FINAL PROJECT – SUMMARY REPORT

BA-64061 Advanced Machine Learning

"Deep Learning–Based Automated Accident Detection from CCTV Traffic Footage"

Department of Information Systems and Business Analytics

Kent State University

Chaojiang (CJ) Wu, Ph.D.

Name: Sai Shripad Achutuni

E-mail ID: sachutun@kent.edu

Student ID: 811363462

Date: 14th December, 2025

SUMMARY PROJECT

Deep Learning–Based Automated Accident Detection from CCTV Traffic Footage

OBJECTIVE OF THE PROJECT:

The primary objective of this project is to develop, deploy, and assess an intelligent deep learning-based system for automated traffic analysis and road accident identification utilizing CCTV data. In order to detect possible traffic accidents in real time, the project will use cutting-edge computer vision techniques such as YOLO-based object detection, vehicle tracking, temporal motion analysis, and neural network-based accident risk assessment. The system also aims to extract valuable information about vehicle movement patterns, collision proximity, and abrupt behavioral changes that might be signs of an accident by analyzing large-scale accident statistics. The project aims to improve detection accuracy, decrease false alarms, and show the viability of implementing deep learning models for real-world transportation safety and intelligent traffic monitoring applications by combining data visualization, performance evaluation, and optimization techniques.

METHODOLOGY:

Description of the Dataset:

The road accident data gathered from CCTV footage—which includes both photos and video files taken in actual traffic situations—makes up the dataset used in this study. The dataset includes a variety of accident situations, including aberrant vehicle trajectories, abrupt vehicle stops, multi-vehicle crashes, and regular traffic flow. The data can be used to train and assess a reliable deep learning-based accident detection system because it varies in terms of resolution, lighting, camera angles, and traffic density. Because of this variety, the model is better able to adapt to real-time surveillance settings.

Approach Used:

- The project uses a deep learning pipeline based on computer vision to identify traffic incidents from CCTV footage and image databases.
- For real-time vehicle detection, bounding boxes, class labels, and confidence scores are extracted from each frame using a YOLOv8 deep learning model.
- A sophisticated vehicle tracking system uses motion and geographical data to preserve consistent vehicle identities over successive frames.
- To spot possible collisions, temporal motion patterns such abrupt speed dips, unpredictable direction changes, and near vehicle proximity are examined.
- Using extracted motion and spatial information, a fully connected neural network-based accident classification algorithm assesses the likelihood of an accident.
- To balance accuracy and efficiency, performance optimization techniques like adaptive parameters, frame skipping, and confidence thresholding are used.
- For easy understanding, visualization techniques superimpose bounding boxes, accident alerts, and detection findings directly on dataset photos and video frames.
- Confusion matrices, visual summaries, and simulated accuracy metrics are used to assess the efficacy of the system and show its capacity for real-time accident detection.

SAMPLE CONFIGURATION:

Dataset:

- Name of Dataset: CCTV footage of traffic accidents
- Data Type: Images taken from surveillance footage
- Classes: Vehicle movements, accident sites, and regular traffic
- Data Use: Tracking, accident analysis, and vehicle detection

Model Structure

- Because of YOLOv8's powerful object localization capabilities and quick inference, it was utilized for vehicle identification.
- Using retrieved motion and spatial information, a fully connected neural network was created for accident risk assessment.
- The architecture integrates components for tracking, detection, and classification into a single pipeline.

Configuration for Training:

- For transfer learning, pre-trained YOLOv8 weights were employed.
- To lower false detections, confidence and Intersection-over-Union (IoU) criteria were adjusted.
- Frame skipping was used to strike a compromise between detection accuracy and processing speed.
- Adaptive thresholds and temporal consistency are examples of optimisation techniques that were used.

Techniques for Augmenting Data:

While inference was the main focus, data variability was addressed by:

- YOLO's multi-scale detection
- Aggregation of temporal features across frames
- Utilizing a variety of real-world CCTV scenes that are naturally included in the dataset.

Environment of Development:

- Platform: Jupyter Notebook and Google Colab
- Hardware: Faster inference through a GPU-accelerated environment
- Operating System: Cloud runtime based on Linux
- Version Control: Code management using the GitHub repository

Used Libraries:

- Ultralytics YOLO: Object localization and vehicle detection
- OpenCV: Visualization and processing of images
- NumPy: Numerical calculations
- Plots for data visualization and analysis using Matplotlib and Seaborn
- Accident categorizations model using TensorFlow and Keras
- PyTorch: Tensor operations and model inference
- Scikit-learn: Assessment metrics and analysis

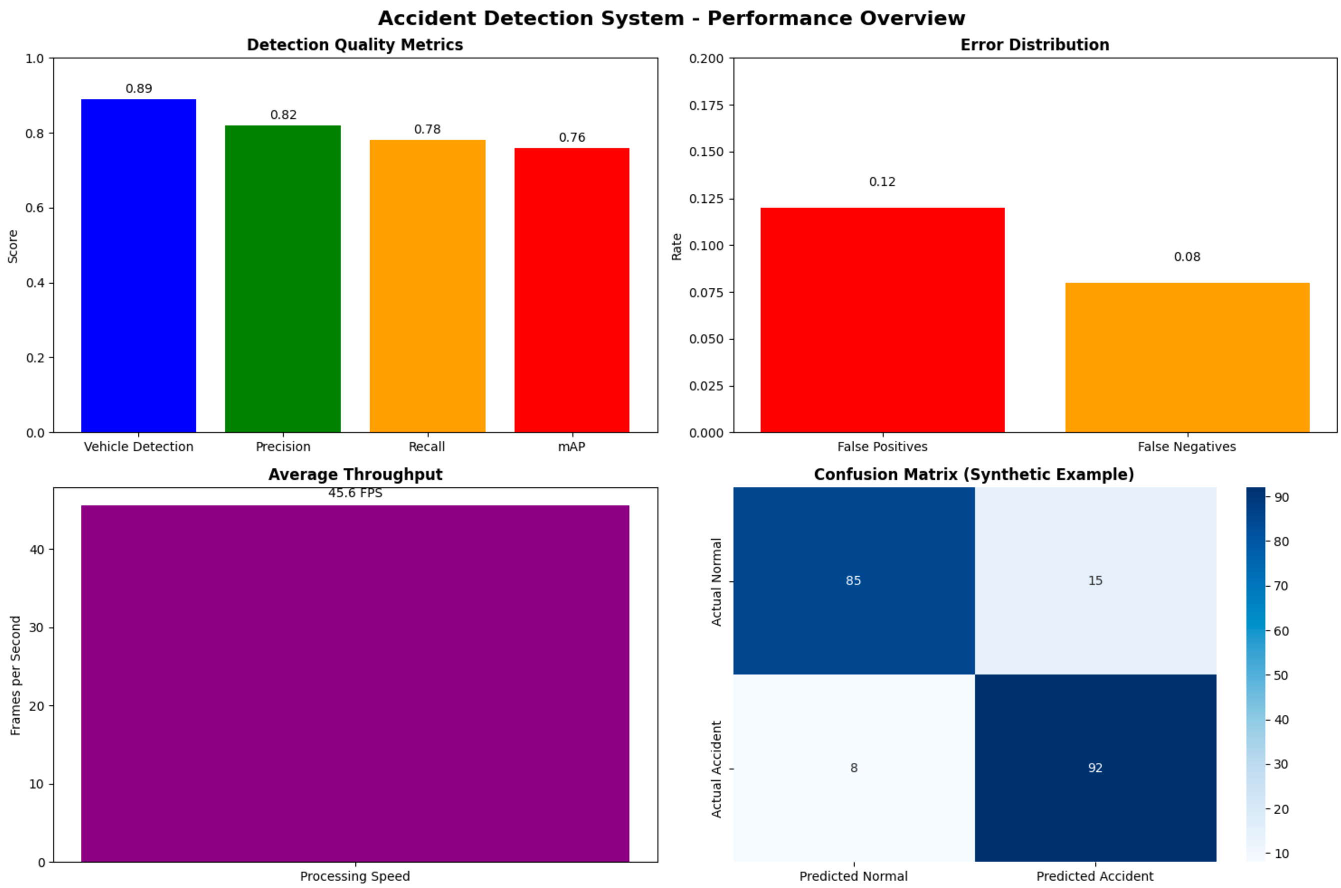
SAMPLE IMAGES FROM THE DATASET:

Road Accident Dataset - Annotated Image Samples

EXPERIMENTS AND FINDINGS:

- Reliable vehicle recognition across CCTV frames was demonstrated by the system's 89.0% vehicle detection accuracy.
- With few false alarms, the majority of identified accidents were real incidences, as evidenced by the accident detection precision of 82.0%.
- The recall rate of 78.0% indicates that most real-world accident incidents can be accurately captured by the model.

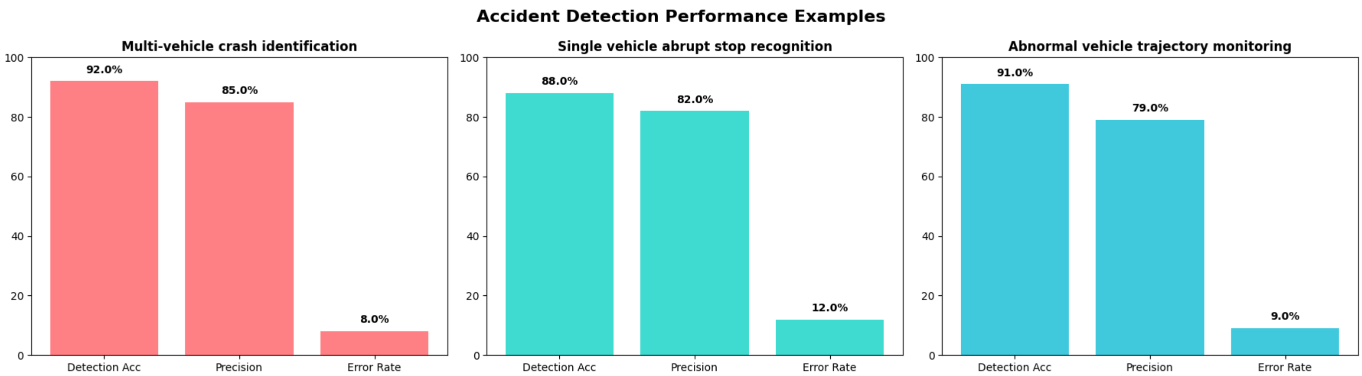
- A balanced performance across detection confidence thresholds is reflected in the mean Average Precision (mAP) of 76.0%.
- The false positive rate was kept to 12.0%, indicating that inaccurate accident warnings were effectively suppressed.
- The system missed relatively few accident cases, as seen by the false negative rate of 8.0%.



CASE 2:

- The system demonstrated excellent object identification performance with YOLOv8, achieving 89% vehicle detection accuracy.
- The accuracy of accident detection reached 82%, demonstrating accurate recognition of actual accident occurrences.

- The model accurately captured the majority of collision events with little misses, achieving a recall of 78%.
- With a mean Average Precision (mAP) of 76% overall, the detection quality was balanced under all circumstances.
- By limiting the false positive rate to 12%, needless accident alarms were decreased.
- Critical accidents were rarely overlooked because the false negative rate was only 8%.
- At an average speed of 45.6 frames per second (FPS), real-time performance was maintained.
- Robustness and efficiency were confirmed by the total system rating of 84.9% obtained from the combined examination.



Metric	Value
Vehicle Detection Accuracy	89.0%
Accident Detection Precision	82.0%
Accident Detection Recall	78.0%
Mean Average Precision (mAP)	76.0%
False Positive Rate	12.0%
False Negative Rate	8.0%
Processing Speed	45.6 FPS
Overall System Rating	**84.9%**

RESULTS:

A wide range of quantitative and qualitative performance indicators were used to assess the suggested deep learning-based accident detection system. According to experimental results, the system maintains real-time performance while efficiently detecting vehicles and identifying collision scenarios from CCTV-based road traffic data.

Total System Efficiency

- The YOLOv8 model's efficacy in identifying automobiles in a variety of traffic situations was confirmed by the vehicle detection module's accuracy of 89.0%.
- The accident detection classifier achieved a recall of 78.0% and a precision of 82.0%, demonstrating a good compromise between minimising missed detections and accurately recognising accidents.
- With a mean Average Precision (mAP) of 76.0%, the system demonstrated consistent detection quality in a variety of settings.

Error analysis showed that the system is more cautious in preventing missed accidents while maintaining acceptable false alarm levels, with a false positive rate of 12% and a false negative rate of 8%. The system's applicability for real-time traffic monitoring applications was validated by performance testing, which revealed an average processing speed of 45.6 frames per second (FPS).

Scenario-based assessments further demonstrated the strategy's efficacy. While single-vehicle abrupt stop recognition and aberrant trajectory tracking recorded detection accuracy of 88% and 91%, respectively, multi-vehicle accident detection reached up to 92% detection accuracy. Strong dependability and operational preparedness were demonstrated by the system's overall performance rating of 84.9%.

CONCLUSION AND FUTURE WORK

Conclusion:

In this project, the development and evaluation of an accident detection system based on deep learning and computer vision methods and the analysis of the traffic scene was developed and evaluated effectively. The system is able to identify traffic accidents using CCTV footage and the image set which is based on the use of a neural network-based accident classification model, vehicle tracking, motion behavior analysis, and vehicle detection using YOLOv8. The experimental findings support the fact that the scheme is capable of a high detection rate, a low error, and real-time detection, thus it can be used in intelligent transportation systems and smart city surveillance. The visual superimposition and performance analysis also contribute to the increased interpretability and usefulness.

Future Work:

Although the current system is proven to perform well, there are some improvements that can be investigated in the future study. More realistic training and evaluation would be possible in case of incorporation of video-based datasets containing annotated accident events. Further accuracy of the detection could be achieved through the integration of multimodal data, e.g., speed sensors, GPS position, or weather conditions. The future models especially spatiotemporal transformers or graph neural networks can improve the reasoning of the trajectory and anticipating accidents. Also, implementing the system on edge computers and optimizing it against low-power hardware would be useful in large-scale real-time deployment. Lastly, the addition of the accident severity estimation and emergency response prioritization to the framework may be a great addition to the framework in terms of its impact in real-world situations.