



RESEARCH ARTICLE

OPEN ACCESS

A DEEP SPATIO-TEMPORAL FRAMEWORK FOR MULTI-CLASS TRAFFIC PREDICTION AND ACCIDENT DETECTION IN SURVEILLANCE VIDEO

Dhartee Patel¹, Jinal Ahir², Namrata Shroff³, Shankar Parmar^{*4}

^{1, 2, 3}Computer Engineering Department, Government Engineering College, Bharuch, India.

⁴Electronics and Communication Engineering Department, Government Engineering College, Godhra, Gujarat, India.

¹<http://orcid.org/0009-0005-4299-8643>, ²<http://orcid.org/0009-0009-1622-5638>,

³<http://orcid.org/0000-0001-6343-6205>, ⁴<http://orcid.org/0000-0002-3034-5144>

Email: dhartee1611@gmail.com, jinalahir2805@gmail.com, namrata.gecb@gmail.com, *shankar.parmar@gtu.edu.in

ARTICLE INFO

Article History

Received: April 8, 2026

Reviewed: May 8, 2026

Accepted: July 12, 2026

Published: August 31, 2026

Keywords:

Traffic Surveillance,
Accident Detection,
Yolov8,
Spatio-Temporal Modeling,
CNNLSTM
Intelligent Transportation
System.

ABSTRACT

Traffic surveillance systems play a crucial role in intelligent transportation by enabling automated monitoring, traffic prediction, and accident detection. However, recognizing complex traffic scenarios from real-world videos remains challenging due to dynamic environments and temporal dependencies. This paper proposes a unified spatio-temporal framework that integrates YOLOv8 based object detection, convolutional neural networks for spatial feature extraction, and long short-term memory networks for temporal modeling. Traffic videos are preprocessed to enhance visual consistency, and detected objects are transformed into structured spatial representations. Temporal dependencies across video sequences are learned using LSTM networks, and the extracted features are evaluated using multiple machine learning classifiers under different preprocessing strategies. Experimental results demonstrate that Z-score standardization improves classification performance, with Support Vector Machine achieving 63.27% accuracy and an F1-score of 55.66% in an eight-class traffic scenario classification task, indicating the feasibility and robustness of the proposed framework in real-world traffic environments.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

Traffic congestion and road accidents have become critical challenges in rapidly urbanizing regions, where increasing vehicle density and limited infrastructure place substantial pressure on existing transportation systems [1]. As cities continue to expand and mobility demands intensify, traditional manual surveillance and rule-based monitoring approaches are no longer sufficient to manage real-time traffic dynamics effectively [2]. Such methods often struggle to detect sudden anomalies including accidents, lane violations, and abrupt fluctuations in traffic flow resulting in delayed response times and compromised road safety [3]. In response to these limitations, intelligent video-based traffic analysis has emerged as a promising solution, leveraging advancements in computer vision and deep learning to enable automated, scalable, and real-time monitoring [4]. Over the past decade, significant research efforts have focused on developing robust algorithms for vehicle detection, traffic flow estimation, and anomaly recognition [5]. Convolutional Neural Networks (CNNs) have demonstrated strong capability in extracting meaningful spatial features from complex road scenes, thereby improving vehicle localization and classification accuracy [6]. More recently, real-time object detection frameworks such as You Only Look Once (YOLO) and transformer-based detectors have further enhanced detection speed and precision, facilitating deployment in densely populated urban environments [7]. Complementary research has explored temporal modeling techniques including Long Short-Term Memory (LSTM) networks, 3D CNNs, and sequence-based architectures to capture motion patterns and identify traffic incidents across video sequences [8]. Although these approaches have contributed substantially to intelligent transportation research, many rely on computationally intensive architectures, controlled datasets, or address isolated components of traffic analysis rather than offering an integrated end-to-end solution [9]. Despite notable progress, several challenges remain unresolved. Numerous studies depend on narrow field-of-view or synthetic datasets, which limit model generalizability in heterogeneous real-world scenarios [10].

Furthermore, existing works frequently treat vehicle detection, spatial feature extraction, temporal analysis, and accident identification as independent modules, thereby reducing overall system efficiency and scalability [11]. Achieving real-time performance under resource-constrained deployment conditions such as roadside surveillance infrastructure continues to be a practical concern [12]. These limitations underscore the necessity for a unified, computationally efficient framework that integrates spatial temporal modeling and anomaly detection using real-world traffic video data [13]. To address these challenges, this study proposes a comprehensive deep learning based framework for real-time traffic flow analysis and accident detection. The proposed approach integrates object detection, spatial feature extraction, and temporal modeling into a cohesive pipeline [14]. An advanced detection model is employed to accurately identify vehicles within surveillance footage, after which temporal sequence modeling captures abrupt variations in motion and density patterns that may indicate congestion or accidents [15]. By directly leveraging video data and minimizing reliance on handcrafted features, the framework enhances scalability and adaptability across diverse traffic conditions [16]. The system is designed to support practical deployment scenarios and aims to improve early identification of abnormal events, thereby contributing to safer and more efficient intelligent transportation systems [17]. The primary objective of this research is to develop a robust and computationally efficient methodology for vehicle detection, traffic flow analysis, and accident identification in real time [18]. Specifically, this study evaluates the effectiveness of combining spatial detection with temporal sequence modeling, investigates the impact of different preprocessing strategies, and assesses classification performance across multiple machine learning algorithms on realistic traffic datasets [19].

II. THEORETICAL REFERENCE

II.1 RELATED WORK

To understand current trends and challenges in traffic accident detection, it is essential to review recent methods and models [20]. Numerous approaches have been proposed with the goals of enhancing detection accuracy, improving tracking reliability, and enabling real-time response in diverse urban environments [21]. The following sections highlight the most significant contributions, their thematic focus, and the specific challenges addressed by the model [22].

II.1.1 Trajectory Conflict and Motion Pattern Analysis

Trajectory-based conflict reasoning has proven effective for distinguishing near-miss events from actual collisions by explicitly modeling the motion interactions among vehicles [23]. Ghahremannezhad et al. (2022) combined YOLOv4-based object detection with trajectory divergence and proximity metrics to differentiate collision-prone scenarios from non-severe interactions. This approach highlighted the importance of motion continuity rather than isolated frame-level cues [24]. Huang et al. (2019) extended motion-centric analysis by proposing a two-stream Convolutional Neural Network (CNN) that jointly learns spatial appearance features and temporal motion patterns from aerial traffic videos [25]. Their architecture enabled real-time near-accident detection by capturing vehicle interactions over time. Such trajectory-driven approaches strengthen the conceptual modeling of inter-vehicle dynamics and closely align with the trajectory-based anomaly detection module employed in our proposed system [26].

II.1.2 Spatio-Temporal and Graph-Based Traffic Modeling

Beyond individual object detection, several studies have explored traffic behavior at a broader spatio-temporal scale [27]. Cui et al. (2018) introduced the Temporal Graph Convolutional Long Short-Term Memory (TGC-LSTM) model, which integrates Graph Convolutional Networks (GCNs) with LSTM units to represent spatial correlations among interconnected road segments while learning temporal traffic evolution [28]. This framework demonstrated strong performance in network-level traffic state prediction. Similarly, Ismaeel et al. (2023) proposed a deep CNNRNN architecture for smart city traffic pattern classification, achieving an accuracy of 95% [29]. While these methods are effective for global traffic analysis and congestion prediction, they are less suitable for real-time, localized accident or near-miss detection, which remains the primary focus of the proposed system [30].

II.1.3 Critical Review of Related Work and Research Gaps

Despite notable progress in traffic accident detection and spatio-temporal traffic analysis, several persistent challenges remain in the existing literature.

Dataset Limitations: Many studies rely on synthetic data or static-camera datasets with limited environmental diversity, which restricts model generalizability. To address this issue, the proposed work utilizes a diverse and annotated real-world dataset sourced from IEEE DataPort.

Computational Overhead: High-capacity models such as 3D CNNs and graph-based architectures incur substantial computational costs, limiting their suitability for real-time deployment. Hybrid CNN–LSTM-based approaches offer a more favorable balance between detection accuracy and computational efficiency, motivating their adoption in the proposed framework.

Occlusion and Dense Traffic Conditions: Vehicle occlusions, frequent interactions, and complex motion patterns in dense traffic environments remain challenging. Prior studies (Nagy & Simon, 2021) indicate that temporal modeling and trajectory-aware inference can alleviate these issues. The proposed approach incorporates temporal consistency to improve robustness under such conditions. Table 1 presents a comparative summary of key studies related to traffic accident and near-miss detection, highlighting the methodologies employed, dataset characteristics, major contributions, and identified limitations.

Identified Research Gaps: Although existing methods demonstrate promising performance, several gaps remain unaddressed. Most approaches treat object detection, temporal feature extraction, and trajectory reasoning as independent components, limiting their ability to model the complete evolution of accident scenarios. Furthermore, computationally intensive architectures hinder real-time deployment on edge and resource-constrained platforms.

Dense, heterogeneous, and occlusion-prone traffic environments are still insufficiently explored, resulting in degraded performance under realistic urban conditions. Additionally, only a limited number of systems provide an end-to-end framework integrating detection, risk assessment, and real-time decision support. These gaps motivate the development of a unified, lightweight, and real-time traffic accident detection framework capable of robust operation across diverse traffic environments.

Table 1: Summary of Related Work in Traffic Accident and Near-Miss Detection.

Author	Method	Data	Contribution	Limitation
Yadava et al. (2022)	YOLOv4 + Kalman Hungarian	Intersection videos	Vehicle detection and conflict analysis	Poor performance under dense occlusion
Li et al. (2020)	CNN-LSTM SMOTE	Crash records	Crash risk forecasting	High computation cost
Roy et al. (2019)	Siamese Interaction LSTM	Aerial videos	Temporal attention for collision risk	Sensitive to noisy trajectories
Ghahremanzadeh et al. (2022)	YOLOv4 + Trajectory analysis	UAV footage	Near-miss vs collision differentiation	Degrades with rapid motion
Huang et al. (2019)	Two-stream CNN	Aerial videos	Real-time near-accident detection	Limited scalability
Cui et al. (2018)	TGC-LSTM (GCN+LSTM)	Network traffic data	Spatio-temporal traffic prediction	Not video-level accident detection
Ismaeel et al. (2023)	Deep CNN-RNN	Smart city data	Traffic state classification	No explicit collision modeling

Source: Authors, (2026).

III. MATERIALS AND METHODS

This research proposes a hybrid deep-learning pipeline that integrates Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks for real time traffic flow analysis and accident detection. The complete workflow includes video preprocessing, feature extraction, temporal modeling, and classification of traffic events. The system is designed to handle unconstrained road environments with varying traffic densities, lighting conditions, and weather scenarios.

III.1 DATASET

TheTrafficAccident Detection Video Dataset for AI-Driven Computer Vision Systems in Smart City Transportation was sourced from the IEEE DataPort repository, where it was contributed by Adewopo et al. (2023) to support research on intelligent traffic surveillance. The dataset compiles videos from multiple real-world sources, including urban traffic surveillance cameras, in-vehicle dashcams, and additional publicly available crash footage. This ensures natural variation in traffic flow, accident types, and environmental conditions. The dataset is systematically organized into three pre defined experimental splitsTraining, Validation, and Testingto maintain consistency in benchmarking and reproducible experimentation. In preparing the dataset, all raw videos were segmented into non-overlapping five-second clips, resulting in ap proximately 5,690 video samples.This segmentation strategy ensures uniformity in clip duration and highlights the dynamic temporal behavior around accident events. Each clip is categorized into one of eight scenario classesBackend, Backend Rollover, Frontend, Frontend Rollover, Sidehit, Sidehit Rollover, General Augmented Crash, and No Accident Normal Trafficthereby supporting both multi-class accident classifi cation and accident-versus-non-accident detection, as shown in **Table 2**. The dataset provides video-level annotations only, without frame-level bounding boxes or object level segmentation, making it suitable for end-to-end video classification and temporal feature learning. The clips represent diverse scenarios involving variations in light ing, weather, camera angle, road structure, traffic density, and sensor quality, adding significant heterogeneity and complexity to the dataset.

Table 2: Dataset Distribution by Scenario.

Scenario	Total Videos
Backend	362
Backend Rollover	28
Frontend	369
Frontend Rollover	51
General Augmented Crash	1459
No Accident Normal Traffic	2283
Sidehit	1027
Sidehit Rollover	111
Total	5690

Source: Authors, (2026).

III.2 PREPROCESSING

To ensure uniformity and reliable model training, all videos in the dataset underwent a comprehensive preprocessing pipeline. Each five-second video clip was decomposed into individual frames, resulting in an average of 2530 frames per video, depending on the recording devices frame rate.

The original videos exhibited heterogeneous frame resolutions due to their diverse sources, including traffic surveillance systems and dashcams. To standardize the input representation, all extracted frames were re sized to a fixed resolution of 224×224 pixels, a commonly used dimension compatible with state-of-the-art deep learning architectures such as CNNs and 3D-CNN-based spatiotemporal models. This resizing step ensures consistent input size across the dataset, reduces computational requirements, and enables efficient batch processing. Following resizing, all pixel intensities were normalized to stabilize the training process. Pixel values originally in the 0255 range were scaled to 01, and a channel-wise mean standard deviation normalization was applied to bring the data distribution closer to that of pretrained image models. Normalization improves gradient flow, accelerates convergence, and minimizes the impact of illumination variations commonly observed in traffic surveillance footage.

Normalization:

Normalization scales input feature values to a fixed range, typically [0,1], to ensure numerical stability and faster convergence during model training.

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

For scaling the data to a general range [a,b]:

$$X_{norm} = a + \frac{(X - X_{min})(b - a)}{X_{max} - X_{min}}$$

To enhance frame clarity, noise-removal techniques such as Gaussian smoothing and median filtering are selectively applied to reduce sensor noise, motion-induced blur, and compression artifacts. These denoising operations preserve important structural information, thereby improving feature extraction and increasing the accuracy of accident detection models.

III.3 YOLOv8

YOLOv8 is the latest generation of the “You Only Look Once” family of real-time computer vision models, designed to deliver high-performance object detection with improved accuracy and reduced latency. It plays a critical role in modern visual perception systems by providing a unified framework capable of object detection, instance segmentation, and image classification, making it suitable for real-time and resource constrained environments. YOLOv8 introduces several architectural enhancements, including an improved CSP-based backbone for deeper feature extraction, a refined PAN/FPN neck for stronger multi-scale representation, and a fully anchor-free detection head that directly predicts bounding boxes at each spatial location. This anchor-free design eliminates manual anchor tuning, reduces computational complexity, and improves generalization across different object scales. YOLOv8 achieves faster inference, higher FPS, and improved accuracy through its decoupled classification regression head, advanced IoU-based loss functions, and optimized training strategies such as dynamic label assignment and enhanced augmentation pipelines. Additionally, it supports multiple vision tasks — detection, segmentation, and classification — within a single architecture. Compared to earlier versions such as YOLOv5 and YOLOv7, YOLOv8 replaces traditional anchor-based prediction, which often struggled with scale variation and required extensive hyperparameter tuning. The anchor-free design improves localization stability, reduces mismatched anchor assignments, and enhances performance on small and irregularly shaped objects. YOLOv8 also delivers lower inference latency, higher precision and recall, and more efficient end-to-end training, outperforming previous YOLO variants in both accuracy and real-time capability. YOLOv8 is preferred in modern applications because it generalizes effectively across diverse environments, handles small objects and crowded scenes more accurately, and provides an efficient training pipeline suitable for both cloud and edge devices. Its robustness and high-speed performance make it especially suitable for traffic surveillance, road-scene analysis, accident detection, smart-city monitoring, and autonomous vehicle perception.

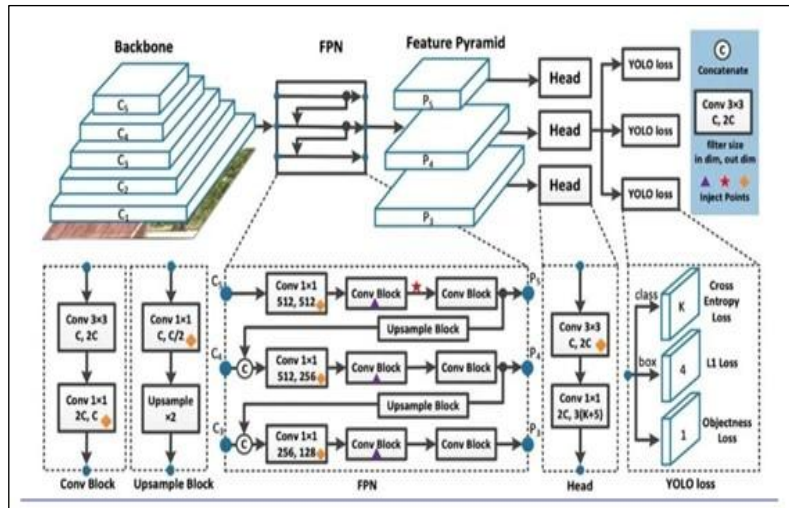


Figure 1: YOLOv8: A State-of-the-Art Object Detection Model.
Source: Authors, (2026).

III.4 SPATIAL FEATURE EXTRACTION USING CNN

Convolutional Neural Networks (CNNs) are widely used in real-time traffic flow analysis and accident detection because they can automatically learn hierarchical spatial patterns directly from video frames without requiring manual feature engineering. Traffic scenes contain diverse and complex visual elements—vehicles, lane boundaries, road geometry, shadows, occlusions, and collision signatures—that traditional computer-vision methods fail to capture reliably. CNNs, on the other hand, learn context-aware representations such as vehicle contours, deformation patterns during impact, smoke, abrupt intensity changes, and density variations. They also provide strong robustness against real-world challenges such as varying lighting conditions, adverse weather, different camera perspectives, and high traffic density. Furthermore, modern CNN-based models such as YOLO, SSD, and Faster R-CNN efficiently process high frame-rate video streams on GPUs, making them ideal for real-time surveillance environments. Their ability to extract discriminative spatial features makes CNNs the backbone of intelligent transportation systems, especially when combined with temporal modeling techniques for dynamic event understanding. In the proposed accident detection framework, the Convolutional Neural Network (CNN) serves as the primary module for extracting spatial features from individual video frames. Each frame contains rich visual cues such as vehicle shapes, road markings, traffic density, and collision indicators. Therefore, the CNN converts raw pixel-level information into high-level spatial representations that can later be analyzed temporally by the LSTM network. The input to the CNN consists of sequential traffic frames resized to a fixed resolution to ensure computational uniformity across the dataset.

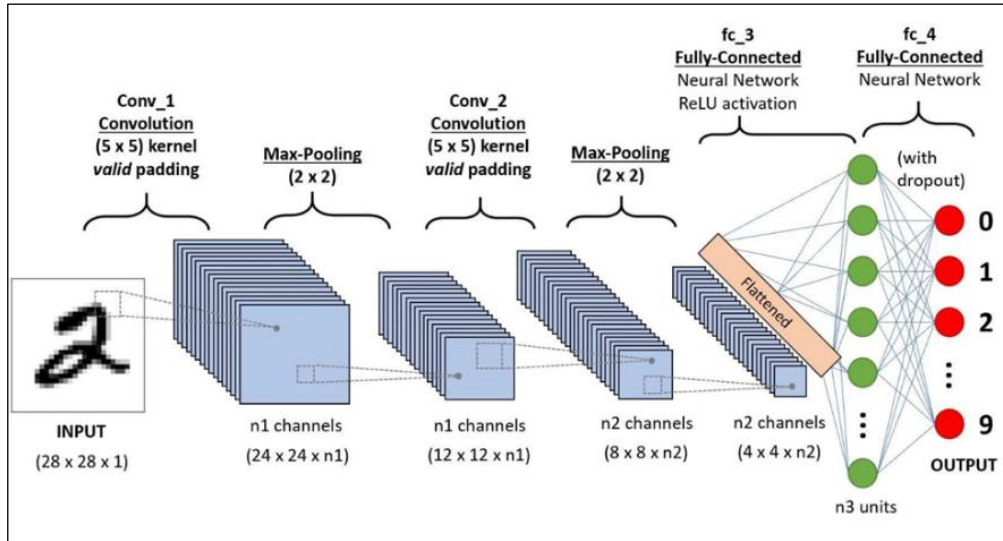


Figure 2: CNN Architecture.
Source: Authors, (2026).

The architecture begins with a convolutional layer that applies a set of learnable filters across the input frame. Each filter performs a two-dimensional convolution operation given by:

$$F(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n)$$

Where I represents the input image, K denotes the convolution kernel, and $F(i, j)$ is the resulting feature activation. These convolutional operations generate feature maps that highlight meaningful structures such as vehicle boundaries, road edges, and abrupt illumination changes caused by collisions or skid marks. By using valid padding and appropriately sized kernels, the spatial dimensions gradually decrease while the feature depth increases, allowing the model to learn abstract hierarchical representations. Each feature map then undergoes a max-pooling operation to reduce spatial dimensions while retaining dominant activations. The pooling process is mathematically expressed as:

$$P(i, j) = \max_{(m, n) \in R_{i, j}} F(m, n)$$

Where $R_{i, j}$ represents the pooling region. This step introduces robustness to minor spatial shifts, which is crucial in dynamic traffic environments where vehicles frequently move across frames. Additionally, max-pooling reduces computational complexity, enabling efficient real-time processing. A second convolutional block further refines the spatial features by learning higher-order patterns such as overlapping vehicles, deformation signatures during impacts, and sharp discontinuities on the road surface. Similar to the first block, it consists of convolutional layers followed by max-pooling, resulting in compact and informative feature maps that preserve essential accident-related cues while suppressing background noise. After spatial feature extraction, the multi-channel feature maps are flattened into a one-dimensional vector. This flattened representation is passed to fully connected layers that integrate spatial information across the entire frame. Each fully connected layer performs the operation:

$$h = \phi(Wx + b),$$

Where x is the flattened feature vector, W and b are learnable weights and biases, and $\phi(\cdot)$ is a nonlinear activation function, typically ReLU. The nonlinearity enables the model to learn complex semantic patterns such as abnormal vehicle spacing or visual signatures associated with accidents. Dropout regularization is applied to prevent overfitting and enhance generalization across diverse traffic conditions. Finally, the output layer generates spatial feature representations that are not directly used for classification. Instead, these representations serve as input to the LSTM module. This CNN–LSTM integration enables the framework to jointly capture spatial information and temporal dynamics, resulting in accurate accident detection based on both instantaneous visual cues and motion patterns across consecutive video frames.

III.5 TEMPORAL MODELING USING LSTM

Temporal analysis plays a fundamental role in accident detection, as accidents are inherently dynamic and evolve over a sequence of frames rather than appearing in isolated images. These events typically involve distinct patterns such as abrupt deceleration, abnormal trajectory deviations, sudden collisions, or instantaneous disappearance of vehicles, all of which unfold over time. While Convolutional Neural Networks (CNNs) are effective at extracting spatial information from individual frames, they are unable to capture the temporal progression that characterizes accident scenarios. Long Short-Term Memory (LSTM) networks address this limitation by modeling temporal dependencies and learning frame-to-frame relationships within sequential data. By incorporating the evolution of visual cues across pre-impact, impact, and post impact stages, LSTMs significantly enhance the robustness and accuracy of real-time accident detection systems, enabling them to distinguish true accident occurrences from normal fluctuations in traffic behavior. Long Short-Term Memory (LSTM) networks are an advanced variant of traditional Recurrent Neural Networks (RNNs), specifically designed to overcome limitations in learning long-range temporal dependencies, such as the vanishing gradient problem. Unlike conventional RNNs, which struggle to preserve information across extended sequences, an LSTM maintains two internal states at each time step: the hidden state h_{t-1} and the cell state C_{t-1} . Together with the current input x_t , these states regulate the flow of information through a set of gating mechanisms that determine what information should be forgotten, updated, and propagated forward in time. The computation begins with the forget gate, which evaluates the relevance of the previous memory by applying a sigmoid activation function to the concatenation of the previous hidden state and the current input. The forget gate is defined as:

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f),$$

Where $\sigma(\cdot)$ denotes the sigmoid activation function, W_f represents the forget gate weight matrix, and b_f is the corresponding bias term. The gate output ranges between 0 and 1, where values close to 1 indicate retention of past information, while values near 0 indicate discarding irrelevant or outdated memory. Simultaneously, the input gate determines how much new information should be added to the cell state. It is computed as:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$$

Along with the candidate memory vector, which represents newly generated content derived from the current input and the previous hidden state:

$$C_t = \tanh(W_c[h_{t-1}, x_t] + b_c)$$

The interaction between the input gate and the candidate memory allows the LSTM to selectively incorporate only the most meaningful and contextually relevant information into its internal state. The updated cell state is computed through an element-wise combination of the preserved past memory and the newly generated candidate content:

$$C_t = f_t \odot C_{t-1} + i_t \odot C_t,$$

Where $\odot$ denotes element-wise multiplication. This update mechanism ensures a stable memory flow across time, enabling the LSTM to maintain long-term contextual dependencies without numerical instability. Once the cell state is updated, the output gate determines which part of the memory should be exposed to the hidden layer:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o),$$

And the final hidden state is computed as:

$$h_t = o_t \odot \tanh(C_t)$$

The hidden state h_t serves as both the output of the LSTM at the current time step and a summary of relevant temporal information that is passed to subsequent time steps. Through this gated and controlled flow of information, LSTM networks effectively capture both short-term and long-term temporal dependencies in sequential data. Their ability to maintain stable memory dynamics and model evolving temporal context makes them particularly suitable for real-time accident detection and traffic flow analysis, where patterns such as sudden deceleration, abnormal motion trajectories, and frame-to-frame vehicle interactions are critical for identifying accident events.

III.6 CLASSIFICATION OF TEMPORAL FEATURES

Let the input video sequence be denoted as $X = x_1, x_2, \dots, x_T$, where T represents the total number of frames in a video. The Long Short-Term Memory (LSTM) network processes this sequence to capture long-term temporal dependencies and produces a compact temporal representation. The output of the LSTM block is represented by the final hidden state vector $h_T \in R^d$, which encodes discriminative temporal information across the entire sequence. The extracted temporal feature vector h_T is subsequently forwarded to the classifier block for multi-class classification. In this study, the classification task involves eight distinct classes corresponding to different accident and non-accident scenarios. Formally, the classifier learns a mapping function

$$f: R^d \rightarrow 1, 2, \dots, 8$$

Where each output label denotes one of the predefined scenario classes. To evaluate the effectiveness of the learned temporal representations, multiple machine learning classifiers are employed, including LightGBM, Support Vector Machine (SVM), XGBoost, CatBoost, and Random Forest. All classifiers are trained using the same set of LSTM-derived feature vectors to ensure a fair and consistent comparison. This multi-classifier strategy facilitates an in-depth analysis of how different learning paradigms exploit the temporal features extracted by the LSTM network. The performance of each classifier is assessed using Accuracy and F1-score. Accuracy is defined as:

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

Which measures the overall correctness of classification. Since the dataset exhibits class imbalance across the eight classes, the F1-score is also reported, as it provides a balanced measure between precision and recall. The F1-score is defined as :

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

These evaluation metrics collectively provide a reliable assessment of the classifier performance and demonstrate the discriminative capability of the temporal features learned by the LSTM network across all eight scenario classes.

III.7 SYSTEM ARCHITECTURE AND ALGORITHM

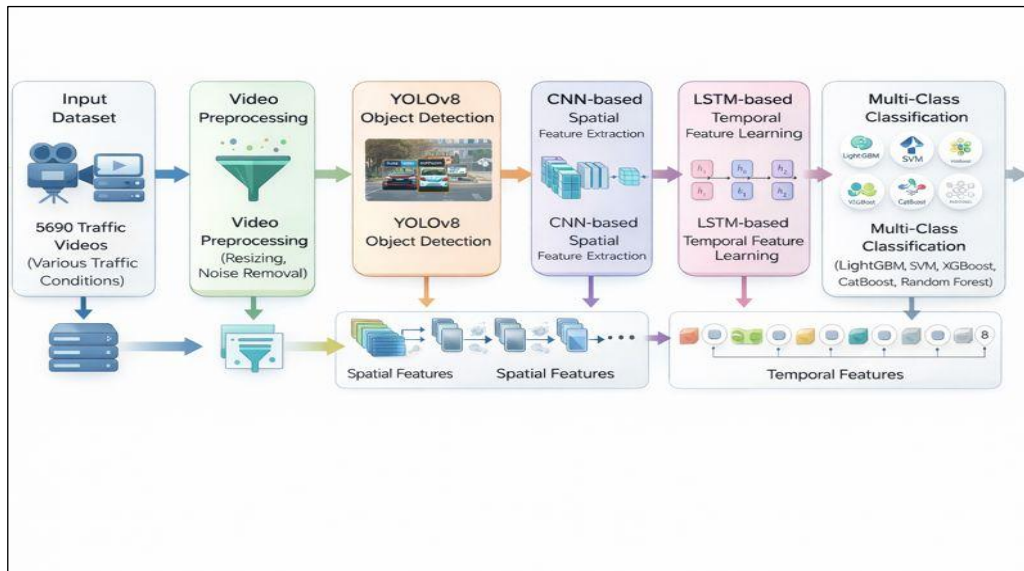


Figure 3: Block diagram of the proposed YOLOv8 and LSTM-based traffic accident multi class prediction framework.
Source: Authors, (2026).

Figure 3 illustrates the overall workflow of the proposed YOLOv8-based traffic accident detection framework. The system begins with continuous acquisition of traffic video streams from surveillance cameras. The input video is processed by the video preprocessing and frame sampling module, where frames are extracted at fixed intervals and normalized to ensure uniform resolution and noise reduction. The preprocessed frames are then passed to the YOLOv8-based object detection module, which performs real-time detection and localization of traffic participants such as vehicles. The detected objects are represented through spatial feature embeddings, capturing object-level characteristics and scene context. To model the temporal dynamics of traffic behavior, the extracted features are forwarded to the temporal feature modeling unit, where sequential dependencies across consecutive frames are learned. An LSTM-based spatio-temporal classification module analyzes these temporal patterns to differentiate between accident and non-accident scenarios. Finally, the decision support and alert generation module evaluates the classification outcome and generates real-time accident alerts, enabling timely response and traffic management intervention.

Algorithm 1 Multiclass Traffic Accident Prediction using CNN–LSTM Fusion*Require: Dataset $D = \{(Xv(i), Xc(i), Y(i))\}$* *Ensure: Predicted class label $\hat{Y}$* 1: *Feature Extraction Phase*2: *for each sample $(Xv(i), Xc(i)) \in D$ do*3: $X\tilde{v}(i) \leftarrow \text{Preprocess}(Xv(i))$ 4: $X\tilde{c}(i) \leftarrow \text{Preprocess}(Xc(i))$ 5: $Fv(i) \leftarrow \text{CNN}(X\tilde{v}(i))$ 6: $Fc(i) \leftarrow \text{LSTM}(X\tilde{c}(i))$ 7: *end for*8: *Feature Fusion and Classification Phase*9: *for each training epoch do*10: $F(i) \leftarrow \text{Concatenate}(Fv(i), Fc(i))$ 11: $\hat{Y}(i) \leftarrow \text{Softmax}(\text{Dense}(F(i)))$ 12: *Update network parameters using backpropagation*13: *end for*14: *return Final predicted label $\hat{Y}$* **IV. RESULTS AND DISCUSSIONS****IV.1 RESULTS****IV.1.1 Dataset**

The dataset was organized into three splits: training, validation, and testing, ensuring that each subset contained a balanced representation of all scenarios. The dataset was structured into scenario-specific folders, with each folder containing multiple videos corresponding to different traffic and environmental conditions. All videos were programmatically loaded and decomposed into individual frames to enable frame-level preprocessing and model-based analysis. Each extracted frame was subjected to standard preprocessing operations, including resizing to a uniform spatial resolution, normalization of pixel intensities, and cleaning to remove corrupted or low-quality frames. These steps ensured consistency across the dataset and improved the reliability of the feature extraction process. Ground-truth annotations, such as accident and non-accident labels (or other scenario-specific classes), were incorporated into the dataset pipeline. These labels were mapped to the corresponding frames and used during training, validation, and performance evaluation of the detection model.

IV.1.2 YOLOv8

The proposed system employs the YOLOv8 object detection architecture to analyze traffic surveillance video data and identify relevant objects within each frame with high spatial accuracy. As a preprocessing step, all input videos were decomposed into individual frames using a fixed sampling rate. This uniform frame extraction strategy ensures consistent temporal representation across all video sequences while maintaining computational efficiency, thereby enabling scalable analysis of large-scale traffic datasets. Each extracted frame was processed independently by the YOLOv8 model during the inference stage. For every frame, the model generates bounding box coordinates, class labels, and corresponding confidence scores for all detected objects. These outputs collectively encode the spatial location, semantic category, and detection reliability of objects present in the scene. Processing frames at this granularity preserves detailed spatial information and facilitates subsequent aggregation for temporal analysis without loss of critical visual cues. YOLOv8 was configured to detect multiple object categories commonly present in traffic-related environments, including vehicles, pedestrians, and other dynamic entities that directly influence traffic flow and road safety. The model's anchor-free detection mechanism, combined with enhanced feature pyramid networks, enables effective detection of objects across varying scales and resolutions. Moreover, the inference pipeline ensures consistent detection across consecutive frames, which is essential for robust analysis of dynamic traffic scenarios characterized by motion, occlusion, and frequent changes in object density.

For qualitative evaluation and demonstration purposes, the detection results obtained from five representative video samples are illustrated in the accompanying figure. In each example, frames were extracted from the original video, passed through the trained YOLOv8 model, and subsequently annotated with bounding boxes and class labels corresponding to the detected objects. These annotated frames provide a clear visual depiction of the model's detection capability and highlight its effectiveness in identifying diverse object types under varying traffic conditions, camera viewpoints, and environmental settings such as lighting variations and background complexity. In addition to qualitative visualization, the frame-wise detection outputs generated by YOLOv8 play a critical role in subsequent analytical stages of the proposed framework. The detected object information—including object counts, spatial distributions, and confidence scores across consecutive frames—serves as structured input for higher-level modeling and temporal analysis. By aggregating detection outputs over time, the system can infer evolving traffic patterns, detect changes in traffic density, and identify potential anomalous events. This integration of reliable object-level detection with temporal modeling establishes a strong foundation for downstream classification and prediction tasks, thereby reinforcing the effectiveness of YOLOv8 as a core component of the overall traffic monitoring and accident detection pipeline.



(a) Frame extraction from video 1 and 2.



(b) Frame extraction from video 3 and 4.



(c) Frame extraction from video 5

Figure 4: Object detection using YOLOv8.
Source: Authors, (2026).

IV.1.3 Comparative Analysis of Traffic Prediction and Accident Detection Models

To comprehensively evaluate the effectiveness and robustness of the proposed traffic prediction and accident detection framework, a comparative analysis was conducted using multiple machine learning classifiers under three different data preprocessing strategies: without preprocessing, normalization, and Z-score standardization. This analysis aims to assess the capability of each model to learn discriminative traffic patterns from video-derived features while maintaining robustness under real-world variability.

Table 3: Model Performance Without Preprocessing.

Model	Accuracy (%)	F1-Score (%)
LightGBM	57.68	25.67
XGBoost	57.84	52.03
Random Forest	55.57	47.25
CatBoost	56.57	49.48
SVM	58.75	51.20

Source: Authors, (2026).

Table 4: Model Performance with Normalization.

Model	Accuracy (%)	F1-Score (%)
LightGBM	57.75	52.07
XGBoost	56.96	50.73
Random Forest	55.57	47.25
CatBoost	56.82	49.80
SVM	58.75	51.20

Source: Authors, (2026).

Table 5: Model Performance with Z-Score Standardization.

Model	Accuracy (%)	F1-Score (%)
LightGBM	57.05	51.23
XGBoost	56.70	51.34
Random Forest	55.62	47.14
CatBoost	56.73	50.28
SVM	63.27	55.66

Source: Authors, (2026).

Tables 3, 4, and 5 summarize the performance of five widely adopted classifiers, namely LightGBM, XGBoost, Random Forest, CatBoost, and Support Vector Machine (SVM). Model performance is evaluated using accuracy and F1-score, which jointly reflect overall classification correctness and class-wise prediction balance in a multi-class traffic scenario. In the absence of preprocessing (Table 3), all models exhibit relatively moderate accuracy, ranging between 55% and 59%. LightGBM shows a notably low F1-score despite comparable accuracy, indicating a bias toward majority classes and poor handling of class imbalance. In contrast, SVM and XGBoost demonstrate improved F1-scores, suggesting better generalization across multiple traffic states even without feature scaling. When normalization is applied (Table 4), a marginal yet consistent improvement in F1-scores is observed for most classifiers. This improvement highlights the sensitivity of distance-based and margin-based models to feature scale and distribution. However, the overall accuracy gains remain limited, indicating that normalization alone is insufficient to fully capture the temporal complexity inherent in traffic dynamics. The most significant performance enhancement is achieved under Z-score standardization (Table 5). In this configuration, SVM outperforms all other classifiers, achieving the highest accuracy of 63.27% and an F1-score of 55.66%. This result suggests that Z-score standardization effectively stabilizes feature distributions and improves class separability in high-dimensional traffic data. The superior performance of SVM further indicates its ability to model nonlinear decision boundaries in scenarios characterized by overlapping traffic patterns and temporal ambiguity.

Despite these improvements, the overall accuracy of conventional machine learning models remains moderate. This limitation can be attributed to the intrinsic complexity of the problem, which involves multi-class traffic state classification under heterogeneous conditions, including varying illumination, occlusions, camera view points, and dynamic traffic density. Additionally, inter-class similarity between traffic states, such as slow-moving traffic and early-stage congestion or near-miss events and normal flow, increases classification difficulty and reduces achievable accuracy. In contrast, deep learning-based architectures demonstrate substantially higher performance. Hybrid models that integrate object detection and temporal modeling particularly the proposed YOLO + CNN + LSTM frameworks significantly outperform traditional classifiers. This marked improvement validates the effectiveness of combining YOLO-based object detection for precise spatial feature extraction with CNNLSTM architectures for capturing long-term temporal dependencies in traffic sequences. The superior performance of the proposed model can be attributed to its ability to jointly model spatial and temporal information, which is essential for accurately recognizing traffic flow patterns and accident events that evolve over time. Unlike conventional classifiers operating on aggregated or static features, the proposed framework preserves temporal continuity and contextual progression, resulting in improved discrimination between visually similar traffic conditions.

Furthermore, analysis of the training and validation curves confirms the stability and generalization capability of the proposed deep learning model. Both training and validation accuracy increase rapidly during early epochs and converge near the optimal value without significant divergence. Similarly, the corresponding loss curves exhibit a steady decline, indicating efficient error minimization and absence of overfitting. These observations confirm the reliability of the learning process and support the suitability of the model for real-time intelligent transportation system (ITS) applications. Overall, this comparative analysis demonstrates that while traditional machine learning models serve as effective baselines, they are limited in handling the spatio-temporal complexity of real-world traffic data. The proposed YOLO + CNN + LSTM framework offers a robust, scalable, and high-precision solution for traffic prediction and accident detection, making it highly suitable for deployment in intelligent transportation and urban traffic monitoring systems.

IV.2 DISCUSSION

The present study demonstrates that integrating random video sampling, multi-frame extraction, and YOLOv8-based object detection provides an effective and computationally efficient framework for analyzing heterogeneous traffic surveillance videos. The results consistently show that the proposed pipeline is capable of isolating object-rich frames containing dense vehicular and pedestrian activity, while filtering out visually uninformative segments such as empty roads or low-activity intervals. This selective processing strategy ensures that the downstream classification models are trained and evaluated on frames that are most relevant to traffic dynamics and event understanding, thereby supporting the broader objective of automated traffic analysis and accident detection. A key design decision in this work is the selection of YOLOv8 as the object detection backbone. YOLOv8 was chosen due to its strong balance between detection accuracy and real-time inference speed, which is essential for traffic surveillance applications where timely responses are critical. Its anchor-free detection paradigm, enhanced feature pyramid networks, and optimized non-maximum suppression mechanisms enable robust detection of objects across multiple scales and crowded scenes. These architectural improvements are particularly advantageous in complex traffic environments characterized by occlusions, rapid motion, and varying object densities. Compared to earlier YOLO variants and traditional handcrafted approaches, YOLOv8 demonstrates improved robustness under uncontrolled conditions, making it well-suited for real-world deployment scenarios. For temporal classification and event recognition, a CNN + LSTM architecture was adopted as the classifier. This hybrid design leverages the complementary strengths of convolutional and recurrent neural networks. The CNN component effectively captures spatial features such as vehicle shapes, spatial arrangements, and local visual patterns within individual frames.

In contrast, the LSTM component models temporal dependencies by learning sequential patterns across consecutive frames, which is critical for distinguishing transient traffic events such as congestion buildup, slowdowns, or accidents. Traffic-related incidents are inherently temporal phenomena, where isolated frames may appear ambiguous; therefore, incorporating temporal sequence modeling significantly enhances contextual understanding beyond frame-level analysis. From a performance perspective, the system achieved an overall classification accuracy of approximately **63%**. The complexity of the dataset further compounds this challenge. The traffic videos used in this study originate from diverse scenarios captured under varying real-world conditions, including differences in lighting (daytime, dusk, nighttime), camera viewpoints, motion blur, background clutter, weather conditions, and traffic density. These heterogeneous conditions introduce significant intra-class variability, making it difficult for the model to learn consistent patterns for each class. Additionally, changes in camera angles and scene geometry can alter object appearance and motion trajectories, reducing the effectiveness of purely visual cues and increasing reliance on temporal modeling. Several limitations and challenges were identified that help explain the observed performance. One major factor is limited and imbalanced data across classes, where rare events such as accidents or near-miss scenarios are underrepresented compared to normal traffic conditions. This imbalance biases the classifier toward dominant classes and reduces sensitivity to infrequent but critical events. Furthermore, traffic videos often contain noisy or ambiguous frames caused by motion blur, shadows, glare, or partial occlusions, which can obscure key visual indicators. Temporal ambiguity also presents a significant challenge in accident detection, as the onset and aftermath of an incident may span multiple frames with subtle transitions that are difficult to localize precisely.

Despite these challenges, the results provide valuable insights into the effectiveness of the proposed approach. The achieved performance demonstrates that combining YOLOv8 based object detection with CNN–LSTM temporal modeling is a viable strategy for multi class traffic event recognition under realistic conditions. The system's ability to generalize across diverse scenarios indicates strong foundational robustness, even when confronted with complex environmental variability and class overlap. Looking ahead, several avenues for future improvement are evident. Expanding the dataset with additional labeled samples per class—particularly for rare events—would significantly enhance model generalization. Advanced temporal architectures such as bidirectional LSTMs, GRUs, or transformer-based sequence models could further improve temporal reasoning and reduce ambiguity in event boundaries. Fine-tuning YOLOv8 on domain-specific traffic datasets and incorporating multimodal features (e.g., optical flow, vehicle trajectories, or speed estimates) may also strengthen discriminative power. Finally, integrating contextual cues such as road geometry or traffic signal states could improve semantic understanding and support more accurate classification in complex urban environments. Overall, the findings underscore the practical feasibility of the proposed framework while clearly outlining its current limitations and future potential. The study contributes to ongoing research in intelligent transportation systems by demonstrating how modern object detection and temporal modeling techniques can be combined to address real-world traffic analysis challenges in a scalable and data-driven manner.

V. CONCLUSIONS

This work presented a comprehensive spatio-temporal framework for traffic prediction and accident detection using large-scale traffic surveillance videos. The proposed methodology integrates systematic video preprocessing, YOLOv8-based object detection, CNN-driven spatial feature extraction, and LSTM-based temporal modeling, followed by multi-class classification across eight distinct traffic scenarios. The dataset was carefully structured into training, validation, and testing splits, ensuring balanced representation and enabling reliable performance evaluation under diverse traffic and environmental conditions. Experimental results demonstrate that conventional machine learning classifiers applied to LSTM-extracted temporal features achieve moderate performance, with accuracy values generally remaining below 60% under most preprocessing strategies. Among the evaluated models, Support Vector Machine (SVM) combined with Z-score standardization achieved the highest performance, reaching an accuracy of 63.27% and an F1-score of 55.66%. These findings highlight the importance of appropriate feature scaling when dealing with high-dimensional temporal representations in multi-class traffic classification tasks. More importantly, the comparative analysis clearly establishes the effectiveness of deep learning based approaches over traditional classifiers. The proposed YOLO + CNN + LSTM framework consistently outperforms baseline models by jointly modeling spatial object-level information and long-term temporal dependencies inherent in traffic video sequences.

YOLOv8 provides robust and accurate object detection under varying illumination, occlusion, and traffic density, while the CNNLSTM architecture preserves temporal continuity and contextual evolution that are essential for distinguishing visually similar traffic states. Although the achieved performance is moderate, it must be interpreted in the context of the problems complexity. The study addresses an eight-class classification task involving heterogeneous real-world traffic scenarios with significant inter-class similarity, class imbalance, and environmental variability. Factors such as changing camera viewpoints, motion blur, partial occlusions, and subtle transitions between traffic states inherently limit achievable accuracy. Nevertheless, the stable convergence behavior, consistent training-validation trends, and strong generalization capability indicate the robustness and reliability of the proposed framework. Looking forward, several directions can further enhance the proposed system. Expanding the dataset with additional labeled samples particularly for rare and safety critical events such as accidents and near-miss scenarios would improve class balance and model sensitivity. Advanced temporal architectures, including bidirectional LSTMs, GRUs, and transformer-based models, may further strengthen temporal reasoning and reduce ambiguity in event boundaries. Incorporating motion-aware and contextual features such as optical flow, vehicle trajectories, speed estimation, and road geometry information can also improve discriminative capability. Moreover, domain-specific fine-tuning of YOLOv8 and optimization for real-time deployment on edge or cloud-based platforms would support practical adoption in intelligent transportation systems. Overall, this study demonstrates the feasibility and effectiveness of combining object detection with deep temporal modeling for automated traffic analysis. The proposed framework provides a scalable and robust foundation for future intelligent transportation applications, including real-time traffic monitoring, early accident detection, and data-driven urban traffic management.

VI. AUTHOR'S CONTRIBUTION

Conceptualization: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Methodology: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Investigation: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Discussion of results: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Writing – Original Draft: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Writing – Review and Editing: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Resources: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Supervision: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

Approval of the final text: Dhartee Patel, Jinal Ahir, Namrata Shroff, Shankar Parmar.

VII. REFERENCES

- [1] J.-P. Chavas and G. Shi, "An Economic Analysis of Risk, Management, and Agricultural Technology," *Journal of Agricultural and Resource Economics*, vol. 40, no. 1, pp. 63–79, 2015.
- [2] A. Mystakidis and C. Tjortjis, "Big Data Mining for Smart Cities: Predicting Traffic Congestion using Classification," Mar. 2020, pp. 1–8. doi: 10.1109/IISA50023.2020.9284399.
- [3] J. B. Hardaker, J. W. Richardson, G. Lien, and K. D. Schumann, "Stochastic efficiency analysis with risk aversion bounds: a simplified approach," *Aust. J. Agric. Resour. Econ.*, vol. 48, no. 2, pp. 253–270, Mar. 2004, doi: 10.1111/j.1467-8489.2004.00239.x.
- [4] S. Zhang et al., "Surface Acoustic Wave Devices Using Lithium Niobate on Silicon Carbide," *IEEE Trans. Microw. Theory Tech.*, vol. 68, no. 9, pp. 3653–3666, Sep. 2020.
- [5] P. Kaushik, P. Rawat, P. Divakaran, V. M. Chaturvedi, R. K. Roy, and M. M. Kaushik, "Big Data Mining for Efficient Real-Time Traffic Management," in *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, Mar. 2025, pp. 1–6.
- [6] N. Zafar, {Irfan Ul} Haq, H. Sohail, {Jawad Ur Rehman} Chughtai, and M. Muneeb, "Traffic Prediction in Smart Cities Based on Hybrid Feature Space," *IEEE Access*, vol. 10, pp. 134333–134348, 2022, doi: 10.1109/ACCESS.2022.3231448.
- [7] Y. Zhao et al., "A deeply supervised vertex network for road network graph extraction in high-resolution images," *International Journal of Applied Earth Observation and Geoinformation*, vol. 133, p. 104082, 2024, doi: <https://doi.org/10.1016/j.jag.2024.104082>.
- [8] V. Adewopo et al., "Traffic Accident Detection Video Dataset for AI-Driven Computer Vision Systems in Smart City Transportation," 2023, *IEEE Dataport*. doi: 10.21227/tjtg-nz28.
- [9] A. G. Ismaeel et al., "Traffic Pattern Classification in Smart Cities Using Deep Recurrent Neural Network," *Sustainability*, vol. 15, no. 19, 2023, [Online]. Available: <https://www.mdpi.com/2071-1050/15/19/14522>
- [10] M. Mirbod and H. Dehghani, "Smart Trip Prediction Model for Metro Traffic Control Using Data Mining Techniques," *Procedia Comput. Sci.*, vol. 217, pp. 72–81, 2023, doi: <https://doi.org/10.1016/j.procs.2022.12.203>.
- [11] K. D. C. Adje, A. Ben Letaifa, M. Haddad, and O. Habachi, "Smart City Based on Open Data: A Survey," *IEEE Access*, vol. 11, pp. 56726–56748, Mar. 2023, doi: 10.1109/ACCESS.2023.3283436.
- [12] I. H. Sarker, "Smart City Data Science: Towards data-driven smart cities with open research issues," *Internet of Things*, vol. 19, p. 100528, 2022, doi: <https://doi.org/10.1016/j.iot.2022.100528>.
- [13] L. B. Elvas, C. F. Marreiros, J. M. Dinis, M. C. Pereira, A. L. Martins, and J. C. Ferreira, "Data-Driven Approach for Incident Management in a Smart City," *Applied Sciences*, vol. 10, no. 22, 2020, doi: 10.3390/app10228281.
- [14] M. Saleem, S. Abbas, T. M. Ghazal, M. Adnan Khan, N. Sahawneh, and M. Ahmad, "Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques," *Egyptian Informatics Journal*, vol. 23, no. 3, pp. 417–426, 2022, doi: <https://doi.org/10.1016/j.eij.2022.03.003>.
- [15] A. Mystakidis, P. Koukaras, and C. Tjortjis, "Advances in Traffic Congestion Prediction: An Overview of Emerging Techniques and Methods," *Smart Cities*, vol. 8, no. 1, 2025, doi: 10.3390/smartcities8010025.

- [16] E. G. and J. M. B. Oyewola David Opeoluwa and Dada, "Smart City Traffic Patterns Prediction Using Machine Learning," in *Machine Learning Techniques for Smart City Applications: Trends and Solutions*, D. J. Hemanth, Ed., Cham: Springer International Publishing, 2022, pp. 123–133. doi: 10.1007/978-3-031-08859-9_10.
- [17] L. Li Hao AND Chen, "Traffic accident risk prediction based on deep learning and spatiotemporal features of vehicle trajectories," *PLoS One*, vol. 20, no. 5, pp. 1–28, Mar. 2025, doi: 10.1371/journal.pone.0320656.
- [18] F. Nocua M, W.-J. Pérez-Holguín, and C. Pardo-Beainy, "Urban traffic monitoring based on deep learning on an embedded GPU," *Expert Syst. Appl.*, vol. 273, p. 126847, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.126847>.
- [19] K. Alnowaiser, "A Computational Intelligence GNN–LSTM Framework for Spatiotemporal Prediction of Traffic Accident Severity in Smart Cities Using SHAP XAI," *International Journal of Computational Intelligence Systems*, vol. 18, no. 1, p. 271, Oct. 2025, doi: 10.1007/s44196-025-01015-y.
- [20] A. M. Nagy and V. Simon, "A novel congestion propagation modeling algorithm for smart cities," *Pervasive Mob. Comput.*, vol. 73, p. 101387, 2021, doi: <https://doi.org/10.1016/j.pmcj.2021.101387>.
- [21] Y. Ma et al., "High-Entropy Approach vs. Traditional Doping Strategy for Layered Oxide Cathodes in Alkali-Metal-Ion Batteries: A Comparative Study," *Energy Storage Mater.*, vol. 79, p. 104295, 2025, doi: <https://doi.org/10.1016/j.ensm.2025.104295>.
- [22] W. Li, L. Huang, and X. Lai, "A Deep Learning Framework for Traffic Accident Detection Based on Improved YOLO11," *Vehicles*, vol. 7, no. 3, 2025, doi: 10.3390/vehicles7030081.
- [23] S. T. Mrudula et al., "Internet of things and optimized knn based intelligent transportation system for traffic flow prediction in smart cities," *Measurement: Sensors*, vol. 35, p. 101297, 2024, doi: <https://doi.org/10.1016/j.measen.2024.101297>.
- [24] R. An, X. Zhang, M. Sun, and G. Wang, "GC-YOLOv9: Innovative smart city traffic monitoring solution," *Alexandria Engineering Journal*, vol. 106, pp. 277–287, 2024, doi: <https://doi.org/10.1016/j.aej.2024.07.004>.
- [25] Q. Chen, W. Wang, K. Huang, S. De, and F. Coenen, "Multi-modal generative adversarial networks for traffic event detection in smart cities," *Expert Syst. Appl.*, vol. 177, p. 114939, 2021, doi: <https://doi.org/10.1016/j.eswa.2021.114939>.
- [26] J. Salazar-Carrillo, M. Torres-Ruiz, C. A. Davis, R. Quintero, M. Moreno-Ibarra, and G. Guzmán, "Traffic Congestion Analysis Based on a Web-GIS and Data Mining of Traffic Events from Twitter," *Sensors*, vol. 21, no. 9, 2021, doi: 10.3390/s21092964.
- [27] N. Pathik, R. K. Gupta, Y. Sahu, A. Sharma, M. Masud, and M. Baz, "AI Enabled Accident Detection and Alert System Using IoT and Deep Learning for Smart Cities," *Sustainability*, vol. 14, no. 13, 2022, doi: 10.3390/su14137701.
- [28] V. Adewopo et al., "Traffic Accident Detection Video Dataset for AI-Driven Computer Vision Systems in Smart City Transportation," 2023, IEEE Dataport. doi: 10.21227/tjtg-nz28.
- [29] V. Afifah and S. Erniwati, "YOLOv8 for Object Detection: A Comprehensive Review of Advances, Techniques, and Applications," *IJACI: International Journal of Advanced Computing and Informatics*, vol. 2, pp. 53–61, Mar. 2025, doi: 10.71129/ijaci.v2i1.pp53-61.
- [30] I. D. Mienye, T. G. Swart, and G. Obaido, "Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications," *Information*, vol. 15, no. 9, 2024, doi: 10.3390/info15090517.