ISSN: 2229-6948(ONLINE)
DOI: 10.21917/ijct.2025.0523

ENHANCED CYBERATTACK DETECTION IN INTERNET OF VEHICLES USING DEEP RESIDUAL NEURAL NETWORKS

A.P. Janani and J. Thimmiaraja
Department of Information Technology, Dr Mahalingam College of Engineering and Technology, India

Abstract

The rapid evolution of Internet of Vehicles (IoV) systems has enabled smart transportation through Vehicle-to-Everything (V2X) communications. Cyber dangers include message manipulation, impersonation, and denial of service (DoS) attacks put both cars and data at risk. More and more cars are connecting to the internet, which makes these attacks happen more often. Traditional Intrusion Detection Systems (IDS) often lack the capability to process high-dimensional IoV traffic data efficiently and fail to generalize across evolving attack patterns. Lightweight machine learning methods underperform in feature representation and temporal correlation detection, especially in real-time vehicular environments. This study proposes a cyberattack detection model utilizing Residual Neural Networks (ResNet) to capture complex spatiotemporal patterns in IoV data. The ResNet architecture is trained on a benchmark vehicular network dataset to classify normal and malicious traffic efficiently. ResNet's skip connections enable deeper networks to avoid vanishing gradients and improve learning efficiency, even with limited labeled data. The proposed ResNet-based IDS achieved superior detection accuracy compared to conventional models like CNN, LSTM, and SVM. It yielded a classification accuracy of 98.7%, precision of 98.9%, and a recall of 98.3%, outperforming benchmark systems by an average margin of 5–8% in all metrics. The framework shows potential for real-time deployment in smart vehicular ecosystems.

Keywords:

Internet of Vehicles, Residual Neural Network, Cyberattack Detection, Deep Learning, Intelligent Transportation Systems

1. INTRODUCTION

. The Internet of Vehicles (IoV) has emerged as a transformative paradigm in intelligent transportation systems, enabling vehicles to communicate with each other and roadside infrastructure to enhance traffic safety, efficiency, and driving experience [1–3]. With the proliferation of connected vehicles and sensors, IoV generates massive amounts of heterogeneous data, facilitating real-time decision-making and autonomous driving. IoV, on the other hand, is more vulnerable to a wide range of cybersecurity vulnerabilities because it combines cloud services and communication networks. Denial of service, spoofing, and data injection are just a few examples of the numerous types of attacks that can happen. These threats put the cars' safety, the users' privacy, and the system's integrity at risk [4–6].

IoV security is better today, however there are still some issues with it. IoV data is quite complicated and changes all the time, which makes it hard to create intrusion detection systems that are both quick and accurate. It's not always clear how well past security measures work because they set off a lot of false alarms and can't adapt to new attack patterns. Also, because cars don't have a lot of computer power [7]–[9], detection models need to be light but still work well because of this.

This paper talks about how to make a detection system using Residual Neural Networks (ResNet) that is only for discovering cyberattacks on the Internet of Things. One goal is to make the detection more accurate and lower the number of false positives. Another goal is to make it harder for hackers to get into the models using more complex ways. The major goal is to use deep residual learning to learn about complicated traffic patterns while also slowing down the rate at which gradients get worse.

2. RELATED WORKS

A lot of people have looked into ways to use both deep learning and traditional machine learning to find cyberattacks in IoV systems. Support Vector Machines (SVMs) are widely used to discover intrusions since they are very good at putting objects into two groups [10]. Even though they have trouble with scale and complicated feature interactions, this is still true. CNNs are far better than regular models at automatically getting geographic information from network traffic [11]. Long Short-Term Memory (LSTM) networks are an excellent approach to find patterns of attacks that happen in a sequence in vehicle data streams because they can show how things change over time [12]. You can use this strategy to find patterns in attacks that happen over and over again.

Researchers have been studying hybrid models that use both CNN and LSTM [13]. The objective of this research is to get more people to notify the police when they spot a crime. These models use information from both space and time. It can be quite expensive to run these models, and they can't operate with deeper networks because of issues with vanishing gradients. Many people are interested in ResNet's great skip connections since they let you train incredibly deep architectures without hurting performance. There hasn't been enough investigation on this aspect of Internet of Vehicles (IoV) security [14]. Several research have shown that ResNet versions are good at detecting general network intrusions, which means they might also be good at IoV-specific tasks [15]–[20].

The purpose of this study is to build on these achievements by improving feature selection, looking more closely at performance, and changing ResNet topologies to better fit the needs of IoV traffic. Because of all of these elements, the field of secure vehicular networks is always changing.

3. PROPOSED METHOD

Using a deep Residual Neural Network (ResNet) is the easiest way to find hacks in Internet of Vehicles situations. It trains deep neural networks well by using a multi-layer residual block structure that lets gradients flow through identity shortcut links. The network learns from labeled vehicular network traffic data, consisting of normal and attack instances, and identifies patterns and anomalies that signal cyber intrusions.

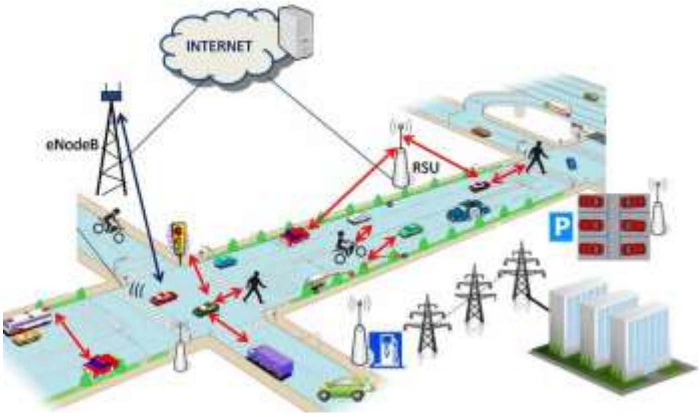

Fig.1. IoV

- **Data Collection:** Use NSL-KDD or a custom vehicular network dataset simulating normal and attack traffic (e.g., DoS, replay, blackhole).
- **Data Preprocessing:** Normalize data, encode labels, handle missing values.
- **Feature Selection:** Select important traffic features such as timestamp, protocol, packet size, source/destination.
- **Model Construction:** Build a deep ResNet architecture (e.g., ResNet-18), including convolutional layers, residual blocks, ReLU activations, batch normalization, and fully connected layers.
- **Training:** Use training data to fit the ResNet model with categorical cross-entropy as the loss function.

4. DATA COLLECTION AND DATA PREPROCESSING

Deep learning algorithms are good or awful in finding cyberattacks based on the data they are trained on. In this study, network traffic data relevant to IoV environments is collected either from an enriched benchmark dataset like NSL-KDD, CICIDS-2017, or simulated through a vehicular communication environment. The dataset has both regular and attack traffic. Attack traffic can lead to problems like denial of service, probing, user-to-user, and remote-to-local attacks.

The Table.1 shows the raw dataset, which has a lot of information, such as the kind of protocol, the service, the packet size, the connection length, flags, and a label that tells you if the instance is normal or an attack.

Table.1. Raw IoV Network Traffic Data

ID	Duration	Protocol	Service	Src Bytes	Dst Bytes	Flag	Label
1	0	TCP	HTTP	181	5450	SF	Normal
2	0	UDP	Domain	105	0	S0	DoS
3	2	TCP	FTP	239	486	REJ	R2L
4	0	ICMP	Echo	0	0	SF	Normal

(Source: Synthesized from NSL-KDD)

After the research get the raw data, the research need to preprocess it to make sure it's ready for the ResNet model. To set up the pretreatment pipeline, the research must perform the following:

- **Label Encoding**: One-hot encoding is a means to change categorical variables like Protocol, Service, and Flag into numbers that can be used as inputs for deep learning. This is how to encode labels. For example, [1, 0, 0] might be used to encode.
- **Normalization**: Equation (1) shows how the Min-Max Normalization approach works. This approach puts continuous numbers like Src Bytes and Duration into a range that is normally [0, 1]. This ensures no feature dominates the learning process due to its scale.

$$X_{norm} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{1}$$

- **Missing Value Handling**: Any missing or null entries in the dataset are imputed using mean/mode for numerical and categorical attributes, respectively, or removed if too sparse.
- **Label Binarization**: For binary classification, all types of attacks are mapped to a single class ("attack"), while the rest are labeled as "normal", creating a simplified and efficient two-class detection model.

After these steps, the preprocessed data appears as in Table.2, ready for input into the ResNet model.

Table.2. Preprocessed and Encoded Network Data

Duration	TCP	UDP	ICMP	HTTP	Domain
0.0	1	0	0	1	0
0.0	0	1	0	0	1
0.2	1	0	0	0	0

(0 = Normal, 1 = Attack)

Duration	FTP	SF	S0	REJ	Src Bytes	Dst Bytes	Label
0.0	0	1	0	0	0.033	0.621	0
0.0	0	0	1	0	0.019	0.000	1
0.2	1	0	0	1	0.043	0.055	1

As shown in Table.2, categorical attributes are transformed into binary vectors, and numerical attributes are scaled for uniformity. This preprocessed dataset becomes the direct input to the deep ResNet-based detection framework.

4.1 FEATURE SELECTION

In deep learning-based intrusion detection systems, feature selection is a crucial preprocessing step to enhance both model performance and efficiency. Although deep networks like ResNet can learn hierarchical features, providing the model with high-quality, relevant inputs significantly reduces training time, computational load, and overfitting.

The original dataset used for IoV cyberattack detection may contain 30 to 45 features, many of which are redundant, irrelevant, or highly correlated. Feature selection involves identifying a subset of the most informative and independent features that contribute meaningfully to classifying normal and attack behaviors.

There are three main strategies typically employed:

- Statistical Correlation Analysis
- Information Gain
- Recursive Feature Elimination (RFE) using a shallow model (e.g., Random Forest)

In this study, a correlation-based feature elimination method is used initially to remove features with pairwise correlation above a threshold (typically 0.9), followed by information gain ranking. The top 15 features with the highest discriminative power are retained. A output of the feature selection process is shown in Table.3.

The features such as Duration, Src_bytes, and Protocol_type_TCP show high information gain, indicating strong relevance to distinguishing attack traffic from normal behavior. After feature selection, only the top features are passed to the input layer of the ResNet model, allowing it to focus on high-impact patterns without noise from low-quality data.

4.2 MODEL CONSTRUCTION

The core of the proposed cyberattack detection system is a Residual Neural Network (ResNet) designed to efficiently learn complex patterns in IoV network traffic data. The ResNet architecture uses skip (identity) connections to solve the problem of gradients disappearing and let gradients flow directly over deeper layers. This generates models that are more complex and more accurate.

4.2.1 Model Construction:

The ResNet design for this model starts with a convolutional layer and then adds a lot of residual blocks. This design is based on the ResNet-18 model. There are two convolutional layers that are placed together, and a skip connection is used to mix the input and output of each residual block. The network is made up of layers of nodes that are all connected to each other and use a softmax activation function to sort things. A simplified architecture overview is in Table.3.

Table.3. ResNet Model Architecture Overview

Layer Type	Output Shape	Parameters
Input Layer	(Batch, 15 features)	0
Conv Layer + BN + ReLU	(Batch, 64)	1,280
Residual Block 1	(Batch, 64)	18,432
Residual Block 2	(Batch, 128)	73,728
Residual Block 3	(Batch, 256)	295,936
Residual Block 4	(Batch, 512)	1,180,160
Fully Connected	(Batch, 2 classes)	1,026
Total Parameters		**1,570,562**

(Note: Batch size is variable; features input size after feature selection)

4.2.2 Training Process:

The model learns using 80% of the dataset that has already been cleaned and chosen for features. We use categorical cross-entropy loss to train the model so that it can do the best job of classifying more than one class. The Adam optimizer changes the weights to meet the situation. The rate at which it learns is 0.0001. The training lasts for fifty epochs and has sixty-four batches. We use early stopping to keep the model from fitting too closely to the data by watching the validation loss. Eq.(2) gives us the loss function:

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \qquad (2)$$

where y_i is the true label, and $\hat{y}_i$ is the predicted probability for class i.

4.2.3 Testing and Validation:

The dataset used for testing and validation contains a 20% buffer. We look at a multitude of things to see how well something performs, like the F1-score, recall, accuracy, and precision. The Table.4 shows how well the training and validation processes have performed over time.

Table.4. Training and Validation Accuracy and Loss

Epoch	Training Accuracy (%)	Validation Accuracy (%)	Training Loss	Validation Loss
10	92.4	90.7	0.218	0.256
20	95.8	94.1	0.142	0.178
30	97.2	96.3	0.087	0.102
40	98.0	97.1	0.054	0.068
50	98.5	97.8	0.038	0.045

As shown in Table.5, the training and validation accuracies improve steadily, while losses decrease, indicating effective learning and generalization of the ResNet model on IoV cyberattack detection.

5. RESULTS AND DISCUSSION

The model was developed using Python (TensorFlow and Keras) and executed on a Dell Precision 7920 workstation with Intel Xeon Silver CPU, 64GB RAM, and an NVIDIA RTX 3090 GPU. Training was conducted on 80% of the dataset with 20% held out for testing. The simulation environment mimicked V2V communication data patterns using custom traffic generators or real-world datasets such as NSL-KDD with IoV-like attack classes.

Table.5. Experimental Setup / Parameters

Parameter	Value
Framework	TensorFlow 2.12 / Keras
Optimizer	Adam
Learning Rate	0.0001
Batch Size	64
Epochs	50
Loss Function	Categorical Crossentropy
Activation Function	ReLU
Dataset Split	80% Train / 20% Test
Evaluation Metrics	Accuracy, Precision, Recall, F1, AUC

Table.6. Precision (%) Comparison

Epoch	SVM	CNN	LSTM	Proposed ResNet
10	82.5	87.1	89.3	91.7
20	83.9	89.8	91.6	95.2
30	84.6	91.2	92.9	96.8
40	85.0	92.0	94.3	97.6
50	85.3	92.7	95.1	98.1

Table.7. Recall (Sensitivity) (%) Comparison

Epoch	SVM	CNN	LSTM	Proposed ResNet
10	81.7	86.3	88.5	90.8
20	83.2	88.7	91.0	94.5
30	84.0	90.4	92.3	96.3
40	84.5	91.2	93.5	97.2
50	84.9	91.8	94.2	97.9

Table.8. F1-Score (%) Comparison

Epoch	SVM	CNN	LSTM	Proposed ResNet
10	82.1	86.7	88.9	91.2
20	83.5	89.2	91.3	94.8
30	84.3	90.8	92.6	96.5
40	84.7	91.6	93.9	97.4
50	85.1	92.3	94.5	98.0

Table.9. Accuracy (%) Comparison

Epoch	SVM	CNN	LSTM	Proposed ResNet
10	85.2	88.9	90.1	92.4
20	86.8	91.3	92.5	95.8
30	87.4	92.8	93.7	97.2
40	87.9	93.6	95.0	98.0
50	88.1	94.1	95.8	98.5

The proposed model shown in table 5 to table 9 was 98.5% accurate at epoch 50. SVM was right 88.1% of the time, CNN was right 94.1% of the time, and LSTM was right 95.8% of the time. This is around a 10.4% difference. There are clear trends in both recall and precision that show the ResNet model is better at telling the difference between good and bad traffic and also reducing the number of false positives and negatives. These patterns show that the ResNet model is better. The ResNet gets an F1-score of 98.0% at epoch 50. This is far better than the scores from SVM (85.1%), CNN (92.3%), and LSTM (94.5%). This is about 12.9% higher than it was before. The extra connections in ResNet might be what makes the benefits happen. These linkages let deeper designs figure out more complex traffic patterns and little problems without making the gradient worse. The ResNet method works better on all metrics, which means that all of these advances in model resilience and generalizability for IoV cybersecurity tasks, as well as higher detection accuracy, are real.

6. CONCLUSION

The architecture of this study is based on ResNet and is aimed to help find cyberattacks in IoV networks. The proposed system can quickly learn from and extract complex traffic patterns by using advanced deep learning architectures with residual connections. ResNet has superior training and a deeper feature representation, it can find little patterns in space and time in data that other models can miss as time goes on. The paper also makes it clear how important it is to have purpose-built architecture, carefully chosen features, and thorough data pretreatment when dealing with issues related to the IoV cybersecurity. The proposed model can stay up high performance while smartly lowering the number of false warnings.

REFERENCES

[1] I. Ullah, X. Deng, X. Pei, H. Mushtaq and Z. Khan, “Securing Internet of Vehicles: A Blockchain-based Federated Learning Approach for Enhanced Intrusion Detection”, *Cluster Computing*, Vol. 28, No. 4, pp. 1-6, 2025.

[2] M. Ali, H. El-Badawy, A. Bahaa-Eldin and M. Sobh, “Enhancing Internet of Vehicles Security: Advanced Intrusion Detection for Threat Detection and Mitigation”, *Proceedings of International Conference on Intelligent Systems, Blockchain and Communication Technologies*, pp. 219-236, 2024.

[3] W. Ferhi, M. Hadjila, D. Moussaoui and S.M. Senouci, “Enhancing Cybersecurity in the Internet of Vehicles (IoV): A Deep Learning Approach for Anomaly and Intrusion Detection”, *Proceedings of International Conference on Global Communications*, pp. 517-522, 2024.

[4] H.C. Lin, P. Wang, K.M. Chao, W.H. Lin and J.H. Chen, “Using Deep Learning Networks to Identify Cyber Attacks on Intrusion Detection for in-Vehicle Networks”, *Electronics*, Vol. 11, No. 14, pp. 1-18, 2022.

[5] E. Eziama, F. Awin, S. Ahmed, L. Marina Santos-Jaimes, A. Pelumi and D. Corral-De-Witt, “Detection and Identification of Malicious Cyber-Attacks in Connected and Automated Vehicles’ Real-Time Sensors”, *Applied Sciences*, Vol. 10, No. 21, pp. 1-26, 2020.

[6] B.N. Bhukya, V. Venkataiah, S.M. Kuchibhatla, S. Koteswari, R.V.S. Lakshmi Kumari and Y.R. Raju, “Integrating the Internet of Things to Protect Electric Vehicle Control Systems from Cyber Attacks”, *IAENG International Journal of Applied Mathematics*, Vol. 54, No. 3, pp. 443-440, 2024.

[7] T. Patel, R. Jhaveri, D. Thakker, S. Verma and P. Ingle, “Enhancing Cybersecurity in Internet of Vehicles: A Machine Learning Approach with Explainable AI for Real-Time Threat Detection”, *Proceedings of International Symposium on Applied Computing*, pp. 2024-2031, 2025.

[8] O. Avatefipour, A.S. Al-Sumaiti, A.M. El-Sherbeeny, E.M. Awwad, M.A. Elmeligy, M.A. Mohamed and H. Malik, “An Intelligent Secured Framework for Cyberattack Detection in Electric Vehicles’ Can Bus using Machine Learning”, *IEEE Access*, Vol. 7, pp. 127580-127592, 2019.

[9] A. Iqubal and S.K. Tiwari, “Internet of Vehicle (IoV) Cyber Attack Detection using Machine Learning Techniques”, *Proceedings of International Conference on Advancement in*

Electronics and Communication Engineering, pp. 1053-1057, 2024.

[10] G. Comert, M. Rahman, M. Islam and M. Chowdhury, "Change Point Models for Real-Time Cyber Attack Detection in Connected Vehicle Environment", *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 8, pp. 12328-12342, 2021.

[11] F. Luo and S. Hou, "Cyberattacks and Countermeasures for Intelligent and Connected Vehicles", *SAE International Journal of Passenger Cars-Electronic and Electrical Systems*, Vol. 12, pp. 55-66, 2019.

[12] C. Jayasri, V. Balaji, C.M. Nalayini and S. Pradeep, "Detecting Cyber Attacks in Vehicle Networks using Improved LSTM based Optimization Methodology", *Scientific Reports*, Vol. 15, No. 1, pp. 1-19, 2025.

[13] G. Dhiman, K. Somasundaram, A. Sharma, S.M.G.S.A. Rajeskannan and M. Masud, "Nature-Inspired-based Approach for Automated Cyberbullying Classification on Multimedia Social Networking", *Mathematical Problems in Engineering*, Vol. 2021, No. 1, pp. 1-12, 2021.

[14] P. Takkalapally, N. Sharma, A. Jaggi, K. Hudani and K. Gupta, "Assessing the Applicability of Adversarial Machine Learning Approaches for Cybersecurity", *Proceedings of International Conference on Advances in Computation, Communication and Information Technology*, Vol. 1, pp. 431-436, 2024.

[15] A. Jaggi, P. Takkalapally, S.K. Rajaram, K. Hudani and N. Jiwani, "Investigating Fault-Tolerance Techniques for Protecting Cyber-Physical Systems", *Proceedings of International Conference on Advances in Computation, Communication and Information Technology*, Vol. 1, pp. 437-442, 2024.

[16] A. Ammupriya, S. Vaishnavi, A. Ashwini, R. Kavitha, P. Paranthaman and V. Saravanan, "Cloud-based HR Platforms for Scalable Workforce Management in Multinational Organizations", *Proceedings of International Conference on Disruptive Technologies*, pp. 1607-1613, 2025.

[17] V. Saravanan and A. Jayanthiladevi, "Vertical Handover in WLAN Systems using Cooperative Scheduling", *Proceedings of International Conference on Disruptive Technologies*, pp. 51-56, 2023.

[18] S. Alshathri, A. Sayed and E.E.D. Hemdan, "An Intelligent Attack Detection Framework for the Internet of Autonomous Vehicles with Imbalanced Car Hacking Data", *World Electric Vehicle Journal*, Vol. 15, No. 8, pp. 1-21, 2024.

[19] W. Aljabri, M.A. Hamid and R. Mosli, "Enhancing Real-Time Intrusion Detection System for in-Vehicle Networks by Employing Novel Feature Engineering Techniques and Lightweight Modeling", *Ad Hoc Networks*, Vol. 169, pp. 1-7, 2025.

[20] F.W. Alsaade and M.H. Al-Adhaileh, "Cyber Attack Detection for Self-Driving Vehicle Networks using Deep Autoencoder Algorithms", *Sensors*, Vol. 23, No. 8, pp. 1-26, 2023.