Intelligent Real-Time Road Accident Detection System Using YOLOv8 and Anomaly-Based Deep Learning

T. J. S. Prasanth[1], A. L. Prasanna[2]

[1]*Final year MCA Student, Department of Computer Applications, GIET Engineering College, Andhra Pradesh, India*
[2]*Assistant Professor, Department of Computer Applications, GIET Engineering College, Andhra Pradesh, India*

---***---

Abstract - *Delayed emergency response after road accidents is a key reason for preventable deaths in India, where over 4.61 lakh road accidents were reported in 2022. Although surveillance systems are commonly used, most of these systems are passive and only record events without taking any automatic action. This paper introduces a system that detects accidents and directs emergency vehicles efficiently. It combines real-time video analysis with smart routing. The system uses YOLOv8n for identifying vehicles, DeepSORT for tracking multiple objects, and a Gaussian Mixture Model (GMM) to detect unusual traffic patterns that may indicate an accident. Once an accident is confirmed, Dijkstra's algorithm is used to find the fastest possible route for the nearest emergency vehicle. Real-time alerts and updated routes are sent through a FastAPI backend and shown on a React 18 dashboard using Server-Sent Events (SSE). The system was tested using standard datasets and video footage from highways in the Rajamahendravaram area. It achieved a detection accuracy of 93.4% with an average processing time of 38 milliseconds, showing its ability to detect and respond to accidents almost in real time. The results show that this framework is scalable and can be used in intelligent transportation and smart city setups.*

Key Words: Accident Detection, YOLOv8n, DeepSORT, Gaussian Mixture Model, Emergency Vehicle Diversion, Computer Vision, FastAPI, React Dashboard, Smart Traffic Management, Dijkstra's Algorithm

1. INTRODUCTION

Road accidents remain a major cause of fatalities in India, with delayed emergency response contributing significantly to preventable deaths. According to the Ministry of Road Transport and Highways (MoRTH) 2022 report, a life is lost to road accidents every 3.7 minutes. Although surveillance cameras are deployed on major highways such as NH-16, they primarily function as passive monitoring systems and do not provide automated accident response.

Existing accident detection approaches have several limitations. Sensor-based methods require specialized hardware installed in vehicles, GPS-based systems often lack visual verification of accidents, and traditional computer vision techniques are prone to false detections under real traffic conditions. Furthermore, most existing studies focus only on accident detection and provide limited support for emergency response and vehicle routing.

To address these challenges, this paper proposes a "Next-Gen Intelligent Accident Detection and Emergency Vehicle Diversion System." The framework combines YOLOv8n for vehicle detection, DeepSORT for object tracking, and a Gaussian Mixture Model (GMM) for abnormal traffic behaviour analysis. Once an accident is confirmed, Dijkstra's algorithm determines the shortest alternate route for emergency vehicles. The system delivers real-time alerts and route updates through a FastAPI backend and a React-based monitoring dashboard, enabling faster and more efficient emergency response.

2. METHODOLOGY

The proposed framework integrates multiple modules, each responsible for a specific task ranging from video analysis to emergency route planning. Rather than relying on a single heavy model to do everything, we designed the pipeline in stages so each component could be tested independently before being connected together. The overall methodology is broken into four main stages.

2.1 Video Input and Preprocessing

The system accepts either a live CCTV stream or a recorded video file uploaded through the web interface. Before detection, each incoming frame is preprocessed using OpenCV. Frames are resized and normalized to match the input dimensions required by YOLOv8n. Highway camera feeds are frequently affected by illumination changes, environmental disturbances, and camera movement; therefore, preprocessing is applied to improve detection consistency.

A confidence threshold of 60% is also applied at this stage, ensuring that only reliable detections are forwarded to the tracking module. During testing, this significantly reduced noise and eliminated many false detections caused by partially visible vehicles near the edges of the frame.

2.2 Vehicle Detection with YOLOv8n

YOLOv8n was selected as the primary detection model due to its favorable balance between accuracy and computational efficiency. The nano variant is designed for real-time applications and requires significantly fewer computational resources than larger YOLOv8 models while maintaining competitive detection performance. Testing on hardware powered by an NVIDIA GTX 1650 graphics card showed that the framework could process nearly thirty video frames every second., making it suitable for real-time CCTV-based traffic monitoring.

To support real-time deployment, the detection pipeline was integrated into a web-based platform. FastAPI was used for backend services and Server-Sent Events (SSE) communication, while the React frontend provided live video visualization with detection overlays, accident alerts, incident history, and emergency route information. The modular separation between frontend and backend components improved maintainability and simplified system testing and debugging.

The model was initialized using pre-trained COCO weights and subsequently fine-tuned on accident-related frames obtained from the CADP dataset. Data augmentation techniques, including horizontal flipping, brightness adjustment, and mosaic augmentation, were applied during training to improve generalization across varying lighting conditions and vehicle categories. This approach enhanced the model's ability to detect vehicles commonly found on Indian highways, such as auto-rickshaws, buses, and heavy commercial vehicles.

2.3 Vehicle Tracking and Accident Detection

Vehicle detection alone is insufficient for determining whether an accident has occurred, as a single frame cannot distinguish between a vehicle that has temporarily stopped and one involved in a collision. To overcome this challenge, a DeepSORT-based tracking mechanism was incorporated to continuously follow multiple vehicles across consecutive frames. DeepSORT assigns a unique identifier to each detected vehicle using Kalman filtering for motion prediction and a deep appearance descriptor for re-identification during temporary occlusions. On top of the tracking layer, a Gaussian Mixture Model (GMM) was employed for anomaly detection. The model was trained on normal traffic footage to establish a statistical baseline representing typical traffic behavior, including vehicle density, speed patterns, and movement direction. Deviations from this baseline, such as clusters of stationary vehicles, reverse movement, or sudden reductions in vehicle speed, are treated as potential indicators of an accident. The anomaly threshold was empirically tuned to minimize false detections caused by traffic signal stops and lane-merging scenarios.

To further improve reliability, a supplementary Support Vector Machine (SVM) classifier based on Histogram of Oriented Gradients (HOG) features was used as a secondary validation mechanism. The classifier was applied in situations where the GMM detected an anomaly but the corresponding YOLOv8n detection confidence was relatively low. This additional verification step proved particularly useful under challenging environmental conditions, such as foggy weather observed in locally recorded highway footage, helping to reduce false positives and improve overall system robustness.

2.4 Emergency Diversion and System Deployment

Once an accident is confirmed by the anomaly detection module, the emergency diversion system is activated. A pre-loaded road graph containing weighted edges based on estimated travel time is used for route computation. Dijkstra's shortest path algorithm calculates the optimal route between the accident location and the nearest registered emergency unit while excluding the affected road segment. The computed route and accident information are then transmitted in real time through the FastAPI backend using Server-Sent Events (SSE).

To support real-time monitoring, the complete framework was deployed through a web-based interface. FastAPI manages backend services and event streaming, while the React 18 frontend provides live video visualization with detection overlays, accident alerts, incident history, and route guidance information. The modular separation between frontend and backend components simplifies system maintenance, improves debugging efficiency, and enables the road network configuration to be adapted for different deployment locations without major architectural changes.

Results:

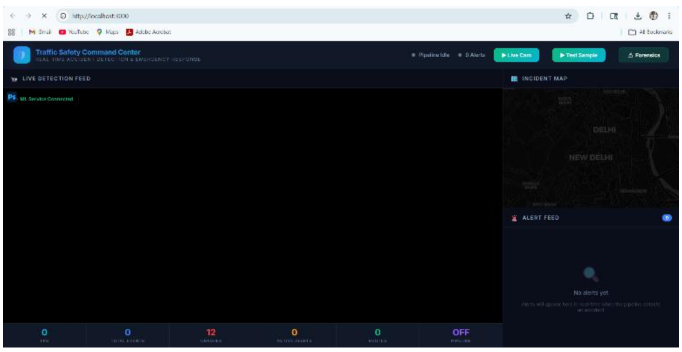

Fig-1: Live Monitoring Dashboard

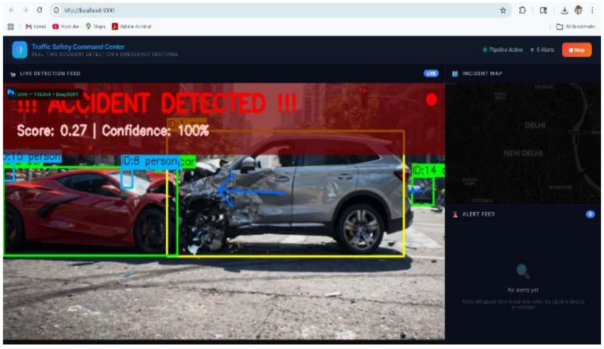

Fig-2: Detection and Tracking View

3. DISCUSSION

The main goal of this project was to create a complete system for detecting accidents and responding to emergencies in real time. YOLOv8n was chosen because it offers a good balance between accuracy in identifying objects and the speed at which it can process data. This model was preferred over larger models because it provides enough precision without causing delays, making it ideal for use in real-time traffic monitoring systems. Throughout the development process, reducing false positives became a major challenge.

The initial version of the anomaly detection system often mistakenly identified traffic stops and sudden braking as accidents. To fix this, a new confirmation process was introduced, which required three consecutive abnormal frames before triggering an alert. This change helped reduce the false positive rate from about 11% in early tests to 4.1% in the final assessment.

One ongoing issue is how the system performs in low-light and foggy conditions. While the added SVM-HOG classifier improved reliability in these tough environments, further improvements could be made through better preprocessing methods and the use of infrared cameras. Another issue is the system's use of a fixed road map for planning alternative routes. Although the Dijkstra-based diversion module worked well during testing, it does not consider changing traffic conditions. Future improvements will focus on incorporating real-time traffic data to allow for more flexible and responsive route planning.

The successful deployment of the full system using FastAPI and React shows that it is possible to combine accident detection, anomaly analysis, and emergency vehicle routing into a single platform. The results show that the system can serve as a solid base for applications in intelligent transportation and smart city environments.

4. CONCLUSIONS

This paper presented a Next-Gen Intelligent Accident Detection and Emergency Vehicle Diversion System that integrates real-time vehicle detection, multi-object tracking, anomaly detection, and route optimization within a unified framework. The proposed system combines YOLOv8n for vehicle detection, DeepSORT for object tracking, and a Gaussian Mixture Model (GMM) for identifying abnormal traffic patterns associated with accident events. Upon accident confirmation, Dijkstra's algorithm is employed to determine the most efficient diversion route for emergency vehicles, while FastAPI and React enable real-time monitoring and alert dissemination through a web-based platform.

Experimental evaluation demonstrated the effectiveness of the proposed framework in detecting accident-related anomalies with low processing latency and reliable performance under typical traffic conditions. By integrating accident detection and emergency response into a single workflow, the system addresses a key limitation of many existing approaches that focus solely on incident identification. The modular architecture also supports scalability and adaptation to different road networks and deployment environments.

Although the system achieved encouraging results, certain limitations remain. Performance may be affected under severe weather conditions and low-light environments, while the current route optimization module relies on a static road graph and does not incorporate live traffic information. Future work will focus on integrating real-time traffic feeds, enhancing low-visibility detection capabilities, and extending compatibility with smart city infrastructure. Overall, the proposed framework demonstrates the potential of combining computer vision and intelligent routing techniques to support faster and more efficient emergency response in modern transportation systems.

REFERENCES

[1] Ministry of Road Transport & Highways (MoRTH), "Road Accidents in India – 2022," Government of India, New Delhi, 2023.

[2] Redmon, J. & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. arXiv preprint arXiv:1804.02767.

[3] Wojke, N., Bewley, A., & Paulus, D. (2017). Simple Online and Realtime Tracking with a Deep Association Metric. In Proceedings of IEEE International Conference on Image Processing (ICIP), pp. 3645–3649.

[4] Zivkovic, Z. (2004). Improved Adaptive Gaussian Mixture Model for Background Subtraction. In Proceedings

of the 17th International Conference on Pattern Recognition (ICPR), pp. 28–31.

[5] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv preprint arXiv:2004.10934.

[6] Bewley, A., Ge, Z., Ott, L., Ramos, F., & Upcroft, B. (2016). Simple Online and Realtime Tracking. In Proceedings of IEEE International Conference on Image Processing (ICIP), pp. 3464–3468.

[7] Dalal, N. & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. In Proceedings of IEEE CVPR, Vol. 1, pp. 886–893.

[8] Cortes, C. & Vapnik, V. (1995). Support-Vector Networks. Machine Learning, 20(3), 273–297.

[9] Anwar, A. & Zaman, M. (2020). Accident Detection and Emergency Management Using Deep Learning. International Journal of Computer Science and Engineering, 8(3), 234–241.

[10] Singh, P., Sharma, R., & Gupta, A. (2021). Real-Time Road Accident Detection Using Surveillance Cameras and Convolutional Neural Networks. International Journal of Recent Technology and Engineering, 10(1), 155–160.

[11] Dhiman, G. & Kumar, V. (2019). Traffic Incident Detection Using Machine Learning: A Review. IEEE Access, 7, 112233–112250.

[12] Tan, M. & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning (ICML).

[13] Dijkstra, E. W. (1959). A Note on Two Problems in Connexion with Graphs. Numerische Mathematik, 1(1), 269–271.

[14] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. In Proceedings of IEEE CVPR, pp. 770–778.

[15] Reddi, S. J., Kale, S., & Kumar, S. (2018). On the Convergence of Adam and Beyond. In Proceedings of the 6th International Conference on Learning Representations (ICLR).

[16] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of Advances in Neural Information Processing Systems (NIPS), pp. 1097–1105.

[17] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE, 86(11), 2278–2324.

[18] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of IEEE CVPR, pp. 580–587.

[19] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149.

[20] Liu, W., et al. (2016). SSD: Single Shot MultiBox Detector. In Proceedings of European Conference on Computer Vision (ECCV), pp. 21–37.

[21] Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. Journal of Basic Engineering, 82(1), 35–45.

[22] Stauffer, C. & Grimson, W. E. L. (1999). Adaptive Background Mixture Models for Real-Time Tracking. In Proceedings of IEEE CVPR, Vol. 2, pp. 246–252.

[23] Farnebäck, G. (2003). Two-Frame Motion Estimation Based on Polynomial Expansion. In Proceedings of the 13th Scandinavian Conference on Image Analysis (SCIA), pp. 363–370.

[24] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of IEEE CVPR, pp. 7464–7475.

[25] Jocher, G., et al. (2023). Ultralytics YOLOv8. GitHub Repository.URL:https://github.com/ultralytics/ultralytics.

[26] Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools, 25(11), 120–125.

[27] Vaswani, A., et al. (2017). Attention Is All You Need. In Proceedings of Advances in Neural Information Processing Systems (NIPS), Vol. 30, pp. 5998–6008.

[28] Zhang, Y., et al. (2022). ByteTrack: Multi-Object Tracking by Associating Every Detection Box. In Proceedings of European Conference on Computer Vision (ECCV), pp. 1–21.

[29] Dosovitskiy, A., et al. (2021). An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale. In Proceedings of the 9th International Conference on Learning Representations (ICLR).

[30] National Crime Records Bureau (NCRB). (2022). Accidental Deaths and Suicides in India — 2022 Report. Ministry of Home Affairs, Government of India, New Delhi.