edge model Reshaping
Linear Projection and Position Embedding
Generating Image Patches as vision transformer input
Patch 1
Patch 2
....
Patch n

Figure 6 Process of Hybrid Cryptography Privacy Preservation

4.2.1. Local Model Construction using Vision Transformer

Vision Transformer (VT) has emerged as a powerful strategy to analyze image datasets and has numerous attributes that fit the IoV environment. The self-attention mechanism offers robust understanding capabilities and captures long-range dependencies, which are very prominent for the dynamic and complex nature of IoV. Therefore, the proposed PPFedSL utilizes VT in LM construction. This phase involves learning the edges with different SL model parameters of distributed vehicles while keeping the data more private in vehicles. Hence, the CNN model parameters are reshaped according to the input requirements of VT. The VT architecture includes self-attention layers, activation function neural networks, and positional encodings. The proposed model modifies the vision transformer architecture to accommodate the FL setting.

The mathematical form of local model construction using a VT is expressed as follows.

$$X_i^{(l)} = \sum_{i=1}^{V} (SL_{Vehicle})_i \quad (12)$$

$$Y_i^{(l)} = X_i^{(l)} \left(LN\left(h_i^{l-1}\right) + \left(MHSAh_i^{l-1}\right) + AF\left(h_i^{l-1}\right)\right) \quad (13)$$

In Equation (12) and (13), the terms $X_i^{(l)}$ and $Y_i^{(l)}$ refer to the input layer and output layer of the vision transformer. The term h represents the hidden layer of VT. Incorporating the vision transformer in FL can offer a collaborative, decentralized learning structure, preserve privacy, and improve model efficiency. It also captures long-range dependencies in images of non-IID datasets more effectively and enhances the IoV data-sharing performance. After constructing the local models, each Edge transfers the LMs of non-critical data through hybrid cryptography to the server for global model generation.

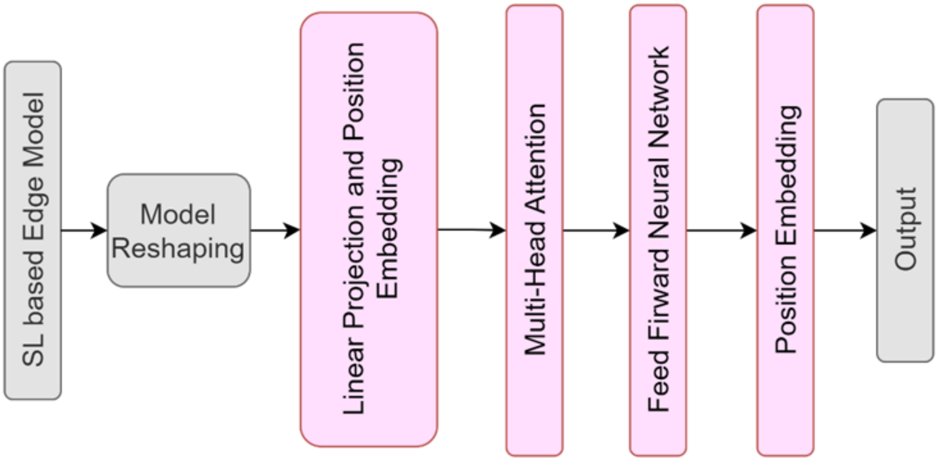


Figure 7 VT Architecture in Proposed Work

4.2.2. GM Aggregation using FedSGD

After receiving the LMs from different edges, the cloud server initially decrypts the LMs using decryption keys generated using hybrid cryptography. Further, it starts aggregation using the FedSGD algorithm.

$$GM = \frac{\sum_{j=1}^{E} W_j * E_j}{\sum_{j=1}^{E} W_j} \quad (14)$$

In Equation (14), GM represents the global model, and the E_j refer to the jth Edge. The term W_j is the weighting factor value that is equal to 1. Thus, the FedSDG utilizes weighted averages to aggregate the LM updates of different edge devices. After completing the aggregation steps, the cloud server returns the GM to the edges for retraining. By retraining the VT model using the updated GM, the edges can enhance its learning efficiency without compromising security and privacy, resulting in high IoV data-sharing performance.

4.2.3. FL Model Sharing Using Hybrid Cryptography

Hybrid cryptography securely shares model updates between edge and server while preserving privacy against disclosure attacks. Hybrid cryptography consolidates the strengths of both symmetric and asymmetric key cryptography to accomplish efficiency and security across edge and server. The proposed hybrid cryptography ingeniously amalgamates the AES robustness with the ECC agility. The design of hybrid cryptography symmetrically encrypts original FL model parameters like LM of edges with AES while utilizing the ECC for asymmetric encryption of the initial AES key [37]. Initially, the hybrid cryptography generates a 128-bit random number as the AES initial key.

Further, it derives the AES full key through subsequent key expansion, alignment, and other key retrieve-related operations. Thus, it neglects the complexities of AES key management and enhances privacy and security without sacrificing the IoV performance. The ECC public key encrypts the AES initial key, resulting in high security. Encrypting the AES initial key is the only choice for attack activities. Moreover, the Edge produces the final encrypted LM update by amalgamating the encrypted AES key and the ciphertext, which are then dispatched to the server. Upon receiving the encrypted LM updates from the edge devices, the server can extract the ciphertext and the encrypted AES key. This way, the LM update is securely shared among edges and servers. This process is reversed from the server to the edges during the GM update.

```
\\ Hybrid Cryptography Privacy Preservation \\
Input: SL vehicle models and LMs;
Operations: Privacy-preserving model sharing and GM generation;
Output: GM;
Initializes the FL model construction and sharing process;
Each edge do
{
    Inputs the SLvehicle to the VT;
    VT do
    {
        Generates the FL model parameters using equation (12) and (13);
        Constructs the LMs based on SL-Vehicle model of different vehicles;
    }
    Transfers the LMs using hybrid cryptography;
    hybrid cryptography do
    {
        Performs encryption on LM using hybrid cryptography keys;
        Shares the LMs to the cloud server;
    }
};
Server do
{
    Performs decryption on LM using hybrid cryptography keys;
    Obtains the original LMs;
    Starts the LM aggregation using FedSGD;
    Generates the global model GM;
    Sends the GM reversely to the edges;
};
```

Algorithm 2 Hybrid Cryptography Enabled Privacy Preserved FL

4.3. Computation Cost Analysis

Computation costs of LDP and HCA exploited for the vehicle-to-edge and edge-to-cloud communication in PPFedSL are comparatively analyzed in table 3. LDP proves high efficiency, with encryption and O(n) decryption costs, where n refers to the data size. This scenario is accomplished through lightweight symmetric encryption and the addition of noise, which also acquires a linear cost of O(n). It is very simple to manage keys in LDP as it relies solely on symmetric keys, resulting in the total computation cost for LDP remaining low at O(n). Thus, it is more suitable for resource-constrained vehicles in IoV environments.

On the other hand, HCA incorporates both symmetric AES and asymmetric ECC-based encryption, leading to higher encryption and decryption costs of $O(n)+O(k^3)$, where $O(n)+O(k^3)$ reflects the ECC operation's polynomial cost depend on the key size k. Like LDP, the HCA does not integrate noise addition, as it exploits strong cryptographic techniques for preserving privacy. Albeit the key management of HCA is complex owing to the combination of AES and ECC keys, it provides stronger security, and its computational overhead is not a significant concern in the context of edge-to-cloud collaboration where high resources are available, and advanced security is prioritized over efficiency.

Table 3 Computation Cost Analysis

Aspect	LDP	HCA
Encryption Cost	Low (O(n))	High ($O(n)+O(k^3)$)
Decryption Cost	Low (O(n))	High ($O(n)+O(k^3)$)
Total Cost	Low (O(n))	High ($O(n)+O(k^3)$)
Noise Addition	Yes (O(n))	No
Key Management	Simple (symmetric key)	Complex (AES + ECC keys)

5. EXPERIMENTAL EVALUATION

This section evaluates the performance of the proposed PPFedSL through Python libraries. The experimental setup consists of hardware and software configurations utilized in the experiments and a dataset description.

The environment setup is configured with the Intel i5 2.5GHZ CPU and a 16 GB memory server hosted by the Ubuntu 20.04 LTS operating system. Parameter settings are shown in Table 4. The PPFedSL implements the FSL within two levels. Consequently, it analyzes the effectiveness of the PPFedSL in terms of different performance metrics.

The results show that the proposed model outperforms various baseline deep learning strategies and existing works. For comparison, the works FedVPS [24], Traffic Sign Classification using Federated Split Learning (TS-FSL) [32], Privacy Preserving IDS using Split Learning (PP-ISL) [30], and Adaptive and Parallel Split Federated Learning in Vehicular Edge Computing (ASFV) [35] are exploited as shown in table 5.

Table 4 Parameter Settings

Components	Parameter	Value/Setting
MobileNetV2	IMG_SIZE	(224, 224, 3)
	Output_layer	‘block_13_expand’
Edge_Model (VT)	optimizer	'adam'
	loss	'categorical_crossentropy'
	metrics	['accuracy']
	IMG_SIZE	(14, 14, 64) (from MobileNetV2)
	patch_size	8
	num_patches	(IMG_SIZE[0] // patch_size) * (IMG_SIZE[1] // patch_size
	num_classes	6
	d_model	23
	num_heads	1
	ff_dim	4 * d_model
	num_layers	3
	dropout	0.1
Global_Model (FedSGD)	IMG_SIZE	(14, 14, 64) (from MobileNetV2)
	patch_size	8
	num_patches	(IMG_SIZE[0] // patch_size) * (IMG_SIZE[1] // patch_size
	num_classes	6
	d_model	23
	num_heads	1
	ff_dim	4 * d_model
	num_layers	3
	dropout	0.1
FL	num_clients	3
	num_epochs_per_round	5
	num_rounds	5

Table 5 Comparison of Proposed PPFedSL with Comparison Works

Works	FedVPS [24]	TS-FSL [32]	PP-ISL [30]	ASFV [35]	Proposed PPFedSL
Major Contribution	Heterogeneity and non-IID handling and privacy preservation	Minimize the computation cost with high accuracy	Construct a privacy-preserved IDS	Adaptive model splitting with parallelized training process	Satisfy strict IoV performance requirements
Dataset Used	BITvehicle	GTSRB	CAN and CICIDS2017	MNIST, Fashion MNIST, CIFAR-10	BITvehcile
Privacy preservation	SMPC	No privacy	SL based	SL based	lightweight differential privacy and hybrid cryptography
Decentralized Structure	FL	FL and SL	SL	FL	FL and SL
Learning Algorithm	MLP	CNN	CNN, LSTM, and GRU	RESNet18	CNN and VT

Aggregation model	Proto-type based	FedAvg	No aggregation	Not Defined	FedSGD
Non-IID handling	Yes	No	No	Yes	Yes
Latency	High	High	High	High	Low
Model Accuracy	Medium	High	Medium	High	Very high
Computation Cost	High	Low	Low	High	Low
Complexity	High	High	Medium	High	Low

5.1. Dataset Description

The proposed PPFedSL utilizes the BITvehicle dataset [38] for experiments due to its widespread exploitation in existing research and the different sets of images that assist in better feature extraction and vehicle identification. The BITVehicle is a challenging dataset comprising 9,850 vehicle images with high-resolution pixel values like1600×1200 px and 1920 × 1080 px. The images comprise a wide range of changes in their illumination, scale, surface color, and location. The images do not comprise the top or bottom parts of some vehicles owing to the capturing delay and the vehicle size. Each image may consist of more than one vehicle, and the same vehicle may present multiple times in different images, so every vehicle location is pre-annotated. Moreover, there is a large number of images capturing vehicles from various viewpoints and under different environmental conditions. The size of the dataset and vehicle diversity enable robust learning and evaluation of different vehicle detection models. The benchmark dataset is divided into subsets with balanced class labels to model the non-IID distributions. Table 6 lists the samples of the BITvehicle dataset and the accuracy results of PPFedSL.

Table 6 Samples of BITvehicle Dataset and Results of PPFedSL

Classes of Vehicles	Sample Count	Training	Testing	SL Accuracy	FL Accuracy	Aggregation Accuracy
Car	6000	5000	1000	97.8632	97.4358	97.8647
Truck	3000	2500	500	97.0085	98.4330	99.1452
Bus	1800	1500	300	96.1538	98.4330	99.0028
Van	2400	2000	400	96.0113	96.0113	97.4358
Motorcycle	1200	1000	200	95.7264	97.2934	98.2905
Bicycle	600	500	100	99.2877	98.7179	99.7150

5.2. Performance Metrics

The performance metrics used to evaluate the efficiency of PPFEdSL are described as follows.

Feature Representation Accuracy: It is the accuracy of feature extraction from the BITvehicle Image dataset using CNN.

Privacy Loss: It refers to the loss of privacy during the SL-vehicle client model sharing among vehicles and edges.

Computation Cost: It is the total amount of computation resources vehicles need to execute the Initial layers of the CNN and lightweight differential privacy process.

Robustness Accuracy: It refers to the heterogeneous data handling efficiency of PPFedSL over large-scale IoV non-IID datasets.

Learning Accuracy: It is the ratio of the number of correct predictions during learning and testing to the total number of correct predictions.

Aggregation Accuracy: It is the accuracy of global model aggregation while ensuring the privacy and security in the network.

Convergence Rate: The global model takes the rate to converge over specific FL communication rounds.

Communication Efficiency: It measures the efficiency of communication between end-edge-cloud during the SL and FL model-sharing process.

5.3. Experimental Results

The comparison of experimental results in Table 7 for computation overhead, latency, and accuracy justifies why SL is preferable over FL, VFL, and SMPC in an IoV-enabled smart city environment.

The results demonstrate that SL outperforms FL, VFL, and SMPC in terms of computational efficiency and latency while accomplishing the highest accuracy at 93.1%. The lower overhead of SL is 45.8%, and the reduced learning time per batch is 4.23s, making it an ideal solution for real-time IoV applications, guaranteeing timely decision-making. Although other strategies like FL and SMPC obtain competitive accuracy, they incur significantly higher computational costs and latency, which may be inappropriate for resource-constrained IoV environments. On the other hand, VFL shows better performance in terms of computation cost, but it needs higher learning time, hindering its scalability in a dynamic IoV environment. Moreover, layer-wise partitioning of SL shrinks both computational and communication overhead at the vehicle level, assuring effective learning on resource-limited devices. Its early unique feature extraction at the edge facilitates low-latency decision-making in real-time IoV applications.

Table 7 SL Empirical Evaluation Results over FL, VFL, and SMPC

Method	Computation Overhead (%)	Latency (s)	Total Learning Time per Batch (s)	Model Accuracy (%)
FL	80.5%	1.35	5.92	92.4%
VFL	65.2%	1.12	6.45	90.8%
SMPC	92.7%	2.81	9.74	91.2%
SL	45.8%	0.78	4.23	93.1%

Table 8 Dynamic Split Point Validation Under Different IoV Resource Scenarios

Scenario (Vehicle Resource Level (Rv))	DSPt Position	CPU Usage (%)	Memory Usage (MB)	Processing Time (s)	Total Training Time (s)	Model Accuracy (%)
Low Rv (Limited Resources)	Early Split (CNN-1)	38.7%	250 MB	0.59	4.01	92.5%
Moderate Rv (Balanced Resources)	Mid Split (CNN-2)	45.8%	310 MB	0.78	4.23	93.1%
High Rv (Sufficient Resources)	Late Split (CNN-3)	68.2%	420 MB	1.15	5.42	91.8%

Results of table 8 empirically validate the selection of dynamic split point, DSPt, under varying vehicle resource Rv conditions, demonstrating its effectiveness in shrinking latency, optimizing resource exploitation, and keeping high learning performance for time-sensitive IoV-enabled smart city application scenarios. Using DSPt, vehicles offload complex CNN layers according to their resource level to the edge, shrinking the burden while maintaining high accuracy.

Table 9 depicts the resource constraint formulation scenario and its results, demonstrating that the selection of DSPt effectively meets the real-world computational constraints in IoV. The proposed PPFedSL ensures efficient allocation of resources, low latency, and good model performance, which are essential for real-time collaboration across vehicles and edges.

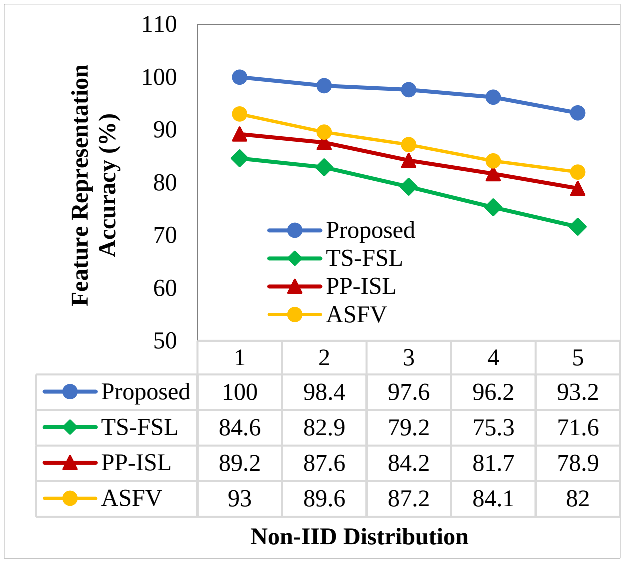
The experimental results of table 10 depict that DSPt-based SL significantly minimizes computational overhead by 45.8% and latency by 0.78s compared to pretrained models that are ResNet-based edge learning with 72.4% of overhead and 1.42s of latency and transformer-driven FL with 85.6% of overhead and 1.75s of latency. Although the transformer-FL accomplishes a high level of accuracy by 95.1%, the proposed SL model also maintains a competitive accuracy of 93.1% with a 45.8% lower overhead, making it more efficient for IoV applications. Additionally, SL obtains the fastest training time of 4.23s, ensuring real-time, timely decision-making in latency-sensitive IoV environments. Moreover, these findings efficiently validate the effectiveness of dynamic split point, which enabled SL to optimize resource utilization while obtaining high model performance.

Table 9 Resource Constraint Formulation of Vehicle and Edge

Scenario (Rv, Re)	DSPt Position	Vehicle CPU Usage (%)	Edge CPU Usage (%)	Memory Usage (MB)	Edge Processing Time (s)	Total Training Time (s)	Model Accuracy (%)
Low Rv, High Re	Early Split (CNN-1)	22.4%	68.7%	250 MB	0.59	4.01	92.5%
Moderate Rv, Moderate Re	Mid Split (CNN-2)	35.1%	45.8%	310 MB	0.78	4.23	93.1%
High Rv, Low Re	Late Split (CNN-3)	61.3%	27.5%	420 MB	1.15	5.42	91.8%

Table 10 Comparison of Proposed SL with Pre-Trained Models

Method	Model Type	Computational Overhead (%)	Latency (s)	Total Training Time (s)	Model Accuracy (%)
ResNet-based Edge Learning	Deep CNN	72.4%	1.42	6.15	94.3%
Transformer-driven FL	Self-Attention	85.6%	1.75	7.89	95.1%
Proposed DSPt-based SL	Adaptive CNN	45.8%	0.78	4.23	93.1%

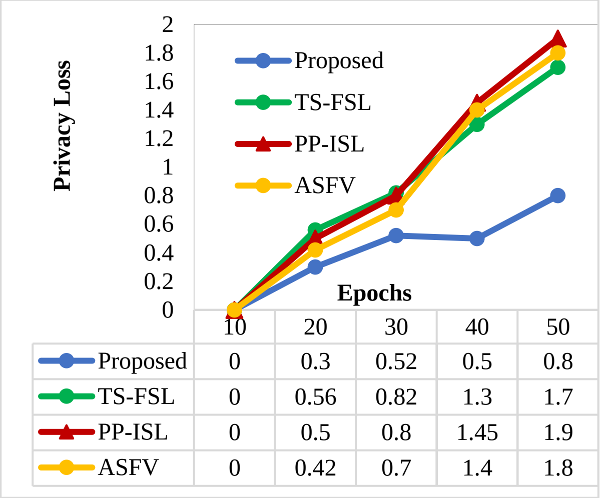


	1	2	3	4	5
Proposed	100	98.4	97.6	96.2	93.2
TS-FSL	84.6	82.9	79.2	75.3	71.6
PP-ISL	89.2	87.6	84.2	81.7	78.9
ASFV	93	89.6	87.2	84.1	82

Figure 10 Feature Representation Accuracy

Figure 10 depicts the accuracy of the feature representation of the proposed PPFedSL using CNN and existing models. The results are obtained by varying the number of non-IID distributions from 1 to 5. By partially running the CNN at vehicles and edges, the proposed model effectively handles the heterogeneity of large-scale massive IoV information and its resource-constrained nature. Therefore, the proposed model improves the feature representation accuracy by 100 for one number of non-IID. After point 1, the feature representation accuracy decreases in the proposed model, as the high number of non-IID makes the model more complex, and there is some loss in split learning accuracy.

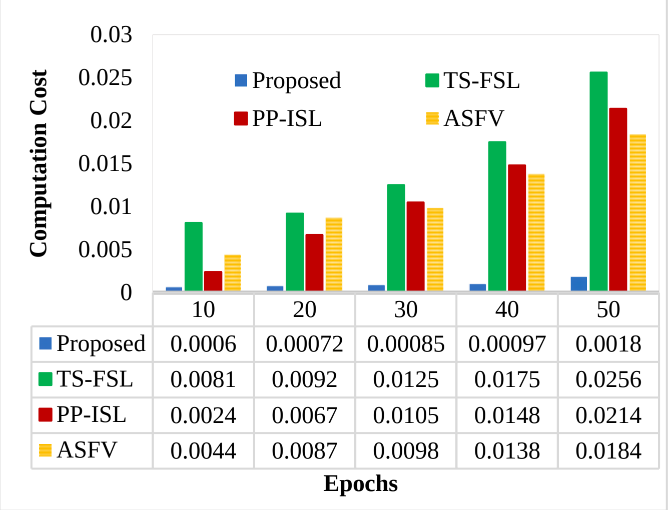
For example, the CNN attains 98.4% and 93.2% feature representation accuracy for non-IID distributions 2 and 5, respectively. However, the PPFedSL outperforms existing TS-FSL, PP-ISL, and ASFV by effectively implementing the SL across vehicles and edges. Although the ASFV utilizes adaptive model splitting to execute the training process parallelly, it fails to handle the non-IID distribution more effectively than PPFedSL. For instance, the CNN in the proposed model enhances the feature representation accuracy by 21.6%, 14.3%, and 11.2% when compared with TS-FSL, PP-ISL, and ASFV, respectively, for non-IID scenarios.



	10	20	30	40	50
Proposed	0	0.3	0.52	0.5	0.8
TS-FSL	0	0.56	0.82	1.3	1.7
PP-ISL	0	0.5	0.8	1.45	1.9
ASFV	0	0.42	0.7	1.4	1.8

Figure 11 Epochs Vs Privacy Loss

Figure 11 obtains the results of privacy loss of PPFedSL by comparing it with existing TS-FSL, PP-ISL, and ASFV protocols. The results are obtained by adjusting the epochs from 10 to 50. At the point of 10 epochs, all algorithms obtain no privacy loss. After point 10, each escalates the privacy loss gradually with increasing the number of epochs. For example, the proposed PPFedSL accomplishes 0.3 and 1.3 privacy loss for 20 and 50 epochs, respectively. The proposed model leverages a lightweight differential privacy algorithm to safeguard the intermediate data as gradients from leakage transferred among vehicles and edges. Careful injection of Laplace noise into the gradients during training enhances robust privacy while maintaining model utility. Consequently, it shrinks the privacy loss and mitigates the privacy risks. By keeping the privacy loss to a minimum, the SL model shrinks the impact of Laplace noise addition on model accuracy and convergence stability. In other words, reducing loss of privacy assists in maintaining high accuracy in model sharing. However, increasing the number of epochs in learning makes the proposed PPFedSL more complex, resulting in some privacy loss. However, the proposed model minimizes privacy loss more than the existing solutions, as shown in Figure 10. For instance, the proposed model minimizes the privacy loss by 0.9, 1.1, and 1 compared to TS-FSL, PP-ISL, and ASFV, respectively, for 50 epochs.

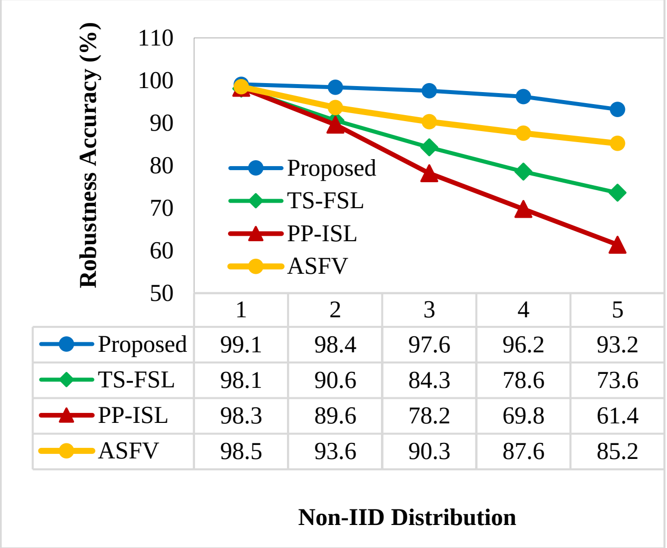


	10	20	30	40	50
■ Proposed	0.0006	0.00072	0.00085	0.00097	0.0018
■ TS-FSL	0.0081	0.0092	0.0125	0.0175	0.0256
■ PP-ISL	0.0024	0.0067	0.0105	0.0148	0.0214
■ ASFV	0.0044	0.0087	0.0098	0.0138	0.0184

Figure 12 Epochs Vs Computation Cost

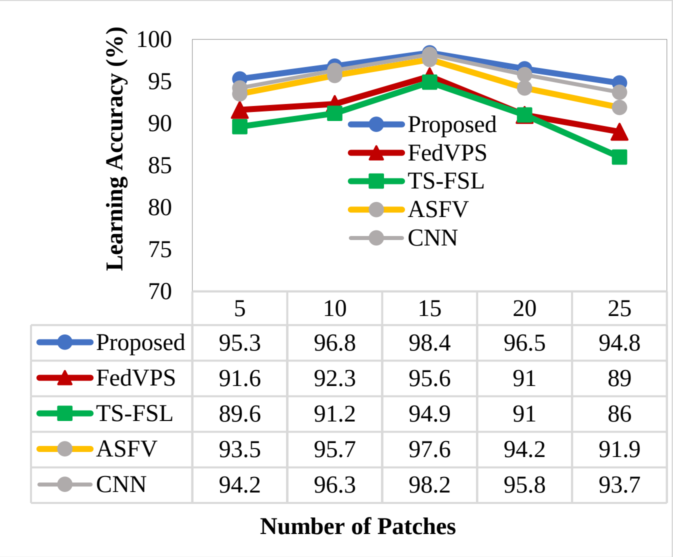
Figure 12 compares the communication cost results of proposed PPFedSL, TS-FSL, PP-ISL, and ASFV under different epochs. All models increase the computation cost by varying the epochs from 10 to 50. For instance, the PPFedSL accomplishes 0.0006 and 0.0018 for 10 and 50 epochs, respectively. The reason is that increasing the number of epochs typically leads to more parameter updates and increases the computation cost, particularly for larger SL models with a high number of CNN parameters. However, the dynamic split point estimation according to vehicle resources effectively adjusts the number of layers. Also, it handles computationally intensive CNN layers at the edges, significantly minimizing the computation cost of PPFedSL compared to existing strategies. This process enhances the proposed model's efficiency without imposing high costs. For example, the PPFedSL decreases the computation cost value by 0.0238, 0.0196, and 0.0166 than the existing TS-FSL, PP-ISL, and ASFV, respectively, for 50 numbers of epochs.

Figure 13 illustrates the robustness accuracy comparison results of proposed TS-FSL, PP-ISL, and ASFV protocols obtained by adjusting the non-IID distribution from 1 to 5. The robustness accuracy gradually decreases by varying the non-IID from low to high. For instance, the proposed model decreases the robustness accuracy by 5.9% when varying the non-IID distribution from 1 to 5. Generally, the high number of non-IID distributions affects the resource utilization capability of vehicles, resulting in unbalanced data. Efficiently partitioning the BITVehicle dataset into balanced sub-models and processing them locally at the vehicles, the proposed PPFedSL enables effective handling of non-IID data distribution. In contrast, conventional methods fail to integrate such dataset partitioning, shrinking robustness and degrading accuracy in their performance. Moreover, the proposed model improves the robustness and accuracy compared to the existing algorithms, as shown in Figure 13. For example, the proposed model improves robustness accuracy by 19.6%, 31.8 %, and 8% compared with TS-FSL, PP-ISL, and ASFV for non-IID scenarios.



	1	2	3	4	5
Proposed	99.1	98.4	97.6	96.2	93.2
TS-FSL	98.1	90.6	84.3	78.6	73.6
PP-ISL	98.3	89.6	78.2	69.8	61.4
ASFV	98.5	93.6	90.3	87.6	85.2

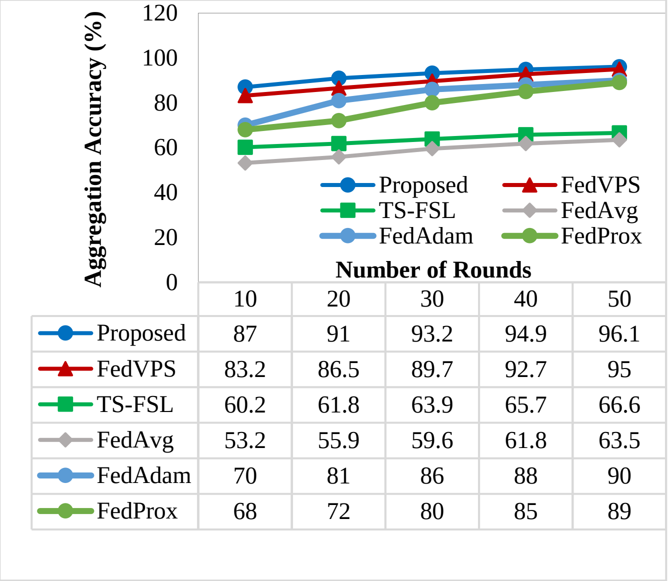
Figure 13 Non-IID Distribution Vs Robustness Accuracy



	5	10	15	20	25
Proposed	95.3	96.8	98.4	96.5	94.8
FedVPS	91.6	92.3	95.6	91	89
TS-FSL	89.6	91.2	94.9	91	86
ASFV	93.5	95.7	97.6	94.2	91.9
CNN	94.2	96.3	98.2	95.8	93.7

Figure 14 Number of Patches Vs Learning Accuracy

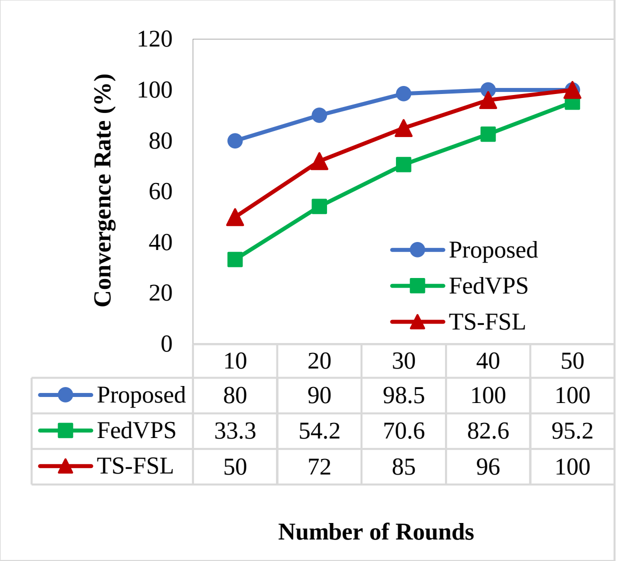
Figure 14 compares the proposed PPFedSL, FedVPS, TS-FSL, ASFV, and baseline CNN. The results are compared for different numbers of learning patches. From points 5 to 15, the proposed model increases the learning accuracy and decreases after point 15. The reason is that a significant number of patches, like 15, can improve the performance of vision transformers. Increasing the patches after 15 can impact the learning efficiency of the vision transformer, resulting in some loss. For example, the PPFedSL attains 95.3% and 98.4% learning accuracy for 5 and 15 patches scenarios, respectively. However, it decreases the value by 3.6% at point 25 compared to the result at point 15. However, the proposed model improves the learning accuracy than the other model in existing and based line works owing to utilizing FL. Implementing effective edge and cloud collaborative learning improves the learning accuracy of PPFed SL. Unlike that, the existing works contribute to improving the accuracy using global parameters provided by the cloud. Thus, it increases the proposed model's accuracy than existing works. For example, the PPFedSL improves learning accuracy by 9.4%, 8.8%, 2.9, and 1.1% compared to the FedVPS, TS-FSL, ASFV, and CNN, respectively, at the point of 15 number of patches.



	10	20	30	40	50
Proposed	87	91	93.2	94.9	96.1
FedVPS	83.2	86.5	89.7	92.7	95
TS-FSL	60.2	61.8	63.9	65.7	66.6
FedAvg	53.2	55.9	59.6	61.8	63.5
FedAdam	70	81	86	88	90
FedProx	68	72	80	85	89

Figure 15 Number of Rounds Vs Aggregation Accuracy

Figure 15 compares the aggregation accuracy results of proposed PPFedSL, FedVPS, TS-FSL, FedAvg, FedAdam, and FedProx under different communication rounds. Figure 15 shows that all models slightly increase the aggregation accuracy by varying the number of communication rounds from 10 to 50. The reason is that increasing the number of rounds provides more local model parameters for FL aggregation, which improves accuracy considerably. For example, the PPFedSL escalates the aggregation accuracy by 9.1% when increasing the number of rounds from 10 to 50. However, the proposed model accomplishes high aggregation accuracy despite minimal rounds due to the advantage of FedSGD in aggregation. Unlike that, the existing models use FedAvg algorithms in which a simple weighted average is taken among the received local model, resulting in poor aggregation accuracy. Moreover, the proposed model outperforms the existing models. For instance, the proposed PPFedSL improves the aggregation accuracy by 1.1%, 29.5%, 32.6%, 6.1%, and 7.1% compared to the FedVPS, TS-FSL, FedAvg, FedAdam, and FedProx, respectively, for 50 numbers of communication rounds.

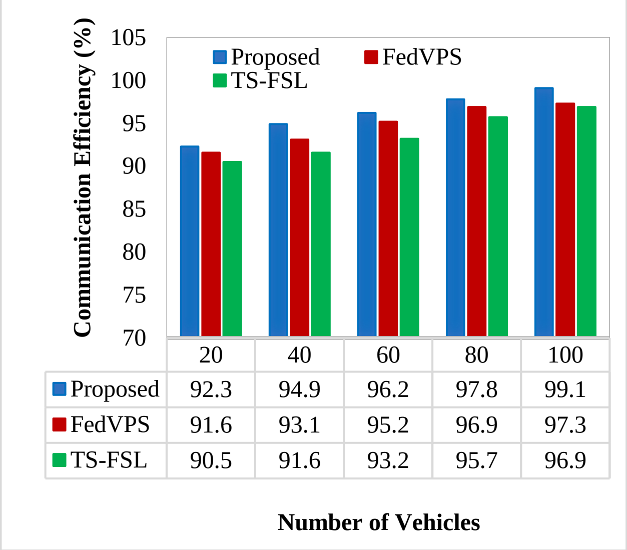


	10	20	30	40	50
Proposed	80	90	98.5	100	100
FedVPS	33.3	54.2	70.6	82.6	95.2
TS-FSL	50	72	85	96	100

Figure 16 Number of Rounds vs. convergence Rate

Figure 16 demonstrates the convergence rate results of the proposed PPFedSL, FedVPS, and TS-FSL for the different number of communication rounds. Each FL strategy necessitates the optimal number of communication rounds for model convergence. For example, the proposed PFedSL requires 40 communication rounds to accomplish a model convergence rate of 100, as the proposed model minimizes the number of communication rounds in model convergence by effectively implementing the SL and FL among IoV entities within tiers. The SL-enabled collaborative learning among vehicles and edges effectively handles the massive heterogeneous data generation and resource-constrained issues of large-scale IoVs. In contrast, the FL among edge and cloud can improve the FL learning and aggregation accuracy by effectively inputting the unique features without raw input. The existing TS-FSL model needs 50 communication rounds for model convergence, and the FedVPS needs more than 50.

Figure 17 shows the communication efficiency of PPFedSL, FedVPS, and TS-FSL compared under different numbers of IoV vehicle scenarios. The communication efficiency is increased by varying the number of vehicles from 20 to 100. This is because many vehicles provide more accurate information about the driving environment, improving IoV data-sharing performance. For example, the proposed PPFedSL accomplishes 92.3% and 99.1% of communication efficiency for 20 and 100 vehicles, respectively. Unlike existing models, the PPFedSL improves communication efficiency by keeping the raw information at local vehicles and making critical decisions at edges without sacrificing security and privacy. Thus, unlike other strategies, it enhances communication efficiency and IoV data-sharing performance without imposing significant latency. For example, the PPFedSL improves communication efficiency by 1.8% and 2.2% compared to the FedVPS and TS-FSL, respectively, when 100 vehicles are presented in the network.



	20	40	60	80	100
🟦 Proposed	92.3	94.9	96.2	97.8	99.1
🟥 FedVPS	91.6	93.1	95.2	96.9	97.3
🟩 TS-FSL	90.5	91.6	93.2	95.7	96.9

Figure 17 Number of Vehicles Vs Communication Efficiency

6. CONCLUSION

In this paper, PPFedSL has been proposed over heterogeneous large-scale IoV in a smart city environment. By leveraging the SL with lightweight cryptography primitives and FL with hybrid cryptography, the PPFedSL enables a distributed collaborative learning model among vehicle-edge-cloud while keeping the privacy of sensitive data. Thus, this proposed strategy effectively handles the heterogeneity of non-IID data distribution and massive dataset generation of IoV without compromising privacy. By running the computational extensive layers of CNN decided using the dynamic split point at edges, the PPFedSL manages the resource deficiency concerns of vehicles, and the critical decision-making at the edge level also effectively meets the latency constraints. Consequently, the PPFedSL model integrates an adaptive deep learning strategy, VT, to improve the FL learning accuracy over heterogeneous models. Moreover, the PPFedSL enables the smart city infrastructure for seamless data exchange with a higher level of privacy and enhanced performance efficiency. Finally, the experimental results show the superiority of the proposed PPFedSL in terms of different performance metrics. The results demonstrate that the proposed PPFedSL enhances the heterogeneous robustness efficiency by 19.6% and learning accuracy by 8.8% when compared with existing TS-FSL, whereas it incurs a higher convergence rate with 40 FL communication rounds due to the privacy-preserving heterogeneous handling structure using both SL and FL.

REFERENCES

[1] B. Ji, X. Zhang, S. Mumtaz, C. Han, C. Li, H. Wen, and D. Wang, “Survey on the internet of vehicles: Network architectures and applications,” IEEE Commun. Standards Mag., vol. 4, no. 1, pp. 34–41, 2020.

[2] W. Duan, J. Gu, M. Wen, G. Zhang, Y. Ji, and S. Mumtaz, “Emerging technologies for 5G-IoV networks: applications, trends and opportunities,” IEEE Network, vol. 34, no. 5, pp. 283–289, 2020.

[3] S. M. Karim, A. Habbal, S. A. Chaudhry, and A. Irshad, “Architecture, protocols, and security in IoV: Taxonomy, analysis, challenges, and solutions,” Security and Commun. Networks, vol. 2022, 2022.

[4] H. Taslimasa, S. Dadkhah, E. C. P. Neto, P. Xiong, S. Ray, and A. A. Ghorbani, “Security issues in Internet of Vehicles (IoV): A comprehensive survey,” Internet of Things, vol. 100809, 2023.

[5] X. Xu, H. Li, W. Xu, Z. Liu, L. Yao, and F. Dai, “Artificial intelligence for edge service optimization in internet of vehicles: A survey,” Tsinghua Sci. Technol., vol. 27, no. 2, pp. 270–287, 2021.

[6] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. V. Poor, “Federated learning for internet of things: A comprehensive survey,” IEEE Commun. Surveys Tutorials, vol. 23, no. 3, pp. 1622–1658, 2021.

[7] Z. Du, C. Wu, T. Yoshinaga, K. L. A. Yau, Y. Ji, and J. Li, “Federated learning for vehicular internet of things: Recent advances and open issues,” IEEE Open J. Comput. Soc., vol. 1, pp. 45–61, 2020.

[8] V. P. Chellapandi, L. Yuan, C. G. Brinton, S. H. Żak, and Z. Wang, “Federated learning for connected and automated vehicles: A survey of existing approaches and challenges,” IEEE Trans. Intell. Vehicles, 2023.

[9] Y. Bao, W. Qiu, X. Cheng, and J. Sun, “Fine-grained data sharing with enhanced privacy protection and dynamic users group service for the IoV,” IEEE Trans. Intell. Transp. Syst., 2022.

[10] U. Bodkhe and S. Tanwar, “P2IOV: Privacy preserving lightweight secure data dissemination scheme for internet of vehicles,” in 2021 IEEE Globecom Workshops (GC Wkshps), 2021, pp. 1–6.

[11] M. Jamjoom, H. Abulkasim, and S. Abbas, “Lightweight authenticated privacy-preserving secure framework for the Internet of vehicles,” Security and Commun. Networks, vol. 2022, 2022.

[12] D. M. Manias and A. Shami, “Making a case for federated learning in the internet of vehicles and intelligent transportation systems,” IEEE Network, vol. 35, no. 3, pp. 88–94, 2021.

[13] X. Zhou, W. Liang, J. She, Z. Yan, I. Kevin, and K. Wang, “Two-layer federated learning with heterogeneous model aggregation for 6G supported internet of vehicles,” IEEE Trans. Veh. Technol., vol. 70, no. 6, pp. 5308–5317, 2021.

[14] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, “Federated learning for data privacy preservation in vehicular cyber-physical systems,” IEEE Netw., vol. 34, no. 3, pp. 50–56, May/Jun. 2020.

[15] P. Zhao, Y. Huang, J. Gao, L. Xing, H. Wu, and H. Ma, “Federated learning based collaborative authentication protocol for shared data in social IoV,” IEEE Sensors J., vol. 22, no. 7, pp. 7385–7398, Apr. 2022.

[16] X. Yuan et al., “A federated bidirectional connection broad learning scheme for secure data sharing in internet of vehicles,” China Commun., vol. 18, no. 7, pp. 117–133, Jul. 2021.

[17] X. Li, L. Cheng, C. Sun, K. Y. Lam, X. Wang, and F. Li, “Federated learning-empowered collaborative data sharing for vehicular edge networks,” IEEE Netw., vol. 35, no. 3, pp. 116–124, May/Jun. 2021.

[18] M. Gawas, H. Patil, and S. S. Govekar, “An integrative approach for secure data sharing in vehicular edge computing using blockchain,” Peer-to-Peer Netw. Appl., vol. 14, pp. 1–18, 2021.

[19] M. Nakanoya, J. Im, H. Qiu, S. Katti, M. Pavone, and S. Chinchali, “Personalized federated learning of driver prediction models for autonomous driving,” arXiv preprint arXiv:2112.00956, 2021.

[20] D. Su, Y. Zhou, and L. Cui, “Boost decentralized federated learning in vehicular networks by diversifying data sources,” in 2022 IEEE 30th Int. Conf. Network Protocols (ICNP), 2022, pp. 1–11.

[21] A. Shamsian, A. Navon, E. Fetaya, and G. Chechik, "Personalized federated learning using hypernetworks," in Int. Conf. on Machine Learning, 2021, pp. 9489–9502.

[22] W. Y. B. Lim, J. Huang, Z. Xiong, J. Kang, D. Niyato, X. S. Hua, and C. Miao, "Towards federated learning in UAV-enabled internet of vehicles: A multi-dimensional contract-matching approach," IEEE Trans. Intell. Transp. Syst., vol. 22, no. 8, pp. 5140–5154, 2021.

[23] X. Yuan, J. Chen, N. Zhang, C. Zhu, Q. Ye, and X. S. Shen, "FedTSE: Low-cost federated learning for privacy-preserved traffic state estimation in IoV," in IEEE INFOCOM 2022-IEEE Conf. on Computer Commun. Workshops (INFOCOM WKSHPS), 2022, pp. 1–6.

[24] H. K. Hangdong, M. Bo, D. H. Darong, and Z. D. Zhaoyang, "FedVPS: Federated learning for privacy and security of internet of vehicles on non-IID data," in 2023 IEEE 12th Data Driven Control and Learning Systems Conf. (DDCLS), 2023, pp. 178–183.

[25] F. Liang, Q. Yang, R. Liu, J. Wang, K. Sato, and J. Guo, "Semi-synchronous federated learning protocol with dynamic aggregation in internet of vehicles," IEEE Trans. Veh. Technol., vol. 71, no. 5, pp. 4677–4691, 2022.

[26] Y. Wang, L. Xiong, X. Niu, Y. Wang, and D. Liang, "A federated learning-based privacy-preserving data sharing scheme for internet of vehicles," in Int. Conf. on Frontiers in Cyber Security, Singapore, 2022, pp. 18–33.

[27] W. Jin, Y. Yao, S. Han, C. Joe-Wong, S. Ravi, S. Avestimehr, and C. He, "FedML-HE: An efficient homomorphic-encryption-based privacy-preserving federated learning system," arXiv preprint arXiv:2303.10837, 2023.

[28] C. Fang, Y. Guo, Y. Hu, B. Ma, L. Feng, and A. Yin, "Privacy-preserving and communication-efficient federated learning in internet of things," Comput. & Security, vol. 103, p. 102199, 2021.

[29] R. Parekh, N. Patel, R. Gupta, N. K. Jadav, S. Tanwar, A. Alharbi, and M. S. Raboaca, "GEFL: Gradient encryption-aided privacy-preserved federated learning for autonomous vehicles," IEEE Access, vol. 11, pp. 1825–1839, 2023.

[30] P. Agbaje, A. Anjum, A. Mitra, S. Hounsinou, E. Nwafor, and H. Olufowobi, "Privacy-preserving intrusion detection system for internet of vehicles using split learning," in Proc. IEEE/ACM 10th Int. Conf. on Big Data Computing, Applications and Technologies, 2023, pp. 1–8.

[31] M. Wu, G. Cheng, D. Ye, J. Kang, R. Yu, Y. Wu, and M. Pan, "Federated split learning with data and label privacy preservation in vehicular networks," IEEE Trans. Veh. Technol., 2023.

[32] A. Padaria, A. A. Mehta, N. K. Jadav, S. Tanwar, D. Garg, A. Singh, and G. Sharma, "Traffic sign classification for autonomous vehicles using split and federated learning underlying 5G," IEEE Open J. Veh. Technol., 2023.

[33] M. Wu, G. Cheng, D. Ye, J. Kang, R. Yu, Y. Wu, and M. Pan, "Federated split learning with data and label privacy preservation in vehicular networks," IEEE Trans. Veh. Technol., 2023.

[34] X. Qiang, Z. Chang, C. Ye, T. Hamalainen, and G. Min, "Split federated learning empowered vehicular edge intelligence: Adaptive parallel design and future directions," arXiv preprint arXiv:2406.15804, 2024.

[35] X. Qiang, Z. Chang, Y. Hu, L. Liu, and T. Hämäläinen, "Adaptive and parallel split federated learning in vehicular edge computing," IEEE Internet Things J., 2024.

[36] Y. Zhao, J. Zhao, M. Yang, T. Wang, N. Wang, L. Lyu, and K. Y. Lam, "Local differential privacy-based federated learning for internet of things," IEEE Internet Things J., vol. 8, no. 11, pp. 8836–8853, 2020.

[37] K. S. Patil, I. Mandal, and C. Rangaswamy, "Hybrid and adaptive cryptographic-based secure authentication approach in IoT-based applications using hybrid encryption," Pervasive Mobile Comput., vol. 82, p. 101552, 2022.

[38] BITVehicle. Kaggle: Your Machine Learning and Data Science Community. [Online]. Available: https://www.kaggle.com/datasets/kuanghangdong/bitvehicle <Accessed: 16/October/2024>.

Authors

Komala Soares graduated from University B.D.T. College of Engineering, Davanagere in Electronics and Telecommunication engineering in 1997 and obtained M.E. in Digital Electronics from S.D.M. College of Engineering, Dharward in 1999. Ms Komala Soares was employed at Padre conceicao College of Engineering, Goa. During her tenure, she held various positions and was promoted to Associate professor. In 2009. She accepted the job as Lecturer, Government Polytechnic, Altinho-Panaji-Goa. In 2015. She was promoted as Head of Department, Electronics and Communication Engineering in 2011. She completed her M.S. in Artificial Intelligence and Machine Learning (online) in 2022. And Joined Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune as research scholar. Mrs. Soares is member of many prestigious technical society like ISTE, IE, ISHRE. Her research interests include Machine Learning application in Vehicular Networks, artificial intelligence, and transport challenges.

Dr. Arundhati A. Shinde Graduated in Industrial Electronics from Shivaji University and obtained Post graduation in in E & TC graduation from The Savitribai Phule Pune University. Also obtained Ph.D in Electronics from Bharati Vidyapeeth (Deemed to be University). Currently she is working as an Head of Department and Professor in Electronics & Communication Engineering, Bharati Vidyapeeth Deemed to be University College of Engineering Pune India. She has 5 years industrial experience, 32 years teaching experience and 7 years research experience. She is Recipient of 'Seva Gaurav Puraskar' conferred by Bharati Vidyapeeth University and published more than 25 research articles in peer reviewed journals and conferences.

Dr. Mangal Patil obtained her B.E. in Electronics Engineering from Shivaji University, Kolhapur, India, in 1999. She subsequently completed her M.E. and PhD at Bharati Vidyapeeth Deemed to be University, Pune, India, in 2006 and 2019, respectively. She currently holds the position of Associate Professor in the Department of Electronics & Communication Engineering at Bharati Vidyapeeth Deemed to be University College of Engineering, Pune, India. With an extensive teaching experience of 20+ years, her research domains encompass Speech Processing, Image Processing, Communication Systems, and Artificial Intelligence and more. She has contributed to over 60 peer-reviewed research papers published in prominent journals and international conferences, focusing on cutting-edge methodologies and emerging technologies.

How to cite this article:

Komala Soares, Arundhati A. Shinde, Mangal Patil, "PPFedSL: Privacy Preserving Split and Federated Learning Enabled Secure Data Sharing Model for Internet of Vehicles in Smart City", International Journal of Computer Networks and Applications (IJCNA), 12(2), PP: 154-177, 2025, DOI: 10.22247/ijcna/2025/11.