Real-Time Systems in Autonomous Vehicles: A Literature Review

Mustafa Can Çalışkan, 150200097

December 15, 2024

Contents

1 Introduction

To provide real-time safety and efficiency, autonomous vehicles (AVs) depend on AI-driven technology, so tackling important issues including transparency, computational restrictions, and adaptation. By means of explainable artificial intelligence (XAI) techniques including LIME and SHAP, deep learning models become more interpretable, hence strengthening confidence and real-time decision-making. While strong lane detection using YOLOv5 guarantees accuracy under dynamic situations, advanced control strategies—including Nonlinear Model Predictive Control (NMPC)—use hardware acceleration for real-time path tracking. Triple Iterative Approximate Dynamic Programming (TIADP) and other approaches maximize real-time responsiveness and energy economy, hence enhancing AV safety and adaptability. Emphasizing their part in allowing real-time, dependable autonomous systems, this paper emphasizes key developments.

2 Explainable AI (XAI) in Autonomous Vehicle Systems [1]

2.1 Introduction

Explainable artificial intelligence (XAI) has attracted a lot of interest in autonomous vehicle (AV) systems since it improves trustworthiness, openness, and safety in AI-driven decision-making procedures. Although AV systems mostly depend on deep learning models for tasks including vision, object detection, and decision-making, their black-box character sometimes makes understanding and interpretation of their outputs difficult. This restriction might erode general system dependability, user confidence, and regulatory approval.

Recently Tahir et al. [1] presented a fresh hybrid XAI solution intended especially for AV systems. Leveraging the characteristics of both approaches, the proposed framework combines Local Interpretable Model-agnostic Explanations (LIME) with Shapley Additive Explanations (SHAP) to attain real-time interpretability with lowest computing overhead. LIME presents fast, regional feature importance approximations; SHAP gives exact, worldwide explanations. Combining these techniques allows the hybrid approach to provide practical understanding of the AV decision-making process without sacrificing onboard performance.

The work provides transparency and real-time decision-making accuracy, therefore addressing the vital demand for interpretable artificial intelligence in AV systems. To strike computational efficiency and accuracy for onboard AV deployment, the authors suggest a hybrid XAI framework incorporating LIME and SHAP techniques.

2.2 Motivation and Problem Statement

The fast developments in AV systems have enhanced their perception and decision-making capacity; nevertheless, the lack of explainability in AI-driven decisions poses safety issues and legal questions. Current XAI methods either fail to estimate global feature importance (e.g., LIME) or lack computing efficiency (e.g., SHAP). This work seeks to close this difference by combining the advantages of both techniques.

2.3 Proposed Framework

The hybrid XAI solution consists of three main components:

1. **Perception and Decision-Making Module:** Combines multi-modal sensor fusion (e.g., images, LiDAR data) and deep learning models (ResNet-18, ResNet-50, SegNet) to detect objects and execute real-time decisions using Markov Decision Processes (MDP).

2. **Explanation Generation Module:** A hybrid LIME-SHAP model generates explanations by approximating feature importance using LIME and refining it with SHAP for top-ranked features.

3. **Integration Module:** The explanations are integrated into the decision-making pipeline and visualized for user interfaces, enhancing transparency.

The framework is implemented on an NVIDIA Jetson Nano with ARM architecture, utilizing Kubernetes for scalable computation and hyperparameter optimization via KATIB.

2.4 Experimental Setup

- **Dataset:** KITTI dataset (8,996 real-world images) and real AV-collected data (Melbourne, Australia).

- **Models:** ResNet-18, ResNet-50, SegNet.

- **Evaluation Metrics:** Intersection over Union (IoU), precision, recall, inference time, and memory usage.

- **Hardware:** NVIDIA Jetson Nano for real-time testing.

2.5 Results and Key Findings

- **Performance Metrics:** The proposed hybrid method achieved significant improvements:

  - IoU: ResNet-18 improved from 0.9459 to 0.9500.
  - Accuracy: ResNet-18 increased from 97.61% to 97.80%.
  - Inference Time: ResNet-18 reduced to 0.28s, SegNet reduced to 3.92s.

- **Computational Efficiency:** Hybrid LIME-SHAP reduced memory usage by up to 10MB for ResNet-18.

- **Interpretability:** Explanation fidelity surpassed 85% with consistency scores above 70%.

- **Adaptability:** The system dynamically adjusted explanations based on environmental complexity, improving decision-making reliability in urban, highway, and rural settings.

2.6 Conclusion

Explainable artificial intelligence for AV systems gains a major breakthrough from the hybrid LIME-SHAP framework. Combining local and global explanations guarantees both computing efficiency and good interpretability by which the technique is fit for real-time, safety-critical uses. Future research will concentrate on extending the framework to additional autonomous systems including drones and robotics as well as optimizing it for more complicated models such segmentation networks.

3 A Real-Time Path Tracking Approach [2]

3.1 Introduction

Targeting to lower traffic accidents and improve safety through enhanced control mechanisms, autonomous vehicle technology has become a major topic of research. A basic feature of autonomous driving, path tracking guarantees that cars follow a desired course precisely and fluidly. Conventions such pure pursuit, Stanley, PID, and LQR control have limits in managing the nonlinear dynamics and constraints of actual vehicles. Model Predictive Control (MPC) has attracted popularity in response to this since it can efficiently manage limitations and forecast future states. Conventional MPC methods, which depend on basic linear models, however, can degrade performance in highly nonlinear and dynamic settings.

3.2 Motivation and Problem Statement

The main focus of this work is on the large computing load of Nonlinear Model Predictive Control (NMPC), which restricts its real-time usability for path tracking in autonomous cars. Although NMPC suffers processing inefficiencies because of handling challenging nonlinear optimization issues, it offers a more realistic portrayal of vehicle dynamics than linear MPC. To tackle this,Xu et al. [2] introduces a new NMPC implementation integrating hardware and software parallelization approaches, so enabling real-time performance without sacrificing tracking accuracy.

3.3 Proposed Framework

The proposed framework includes the following key components:

- A "Nonlinear Vehicle Dynamics Model' based on a 3-DoF system that captures the coupling between longitudinal, lateral, and yaw motions.
- An "Integrated NMPC Controller" that outputs steering angles and longitudinal forces to minimize path tracking errors under actuator constraints.
- A "Parallel Newton Optimization Algorithm" to solve the NMPC problem efficiently by decoupling the recursive process into independent subproblems, enabling parallel execution.
- FPGA-based hardware acceleration using High-Level Synthesis (HLS) tools. Optimizations such as array partitioning, loop unrolling, and pipelining are applied to improve computational performance while balancing resource usage.

The framework achieves significant acceleration through hardware-software co-design, ensuring real-time path tracking.

3.4 Experimental Setup

The experimental validation is conducted using both simulations and real-time hardware-in-the-loop (HIL) experiments:

- **Simulation Environment:** MATLAB and CarSim are used to evaluate the NMPC controller under double-lane-change (DLC) and slalom maneuvers. Vehicle parameters, prediction horizon ($N = 18$), and constraints are carefully defined.
- **Hardware Platform:** A Xilinx Zynq FPGA platform (ARM Cortex-A9 and FPGA) is employed for real-time hardware implementation.
- **Optimization Strategies:** Three FPGA configurations are tested, combining array partitioning, loop unrolling, and pipelining techniques to achieve optimal performance.
- **Latency Measurement:** The communication latency between the FPGA controller and the CarSim environment is measured via UDP/Ethernet.

3.5 Results and Key Findings

The proposed NMPC controller demonstrates superior performance in both simulations and real-time experiments:

- Compared to Linear MPC (LMPC), the NMPC controller achieves better path tracking accuracy, especially under low road adhesion conditions ($\mu = 0.35$).
- The Parallel Newton Algorithm reduces computation time significantly compared to Sequential Quadratic Programming (SQP) and Interior Point methods.
- FPGA hardware acceleration achieves a solution time of 9.43 ms, meeting the real-time sampling period requirements, while consuming 51% of the available hardware resources.
- Real-time closed-loop experiments confirm stable and accurate trajectory tracking with minimal tracking errors for lateral position, heading angle, and longitudinal speed.

The FPGA implementation balances computational speed and resource utilization, ensuring practical deployment for real-time autonomous vehicle systems.

3.6 Conclusion

A real-time NMPC-based path tracking controller using FPGA hardware acceleration and a parallel Newton optimization method is presented in this work Achieving great accuracy and computing efficiency, the suggested approach efficiently manages the non-linear dynamics and restrictions of autonomous vehicles. Real-time experimental data confirm controller performance, hence lowering solution times to 9.43 ms.

4 A Lane Detection Technique for Autonomous Vehicles [3]

4.1 Introduction

Growing acceptance of autonomous cars emphasizes the need of accurate and real-time lane detecting systems to guarantee dependability and safety in transportation. This work presents a YOLOv5 Segmentation Large Model based real-time lane detecting system. Combining cutting-edge computer vision and deep learning methods, the approach greatly increases lane detecting accuracy and processing speed. The work of Swain and Tripathy [3] uses ResNet-50 as a backbone for feature extraction and produces strong results under several environmental situations.

4.2 Motivation and Problem Statement

Traditional lane recognition techniques struggle in unstructured situations, changing weather, and real-time performance. Many times, present technologies lack the resilience needed for implementation in autonomous cars. Driven by these constraints, this work addresses using advanced segmentation models improving detection accuracy, resilience, and real-time efficiency. By Swain and Tripathy [3], the suggested method addresses problems including lighting variability, noise, and missing lane markers by using pre-processing techniques and YOLOv5.

4.3 Proposed Framework

The framework employs the YOLOv5 Segmentation Large Model with ResNet-50 for feature extraction. Key components include:

- **Data Collection and Annotation:** A diverse dataset of American lane videos was meticulously annotated using the LabelMe tool.
- **Pre-processing:** Techniques such as resizing, augmentation, noise addition, grayscale conversion, and random rotations were applied to enhance dataset robustness.
- **Model Configuration:** The model integrates ResNet-50 with YOLOv5 architecture, using loss functions like Dice Loss and Cross-Entropy Loss to optimize segmentation accuracy.
- **Training and Fine-Tuning:** Transfer learning from pre-trained COCO weights was used, followed by hyperparameter optimization.

The model they validated using k-fold cross-validation to ensure generalizability.

4.4 Experimental Setup

The experimental setup involved:

- **Dataset:** Annotated videos were processed into frames and split into training (80%), validation (20%), and test sets.

- **Hardware:** NVIDIA GeForce RTX 3050 GPU was used to achieve real-time processing.
- **Metrics:** Performance was evaluated using pixel accuracy, mean Average Precision (mAP), precision-recall curves, F1-score, and Intersection over Union (IoU).

The YOLOv5-ResNet50 model was compared against Mask-RCNN and YOLOv5 with VGG-16 backbone.

4.5 Results and Key Findings

The proposed YOLOv5-ResNet50 model outperformed existing methods with the following key results:

- **Pixel Accuracy:** 89% accuracy.
- **Processing Speed:** Achieved 48 frames per second (FPS), ensuring real-time applicability.
- **mAP:** 0.885 for bounding box detection and 0.731 for mask image segmentation at IoU=0.5.
- **Precision-Recall Curves:** The model demonstrated high precision and recall, even at varying confidence thresholds.

The comparison highlighted that YOLOv5-ResNet50 achieves a superior balance of accuracy and efficiency, making it robust across diverse road conditions.

4.6 Conclusion

This work presented a strong real-time lane detecting system leveraging YOLOv5 Segmentation Large Model with ResNet-50 for autonomous vehicles. High accuracy, mAP ratings, and real-time processing speed were attained by the suggested model. Its adaptability for autonomous driving systems is shown by its consistent performance under many circumstances.

5 Minimizing Collision Impact in Autonomous Vehicles [3]

5.1 Introduction

With 94% of road accidents accounted for, autonomous vehicles (AVs) have become a potential option to lower human errors and traffic deaths [4]. AVs cannot totally prevent accidents despite their developments because of environmental factors and physical constraints. Bosia et al. [4] shows a hybrid technique that is developed combining real-time simulations utilizing a multibody dynamic (MBD) program with a precomputed database of crash events to identify the safest trajectory that reduces occupant injuries.

5.2 Motivation and Problem Statement

The certainty of some mishaps calls for investigation on best safety precautions during collisions. Current research mostly rely on finite element methods (FEM), either of which have limited applicability or computational inefficiencies. Machine learning models are the main focus here. Inspired by these difficulties, Bosia et al. [4] developed a solution to minimize damage severity in inevitable collisions. The paper also emphasizes the moral, legal, and computational difficulties connected with autonomous cars.

5.3 Proposed Framework

The framework combines two methods for evaluating crash severity:

- **Real-Time Simulations:** The MBD software performs on-the-fly crash simulations to determine severity indices like THIV, OIV, and ASI.
- **Precomputed Database:** A database of severity indices is generated using FEM results and enhanced MBD simulations, enabling near-instantaneous decision-making for standard accident scenarios.

By use of contact calculations, filter implementation, and targeted collision region emphasis, the system minimizes computing delays. Severity indices offer a strong criterion for choosing the course of least damage.

5.4 Experimental Setup

Two main experimental scenarios were designed:

1. **Car-to-Car Collisions:** Simulations were performed for varying angles (0-180°) and speeds (30-75 km/h).
2. **Vehicle-to-Barrier Collisions:** Tests involved impacts with New Jersey barriers at angles (15-40°) and speeds (30-105 km/h).

The MBD software's results were compared against FEM (Ls-Dyna) benchmarks. A case study simulated real-life dilemma situations, where the AV selected the least injurious trajectory among five possible options.

5.5 Results and Key Findings

- **Real-Time Simulations:** MBD software provided promising results with a computational time of 80 ms, well within the allowable 0.3 seconds for decision-making.
- **Severity Index Correlation:** THIV showed the highest correlation between MBD and FEM results, ensuring consistency in trajectory ranking.
- **Case Study:** In over 90% of cases, the MBD software successfully identified one of the three least injurious trajectories. Using the precomputed database further improved accuracy and reduced discrepancies.
- **Performance for Barriers:** Collisions involving New Jersey barriers demonstrated near-perfect alignment between MBD and FEM simulations.

5.6 Conclusion

This work shows that real-time crash severity assessments in autonomous cars is feasible with MBD tools. High accuracy and computational efficiency are obtained by the suggested hybrid architecture, which combines real-time simulation with a precomputed database. The findings imply great possibility to reduce the degree of damage in inevitable collisions.

6 A Real-Time Tracking Control Strategy [5]

6.1 Introduction

This work proposes a robust tracking control approach for autonomous cars employing a Triple Iterative Approximate Dynamic Programming (TIADP) paradigm. While minimizing outside disturbances, the method guarantees real-time responsiveness, strong tracking performance, and energy economy. By the work of Li et al. [5], the framework blends value function iteration, control policy iteration, and disturbance policy iteration inside a deep neural network structure, therefore guaranteeing computational efficiency and resilience in the face of dynamic uncertainty.

6.2 Motivation and Problem Statement

Because of complicated, nonlinear dynamics and external disturbances including road conditions and environmental noise, exact tracking control for autonomous cars is difficult. Concerning real-time performance and disturbance reduction, conventional control techniques including PID and MPC have shortcomings.The work Li et al. [5] build, defining the tracking control as a zero-sum differential game where control strategies minimize mistakes while disturbance policies maximize them. The answer seeks to balance computing efficiency, energy economy, and trajectory tracking accuracy.

6.3 Proposed Framework

The proposed *Triple Iterative ADP* framework optimizes:

- **Value Function:** Evaluates the overall performance of the control and disturbance policies.
- **Control Policy:** Minimizes trajectory tracking error and energy consumption.
- **Disturbance Policy:** Maximizes the disturbance impact for robustness validation.

Iteratively through value function updates, control policy refinement, and disturbance policy optimization, the TIADP algorithm performs via parameterizing the value function and policies, neural networks guarantee convergence via strict mathematical proofs and enable speedier real-time solutions.

6.4 Experimental Setup

Experiments include both numerical simulations and real-world validation using a micro-vehicle platform. The experiments test the proposed controller under:

- Double-lane change scenarios.
- Serpentine trajectory tracking.
- Artificially induced lateral disturbances.

PyTorch's hyperparameters tuned for convergence help the neural networks to be trained. Against Model Predictive Control (MPC) and conventional ADP, performance measures including trajectory inaccuracy, control input magnitude, and computation time are evaluated

6.5 Results and Key Findings

The experimental results show the suggested framework's efficiency:

- With a single-step computation time of 1.44 ms on a Raspberry Pi, TIADP achieves faster processing times—four times improvement over MPC.
- With errors within reasonable limits, the controller offers strong tracking performance under disturbance.
- Minimizing energy usage helps to balance tracking accuracy and control effort with more traditional techniques.

Numerical simulations and micro-vehicle platform testing both support TIADP's reliability, economy, and pragmatic applicability in actual autonomous driving scenarios.

6.6 Conclusion

A new "Triple Iterative Approximate Dynamic Programming" framework for effective and resilient autonomous vehicle tracking control is presented in this work. The proposed approach uses deep neural networks to ensure real-time performance, effectively controls outside disturbances, and best uses of energy usage. Experimental validations clearly indicate advantages over more conventional methods like MPC, so TIADP offers considerable potential for pragmatic applications.

7 Comparative Analysis of Reviewed Techniques

This section compares the methods under literature search in the research in respect to real-time applications, processing efficiency, accuracy, and adaptability in autonomous vehicle (AV) systems. The focus of the comparison is important components including lane detecting, collision mitigation, tracking control, model predictive control (MPC), and explainable artificial intelligence (XAI).

7.1 Real-Time Performance

Table 1 summarizes the computational efficiency and execution times reached by the researched methodologies since AV systems must guarantee fast decision-making and control by means of real-time performance.

Table 1: Real-Time Performance Comparison of Reviewed Techniques

Technique	Execution Time	Hardware Used	Efficiency Enhancements	Real-Time Suitability
XAI Framework	0.28 s (ResNet-18)	NVIDIA Jetson Nano	Hybrid LIME-SHAP, reduced memory overhead	High
NMPC Path Tracking	9.43 ms	Xilinx Zynq FPGA	Parallel Newton Algorithm, FPGA acceleration	High
YOLOv5 Lane Detection	48 FPS	NVIDIA RTX 3050 GPU	ResNet-50 backbone, optimized pre-processing	Very High
Collision Mitigation (MBD)	80 ms	General CPU (MBD Software)	Precomputed database, real-time filters	High
TIADP Tracking Control	1.44 ms	Raspberry Pi	Triple Iterative ADP, optimized NN training	Very High

High real-time applicability of the NMPC route tracking framework obtaining a solution time of 9.43 ms [2] using FPGA-based parallel processing was demonstrated. The lightweight neural network-based control of the TIADP framework acquired remarkable performance with a 1.44 ms [5] single-step computation time. Attaching real-time frame rates of 48 FPS [3], the YOLOv5-based lane identification technology also aligned with fast-paced AV applications.

7.2 Computational Efficiency

Computing efficiency of every technique is assessed in respect to hardware usage, memory consumption, and optimization strategies:

- With hybrid LIME-SHAP models, the **XAI framework** effectively reduced memory utilization by up to **10 MB** [1] without compromising speed.
- Using FPGA acceleration methods including pipelining and array partitioning, **NMPC path tracking** achieves great computational efficiency with **51% hardware resource** [2] use.
- Precomputed databases were used for collision mitigation to avoid intricate real-time simulations and hence lower computational delays to **80 ms** [4].
- Optimized neural network-based value and control policy iterations to accomplish **4x faster computation** compared to conventional MPC, thereby addressing TIADP tracking control [5].

7.3 Accuracy and Robustness

Real-world AV systems rely on dependability against environmental changes and correctness. Table 2 shows the achieved accuracy values among the several techniques.

Table 2: Accuracy Comparison of Reviewed Techniques

Technique	Accuracy Metric	Robustness	Key Result
XAI Framework	97.80% (ResNet-18 Accuracy)	Dynamic urban/rural adaptability	Explanation fidelity > 85%
NMPC Path Tracking	Path error minimized	Robust under $\mu = 0.35$ conditions	Improved lateral tracking accuracy
YOLOv5 Lane Detection	89% Pixel Accuracy	Robust to weather/noise variations	mAP=0.885, 48 FPS
Collision Mitigation	THIV & ASI correlations	Consistent for barrier/car crashes	90% accurate trajectory selection
TIADP Tracking Control	Minimal tracking error	Resilient to disturbances	**1.44 ms** real-time response

Achieving 89% [3] pixel precision, the YOLOv5-based lane detection system proved durable against changing lighting and weather. Under external disturbances, the NMPC controller effectively lowered route tracking errors while the TIADP framework guaranteed perfect tracking even under external disturbances.

7.4 Adaptability to Dynamic Environments

AV dependability and safety depend on adaptability to dynamic and uncertain surroundings. High adaptability techniques include XAI, NMPC, and TIADP shown:

- **XAI Framework:** Dynamically changed explanations guaranteed by environmental complexity, so guaranteeing openness.
- **NMPC Path Tracking:** Oversaw nonlinear dynamics and actuator limitations to guarantee consistent performance over difficult motions.
- **TIADP Control:** Reduced outside noise guarantees consistent and energy-efficient trajectory tracking.
- **YOLOv5 Lane Detection:** Maintained accuracy under unstructured environments and unorganized surroundings.

7.5 Summary

The examined methods solve important issues like computational economy, accuracy, and flexibility, so offering major improvements in real-time AV systems. Table 3 offers a high-level overview.

Table 3: Summary of Reviewed Techniques

Technique	**Real-Time**	**Accuracy**	**Efficiency**	**Adaptability**
XAI Framework	High	Very High	High	High
NMPC Path Tracking	High	High	Very High	High
YOLOv5 Lane Detection	Very High	High	High	Very High
Collision Mitigation	High	High	High	Moderate
TIADP Tracking Control	Very High	Very High	Very High	High

The results show that real-time route tracking employing NMPC and tracking control via TIADP shine in computational efficiency and real-time responsiveness. Dynamic AV setups would find the XAI and YOLOv5 frameworks fit since they provide excellent accuracy and adaptability. The key targets of further research should be improving hardware efficiency, lightweight deployment, and integration of different frameworks for end-to- end AV systems.

8 Challenges and Future Directions

Development of real-time autonomous vehicle systems still presents various difficulties although major progress. Ensuring the general acceptance, safety, and scalability of AV systems depends on addressing these obstacles.

8.1 Computational Complexity

Real-time performance calls both effective hardware optimization and algorithms. Often requiring large processing resources, techniques as NMPC and XAI limit adoption on edge devices with limited capabilities. Future projects should concentrate on:

- Creating light-weight neural network models and optimisation techniques.
- Edge-based computation using hardware accelerators such FPGAs, GPUs, and TPUs.
- Using quantizing, pruning, and sparsity methods to lower inference time and model size.

8.2 Robustness in Unstructured Environments

AV systems have to run consistently under a range of environmental factors including unstructured roads, sensor noise, and changing weather. Dealing with robustness requires:

- Improving sensor fusion methods that combine cameras, LiDAR, radar, and multi-modal data.
- Developing machine learning models to manage low-visibility situations, insufficient data, and occlusions.
- Building varied and large datasets including model training's edge-case situations.

8.3 Scalability and Deployment

Real-world application of AV systems still depends much on scalability. Real-time systems have to strike equilibrium between hardware restrictions, latency, and precision. Future paths call for:

- Creating scalable systems capable of adjusting to several hardware setups and vehicle platforms.
- Optimizing communication protocols for vehicle-to-everything (V2X) interactions to enable collaborative decision-making.
- Ensuring low-latency communication in edge-cloud architectures for distributed processing.

8.4 Explainability and Trustworthiness

Regulatory approval and user acceptance depend critically on AV decision-making procedures being transparent and interpretable. Problems consist in:

- Enhancing XAI approaches' real-time performance and correctness.
- Developing domain-specific XAI models tailored for AV tasks, such as collision avoidance and path tracking.
- Integrating user-friendly visualization tools to communicate AV decisions effectively.

8.5 Energy Efficiency

Energy consumption is a primary limitation for AV systems especially for electric cars. Future research has to aim to:

- Using energy economy, design control strategies to optimize power consumption during path tracking and navigation.
- Research low-power hardware choices for quick calculations.
- Mix advanced energy management systems with renewable energy sources for ecologically responsible AV operations.

8.6 Ethical and Legal Considerations

Particularly in accident scenarios and decision-making, the use of AVs begs moral dilemmas and legal problems. overcoming these challenges demands:

- Developing moral rules for AV selection under extreme conditions.
- Ensuring compliance with evolving safety regulations and standards.
- Promoting collaboration between policymakers, researchers, and industry stakeholders to establish transparent guidelines.

9 Conclusion

Especially in explainability, decision-making, and control strategies, this review of the literature highlights how crucial artificial intelligence is in developing real-time systems for autonomous cars. Like the hybrid LIME-SHAP method, explainable artificial intelligence models improve openness and trust, therefore guaranteeing regulatory compliance and user confidence. By addressing nonlinear dynamics and disturbances, AI-powered approaches include nonlinear model predictive control (NMPC) and triple iterative ADP (TIADP) provide real-time, resilient path tracking.

Under different settings, AI-driven computer vision models including YOLOv5 with ResNet-50 attain great accuracy and real-time performance in lane detection. To similarly minimise crash severity, AI-assisted collision avoidance techniques mix simulations with precomputed data.

Even if artificial intelligence has greatly raised real-time responsiveness, accuracy, and robustness, problems including computing complexity, energy efficiency, and scalability still exist. Future research should mostly focus on lightweight, effective AI models and hardware optimization if people want to assure wide implementation. All things considered, AI-driven real-time systems create the route for smart, flexible, safer autonomous driving solutions.

References

[1] H. A. Tahir, W. Alayed, W. U. Hassan, and A. Haider, “A novel hybrid XAI solution for autonomous vehicles: Real-Time interpretability through LIME–SHAP integration,” Sensors, vol. 24, no. 21, p. 6776, Oct. 2024, doi: 10.3390/s24216776.

[2] F. Xu, X. Zhang, H. Chen, Y. Hu, P. Wang, and T. Qu, “Parallel Nonlinear Model Predictive Controller for Real-Time Path Tracking of Autonomous Vehicle,” IEEE Transactions on Industrial Electronics, vol. 71, no. 12, pp. 16503–16513, May 2024, doi: 10.1109/tie.2024.3390738.

[3] S. Swain and A. K. Tripathy, “Real-time lane detection for autonomous vehicles using YOLOV5 Segmentation Model,” International Journal of Sustainable Engineering, vol. 17, no. 1, pp. 1–11, Sep. 2024, doi: 10.1080/19397038.2024.2400965.

[4] L. Bosia, L. Manganotto, M. Anghileri, S. Dolci, P. Astori, and C.-D. Kan, “Real-Time Collision Mitigation Strategies for Autonomous Vehicles,” IEEE Transactions on Intelligent Transportation Systems, pp. 1–11, Jan. 2024, doi: 10.1109/tits.2024.3400680.

[5] W. Li et al., “Real-Time Resilient Tracking Control for Autonomous Vehicles Through Triple Iterative Approximate Dynamic Programming,” IEEE Transactions on Intelligent Transportation Systems, pp. 1–14, Jan. 2024, doi: 10.1109/tits.2024.3489019.