



International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991



Vol. 21 No. 3 (1) 2025

ijerst.editor@gmail.com
editor@ijerst.com

*Research Paper***DEEP CNN FOR SMART MOBILITY: AN AI-BASED TRAFFIC FLOW ANALYZER FOR ADAPTIVE URBAN INFRASTRUCTURE**

G. Divya, Pulgam Manjunath Reddy, Bhukya Muralikrishn A, Pakala Parsharam
Department of Computer Science and Engineering (AI&ML), Kommuri Pratap Reddy
Institute of Technology, Ghatkesar, Medchal, 500088.

Received: 05-6-2025

Accepted: 03-7-2025

Published: 10-7-2025

ABSTRACT

This research introduces a novel hybrid deep learning-based system for intelligent road traffic monitoring, aimed at improving transportation safety through accurate identification and classification of road incidents. The system integrates multiple machine learning models—including a Random Forest Classifier, a Deep Neural Network (DNN), and a hybrid model that combines a Convolutional Neural Network (CNN) with an Extra Trees Classifier (ETC)—to detect and classify incidents such as accidents, dense traffic, fire, obstacles, smoke, and sparse traffic. A comprehensive dataset of road scene images is preprocessed, divided into training and testing sets, and used to train these models. Among them, the hybrid CNN+ETC model achieved the highest accuracy of 95.14%, significantly outperforming the others. The system is equipped with an intuitive graphical user interface (GUI) that allows users to upload datasets, preprocess images, train models, and perform incident predictions on test images. It also offers visualization tools such as accuracy and loss graphs, along with confusion matrices for performance evaluation. The model has shown precise prediction capabilities, correctly identifying scenarios like smoke, accidents, and heavy traffic. This application not only demonstrates strong practical relevance for real-world use but also represents a promising step forward in advancing real-time traffic incident detection and smart transportation management systems.

Keywords: Intelligent Traffic Monitoring, Image-Based Incident Classification, Smoke and Fire Recognition, Accident Detection.

1. INTRODUCTION

One of the most significant and difficult jobs for the traveler and transport management industries is the real-time identification and analysis of traffic incidents. In addition to seriously injuring or even killing people, traffic accidents have an impact on traffic operations and flow. 1.35 million individuals worldwide lose their lives in road accidents each year, according to the World Health Organization (WHO). Furthermore, traffic-related incidents such as accidents and traffic congestion cost the United

States of America an average of US\$160 billion annually; by the end of 2020, that amount might rise to US\$192 billion. Scholars categorize traffic congestion into occurrences that are predictable and unpredictable, or recurrent and non-recurring. One of the primary one-time occurrences that disrupt traffic flow is a traffic accident. Therefore, it is crucial to provide drivers with condition information and precise and timely detection of traffic incidents in real time so they may avoid risk areas and select the quickest and safest routes. It also assists state police and

InceptionResNetV2 outperformed ResNet in terms of performance. The suggested method was verified on the toy automobile in order to gauge its effectiveness in the actual world.

Khan SW et.al [3] analysed deep learning techniques for Anomaly detection in Traffic surveillance videos. They specifically focused on accidents in traffic videos, utilizing Video Traffic Surveillance Cameras (VTSS) for monitoring traffic. They proposed a methodology for automatically detecting accidents in surveillance videos. Convolutional Neural Networks (CNNs), a specialized deep learning approach for grid-like data, were employed for image and video analysis. They utilized CNNs to identify anomalies (accidents) in videos captured by VTSS and implemented a rolling prediction algorithm for high accuracy. The CNN model was trained on a dataset (VAID) comprising images with anomalies. Testing the methodology on multiple videos yielded successful results, with an accuracy of 82% in detecting traffic accident events in the traffic surveillance system videos.

Razi et.al [4] proposed on the deep learning techniques for traffic safety . They investigated deep learning techniques that may be used to: traffic: video: analysis with an emphasis on human and autonomous vehicle driving safety. A typical processing pipeline that includes steps like video enhancement, stabilization, object detection, classification, trajectory extraction, speed estimation, event analysis, modeling, and anomaly detection was presented by them in order to comprehend and interpret traffic videos. The intention was to provide comparative studies of effective traditional and DL-based algorithms for each stage, guiding traffic analyzers in the development of bespoke processing frameworks. Along with reviewing current open-source tools and public datasets for

training DL models, they also addressed sample traffic challenges and provided detailed solutions for each. Investigations were conducted into links to adjacent study fields, including the assessment of drivers' cognitive abilities, crowdsourcing-based monitoring systems, edge computing in roadside infrastructures, and automated driving systems installed in automobiles. They also discussed the future prospects, unresolved issues, and obstacles to the broad adoption of traffic monitoring systems, as well as commercial implementations of these systems.

Prajapati et.al [5] considered the machine learning techniques for road accident prediction. Despite the greatest efforts of the car industry's engineers and researchers, traffic accidents continued to occur. In an effort to better understand the causes of risky traffic occurrences, it was deemed helpful to design a prediction system capable of automatically categorizing the severity of injuries sustained in various traffic accidents. For policies to be effective, they needed to be based on rigorous scientific research into the root causes of accidents and the extent of injuries. The system incorporated multiple machine learning-based injury severity prediction algorithms.

ElSahly et.al [6] evaluated using a thorough study of algorithms for detecting traffic incidents. To create Automatic Incident Detection (AID) systems with reasonable performance standards, they have carried out a number of research. The two components that make up an AID system are usually the data processing module and the data gathering module. Mean Time to Detect (MTTD), False Alarm Rate (FAR), and Detection Rate (DR) are the three main metrics used to evaluate the effectiveness of AID systems. Four groups are revealed by classifying AID according to data processing and event detection algorithms: statistical, artificial intelligence-based, video-image

processing, and comparison. They evaluated the effectiveness, drawbacks, and associated data gathering and processing methods of the current AID systems in order to study and describe them. The study is a useful resource for researchers interested in creating new AID systems since it was conducted using a systematic literature review that adheres to the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) criteria.

Bakheet et.al [7] constructed deep neural network for detecting car accidents. Lately, machine learning applications related to the development of Intelligent Transportation Systems (ITS), which are essential for developing smart cities, have focused on the prediction and detection of vehicular accidents. Using motion temporal templates and fuzzy time-slicing, they introduced a new vision-based framework for real-time vehicle accident prediction and detection. The framework works in stages, starting with the automated identification of moving objects (pedestrians crossing the road or automobiles). Using supervised learning, clustering, and temporal templates, it tracks the identified moving objects in real time. The temporal templates of moving objects are then used to extract a large number of local characteristics. At last, the collected characteristics are used to train an efficient deep neural network (DNN) model, which can identify anomalous vehicle behavioral patterns and anticipate accidents before they happen. Trials carried out on actual car accident footage show that the framework produces generally encouraging outcomes, with a 98.5% hit rate and a 4.2% false alarm rate. Zhou, et.al [8] created spatiotemporal feature encoding in a VANET context to identify traffic accidents. They suggested using a multilayer neural network to encode spatiotemporal features as the basis for an accident detection method. In

particular, the temporal characteristics of the video were encoded using a multilayer neural network in order to cluster the video frames. They identified the border frames as maybe accident frames based on the acquired frame clusters. To determine if these frames were indeed accident frames, they then recorded and encoded the spatial connections of the items found from these perhaps accident frames. The comprehensive tests proved that the suggested method satisfied the real-time detection requirement in the VANET environment and obtained promising detection efficiency and accuracy for traffic accident detection.

3. PROPOSED SYSTEM

The main objective of this research is to create a machine learning model that can correctly identify scenarios from images as "High traffic" , "Low traffic", "Accidents" or "Fire accident" based on attributes that have been collected from the images. The proposed model presents a novel hybrid deep learning framework for intelligent road traffic monitoring, integrating conventional Random Forest Classifier (RFC), Deep Neural Network (DNN), and a Convolutional Neural Network (CNN) coupled with an Extra Trees Classifier (ETC). The method begins with image preprocessing and dataset splitting, where traffic surveillance images are prepared and labeled data is partitioned.. The innovative core lies in fusing CNN-extracted image features with ETC to boost prediction robustness and capture high-dimensional spatial dependencies. This layered hybridization has not been comprehensively addressed in prior traffic safety studies, making it an effective and original approach for urban traffic monitoring and accident risk prediction.

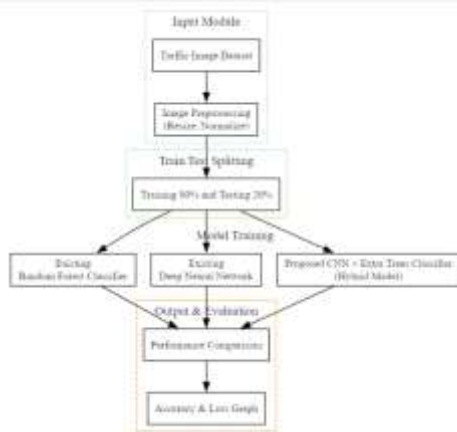


Fig. 2: Proposed System Architecture

Proposed CNN Feature Extraction

The Convolutional Neural Network (CNN) method is highly effective for tasks involving spatially structured data, such as images from traffic surveillance systems. In this application, CNN is beneficial due to its ability to automatically detect features like vehicle shapes, road patterns, and congestion levels from raw image inputs. The architecture is designed to handle application-specific image inputs of fixed size (64x64x3), ensuring compatibility with traffic scene datasets. Its hierarchical structure enables learning of low-level edges to high-level semantic features, making it ideal for automated traffic condition recognition and classification. The model is efficient, scalable, and requires minimal manual feature engineering.

Step 1: Initializing the CNN Model The process begins by initializing a sequential model, which allows layers to be added one after the other in a linear fashion. This structure is suitable for simple yet deep networks where the data flows from input to output through each layer without branching.

Step 2: First Convolution and Pooling Layer The first convolution layer applies 32 filters of size 3x3 to the input image of shape 64x64x3. This layer detects basic patterns such as lines and corners. It uses the ReLU activation function to introduce non-linearity and improve learning

capability. A max pooling layer follows, which downsamples the feature maps by taking the maximum value in each 2x2 region. This reduces the spatial dimensions and helps control overfitting.

Step 3: Repeating Convolution and Pooling This process of convolution followed by max pooling is repeated three more times. Each repetition allows the network to learn increasingly complex and abstract features. The multiple layers ensure that detailed and hierarchical information is captured from the input images, enhancing the model’s ability to differentiate between traffic classes.

Step 4: Flattening the Feature Maps Once the convolution and pooling layers have extracted features, the multi-dimensional output is flattened into a one-dimensional vector. This conversion prepares the data for the fully connected layers by turning spatial features into a format suitable for classification.

Step 5: Fully Connected Layers and Output The flattened vector is passed through a dense layer with 256 neurons and ReLU activation, which functions as a high-level feature combiner. This layer integrates the extracted features to form a comprehensive representation. Finally, a dense output layer with softmax activation produces a probability distribution over the target classes, allowing the model to make multi-class predictions based on the most probable traffic category.

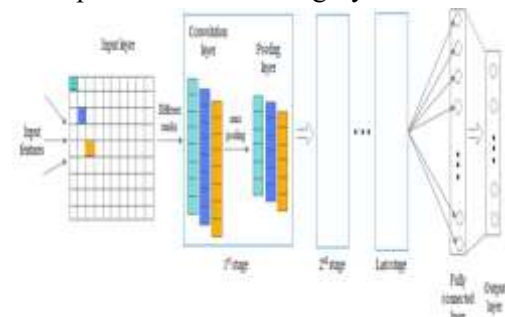


Fig. 3: CNN Block Diagram.

Advantages

Convolutional Neural Networks (CNNs) excel at automatically learning

hierarchical representations of data. In the early layers, they extract low-level features such as edges and textures, while deeper layers capture more abstract and complex patterns. This hierarchical feature learning allows CNNs to effectively understand the intricate structures present in images and other data types. Additionally, CNNs leverage convolutional and pooling operations to achieve translation invariance, enabling them to recognize patterns regardless of their position in the input space. This makes them robust to spatial transformations and enhances their generalization capabilities.

Extra Trees Classifier

The Extra Trees Classifier (ETC) is an ensemble learning method well-suited for application-specific problems involving high-dimensional, structured datasets like traffic monitoring metadata or extracted image features. It offers key advantages such as reduced overfitting, faster training compared to other ensemble methods like Random Forests, and enhanced generalization through its randomized node splitting mechanism. For traffic classification, ETC is particularly effective when combined with deep learning-based feature extraction (e.g., CNN), as it can efficiently handle the rich, high-level features generated by deep networks. This makes ETC an ideal choice for hybrid models where both speed and accuracy are critical for real-time decision-making in smart city applications.

Step 1: Initializing the Ensemble Model

The Extra Trees Classifier begins with the initialization of multiple decision trees. Unlike traditional decision tree models, ETC constructs a large number of trees in parallel without bootstrapping the dataset. This use of the entire training dataset for each tree improves stability and diversity.

Step 2: Randomized Feature and Split Selection During tree construction, ETC introduces additional randomness by

selecting a random subset of features and random split thresholds for each node. This contrasts with standard decision trees, which choose the best split based on criteria like Gini impurity or entropy. The increased randomness leads to more diversified trees and reduces the chance of overfitting.

Step 3: Growing Decision Trees Each decision tree in the ensemble is grown fully without pruning. The trees are trained independently on the full dataset, but due to the random feature and threshold selection, they produce different splits, resulting in varied structures and predictions. This diversity contributes to the ensemble's robustness.

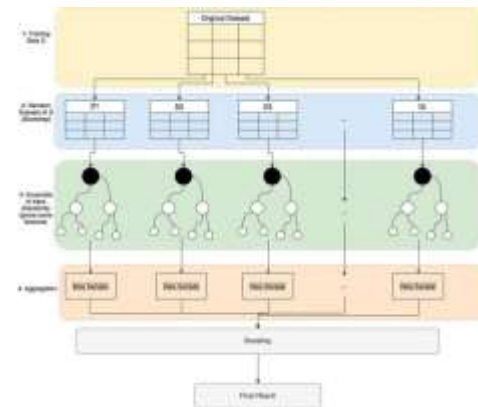


Fig. 4: Proposed ETC.

Step 4: Aggregating Predictions Once all trees are constructed, the classifier aggregates their outputs. For classification tasks, the final output is determined by majority voting, where each tree casts a vote for a class label, and the class with the most votes is selected. This collective decision-making process enhances predictive accuracy and reduces variance.

Step 5: Application to Feature Sets In practical implementation, ETC is applied to features extracted from either structured metadata (like vehicle counts, time, and location) or image embeddings derived from deep CNN layers. The classifier learns from these complex feature spaces to accurately categorize traffic conditions or incident types, providing a powerful

backend model for intelligent transportation systems.

4. RESULTS AND DISCUSSION

Figure 5 displays the confusion matrix for the proposed CNN with Extra Trees Classifier (ETC) model. The matrix follows the same structure with true classes on the y-axis and predicted classes on the x-axis. The diagonal values, indicating correct predictions, are: 154 accident occurred, 188 dense traffic, 177 fire, 97 obstacle detected, 177 smoke detected, and 166 sparse traffic. Misclassifications are minimal, with notable errors including 16 dense traffic instances predicted as accident occurred and 14 sparse traffic instances predicted as accident occurred. The matrix shows high accuracy across all classes, with smoke detected (177) and dense traffic (188) having the highest correct predictions, and the color intensity reflects the improved performance with brighter diagonal values.

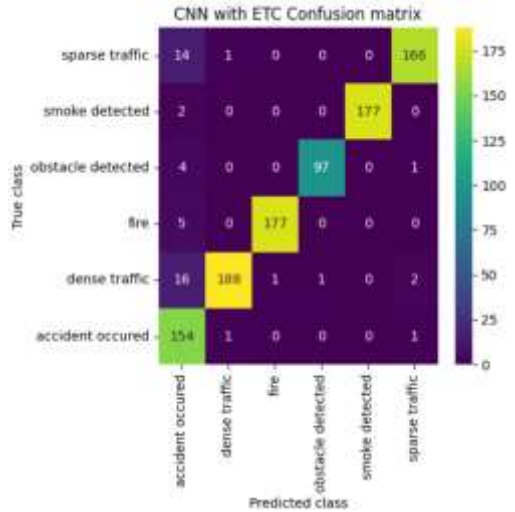


Fig. 5: Proposed CNN with ETC Confusion Matrix.

Figure 6 shows a test image classified by the CNN with ETC model as "smoke detected". The image depicts a road scene with a bus emitting visible smoke from its exhaust, alongside other vehicles in an urban setting with billboards and buildings in the background. The classification label "Classified as: smoke detected" is overlaid

on the image in yellow text with a black outline, positioned at the top. The image dimensions are 500x500 pixels, as specified in the code, and the prediction aligns with the visible smoke, indicating accurate detection by the model.



Fig. 6: Predicted From Test Image as Smoke Detected.



Fig. 7: Predicted From Test Image as Dense Traffic.

Figure 7 shows a test image classified by the CNN with ETC model as "dense traffic". The image captures a heavily congested road with numerous vehicles, including cars, vans, and buses, closely packed and moving slowly. The scene suggests a traffic jam, likely in an urban area. The label "Classified as: dense traffic" is overlaid in yellow text with a black outline at the top of the image. The image is resized to 500x500 pixels, and the prediction correctly identifies the high

vehicle density, reflecting the model’s capability to recognize traffic conditions.



Fig. 8: Predicted From Test Image as Accident Occurred.

Figure 8 shows a test image classified by the CNN with ETC model as "accident occurred". The image depicts a collision between a green car and a blue truck, with the car’s front end significantly damaged and the truck’s rear impacted. The car has reflective tape and appears to be a service vehicle, possibly involved in road maintenance. The label "Classified as: accident occurred" is overlaid in yellow text with a black outline at the top. The image is resized to 500x500 pixels, and the prediction accurately identifies the accident scenario, demonstrating the model’s effectiveness in detecting road incidents.

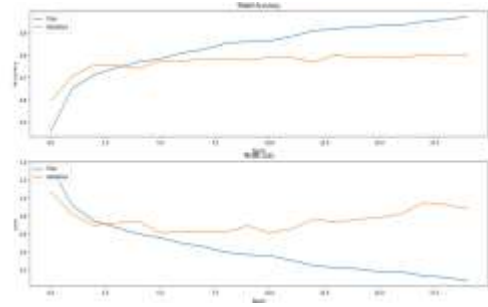


Fig. 9: Accuracy and Loss of Graph of CNN.

Figure 9 shows the accuracy and loss graphs of the CNN with ETC model over 20 epochs, plotted in two subplots. The top subplot, titled "Model Accuracy", displays training accuracy (blue line) and validation accuracy (orange line). Training accuracy starts at around 0.5 and rises steadily to approximately 0.95 by epoch 20, while validation accuracy starts at 0.6, fluctuates slightly, and stabilizes around 0.85. The bottom subplot, titled "Model Loss", shows training loss (blue line) and validation loss (orange line). Training loss begins at 1.2, drops sharply to 0.4 by epoch 5, and gradually decreases to around 0.1 by epoch 20. Validation loss starts at 1.0, decreases to 0.6, fluctuates, and ends around 0.5. The graphs indicate effective learning with training accuracy and loss improving consistently, though validation performance shows some variability, suggesting potential overfitting.

Table 1. Comparison of Model Performance

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Sensitivity (%)	Specificity (%)
Existing RFC	67.36	68.43	66.22	66.76	80.15	82.17
Existing DNN	59.62	60.92	58.31	57.87	69.31	77.01
Existing CNN with ETC	95.14	95.66	95.34	95.26	99.35	92.16

The comparison in table 1 summarizes the performance metrics of the three models: Existing RFC, Existing DNN, and CNN

with ETC, across accuracy, precision, recall Existing RFC achieved an accuracy of 67.36%, with a precision of 68.43%,

recall of 66.22%, and F1-score of 66.76%. Its sensitivity is 80.15%, and specificity is 82.17%, indicating decent performance but with room for improvement, especially in correctly identifying positive cases. Existing DNN performs worse, with an accuracy of 59.62%, precision of 60.92%, recall of 58.31%, and F1-score of 57.87%. Its sensitivity (69.31%) and specificity (77.01%) are lower than RFC, reflecting more misclassifications. CNN with ETC outperforms both, achieving a high accuracy of 95.14%, precision of 95.66%, recall of 95.34%, and F1-score of 95.26%. Its sensitivity (99.35%) and specificity (92.16%) are significantly better, demonstrating superior performance in both identifying positive cases and minimizing false positives. The comparison highlights CNN with ETC as the best model for road incident identification.

5. CONCLUSION

The application for "Efficient Road Incident Identification" successfully demonstrates the effectiveness of hybrid deep learning techniques in enhancing transportation safety through intelligent road monitoring. By integrating a Random Forest Classifier (RFC), a Deep Neural Network (DNN), and a hybrid Convolutional Neural Network with Extra Trees Classifier (CNN+ETC), the system achieves robust classification of road incidents, such as accidents, dense traffic, fire, obstacles, smoke, and sparse traffic, with the CNN+ETC model outperforming others at an accuracy of 95.14%, precision of 95.66%, and F1-score of 95.26%. The user-friendly Tkinter-based GUI facilitates seamless interaction, allowing users to upload datasets, preprocess images, train models, predict incidents, and visualize performance metrics, as evidenced by the confusion matrices and accuracy/loss graphs. The system's ability to accurately classify test images, such as identifying smoke, dense traffic, and accidents,

underscores its practical utility in real-time road monitoring, contributing significantly to improved safety measures.

REFERENCES

- [1] Li, Linchao, Yi Lin, Bowen Du, Fan Yang, and Bin Ran. "Real-time traffic incident detection based on a hybrid deep learning model". *Transportmetrica A transport science* 18, no. 1 (2022) 78-98.
- [2] Pathik, N., Gupta, R. K., Sahu, Y., Sharma, A., Masud, M., & Baz, M. (2022). "AI enabled accident detection and alert system using IoT and deep learning for smart cities". *Sustainability*, 14(13), 7701.
- [3] Khan SW, Hafeez Q, Khalid MI, Alroobaea R, Hussain S, Iqbal J, Almotiri J, Ullah SS. "Anomaly Detection in Traffic Surveillance Videos Using Deep Learning". *Sensors*. 2022; 22(17) 6563.
- [4] Razi, Abolfazl, Xiwen Chen, Huayu Li, Hao Wang, Brendan Russo, Yan Chen, and Hongbin Yu. "Deep learning serves traffic safety analysis A forward looking review." *ET Intelligent Transport Systems* 17, no. 1 (2023) 22-71.
- [5] Prajapati, Gaurav, Lav Kumar, and Smt Rekha S. Patil. "Road Accident Prediction Using Machine Learning." *Journal of Scientific Research and Technology*(2023) 48-59.
- [6] ElSahly, Osama, and Akmal Abdelfatah. 2022. "A Systematic Review of Traffic Incident Detection Algorithms" *Sustainability* 14, no. 22 14859.
- [7] Bakheet, Samy, and Ayoub Al-Hamadi. "A deep neural framework for real-time vehicular accident detection based on motion temporal templates." *Heliyon* 8, no. 11 (2022).
- [8] Zhou, Zhili, Xiaohua Dong, Zhetao Li, Keping Yu, Chun Ding, and Yimin Yang. "Spatio-temporal feature encoding for traffic accident detection in VANET environment." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 10 (2022) 19772-19781.