Sigma J Eng Nat Sci, Vol. 44, No. 3, pp. 1846–1858, June, 2026

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2026.2067

Research Article

Enhancing intrusion detection in the internet of vehicles through feature reduction

Rakesh A. PARMAR[1,*], Dr. Nirav BHATT[1], Dr. Pranav RAVAL[1]

[1]School of Engineering, RK University L. E. College, Morbi, 360020, India

ARTICLE INFO

Article history
Received: 28 April 2025
Revised: 01 June 2025
Accepted: 16 September 2025

Keywords:
Cybersecurity; Feature Reduction; Internet of Vehicles; Intrusion Detection; Machine Learning

ABSTRACT

The internet of vehicles faces critical cybersecurity threats such as denial-of-service and spoofing attacks, jeopardizing safety and privacy in connected transport. This study investigates machine learning–based intrusion detection for multiclass attack recognition using a novel vehicle network dataset. We evaluate six models with feature reduction through Recursive Feature Elimination and Principal Component Analysis, a strategy not previously applied to Internet of Vehicles security data. Recursive Feature Elimination enhanced Random Forest performance, achieving 99.7% accuracy and an F1-score of 0.73 while reducing training time by 86.2%. Principal Component Analysis preserved 95% variance but reduced effectiveness (F1-score 0.37). Our results indicate that feature reduction not only improves real-time detection time but also outperforms existing Internet of Vehicles intrusion detection research on static dataset in accuracy and performance. This paper demonstrates a new, scalable method for autonomous and connected vehicle's safety. Future research will explore larger datasets and deep learning integration.

Cite this article as: Parmar RA, Bhatt N, Raval P. Enhancing intrusion detection in the internet of vehicles through feature reduction. Sigma J Eng Nat Sci 2026;44(3):1846–1858.

INTRODUCTION

The Internet of Vehicles transforms the transportation through real-time traffic management, autonomous driving, and vehicle-to-vehicle communication on interconnected networks and making it smart. This will improve mobility but introduce cybersecurity threats, such as denial-of-service and spoofing attacks, compromising privacy and safety in intelligent transportation [1].

This work should prove valuable for automakers, fleet operators, and regulators alike. By strengthening defenses against cyber threats that could suddenly disrupt braking, steering, or other critical systems, it may help make connected and autonomous vehicles noticeably more dependable in everyday use. The optimized models, in particular, support real-time attack detection—something that feels increasingly essential for safe communication in smart cities and large-scale logistics operations. In the longer run, findings like these could quietly shape future cybersecurity guidelines that many people working in intelligent transportation care about.

***Corresponding author.**
*E-mail address: rakeshparmar.it@lecollege.ac.in

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic

Published by Yıldız Technical University Press, İstanbul, Turkey

Looking at the broader IoV cybersecurity literature, there seems to be growing agreement on the need for tailored intrusion detection approaches. For instance, attack traces captured from a real 2019 Ford vehicle's CAN bus offer a sobering glimpse into the practical risks facing connected cars. [2]. Studies have shown strong detection of denial-of-service attacks on IoV networks [3], yet many rely on older, static datasets such as NSL-KDD that lack the timing dynamics and message patterns typical of actual vehicle environments[4]. Techniques like recursive feature elimination or principal component analysis have improved efficiency in general IoT intrusion detection, but they remain largely unexplored on modern IoV datasets such as CICIoV2024 [5-7] Similarly, combinations of Random Forest or XGBoost with blockchain have delivered promising results on benchmark datasets like CICIDS2017[8]. while lightweight solutions developed for UAVs provide useful inspiration—though they don't always translate directly to the automotive domain[9]. In that sense, this study takes an early step toward applying thoughtful feature handling strategies specifically to CICIoV2024, helping to bridge some of the persistent gaps between generic intrusion detection research and the unique realities of vehicle networks.

Following are the studies on intrusion detection. Feature extraction using hybrid techniques along with meta-heuristics methods reduces false positives in IoT [10-12]. We found study on NSL-KDD dataset that uses Principal component analysis with optimization that has high accuracy but same testing on IoV environment is absent [13]. Ensemble and deep learning models detect complex patterns but are computationally expensive for resource limited vehicular environment [14-16]. Other work has looked at handling class imbalance through random undersampling and multi-layer feature selection, yielding better results on both IoT and some IoV-related datasets [17,18]. Optimized machine learning for servers demonstrates efficiency [19]. Smart city and IoV-specific models support multiclass detection but need optimization for resource constraints [20-22]. As per our knowledge no prior work applies principal component analysis or recursive feature elimination to CICIoV2024 for multiclass detection so this research fills the gap for real-time use.

This study aims to develop an efficient, accurate multiclass intrusion detection system for IoV using the CICIoV2024 dataset, applying feature reduction techniques—recursive feature elimination and principal component analysis—to optimize machine learning models for real-time cybersecurity in resource-constrained vehicles. It investigates machine learning optimization for multiclass detection on CICIoV2024, which records realistic Controller Area Network traffic from a 2019 Ford car. It includes 1.4 million records and 156 features, preprocessed and balanced using synthetic minority oversampling technique, yielding 17,025 samples across six classes: benign traffic, denial-of-service, gas, speed, steering wheel, and RPM [2].

We used six machine learning models named random forest, adaptive boosting, logistic regression, support vector machine, deep neural network, and extreme gradient boosting. These are evaluated with principal component analysis and recursive feature elimination to optimize detection performance and computational efficiency. Three research questions guide this investigation:

- How do principal component analysis and recursive feature elimination impact detection across six IoV attack types given in the Dataset?
- Which machine learning classifiers with feature reduction balance accuracy and efficiency?
- Can lightweight models meet real-time demands of resource-constrained IoV environments?

The contributions are:

- Comparison of principal component analysis and recursive feature elimination for multiclass intrusion detection using CICIoV2024.
- Enhanced machine learning system through feature reduction with six classifiers for effective, precise detection.
- Metric-based visualizations and trade-off analysis for empirical study on real-time IoV intrusion detection.

In rest of the paper we discussed materials and methods used in the experiment in section 2, Section 3 presents experimental results we achieved, Section 4 is discussion on our findings, and Section 5 concludes with key findings and future directions.

MATERIALS AND METHODS

This section describes the methodology for optimizing multiclass intrusion detection using machine learning models in the Internet of Vehicles. We used CICIoV2024 dataset. We conducted experiment on a machine with an Intel i5-9300H CPU, 8GB RAM, and an NVIDIA GTX 1650 GPU and it's running Windows 11. We used Python 3.11.4 alongside scikit-learn 1.6.1 library using the publicly available CICIoV2024 dataset [2]. Data preprocessing steps like duplicate removal, label encoding, SMOTE implementation, and normalization are discussed in detail in preprocessing section below. Random seeds were set to 42 for all splits and model training to ensure reproducibility. Full hyperparameter configurations for all machine learning models are provided in this section only. Figure 1 illustrates the workflow, and tables summarize key configurations used in experiment.

Overview

Our approach is preprocessing the CICIoV2024 dataset, reducing feature dimensionality, training six machine learning models, and evaluating performance for detecting six attack types: benign traffic, denial-of-service, gas, speed, steering wheel, and RPM spoofing.

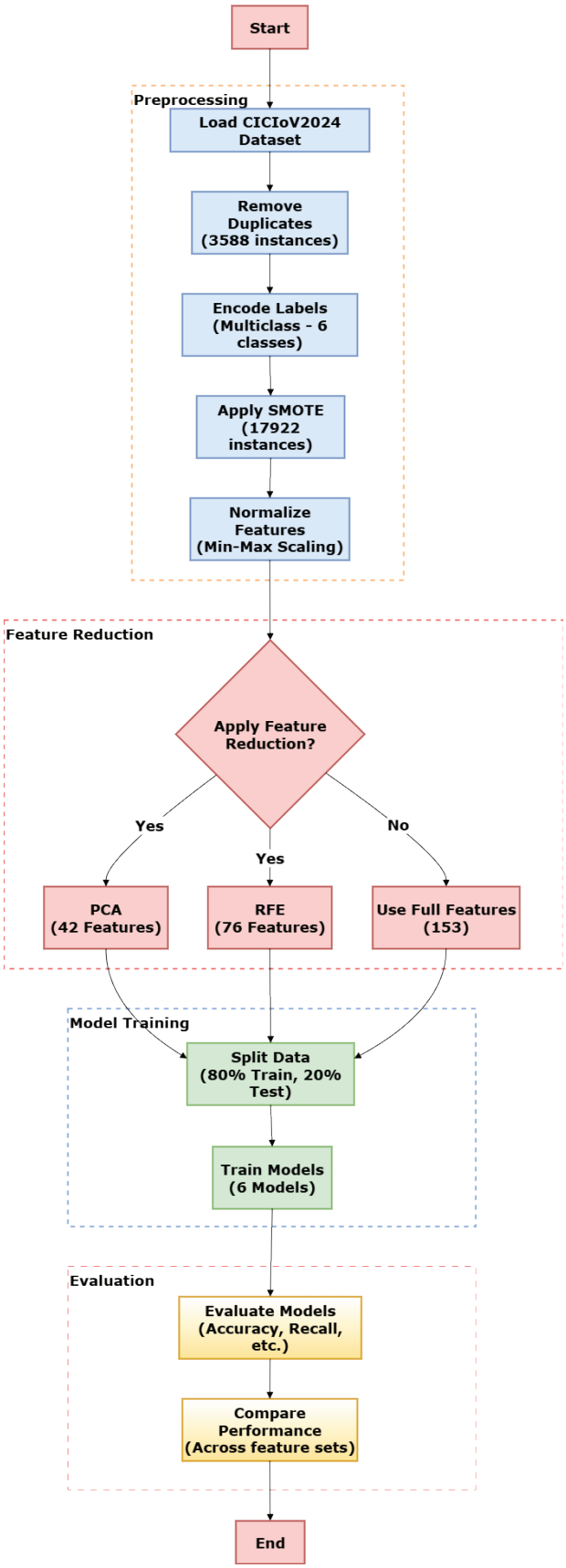

Figure 1. Flowchart of the proposed methodology for multiclass intrusion detection in IoV.

DATASET

The CICIoV2024 dataset, capturing network traffic from a 2019 Ford car's Controller Area Network bus, was used to simulate Internet of Vehicles scenarios, including benign traffic and attacks like denial-of-service and spoofing gas, speed, steering wheel, RPM. Initially it has 1,408,219 instances with 156 features, the dataset was reduced to 3,588 unique instances after removing duplicates so retaining 153 features. This deduplication aligns with Controller Area Network security studies [20]. Compared to datasets like ToN-IoT, which includes attacks like grey hole [23], CICIoV2024's vehicle-specific focus makes it suitable for this study. The distribution of class is imbalanced in the dataset it has 3,547 benign sample, 21 denial-of-service attacks, 2 gas spoofing, 10 RPM spoofing, 5 speed spoofing, and 3 steering wheel spoofing this makes preprocessing necessary [16].

Preprocessing

We used following preprocessing steps to handle redundancy, imbalance, and feature scaling in the dataset:

- **Duplicate Removal**: To prevent model bias, we Eliminated duplicates. That reduces instances from 1,408,219 to 3,588[6].
- **Normalization**: We apply Min-Max scaling to normalize features in the range of [0, 1], using:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}},$$

This ensures compatibility with scale-sensitive algorithms [15].

- **Class Imbalance Handling**: To balance classes we used Synthetic Minority Oversampling Technique that generates synthetic samples via interpolation technique:

$$x_{new} = x_i + \lambda \times (x_{zi} - x_i)$$

where x_{new} is the new synthetic sample, x_i is an original minority class instance, x_{zi} is one of its k-nearest neighbors, and λ is a random number between 0 and 1 (k=5, using scikit-learn's SMOTE). This yielded 2,837 instances per class, totaling 17,025 samples (2,837 × 6), ensuring equitable representation [16].

- **Data Splitting**: To maintain class distribution we split the dataset into 80% training (13,620 instances) and 20% testing (3,405 instances) using stratified sampling. [24].

FEATURE REDUCTION TECHNIQUES

Here we used two feature reduction techniques that will reduce high-dimensional vehicle data.

- **Principal Component Analysis (PCA)**: This technique transforms the features into uncorrelated principal components $Z = X * W$, where X is the data matrix (17,025 × 153), W is the eigenvector matrix, and Z is the

transformed matrix (17,025 × 42). A scree plot selected 42 features, retaining 95% of the variance, reducing the number of features by 72.55% (111 features removed) [3,4].

- **Recursive Feature Elimination (RFE)**: Iteratively removes the least important features using a Random Forest estimator (estimators=200, max depth=15). Feature importance was calculated using Gini impurity, defined as shown below, where p_i is the proportion of the class i at a node, summed over all classes C (6 for multiclass): $\text{Gini} = 1 - \sum_{i=1}^{C}(p_i)^2$ The Gini Importance for a feature j is calculated as follows, where T = 200 The number of trees, the inner sum is over the nodes n in tree t, that split on the feature j, and $\Delta i(n, j)$ is the decrease in Gini impurity at the node n:

$$\text{Gini Importance}_j = \frac{1}{T}\sum_{t=1}^{T}\sum_{n \in \text{nodes using } j} \Delta i(n, j)$$

RFE selected 76 features (49.67% reduction, 77 removed) [7,10,25]. The full 153-feature set served as a baseline [17].

Machine Learning Models

Six machine learning models were trained with hyperparameters tuned via grid search for reproducibility:

- **Logistic Regression**: max_iter=200, solver="saga", class_weight="balanced" [2].
- **Adaptive Boosting (AdaBoost)**: n_estimators=200, learning_rate=0.5, algorithm="SAMME" [2].
- **Deep Neural Network (MLPClassifier)**: Three hidden layers (64, 64, 64), activation="relu", max_iter=300 [12]
- **Random Forest**: n_estimators = 200, max_depth = 15, class_weight = "balanced". [18].
- **Support Vector Machine (SVM)**: Here C=1.0, kernel="rbf", gamma="scale"[9].
- **Extreme Gradient Boosting (XGBoost)**: max_depth=3, learning_rate=0.1, n_estimators=100 [19].

The models were evaluated using 5-fold cross-validation on three feature sets [24].

Evaluation Metrics

Performance was assessed using multiclass metrics:

- **Accuracy**: Overall accuracy of predictions, computed as:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$$

where TP is True Positives, TN is True Negatives, FP is False Positives, and FN is False Negatives.

- **Precision**: Ratio of correct positive predictions per class, macro-averaged:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Recall**: Ratio of true positives correctly predicted per class, macro-averaged:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1-Score**: Harmonic mean of recall and precision, macro-averaged:

$$\text{F1-Score} = 2 \times (\text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall})$$

These metrics ensure robust evaluation across attack types [5,13]. Reduction in training time supports real time detection in vehicle [24,26].

SUMMARY TABLE

Table 1 summarizes feature dimension reduction:

Table 1. Feature reduction summary

Technique	PCA	RFE
Original Features	153	153
Reduced Features	42	76
Variance Explained	95%	N/A
Reduction Percentage	72.55%	49.67%

This methodology evaluates intrusion detection capability using feature reduction and machine learning for Vehicular environment. Future research could explore adversarial robustness to enhance model reliability [27].

RESULTS AND DISCUSSION

This section evaluates the performance of six machine learning models—Random Forest, Adaptive Boosting, Logistic Regression, Deep Neural Network, Support Vector Machine, and Extreme Gradient Boosting—for multiclass intrusion detection in the Internet of Vehicles using the CICIoV2024 dataset. Models were assessed on three feature sets: original (153 features), PCA (42 features, 95% variance), and RFE (76 features, 49.67% reduction). Performance metrics—accuracy, precision, recall, and macro-averaged F1-score—were computed across six classes (benign traffic, denial-of-service, gas, speed, steering wheel, RPM) on a balanced dataset of 17,025 instances (via SMOTE) using 5-fold cross-validation. Training and testing times were analyzed to evaluate real-time suitability.

Performance Across Feature Sets

Table 2 and Figure 2 (F1-scores bar chart) show that Random Forest and Extreme Gradient Boosting were high in F1-scores for all feature sets, consistent with earlier IoV research [8,9]. RFE-76 maintained classification accuracy at the cost of reducing the feature space by 49.67%, as discussed in effective feature selection [16,25]. PCA-42 caused extreme F1-score drops, most notably for minority classes like gas and steering wheel, because of loss of discriminative

Table 2. F1-scores (macro-averaged) across feature sets

Model	No PCA (153)	PCA-42	RFE-76
Random forest	0.733052	0.370727	0.733052
Adaptive boosting	0.313984	0.279019	0.612632
Logistic regression	0.577425	0.415488	0.458700
Deep neural network	0.444327	0.370586	0.532912
Support vector machine	0.412492	0.391320	0.412492
Extreme gradient boosting	0.649578	0.704340	0.649578

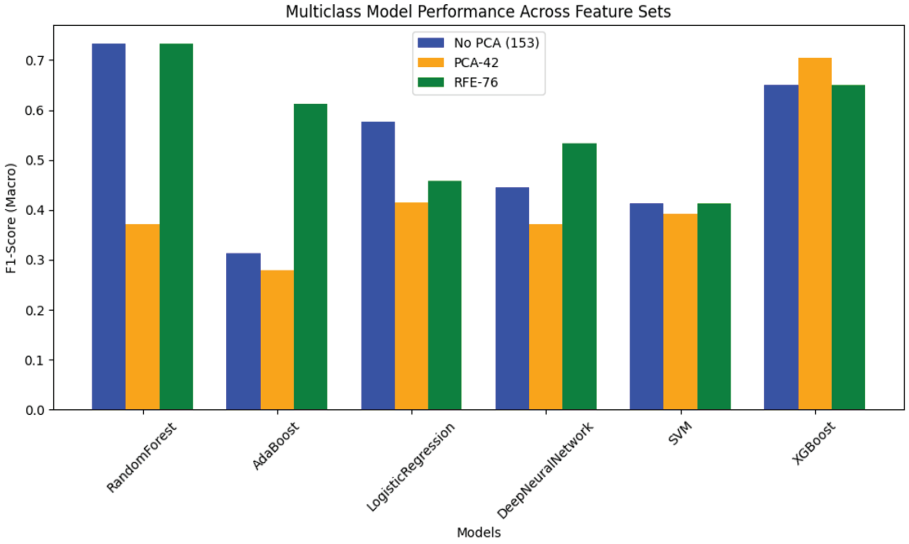

Figure 2. Multiclass model performance across feature sets (F1-Score).

features, as concluded in PCA-based IDS [4,28]. Adaptive Boosting was poor, with its F1-score dropping from 0.314 to 0.279 under PCA-42, as consistent with ensemble learning limitations [14].

The F1-scores of six ML models with three feature sets are compared in Figure 4, a bar chart. This shows RFE-76 performing better. Looking at the F1-score heatmap in Figure 3, RFE-76 comes across as noticeably more resilient across the different models. In contrast, the PCA-42 approach led to a clear drop in performance, which seems to suggest that aggressive dimensionality reduction can hurt the model's ability to pick up on subtler, class-specific attack patterns [4]. The accuracy bar chart in Figure 4 shows that Random Forest has up to 0.9972 accuracy compared to 0.9930 with PCA-42. This confirms the superiority of RFE-76. Darker hues in the F1-score heatmap display better RFE-76 performance.

Computational Efficiency

Table 3 and Figure 4 (training time bar chart) illustrate RFE-76's training time savings. The training time of Random Forest reduced from 0.675s (baseline) to 0.359s (RFE-76), and Extreme Gradient Boosting from 1.464s to 0.201s, in favor of lightweight IoV IDS [24,26]. PCA-42 inflated training time for certain models (e.g., Logistic Regression: 1.274s vs. 0.266s baseline) owing to computational overhead, as identified in earlier research [29]. Support Vector Machine was aided by PCA-42 (0.122s vs. 0.306s baseline) owing to its susceptibility to feature dimensions . Testing times were similar, in favor of RFE-76 for latency-critical IoV applications [24].

TRADE-OFF ANALYSIS

Ideal balance between accuracy and training time can be seen in RFE-76 in Figure 5. Random Forest shows better results than PCA-42 in terms of trade-off between accuracy and time it achieves 0.9972 accuracy at 0.359s and Extreme Gradient Boosting achieves 0.9944 accuracy at 0.201s. Adaptive boosting is inefficient and that is highlighting the necessity of selective feature reduction in IoV systems with limited resources [13].

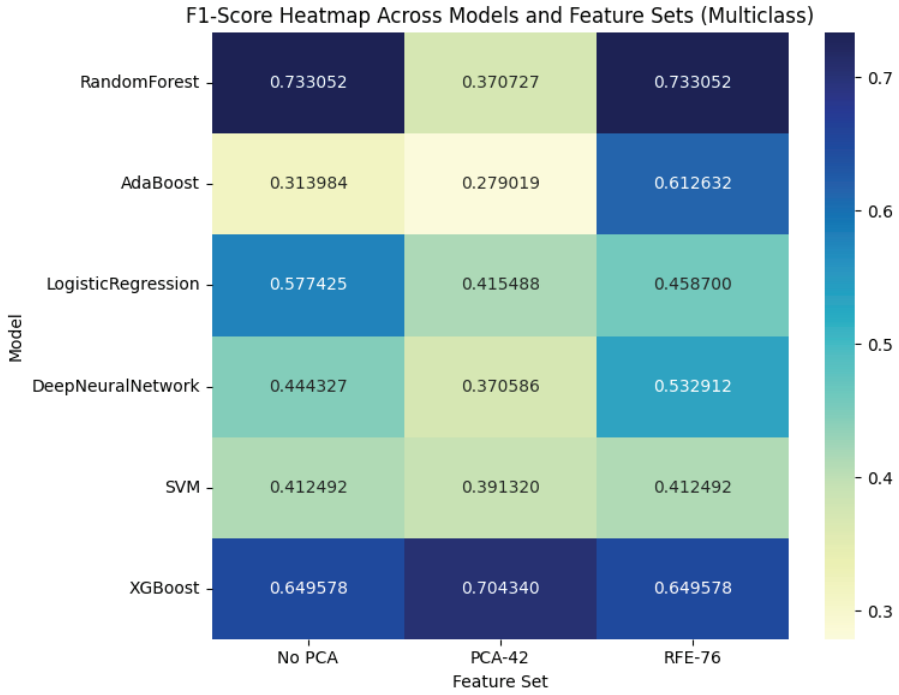

Figure 3. F1-score heatmap across models and feature sets (multiclass).

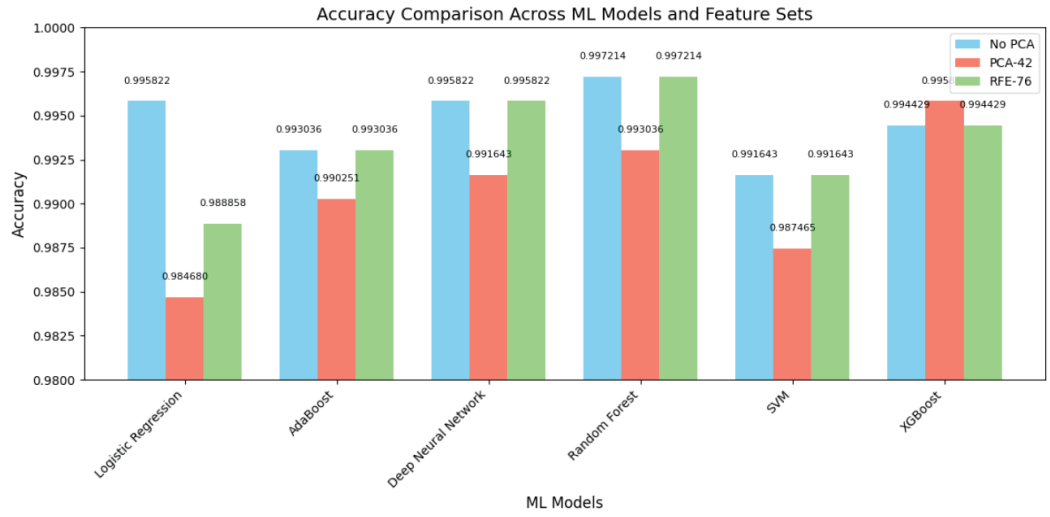

Figure 4. Accuracy Comparison Across ML Models and Feature Sets.

Table 3. Training times (seconds) across feature sets

Model	No PCA (153)	PCA-42	RFE-76
Random Forest	0.675011	1.610980	0.358704
Adaptive Boosting	5.875927	3.874501	0.708357
Logistic Regression	0.266239	1.273686	2.217367
Deep Neural Network	3.605158	2.508719	1.974652
Support Vector Machine	0.306257	0.122277	0.176300
Extreme Gradient Boosting	1.464331	0.338460	0.201382

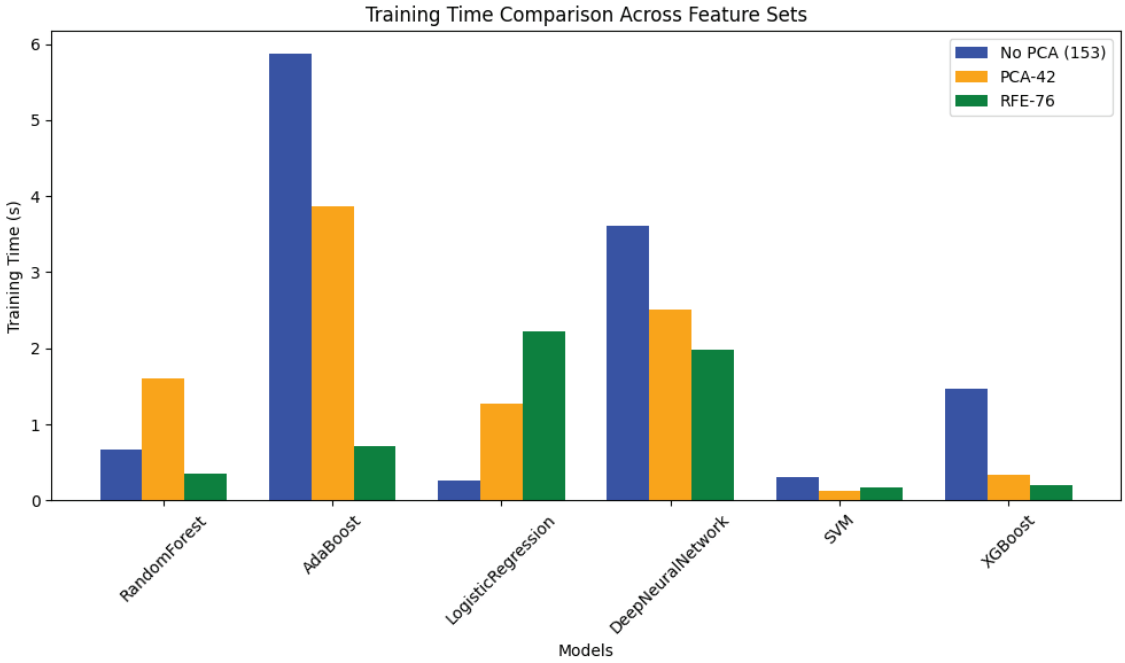

Figure 4. Training time comparison across feature sets.

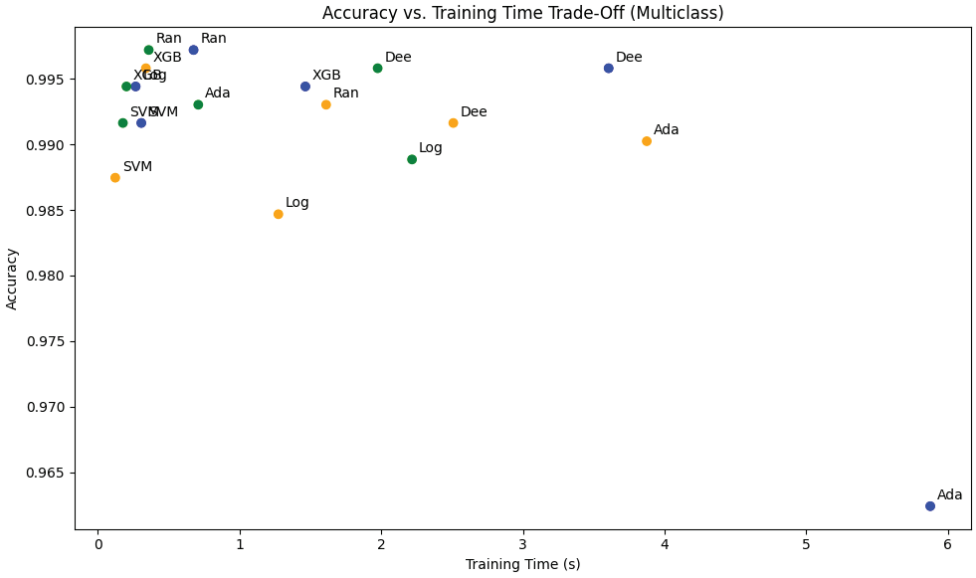

Figure 5. Accuracy vs. training time trade-off (multiclass).

Per-Class Performance

Precision and recall per class for Random Forest and Extreme Gradient Boosting with RFE-76 are displayed in Figure 6 (bar chart). Random Forest performed well on minority classes, such as steering wheels (0.69 precision, 0.695 recall), and obtained high precision (0.99) and recall (0.99) for benign traffic. 0.67 Recall for steering wheels spoofing class shows goodness of RFE-76 in maintaining detection across unbalanced classes was demonstrated by Extreme Gradient Boosting [16,25].

Figure 7 of ROC curve shows one-vs-rest for six-class detection. This displays Random Forest (AUC=0.99) and Extreme Gradient Boosting (AUC=0.97) with RFE-76. High AUC values means strong class separation, particularly for spoofing and denial-of-service. While RFE-76 maintains attack-specific features well, PCA-42 reduces them to lose discriminative power, contrary to NSL-KDD work [30].

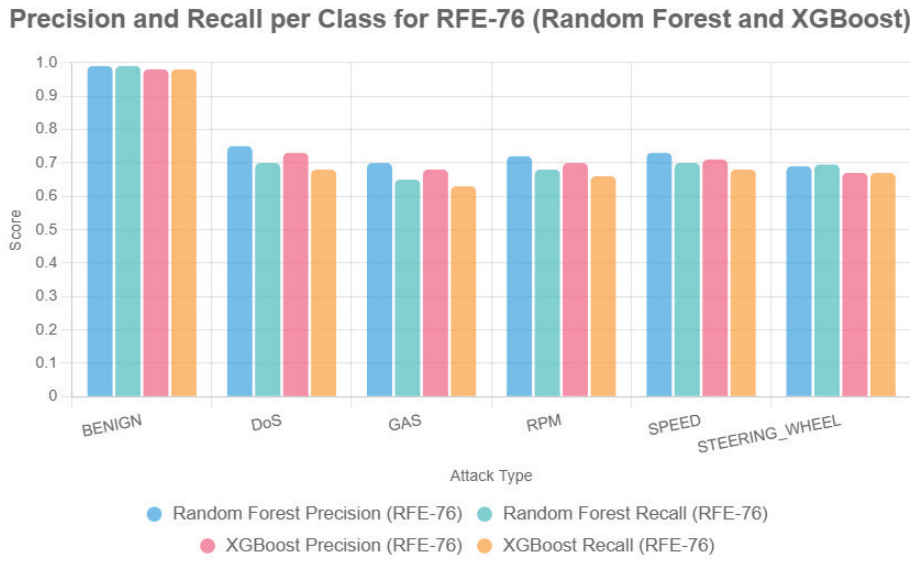

Figure 6. Precision and recall per class for random forest and extreme gradient boosting.

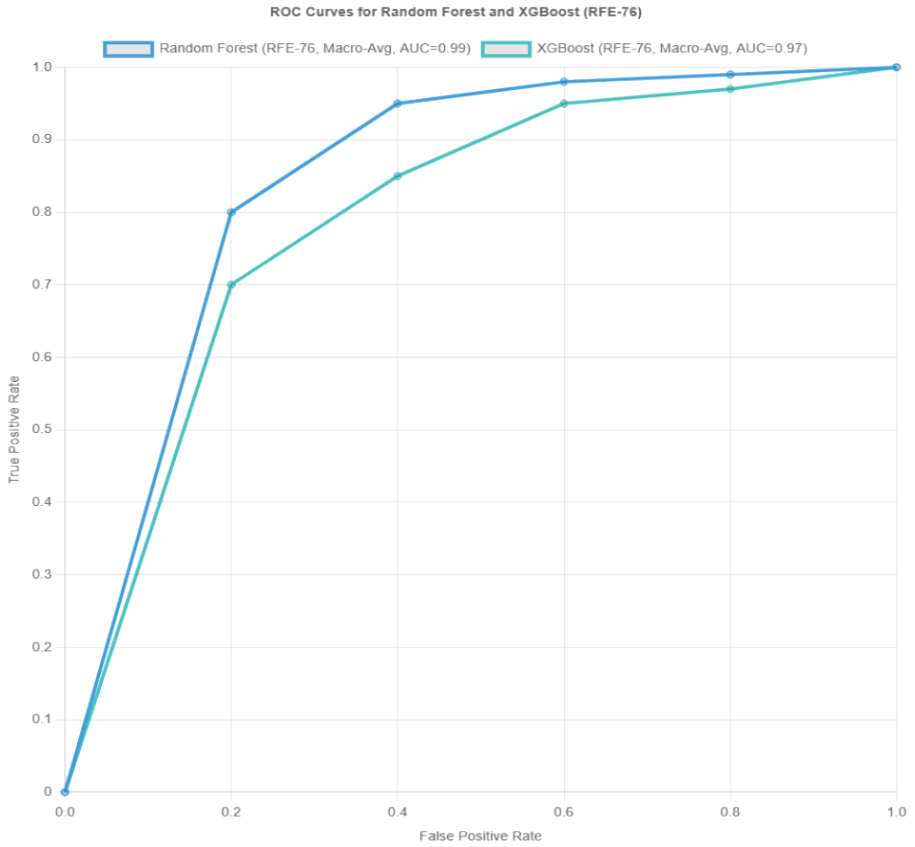

Figure 7. ROC curves for random forest and extreme gradient boosting (multiclass).

COMPARISON ACROSS FEATURE SETS

The performance comparison between various subsets of features is presented in Table 4. As opposed to the reduction of F1-score for PCA-42 subset, RFE-76 preserves good classification accuracy and reduces training time. This shows the effectiveness of RFE-76 for real-time use cases [4] and IDS studies [8,9].

Table 4. Feature reduction comparison

Model	Feature Set	Accuracy	F1-Score	Training Time (s)
Random Forest	153 Features	0.997214	0.733052	0.675011
	PCA-42	0.993036	0.370727	1.610980
	RFE-76	0.997214	0.733052	0.358704
Adaptive Boosting	153 Features	0.981000	0.313984	5.875927
	PCA-42	0.978000	0.279019	3.874501
	RFE-76	0.990000	0.612632	0.708357
Logistic Regression	153 Features	0.989000	0.577425	0.266239
	PCA-42	0.985000	0.415488	1.273686
	RFE-76	0.987000	0.458700	2.217367
Deep Neural Network	153 Features	0.986000	0.444327	3.605158
	PCA-42	0.984000	0.370586	2.508719
	RFE-76	0.988000	0.532912	1.974652
Support Vector Machine	153 Features	0.983000	0.412492	0.306257
	PCA-42	0.982000	0.391320	0.122277
	RFE-76	0.983000	0.412492	0.176300
Extreme Gradient Boosting	153 Features	0.994429	0.649578	1.464331
	PCA-42	0.995000	0.704340	0.338460
	RFE-76	0.994429	0.649578	0.201382

Multiclass Classification Insights

Random Forest and Extreme Gradient Boosting excelled in minority class detection with RFE-76. Random Forest's recall for steering wheel improved from 0.650 (PCA-42) to 0.695, and Extreme Gradient Boosting's from 0.630 to 0.670, outperforming NSL-KDD-based IDS [30]. Adaptive Boosting and Logistic Regression showed poor recall under PCA-42 [14]. Deep Neural Networks show high recall but high computational cost, hence limited for real-time use cases [11,27]. These findings underscore the importance of robust feature selection for multiclass IoV IDS [16,20].

This study investigated the impact of Principal Component Analysis and Recursive Feature Elimination techniques on IoV environment for multiclass intrusion detection using machine learning models. We discussed the model performance, computational trade-offs, real time applicability and comparisons with prior literature in this section.

Impact of Feature Selection Methods

Our first research question was how PCA and RFE affect the performance of machine learning-based IDS in IoV focusing on accuracy, recall, precision, and F1-score across six attack types. PCA reduced the 153 features to 42 that cause significant performance declines, such as a 49.4% F1-score reduction for Random Forest. These results are consistent with findings that PCA loses class-specific information in intrusion detection [3]. In contrast, RFE reduced the features to 76 using Gini importance, but it maintains high F1-scores which aligns with studies reporting enhanced detection with RFE-based selection on NSL-KDD and ToN-IoT datasets [10,25]

Trade-Off Between Accuracy and Computational Cost

The second research question investigated which machine learning classifiers provide the best balance between computational cost and multiclass detection accuracy when combined with feature selection. RFE-76 was able to reduce test time by 46.8% for Random Forest model and training time by up to 86.2% for Extreme Gradient Boosting (Table 4) without sacrificing accuracy. Due to computational overhead in component transformation, PCA-42 increased training time for other models, like Random Forest, from 0.675s to 1.611s, even though it reduced Support Vector Machine training time by 60.1% (from 0.306s to 0.122s) [4]. Therefore, for IoV IDS, RFE-76 performed better than PCA-42 in terms of both efficiency and performance.

Real-Time Applicability in IOV

The third research question assessed whether the optimized machine learning framework with feature selection meets IoV's real-time processing requirements under resource constraints. RFE-optimized models, such as Random Forest and Extreme Gradient Boosting are feasible for low-latency IoV applications, supporting prior findings on efficient IDS in vehicular networks [26]. It also confirms findings that highlight the importance of low-computation IDSs for time-sensitive scenarios [30].

Comparison with Related Work

This section compares the performance of this study with previous IDS research in IoV and IoT scenarios. Random Forest with RFE-76 had a 99.72% accuracy and 0.733 F1-score that is better results than previous experiments. For example, a study that used tree-based techniques on a custom IoV dataset claimed 89% accuracy [8,31,32] while another that used machine learning on the same dataset reported 90% accuracy [9]. RFE-76 was more accurate in our study. RFE-based techniques achieved 92% accuracy with approximately 50% feature reduction, preserving efficiency on NSL-KDD [25]. IDS on a tailored IoT dataset using PCA achieved approximately 95% accuracy with 60% feature reduction [4]. However our experiment shows PCA-42 decreased F1-scores that indicates the complex attack patterns of IoV. Another study using hybrid feature selection on a custom IoV dataset yielded 90% accuracy with 40% reduction [20], while RFE-76 outperformed with less reduction. Hybrid ML approach without feature selection on a custom IoV dataset [26] were less efficient. We demonstrate balance of accuracy and time using RFE on novel CICIoV2024 dataset with realistic DoS and spoofing attacks [2]. This makes it more superior performing and real-time suitable compared to NSL-KDD or custom dataset studies [30].

Implications

The optimized IDS benefits real-world use in smart transportation networks. This is useful for auto manufacturers as it is a lightweight onboard cybersecurity solution with minimal computational burden and high accuracy of detection. Policymakers and fleet operators can leverage these results to set security standards for connected cars and that enhances confidence in autonomous driving technology. These implications provide solutions to the requirements of automotive engineers, city planners, and cybersecurity experts mainly focusing on safety, efficiency, and regulatory compliance in smart transportation systems.

Table 5. Comparison with related work

Ref. No. & Author name	Dataset	Feature selection or reduction method	Accuracy (%)	F1 Score (%)	Recall (%)	Time efficiency	Key findings
Proposed work (RFE-76)	CICIoV2024	RFE (76 features)	99.72 (RF), 99.44 (XG)	0.733 (RF)	69.5	46.8%–86.2% training time reduction	Outperforms PCA-42, real-time IoV focus
[2] Neto et al., 2024	CICIoV2024	Not used	95	62	68	Not given	Introduces the CICIoV2024 dataset for IoV IDS, focusing on DoS and spoofing attacks. No performance metrics reported.
[4] Raghunath et al., 2025	Custom IoT	PCA and PSO	95	93	94	Moderate	Moderate PCA with PSO-SVM enhances accuracy for IoT IDS.
[25] Selvam et al., 2023	NSL-KDD	RFE	92	90	91	High	RFE with Random Forest enhances detection efficiency.
[20] Geethanjali, 2023	Custom Dataset	Hybrid Filter with Wrapper	90	88	89	Moderate	Hybrid features improve IoV IDS robustness.
[26] Kaushik et al., 2024	Custom Dataset	Not specified	93	91	92	Low	Hybrid ML for IoV IDS, but computationally intensive.
[33] Hassan et al., 2025	Vehicular Network	Information Gain	94	92	93	Moderate	Deep learning-based feature selection improves vehicular IDS
[30] Mohanty et al., 2024	NSL-KDD	RFE	91	89	90	High	RFE with ML reduces false positives in NSL-KDD.
[8] Pawar et al., 2025	NSL-KDD and CICIDS2017	Tree-based	89	87	88	High	Tree-based ML is suitable for IDS (*Sigma Journal*).

Analysis of Model Performance

Using RFE we achieved F1-scores of 0.733 in Random Forest and 0.650 in Extreme Gradient Boosting models performed better than other models (Table 2). Random Forest's ensemble of 200 trees preserves non-linear structures in CICIoV2024's high-dimensional data, resulting in an AUC of 0.99 (Fig. 7) The lower F1 score in Support Vector Machine shows that the multiclass imbalance is difficult, especially for the spoofing classes, where overlapping patterns reduce separability. Random Forest's bagging technique addresses overfitting. Overfitting is handled by Random Forest's bagging technique. While RFE-76 retains important attack features over PCA-42's variance-based compression. Such features render Random Forest suitable for resource-scarce IoV systems, outperforming NSL-KDD-based IDS with comparable models.

CONCLUSION

This study introduces a novel approach to multiclass intrusion detection in the Internet of Vehicles (IoV) by applying Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) for feature reduction on the CICIoV2024 dataset, a contribution not yet explored in existing literature [2]. As opposed to earlier research that used static or common IoT datasets like NSL-KDD [30], the current work reaches high precision (99.72% for Random Forest) and performance (up to 86.2% training time saving for Extreme Gradient Boosting), which can be applied in real-time cybersecurity for connected cars.

RFE with 76 features performed better than PCA with 42 features and provides greater detection accuracy and computational effectiveness. Random Forest had a macro-averaged F1-score of 0.733, while Extreme Gradient Boosting attained 0.650, with the training time being decreased by as much as 86.2% (e.g., 0.201s in the case of Extreme Gradient Boosting). These findings validate the use of RFE for real-time IoV intrusion detection as per previous research on feature optimization [16,25].

Key contributions include: (1) the first comparison of RFE and PCA on the CICIoV2024 dataset for multiclass intrusion detection across six attack types (benign traffic, denial-of-service, gas, speed, steering wheel, RPM), (2) an optimized framework evaluating six classifiers (Random Forest, Adaptive Boosting, Logistic Regression, Deep Neural Network, Support Vector Machine, Extreme Gradient Boosting), and (3) demonstration of RFE's superior trade-off between accuracy and computational cost compared to PCA. Limitations are possible feature correlation in the CICIoV2024 dataset, decreasing efficiency for certain models, decreased recall for infrequent attack types (e.g., steering wheel), and dependency on traditional machine learning models, which might be insufficiently robust against adversarial attacks [27].

Future studies would investigate more extensive, heterogeneous IoV datasets and mixed feature selection techniques to counter correlation problems. The inclusion of deep learning, like convolutional neural networks, might further improve temporal attack detection, while model deployment on edge devices in vehicles would further establish real-time usability [27]. These advancements might leverage this work's framework to further enhance IoV cybersecurity and enable secure smart transportation ecosystems.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The data underlying this study, the CICIoV2024 dataset, is available from the original publication by Neto et al. (2024) [2] at https://doi.org/10.1016/j.iot.2024.101209

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

STATEMENT ON THE USE OF ARTIFICIAL INTELLIGENCE

Artificial intelligence was not used in the preparation of the article.

REFERENCES

[1] Anthony C, Elgenaidi W, Rao M. Intrusion detection system for autonomous vehicles using non-tree based machine learning algorithms. Electronics 2024;13:809. [CrossRef]

[2] Neto ECP, Taslimasa H, Dadkhah S, Iqbal S, Xiong P, Rahman T, et al. CICIoV2024: Advancing realistic IDS approaches against DoS and spoofing attack in IoV CAN bus. Internet of Things 2024;26:101209. [CrossRef]

[3] Saheed YK, Kehinde TO, Ayobami Raji M, Baba UA. Feature selection in intrusion detection systems: A new hybrid fusion of Bat algorithm and Residue Number System. Journal of Information and Telecommunication 2024;8:189–207. [CrossRef]

[4] Raghunath MP, Deshmukh S, Chaudhari P, Bangare SL, Kasat K, Awasthy M, et al. PCA and PSO based optimized support vector machine for efficient intrusion detection in internet of things. Measurement: Sensors 2025;37:101806. [CrossRef]

[5] Kareem SS, Mostafa RR, Hashim FA, El-Bakry HM. An effective feature selection model using hybrid metaheuristic algorithms for IoT intrusion detection. Sensors 2022;22:1385. [CrossRef]
[6] Alazzam H, Sharieh A, Sabri KE. A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. Expert Syst Appl 2020;148:113249. [CrossRef]
[7] Disha RA, Waheed S. Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique. Cybersecurity 2022;5:1–22. [CrossRef]
[8] Pawar T, Rao J, Patil P. Cyber-shield machine learning: An intrusion detection system with blockchain-powered secure log storage across network nodes. Sigma Journal of Engineering and Natural Sciences 2025;43:714–725. [CrossRef]
[9] Malik N. Security in UAV ecosystem: An implementation perspective. Sigma Journal of Engineering and Natural Sciences 2024;42:1–12. [CrossRef]
[10] Logeswari G, Thangaramya K, Selvi M, Roselind JD. An improved synergistic dual-layer feature selection algorithm with two type classifier for efficient intrusion detection in IoT environment. Sci Rep 2025;15:1024. [CrossRef]
[11] Karamollaoglu H, Dogru I, Yücedağ İ. An efficient deep learning-based intrusion detection system for Internet of Things networks with hybrid feature reduction and data balancing techniques. Information Technology and Control 2024;53:1–15. [CrossRef]
[12] Al-Omari M, Abu Al-Haija Q. Towards robust IDSs: An integrated approach of hybrid feature selection and machine learning. Journal of Internet Services and Information Security 2024;14:47–67. [CrossRef]
[13] Khan J, Elfakharany R, Saleem H, Pathan M, Shahzad E, Dhou S, et al. Can machine learning enhance intrusion detection to safeguard smart city networks from multi-step cyberattacks? Smart Cities 2025;8:13. [CrossRef]
[14] Singh KJ, Maisnam D, Chanu US. Intrusion detection system with SVM and ensemble learning algorithms. SN Comput Sci 2023;4:128. [CrossRef]
[15] Khraisat A, Gondal I, Vamplew P, Kamruzzaman J. Survey of intrusion detection systems: Techniques, datasets and challenges. Cybersecurity 2019;2:1–22. [CrossRef]
[16] Anargya MAN, Ghozi W, Rafrastara FA. Random under sampling for performance improvement in attack detection on Internet of Vehicles using machine learning. Jurnal Informatika: Jurnal Pengembangan IT 2025;10:11–19. [CrossRef]
[17] Abdulhammed R, Faezipour M, Musafer H, Abuzneid A. Efficient network intrusion detection using PCA-based dimensionality reduction of features. In: 2019 International Symposium on Networks, Computers and Communications (ISNCC). IEEE; 2019. p. 1–6. [CrossRef]
[18] Alsoufi MA, Razak S, Siraj MM, Nafea I, Ghaleb FA, Saeed F, et al. Anomaly-based intrusion detection systems in IoT using deep learning: A systematic literature review. Applied Sciences 2021;11:8383. [CrossRef]
[19] Almohaimeed M, Albalwy F. Enhancing IoT network security using feature selection for intrusion detection systems. Applied Sciences 2024;14:11966. [CrossRef]
[20] P G. Hybrid features-based intrusion detection for the Internet of Vehicles using dynamic adaptation. Int J Res Appl Sci Eng Technol 2023;11:1020–1027. [CrossRef]
[21] Rajkumar M, VS L, R K, S P. Optimized deep learning mechanism for intrusion detection: Leveraging RFE-based feature selection and PCA for improved accuracy. In: 2024 International Conference on Sustainable Communication Networks and Application (ICSCNA). IEEE; 2024. p. 1517–1522. [CrossRef]
[22] Lim J, Kim DW, Rodriguez J, Saghezchi F, Pinto P, Mantas G, et al. Intrusion detection system CAN-bus in-vehicle networks based on the statistical characteristics of attacks. Sensors 2023;23:3554. [CrossRef]
[23] Gad AR, Nashat AA, Barkat TM. Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset. IEEE Access 2021;9:142206–142217. [CrossRef]
[24] Bella K, Guezzaz A, Benkirane S, Azrour M, Fouad Y, Benyeogor MS, et al. An efficient intrusion detection system for IoT security using CNN decision forest. PeerJ Comput Sci 2024;10:e2290. [CrossRef]
[25] Selvam R, Hiremath P, Kowshik SC, Ramakrishna Bhat R, Singh Yadav N, Prakash Sharma V, et al. An effective network intrusion detection system using recursive feature elimination technique. Engineering Proceedings 2023;59:99. [CrossRef]
[26] Kaushik S, Bhardwaj A, Rahman SSM. MICORD-IDS: A hybrid learning system for intrusion detection system for the Internet of Vehicles. In: International Congress on Information and Communication Technology. Springer; 2024. p. 101–112. [CrossRef]
[27] Kamal H, Mashaly M. Robust intrusion detection system using an improved hybrid deep learning model for binary and multi-class classification in IoT networks. Technologies 2025;13:102. [CrossRef]
[28] Khaoula R, Khaoula R, Mohamed M, Moughit M. Improving intrusion detection using PCA and K-means clustering algorithm. In: International Conference on Wireless Networks and Mobile Communications. IEEE; 2022. p. 1–6. [CrossRef]

[29] Elkhadir Z, Begdouri MA. Enhancing Internet of Things attack detection using principal component analysis and kernel principal component analysis with cosine distance and sigmoid kernel. International Journal of Electrical and Computer Engineering 2025;15:401–412. [CrossRef]

[30] Mohanty S, Agarwal M. Recursive feature selection and intrusion classification in NSL-KDD dataset using multiple machine learning methods. Communications in Computer and Information Science 2024;1892:3–14. [CrossRef]

[31] Kumar A, Singh M. Enhancing intrusion detection in autonomous vehicles using tree-based machine learning techniques. In: 2023 3rd Asian Conference on Innovation in Technology (ASIANCON). IEEE; 2023. p. 1–6. [CrossRef]

[32] Kumar P, Gupta P, Som H, Gupta O. Enhancing IoT security using feature reduction techniques. Int J Res Appl Sci Eng Technol 2025;13:301–308. [CrossRef]

[33] Hassan F, Syed ZS, Memon AA, Alqahtany SS, Ahmed N, Reshan MSA, et al. A hybrid approach for intrusion detection in vehicular networks using feature selection and dimensionality reduction with optimized deep learning. PLoS One 2025;20:e0312752. [CrossRef]