

DESIGN AND DEVELOPMENT OF ADAPTIVE FRONT LIGHTS IN AUTOMOBILE USING DEEP LEARNING ALGORITHMS

Thesis Submitted for the Award of the Degree of

DOCTOR OF PHILOSOPHY

in

Electronics and Communication Engineering

By

Glenson Toney

Registration Number: 42000326

Supervised By

Dr Gaurav Sethi (11106)

Professor & Additional Dean,

School of Electronics and Electrical

Engineering,

Lovely Professional University, Punjab

Co-Supervised By

Dr Cherry Bhargava

Senior Manager,

Eaton India

Innovation Center

Maharashtra



LOVELY PROFESSIONAL UNIVERSITY, PUNJAB

2025

DECLARATION

I, hereby declared that the presented work in the thesis entitled “Design and Development of Adaptive Front Lights in Automobile using Deep Learning Algorithms” in fulfilment of degree of **Doctor of Philosophy (Ph. D.)** is outcome of research work carried out by me under the supervision of Dr. Gaurav Sethi, working as Professor & Additional Dean in the School of Electronics and Electrical Engineering of Lovely Professional University, Punjab, India and Dr. Cherry Bhargava, working as Senior Engineer, Eaton India Innovation Center, Maharashtra, India. In keeping with general practice of reporting scientific observations, due acknowledgements have been made whenever work described here has been based on findings of other investigator. This work has not been submitted in part or full to any other University or Institute for the award of any degree.

(Signature of Scholar)

Glenison Toney
Registration No.: 42000326
School of Electronics and Electrical Engineering
Lovely Professional University,
Punjab, India

CERTIFICATE

This is to certify that the work reported in the Ph. D. thesis entitled “Design and Development of Adaptive Front Lights in Automobile using Deep Learning Algorithms” submitted in fulfillment of the requirement for the award of degree of **Doctor of Philosophy (Ph.D.)** in the School of Electronics and Electrical Engineering, is a research work carried out by Glenison Toney, 42000326, is bonafide record of his original work carried out under my supervision and that no part of thesis has been submitted for any other degree, diploma or equivalent course.

(Signature of Supervisor)

Dr. Gaurav Sethi
Professor & Additional Dean,
School of Electronics and Electrical
Engineering,
Lovely Professional University



(Signature of Co-Supervisor)

Dr Cherry Bhargava
Senior Manager,
Eaton India Innovation Center
Maharashtra

ABSTRACT

Nighttime driving presents critical safety challenges due to limited visibility and glare from oncoming vehicles. Conventional fixed headlamps often fail to illuminate curving roads adequately and can blind approaching drivers, inducing temporary visual impairment (Troxler effect) that slows reaction times and contributes to accidents. To address these issues, this research develops an integrated adaptive headlamp system that dynamically adjusts beam direction and intensity in real time. The work spans vehicle dynamics modeling, control systems, and computer vision to enhance nighttime driving safety. The thesis contributes a holistic framework combining predictive vehicle control with intelligent perception, and is organized around four core objectives: (1) designing a vehicle dynamics-based predictive headlamp controller for illuminating road curves, (2) creating a perception algorithm for low-light object and traffic detection to enable adaptive dimming, (3) fusing the predictive control and perception modules and controller into a unified intelligent headlamp, and (4) validating the system's performance.

The first objective is to develop a mathematical model-based controller that can steer headlamp beams proactively when the vehicle traverses a curve. Traditional adaptive lighting systems typically rely on steering wheel angle alone, which may not accurately represent the vehicle's actual path during high-speed maneuvers or skids. The current research formulates a high-fidelity vehicle dynamics model with longitudinal, lateral, and yaw dynamics and tire force characteristics, aerodynamics, and braking. A key outcome of this modeling is the accurate determination of the vehicle's slip angle—the difference between its direction and direction of travel. The slip angle forecasts the car's actual path through a turn. The technology can align the headlamps with the true path of the vehicle and not merely the steering input by using the slip angle, so ensuring that the beams light the road where the car is truly going even if tire slip or curvature cause deviations from the steering direction.

Based on this, a predictive beam-steering controller is designed. Initially, an elementary proportional control method is implemented: the desired slip angle computes the headlight deflection angle. The simple Proportional Controller illustrates the concept of path-aware beam steering by directing the beam to follow the vehicle's path. But cars have rapid steering changes and actuator response times that could cause the beams to jitter or

overshoot. A Filtered Proportional Controller is employed to account for these dynamics by introducing a filtering element to dampen rapid changes. The filtered controller creates a smooth transition of a reliable beam during rapid motion by anticipating the actuator delay and compensating the beam motion accordingly. This is necessary for driver comfort and consistent road lighting because it prevents sudden headlight jerks and eliminates light beam oscillations.

Under various driving conditions, the dynamic model and its controllers were simulated carefully. The controller performance was verified using various synthetic road profiles of different curve radii and vehicle speeds. It is observed that on curves, the predictive control system based on slip angle gives steady and reactive headlamp alignment. Unlike a static headlamp, the beam greatly increases lighting at bends by following the road curve. Even during abrupt steering inputs or when the vehicle experiences understeer/oversteer, the controller keeps the beams focused on the roadway ahead, thanks to the slip-angle feedback. This demonstrates the feasibility of a model-based approach to proactive beam steering and establishes a baseline for further enhancements. The Fused Controller is realized by merging advanced control algorithms with vehicle trajectory estimation to manage headlamp direction. The Fused Controller uses filtered feedforward and extended model predictive control (FF-MPC and E-MPC) strategies to estimate road heading from the dynamic model and adjust beam orientation proactively, ensuring alignment with the vehicle's actual path through corners or maneuvers.

The second objective targets the development of an intelligent perception module to detect on-road objects and oncoming traffic under low-light conditions, enabling the headlamp system to adapt its beam intensity and avoid glaring other drivers. Nighttime environments challenge conventional vision algorithms due to poor lighting and small, low-contrast hazards (e.g., distant vehicle lights, pedestrians in dark clothing). To overcome these challenges, this thesis investigates state-of-the-art deep learning detectors and tailors them for nighttime use. Initially, existing object detection frameworks such as Single Shot Detector (SSD), Faster R-CNN, and the YOLO family (including Tiny-YOLO for efficiency) are evaluated on nighttime driving scenarios. This analysis reveals limitations: while modern detectors are effective in daytime or well-lit conditions, their performance degrades with low visibility, often missing small or dim objects or producing false detections under headlight glare and noise. These findings underscored the need for a

specialized detection approach attuned to night driving. Also a modified SSD and multi-faceted object detection model was proposed to meet these challenges. However, varying light conditions and small object detections were an issue that left unaddressed.

To address this requirement, a new deep learning model named LSDNet (Low-Light and Small-object Detection Network) is developed as the perception module's central component. LSDNet is specifically designed for the robustness of low-illumination conditions as well as for detection of small, poorly visible road objects. LSDNet uses a state-of-the-art convolutional detection backbone optimized for the accuracy-speed trade-off; real-time operation is essential for a live headlamp system. A few stages of the network bring about novel enhancements. Preprocessing and data augmentation are first used to simulate nighttime – such as using dynamic contrast enhancement to accommodate changing lighting and simulating glare or noise to make the model robust.

Second, LSDNet ensures the recognizability of objects of differing sizes—particularly smaller ones such as distant taillights or wildlife on the road—by employing a multi-scale feature extraction process. Context-sensitive layers based on surrounding scene data—e.g., the presence of a road or lane markers—augment this multi-scale approach by separating items from background noise in the dark. Third, an inference-level adaptation method is employed: LSDNet adjusts its confidence levels of detection dynamically based on input-estimated ambient illumination. The network is more sensitive (lower thresholds) in order not to miss dim objects under very dark conditions; in more lighted ones it can be more discriminative to avoid false alarms. Also, a light contrast correction and enhancement module is included in the pipeline such that low-contrast details (such as a pedestrian just outside of the reach of headlights) are enhanced for the detector.

LSDNet's performance is benchmarked against the existing detectors. The custom model demonstrates significantly improved recall and precision for low-light object detection. It reliably identifies oncoming vehicles, pedestrians and roadside obstacles with higher accuracy. Importantly, LSDNet maintains high performance on small objects, a common failure case for standard detectors at night. Moreover, LSDNet achieves this improved detection quality while operating with low latency, meeting real-time requirements. On a moderate GPU platform, the network processes frames at a high frame rate, meaning it can be deployed in an embedded automotive computer without causing delays in response. This perception capability directly feeds into the headlamp control logic: with accurate detection

of oncoming vehicles, the system can automatically dip the high beams in time to prevent dazzling the other driver, and with recognition of pedestrians or obstacles ahead, it can keep beams elevated or focused to ensure those hazards are well lit for the driver. Thus, Objective 2 yields a perception-aware vision system that is cognizant of the challenging lighting conditions of night.

In the third objective the LSDNet model that performs real-time object detection optimized for low-light and small-scale hazards and the Fused controller is integrated in the adaptive headlamp. When objects are detected within the headlamp's field of view, LSDNet triggers an intensity adjustment mechanism, enabling smooth dimming (dip) of the beam to reduce glare or improve hazard visibility. This beam intensity modulation works in parallel to the directional control managed by the Fused Controller. Both systems are independently controlled yet tightly integrated, such that the vehicle simultaneously receives inputs for beam angle (from the Fused Controller) and beam intensity (from LSDNet). The coordination of these modules within a single embedded platform results in a cohesive adaptive headlamp system that responds intelligently to both vehicle dynamics and the external environment.

The final objective is to rigorously evaluate the performance of the proposed adaptive headlamp system in terms of efficiency, latency, and overall effectiveness under representative driving conditions. Extensive testing was conducted, including controlled simulations and real-world scenario emulations, to measure how well the system meets safety and responsiveness criteria compared to existing solutions. A 2-kilometer virtual test track was designed with a mix of straight segments, various curvature turns (gentle curves to sharp bends), and a range of traffic conditions such as oncoming vehicles and roadside objects. This provided a comprehensive proving ground for both the beam-steering control and the object detection under low light. The results from simulation tests are promising. The advanced controllers (FF-MPC and E-MPC) introduced for predictive beam steering showed marked improvement in beam alignment accuracy and responsiveness over the conventional approach. Specifically, when the car negotiated curves, the model predictive controllers kept the beam focused on the road centerline far better than a simple steering-linked system, especially at higher speeds or when quick adjustments were needed. The headlights under MPC control settled to the correct angle with minimal overshoot, whereas the conventional system tended to either lag behind the turn or overshoot slightly,

illuminating the wrong part of the road for brief periods. This translates to more consistent visibility for the driver. In terms of response speed, the predictive controllers were able to start reorienting the beams slightly before the vehicle fully entered a curve due to the feedforward aspect. The Fused Controller further enhanced performance by handling unforeseen or dynamic elements on the road.. The latency of the entire loop remained below the typical reaction time for driving scenarios. This means the system’s actions feel instantaneous from the driver’s perspective, maintaining safety

Quantitatively, the object detection module LSDNet achieved a high mean Average Precision (mAP) on the test scenarios, outperforming the compared detectors in correctly identifying vehicles and hazards under varying low-light conditions. It also sustained real-time inference speeds (measured in frames per second) on the embedded hardware, validating that the deep learning model can run onboard without causing delays. There was no significant drop in detection accuracy at higher frame rates, indicating the model is both accurate and efficient. Through rigorous comparative analysis, the integrated system was shown to outperform existing adaptive lighting setups on all examined criteria: curve lighting quality, reaction to oncoming vehicles, detection of roadside objects, system response time, and overall reliability.

The adaptive headlamp framework not only proves the concept of combining predictive control with perception, but also demonstrates measurable safety and performance benefits. By keeping the full control-perception loop latency low and the algorithms efficient, the research shows that such a system is practical for real-world deployment. Drivers would experience improved visibility of the road and potential hazards at night, while also causing less discomfort to others on the road – a dual benefit that directly addresses the main problems of night driving safety and the Troxler effect.

In summary, this work delivers a novel, comprehensive solution to enhance automotive headlighting. The research achieved its four primary objectives by developing: (1) a slip angle-based predictive headlamp control that illuminates curves in alignment with the vehicle’s true trajectory, (2) a deep learning vision model (LSDNet) tailored for low-light object detection to inform beam dimming and aiming, (3) an integrated Fused Controller combining these predictive and perceptive elements for context-aware beam control, and (4) thorough validation proving superior performance to state-of-the-art adaptive headlights. The contributions to knowledge include the introduction of a dynamic vehicle

model into headlamp control (improving upon purely kinematic or steering-linked approaches), the creation of a specialized object detector for nighttime driving conditions, and the demonstration of an integrated control system that marries control theory with artificial intelligence in the automotive lighting domain. This holistic approach to adaptive headlamps represents a significant step forward in automotive safety technology. The outcomes of this thesis lay a strong foundation for intelligent headlamp systems that proactively respond to both the driver's path and the environment, greatly improving visibility and reducing accident risk during night drives. The framework and techniques developed can be extended and scaled in future vehicles, indicating a clear path toward smarter, safer illumination systems on our roads.

Acknowledgements

*First and foremost, I bow in gratitude to the **Lord Almighty**, whose grace, guidance, and blessings throughout this journey. Without the divine will, this thesis would not have been possible.*

*I extend my deepest gratitude to my supervisor, **Dr. Gaurav Sethi**, for his unwavering support, insightful critiques, and constant motivation throughout my research. His academic rigor and thoughtful mentorship have been instrumental in shaping this work. I am equally indebted to my co-supervisor, **Dr. Cherry Bhargava**, whose steadfast encouragement, keen attention to detail, and deep commitment to academic excellence have had a profound impact on the quality and clarity of this dissertation.*

*I would like to express my sincere appreciation to the **members of the Research Advisory Committee** for their valuable feedback, constructive suggestions, and guidance, which significantly enhanced the depth of my work. I am also thankful for the support services extended by the university and the helpful assistance provided by the faculty and staff at **CRDP, Lovely Professional University**.*

*I owe an immense debt of gratitude to my **parents, Late Sherly and Toney Geroge**, my brother **Gladson** whose love, prayers, and belief in me laid the foundation for my educational journey. To my **partner, Adiya**, and my daughters **Tiara and Emeera**; their endless patience, encouragement, and sacrifice have been my anchor and driving force during the most demanding phases of this journey.*

*A special note of thanks to my esteemed colleagues, **Dr Dayakshini, Dr Aldrin C Vas, Mr Sathyendra Bhat, Dr S Ruban** and the **Leadership team at St Aloysius (Deemed to be University), Mangaluru** and **St Joseph Engineering College, Mangaluru** for their consistent support, encouragement, and understanding. Your faith in me has meant a great deal to me.*

*I would also like to acknowledge my **peers and fellow researchers** for the many enriching discussions, academic insights, and camaraderie we have shared. Your wisdom and companionship have been an integral part of this journey. This journey has been as much about people as it has been about research, and I remain deeply grateful to all who walked this path with me.*

TABLE OF CONTENT

Details	Page No.
Declaration	i
Certificate	ii
Abstract	iii
Acknowledgements	ix
Table of Content	x
List of Tables	xiv
List of Figures	xv
CHAPTER 1 INTRODUCTION	
1.1 Overview	1
1.2 Adaptive headlamps in Vehicles	2
1.2.1 Adaptive headlamps from major Vehicle manufacturers	9
1.3 Motivation	12
1.4 Thesis Organization	13
CHAPTER 2 LITERATURE REVIEW	
2.1 Overview	14
2.2 On-road Vehicle or object detection techniques	14
2.3 Vehicle Models for Adaptive Headlamps	20
2.4 Lane detection Models for Adaptive Headlamps	22
2.5 Research Gap	26
2.6 Objectives of the research	27
2.7 Research Methodology Overview	28
2.8 Novelty of the Research	30
CHAPTER 3 OPTIMIZING VEHICLE HEADLAMP ALIGNMENT VIA DYNAMIC MODELING AND DESIGN OF FILTERED PROPORTIONAL CONTROLLER FOR SLIP ANGLE BASED ADJUSTMENTS	
3.1 Introduction	33
3.1.1 Structure of the Chapter	35
3.2 Assessing the slip angle to ascertain the variance between the vehicle's positioning and the headlamp's placement	35

3.3 Mathematical model for assessing body slip angle and adjusting headlamp direction	36
3.3.1 Vehicle Dynamics (Longitudinal, Lateral and Yaw based)	38
3.3.2 Dynamics of Slippage	38
3.3.3 Tire Force Modelling	39
3.3.4 Steering and Wheel Dynamics	39
3.3.5 Longitudinal Motion and Rolling Resistance	39
3.3.6 Brake Force Distribution	40
3.3.7 Stability and Handling Parameters	40
3.3.8 Aerodynamics and Load Transfer	40
3.3.9 Cornering Limits and Critical Speed	40
3.3.10 Dynamic Load Transfer for Braking	40
3.3.11 Slip Angle and Headlamp Logic	41
3.4 Design of the Filtered Proportional Controller	41
3.4.1 Continuous-Time Control Law	42
3.4.2 Transfer Function Representation	42
3.4.3. Discrete-Time Implementation	42
3.4.4 Time and Frequency Domain Behavior	42
3.4.5. Stability and Causality	43
3.4.6. Actuator Constraints and Saturation	43
3.4.7. Integrating with the Vehicle Dynamics Model	43
3.4.8 Parameter tuning	43
3.5 Raw Full-State MPC with Injection of Direct Road Heading	45
3.6 Extended Model Predictive Control (MPC)	46
3.6.1 Dynamic Vehicle Model and Slip Angle Estimation	46
3.6.2. Control Architecture	46
3.7 Design of the proposed Feed Forward-Enhanced Model Predictive Controller (FF-MPC)	47
3.7.1 State Variables	48
3.7.2 Vehicle dynamics	49
3.7.3. The Feedforward for Headlamp Deflection Control	49
3.7.4. Road Heading Calculation	49
3.7.5. Steering angle calculation	49

3.7.6. Controller output	49
3.7.7. Constraints and Physical Limits	50
3.8. Summary of Controller Architectures	50
3.9 Results and discussion on the performance of Predictive control models for trajectory-aligned vehicle headlamp adjustment	52
3.9.1 Evaluation of the performance of the Dynamic Predictive Control (DPC) model	52
3.9.2 Evaluation of the performance of the Raw State Model Predictive Control (RS-MPC)	60
3.9.3 Evaluation of the performance of the Extended-Model Predictive Control (E-MPC)	62
3.9.4 Evaluation of the performance of the Feed Forward-Model Predictive Control (FF-MPC)	70
3.9.5 A Case Study on Segment-Wise Performance Analysis of Control Models	77
3.10 predictive headlamp control system using Vehicle dynamics and Sensor fusion	83
3.10.1 The mathematical Model of the Fused Controller (F-Controller)	83
3.11 Results and discussion	85
3.11.1 Performance of the Fused Controller	85
3.11.2 Comparison against FP, RS-MPC, E-MPC and FF-MPC	96
3.11.3 Performance Comparison: Fused Model vs. E-MPC on a 2 km Segment discussed in section 3.9.5 and Summary	101
 CHAPTER 4 INTEGRATED DEEP LEARNING FRAMEWORK FOR OBJECT DETECTION AND LANE-AWARE BEAM CONTROL IN NIGHTTIME DRIVING	
4. 1 Overview	105
4.1.1 Chapter overview	105
4.2 Object detection techniques	106
4.3 Deep Learning Framework for on-road Object detection for Adaptive Headlights	107
4.3.1 Modified SSD Architecture	107
4.3.2. YOLO V8 based Object detect	110

4.3.3 Architecture of the proposed Multi-Faceted framework	113
4.3.4 Inference-Optimized LSDNet Context-Aware Adaptation for Robust Object Detection in small Object and Low-Illumination Scenarios	122
4.4 Summary of the object detection models discussed	131
4.5 Results and discussion on the On-road object detection models	132
4.5.1 Performance of YOLO, Tiny YOLO, RCNN and SSD Models	133
4.5.2 Model Comparison	138
4.5.3 Modified SSD Architecture	141
4.5.4 YOLO V8 based Object detect	142
4.5.5 Performance of the proposed Multi-Faceted Framework	147
4.5.6 Performance of the proposed Inference-Optimized LSDNet Context-Aware Adaptation Network	152
4.5.7 Benchmarking against state-of-the-art models	160
4.5.8 Model Trade-off Analysis	171
4.5.9 Summary of the performance of LSDNet and future directions	173
4.5.10 Summary of the performance analysis of the object detection models	173
4.6 Overview on the Lane detection techniques	174
4.7 Framework for lane detection and integrating LSDNet model to operate in multiple driving scenarios	175
4.7.1 The lane detection	178
4.7.2 The calculation of the angle of vehicle wrt to the lane curvature	180
4.8 Integrating the multifaceted object identification model and the lane detection model	182
4.9 Discussion	189
4.10 Summary	189
CHAPTER 5 CONCLUSION AND FUTURE SCOPE	
5.1 Conclusion	191
5.2 Future Developments	193
REFERENCES	195
LIST OF PUBLICATIONS AND CONFERENCES	204
Annexure A1	207

LIST OF TABLES

No.	Label	Page No.
1.1	The Scanning units diode characteristic requirements for precise generation of single color light [10]	7
2.1	Chapter-wise details of Objective Implementation	30
3.1	Error rates across the four models in the various road segments	79
3.2	Comparison of the controllers for adaptive headlamp adjustment	80
3.3	Overview of the test results on the performance of F-MPC	96
3.4	Comparison of Overshoot and Undershoot with Interpretations for Beam Controllers.	97
3.5	Comparison and Interpretation of Beam Jitter Index Across Controllers	98
3.6	Qualitative Comparison of Headlamp Controllers	100
3.7	Comparison of Segment-wise RMSE between the Fused Model and the E-MPC Controller	102
4.1	Comparison of the Mod SSD against SSD 300	109
4.2	Overview of Night Driving Challenges addressed in the current research and the key contributions to enhance nighttime driving.	123
4.3	Major modifications in the LSDNet Model	126
4.4	Comparison of the three Models based on Frames per Second and mean Average Precision	140
4.5	Comparison of the three Models based Average Precision of detection	140
4.6	Training and validation losses and learning rate for different epoch	146
4.7	Comparison of the DetD0	160
4.8	Box plot Quartile values for various metrics across multiple models	161
4.9	Relative Performance of LSDNet, MF Model, and Modified SSD	169
4.10	Comparison of Object Detection Models for Nighttime Applications	170
4.11	The headlamp adjustment criteria based on the lane trajectory and the values of the intermediate variables r and θ	181
4.12	The adaptive headlamp compensation details to the changing conditions encountered during on-road driving. The illustrates the intermediate status of the hardware prototype, indicating the control points for this implementation wrt the Figure 4.52	188

LIST OF FIGURES

No.	Label	Page No.
1.1	The beam lobe projections in a Vehicle with and without adaptive framework. The image displays the difference in the beam lobe projections and the improvement in illumination an AF enabled system offers.	2
1.2	The arrangement of the LED array in the Parabolic AF system	3
1.3	The beam lobes emitted by the combination of the LED Array in its Zero state	3
1.4	The headlamp beam lobe adjustment relies on the relative angle of the driver with reference to the headlamp	4
1.5	The geometric calculation of the HSD value for the beam adjustment algorithm while the vehicle undertakes a turn	4
1.6	The control design of the preview-control based model showing the mathematical relations and the control segments	5
1.7	The pattern of the AF headlamp beam lobes in the different modes of vehicle transit. a. Class C-Basic/ country, b. Class V-Urban, c. Class E-Highway and d. Class W-Wet road	6
1.8	The uni-color light produced from multiple Laser sources having diodes that generate lights of different frequency undergoing recombination to produce a desired pattern	7
1.9	The multiple diode arrangement for the single ECE standard light generation setup	8
1.10	Three staged adaptive frameworks piping diagram displaying the inherent latency at the multi-stage system	8
1.11	The hardware prototype of the 3 staged architecture based adaptive headlamp set for fetching live data for the validation	8
1.12	The improved beam projections of the adaptive framework at different test cases on on-road driving scenario developed by Ford Corp.	10
1.13	The case of beam lobe adjustment across a curve for improving the driving perception using the framework from Ford	10

1.14	The system representation of the AF of Nissan-Motor-Corporation	11
1.15	The beam lobe projections at curves and at the crossroads using the Nissan AFS depicting an improved illumination	11
1.16	Comparison of pricing of the adaptive headlamps by various vehicle manufacturers	11
2.1	Sophisticated Object detection techniques suitable for night drive obstacle/ vehicle detection	15
2.2	Classifiers apt for the object detection application	15
2.3	Performance comparison of traditional Object Detection Features	16
2.4	Comparison of Machine Learning Classifiers use for Vehicle detection	17
2.5	Comparison of Deep Learning-based Object Detection Model for Adaptive Headlamp Applications	19
2.6	Passenger cars Vehicle models suitable for lane predictions	20
2.7	Comparative analysis of Vehicle Path Models for Adaptive Headlamps	21
2.8	Normalized comparison of Model Stability and parameter dependence	21
2.9	Trend of Model evolution from Geometric to Dynamic formulations	22
3.1	Front and rear wheel slip angle as vehicle goes through a cornering	36
3.2	The sequences in the slip angle-based headlamp deflection adjustment	37
3.3	The process of parameter tuning	44
3.4	The process flow of the E-MPC	47
3.5	The process flow of the FF-MPC	48
3.6	Headlamp deflection based on changing slip angle	53
3.7	Proportionality of Slip Angle to the Road Curvature	53
3.8	Correlation between Slip Angle and Road curvature vs Speed	54
3.9	Step response of the DPC Model	54
3.10	The Impulse response	55
3.11	Slip Angle-Driver Input Variation	56
3.12	Effect of Aerodynamic Drag on Headlamp deflection	56

3.13	Effect of Braking on Headlamp deflection	57
3.14	Effect of Tire stiffness on Headlamp deflection	57
3.15	Effect of Yaw inertia on Headlamp deflection	58
3.16	Bode Plot of DPC	58
3.17	Nyquist Plot of DPC	59
3.18	Headlamp deflection against heading road at varied intervals	61
3.19	Tracking Error of RS-MPC Model	61
3.20	Tracking Error of RS-MPC Model	61
3.21	Headlamp deflection as filtered response to slip angle	63
3.22	Tracking Delay-step response	63
3.23	Ramp Response	64
3.24	Impulse Response	64
3.25	Disturbance rejection	65
3.26	Noise Sensitivity	65
3.27	Disturbance Rejection	65
3.28	Input saturation behavior	66
3.29	Rate of Change of Heading Angle	66
3.30	Filtered Response with and Actuator Constraints	66
3.31	Tracking Performance of E-MPC's Road Deflection, Slip Angle and Headlamp Response	67
3.32	Extended MPC response in Constant Speed	68
3.33	Extended MPC Case 2 Varying Speed	68

3.34	Bode Plot of E-MPC	69
3.35	Nyquist Plot Extended MPC	69
3.36	Headlamp Vs Road Heading at constant speed	71
3.37	Headlamp Vs Road Heading at varying speed	71
3.38	Step Input Response	72
3.39	Ramp Input Response	73
3.40	Error profiles under step input under various parametric combinations	73
3.41	Error profiles under ramp input under various parametric combinations	74
3.42	Bode plot of the FF-MPC First Order Filter	74
3.43	Nyquist plot of the FF-MPC First Order Filter	75
3.44	Bode plot of the full FF-MPC Model	75
3.45	Nyquist plot of the full FF-MPC Model	76
3.46	Headlamp vs Road Heading of FF-MPC	76
3.47	A Synthetic 2 km Road stretch with 10 segments	77
3.48	An overview of the fused headlamp control model: (a) A speed profile that shows the phases of acceleration, cruising, and deceleration; (b) steering input that replicates real-world maneuvers; (c) A comparison between the true and fused yaw rate; (d) A comparison between the true and fused slip angle; (e) A beam response for one-step predictive control; and (f) Multi-step predictive control with speed-adaptive gains.	86
3.49	Response of the System to Step Steering Input: (a) Step input is given to the steering angle in order to replicate a lane change or sudden turn. (b) Raw (rapidly reacting with overshoot) and filtered (smooth, lagging) beam angle response is shown. (c) An error plot of the	87

difference between the raw and filtered beam angles that depicts the momentary smoothing action of the filter.

- 3.50 Dynamic Response of the Headlamp Controller to Sine Sweep 88
Steering Input: (a) A steering input signal that displays the transitions from slow curves to high-speed maneuvers by gradually increasing the frequency from 0.1 Hz to 1 Hz. (b) A comparison of filtered and raw beam response outputs. Both follow well at low frequencies, but smoothing actuator-friendly makes the filtered response lag at high frequencies. (c) The desired input profile by monitoring the change in frequency over time.
- 3.51 Effect of Sensor Noise and Fusion on Controller Stability: 89
(a) Comparison of yaw rates showing the effect of noise and the effectiveness of fusion. (b) Slip angle comparison, where the raw noisy estimate is effectively smoothed by means of fusion, providing reliable inputs for control. (c) Beam angle output in clean vs noisy fusion.
- 3.52 Beam Alignment Relative to Road Orientation under Dynamic 90
Curvature: (a) Comparison of the beam angle with the road heading angle. (b) Beam alignment error during the movement.
- 3.53 Context-Aware Beam Spread Control for Road Curvature: a. The 91
shaded area depicts dynamic beam dispersion, while the beam center (solid line) follows the road heading (dashed line). Beam fans out as road curves, increasing visibility where it matters. b. Beam spread width plotted over time shows how the system reacts, extending during turns and contracting on straight parts to balance visibility and energy economy.
- 3.54 Actuator Rate Limiting Effects: a. Delay between commanded and 92
actual beam angle as a result of actuator limitations. b. Tracking error is at its highest during rapid changes but stays within bounds.
- 3.55 Latency Effect on Beam Tracking: (a) Beam response with and 93
without 300 ms delay shown with a critical lag in abrupt transitions. (b) Sudden maneuvers maximize latency-induced error, demanding predictive or low-latency control.

3.56	Urban vs. Highway Controller Behavior: (a) Beam angle changes dramatically in cities but is constant on highways. (b) Urban speed varies (4-8 m/s) but highway speed is constant (18-22 m/s).	94
3.57	Dual Beam Directional Illumination: (a) Only the turning-side beam changes since left and right beams deflect asymmetrically while turning. (b) Combined illumination is directed toward the curve, enhancing visibility without lighting up the other side excessively.	95
3.58	Beam Alignment Error and Drift Over Time (a) Instantaneous alignment error stays at $\pm 2-3^\circ$, which shows steady short-term tracking. (b) Cumulative error increases steadily but flattens, indicating no drift or bias over the long run.	95
3.59	Comparison of five headlamp control strategies—FPC, RS-MPC, E-MPC, FF-MPC, and the present fused method—versus the road heading reference. Each subplot illustrates the beam tracking behavior of a single controller, thus revealing differences in responsiveness, alignment accuracy, and steady-state behavior.	99
3.60	Overhead view of the improved dual-beam projection along a S-curve.	100
4.1	The Modified SSD Architecture	108
4.2	The YOLO V8 based object detection Convolutional Network architecture.	111
4.3	The detailed layers of the MF Architecture: Head-Backbone-Tail.	115
4.4	The major sub-blocks of the MF Architecture	116
4.5	The overall LSDNET Architecture	125
4.6	YOLO model object detection for dense traffic at night	134
4.7	Frames per second on YOLO Model	134
4.8	YOLO Model Emphasized by mean Average Precision	134
4.9	YOLO-COCO AND PASCAL Metrics	135
4.10	Object detection using YOLOv3-Tiny in dense traffic at night	135
4.11	Frames per second for YOLOv3-Tiny Model	135
4.12	mean Average Precision value for YOLOV3 Tiny Model	136

4.13	Screenshot of the statistics as obtained for the YOLO-Tiny Model calculated with PASCAL VOC	136
4.14	TP & FP for the RCNN Model	137
4.15	mAP of the RCNN Model	137
4.16	Object detection using SSD Model in dense traffic at night	133
4.17	Frames per second for SSD Model	138
4.18	mean Average Precision value for SSD Model	139
4.19	COCO & PASCAL metrics for SSD Model	139
4.20	ROC Curve for Modified SSD	141
4.21	Confusion Matrix of V8 based Model	143
4.22	F1-Confidence matrix for V8 based Model	144
4.23	Precision-Recall for V8 based Model	144
4.24	Precision-Confidence Curve- V8 Model	145
4.25	Recall-Confidence Curve- V8 based Model	145
4.26	Training & Validation losses for various epochs- MF based Object detection Model	149
4.27	Losses and learning rate for Successive Epochs of the MF model.	151
4.28	F1 Vs Confidence Curve- MF Model	153
4.29	Recall Vs Confidence Curve- MF Model	154
4.30	Precision Recall Curve- MF Model	155
4.31	ROC Curve- MF Model	156
4.32	Variation of metrics under varying lighting conditions, from Very Dark (0.3), Dark (0.5), Dim (0.7), Normal (1.0), Bright (1.3), Very Bright (1.5), to Extreme bright (1.7). a. Average Precision Vs	158

	Varying brightness, b. Accuracy Vs Varying brightness, c. Recall Vs Varying brightness, d. F1 Score Vs Varying brightness, e. FPS Vs Varying brightness.	
4.33	Metrics comparison across models	161
4.34	Radar Chart of the multiple model metrics	162
4.35	Heat map of multiple model plotted across the performance metrics	163
4.36	Scatter Pair plot of the multiple models being compared	164
4.37	Plot of trade-off between FPS and Recall across various models	165
4.38	Box plot of the performance distribution of metrics across various models	166
4.39	Z Score and relative performance comparison of various models	168
4.40	The lane detection intermediate processes	176
4.41	An original frame	179
4.42	Image-Gamma Corrected	179
4.43	Image with Bilateral Filter	179
4.44	Picture employing Canny Edge recognition	179
4.45	Hough Lines Clustered in an Image	179
4.46	Filtered Image with HSV	179
4.47	Both during lane changes and while the vehicle moves across the path, lane recognition and projection are performed. The road that the	180
4.48	car is traveling on is reflected in these projections.	
4.49	The left and right curvatures are computed by the model based on the lane projections. In a and b, the model that forecasts the lane	180
4.50	projections under various lighting scenarios is displayed.	

4.51	The dashboard perspective to compute the compensation value for adjusting the headlamp to enhance visibility when the vehicle executes a turning maneuver within the lane.	181
4.52	The dashboard view for the calculation of the value for compensating the headlamp for improving visibility as the vehicle does a turning maneuver across the lane	183
4.53	The PCB designs and implementation.	184
4.54	The software design and the hardware prototype of the adaptive headlamp model for integrating the LSDNet and the Lane detection techniques.	185
4.55	The plot illustrates the pulse modulation responsible for activating and deactivating the MOSFET, subsequently regulating the high beam strip of the vehicle's headlamp.	186
4.56	Variation plot of the vehicle along the trajectory and the corresponding angles calculated by the lane detection model.	186
4.57	The duty cycle of servo motors 1 and 2 during left and right maneuvers of the vehicle across frames.	187
4.58	The model integration enables the concurrent detection of on-road objects and the trajectory angle.	188

CHAPTER 1

INTRODUCTION

1.1 Overview¹

Night driving using conventional headlights is very dangerous and this results in accidents to temporary blindness caused from the beams of the oncoming vehicles. Headlights with high beams induce a temporary visual impairment in humans called the Troxler effect. This reduces reaction time but also contributes to poor visibility which causes most accidents occurring at night. Traditionally, the adjustment of headlamps was done manually whereby lack of skills and errors in judgment could lead to devastating outcomes. Additionally, some accidents occur due to inadequate lighting as the present typical headlamp set-up cannot give precise illumination for instance on bends and uneven terrains. Therefore, automobile adaptive headlamps that can prevent Troxlers' effect happening on the opposite drivers while still not affecting road illumination for the driver inside should be developed. Automotive Electronics has gained a reputation as it is involved in more than just enhancing convenience; it also assists in safety purposes. It has been confirmed that there are more accidents during nights than day times even though traffic volume during darkness is relatively lower compared to its counterpart earlier on stated above [1].

One third of accidents in the Indian sub-continent happen due to poor visibility at night, as stated by NIMHANS, Bengaluru [2]. This is attributed majorly to the road lighting conditions, and the Troxler's effect. The majority of cars fitted with headlamps still have them on manual control, and only a few drivers change between the high beam and low; this portends an instant crash in case the illuminated light leads to temporary blindness for the approaching driver or there are some reflections from the mirrors if another car goes ahead. This is what is known as the Troxler's effect [3]. Further destruction results when such a route happens to be curved or uphill. Normally, low beam systems do not illuminate properly along curved roads leading to more pedestrian accidents taking place. Therefore, it calls for a sophisticated framework that can conveniently switch between high and low beams thereby showing the way effectively [4]. In order to increase visibility among

¹ This chapter is published in IEEE Access Journal. Details: Toney, Glenson, and Cherry Bhargava. "Adaptive headlamps in automobile: A review on the models, detection techniques, and mathematical models." *IEEE Access* 9 (2021): 87462-87474.

motorists, this serves as a significant leap into enhanced safety and driving experience through illumination of roads at turns during nights.

This chapter intends to determine the need for technology solutions that can aid in preventing accidents at night, specifically due to the effect of high beam spectrum of the headlamps and its ability in saving lives through affordable technological interventions. The chapter discussed existing adaptive headlamp frameworks in the market, outlines various object detection methodologies and lane detection techniques that can be used for adaptive headlamps.

1.2 Adaptive headlamps in Vehicles

The Adaptive Front (AF) light system framework increases the perceptibility of drivers by dynamically varying the beam projections as the vehicle moves [5]. Figure 1.1 illustrates an AFS that modifies headlamp beam-lobes when the driver makes a turn to right or left sides to improve road visibility for drivers and hence avoiding probable accidents that may involve pedestrians or any objects.

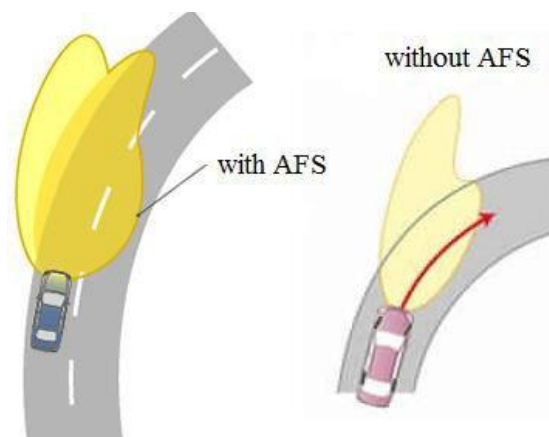


Figure 1.1 The beam lobe projections in a Vehicle with and without adaptive framework. The image displays the difference in the beam lobe projections and the improvement in illumination an AF enabled system offers. [5]

This model modulates the divergence of headlamp using a stepper motor. Furthermore, Dahou, H. et al. [5] also came up with an AFS on FPGA Board through PWM Technique which was meant to assist not only the onboard driver but also other drivers who may be coming from the opposite direction since their lives are equally important like those of their counterparts. In this case, the team came up with a lighting system that is parabolic in nature as shown in figure 1.2, consisting of four LED-lamps for illuminating the road.

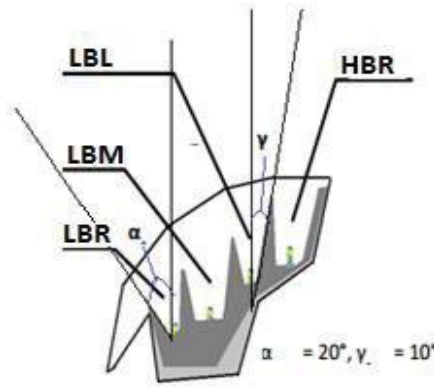


Figure 1.2 The arrangement of the LED array in the Parabolic AF system [5]

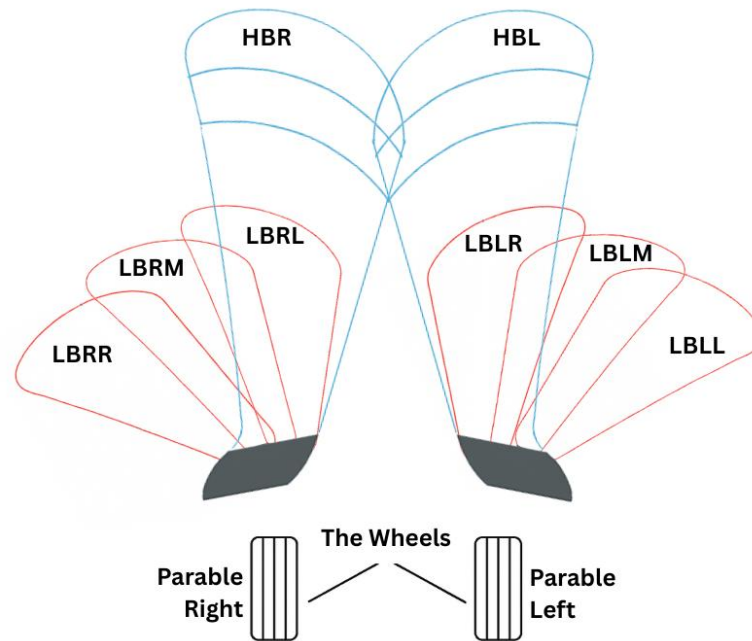


Figure 1.3 The beam lobes emitted by the combination of the LED Array in its Zero state [5]

Each LED contributes to one of these modes: Right or Left High Beam- HBR/HBL, Right or Left Low beam- LBL/LBR, Low beam Middle- LBM, and Right or Left Low beam- LBR/ LBL. At the roll axis, HB as well as the LBM lamps are aligned in parallel and deflected at angles 10 degrees and 20 degrees respectively as they project side beams for the left headlamp through the LBL mode & LBR respectively. The beam projections from each of these lamps form a combined lobe which is varied with respect to the steering wheel angle. Figure 1.3 depicts the beam lobe in zero state. HB lamp illuminates highly intense light along its path while PWM Modulator controls LB lamps projection ON/OFF using vehicle motion. Stepper motor's frequency and current is regulated by PWM technique and it also regulates the electric power applied to vehicles' parabolic lights. The

brightness of low-beam lamps is adjusted using the PWM, depending on what mode it is at. Bending angles within a range of 0 degrees to about 45 degrees were tested for most cases, and these results have shown to be effective. But, examining system latency that defines how long it takes for it to readjust driving conditions (that means time required for the response) which plays a crucial role in automotive industry is crucial. However, this system does not assist drivers in coming vehicles even though it enhances onboard driver visibility.

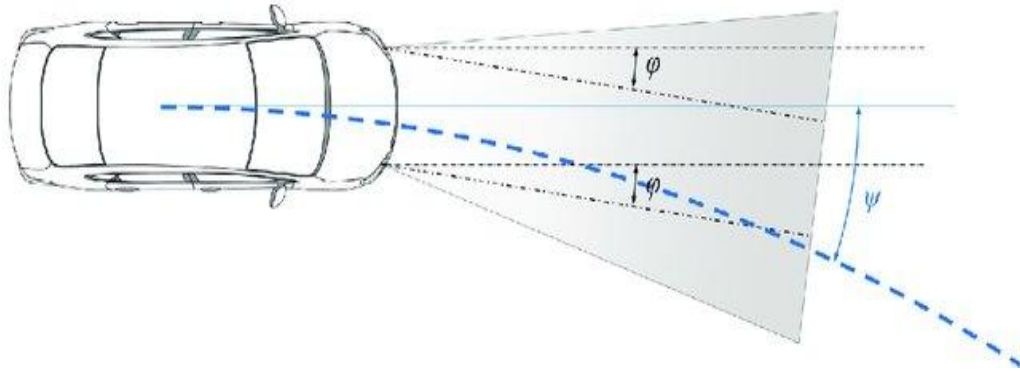


Figure 1.4 The headlamp beam lobe adjustment relies on the relative angle of the driver with reference to the headlamp, the angle of which is shown as ψ [6]

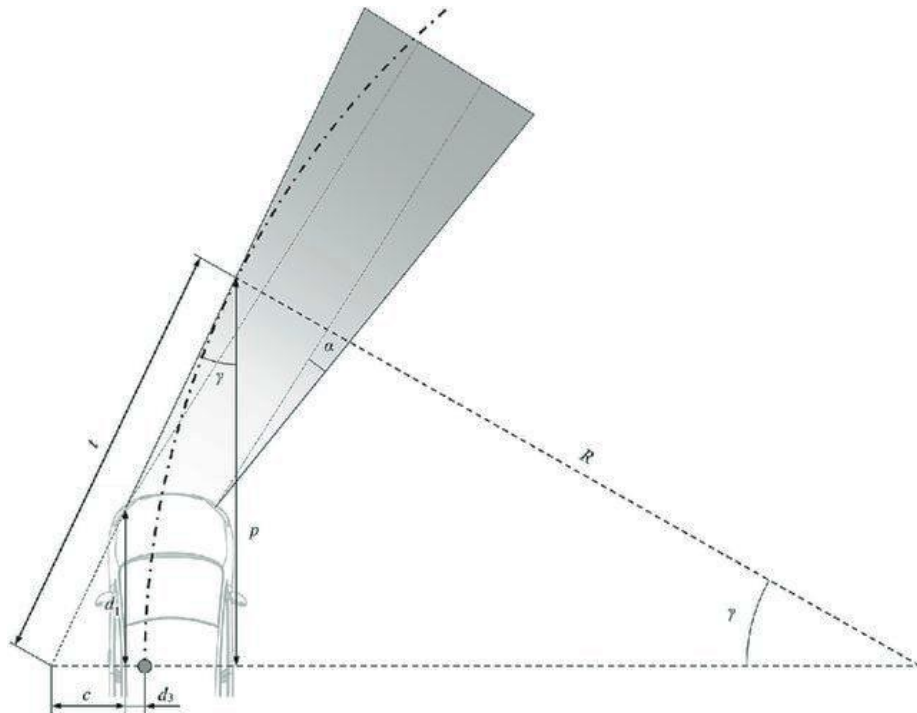


Figure 1.5 The geometric calculation of the HSD value for the beam adjustment algorithm while the vehicle undertakes a turn [6]

The Swiveling algorithm based headlamp [6] has its base relying on the highway geometry to predict the Headlight-Sight-Distance (HSD) and its effect on varying road designs were studied as shown in figure 4. The vehicles headlamp position in relation to the driver is

used to determine the HSD which in turn adjusts the beam-lobes. This methodology relies on SGPSA-HSD value which is the Steering-Governed Predictive Swiveling Algorithm. The geometrical scheme of calculation for this is depicted in figure 1.5. The system is simulated for various on-road test cases on MATLAB and effectively demonstrated that the compensated headlamp improves the sight distance and contributes to improving visibility and driving comfort. The model does offer improvement in illumination of the driving path with reference to the steering angle. As depicted in the figure 1.6, the arc design for the HSD prediction plays a crucial role as they compensate for the curve transitions. But the system fails to perform as expected during sudden curvature changes (which cannot be ruled out in real-world instances). The system needs to be aligned to the highway road standards, which is essential. Also, the system's ability to perform for different speed ranges is not discussed and it is quintessential in an application like adaptive headlamp frameworks.

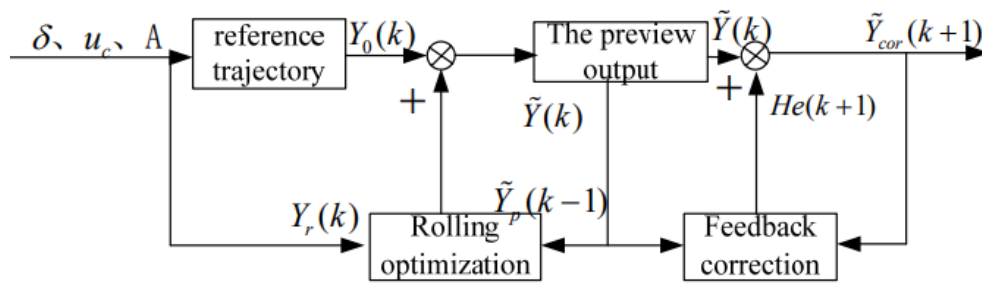


Figure 1.6 The control design of the preview-control based model showing the mathematical relations and the control segments [7]

A uni-track non-linear model of an elementary engine for producing different position estimate based on the changing values of the steering angle, yaw rate, throttle inputs and velocity is used in Hardware-in-loop based model [7]. the vehicle and on-road simulator models of the system simulate multiple test cases as applicable for the adaptive headlamps. Based on the varying inputs, the controller predicts the changing angle of trajectory and corresponds the motor steps accordingly. The output in this case is smoother and so is the headlamp compensations unlike the earlier model which has abrupt behavior at sharp turns. This dissolves behavior that are uncontrolled and can operate effectively in different driving conditions too. The model however has to be optimized for real-time applications for power and size and have to be rigorously validated on test-cases comparable to actual environment.

A preview control based bending mode controller was designed to compensate for the lag issues usually found in adaptive frameworks [8]. This implementation banks on the inherent relationship that controls the braking distance (safe stretch), the steering angle, its Wheel angle and also the turn radius. The controller design is shown in the Figure 1.6. This model exhibits improved response time in comparison with the conventional servo-powered smart beam lamps. When compared with the servomotor based models at a vehicle acceleration of about 40 KM an hour, this model leads with the beam lobe adjustment with a faster time of about 0.4 seconds with improved performance against changing driving angles. The model leads in the driving speed range of low and medium however, the performance at higher speeds and the associated lag needs to be evaluated.

Another prominent adaptive framework designed in line with the United Nations(ECE324-R123) operates effectively in four classes as defined by the standard [9]. As shown in Figure 1.7, the quad class of operation has a neutral or the country light state (Class C), an urban light state (Class V), a highway drive state (Class E), or an adverse weather condition light (Class W). The implementation has a specially designed optics based on the ON-OFF cases of the LED array that operates in the cutoff (the narrow central area is lighted up using this mode which propels high intensity light beams) and the spread module mode (generates low intensity beams).

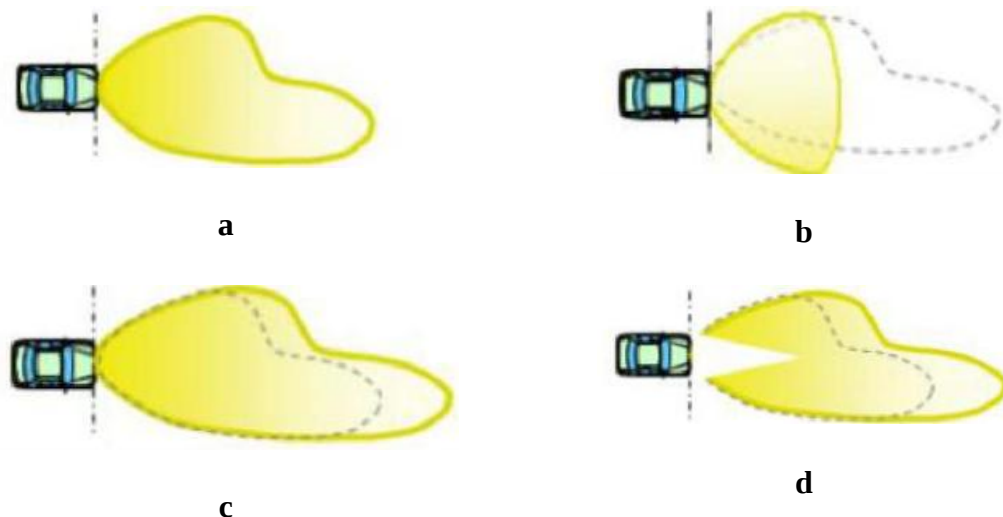


Figure 1.7 The pattern of the AF headlamp beam lobes in the different modes of vehicle transit [9]. a. Class C-Basic/ country, b. Class V-Urban, c. Class E-Highway and d. Class W-Wet road

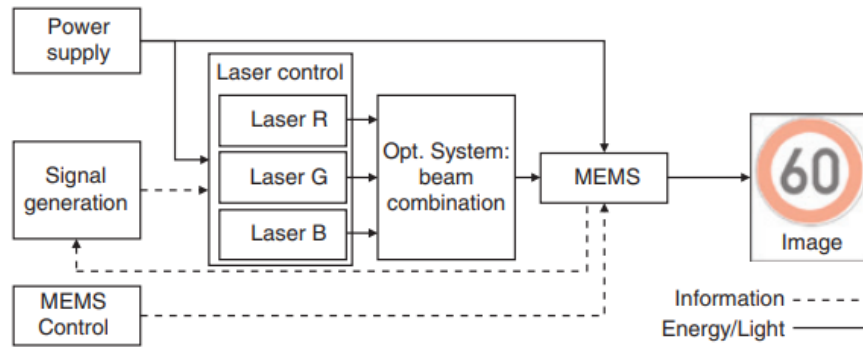


Figure 1.8 The uni-color light produced from multiple Laser sources having diodes that generate lights of different frequency undergoing recombination to produce a desired pattern [10]

The cutoff and the spread modes of the optics form the beam patterns for the four classes using the optic module that has the multiple array of LED's. The selection of the mode of operation is primarily dependent on the vehicle speed relying majorly based on the fact that the European Union has strict vehicle operating speed ranges in each of these classes. But the universal application of this may not be practical in other parts of the world.

Table 1.1 The Scanning units diode characteristic requirements for precise generation of single color light [10]

Diode Color	Red	Green	Blue
Output power (Optical unit)	0.5W	0.15W	1.6W
Wavelength	$638 \pm 6\text{nm}$	$520 \pm 10\text{nm}$	$450 \pm 10\text{nm}$
Divergence- Max(I_{FMHW})	36	23	23
Divergence- Min (II_{FMHW})	6	7	7
Relative Polarization orientation	p	s	p

By mixing the light of two diodes having different colored output and by blending three or more diodes for a Laser based scheme that has adaptable colored symbol projects were proposed [10]. Here the different colored light sources undergo optical beam combination as shown in figure 1.8 to produce a single colored light output with improved quality. This also included the generation of the longer UV wavelength (blue signal) using the shorter wavelength Phosphor. The white light is generated by blending the generated blue light with the converted yellow light. Since the system is laser based, the light is monochromatic in nature. While this emerges as a good option for projection, the need to adhere to the ECE standard regulation on Color-rendering Index is crucial for illumination application which

is the said case. This strict need for adherence to the standard is a bottleneck and the required range is shown in the Table 1.1.

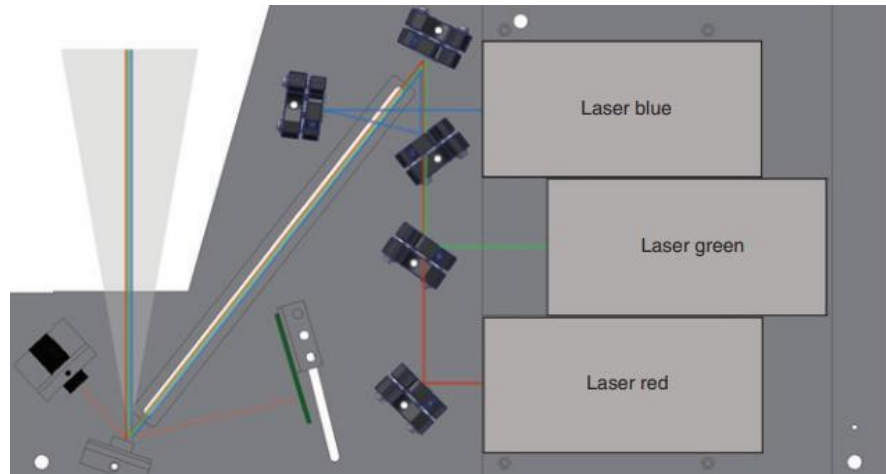


Figure 1.9 The multiple diode arrangement for the single ECE standard light generation setup [10]

The method aids in better and controlled illumination but does come with an overhead of precise combination of laser diodes to generate colors that are specific. Also, the need for aligning the laser diode precisely as shown in Figure 1.9 to obtain the desired light streak is a challenge. Another demanding task in this case is to have proper thermal compensation because the diode output wavelength can get affected due to temperature generating inappropriate light streaks. Another scheme with reduced latency of detection and response to accommodate climatic changes is the visual framework [11]. This method is found to be performing well even at higher speeds effectively at the highway limits and implements: lane marking detection, sidewalks identification, quick detection of warning signs to aid for improving driving comfort.

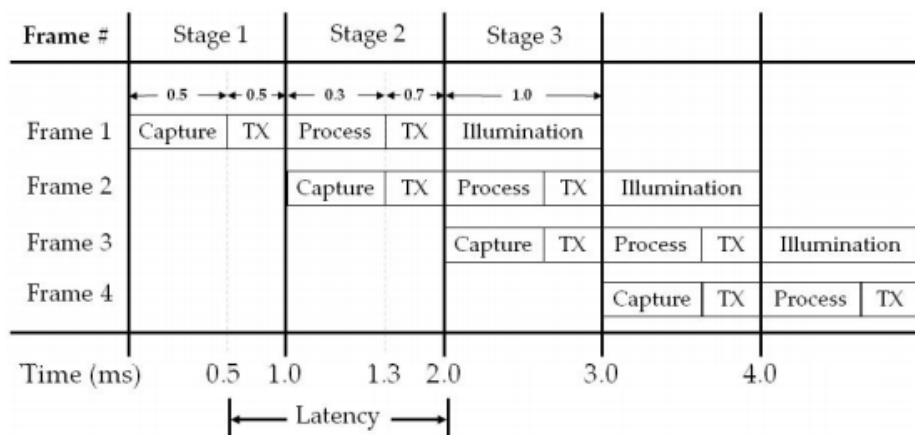


Figure 1.10 Three staged adaptive frameworks piping diagram displaying the inherent latency at the multi-stage system [11]



Figure 1.11 The hardware prototype of the 3 staged architecture based adaptive headlamp set for fetching live data for the validation [11]

With three stages, Capturing, Processing and Transferring, the framework displays low latency and higher accuracy (Figure 1.10). Stage 1 involves scenario acquisition through cameras, stage two includes image processing to identify required knowledge from the background (this stage contributes to the latency) followed by the stage 3 which is the action phase that actuates the control module for headlamp beam adjustments. The three-stage architecture displays latency of 1ms which is permissible in the application under discussion and also finds that the reaction time are within the standard deviation peaks for over 63% of times. The results in this case are promising but the fact that these are based on just six trials calls for the need for more test case simulations to determine the performance in varied scenarios. Also, optimization in terms of area and power are predominantly important for this model displayed in figure 1.11.

1.2.1 Adaptive headlamps from major Vehicle manufactures

With the advent of smart vehicles and now autonomous vehicles, the need for adaptive headlamps has become paramount and vehicle manufacturers are leading in this segment. Major brands like Nissan, Hella, Ford Corporations are industry leaders in this area. The section discusses various industry solutions and patents in the sector. The dynamic LED based headlamps of Ford Global Technology showcased in figure 1.12 has a driving framework for light that has a source of beam, set of projection lenses fused along with micro-mirrors (digital) and has been granted a Mexican patent [12]-[15]. The systems data acquisition is enabled through a camera and is predominantly designed to identify the

parking section and adjust the headlamp to light the parking limit perpetually.

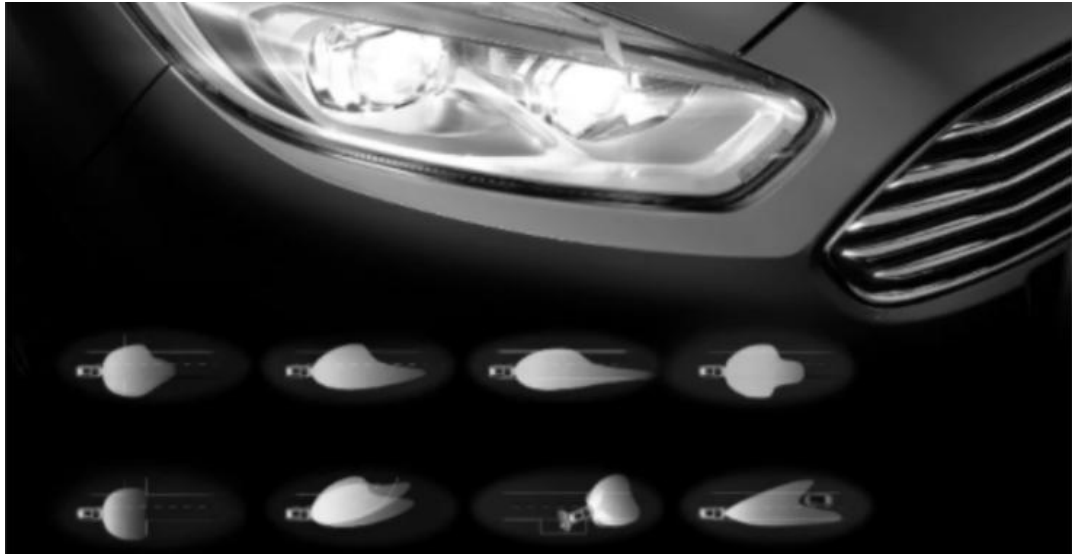


Figure 1.12 The improved beam projections of the adaptive framework at different test cases on on-road driving scenario developed by Ford Corp. [14]



Figure 1.13 The case of beam lobe adjustment across a curve for improving the driving perception using the framework from Ford [14]

Yet another Ford product is the auto beam adjustment system which acts as a conducive driving aid at night detects the traffic (approaching) or even a vehicle that is ahead (acts as reference vehicle) and inherently reduces the troxlers' effect on the other drivers and also curtails the onus on the driver of the vehicle with this system to automatically adjust the beam lobes. These headlamps display accurate light pattern and immensely improves the illumination as shown in Figure 1.13. A framework designed by Nissan was granted the European patent for its ability to transcend the beam lobes by following the road curves [16].

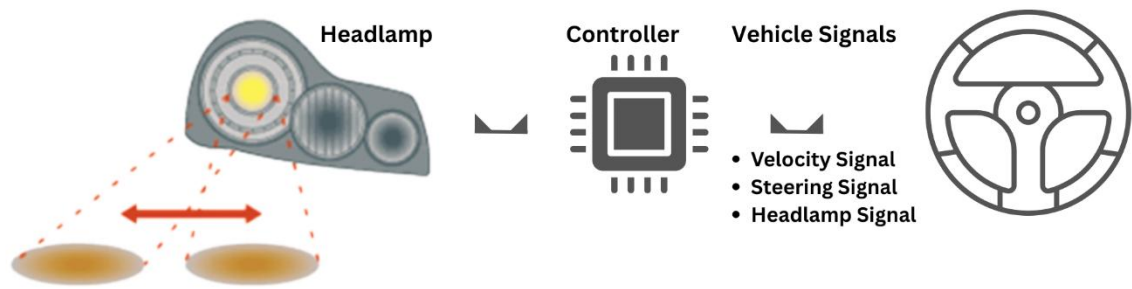


Figure 1.14 The system representation of the AF of Nissan-Motor-Corporation [16]

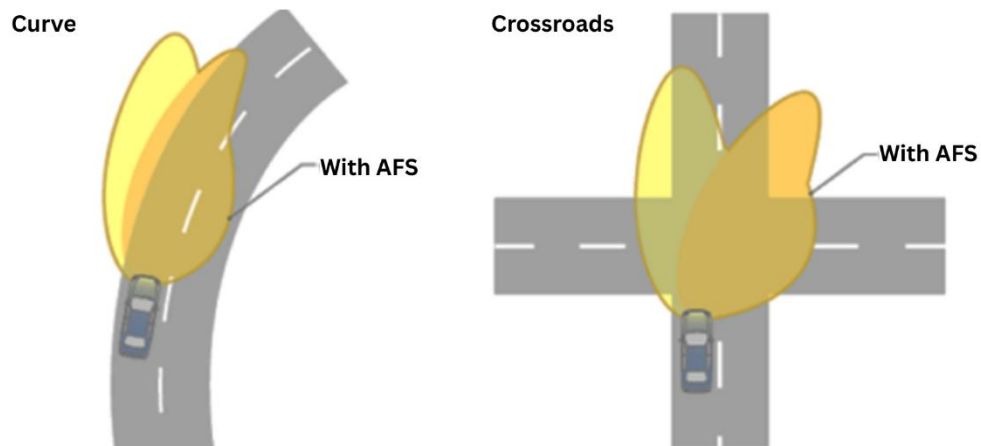


Figure 1.15 The beam lobe projections at curves and at the crossroads using the Nissan AFS depicting an improved illumination [16]

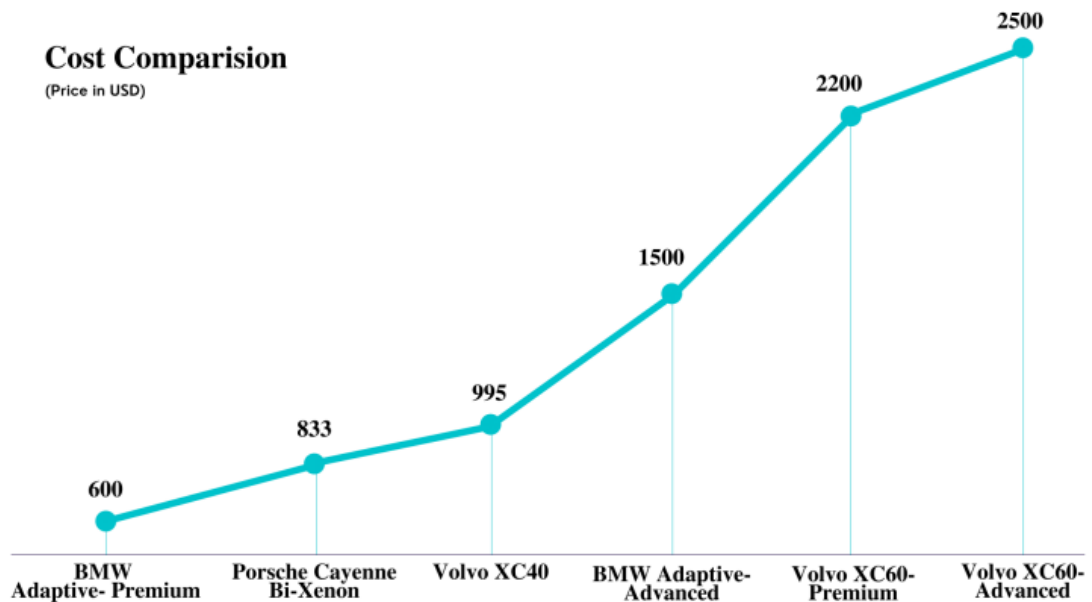


Figure 1.16 Comparison of pricing of the adaptive headlamps by various vehicle manufacturers [17]-[19]

The framework is depicted in Figure 1.14 and at the road junctions, where the vehicle needs to maneuver a turn as shown in Figure 1.15, the headlamp illuminates the intended direction

of turn based on multiple sensor inputs to improve illumination and hence driving efficiency. Depicted in Figure 1.14, an AFS developed by Nissan Global teams which turns on low beam just like as per vehicle speed and steer to give the driver better visibility. Companies such as BMW, Toyota Corporations, Hella, Valeo, and Benz has designed and patented adaptive headlamps. A cost-comparison of the adaptive headlamps by the renowned automobile manufacturers are depicted in Figure 1.16 [17]-[19]. This is to say that researchers should put attention to this cost of these products in an effort to develop low-cost alternatives. Because if for example in India which mainly have mid-range cars on the road most expensive features will be difficult for most manufacturers to adopt.

1.3 Motivation

1. **Safety:** Due to reduced vision, driving at night is much unsafe than driving during the day. Conventional headlights provide limited illumination on bends and uneven terrain, and they can temporarily blind oncoming cars (Troxler effect). By automatically adjusting the beam pattern, adaptive headlights can reduce accidents by increasing driver visibility without blinding onlookers.
2. **Improved driving experience:** Adaptive headlights improve visibility on curves and uneven roadways, enabling drivers to detect possible hazards more clearly and respond more quickly. This may result in a more secure and enjoyable nighttime driving experience.
3. **Reduce physical excretion and fatigue:** It can be exhausting for drivers to continually switch between high and low beam while using traditional headlights. This procedure is automated by adaptive headlights, which lessens driving strain and tiredness.
4. **Simple yet powerful solution with less reliance on sensors:** Modern adaptive headlights majorly rely on sensors and integrated systems for operation, which might be error-prone and limit response time and performance for complicated road features or at high speeds. Therefore, in order to recognize objects and pedestrians, predict approaching bends, and modify headlamp beams accordingly, camera-based systems with fusion algorithms are required. Faster reaction times and more accurate beam control would result from this, especially in emergency scenarios.
5. **Less expensive for markets in developing countries:** Adaptive headlights have a significant potential to increase traffic safety at night. Adoption may be hindered by their present cost, especially in developing nations like India. There is a need to create low-cost solutions with simpler sensor setups, efficient control algorithms, and alternative light

sources. This could facilitate a broader market acceptance, particularly for the mid-range cars that are popular in India.

1.4 Thesis Organization

This thesis is organized into six chapters.

Chapter 1 outlines the issues with night-time driving, the challenges associated outlining the need for the research. The chapter also discusses various adaptive frameworks available in the market. The chapter draws motivation for the research based on the discussion.

Chapter 2 includes a thorough literature review on the various object detection methods, lane detection techniques, controllers and vehicle models apt for adaptive headlamp design. The chapter discusses the research gaps and then presents the objectives of the research work.

Chapter 3 discusses the mathematical modeling of the controller for the beam lobe adjustment using the slip angle of the vehicle and its effect to varying body stiffness. The chapter introduces 5 controllers progressively and performs various time and frequency domain tests to evaluate them for the application under discussion.

Chapter 4 compares the existing state of the art object detection models and identifies the apt algorithm for the case. The chapter shows modification of the SSD framework for night drive object detection and proposes a multi-faceted object detection model for object detection. It also introduces a custom LSDNet model for night time object detection and compares its performance with state of the art models and also its performance in low-light conditions and its ability to detect small objects. The second half of the chapter detects a lane detection technique and integration of it with the LSDNet model to control headlamps in lanes where the road markings are effective.

Chapter 5 discusses the conclusion and future scope of the research.

CHAPTER 2

LITERATURE REVIEW²

2.1 Overview

Reduced vision during night drive reduces the driving comfort, increasing physical excretion and causing mishaps that often prove fatal. Object detection and lane detection becomes pivotal in the design of adaptive headlamps and associated systems. This chapter reviews multiple object detection techniques as apt for night time driving, their applicability and area for improvements. It deals with lane detection methods and also vehicle models suitable for this application. The chapter chalks out the gaps in the existing methods and defines the objectives of this research. It also outlines the methodology followed in the research work.

2.2 On-road Vehicle or object detection techniques

To control the beam lobes during adaptive headlamp design, target detection and tracking are necessary. In frame identification process detects the object while repeated detection of the object in each frame is referred to as tracking. However this method has a limitation speed because it is computationally expensive. So, size, shape, direction based target-tracking algorithms form the foundation for predicting the position of the objects in the subsequent frames. This method reduces the response time by avoiding searches in frames that are large. While the detection of target and tracking is performed parallely, the potential of failure in the model being able to detect the object is a possibility because of its prodigious dependence on the features. This section discusses paramount methods that are used for target detection through features extraction, classification, and the subsequent deep learning models.

As illustrated by figure 2.1 this section focuses on various methods employed for object identification techniques especially applicable to roads. On the other hand, figure 2.2 shows classifiers for detection purposes. Vehicle or object detection is usually associated with delineating the field of interest in the frame, feature retrieval and its classification. Video based vehicle identification may use motion or even appearance based [20]. The motion based models work on comparing the multiple frames against the background while the appearance dependent models predominantly rely on the features: size, color or even shape.

² This chapter is published in IEEE Access Journal. The details of the paper: Toney, Glenson, and Cherry Bhargava. "Adaptive headlamps in automobile: A review on the models, detection techniques, and mathematical models." *IEEE Access* 9 (2021): 87462-87474.

To distinguish between foreground and background a priori knowledge is used by former. These models use common feature extracts; dominant among them are SIFT extractors [21], SURF [22] and the HOG [23] concatenated with complex classifiers like SVM, along with deep learning models like R-CNN that are region based which boast of powerful hierarchical feature extractions contributing to improves accuracy and increased semantic-segmentation, faster R-CNN and noteworthy regression models used include SSD's and multiple versions of YOLO.

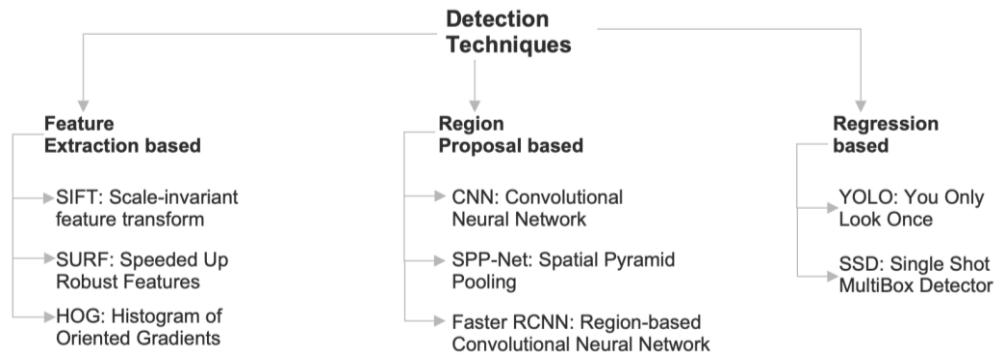


Figure 2.1 Sophisticated Object detection techniques suitable for night drive obstacle/ vehicle detection

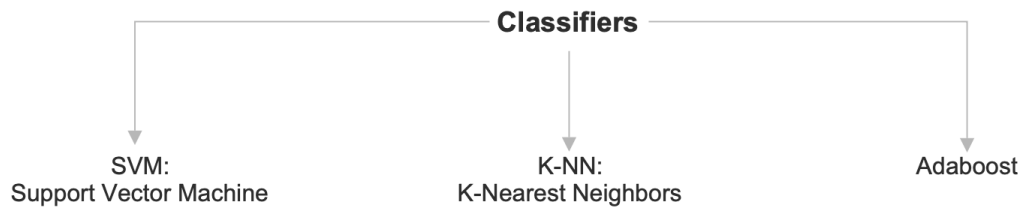


Figure 2.2 Classifiers apt for the object detection application

In videos with cars or bicycles, SIFT can be used to identify certain valuable features from a frame by segmenting it since the method had been widely employed in detecting an object from a video where it enables recognition of vital features. The invariant nature of feature points defined in SIFT method makes them suitable for object tracking even when an object changes poses frequently or its illumination varies much[20]. It mostly works better for 2D than 3D planar objects. New images are compared using this feature against those known before; thus Euclidean distance is applied to discover matching features. Then after matching these characteristics, probability of presence of an object are calculated. The literature suggests that SIFT-based descriptors are better than others and are invariant to scale, rotation and brightness [21] because of their region based nature which is useful for matching features. SURF is a popular image recognition, registration and classification method used in computer vision which relies on the integral images computed using

Hessian approximations thus making detection much quicker than the SIFT model [22]. The selection of SURF or SIFT is dependent on the problem as well as the response speed requirement. In applications with ample computational time SIFT performs superior while Gaussian derivative-based descriptors for SIFT are found to be better than SURF [23]. The comparison of these models (normalized values) is shown in Figure 2.3.

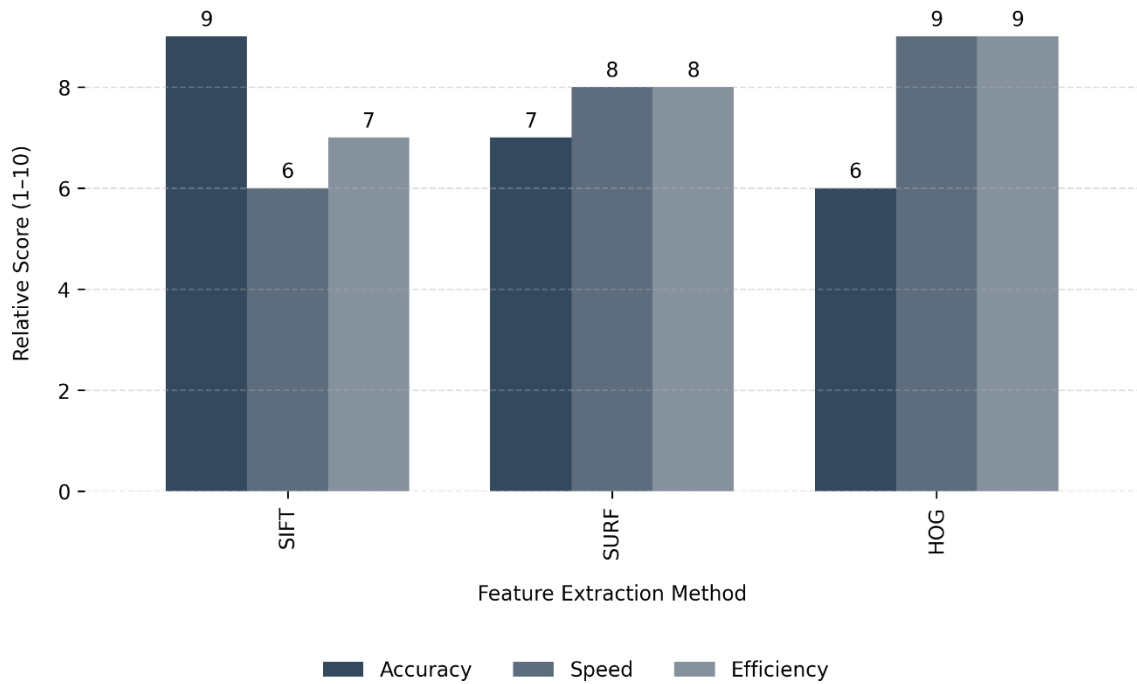


Figure 2.3 Performance comparison of traditional object detection features (normalized values) [Refer Annexure-A1-1]

A histogram of gradients (HOG) descriptor has also been widely applied for object categorization because it defines the features as edge orientations or gradients of an image [24]. In this case, normalization procedures of gamma and color parameters are executed in small images from larger frames to optimize resource usage. A section-by-section scanning is performed by the detection window on the primary image to deduce smaller images that are scalable. Here SVM based models are a favorite. By not implementing a central dependent technique and rather using combination of localization, contrasting and normalization of the smaller and scaled up windows, these surpass wavelet models. For instance, SIFT showed better performance than wavelets but much closer performance when gradient-based detection was used. In contrast, unlike the other approaches of extraction on which deterministic methods have been applied, a salient localized region is defined here through a non-deterministic mechanism based on image intensity profile properties within an image window in order to detect faces with different resolutions. Another major reason for choosing this approach is that it is distinct from its computation

efficiency even under difficult lighting conditions and object positions. The method is efficient computationally even when illumination conditions are varied and objects change their position [25]-[28]. The class differentiation at the feature space is achieved by employing logistic regression, linear discriminant analysis, standard correlation coefficient or even principal component analysis. They may even bank on the interdependence between multiple variables. This means that models fed with features should attain their decisions around the decision boundary [25].

A number of factors such as aspect ratio, color-based detection can be used to determine if an object exists in a frame or not by using supervised classification algorithms. A supervised classifier usually preferred in object detection from video is SVM; it allows learning an optimal hyperplane amongst infinitely many that separate two classes [29]. Based on this distance from hyperplane margin SVM's decides. Image classification tasks have been simplified by use of fast R-CNN because no additional processing step like scanning entire images or extracting regions of interests has been performed. Locating the most discriminative features from raw RGB pixels can be carried out using filters and pooling.

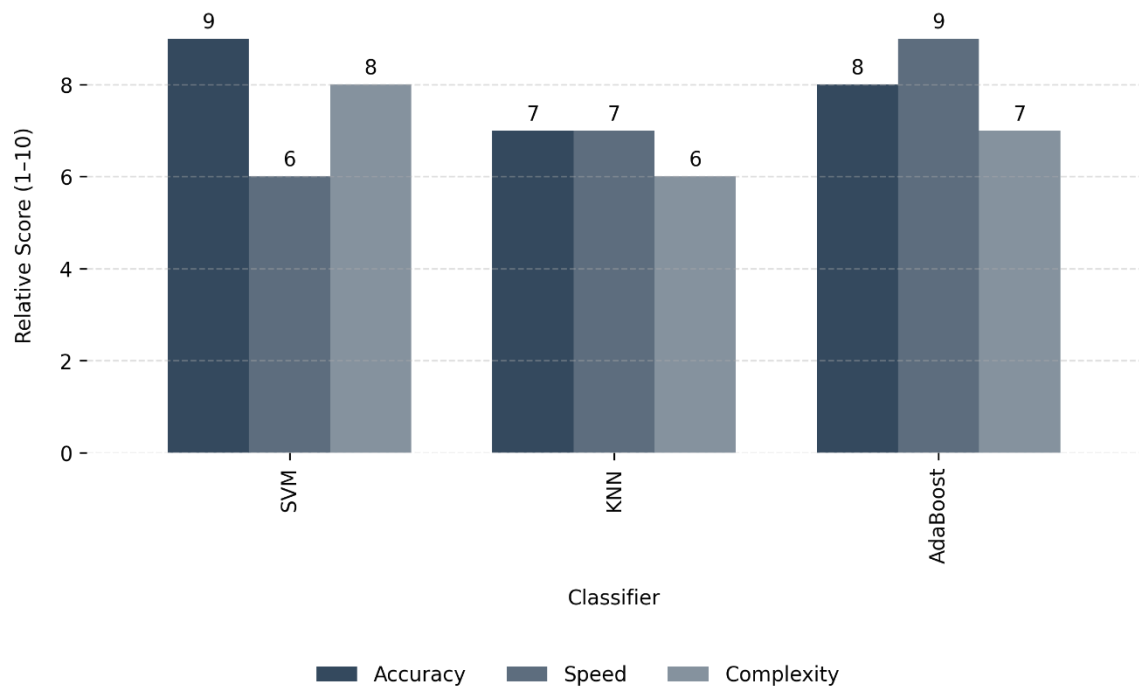


Figure 2.4 Comparison of Machine Learning Classifiers used for vehicle detection (normalized values) [Refer Annexure-A1-2]

Suppressing the errors based on the choice of appropriate thresholds and by neglecting the parameters that are insensitive is achieved through quadratic processing or programming and Hilbert Spaces [29]. For a large dataset, a choice of kernel is important, as well as the

time it takes to run. K-NN [30] [31], a non-parametric method considers the position of the training sample in relation to the class. It calculates an efficient distance between data set and training sample. The samples that are close to k-threshold or from the inbound sample classes are grouped in the same class. Euclidean distance is used when there exist several dimensions. Using the appropriate k-value, the objects are numbered, metrics of similarity are generated or even the distance from the k is calculated for precise classification. But this has limitations such as multi-modal classes where error can be approximated and easily estimated because it approaches Bayes's error but with such a computational burden upon data getting bigger or more complex. AdaBoost algorithm has been widely used in object detection where strong classifiers are obtained through weak ones [32] [33]. At the end of each learning cycle, these classifiers are adjusted by that algorithm which is weaker. Also, though it separates Adaboost from other classifiers in terms of ease of implementation, it allows for faster convergence as well as no need to know preexisting state of weak classifiers and performs very well also. The normalized comparison of these models is depicted in Figure 2.4.

Machine learning models find extensive application in applications that involve huge training dataset. Single stage models (YOLO & SSD) use CCN's and dual stage models (RCNN, Faster RCNN, and SPPNet) usually are generate target box which is followed by classification. The R-CNN [34] shows better performance compared with the earlier conventional models. The algorithm (multi-stage) performs selective search to suggest about 2000 regions for each image [35]. It then crops the region of interest from the proposed region extracting features to form feature vector of a dimension of 4096, making it highly robust. This feature array is used to predict the object in the frame.

While extracting features and subsequent training are tedious, it is possible for the algorithm to have fixed image size leading to making of redundant proposals. The requirement of the image sizes to be fixed is important here and the issue is dealt with through SPP-Net [36]. The algorithm has a SPP which results in a vector of features with non-uniform image sizes. The technique outperforms R-CNN but comes at a cost of time as this involves connected multiple layers. Fast R-CNN has detector to estimate feature-length and determine the region of interest of pooling that does the fixed feature size prediction that contributes to better classification exploiting the bounding boxes of the CNNs [36] [37]. With a single stage processing in the network layers that performs convolution per image, Fast RCNN exhibit fast response in comparison to multiple regions proposed in the earlier method. While this method does selective search, Faster RCNN

computes the features of an image at once and the ROI is proposed by a different network. To improve this stated problem, the full image convolutions in a Faster R-CNN [38] have been adopted and selective search for ROI is avoided through separate network. The Region-Proposal-Layer can operate on the images of multiple size to predict the feature vector which feeds the regression as well as the classifier layers. Even though it has faster response time, there are limitations when it comes to pictures containing extreme scales or even shapes. The comparison of the models in terms of accuracy and inference time is plotted in Figure 2.5.

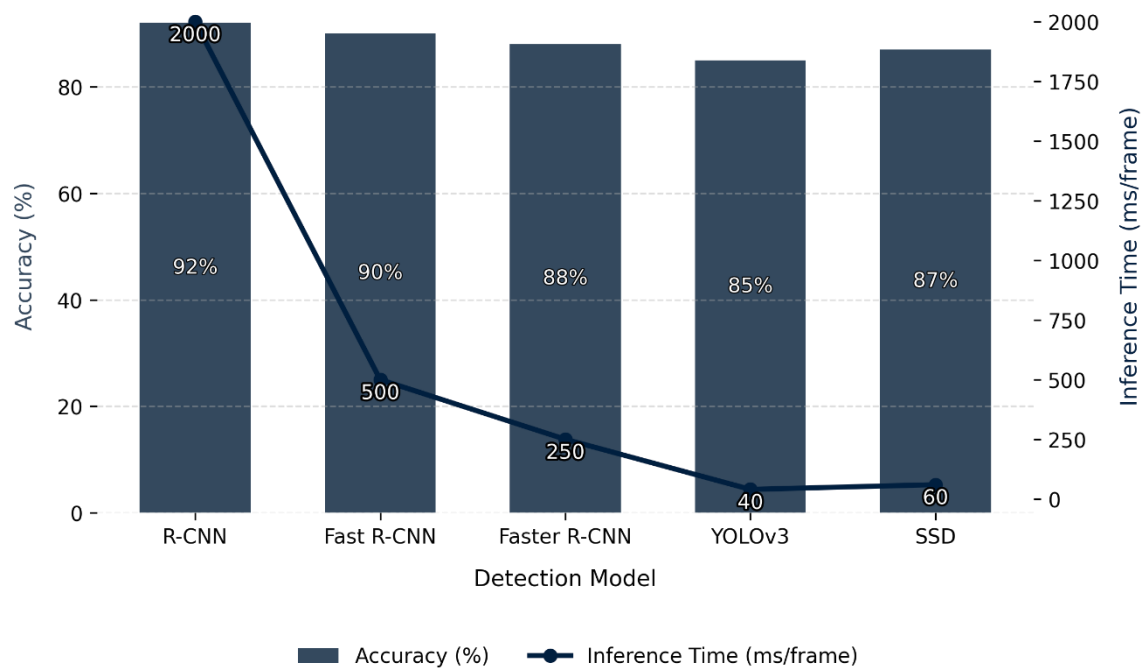


Figure 2.5 Comparison of Deep Learning-based Object detection models for adaptive headlamp applications [Refer Annexure-A1-3]

In applications with real-time requirement where speed takes precedence, region-proposal algorithms like YOLO are effective as they are focused on probable areas rather than the entire image [40]. YOLO splits photos into grids with each cell representing an input proposal. Higher-level iterations such as YOLOv2 and YOLOv3 [40]–[42] are used greatly in video object detection [43]–[46], enjoying better speed and accuracy; however, they still face the challenges of detecting minor objects and tolerating a range of aspect ratios. SSD [47] [48] eliminates such limitations by incorporating region-proposal and classification in a single stage, achieving speeds comparable to Faster R-CNN. It uses anchor boxes and aspect ratios to create bounding boxes and aggregates predictions across different feature maps to support objects of different sizes. Compared to CNNs, SSD does not include intermediate filtering, enabling faster detection with slightly reduced accuracy. For

increased accuracy, SSD uses non-max suppression to aggregate similar bounding boxes and employs hard negative mining to remove false positives due to background noise.

2.3 Vehicle Models for Adaptive Headlamps

Another essential adjustment for adaptive headlamps is the need for adjusting the beam-lobes in accordance with the road curve. To do so modeling of vehicle is an integral part herein. Multiple models of passenger vehicles is depicted in Figure 2.6. Path tracking (Geometric based) [49] corresponds to vehicle position, dimensionality and orientation vectors along its trajectory and look ahead distance with respect to curvature and is independent of the velocity of the vehicle and external factors. The complexity of look-ahead distance is a bottleneck in this model. The challenge is that at steep curves the look ahead estimates the road to straight line skipping the actual path and goes straight to a new point leading to the generation of oscillations at higher speeds. The non-consideration for internal and external forces but rather the position of the automobile and the acceleration with respect to co-ordinates (both the local and the global) makes Kinematic model apt. This method [50]-[53] assumes that there are front wheels that are steerable with front axle being the origin. Controllers [54] [55] to account for both the linear alongside the rotational motion to ensure better stability are designed. Unlike its previous case known as Kinematic model, this one also takes into account tire slip, the coupling and the coefficient of friction [56] [57].

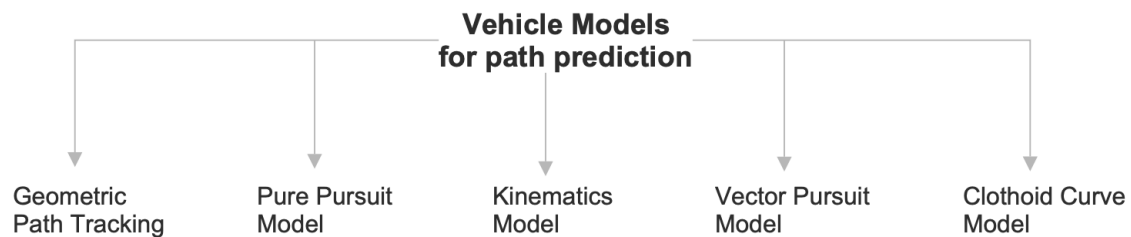


Figure 2.6 Passenger cars Vehicle models suitable for lane predictions

In pure pursuit algorithm [58] in which besides looking ahead; error on direction between vehicles final destination is defined through drawing circular arc from current location up to goal or target point that is given by look ahead. The use of this type [60]-[62] for other applications [59], however, is limited as a headlamp adaptation model since it depends on the length of road ahead. In vector pursuit, which employs coordinates to predict the most suitable position at a given time to reach an endpoint it can be used in adaptive headlamps

[63]; However, this technique has been found to have a great computational overhead. The clothoid curve method can avoid arches and has real time operation with increased performance that is dependable [64]. The tire and soil contact forces coupled with the forces acting on the wheel: lateral and the longitudinal are calibrated in the Dynamic-path tracking model [65] [66] and is complex as it involves multiple factors to be processed and error compensation becomes difficult.

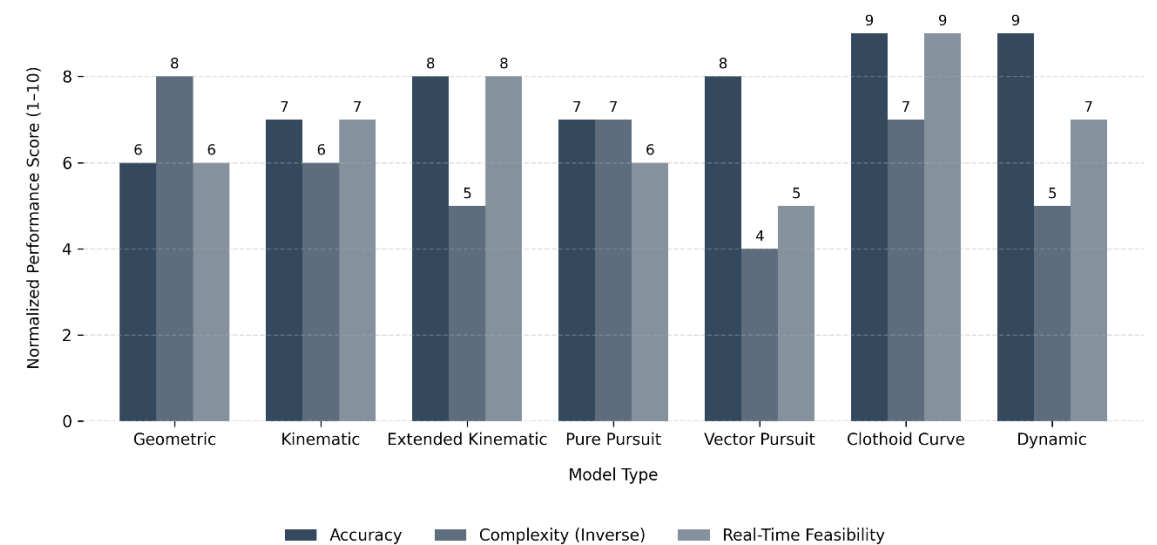


Figure 2.7 Comparative analysis of Vehicle Path Models for Adaptive Headlamp Applications *[Refer Annexure-A1-4]*

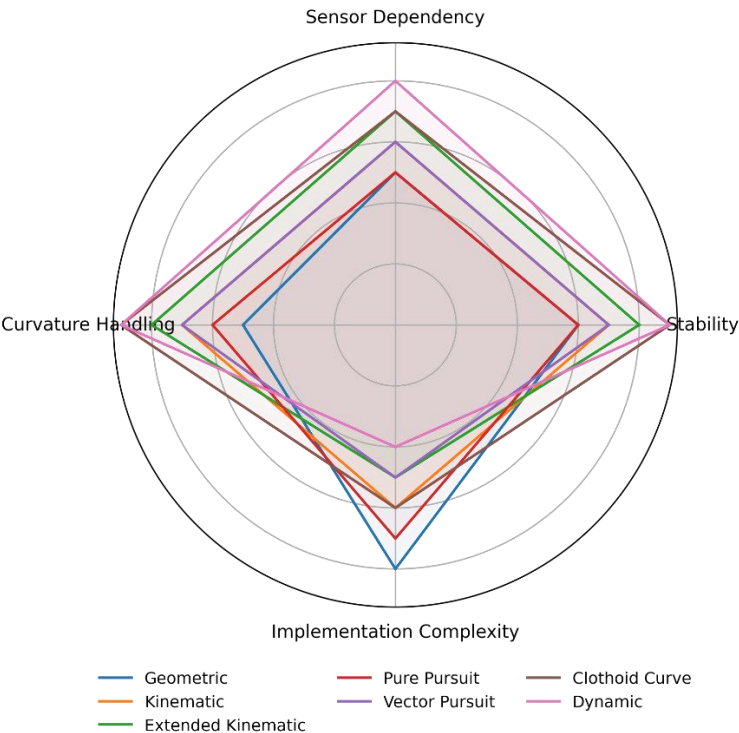


Figure 2.8 Normalized comparison of model stability and parameter dependence *[Refer Annexure-A1-5]*

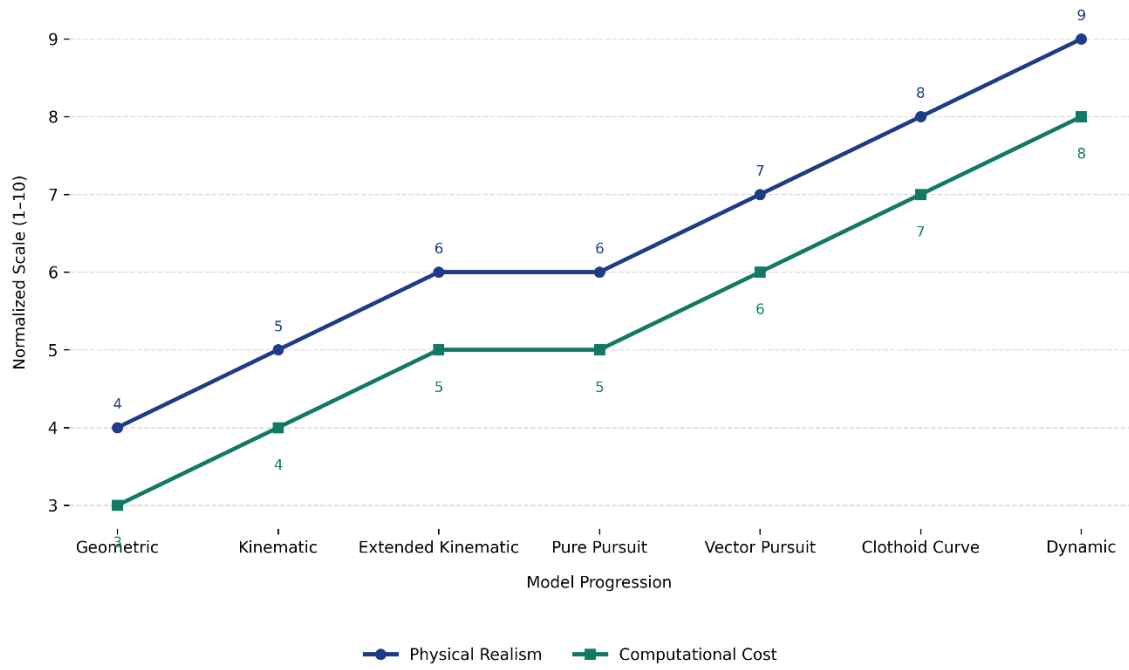


Figure 2.9 Trend of Model evolution from Geometric to Dynamic Formulations
(Normalized representation) [Refer Annexure-A1-6]

Kinematic as well as dynamic model based controllers are prevalent adaptive controllers [67]-[69] in addition to the types dependent on neural networks, PID with high stability at varied driving conditions but with difficult design constraints [70] [71]. This leads to complex development of neural networks based on PID controller, neural network-based adaptive control (NNAC) system [45] [72]. The algorithms optimization and cost overhead reduction by Predictive Controllers. Nonlinear MPC [72] and also extended-kinematic models [45] are among the widely accepted models. Though complex in terms of design, Robust Controllers have the ability to compensate for rapidly changing dynamic conditions [73]-[75]. Normalized comparisons of these models are shown in Figure 2.7 through 2.9 with respect to their usage for adaptive headlamp application.

2.4 Lane detection Models for Adaptive Headlamps

This section analyses modern lane finding algorithms for night driving, to evaluate their efficacy for path prediction and investigates nighttime lane recognition. The aim was to develop a model that would address the limitations of autonomous driving on unmarked roads using deeplabV3+ semantic segmentation for road detection [76]. Its dataset contained 15,000 annotated images specifically made for drivable road detection, which use pixel-level segmentation to predict steering angles with precision in automated vehicles. A comparison analysis also shows it aids in improving steering control in level-5 self-driving cars especially where there are no lane lines or they have faded away. Vehicle and

pedestrian identification algorithms are moving towards CNN's that are strong enough. Traditionally, two-staged architectures require first some vital elements from raw images like low-level representations such as pixel gradients or local image patterns are extracted and then learning methods applied for full scene understanding [77]. HoG [24] stands out by exhibiting high accuracy while reducing computational cost hence making it suitable for autonomous driving. Again, novel solutions as presented in [78] entail an extra middle layer after low-level feature computation is done.

These algorithms find the best combination of features that results in improved accuracy, even if it comes at the cost of increased processing time. In CNN's [79], feature extraction takes place during learning and comprises many layers. Trained filters within each convolutional layer process input images, enabling automatic feature extraction from training data. Although CNNs have acceptable accuracy in pedestrian or object detection, their computation intensive processes lead to improvements such as fast R-CNN [39] and YOLO [41] to reduce computational costs and robust lane detection under difficult conditions respectively. For instance, [80] combines object identification and tracking into a probabilistic framework for real-time lane detection with Robustness even under challenging environments like curved lanes, faded markers, or shifting lanes. The approach is based on vehicle motion models and inertial sensor data which therefore in turn makes it vulnerable when dealing with non-predictable motion patterns of a vehicle. Therefore, [81] provides a model based on fixed lane markings that performs well in very difficult situations both in day light as well as at night fall. Obstacle detection is limited in its ability to identify drivable space due to the presence of invisible objects and indefinable barriers. LIDAR sensors are a viable alternative but their cost effectiveness necessitates semantic segmentation research. To automatically extract local features, CNN can be trained, which has better performance than traditional techniques [82]. But the problem of high computational overhead still remains.

In [83] an alternate approach reduces drivable space estimation into a 1-D graph inference issue using lightweight methods for real-time feature computation and inference with better results on difficult datasets. Consequently, this study employs sensor fusion consisting of LiDAR and camera sensors for the purpose of detecting robust drivable road. Using edge detection and color-based segmentation, this technique [84] generates lane binary images from camera data. Though it demonstrates good performance across different types of urban roads such as: two lane marks; a lane on one side; and a curb on the other side; or roads bordered by curbs on both sides but there is no universal implementation of the model.

The HMRF model [85] magnifies boundary issues and multi-feature learning for enhanced road detection. Improvement in detection by handling noisy data and reducing redundant information with a bootstrapped learning strategy and hierarchical multi-feature segmentation framework. The efficiency of this model has been validated multidimensional across various datasets to affirm its accuracy; it is applicable in complex road environment. In addition, it enhances safe driving-assistance system, detects obstacles and improves real-time drivable area extraction using HMRF framework. However, the performance of the model at large dataset needs to be optimized through coding. This algorithm segments road regions by combining road features and a model using graph-based manifold ranking approach to fit a road model. Its robustness against such adverse elements as water stains, kerbstones etc., makes it less susceptible to violations of road model assumptions. Precision improves when this model is integrated with the feature-based framework [86].

Experimental validation on Kitti-Road against the actual-road datasets demonstrates high performance with AUC values of 0.956 and 0.989 respectively [89]. This approach further integrates these constraints into visual odometry systems for depth factor calculations hence improving unstructured roads detection that can also operate under difficult conditions without failure [87]. To estimate road regions in complicated environments where there are diverse objects that could overlap with roads, one can use vanishing point estimation method on the boundaries surrounding those roads. This strategy quickly computes and is robust to shadows and complicated road surfaces by using boundary alignment from voting points & line-soft-voting with maximum weight. Nevertheless, color variations in images and errors in the estimation of the road boundaries are affecting the accuracy of finding the vanishing point despite its flexibility in handling different interferences on roads. Lane detection & tracking strategy employing digital image processing enhances lane detection accuracy, handling multiple scenarios [88]. In order to avoid collision, this method uses ROI sizes along with IPM algorithm for range determination. Additionally, there were cases where the tracking performed well or poorly as a result of factors such as lane visibility issues (glare, reflectivity, paint deterioration) among others qualitatively analyzed. Continuous lane markings work well whereas double markings reduce tracking precision because of varied tracings.

To overcome some problems faced by conventional networks in spatial detail recovery, two attention mechanisms are employed in this module [89]: Upper-level Prediction Attention along with Upper-level Boundary Attention. Moreover, a top-down refinement process utilized within a decoder network improves segmentation accuracy without

increasing computational cost. As per evaluation conducted over Cityscapes and CamVid datasets improvement concerning efficiency & accuracy was noted. The module is adaptable to different encoder-decoder networks implying that it can be used for optimization and segmentation applications more generally. It works as a separate module and thus it can be used with any segmentation network which has an encoder-decoder structure. Its efficacy is influenced by the segmentation network. Therefore, the network of semantic segmentation for the prediction of the trajectory of the ego-vehicle, with a front-view monocular-camera as input for consecutive images which produces trajectory mask that improves predictions accuracy [90]. Consequently, results from KITTI datasets demonstrate better performance than baseline models. By integrating human intentions on intersection turns and generating additional training data through simulations, accurate trajectory prediction will improve lane-changing prediction performance. The lane detection network utilise SPP & the atrous convolution for semantic segmentation which is performed pixel-wise. The network includes an encoder-decoder for binary segmentation & feature mapping [91]. This was shown in its' experimental results on Tusimple dataset where it outperformed all other architectures that are essential when achieving autonomous driving. In this case Spatial Propagation & Transformation is an end-to-end network fusing image and point cloud data for road detection.

It entails a model-level combination as well as dual-view fusion, which enriches street illustration and applies a data-based combining approach to present the competitive performance on KITTI Road Benchmark. For instance, RNN's joint anisotropic diffusion has limitations that can be improved by combining it with GNN and extending spatial propagation fusion for wider object detection tasks. The proposed SPSTFN network reveals potential but needs fine-tuning on more intuitive and flexible fusing methods. One of these is how lane identification strategies are implemented in autonomous driving through this review. Classic techniques such as HoG and RealBoost are computationally efficient; nonetheless, they are being replaced by CNN-based models due to their automatic feature extraction capabilities. Although CNNs exhibit high accuracy levels, they demand significant processing resources thereby necessitating improvements like rapid R-CNN and YOLO for real-time operations. There is potential for deep learning based semantic segmentation networks. The survey encompasses sensor fusion, model level integration among others aiming at improving road recognition accuracy.

Lane detection has evolved from probabilistic frameworks based on vehicle motion models to models such as HMRF that thrive in unstructured settings. HMRF addresses boundary

difficulties and reveals a high level of strength, notably in complex urban environments. Lane detection accuracy is significantly enhanced by systems using graph-based manifold ranking, vanishing point estimation, or image processing algorithms. A more efficient approach to improve segmentation accuracy without requiring extensive processing resources involves modules like semantic segmentation networks utilizing a top-down refinement method. It is further suggested by the paper that some aspects of specific methods have limited room for application development calling for more studies on it possibly alongside Graph Neural Networks (GNN). The importance of improving fusion techniques in the SPSTFN to make them more intuitive and versatile is underlined. In summary, this survey underscores the shift from traditional to deep learning-based methodologies emphasizing CNNs for enhanced accuracy despite their computational loads. The assessed approaches present progress made in road identification leading to a stronger and more accurate autonomous driving capability; at the same time, they identify areas that need more research and development efforts.

The review, however, suggests that the headlights available are inadequate in terms of ensuring safe driving. The focus area in this case is specific and current solutions cater only for limited test cases without taking into account the different speeds on which a vehicle operates and also on the road conditions. Solutions from the vehicle manufacturers are dependent majorly on the lanes and line markings limiting it to become a universal solution. Therefore, the system should be intelligent enough to anticipate and control headlamps regardless of lane markings. Their universal implementation is still at early stages as they have limited functionality. However, till date, urban limits have been covered by these where sudden turns, multiple crossover lanes must be addressed. Therefore, there has been limited study on how speed affects most of the systems' performance making them perform between a certain range of speed very well which is vital in safety device design leaving room for more research activities. Thirdly, with most automotive embracing camera based IOT systems; there exists large potential to utilize artificial intelligent tools as well as deep learning in designing more dynamic responsive systems.

2.5 Research Gap

- **Need for Enhanced Object Detection and Recognition:** While the system demonstrates good performance in detecting and illuminating objects, there's a limitation in detecting oncoming traffic or adjusting illumination accordingly. Future research could focus on improving object detection capabilities, especially

in dynamic traffic environments, to enhance safety and driver awareness. Develop streamlined models for real-time lane detection in low-light conditions, balancing accuracy and computational efficiency.

- **Optimization for Adaptive Headlamp Designs:** Although the system is deemed suitable for adaptive headlamp designs, there's a need to address the overhead associated with look-ahead direction and complexity. Further research is required to optimize the system for adaptive headlamp configurations, reducing complexity and overhead to improve efficiency and performance in adaptive lighting applications.
- **Robustness to Environmental Variability:** Create algorithms that adapt to diverse conditions (e.g., lighting, weather) without sacrificing accuracy, enhancing model robustness.
- **Latency and Real-time Performance Optimization:** The system lacks discussion on latency, crucial for real-time applications. Research should focus on reducing latency, especially for sudden curvature changes, to enhance real-time responsiveness while maintaining accuracy.
- **Reducing Data Dependency:** Explore methods to minimize reliance on large annotated datasets, such as transfer learning or semi-supervised techniques, while maintaining high performance.

2.6 Objectives of the research

The objectives of the proposed work are as follows:

- To simulate a mathematical model based controller design for headlamp beam adjustment when the vehicle traverses through a curve.
- To propose an algorithm to identify objects on road, oncoming traffic and hence adjust the headlamp beams to reduce troxlers' effect based on comparative analysis of the state-of-the art Deep Learning algorithms for object detection.
- To integrate the beam adjustment across a curve and the deep learning model developed for identifying object/ oncoming traffic detection to achieve better beam lobe adjustments.

- To evaluate the performance of the proposed solution against efficiency, latency and performance in road test cases with state-of-the-art solutions.

2.7 Research Methodology Overview

Objective 1

- Employing vehicle dynamics, the goal centers on creating predictive beam steering techniques through Design of Dynamic Model-Based Controllers.
- Road direction (ψ) and slip angle (β) are integrated into a detailed dynamic vehicle model. Two advanced control strategies are designed: Extended Model Predictive Control (E-MPC), Filtered Feedforward Model Predictive Control (FF-MPC) and Fused Controllers.
- While E-MPC incorporates state and input constraints for optimal control, FF-MPC employs feedforward behavior to estimate road curvature. The Fused Controller combines predictive modeling of road geometry through slip angle and steering-based prediction with real-time environmental perception through object detection.
- To evaluate headlamp alignment precision, responsiveness, and robustness in a range of speed and curvature scenarios, these controllers are simulated over synthetic road profiles.

Objective 2

- To control beam adjustment based on environmental cues, this objective includes vision-based perception modules.
- Lane-Based Prediction of Beams Utilizing Image Processing: A standard computer vision pipeline is employed for cases where the lanes are well-defined.
 - Lane boundaries are detected in low light with techniques such as gamma correction, bilateral filtering, Canny edge detection, and Hough transform.
 - To enhance flexibility in organized scenarios, a steering angle model is constructed to adaptively match light beams with road curvature.
- Object Detection with Deep Learning: To enhance object perception and overcome the Troxler effect, a perception module based on deep learning is introduced.
 - Low-light and Small-object Detection Network (LSDNet) is designed specifically and compared against popular detectors such as RCNN, SSD,

Tiny YOLO, and YOLO. LSDNet demonstrates enhanced robustness and detection ability across various light levels.

- The module facilitates responsive beam adaptation through forward-looking object tracking and scene understanding.

Objective 3

- Lane-based beam control models and object detection are supported by an embedded processing unit. Headlamp beams are dynamically steered by actuators such as servo motors.
 - In response to Steering angle (inferred from curvature estimation)
 - Oncoming traffic or object (inferred from AI inference) for autonomous driving modes, the built-in system dynamically adjusts beam direction and intensity.

Objective 4

- Assessment of Controller Performance: The following performance criteria are applied to benchmark the four predictive controllers:
 - Step response, impulse response, frequency response, and steady-state error
 - Precision of road tracing and corresponding beam deflection loyalty
 - Case study on performance on a two-kilometer synthetic road with segment-specific profiles of curvature
- Comparison of Object Detection Models
 - Current detectors are employed to benchmark the AI models using: Detection accuracy and processing rate (FPS), mAP, F1 score, and recall
 - Change sensitivity to illumination and object scale
 - These systems are tested for the following: robustness, invariant accuracy across varying test conditions, efficiency, and latency.

Table 2.1 Chapter-wise details of Objective Implementation

Objective	Sections in which it is discussed
Development of vehicle dynamics-based predictive beam steering techniques (5 Controller models).	Chapter 3: • 3.2 Dynamic Vehicle Modeling • 3.3 through 3.8 Controller Designs (FPC, RS-MPS, E-MPC, FF-MPC) • 3.10 Fused Controller
Vision based object detection: Object detection models and Lane detection	Chapter 4: • 4.3 Deep Learning Framework for on-road Object detection for Adaptive Headlights: Modified SSD, YOLO V8 based Model, MF Model, LSDNet • 4.7 Lane-Guided Beam Prediction
Integration of embedded systems for beam direction.	Chapter 4: 4.8 Embedded Implementation and Actuation Control: Integrating the multifaceted object identification model and the lane detection model
Comparative analysis and performance evaluation of the perception and control modules.	Chapter 3: • 3.9 Simulation Results of the controllers explained from 3.3 to 3.8-Controller benchmarking • 3.11 Performance analysis of Fuse Controller and its benchmarking against other controllers • 3.5 Controller Benchmarking Chapter 4: • 4.5 Benchmarking of the object detection models discussed in section 4.3 and model tradeoff analysis • 4.8 Lane Detection Evaluation

2.8 Novelty of the Research

This research addresses the challenges of dynamic trajectory tracking and low-visibility object detection by proposing an integrated control-perception framework for adaptive headlamp setting in night-time automobile environments. Two control front prediction strategies are constructed: an Extended Model Predictive Controller (E-MPC) combining full state representation of the vehicle with input and output constraint expression for optimum control in dynamic conditions, and a Filtered Slip-Angle-Based Feedforward Model Predictive Controller (FF-MPC) based on lateral slip and road heading angle

(ψ_{road}) to predict curvature-induced misalignments to pre-steer beams. A split perception pipeline is employed to complement these control measures. A lane detection algorithm is developed for structured road scenes with well-defined lane markings. It approximates curvature and guiding beam direction using conventional computer vision methods such as gamma correction, bilateral filtering, Canny edge detection, and Hough transformations. In contrast, a deep learning model named LSDNet (Low-light and Small-object Detection Network) is developed, designed, and trained to maintain high detection fidelity in unstructured or low-light illuminated conditions. It is able to process scale-variant object situations and low light. LSDNet demonstrates better performance in precision, recall, and inference robustness when evaluated across a range of brightness regimes and compared to the best object detectors. By integrating predictive modeling and context-adaptive visual perception, these features offer a perceptually aware, trajectory-aware headlamp control system that surpasses the state-of-the-art by enabling anticipatory lighting decisions.

CHAPTER 3

OPTIMIZING VEHICLE HEADLAMP ALIGNMENT VIA DYNAMIC MODELING AND DESIGN OF FILTERED PROPORTIONAL CONTROLLER FOR SLIP ANGLE BASED ADJUSTMENTS ^{3,4}

Notations used

$U(t)$	: Longitudinal velocity (m/s)
$V(t)$	: Lateral velocity (m/s)
$\Omega(t)$	: Yaw rate (angular velocity) (rad/s)
θ	: Vehicle heading angle (angular displacement) (rad)
$\beta(t)$	: Body slip angle (rad)
a, b	: Distance from CG to front/rear axle (m)
M	: Vehicle mass (kg)
J	: Yaw moment of inertia (kg m ²)
F_x	: Longitudinal force (N)
F_{yf}, F_{yr}	: Lateral forces at front/rear tires (N)
$C_{\alpha f}, C_{\alpha r}$	: Cornering stiffness (front/rear) (N/rad)
C_A	: Aerodynamic drag coefficient
C_L	: Lift coefficient
C_r	: Rolling resistance coefficient
ρ	: Air density (kg/m ³)
A	: Frontal area of vehicle (m ²)
F_D, F_L	: Aerodynamic drag/lift forces (N)
R	: Wheel radius (m)
ω	: Angular speed of wheel (rad/s)
J_w	: Rotational inertia of wheel (kg/m ²)
T_d	: Driving torque (Nm)
λ	: Wheel slip ratio
g	: Acceleration due to gravity (m/s ²)
μ	: Coefficient of friction (tire-road)
δ	: Steering angle (rad)
k_δ	: Steering feedback gain
k_d	: Headlamp deflection gain
k_h	: Proportional gain mapping slip angle to headlamp angle
k_ψ	: Proportional gain on road heading
F_{bf}, F_{br}	: Braking force front/rear (N)

³ The mathematical modelling based on Vehicle dynamic modelling (Section 3.1 to Section 3.3) is published in Engineered Science Journal. Details: Toney, G., Sethi, G., & Bhargava, C. (2025). Optimal Headlamp Adjustment for Vehicles through Slip Angle and Stiffness Analysis using Dynamic Vehicle Model. *Engineered Science*, 34, 34.

⁴ The various controller designs, its performance descriptions and comparisons (Section 3.4 to 3.11) are published in the Journal of Robotics and Control. Details: Toney, G., Sethi, G., Bhargava, C., Vaz, A. C., & Hegde, N. T. (2025). Sensor Fusion and Predictive Control for Adaptive Vehicle Headlamp Alignment: A Comparative Analysis. *Journal of Robotics and Control (JRC)*, 6(5), 2166-2183.

F_b	: Total braking force (N)
F_{zf}, F_{zr}	: Normal load front/rear (N)
h	: Height of CG above ground (m)
L	: Wheelbase (m)
a_x, a_y	: Longitudinal/lateral acceleration (m/s^2)
$\theta_{HL}(t)$	: Headlamp deflection angle (rad)
τ	: Time constant of the headlamp actuator (s)
T_s	: Sampling time interval (s)
α	: Discretization smoothing coefficient
t_r	: Rise time (s)
t_s	: Settling time (s)
ω_c	: Cutoff frequency (rad/s)
θ_{max}	: Max allowable headlamp deflection (rad)
β_{max}	: Max expected slip angle (rad)
α	: Filter coefficient ($0 < \alpha < 1$)
ψ_{road}	: Road heading angle (rad)
γ	: Fusion weighing factor ($0 < \gamma < 1$)

3.1 Introduction

This chapter introduces the formulation and design of a dynamic car model to estimate the body slip angle in real time and leverage the value for controlling the headlamp direction. In typical systems, headlamps track steering input as representing the path of the vehicle. But under dynamic maneuvers—e.g., hard cornering, emergency maneuvers, or low-traction road surfaces—the actual vehicle direction of motion will differ from the desired steering direction because of lateral slip. Such a difference can compromise safety by impairing vision. A severe slip angle evaluation is built into the new model to mitigate against this, and the direction of the headlight beam is then altered proportionally.

A set of nonlinear dynamic equations modeling yaw, lateral, and longitudinal movement underlies the model. The equations are linked to tire force models that accurately model the actual vehicle response using steering angles, cornering stiffness, and slip ratios. For further enhancing the model, some more factors are included, which are rolling resistance, aerodynamic drag and lift, and dynamic load transfer in braking and acceleration. The yaw rate and longitudinal and lateral velocities are then computed in real time based on each component. The slip angle, which is the arctangent of the ratio of lateral to longitudinal velocity, is computed by these quantities as inputs. A proportional control law takes the calculated slip angle as an input and uses it to compute the needed headlight deflection.

This approach improves night driving safety by aligning the orientation of the headlamp with the actual route of the vehicle, especially during dynamic maneuvers. Besides being

used in adaptive headlamps, the approach forms a good basis for being combined with autonomous vehicle control policies and intelligent driver-assistance systems (ADAS), which require a high degree of precision awareness of vehicle status.

The Filtered Proportional Controller for headlight control based on slip angle is presented in the latter part of the chapter. For use while driving under dynamically changing circumstances, this is the optimal strategy for adaptive headlamp activation. Mechanical inertia and latency in headlamp motors generally render them less than ideal for sudden or instantaneous commands. A filtered P-controller's low-pass filtering element introduces a first-order lag which closely simulates actuator dynamics, creating smoother transitions than raw proportional control can, leading to oscillation or overshooting. Moreover, sensor noise or momentary fluctuations in the motion of the vehicle might produce minute changes in the body slip angle. The filtering mechanism effectively damps out the high-frequency noise to avoid unwanted flicker or jitter in the light response. In an implementation context, the structure of the controller is very deployable on embedded platforms like microcontrollers or ECUs due to it being computation-efficient with only two movable parameters: the gain and the time constant.

With gradual and organic beam changes instead of jarring and distracting ones, the filtered response also provides the driver with a smoother visual experience. This is in alignment with user comfort expectations and automotive industry safety norms. Notably, the control philosophy is also in line with common automotive industry practice, whereby filtered or blended control laws are common in active systems to achieve robust performance in real-world scenarios. Generally, filtered P-control provides the best tradeoff of responsiveness, smoothness, robustness, and integratability and is especially well-suited for adaptive headlamp systems controlled by slip-angle-based dynamics.

Additionally, this chapter presents and compares the four models along with the model designs. All of these models are designed analytically and validated using real-world situations. Proper headlamp tracking, smoothness of control, and adaptability to varying vehicle dynamics are the performance measures based on which they are compared. A comprehensive case study is conducted using a 2 km artificial road divided into 10 distinct sections, each of which reproduces a different curvature and maneuvering condition, in order to further assess their practical effectiveness. The segment-wise analysis provides significant details regarding the strengths and weaknesses of every control method in diverse driving conditions.

3.1.1 Structure of the Chapter

The chapter is organized to systematically progress from basic principles to advanced controller architectures and their comparison. Section 3.2 discusses the role of slip angle in vehicle orientation and its impact on headlight alignment. Section 3.3 outlines mathematical modeling of vehicle dynamics, such as longitudinal and lateral motion, tire force modeling, braking effects, and aerodynamic effects, leading to a unified formulation for headlight deflection based on slip angle. Section 3.4 outlines design and implementation of a Filtered Proportional Controller, including its continuous and discrete-time forms, stability issues, actuator saturation, and parameter tuning. Section 3.5 provides a Raw Full-State Model Predictive Control (RS-MPC) model which incorporates road direction directly.

Section 3.6 explains an Extended Model Predictive Control (E-MPC) architecture that includes slip dynamics in the predictive scheme. Section 3.7 presents a Feedforward-Enhanced Model Predictive Controller (FF-MPC) making use of estimated road heading and slip angle for pre-emptive headlamp control. Section 3.8 provides an in-depth description of all the controller architectures. Section 3.9 presents a comprehensive performance evaluation of the predictive control models using trajectory-aligned scenarios and segment-by-segment analysis. Section 3.10 shows a predictive controller that combines sensor fusion with dynamic modeling for enhanced flexibility. Section 3.11 finally shows the performance results of the fused controller and a comparative study against the models previously developed.

3.2 Assessing the slip angle to ascertain the variance between the vehicle's positioning and the headlamp's placement

Slip angle, the variance between a vehicle's actual direction of movement and its intended course, is crucial for understanding tire dynamics [92]. Illustrated in Figure 1, slip angle (α) indicates this discrepancy. Slip angle as the distinction between actual and desired trajectories [93]. Even when a wheel is turned during a turn, the vehicle's body may not align with the intended path, affecting headlamp illumination. This disparity can impact driving safety, as the headlamp may not accurately illuminate the direction of travel. The slip angle serves as a reliable metric for assessing a vehicle's deviation from its intended path and adjusting headlamp direction accordingly [94]. Whether a vehicle is executing a standard maneuver, oversteering, or understeering during a turn, the body slip angle

consistently reflects the disparity between actual and desired trajectories. In designing the headlight adjustment model, this discrepancy was considered to improve road lighting when turning. Slip angle is a reliable predictor of vehicle trajectory, with prediction errors as low as 0.3 m [95]. It reflects the discrepancy between the velocity vector's direction and the heading angle, crucial for determining vehicle path. Traditional slip angle calculation methods [96] [97] [98] [99] include direct integration and linear observer estimation techniques. These approaches effectively calculate slip angle, aiding in predicting deviations between actual and intended paths, facilitating headlamp adjustment.

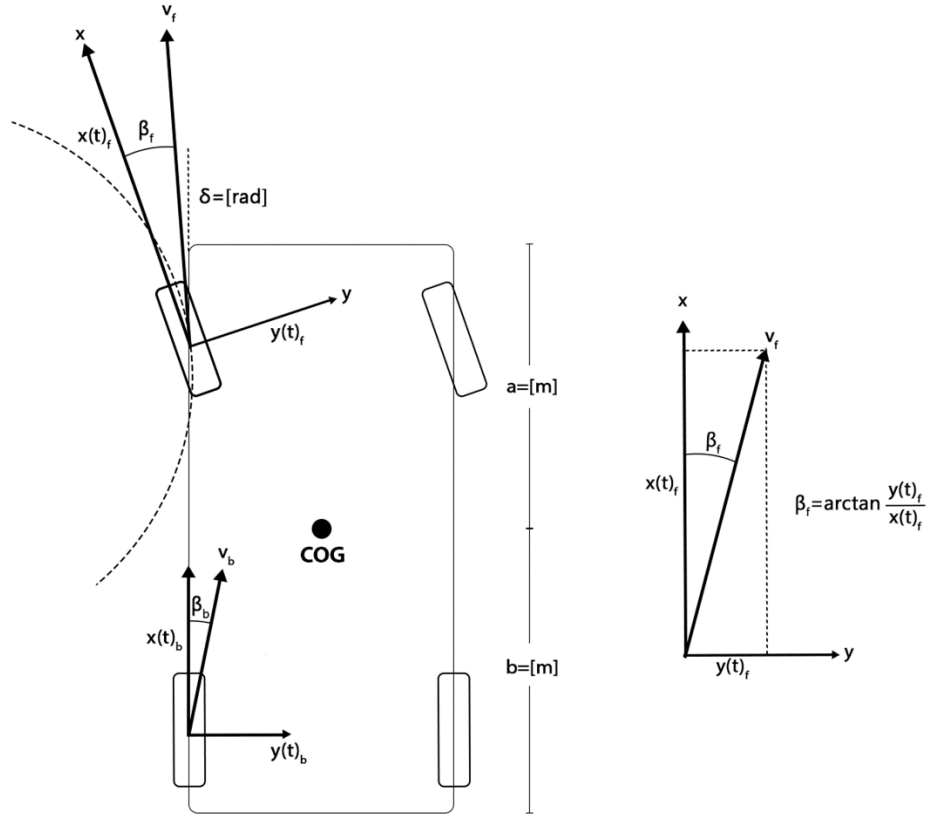


Fig 3.1 Front and rear wheel slip angle as vehicle goes through a cornering where β_f is the front slip and β_r the rear slip. If $\beta_f > \beta_r$, car is experiencing understeer, when $\beta_f < \beta_r$, car is oversteering and $\beta_f = \beta_r$ is a neutral condition. [94]

3.3 Mathematical model for assessing body slip angle and adjusting headlamp direction [94]

The purpose of this model is to calculate the vehicle body slip angle $\beta(t)$ dynamically and apply it for proportional modification of the headlamp deflection angle $\theta_{HL}(t)$. The system improves the driver's visibility while making turns or dynamic maneuvers by orienting the headlamp with the vehicle's actual path, and not merely its direction. The model architecture proposed in this simulates the overall vehicle behavior for body slip angle and

then headlamp orientation estimation in real time. Equations 3.1 to 3.3 are the fundamental motion model describing longitudinal, lateral, and yaw motions of the vehicle in terms of inertial forces, wheel actuations, and aerodynamic drag.

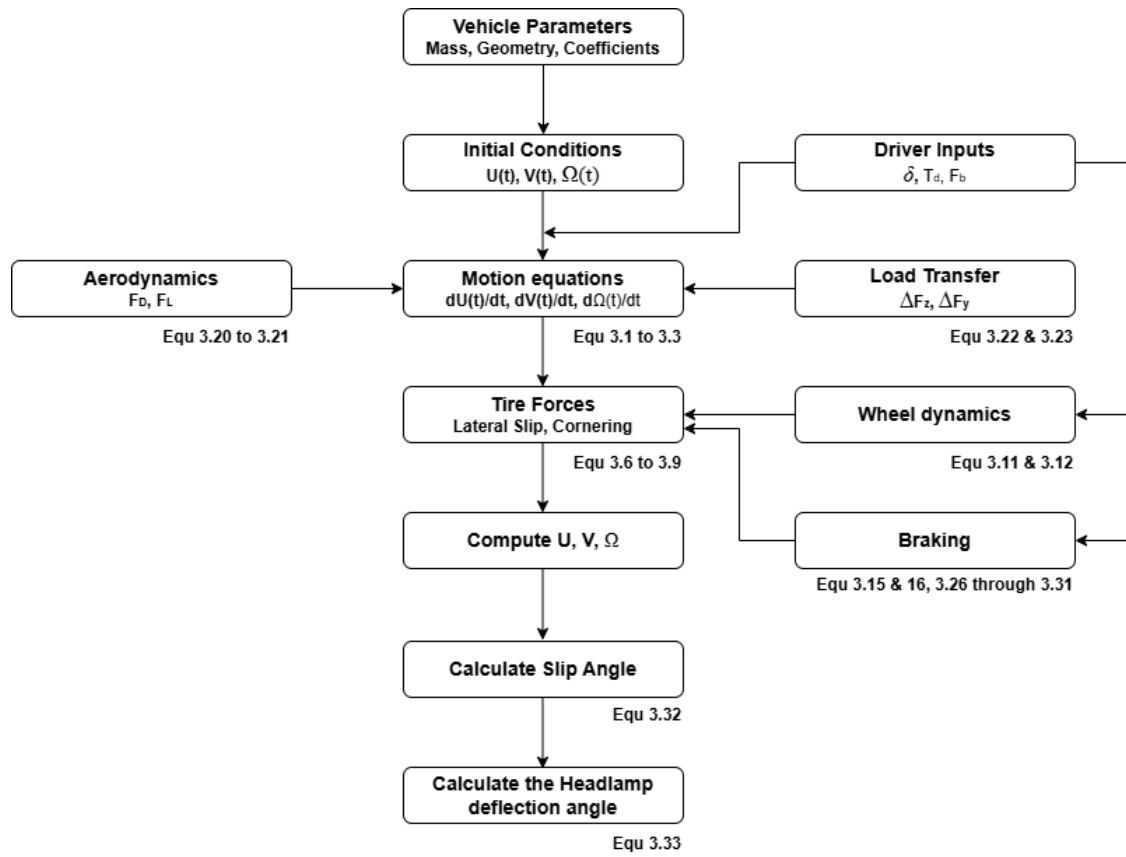


Fig 3.2 The sequences in the slip angle-based headlamp deflection adjustment

These are acted upon directly by tire forces, which are modeled in Equations 3.6 and 3.7 using linear approximations to cornering stiffness and lateral slip and are solved for the system using Newtonian equations in Equations 3.8) and 3.9. The dynamics of the slip angle of the body, a critical parameter that defines the angular difference between vehicle direction of travel and heading, are dynamically modeled in Equation 3.4 and geometrically in Equation 3.5. Equations 3.10 through 3.12 model steering feedback, wheel rotation, and traction forces, all significant in order to model real-world phenomena influencing vehicle motion.

Longitudinal acceleration is governed by Equations 3.13 and 3.14, including rolling resistance and propulsion. Static and dynamically load-adjusted braking force allocation is addressed in Equations 3.15 to 3.16 and elaborated again in Equations 3.26 to 3.31, including braking load transfer. Aerodynamic effects and lateral load redistribution due to cornering are addressed in Equations 3.20 to 3.23. The margins of stability, cornering

capacity, and understeer characteristic are imbedded in Equations 3.17 to 3.25. The overall outcome of solving these equations gives accurate real-time values for longitudinal and lateral velocities, which are used in Equation 3.32 to compute the slip angle. Finally, Equation 3.33 applies a proportional control method to transform the slip angle into headlamp deflection in such a way that the beam is dynamically aligned with the actual path of the vehicle.

3.3.1 Vehicle Dynamics (Longitudinal, Lateral and Yaw based)

Using Newtonian mechanisms, velocity and yaw rates are modelled using equations 3.1 through equation 3.3.

$$\dot{U}(t) = V_f(t) \Omega(t) + \frac{1}{M} [U_f(t) \cos(\theta) - V_f(t) \sin(\theta)] + U_b(t) - C_a U_f^2(t) \quad 3.1$$

This equation represents the development of the longitudinal velocity of the vehicle. The expression $V_f(t) \Omega(t)$ is the contribution of lateral velocity to longitudinal motion, and $U_b(t)$ and $-C_a U_f^2(t)$ are for forces like braking and aerodynamic drag.

$$\dot{V}(t) = -U_f(t) \Omega(t) + \frac{1}{M} [U_f(t) \sin(\theta) - V_f(t) \cos(\theta)] + V_b(t) \quad 3.2$$

This equation controls the lateral velocity of the vehicle. The expression $-U_f(t) \Omega(t)$ is the coupling between longitudinal velocity & yaw rate and $V_b(t)$ is the external lateral forces.

$$\dot{\Omega}(t) = \frac{1}{J} [a U_f(t) \sin(\theta) - V_f(t) \cos(\theta)] - b V_b(t) \quad 3.3$$

This represents the yaw rate dynamics as a result of the axle forces on the vehicles.

3.3.2 Dynamics of Slippage

The crucial rate of change of the body slip angle that majorly influences the vehicle stability is calculated using

$$\dot{\beta}(t) = \frac{1}{M} (F_{yf} + F_{yr}) - \Omega(t) \quad 3.4$$

The lateral tire forces at the front and rear tires, as well as the yaw rate contribute to it.

$$\beta(t) = \tan^{-1}\left(\frac{V(t)}{U(t)}\right) \quad 3.5$$

which is the (kinematic) estimate of the body slip angle using the lateral and longitudinal velocities.

3.3.3 Tire Force Modelling

The lateral tire forces are represented as:

$$F_{yf} = C_{\alpha f}(\delta - \beta - \frac{a\Omega}{U}) \quad 3.6$$

These equations express the lateral forces on the front and rear tires in terms of yaw rate Ω , steering angle δ , and slip angle β .

$$F_{yr} = C_{\alpha r}(-\beta + \frac{b\Omega}{U}) \quad 3.7$$

The entire lateral motion is expressed by this equation, which is the sum of the two lateral forces on the tires.

$$MV = F_{yf} + F_{yr} \quad 3.8$$

This formula shows how the lateral forces influence the yaw acceleration of the vehicle.

$$J\dot{\Omega} = aF_{yf} - bF_{yr} \quad 3.9$$

3.3.4 Steering and Wheel Dynamics

The steering dynamics are given by:

$$\delta(t) = \delta_{cmd} - k_{\delta}\Omega(t) \quad 3.10$$

This equation represents the feedback-corrected steering angle, where k_{δ} is the steering feedback gain.

$$\lambda = \frac{R\omega - U}{\max(R\omega - U)} \quad 3.11$$

This equation computes the slip ratio of the wheel as a function of the wheel's angular velocity ω and the longitudinal vehicle velocity.

$$J_w\dot{\omega} = T_d - RF_x \quad 3.12$$

This is the rotational dynamics of the wheels, where T_d is the driving torque and R is the wheel radius.

3.3.5 Longitudinal Motion and Rolling Resistance

$$\dot{U} = \frac{F_x - R_d}{M} \quad 3.13$$

The forward acceleration which is resulted from the net longitudinal force is deduced through this equation.

$$R_d = C_r M_g \quad 3.14$$

The rolling resistance which is defined as a function of the weight of the vehicle is shown.

3.3.6 Brake Force Distribution

The distribution of the braking force between the front and rear wheels is represented as,

$$F_{bf} = \frac{b_r}{b_f + b_r} F_b \quad 3.15$$

$$F_{br} = \frac{b_f}{b_f + b_r} F_b \quad 3.16$$

Here the braking force (distribution) is modeled using the vehicle geometry.

3.3.7 Stability and Handling Parameters

$$\left| \frac{C_{\alpha r}b - C_{\alpha f}a}{MU^2} \right| < 1 \quad 3.17$$

Represents the stability criterion for the vehicle.

$$K_u = \frac{C_{\alpha r}b - C_{\alpha f}a}{Mg} \quad 3.18$$

Which is the understeer gradient and it is a reference to the handling characteristics of the vehicle.

$$a_{max} = \mu g \quad 3.19$$

Which represents the acceleration (maximum possible).

3.3.8 Aerodynamics and Load Transfer

The aerodynamic drag force is given by

$$F_D = \frac{1}{2} \rho C_D A U^2 \quad 3.20$$

The Aerodynamic lift force using,

$$F_L = \frac{1}{2} \rho C_L A U^2 \quad 3.21$$

The load transfer(vertical) during acceleration is calculated as

$$\Delta F_z = \frac{Mha_x}{L} \quad 3.22$$

The load transfer(lateral) during cornering is calculated as

$$\Delta F_y = \frac{Mha_y}{L} \quad 3.23$$

3.3.9 Cornering Limits and Critical Speed

The maximum safe cornering speed is defined as

$$U_{max} = \sqrt{\frac{\mu g L}{K_u}} \quad 3.24$$

The critical speed to ensure that the vehicle does not lose control is given by

$$U_c = \sqrt{\frac{J}{M} \cdot \frac{C_{\alpha f}a - C_{\alpha r}b}{C_{\alpha f}C_{\alpha r}}} \quad 3.25$$

3.3.10 Dynamic Load Transfer for Braking

During breaking,

a. The front wheel normal load shift is given by,

$$F_{zf} = \frac{b_r}{L} Mg + \frac{h}{L} Ma_x \quad 3.26$$

b. The rear wheel normal load shift is given by,

$$F_{zr} = \frac{b_f}{L} Mg + \frac{h}{L} Ma_x \quad 3.27$$

The front wheels braking force is given by

$$F_{bf} = \mu \cdot F_{zf} \quad 3.28$$

And at the rear wheels,

$$F_{br} = \mu \cdot F_{zr} \quad 3.29$$

The total braking force (a function of the weight of the vehicle) is given by

$$F_b = F_{bf} + F_{br} = \mu Mg \quad 3.30$$

The distribution of this braking force (between the front and the rear wheels) is given by

$$F_{bf} = \frac{F_{zf}}{F_{zf} + F_{zr}} F_b \quad 3.31$$

3.3.11 Slip Angle and Headlamp Logic

Finally, the Slip angle is computed as

$$\beta(t) = \tan^{-1}\left(\frac{v(t)}{u(t)}\right) \quad 3.32$$

The Headlamp deflection adjustment based on the slip angle is given by

$$\theta_{HL}(t) = k_d \cdot \beta(t) \quad 3.33$$

3.4 Design of the Filtered Proportional Controller [107]

The dynamic model formulated in the previous section addresses the longitudinal, side, and yaw vehicle dynamics, tire force response, load transfer, traction, aerodynamic effects, and braking behavior. These equations together allow the calculation of the slip angle $\beta(t)$ of the vehicle, which is the angle between the vehicle direction and heading. After the $\beta(t)$ is estimated, it is used as an input to a control system that adjusts the headlamp deflection angle $\theta_{HL}(t)$. In this section, the design of a filtered proportional controller is given to map the estimated slip angle into a headlamp deflection order. The function of the controller is to match the headlamp beam direction to the true be more than the desired steering angle. This necessitates a control law that:

- Reacts in proportion to the slip angle,
- Filters high-frequency fluctuations to avoid actuator jitter,
- Imitates the inertial behavior of the physical headlamp mechanism.

Hence, a first-order low-pass filtered proportional controller is chosen.

3.4.1 Continuous-Time Control Law

The controller's core idea is to filter the slip angle through a first-order system that exhibits physical smoothness and does not produce sudden changes in the output.

The control law is:

$$\dot{\theta}_{HL}(t) = -\frac{1}{\tau} \theta_{HL}(t) + \frac{k_h}{\tau} \beta(t) \quad 3.34$$

This differential equation is based on the generic first-order linear system:

$$\tau \dot{\theta}_{HL}(t) + \theta_{HL}(t) = k_h \beta(t) \quad 3.35$$

3.4.2 Transfer Function Representation

Applying the Laplace transform to Equation (35) with zero initial conditions, we have the following transfer function:

$$H(s) = \frac{\theta_{HL}(s)}{\beta(s)} = \frac{k_h}{\tau s + 1} \quad 3.36$$

This transfer function describes the controller as a low-pass single-pole filter. It specifies the response of the headlamp deflection angle to the slip angle variation.

3.4.3. Discrete-Time Implementation

For real-time implementation on a digital control platform, the system is discretized by backward Euler:

$$\theta_{HL}[k] = \alpha \cdot \theta_{HL}[k-1] + (1 - \alpha) \cdot k_h \beta[k] \quad 3.37$$

Where,

$$\alpha = \frac{\tau}{\tau + T_s} \quad 3.38$$

3.4.4 Time and Frequency Domain Behavior

The controller exhibits the following characteristics:

- Rise Time: $t_r \approx 2.2\tau$

The time constant τ decides how fast the system responds to input changes. If τ is small, the rise time is small, and the system responds fast. A greater value causes a slower response.

- Settling Time: $t_s \approx 4\tau$

This indicates how rapidly the system settles following a change. A lower τ would result in quicker settling, while a larger value would make the system settle more slowly.

- Bandwidth: $\omega_c = \frac{1}{\tau}$

This means that bandwidth is inversely related to time constant τ . For fast response, τ is small and the system can respond to inputs of higher frequency (greater bandwidth). Otherwise, the system can only respond to lower-frequency inputs (lower bandwidth).

3.4.5. Stability and Causality

The continuous system is stable for any $\tau > 0$. The discrete system is stable for $\alpha \in (0,1)$, providing robust implementation. The system is continuous in theory, since the vehicle dynamics are described continuously. The system is discrete in practice, however, since it must be realized on a real-time digital system. The stability of the discrete system is guaranteed by the condition α belongs to $(0,1)$, ensuring strong implementation in a computerized setting.

3.4.6. Actuator Constraints and Saturation

The actuator output saturation is laid so as to protect the actuator,

$$\theta_{HL}(t) \in [-\theta_{max}, \theta_{max}] \quad 3.39$$

3.4.7. Integrating with the Vehicle Dynamics Model

Slip angle is computed using (32). This is input to the controller (Eqs. 34–39) to produce $\theta_{HL}(t)$, finishing the closed-loop headlamp deflection system.

3.4.8 Parameter tuning

The performance of the filtered proportional controller in slip-angle-based headlamp control is greatly sensitive to the proper choice of gain k_h and time constant τ . The tuning procedure is designed to strike a compromise between responsiveness, smoothness, and actuator feasibility, in such a way that the headlamp response stays intuitive and physically bounded under dynamic vehicle maneuvers.

This section describes the motivation for parameter tuning, the function of every tuning component, and how they affect the total system behavior. The tuning activity is based on both analytical necessity and practical requirements for implementation, the compromise between theoretical optimality and physical limits.

3.4.8.1. Proportional Gain k_h

The gain k_h controls the headlamp deflection sensitivity to variation in estimated slip angle. It specifies the static mapping from the input slip angle $\beta(t)$ to desired output deflection $\theta_{HL}(t)$.

A convenient choice for initial gain selection is:

$$k_h = \frac{\theta_{max}}{\beta_{max}}$$

3.40

Here: θ_{max} is the largest permissible headlamp deflection (generally 10° to 15°), and β_{max} is the anticipated maximum slip angle (typically 8° to 12°) under hard driving.

This prevents the controller from exhausting the actuator's range to avoid going past mechanical boundaries. Tuning k_h serves to position the physical behavior so that the responsiveness desired matches it of the system.

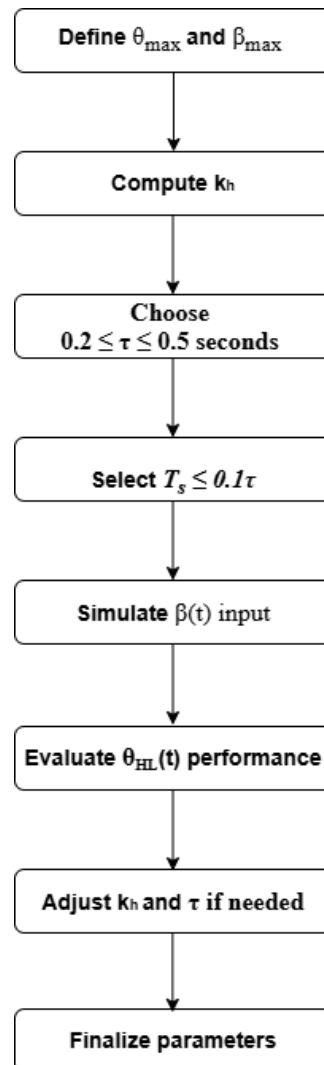


Fig 3.3 The process of parameter tuning

3.4.8.2. Filter Time Constant τ

The time constant τ controls how quickly the headlamp reacts to slip angle changes. A lower τ provides quicker response but can cause unwanted jitter or actuator wear. A higher τ creates smoother transitions but causes lag.

A value within the range:

$$0.2 \leq \tau \leq 0.5 \text{ seconds} \quad 3.41$$

is suggested on the basis of actuator bandwidth and vehicle dynamics. This parameter mimics the inertial response of real actuators. Adequate tuning of τ is such that headlamp adaptations become visually acceptable and mechanically tolerable.

3.4.8.3. Discretization Issues

When applied digitally, the selection of sampling period T_s also influences controller actions. To prevent aliasing and maintain stability:

$$T_s \leq 0.1\tau \quad 3.42$$

should be preserved. This guarantees that the discrete-time version adequately reproduces the continuous-time controller response without any temporal distortion or delay.

3.5 Raw Full-State MPC with Injection of Direct Road Heading

With the road heading angle $\psi_{road}(k)$ as the reference path to the headlamp deflection angle $\theta_{HL}(k)$, a Model Predictive Control structure is employed in this design. The model aims to point the beam toward the path planner projected curvature while refraining from dynamic state input like slip angle β .

The definition of the optimization problem is given by,

$$\min_{\theta_{HL}} \mathcal{L} = \sum_{k=0}^N [(\theta_{HL}[k] - \psi_{road}[k])^2 + \lambda \cdot (\Delta\theta_{HL}[k])^2] \quad 3.43$$

Here, the rate at which the headlamp angle changes is penalized by,

$$\Delta\theta_{HL}[k] = \theta_{HL}[k] - \theta_{HL}[k-1] \quad 3.44$$

The headlamp deflections and its rate is constrained by,

$$\theta_{HL}[k] \in [-\theta_{max}, \theta_{max}], \quad |\Delta\theta_{HL}[k]| \leq \delta_{max} \quad 3.45$$

Although it is a simple structure, this control method may be vulnerable to prediction error in $\psi_{road}(k)$ or curvature noise. Its lack of feedback from vehicle dynamics (e.g., sideslip

or yaw) reduces its robustness against lateral disturbances, which emphasizes the need for a more stable, compensated MPC method discussed in the following section.

3.6 Extended Model Predictive Control (MPC)

Early tests with a Filtered Proportional Controller (FPC), in which headlamp deflection was proportionally related to the slip angle estimate, showed a number of shortcomings. Although the FPC provided smooth actuator behavior, it did not have anticipatory properties and showed steady-state errors consistently, particularly during ramp-like direction changes. The purely reactive nature of the controller and lack of consideration of future trajectory led to poor tracking in curves, making it unsuitable for real-world driving scenarios. In order to resolve these issues, an Extended Model Predictive Control (MPC) strategy was implemented. By modeling the vehicle dynamics in a predictive optimization scheme, the extended MPC provides improved control of system constraints, enhanced transient response, and the opportunity to involve extra vehicle states. This framework provided a basis for extending further via road geometry compensation, which resulted in the creation of the Feedforward-Enhanced MPC.

3.6.1 Dynamic Vehicle Model and Slip Angle Estimation

The continuous-time state space model for the vehicle's lateral dynamics is given by

$$\begin{bmatrix} \dot{\beta} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} -\frac{C_{\alpha f} + C_{\alpha r}}{MV} & -1 - \frac{aC_{\alpha f} - bC_{\alpha r}}{MV^2} \\ -\frac{aC_{\alpha f} - bC_{\alpha r}}{J} & -\frac{a^2C_{\alpha f} - b^2C_{\alpha r}}{JV} \end{bmatrix} \begin{bmatrix} \beta \\ r \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha f}}{MV} \\ \frac{aC_{\alpha f}}{J} \end{bmatrix} \delta \quad 3.46$$

Using Euler's approximation, the continuous controller model is discretized as,

$$X[k+1] = A_d x[k] + B_d \delta[k] \quad 3.47$$

Where $X[k] = [\beta[k] \ r[k]]^T$ and $A_d = I + T_s A_c, B_d = T_s B_c$

3.6.2. Control Architecture

In contrast to conventional vehicle control systems, which control β , our method takes β to be estimated in real-time and utilizes it as a reference to produce a smooth headlamp deflection command θ_{HL} . This simplifies the problem to a tracking and filtering problem for θ_{HL} . The process flow is shown in Figure 3.4.

Assumptions

- The slip angle β is estimated from the dynamic model or an observer.
- The actuator (headlamp motor) has physical deflection limits: $|\theta_{HL}| \leq \theta_{max}$.

- Smooth θ_{HL} transitions are wanted to avoid actuator jitter.

A first-order lag is used to produce θ_{HL} using β ,

$$\theta_{HL}[k + 1] = \alpha \cdot \theta_{HL}[k] + (1 - \alpha) \cdot k_h \cdot \beta[k] \quad 3.48$$

The actuator constraints are incorporated through,

$$\theta_{HL}[k + 1] = \min(\theta_{max}, \max(-\theta_{max}, \theta_{HL}[k + 1])) \quad 3.49$$

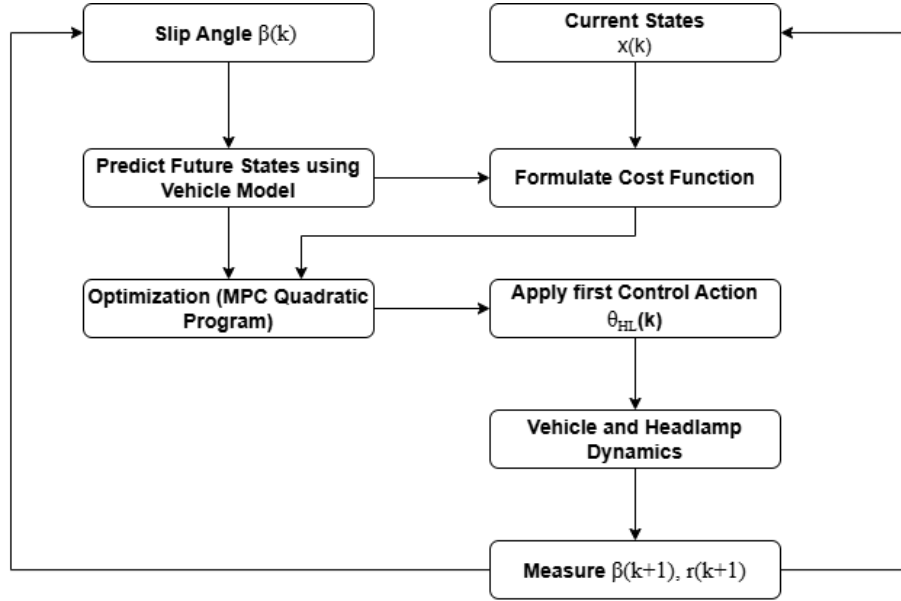


Fig 3.4 The process flow of the E-MPC

Extended MPC allows constrained predictive control solely through slip angle dynamics. Although efficient in actuator response smoothing and beam direction stabilizing, being purely reactive diminishes anticipatory performance.

3.7 Design of the proposed Feed Forward-Enhanced Model Predictive Controller (FF-MPC)

This part outlines the modeling and control methodology adopted for adaptive headlamp directionality as a function of vehicle motion and road geometry. The methodology is aimed at the design and implementation of a FF-MPC, which will be able to dynamically adjust the headlamp beam as a function of vehicle motion as well as predicted road curvature. The FF-MPC architecture comprises three principal components: a vehicle dynamics model that measures lateral velocity and yaw rate, a filtered actuator model that simulates the headlamp system, and a feedforward compensation term derived from real-time estimates of road heading. A dynamic bicycle model is used to estimate the slip angle,

which is subsequently merged with the expected road heading to give a reference beam direction. For actuator smoothness and physical constraints, this combined reference is then filtered. Figure 3.5 illustrates the whole process.

FF-MPC facilitates anticipatory headlamp alignment through the integration of feedforward road geometry and slip angle feedback, enhancing visibility while driving around bends. The controller employs a predictive approach that utilizes constrained optimization methods to optimize over a finite horizon. To evaluate the tracking precision and robustness of the controller, the process also involves modeling actuator saturation, simulating various road conditions, and performing sensitivity analysis of control settings. For headlamp beam alignment specifically, FF-MPC provides a new control architecture that combines anticipatory feedforward action with reactive feedback.

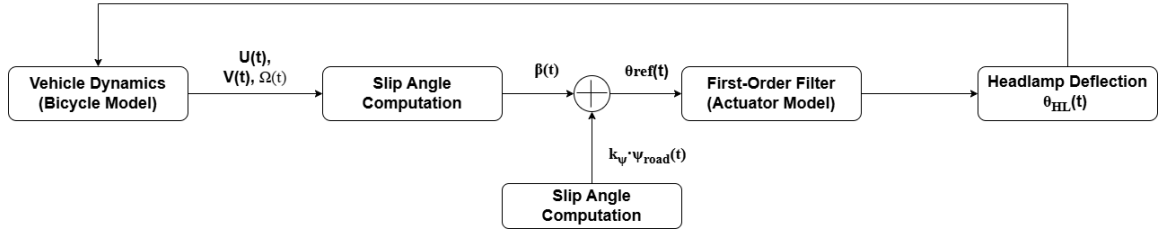


Fig 3.5 The process flow of the FF-MPC

Unlike other adaptive front lighting system studies that use steering angle-based heuristics or slip angle inputs, the FF-MPC utilizes real-time road heading ($\psi(t)$) information in the control framework. This enhances safety in low-light driving on curving roads by enabling anticipatory headlight positioning even before yaw or slip becomes substantial. In addition, its practicality is enhanced by merging constraint handling, predictive optimization, and filtered actuator model into one framework. Testing over a wide range of simulated road conditions with changing curvature and velocity profiles supports the technique, which is distinct from standard MPC in that it has inherent road geometry adjustment. The model monitors the lateral vehicle dynamics and compensates the headlamp deflection in real time. An added feedforward term, road heading angle $\psi_{road}(t)$ is included to improve responsiveness and line up the headlamp beam along the path of the road.

3.7.1 State Variables

The state vector is defined as

$$\mathbf{x}(t) = \begin{bmatrix} V(t) \\ \Omega(t) \\ \theta_{HL}(t) \end{bmatrix} \quad 3.50$$

with the Longitudinal Velocity $U(t)$ assumed to be either constant or varying with a known profile.

Slip angle is calculated using (3.32).

3.7.2 Vehicle dynamics

$$\frac{C_{\alpha f} + C_{\alpha r}}{MV} \quad 3.51$$

The lateral and yaw dynamics are modelled using,

$$\dot{V}(t) = \frac{1}{M} [-(C_{\alpha f} + C_{\alpha r})V(t) + (aC_{\alpha f} - bC_{\alpha r})\Omega(t) + C_{\alpha f}\delta(t)] \quad 3.52$$

$$\dot{\Omega}(t) = \frac{1}{J} [(aC_{\alpha f} - bC_{\alpha r})V(t) - (a^2C_{\alpha f} + b^2C_{\alpha r})\Omega(t) + aC_{\alpha f}\delta(t)] \quad 3.53$$

3.7.3. The Feedforward for Headlamp Deflection Control

The composite deflection angle is given by

$$\theta_{ref}(t) = k_h \cdot \beta(t) + k_\psi \cdot \psi_{road}(t) \quad 3.54$$

This combines slip angle feedback with a feedforward term from the road direction.

The dynamics of headlamp deflection are simulated as a first-order system:

$$\dot{\theta}_{HL}(t) = \frac{1}{\tau} (\theta_{ref}(t) - \theta_{HL}(t)) \quad 3.55$$

3.7.4. Road Heading Calculation

For a given path $x(s), y(s)$, the heading is calculated using

$$\psi_{road}(s) = \arctan2\left(\frac{dy}{ds}, \frac{dx}{ds}\right) \quad 3.56$$

This orientation is then interpolated in time to serve as a feedforward input during real-time control.

3.7.5. Steering angle calculation

Assuming the vehicle follows the road, the steering input is approximated from road curvature $\kappa(t)$,

$$\delta(t) = \kappa(t) \cdot L \quad 3.57$$

Where $L = a + b$

3.7.6. Controller output

The controller's output is the headlamp deflection,

$$\theta_{HL}(t) = \text{output of a first-order filter driven by } \theta_{ref}(t) \quad 3.58$$

This angle determines the side orientation of the headlamp beam.

3.7.7. Constraints and Physical Limits

For safe operation and hardware protection:

$$\theta_{HL}(t) \in [-\theta_{max}, \theta_{max}] \quad 3.59$$

Where θ_{max} is the maximum permissible headlamp sweep (15°).

The model utilizes vehicle dynamics to estimate slip angle and couples it with road heading to produce a reference beam deflection. The control strategy enhances visibility around curves by pre-emptively compensating the headlamp using road geometry, as well as responding to real-time lateral behavior. The filter ensures smooth transitions appropriate for actuator response.

3.8. Summary of Controller Architectures

Mathematical foundations and control strategies employed for adaptive headlamp control are explained in this section. The equations outline the development of four significant models: Feedforward MPC (FF-MPC), Raw MPC, Extended Model Predictive Control (EMPC), and Filtered Proportional Controller (FPC). To illustrate how these model's comprehend vehicle dynamics and maximize beam deflection in different road and environmental conditions, governing equations, objective functions, and system constraints for each controller are discussed.

These sections (3.1 through 3.7) offers extensive modeling and control frameworks for dynamically adjusting vehicle headlamps according to real-time slip angle estimation. Starting with an elaborate dynamic vehicle model including lateral, longitudinal, and yaw dynamics, tire forces, braking characteristics, and aerodynamic influences, the chapter sets a sound basis for body slip angle estimation. The slip angle is subsequently employed to calculate the difference between the heading of the vehicle and its actual trajectory, and this difference directly affects headlamp direction during emergency maneuvers and cornering.

Filtered Proportional Controller (FPC)		
Control Law	: $\dot{\theta}_{HL}(t) = -\frac{1}{\tau} \theta_{HL}(t) + \frac{k_h}{\tau} \beta(t)$	(3.34)
Transfer function	: $H(s) = \frac{\theta_{HL}(s)}{s} = \frac{k_h}{\tau s + 1}$	(3.36)
Discrete Form	: <i>Euler Approximation</i> $\theta_{HL}[k] = \alpha \cdot \theta_{HL}[k-1] + (1-\alpha) \cdot k_h \cdot \beta[k]$ <i>where $\alpha = \frac{\tau}{\tau + T_s}$</i>	(3.37)

Saturation Constraint	: $\theta_{HL}(t) \in [-\theta_{max}, \theta_{max}]$ Applies a low-pass filter to smooth out actuator commands. Reactive in nature, but good for stability and comfort.
-----------------------	---

Raw State Model Predictive Controller (E-MPC)

Control Objective	: $\min_{\theta_{HL}} \mathcal{L} = \sum_{k=0}^N [(\theta_{HL}[k] - \psi_{road}[k])^2 + \lambda \cdot (\Delta\theta_{HL}[k])^2] \quad (3.43)$
Constraint Handling	: $\theta_{HL}[k] \in [-\theta_{max}, \theta_{max}]$ Doesn't use slip feedback and goes along road in a straight direction. Unable to dynamically merge and sens to curvature variations.

Extended Model Predictive Controller (E-MPC)

State-Space Model	: $X[k+1] = A_d x[k] + B_d \delta[k] \quad (3.47)$ Where $X[k] = [\beta[k] r[k]]^T$ and $A_d = I + T_s A_c, B_d = T_s B_c$
Control Objective	: $\min_{\theta_{HL}} \mathcal{L} = \sum_{k=0}^N [(\theta_{HL}[k] - k_h \cdot \beta[k])^2 + \lambda \cdot (\Delta(\theta_{HL}[k]))^2]$
Rate Smoothing	: $\Delta(\theta_{HL}[k]) = \theta_{HL}[k] - \theta_{HL}[k-1] \quad (3.48)$
Constraint Handling	: $\theta_{HL}[k] \in [-\theta_{max}, \theta_{max}] \quad (3.49)$

Feedforward-Enhanced MPC (FF-MPC)

Reference Beam Angle	: $\theta_{ref}(t) = k_h \cdot \beta(t) + k_\psi \cdot \psi_{road}(t) \quad (3.54)$
Filtered Actuation	: $\dot{\theta}_{HL}(t) = \frac{1}{\tau} (\theta_{ref}(t) - \theta_{HL}(t)) \quad (3.55)$
Control Objective:	: $\min_{\theta_{HL}} \mathcal{L} = \sum_{k=0}^N [(\theta_{HL}[k] - \theta_{ref}[k])^2 + \lambda \cdot (\Delta(\theta_{HL}[k]))^2]$
Constraint Handling	: $\theta_{HL}(t) \in [-\theta_{max}, \theta_{max}]$ Makes use of feedforward to anticipate curvature and filtering and slip feedback in order to stabilize

A Filtered Proportional Controller (FPC) was originally developed to filter the deflection command, enhancing actuator response and driver visual comfort. FPC, being reactive, had steady-state errors and inadequate anticipation of curves. This prompted the creation of an Extended Model Predictive Controller (MPC), which included predictive optimization and actuator dynamics but still did not have road geometry foresight. To counter this, a

Feedforward-Enhanced MPC (FF-MPC) was developed, incorporating road heading as a feedforward input. FF-MPC exhibited enhanced accuracy, promptness, and tracking in dynamically curved trajectories and thus posed a more efficient and smart control solution for adaptive headlamp systems.

3.9 Results and discussion on the performance of Predictive control models for trajectory-aligned vehicle headlamp adjustment

The following sections examine four control methods—Dynamic Predictive Control (DPC), Raw State Model Predictive Control (RS-MPC), Extended MPC, and Feedforward MPC (FF-MPC). The Raw MPC model gives data regarding classical prediction-based control without dynamic state augmentation. Though the Extended MPC enhances performance by incorporating other vehicle states such as yaw rate and slip angle, the DPC model incorporates real-time slip dynamics. By introducing a feedforward path-planning module to forecast vehicle action, the FF-MPC enhances the control.

Various performance measures, such as tracking error, headlamp deflection accuracy, and control smoothness, are compared in addition to a range of scenarios, such as sudden turns and sudden speed changes. Also, a focused case study on a 600-meter road length illustrates each controller's respective strengths and real-world applicability. To assist in the selection of the most effective control strategies according to specific driving situations and performance demands, the chapter concludes with a summary of the key findings and observations.

3.9.1 Performance Evaluation of the Dynamic Predictive Control model

The capability of the model to sustain accurate trajectory tracking and appropriate headlamp positioning in a range of driving conditions, including sudden curves and changes in speed, is the focal point of the study. DPC model shows enhanced responsiveness and stability through the application of real-time slip angle and yaw rate feedback. To confirm the effectiveness of the DPC strategy in dynamic situations, important performance indicators like tracking error, control effort, and headlamp alignment accuracy are considered. The DPC controller, immune to transient longitudinal dynamics, accurately translates slip angle into headlamp deflection during constant speed operation as shown in Figure 3.6.

The actuation of lateral dynamics is successful as evidenced by the close tracking of slip and deflection angles. It enhances preview visibility, lane awareness, and obstacle detection by extending beam projection into curves. This enhances nighttime driver

comfort and safety on curved roads. This meets minimum performance standards for operation in the real world and displays responsive, seamless tracking. Its stability affirm the controller's capability to maintain visibility through turns, prevent cutting corners, and enhance steering confidence.

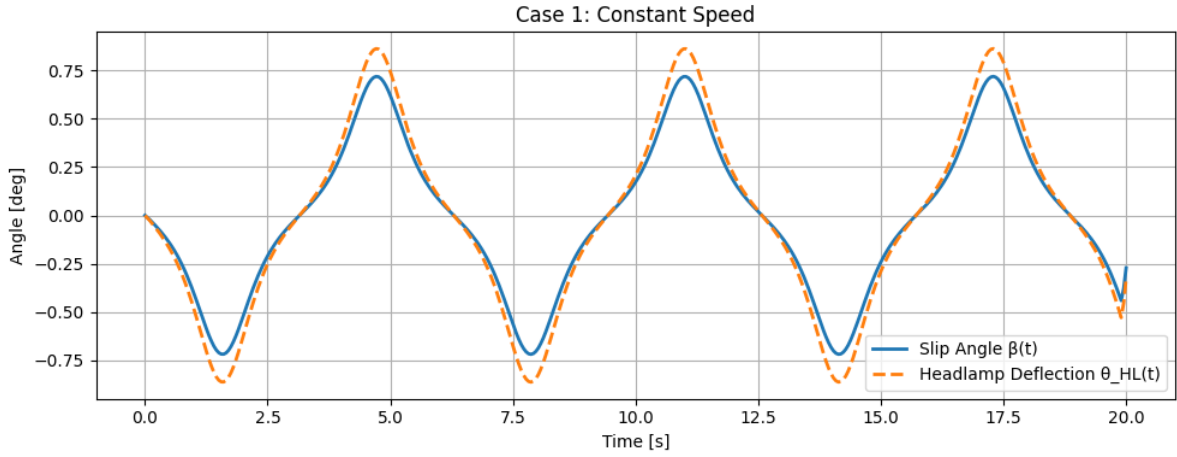


Figure 3.6: Headlamp deflection based on changing slip angle

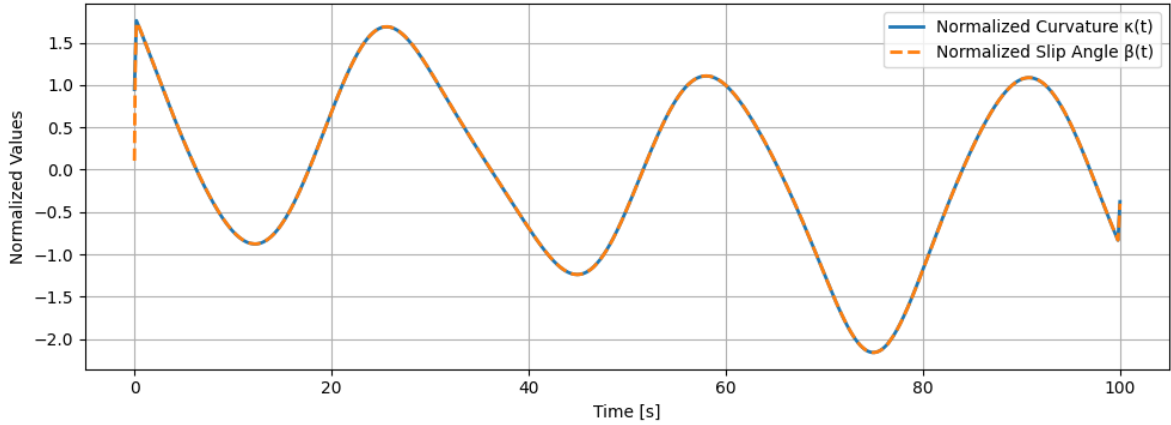


Figure 3.7 Proportionality of Slip Angle to the Road Curvature

Over a 100-second, the plot (Figure 3.7) displays the normalized road curvature $\kappa(t)$ and slip angle $\beta(t)$ along the zig-zag path. The path mimics realistic semi-urban or hilly nighttime driving conditions by causing moderately strong alternating left-right turns. The accuracy of the DPC controller in capturing minute lateral motions is verified by the close overlap between $\kappa(t)$ and $\beta(t)$. Transitions are free of lag or overshoot and smooth indicating that the system responds to slight curvature without overshoot. This confirms that the controller can be utilized in general curved-road driving scenarios where visual predictability—more than vigorous reaction—is of prime importance.

Figure 3.8 illustrates the dependence of slip angle (β) on road curvature (κ) at various speeds under constant as well as variable speed conditions. Under constant speed, the correlation remains high (≈ 1) across all speeds and indicates that slip angle and road

curvature are directly aligned. But in case of varying speed, the correlation initially is poor at low speeds (≈ 0.18 at 5 m/s) but shows a substantial gain with speed rise, reaching plateau beyond 20 m/s. This proves that DPC model works extremely well for curvature tracking even at high speeds under dynamic speed profiles.

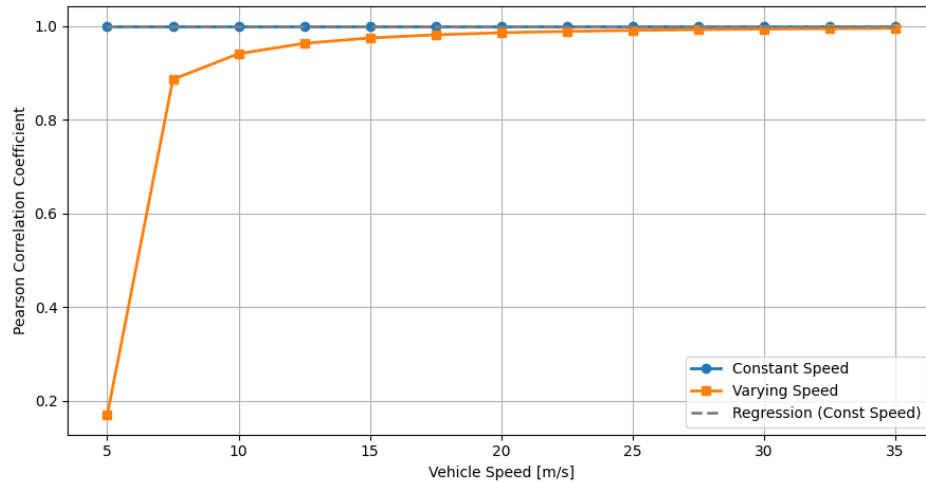


Figure 3.8 Correlation between Slip Angle and Road curvature vs Speed

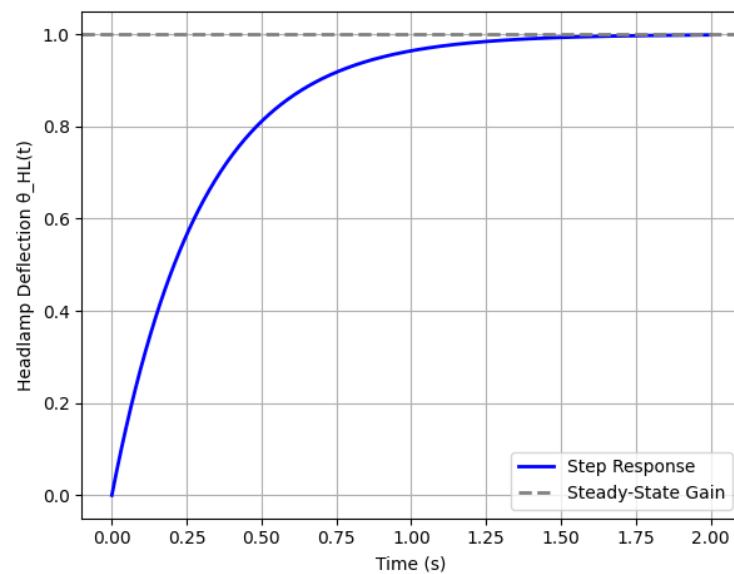


Figure 3.9 Step response of the DPC Model

Figures 3.9 and 3.10 show the step and impulse responses of the filtered proportional controller and provide key insight into its real-time response when reacting to changes in slip angle—whether it be progressive curve entry or abrupt disturbances. As the figure shows, the step response exhibits smooth, non-oscillatory increase to steady-state, which suggests that the headlamp deflection moves smoothly without sharp motion. Figure 3.10 illustrates that the impulse response quickly declines, efficiently rejecting short-duration transients. These dynamic properties are both desirable from a safety and user experience standpoint, providing stable, distraction-free lighting. Damping behavior also prevents

short-lived disturbances such as impacts over potholes, steering jitters, or sensor noise from causing erratic beam motion. This supports the system's objective to offer steady, context-driven headlamp control under nighttime driving conditions, without causing visual unease or instability.

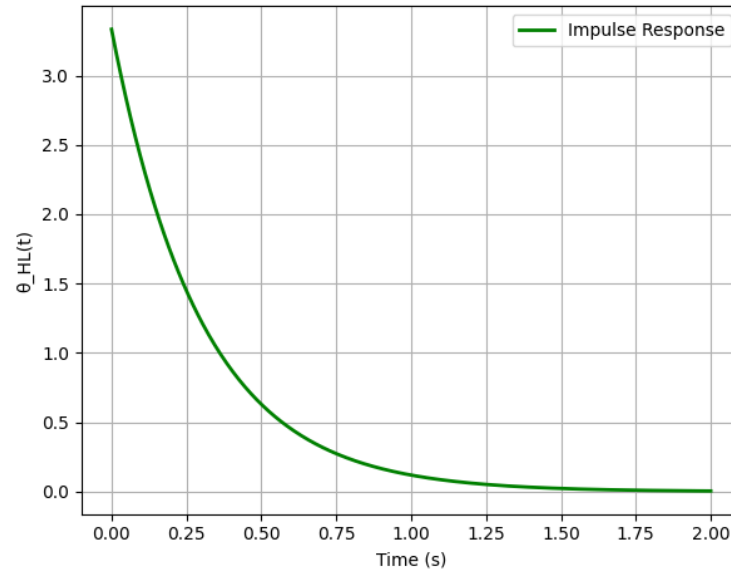


Figure 3.10 The Impulse response

Figure 3.11 illustrates the slip angle trajectories for three different steering inputs: high-frequency input ($\pm 5^\circ$ HF), low amplitude ($\pm 5^\circ$), and larger amplitude ($\pm 10^\circ$). As expected, larger slip angles are produced by increasing the amplitude of the steering input. The headlight deflection angle (θ_{HL}) is scaled accordingly by the Dynamic Preview Controller (DPC) in response. The slip angle oscillates at a high frequency in the high-frequency case. A measurable input-output lag exists even when the controller maintains the amplitude of the response within reasonable limits. This demonstrates the filtering action of the controller, which balances between responsiveness and stability. This is especially necessary in dealing with temporary driver adjustments or traffic interruptions. Despite varying driver intentions, the controller maintains proportionality. Lag does, however, become evident in high-frequency maneuvers (e.g., such as fast zigzags), which can lead to perceivable delays in beam redirection. While safety is not likely to be severely compromised, this could make drivers less confident in making rapid turns at night.

The trend of the Slip Angle – Driver Input Variation and the Headlamp Deflection – Driver Input is similar. This is due to the fact that the slip angle directly controls the headlight deflection through a proportionate gain. The deflection system efficiently monitors and translates alterations in the slip angle into beam motions as the slip angle responds to

alterations in the amplitude or frequency of steering driver input. In an effort to offer timely and context-related illumination during maneuvers, especially on bends and on sudden corrections, this close coupling ensures that the headlight orientation dynamically aligns with the vehicle's instantaneous trajectory.

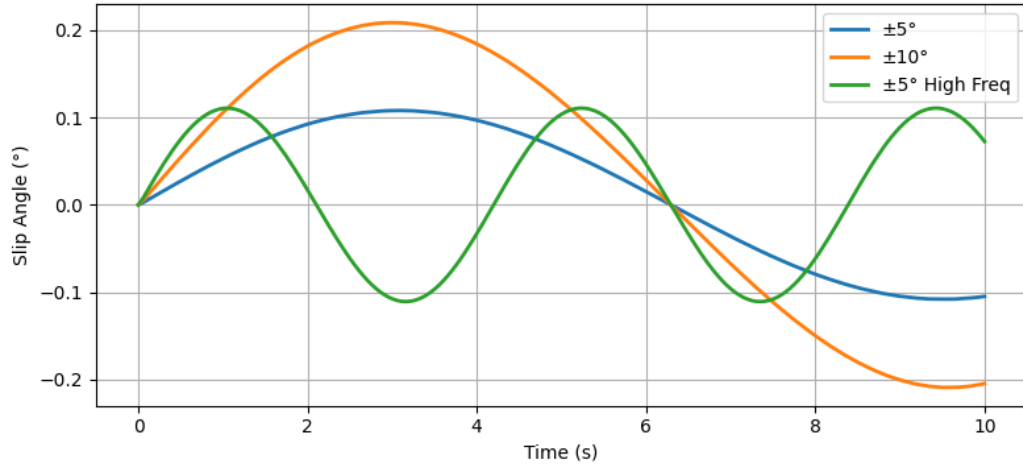


Figure 3.11 Slip Angle-Driver Input Variation

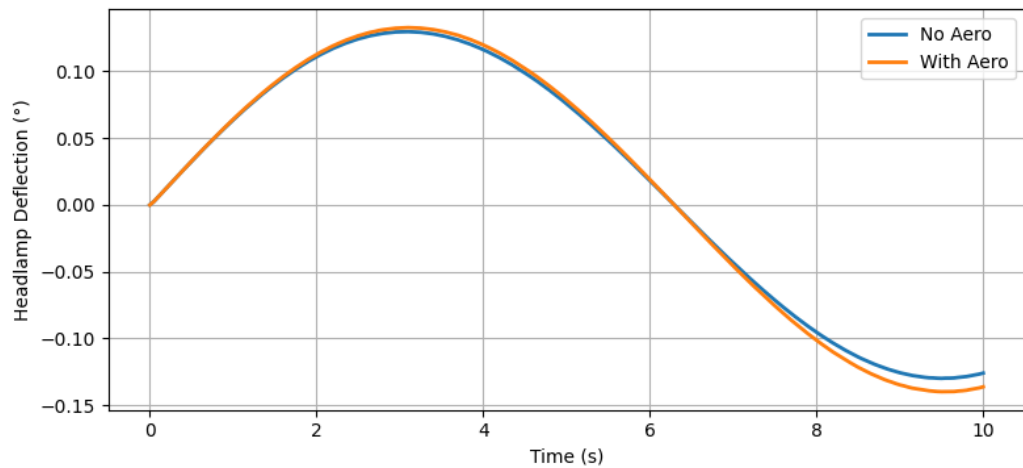


Figure 3.12 Effect of Aerodynamic Drag on Headlamp deflection

Due to the added stability of downforce and aerodynamic drag, the slip angle gets slightly damped, especially at high speeds. As a reaction to these subtleties, the DPC system adjusts the headlight deflection accordingly. On a positive note, the system readily adapts to alterations in the airflow characteristics or vehicle body shape without being re-tuned. Headlamp deflection and aerodynamic plots mirror the damping of slip angles indicated in the previous subplot. For the aerodynamic case, the lesser β represents the reduced magnitude of θ_{HL} (Figure 3.12). This illustrates how the system adapts to alterations in vehicle structure. It illustrates the fact that, if the underlying β is known exactly, any controller can be utilized across various car classes (e.g., sedans and SUVs).

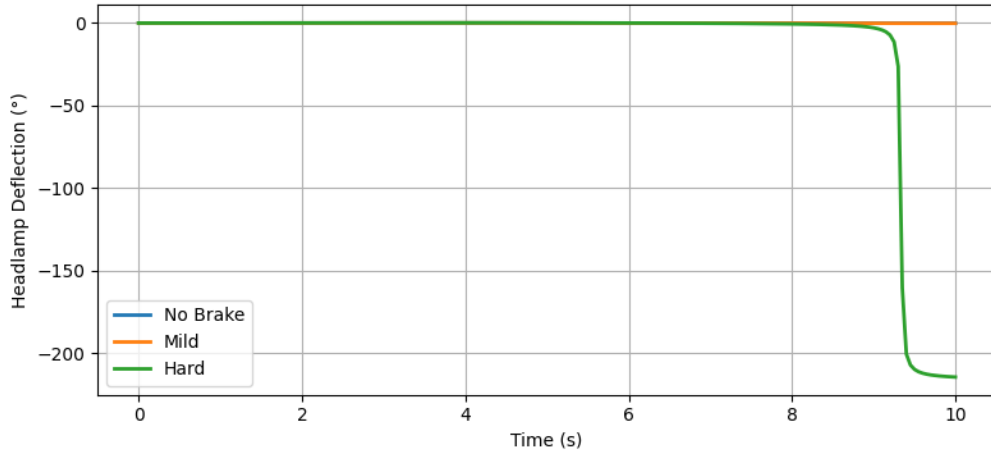


Figure 3.13 Effect of Braking on Headlamp deflection

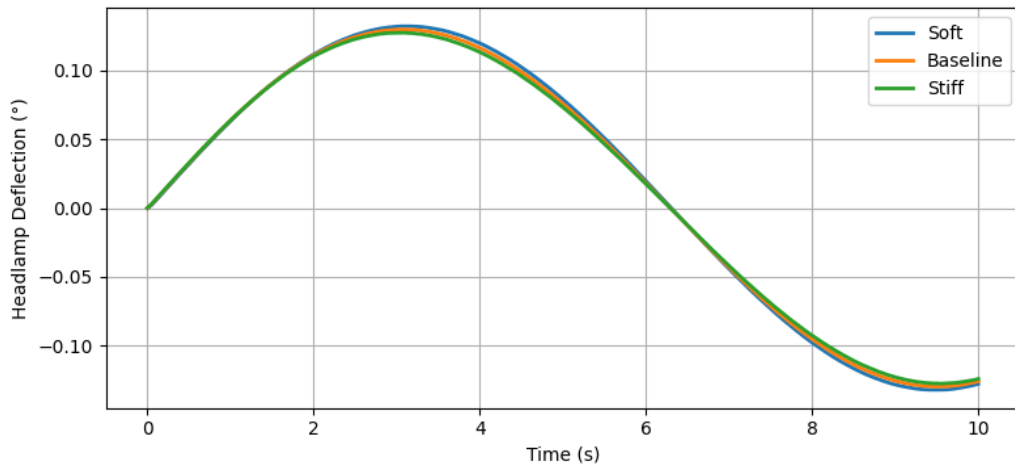


Figure 3.14 Effect of Tire stiffness on Headlamp deflection

The slip angle during braking, especially hard braking, drops sharply. The sudden forward load transfer and reduced lateral grip at the rear tires are responsible for this. Such sudden dynamic changes create large angular movements in β , and if they are transmitted directly into the headlamp system, they can cause beam swing to be too great, which might possibly confuse the driver. The conclusion suggests that unwanted spikes need to be minimized by either modulation or braking-aware filtering. The large slip angle movements are translated to proportional headlight deflections by the controller. For hard braking, θ_{HL} goes down abruptly, which to the driver may seem like beam jitter or beam flicker as shown in Figure 3.13. These sudden oscillations would be hazardous without additional conditioning if they occur during emergency braking. To maintain visual stability in panic stops or during ABS activation, the system should provide brake-sensitive damping or deflection limits.

The size of the slip angle is small but clearly affected by tire stiffness variations (stiff, baseline, and soft). Since they are more laterally compliant, the soft tires possess a relatively larger β , and the stiff tires possess a relatively smaller one. The DPC controller

does not require frequent re-tuning for different types of tires since it can accommodate these variations without losing stability. In real-world operation, where tire conditions vary due to wear, inflation, or seasonal variations, such robustness is important.

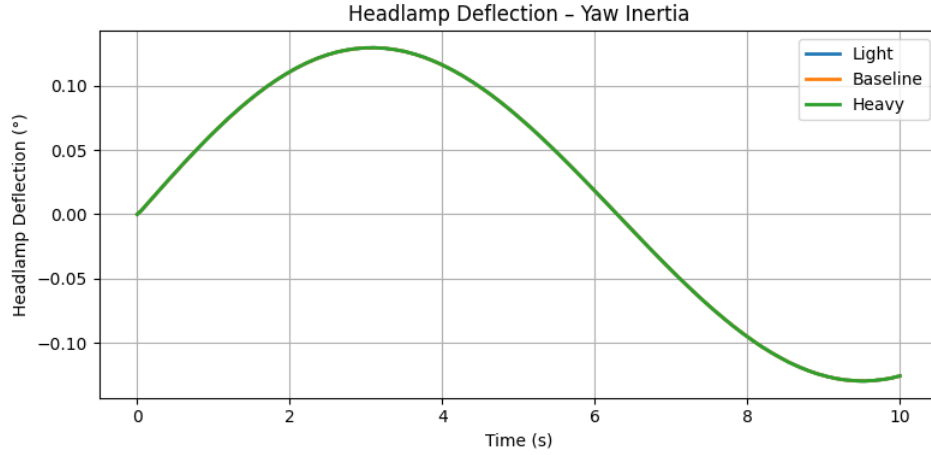


Figure 3.15 Effect of Yaw inertia on Headlamp deflection

As can be observed in Figure 3.14, the amplitude changes of the headlight deflection are small and non-disruptive. This proves once again that the system is able to adapt passively to mechanical variation without compromising fundamental functionality. This allows for easy integration into many types of tires and automobile models.

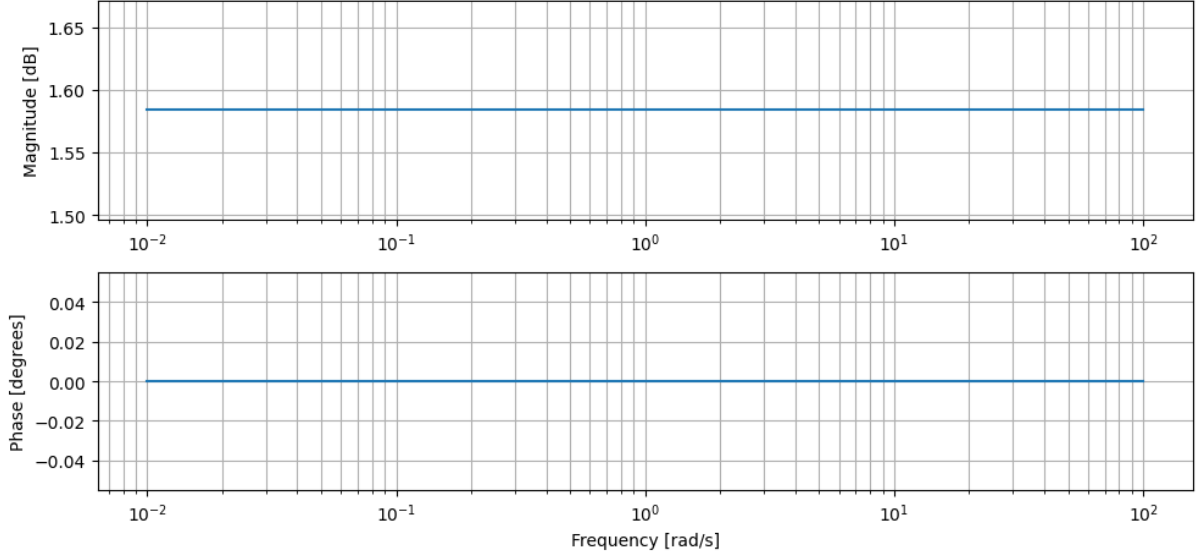


Figure 3.16 Bode Plot of DPC

The rate at which the slip angle varies depends on yaw inertia variations. Heavier or more rear weight-biased vehicles respond to steering more slowly, leading to smoother β profiles. Increased inertia results in phase lags increasing and absolute magnitudes remaining unchanged. While the effect is minimal in this case, this would suggest that the DPC system can deliver slightly delayed headlight compensation in heavier vehicles. The same trend is

followed by the headlight system. The time is varied to some extent with inertia, but the magnitude of the deflection angle is constant (Figure 3.15). Even in vehicles with asymmetric mass distributions or under payload transitions, the controller's performance is assured by its scalability with yaw dynamics. Tonal calibration, however, would be beneficial to payload systems that are highly dynamic (such as logistics vans).

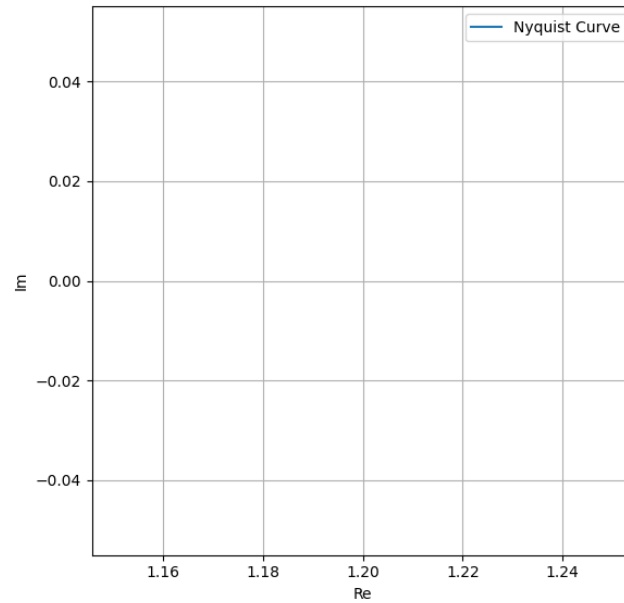


Figure 3.17 Nyquist Plot of DPC

There is no dynamic filtering or time shaping in the DPC, a static gain-only system. This is substantiated by frequency-domain analysis based on Bode and Nyquist plots. Instantaneous output tracking without phase lag or delay is manifested by the Bode magnitude (Figure 3.16) being flat at +1.58 dB (gain = 1.2) and the phase remaining at 0° across all frequencies. DPC's Nyquist plot, presented below in Figure 3.17, unmistakably confirms its static, gain-only status. With very little variation along the imaginary axis, frequency response is confined to a short length along the real axis, at $\text{Re} \approx 1.2$. This indicates that there is no dynamic response or phase shift in the system over the frequency range. The plot illustrates how DPC directly applies the input to the output without filtering or time-shaping, making it a memoryless system with stable constant gain ($k_h = 1.2$). Even though inherent stability is established through the Nyquist trajectory (Figure 3.17) not encircling the critical point $(-1,0)$, the absence of phase dynamics or bandwidth limitation also indicates that the controller is highly sensitive to high-frequency input disturbance. This accentuates the weaknesses of DPC in handling real-world situations, where a more robust, frequency-aware control strategy is called for because of sensor noise, actuator delays, and transient steering corrections.

3.9.1.1 Summary on performance of DPC

When subjected to different dynamic conditions, the DPC controller demonstrates good conformity with its main goal: offering real-time, proportional headlamp deflection based on slip angle, which improves road light while driving on curved roads at night. Its simplicity, fast response time, and consistent performance over a variety of vehicle parameters, such as tire stiffness, aerodynamics, and inertia, are its primary strengths. Due to its flexibility, it can be applied on a wide range of vehicle platforms, specifically at moderate to high speeds when lateral dynamics are palpable and predictive lighting is essential.

Limitations are present at low speeds, with high frequency steering inputs, and particularly hard braking. Sudden changes in slip angle under these conditions can make headlamps move rapidly and perhaps distractingly, which can be a problem in parking, urban, or emergency maneuvers where beam stability is paramount. In addition, using only $\beta(t)$ can be misleading under low-dynamic conditions because β is not very informative about curvature. Improvements such as dynamic gain modulation, multi-sensor fusion based on steering angle or yaw rate, and filtering of input (e.g., based on longitudinal acceleration) are recommended to address these shortcomings. Stability can also be enhanced for violent motion by using adaptive damping and deflection rate limiting. In general, the DPC performs well for night-time driving on winding roads, but it must be better for edge-case robustness and real-world dependability.

3.9.2 Evaluation of the performance of the Raw State Model Predictive Control (RS-MPC)

In order to explore predictive control possibility using road heading information, a Raw State MPC was created after the DPC model. Measuring directly the road heading angle, the formulation circumvents dynamic vehicle feedback in the form of slip angle and aligns the headlight beam with predicted travel. It was attempted to see if predictive tracking in isolation, with no added complexity of the model, could provide better deflection accuracy and lower jitter.

In Figure 3.18, the headlamp deflection (θ_{HL}) exhibits an intangible, spike near $t = 8$ s, with magnitudes on the order of 10^{11} degrees, whereas the reference road heading (ψ_{road}) is nearly constant. This instability results from the absence of feedback from the vehicle dynamics, rendering the controller highly susceptible to even minute disturbances or errors in curvature prediction. The optimizer generates control outputs which are not operational

or diverge due to the fact that it takes even small numerical error in ψ_{road} to destabilize them. The response highlights one of the main disadvantages of the raw formulation and justifies the need to incorporate dynamic states in order to ensure stability and useful performance in practice.

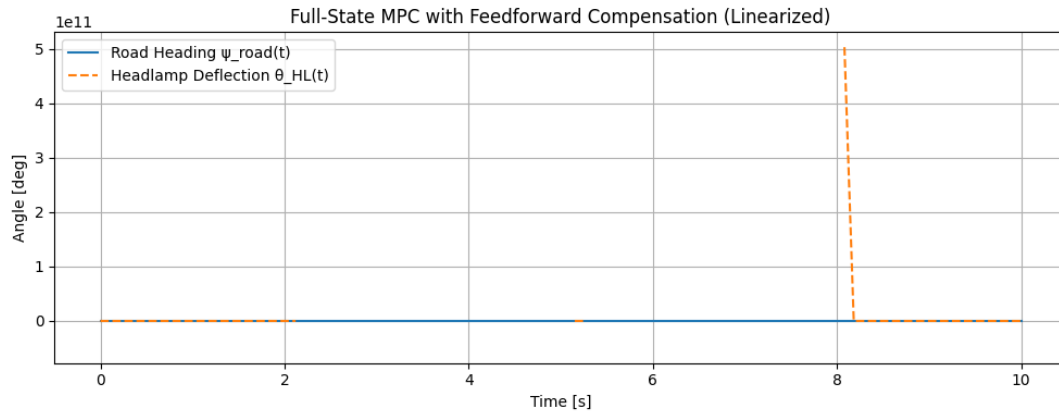


Figure 3.18 Headlamp deflection against heading road at varied intervals

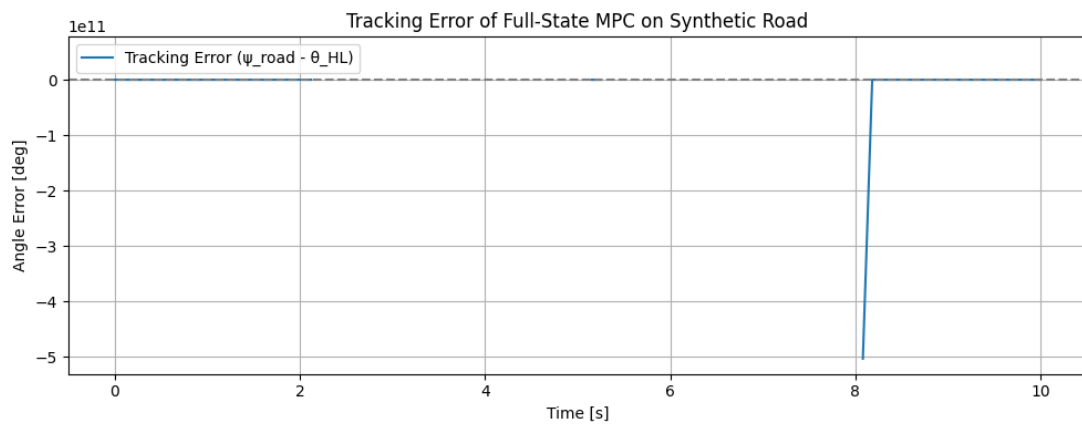


Figure 3.19 Tracking Error of RS-MPC Model

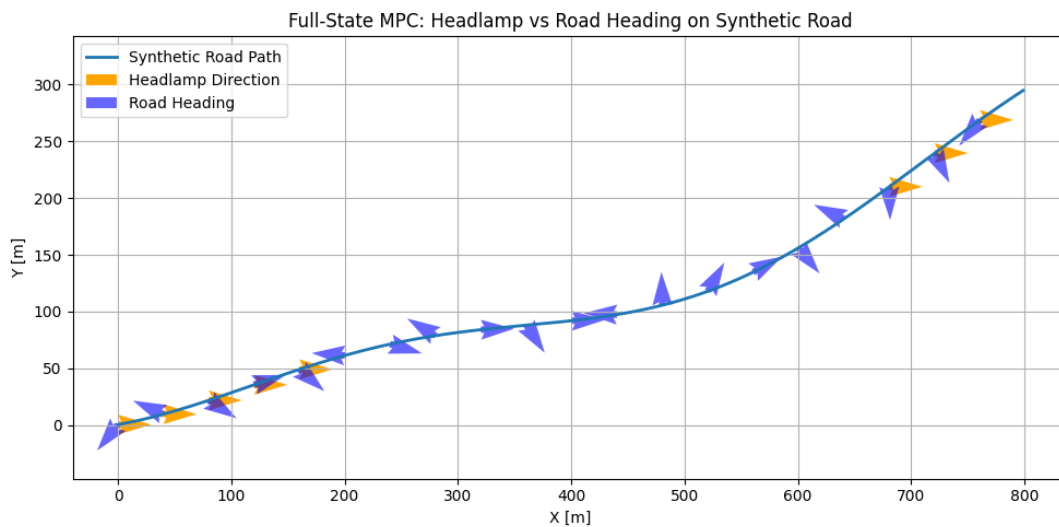


Figure 3.20 Tracking Error of RS-MPC Model

The error of tracking in the RS-MPC model is shown in Figure 3.19. Initially, the controller maintains error near zero, reflecting effective tracking. At $t = 8$ s, however, there is a large and unbounded error divergence, with the error reaching over -5×10^{11} degrees. Numerical instability due to prediction noise, optimizer sensitivity, and absence of stabilizing feedback from vehicle dynamics is what this behavior mirrors. The controller is devoid of internal adjustment to reject disturbances or noise in ψ_{road} if inputs from slip angle and yaw rate are not supplied.

The road heading and headlight direction, as determined by the Raw State MPC, are represented by overlay vectors on the synthetic road path in Figure 3.20. Even though the vehicle's trajectory is smooth, there are several places throughout the route where it is easy to see the difference between the headlight direction (orange) and the actual road heading (blue). These angular discrepancies show that, particularly in sections with abrupt bends or heading rate changes, the Raw State MPC is unable to reliably align the beam with the road curvature.

3.9.2.1 Summary on performance of RS-MPC

The controller's dependence on feedforward tracking of ψ_{road} alone, without taking into account dynamic vehicle states like yaw rate or slip angle, is the cause of the problem. Consequently, the precision of headlamp deflection is immediately affected by any latency, optimizer instability, or curvature prediction inaccuracy. This makes it more difficult for the controller to maintain context-aware illumination, which is essential while driving at night when beam misalignment can jeopardize safety and visibility. The plot supports previous findings on instability and low resilience, emphasizing the shortcomings of the Raw State MPC in practical applications and providing more evidence in favor of switching to an Extended MPC design that makes use of dynamic feedback.

3.9.3 Evaluation of the performance of the Extended-Model Predictive Control (E-MPC)

To enhance headlight deflection performance, the Extended Model Predictive Control (E-MPC) approach builds on earlier formulations by incorporating rate limits, predictive filtering, and vehicle dynamics. E-MPC adjusts the beam dynamically based on slip angle trends while respecting actuator constraints, balancing responsiveness and stability. This section evaluates the behavior of the controller in numerous scenarios, highlighting its noise robustness, robustness, and applicability to adaptive lighting under dynamic driving conditions in real-world environments.

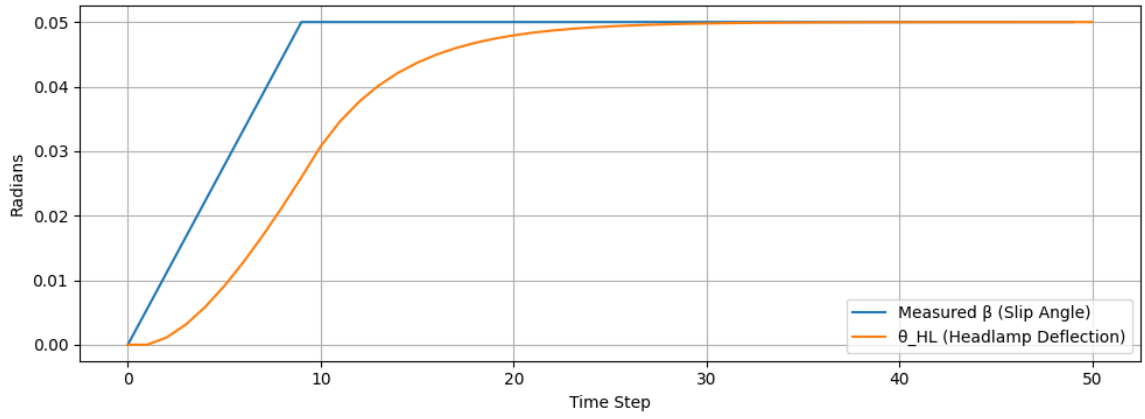


Figure 3.21 Headlamp deflection as filtered response to slip angle

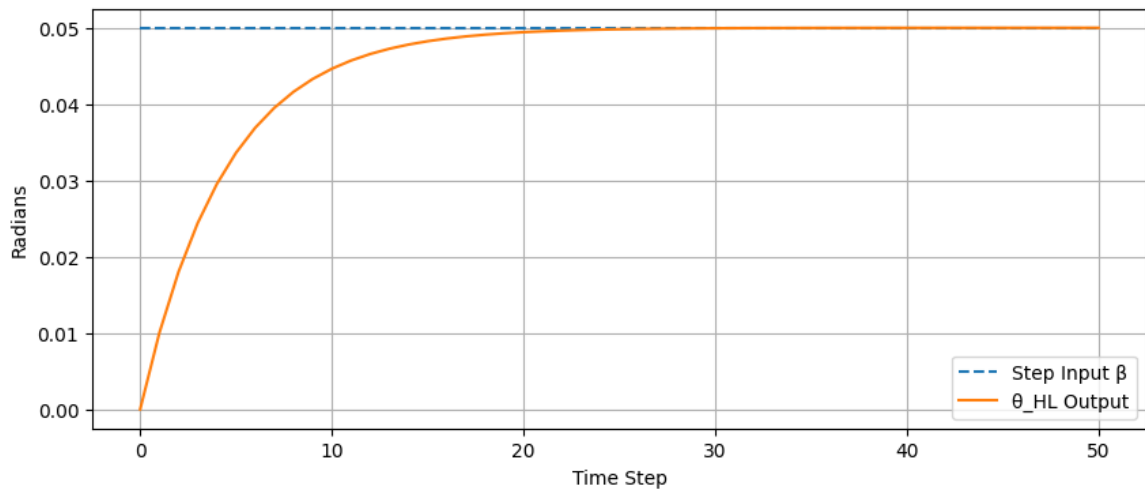


Figure 3.22 Tracking Delay-step response of θ_{HL} to β

The response of the system when the slip angle β reaches a saturated limit is demonstrated by Figure 3.21. The headlamp deflection θ_{HL} follows smooth asymptotic tracking without overshoot as it gradually becomes closer to the same value. The gradual slope confirms the presence of internal rate limitations or filtering, which prevent abrupt actuator commands. This is ideal for minimizing driver distraction and ensuring mechanical durability. A ramp input on β in Figure 3.22 demonstrates a small but noticeable lag in θ_{HL} . The controller follows the increasing slide angle adequately, but the profile is smooth and slightly delayed. This delay is caused by intentional dampening that was incorporated into the E-MPC configuration to trade-off between stability and responsiveness, as can be seen in real situations where steering correction might be jerky or sensor input noisy.

The slip angle increases linearly and saturates in this graph, while θ_{HL} increases filtered and levels off once the input no longer varies (figure 3.23). This response mirrors the low-pass filter function built into the controller and serves to prevent beam jitter and stable

illumination during drastic changes in vehicle dynamics. When the E-MPC is subjected to a step command in slip angle, it presents a smooth and highly damped response. Figure 3.24 displays how the headlight deflection settles in steady-state without overshoot or oscillation. A good prediction horizon and control gain structure within the MPC design are reflected in its stability, which also displays excellent temporal control.

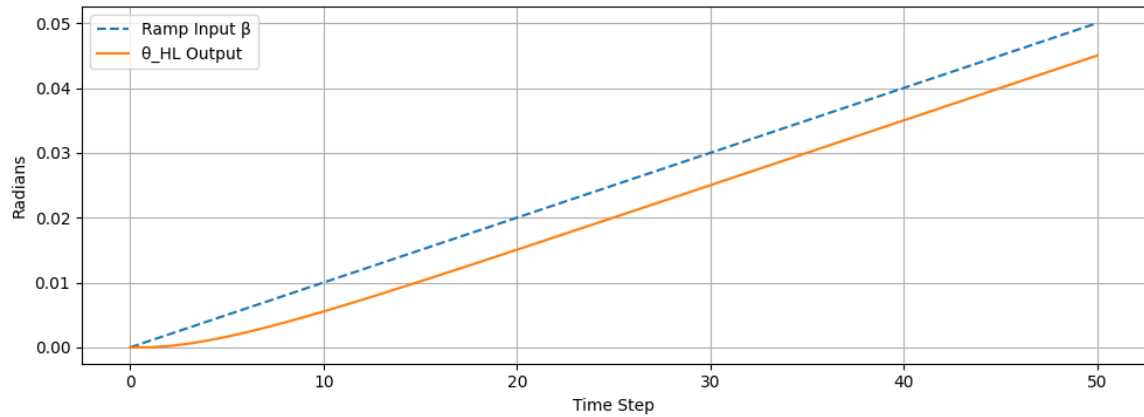


Figure 3.23 Ramp Response of θ_{HL} to β

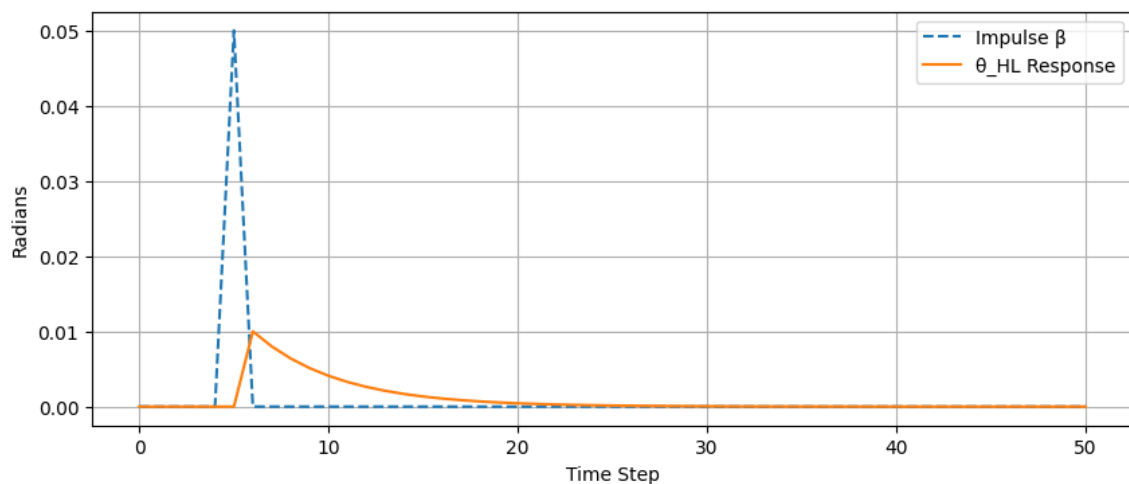


Figure 3.24 Impulse Response of θ_{HL} to β

The θ_{HL} output also rises with a similar curvature but at a reduced slope for an ever-increasing β input (Figure 3.25). The slight difference is intentional and confirms the internal smoothing algorithm, required so as not to place high-rate actuator orders on the lighting system, or reduce the driver's visual comfort. Impulse test plot in Figure 3.24 shows a damped θ_{HL} response following a short, sudden β disturbance. The controller's ability to reject transient noise and prevent it from propagating through the beam control channel is evidenced by the rapid return to baseline. This behavior is needed to sustain constant lighting in the presence of road roughness or steering jerks.

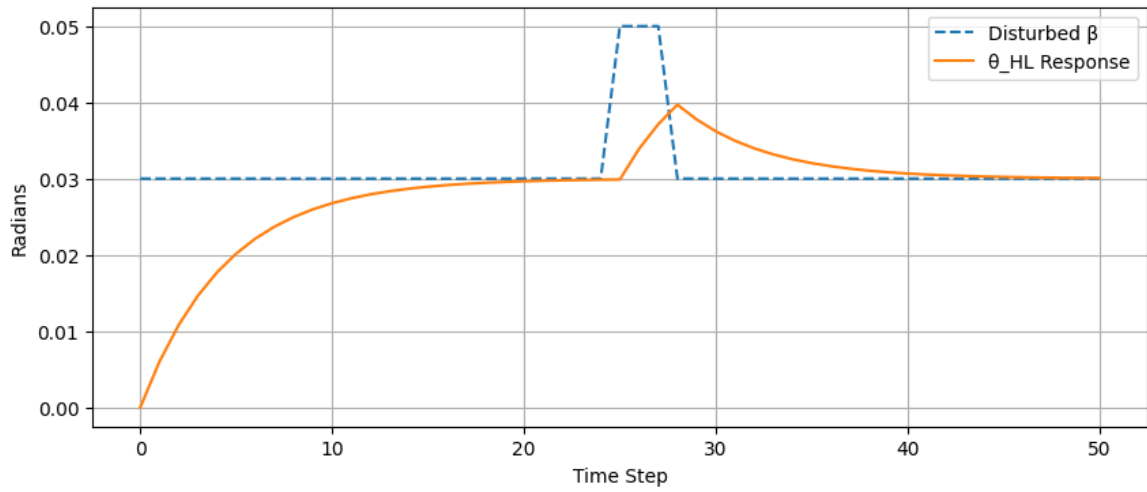


Figure 3.25 Disturbance rejection

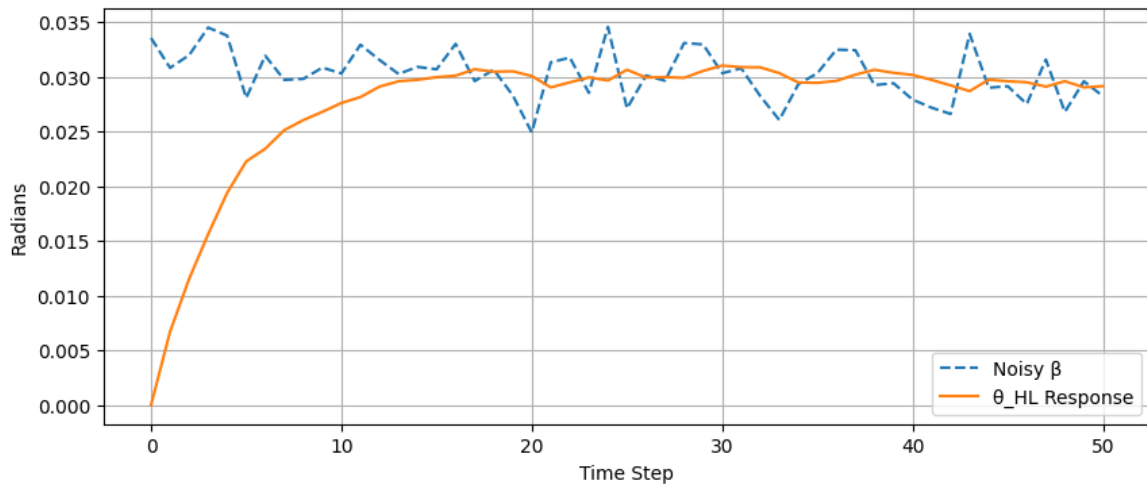


Figure 3.26 Noise Sensitivity

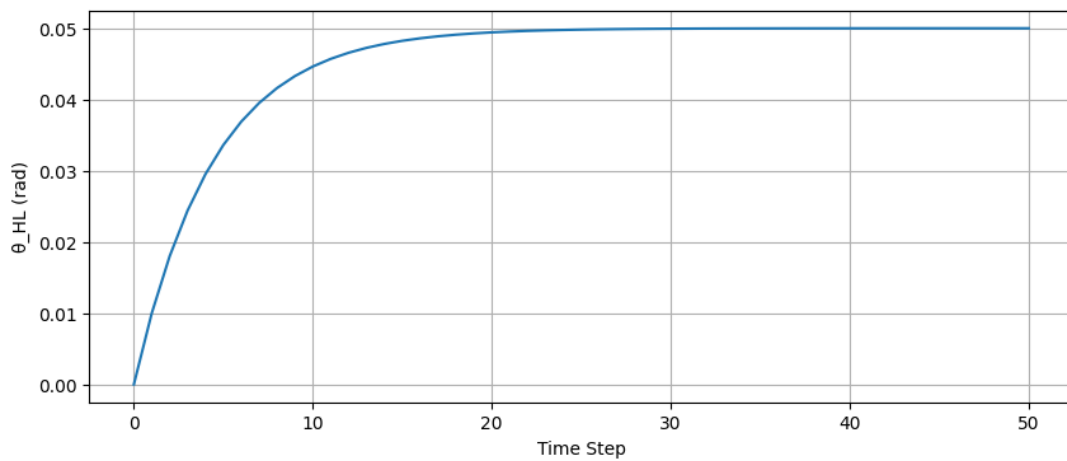


Figure 3.27 Disturbance Rejection

As indicated in Figure 3.26, the headlamp deflection shows a steep rise and a smooth return to its original value when challenged by a short-term disturbance along a constant β baseline. This enhances the E-MPC's robustness for dynamic cornering maneuvers by

illustrating its ability to reject transient side shocks with virtually no delay or residual error. Output from the controller is smooth and very insensitive to noisy β input. The efficacy of E-MPC's intrinsic filtering or reduction of noise through its techniques can be seen from Figure 3.27 & 3.28. This trend averts rapid beam direction changing, which reduces driver distraction and visual discomfort against mere proportional systems.

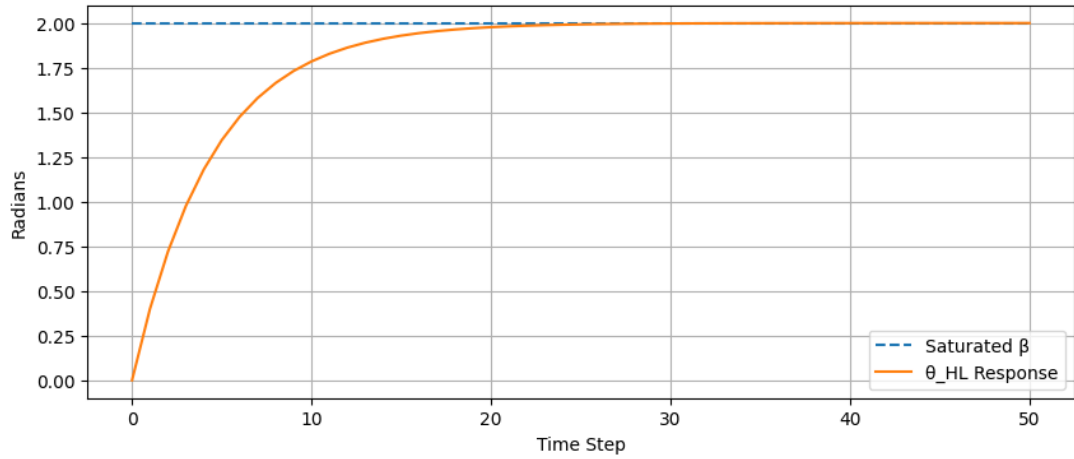


Figure 3.28 Input saturation behavior

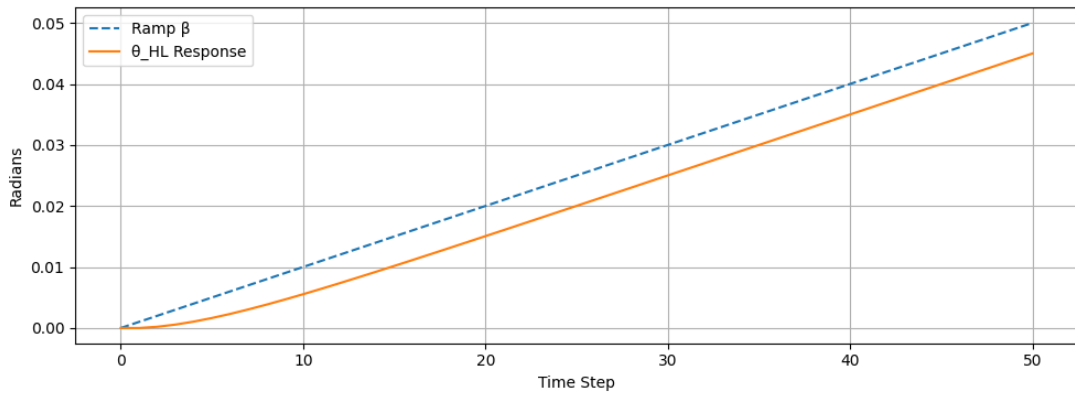


Figure 3.29 Rate of Change of θ_{HL} (Tracking delay)

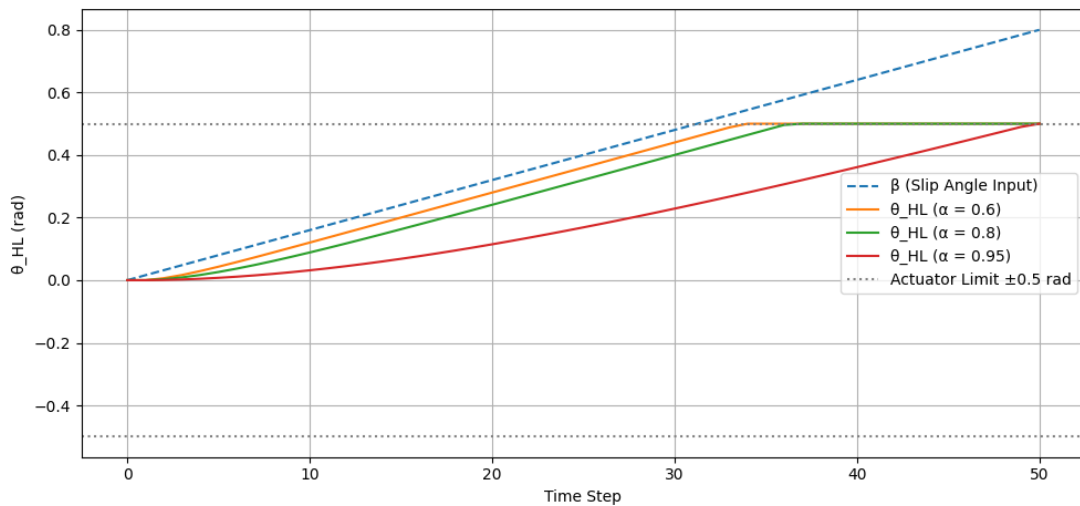


Figure 3.30 Filtered θ_{HL} Response with Varying α and Actuator Constraints

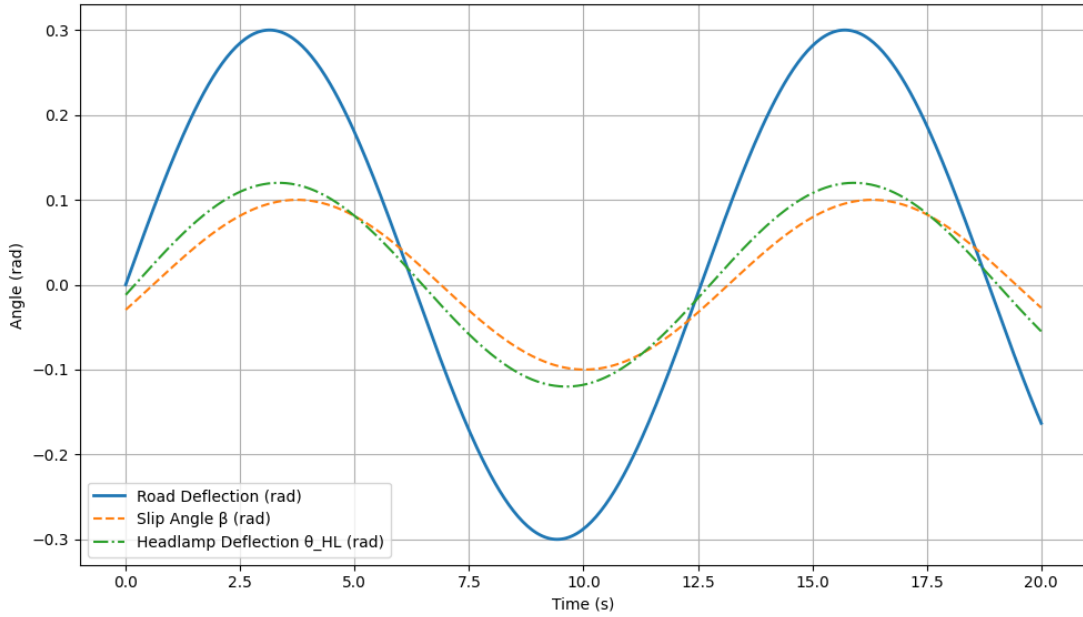


Figure 3.31 Tracking Performance of E-MPC's Road Deflection, Slip Angle and Headlamp Response

The Figure 3.29 plot illustrates that the rate of change in headlamp deflection is carefully controlled, rising rapidly but saturating smoothly without sudden breaks. In keeping actuator commands within safe operating limits, this constraint enhances system longevity and passenger satisfaction. Here, under actuator limitations, different filter coefficient α values are explored (see Fig. 3.30). θ_{HL} follows β more tightly with increasing α , but is also at risk of hitting actuator limits earlier.

This trade-off illustrates how E-MPC's flexibility in tuning enables it to balance between hardware safety constraint and responsiveness, which is imperative for embedded vehicle systems. The performance of the Extended MPC in simulating road deflection via slip angle and headlamp deflection responses is depicted in Figure 3.31. The road deflection (solid blue) serves as the reference, while the vehicle's dynamic response is represented by β (dashed orange).

The headlight deflection (dash-dot green) has a constant phase relationship by tracking β very closely with a slight smoothing. The alignment is well-controlled and consistent despite both reactions being slightly behind the road curvature. This shows how Ex-MPC is able to screen out high-frequency disturbances while allowing for accurate and fast beam control, which is necessary to maintain the best vision possible in the case of continuous cornering.

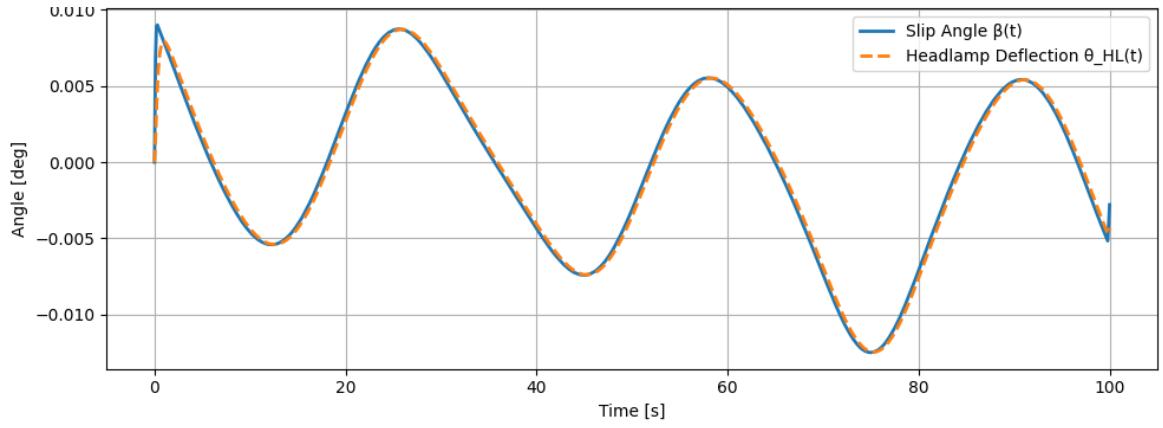


Figure 3.32 Extended MPC response in Constant Speed

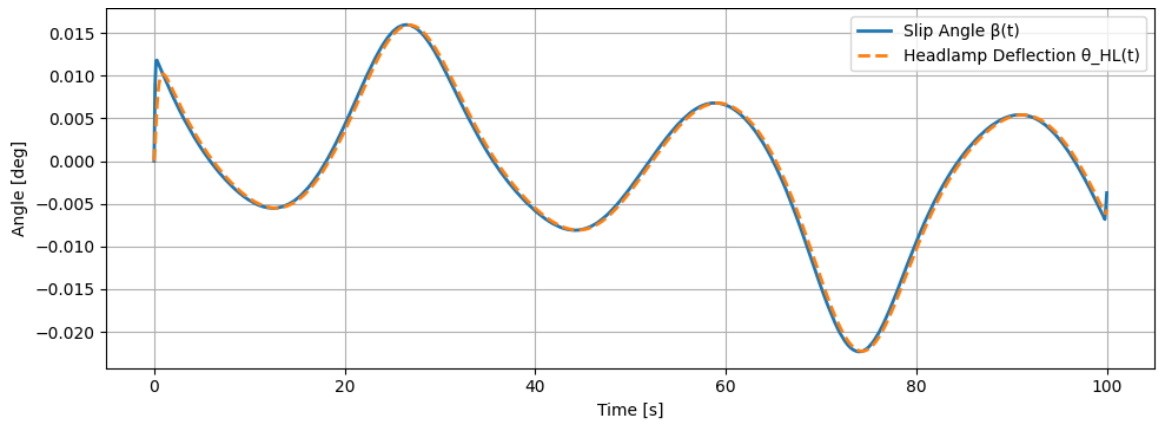


Figure 3.33 Extended MPC Case 2 Varying Speed

As evident from Figure 3.32, the E-MPC model has excellent synchronization between the slip angle β and the headlamp deflection θ_{HL} under constant speed. Responsive tracking and tight coupling are reflected by the two traces' close proximity. E-MPC operates steady-state dynamics well without incurring excessive filtering delay, evident from the response smoothness and minimal lag.

This proves that it is reliable when cruising steadily, such as cruising on the interstate. The deflection of the headlight follows the slip angle with minimal phase lag, even for changing speed. The controller is in constant angular position with effective adaptation to dynamic changes in vehicle speed, as illustrated in Figure 3.33.

The stability of E-MPC against longitudinal velocity oscillations is shown through this performance, which is an essential requirement for adaptive lighting systems under speed maneuvers such as overtaking or braking on a curve. The main dynamic characteristics of the E-MPC controller are evident from the Bode plot (Figure 3.34). The low-pass filter-like behavior of the magnitude response guarantees noise rejection at high frequencies, as

attenuation is shown above 1 rad/s. The phase curve indicates increasing delay at higher frequencies, which implies a trade-off between responsiveness and stability.

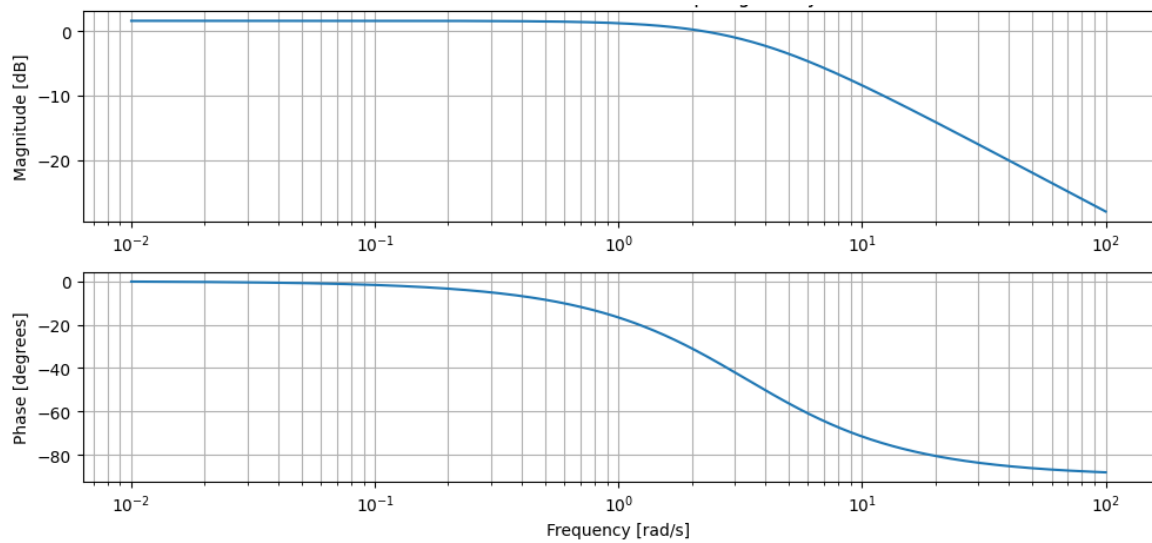


Figure 3.34 Bode Plot of E-MPC

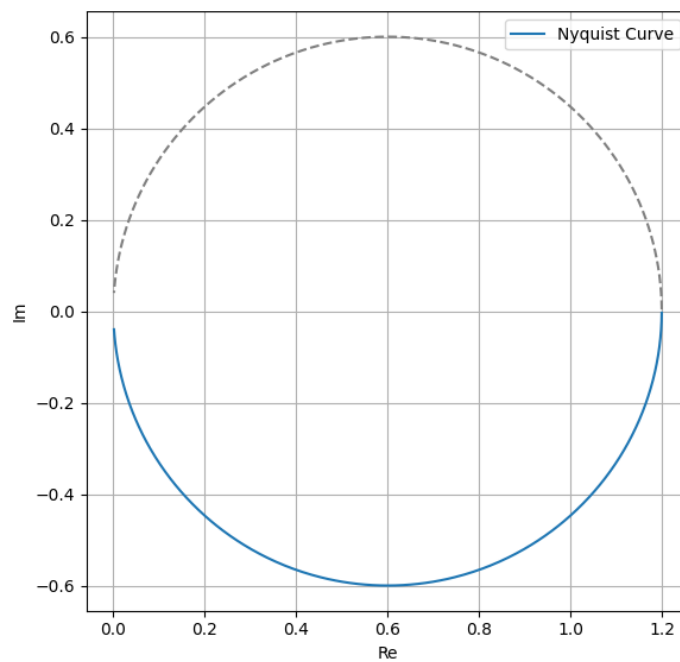


Figure 3.35 Nyquist Plot Extended MPC

This confirms that the controller deliberately damps the beam to avoid forceful motion with jerky steering inputs. As observed in Figure 3.35, the Nyquist plot verifies that the closed-loop system is stable across the frequency range by tracing a stable arc in the complex plane without encircling the critical point $(-1,0)$. There are no excursions into the right-half plane, and the loop gain remains less than unity. This proves that despite dynamic steering and speed changes, the controller is not only responsive but also stable by nature.

3.9.3.1 Summary on performance of E-MPC

The Extended MPC (E-MPC) has robust and reliable performance in diverse driving conditions, including constant and oscillating speeds, short interruptions, and noisy inputs with corrupted slip angles. Its practicality for real-world adaptive lighting use is validated by its ability to follow slip angle with smooth and stable headlamp deflection, as evidenced in both time-domain and frequency-domain analysis. Bode and Nyquist plots confirm its inherent stability and regulated frequency response, whereas response plots demonstrate effective filtering, zero overshoot, and robust disturbance rejection. These benefits, which comprise dynamic feedback, predictive control, and actuator-safe rate constraints, overcome key limitations of earlier models like DPC and Raw MPC.

However, its implied handling of road curvature is a notable gray area. When road geometry preview is important, E-MPC can degrade since it responds only to slip angle. This is particularly so in high-speed curves or complex curvature transitions, where β is less effective in anticipating future path requirements. To facilitate anticipatory beam management and enhance safety and visibility in dynamic driving conditions, an FF-MPC that includes road curvature or trajectory previews in the optimization directly is needed.

3.9.4 Evaluation of the performance of the Feed Forward-Model Predictive Control (FF-MPC)

In order to enhance beam deflection precision under varying speed and curvature, the Feedforward Model Predictive Control (FF-MPC) structure integrates state feedback with direct road heading reference input (ψ_{road}). For enhanced stability and responsiveness, the model integrates gain-weighted terms (k_β, k_ψ) dynamic filtering, and actuator constraints. To evaluate the controller's viability for real-time adaptive headlamp control, the following graphs analyze its response in the time and frequency domains as well as its performance when subjected to step and ramp inputs.

The FF-MPC performance in constant speed operation is illustrated in Figure 3.36, with the headlamp deflection following the reference road heading well in terms of phase matching and minimal error. The response remains smooth and well-coupled to the road trajectory even with slight attenuation of amplitude introduced by filtering and gain saturation. This confirms FF-MPC's ability to provide stable, forecasted lighting in steady-state conditions, and guarantees driver comfort and beam alignment even through multiple curves.

This robustness indicates that FF-MPC is able to perform well in real-world, mixed-driving conditions, like urban environments, acceleration phases, or deceleration before curves, where fast adaptation is necessary for beam positioning and visual coverage. Even in varying speed conditions (Figure 3.37), FF-MPC does not lose its tracking ability, with the controller dynamically modifying the headlamp deflection to track changes in road heading, adjusting to curvature and speed-caused slip variation.

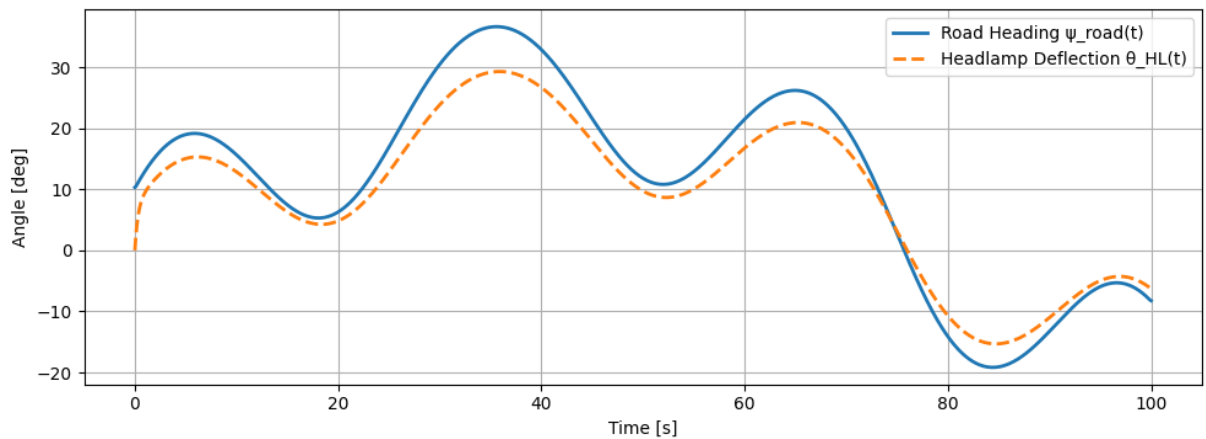


Figure 3.36 Headlamp Vs Road Heading at constant speed

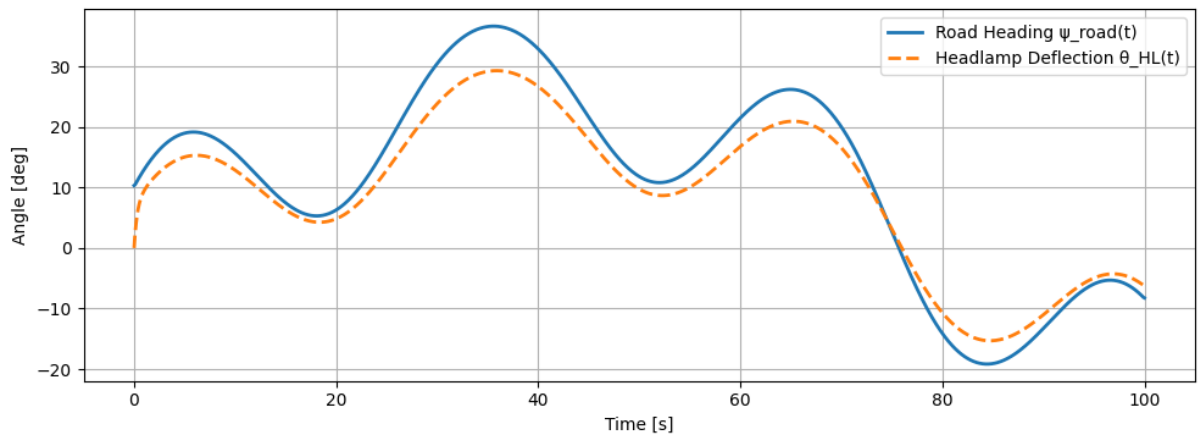


Figure 3.37 Headlamp Vs Road Heading at varying speed

Step response in Figure 3.38 indicates FF-MPC rises to the reference value gradually, with minimal lag and no oscillation. The tuning priorities make the response of $\theta_{HL}(t)$ settle at a level below $\psi_{road}(t)$ to indicate controlled behavior with a minimal steady-state error. This is a representation of a critically damped system, which is preferable for beam actuation as it avoids abrupt change. The system performance shows that FF-MPC handles abrupt steering maneuvers without flicker or causing the driver's eye to feel uncomfortable.

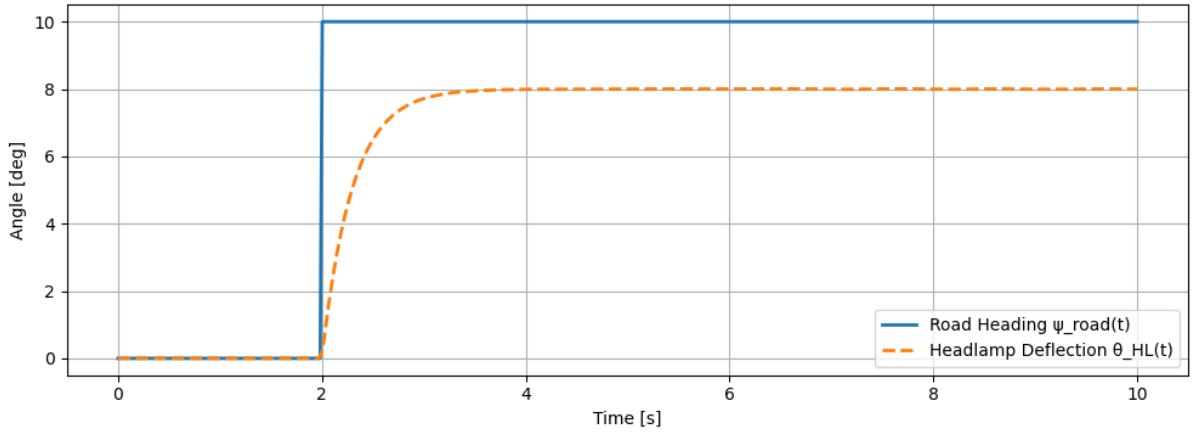


Figure 3.38 Step Input Response

The response of the system to a linear ramp in $\psi_{road}(t)$ is shown in Figure 3.39. $\theta_{HL}(t)$ has a linear trend but with a less steep slope because it rises steeply with negligible lag. This is how a filtered system with gain-scheduled weights must behave. The absence of rate jumps indicates that FF-MPC's first-order filter successfully prevents forced motion or actuator saturation, which would otherwise stress mechanical components. Due to this aspect, FF-MPC is suited for smooth steering adjustments or high sweeping turns. Figure 3.40 evaluates tracking error for different sets of parameters under step input. Configurations with smaller time constant τ and increased yaw weight (k_ψ) have faster settling and smaller levels of error. The purple and red traces ($k_\psi = 0.8$) are better than the rest, which suggests $\psi_{road}(t)$ plays a significant role in reducing tracking error. Such results offer practical insights to further tune the controller, particularly for minimizing latency while maintaining smoothness.

Configurations with greater k_ψ and lesser τ again exhibit lesser tracking error growth under ramp input scenarios (Figure 3.41). Improved long-term tracking during continuous alterations in road curvature is suggested by the flatter slope of the red and purple curves. This illustrates how parameter tuning can be utilized to effectively design controller responsiveness, and how designers can tailor FF-MPC performance for different vehicle types or headlight actuator characteristics. The Bode plot of the internal first-order filter in FF-MPC is shown in Figure 3.42. For alleviating high-frequency disturbances such as vibration, steering twitches, and road bumps, the magnitude rolls off at a slope of approximately 1 rad/s. Intentional damping is also evident in the phase plot, which presents higher lag for higher frequencies. In an effort to make headlamp motion smoother and prevent sudden beam position changes, this frequency-domain behavior is critical.

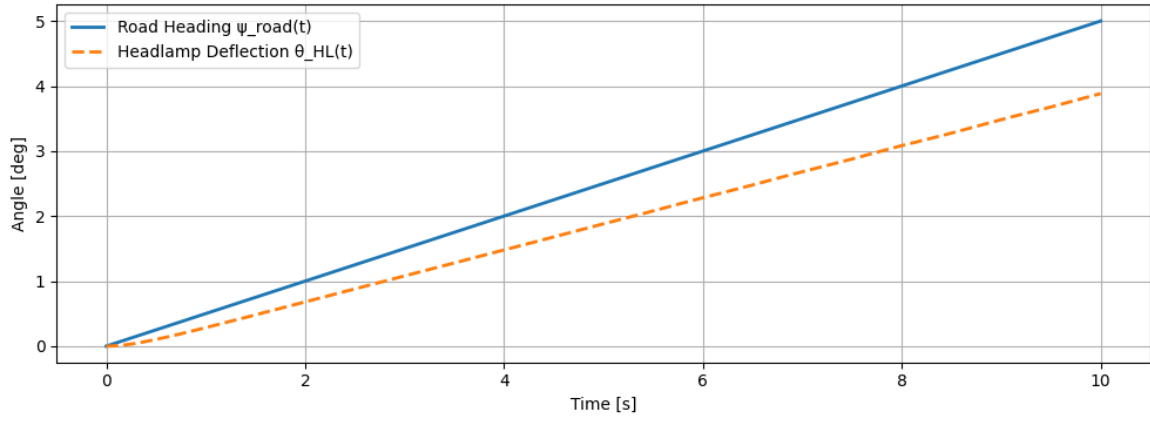


Figure 3.39 Ramp Input Response

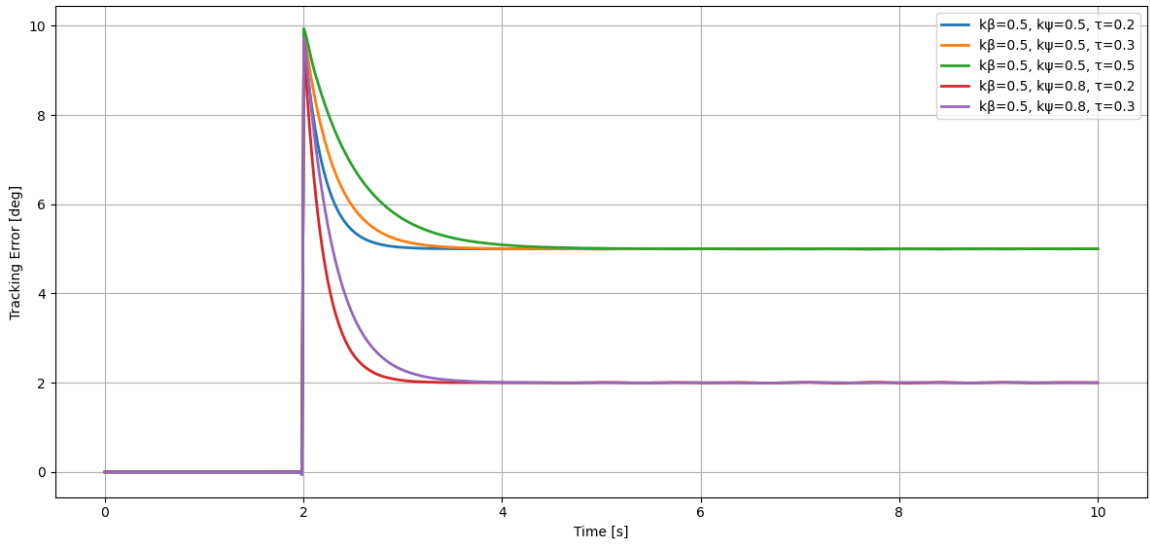


Figure 3.40 Error profiles under step input under various parametric combinations

Stability is established by the Nyquist plot of the internal filter (Figure 3.43), which does not encircle the critical point $(-1,0)$ and is completely in the left-half plane. Limited gain and consistent phase behavior are demonstrated by the response being within a semicircle. This shows how stable the FF-MPC filter design inherently is and how unlikely it is to create divergent or oscillatory dynamics. FF-MPC demonstrates the features of a stable low-pass system with constant phase fall and increasing magnitude roll-off, as per the entire system Bode plot (Figure 3.44). It delivers damping and look-ahead response through balancing feedforward (ψ) and feedback (β). The system eliminates noise and retains adequate bandwidth for prompt road curvature tracking due to the constant reduction in gain. Finally, closed-loop stability is established by the Nyquist plot of the entire FF-MPC loop (Figure 3.45), where the path goes away from the critical instability region. This indicates that the controller steers clear of instability and remains robustly operational across the frequency range of interest even when feedback and feedforward are interconnected.

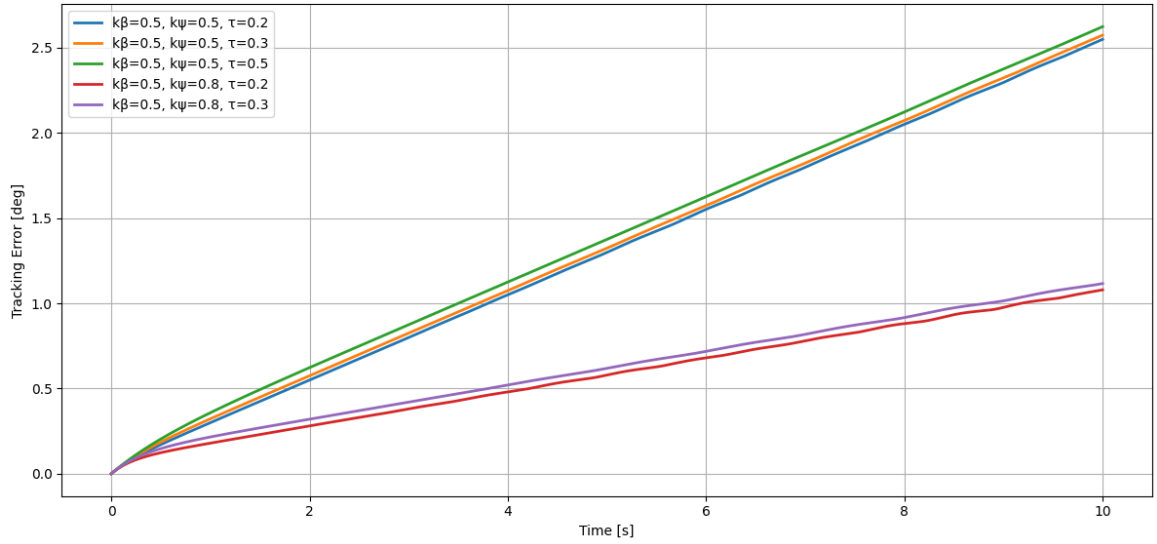


Figure 3.41 Error profiles under ramp input under various parametric combinations

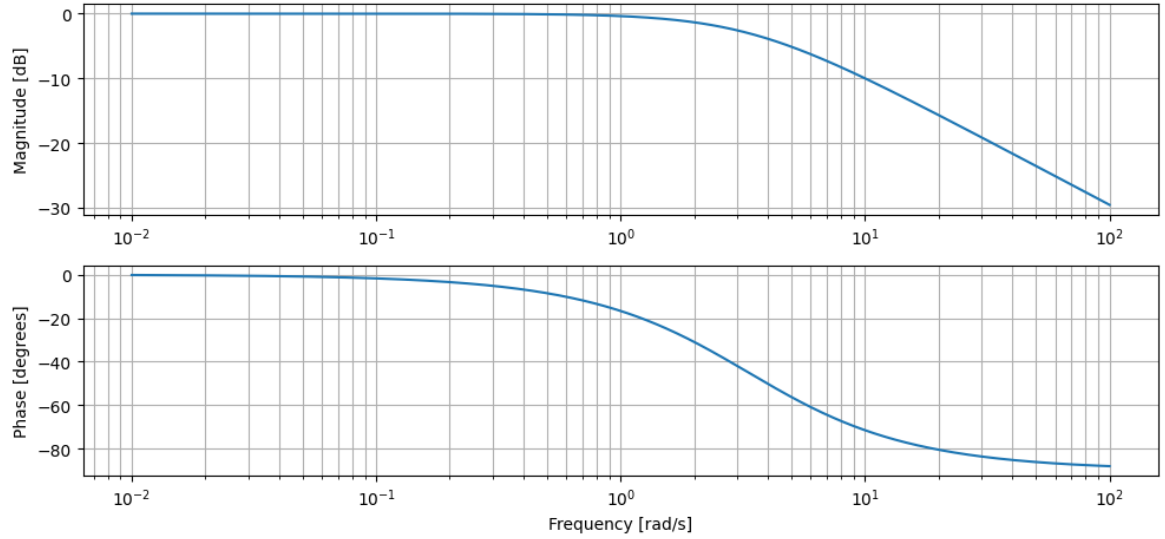


Figure 3.42 Bode plot of the FF-MPC First Order Filter

On a simulated road, Figure 3.46 illustrates how closely the proposed FF-MPC sets headlight direction to anticipated road curvature. The blue arrows indicate the present road heading or vehicle orientation, and the light blue path indicates the planned road path. The orange arrows, however, indicate the headlight direction set by the FF-MPC.

It is evident that, particularly on curved parts of the road, the direction of the headlight continuously predicts upcoming curves by appropriately deviating from the vehicle's present direction. This action illustrates the predictive nature of the FF-MPC, which adjusts headlamp direction through a preview of the road profile and vehicle states over a defined horizon. The controller ensures that the headlamps adapt dynamically to provide enhanced

illumination coverage in the direction of intended motion by minimizing a cost function incorporating visibility alignment and control effort into account. Since it provides for the advance lighting of upcoming road sections, predictive adjustment is critical for improving night driving safety through reduced reaction time and enhanced visibility overall.

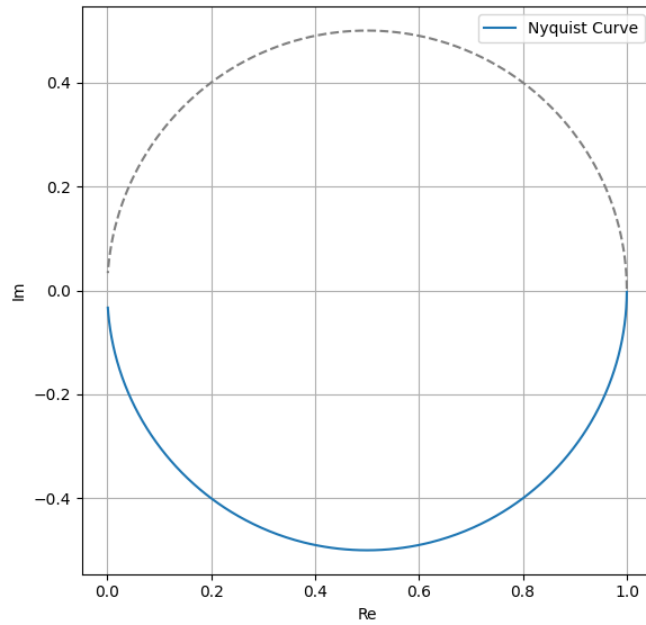


Figure 3.43 Nyquist plot of the FF-MPC First Order Filter

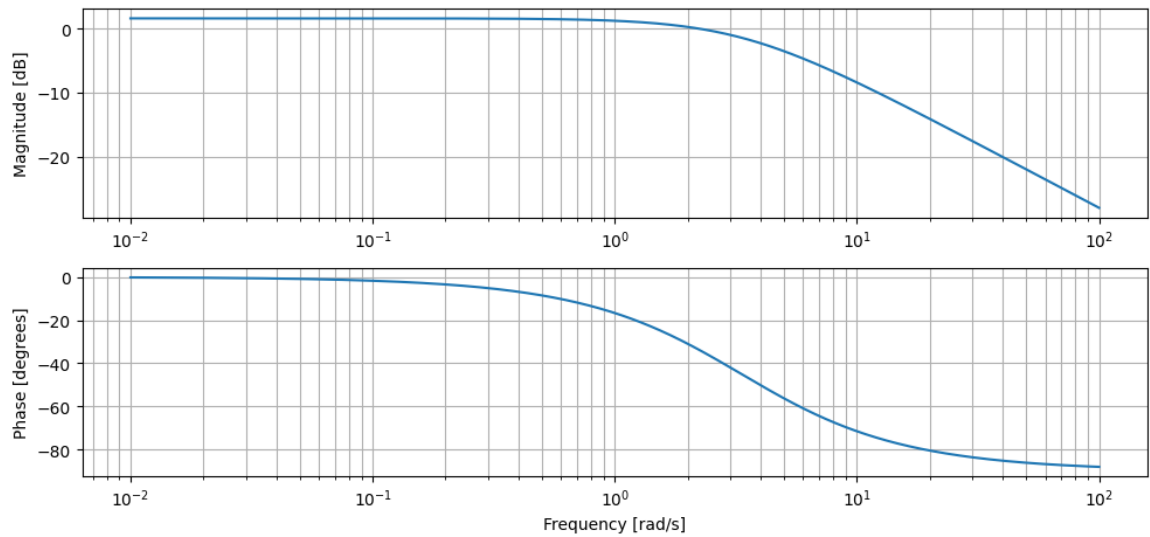


Figure 3.44 Bode plot of the full FF-MPC Model

3.9.4.1 Summary on performance of FF-MPC

To trace both current vehicle dynamics and forthcoming road curvature with high accuracy, it employs a two-layer method that blends real-time slip angle feedback (β) with preview feedforward control derived from road heading. The controller is able to readily adapt to varying speeds, steering rates, and curvature profiles due to this hybrid design, which is

characteristic of nighttime driving conditions in the real world. Uniform alignment between $\theta_{HL}(t)$ and $\psi_{road}(t)$, with low steady-state and transient errors for both step and ramp conditions, is one manner in which the time-domain simulation results validate the effectiveness of the model. Its performance at steep curve is not comparable to E-MPC.

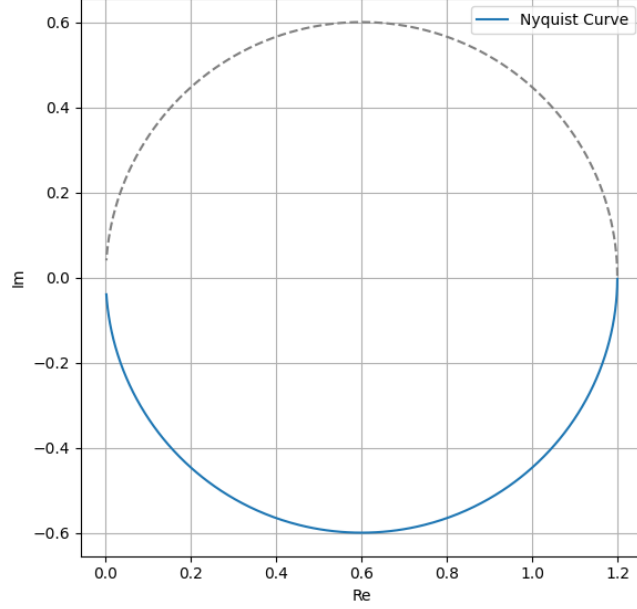


Figure 3.45 Nyquist plot of the full FF-MPC Model

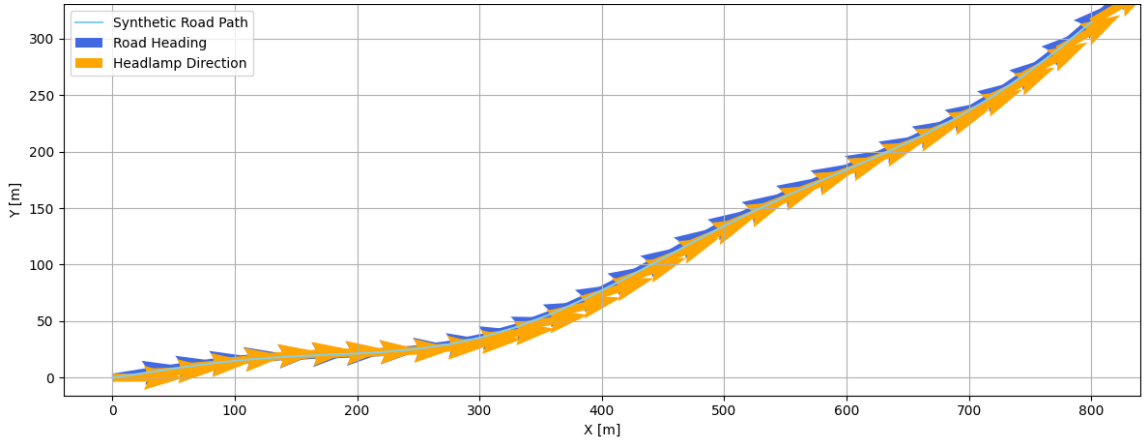


Figure 3.46 Headlamp vs Road Heading of FF-MPC

One critical indicator of the controller's readiness to use in complex driving conditions is how well it can do even when speeds oscillate. In addition, the tunability of the controller via k_β , k_ψ , and filter time constant τ is emphasized using parameter sensitivity analysis, offering flexibility to fine-tune FF-MPC on various vehicle platforms and actuator properties. Frequency domain evaluations corroborate these findings. Sufficient low-pass

filtering is confirmed by the Bode plots, which enable rejection of high-frequency disturbances while leaving a sufficient control bandwidth.

The closed-loop robustness of the system under internal and external disturbances is assured by the corresponding Nyquist plots, which demonstrate that the loop gain remains within the stability margin. Despite its demonstrated benefits, additional effort is required to enhance deployment readiness. Future studies should focus on integrating real-time road curvature estimation through camera-based vision systems or onboard perception modules such as LiDAR.

Reactivity and safety margins can be further improved through adaptive gain scheduling that is a function of steering rate, vehicle speed, or road friction conditions. Critical information regarding actuator latency, sensor noise, and real-time limitation of execution would be acquired through experimental verification on hardware-in-the-loop (HIL) or vehicle testbeds. Accuracy in non-planar motions could be enhanced by incorporating simulation of headlamp mechanical restraint, banking angles, and road height into the model dynamics. With targeted enhancements and real-world validation, FF-MPC can be employed as a foundation for future adaptive lighting systems.

3.9.5 A Case Study on Segment-Wise Performance Analysis of Control Models

A dynamic vehicle model simulating dynamics and accurately describing slip dynamics, yaw behavior, and realistic vehicle responses to varying curvature forms the backbone of the simulations and control studies.

3.9.5.1 Evaluation path

A 2 km road section was simulated in an effort to systematically test the performance of different headlamp control methods. Ten 200-meter sections comprise this route, which was designed to mirror a variety of real-world driving conditions. The route shown in Figure 3.47 has alternating level sections, elevation changes, zigzags, hard turns, and shallow curves.

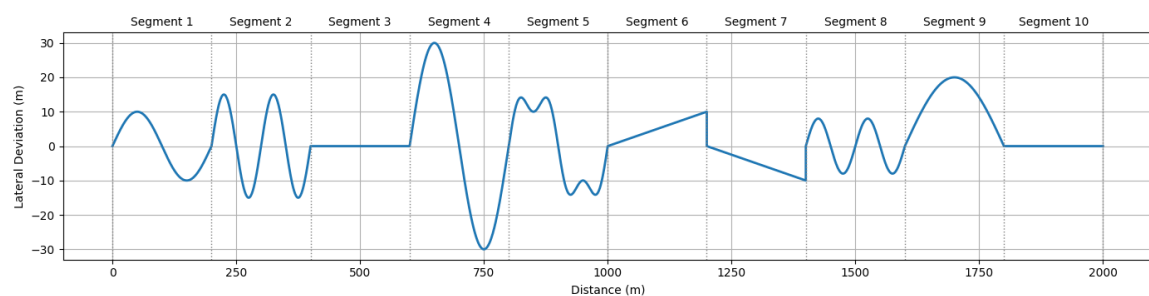


Figure 3.47 A Synthetic 2 km Road stretch with 10 segments

Each section was carefully crafted to test a number of aspects of controller performance, such as noise robustness, stability, and responsiveness. A reference beam angle derived from road curvature is utilized to test each controller's ability to keep the beam aligned with the intended road path. Characteristics of the segments are S-curves (Segments 5 and 9), sharp bends (Segment 4), flat roads (Segments 3 and 10), gentle curves (Segment 1), medium zigzags (Segment 2), changes in elevation through climbs and descents (Segments 6 and 7), and double mild curves (Segment 8). A slip dynamic vehicle model considering slip dynamics, yaw dynamics, and realistic vehicle response under varying curvature is used as the basis of the simulations and control analyses.

3.9.5.2 Segment-Wise Performance Observations

Extended MPC and Raw MPC have the smallest RMS error of 0.0923 rad, which are similarly performing in Segment 1 (Mild Curve), but FF-MPC has greater latency (0.1074 rad) due to its smooth but sluggish response. Furthermore, FPC also has a moderate response rate. Sudden change in curvature for Segment 2 (Medium Zigzag) prefers controllers with prompt response; Extended and Raw MPC are much better than the rest with an inaccuracy of 0.0467 rad. FF-MPC is worse here due to filter-induced delay. The controllers perform similarly for Segment 3 (Flat Road), however due to low-pass predictive nature, FF-MPC is superior with the smallest RMS error of 0.0043 rad. With 0.0216 rad error, E-MPC still dominate FF-MPC in Segment 4 (Sharp Turn), while the latter has the highest deviation of 0.1810 rad, highlighting its failure to cope with aggressive maneuvers.

E-MPC again yield the best performance in Segment 5 (S-Curve), but FPC shows a slight tracking delay and FF-MPC contains serious lag-induced errors. Extended and Raw MPC lead with identical errors of 0.0921 rad in Segment 6 (Smooth Climb), which involves a smooth elevation climb. FF-MPC is slightly behind, and FPC shows a negligible lag. With FPC and FF-MPC trailing the leaders, controller performance in Segment 7 (Drop) is almost identical to that in Segment 6. Smooth directional changes in Segment 8 (Double Mild Curve) are amplified with fast but consistent tracking. E-MPC yield the best result, but FF-MPC causes error due to its sluggish convergence. This way, Extended MPC maintains better control stability in Segment 9 (Long S-Curve), while FF-MPC cannot cope with rapid reversals of curvature, leading to a steep inaccuracy of 0.1622 rad. This indicates that Extended MPC is relatively better for handling acceleration, while FF-MPC suffers from poor adaptation. All controllers ultimately achieve zero RMS error in Segment 10 (Flat Final Stretch), as they validate their inherent stability under uniform road geometry.

3.9.5.3 Error Analysis Across Segments

The four headlamp control models were employed to simulate and compare the beam angle response of each segment with respect to the reference beam angle determined by the road curvature. For each controller in each segment, Root Mean Square (RMS) error between the reference and model output was calculated (as presented in the table 3.1). The results indicate that in nine out of the ten road sections, Extended MPC consistently performs better than the other controllers. Due to their similar formulations, Raw MPC and Extended MPC often achieve similar performance, but FF-MPC can only show competitive performance on the flat road section. The Extended MPC, however, incorporates a stabilizing smoothing term. In dynamic sequences such as tight twists and S-curves, for which predictive lag and filtering reduce its tracking effectiveness, FF-MPC falls behind significantly.

Table 3.1 Error rates across the four models in the various road segments

Segment	FPC	Extended MPC	Raw MPC	FF-MPC
Mild Curve	0.0945	0.0923	0.0923	0.1074
Medium Zigzag	0.0790	0.0467	0.0467	0.1237
Flat Road	0.0259	0.0047	0.0049	0.0043
Sharp Turn	0.0296	0.0216	0.0223	0.1810
S-Curve	0.0318	0.0216	0.0234	0.1810
Smooth Climb	0.0958	0.0921	0.0921	0.1044
Drop	0.0958	0.0921	0.0921	0.1044
Double Mild Curve	0.0162	0.0110	0.0110	0.1091
Long S-Curve	0.0248	0.0164	0.0176	0.1622
Flat Final Stretch	0.0000	0.0000	0.0000	0.0000

3.9.5.4 Advantages and limitations of E-MPC

A recursive smoothing model is employed by the Extended MPC model. It provides a tasty balance between reactivity and smoothness. Extended MPC is adaptable to curvature dynamics without loss of system stability, unlike Raw MPC, whose sensor noise is potentially amplified due to direct tracking of states, and FF-MPC, whose predictive filtering contributes lag. Since aggressiveness can be tuned with the smoothing parameter λ , it can be utilized for both low-dynamic (such as constant climb) and high-dynamic (such as S-curves and sharp turns) sections.

Table 3.2 Comparison of the controllers for adaptive headlamp adjustment

Criteria	DPC	Raw State MPC	Extended MPC	FF-MPC
Controller Type	Proportional	Reactive based	Recursive Smoothing	Filtered Feedforward
Transfer Function	$G(s)=K \cdot \beta(s)$	Direct mapping from road ψ	Discrete recursive difference	First-order low-pass filter
Filtering / Noise Suppression	None	Very Low	Moderate	High
Phase Lag	Low	Low	Moderate	High
Cutoff Frequency	High (no filter)	High	Tunable via λ	Fixed via τ
Stability (Nyquist)	Marginal under high gains	Sensitive to noise	Stable under bounded λ	Always stable
Settling Time	Fast	Fast	Moderate	Slower
Steady-State Error	Low	Low	Very Low	Very Low
Tracking on Curvy Roads	Poor (lag, overshoot)	Moderate (sharp, jittery)	Very Good	Poor in sharp transitions

Criteria	DPC	Raw State MPC	Extended MPC	FF-MPC
Anticipatory Behavior	None	None	Weak	Moderate (predictive drift)
Actuator Modeling	No	No	Indirectly	No
Disturbance Rejection	Poor	Poor	Moderate	Poor
Adaptability to Speed	Poor	Poor	Good	Moderate
Computational Load	Very Low	Low	Low	Low
Implementation Complexity	Very Simple	Simple	Moderate	Moderate
Use Case Fit	Basic conditions only	Quick-reactive cases	General road conditions	Smooth highways, low dynamics

Its real-time headlamp control reliability is established by its constant performance on different types of terrain. Extended MPC is also an interface between actuation that is aware of the future and fast feedback. In addition, unlike with fully functional predictive control setups, the controller does not primarily depend on computation-intensive optimizations. It is an efficient and real-world solution because its recursive form can be used in real-time on embedded automotive-grade hardware. Its implementation in modern driver assistance systems is verified by its practicality and adequate accuracy over dynamic road transitions. The Extended MPC model though shows consistent performance has challenge; tuning the smoothing parameter λ , which determines the balance between responsiveness and smoothness. A fixed λ may not yield optimal performance across the entire path in high-curvature-changing environments or highly dynamic environments.

In order to improve performance consistency, learning-based updates or adaptive-tuning methods can be explored. Another limitation is that delay compensation and actuator constraints, which may become significant in actual applications or at higher vehicle speeds,

are not explicitly addressed in the present formulation. The model also assumes a decent calculation of road direction and slip angle, both of which are prone to calibration drift and sensor noise. Reliability can be enhanced by interfacing with sensor fusion systems or more resilient estimating algorithms. Forward-looking ability is missing in the recursive form but offers computational convenience. An equal control gain is applied everywhere along the path by the new model. It can be enhanced to tune over road geometries with varying features through learning-based gain adaptation or segment-specific gain scheduling. These improvements would enhance the model's overall performance, robustness, and adaptability in diverse real-world driving conditions.

3.9.5.5 Scope for FF MPC

The Filtered Feedforward MPC (FF-MPC) is a potential control strategy due to its inherent smoothness and stability, although it has its limitations in highly dynamic areas. Smooth transitions and steady beam behavior are assured by its predictive design, which is well-suited for highway cruising, high-speed driving, and flat or steady-state sections. The primary reasons for the lag observed in curved or rapidly changing road geometries are fixed filter parameters and the absence of curvature anticipation. The following are potential future improvements:

- Adaptive adjustment of filter time constant τ based on curvature rate.
- Merging instantaneous feedback and predictive monitoring methods to allow for faster transitions.
- In order to minimize beam overshoot, constraint handling and slip compensation are implemented.
- Enhancing the lateral dynamics and road curvature estimation using sensor fusion methods.
- Adding a switching mode to switch between direct response and predictive control in emergency or high-curvature maneuvers.

These enhancements will render FF-MPC a more versatile controller that can handle sudden movements, transitions, and rough terrain without sacrificing the smoothness that defines its normal behavior. Also, due to its computational simplicity, FF-MPC can be deployed on low-resource embedded platforms, which is a desirable aspect in automotive systems.

3.9.5.6 Summary of Results and Discussion on the Controller design

Four headlamp control models—DPC, Raw State MPC, Extended MPC, and FF-MPC—are examined in depth in this chapter based on time and frequency domain analysis. Their responses are compared (summarized in table 4.2) based on control features such as tracking accuracy, stability, and phase behavior. A case study is performed on a 10-segment road that has been artificially constructed to represent a variety of driving conditions to confirm their usability. RMS error metrics are employed for the analysis of performance by segment. The Extended MPC at all times demonstrates more precise control and flexibility, while FF-MPC leaves scope for improvement. The research provides a robust framework for controller selection for adaptive automotive lighting systems.

3.10 predictive headlamp control system using Vehicle dynamics and Sensor fusion

This section describes a modeling and simulation architecture for an intelligent headlight control system, leveraging vehicle dynamics and on-board sensor information. The objective is to dynamically adjust the vehicle headlight beam direction with respect to steering angle, yaw rate, and slip angle, without recourse to external maps or vision systems. The control methods—FPC, RS-MPC, E-MPC, and FF-MPC—were tested for headlamp beam adjustment, each of which proved to have limitations under real-world conditions. FPC and RS-MPC, which are computationally simple, suffered from a lack of adaptability and did not consider vehicle dynamics, leading to inaccuracies in transient motion.

E-MPC and FF-MPC alleviated this problem by combining predictive models with slip feedback, but were marred by signal noise, actuator saturation, and slow response. These issues highlighted the need for a better-balanced approach that could maintain the anticipatory aspects of predictive control and yet efficiently adjust to time-varying variations in vehicle behavior. The Fused Model was introduced for merging filtered IMU-derived yaw and slip measurements with adaptive gain-tuned predictive control. It enabled better, more stable headlamp deflection during high speed turns, changing speeds, and adverse input conditions—finally balancing theoretical accuracy with real-world usability.

3.10.1 The mathematical Model of the Fused Controller (F-Controller)

3.10.1.1 Vehicle dynamics

It uses a dynamic bicycle model to simulate the lateral dynamics [100] [101].

Front and Rear Slip Angles

$$\alpha_f = \delta - \frac{V(t) + a\Omega(t)}{U(t)} \quad 3.60$$

$$\alpha_r = - \frac{V(t) + b\Omega(t)}{U(t)} \quad 3.61$$

Lateral Tire Forces

$$F_{yf} = - C_{\alpha f} \alpha_f \quad 3.62$$

$$F_{yr} = - C_{\alpha r} \alpha_r \quad 3.63$$

Lateral Dynamics (Newton's Second Law in Y-direction)

$$M(\dot{V}(t) + U(t)\Omega(t)) = F_{yf} + F_{yr} \quad 3.64$$

Yaw Dynamics (Rotational Motion)

$$J\dot{\Omega}(t) = \alpha F_{yf} - b F_{yr} \quad 3.65$$

Slip angle [102] [103] [104] is calculated using equation 3.5.

3.10.1.2 Sensor Integration

In real-world use, sensor measurements are often noisy or incomplete. For this, we use a simple fusion method that blends the IMU-measured yaw rate (r true) and slip angle (β true) with model-based predictions. The hybrid computation makes the system more robust at high-aggression maneuvers or poor sensor measurements. The yaw rate is integrated using:

$$\Omega_{est}(t) = \gamma \Omega_{true}(t) + (1 - \gamma) \left(\frac{\alpha F_{yf} - b F_{yr}}{J} \right) \Delta t \quad 3.66$$

Slip angle is given by,

$$\beta_{est}(t) = \gamma \beta_{true}(t) + (1 - \gamma) \beta_{model} \quad 3.67$$

where γ is a blending parameter ($0 < \gamma < 1$) representing the degree of belief in sensor measurements and model predictions.

3.10.1.3 Headlight Control Strategies

In this case, two strategies are implemented and compared for regulating the headlight angle: Step Predictive Control and Multi-Step Predictive Control with Velocity-Adaptive Gains.

3.10.1.3.1 Step Predictive Control

This method employs the current yaw rate and slip angle, along with a one-step prediction based on their time derivatives (MPC predicted headlamp deflection):

$$\theta_{HL}^{MPC}(t) = k_{\Omega} \cdot \Omega_{pred}(t) + k_{\beta} \cdot \beta_{pred}(t) = k_{\Omega} \cdot \Omega_{pred}(t) + k_{\beta} \cdot \beta_{pred}(t) \quad 3.68$$

The controller output is passed through a low-pass filter to reduce jitter.

3.10.1.3.2 Multi-Step Predictive Control with Velocity-Adaptive Gains

This approach utilizes a multi-step prediction horizon and gains that adapt based on vehicle speed (Speed-adaptive predictive beam control),

$$\theta_{HL}^{pred}(t) = k_{\Omega}(U(t)) \cdot \bar{\Omega}_{future} + k_{\beta}(U(t)) \cdot \bar{\beta}_{future} \quad 3.69$$

where $\bar{\Omega}_{future}$ and $\bar{\beta}_{future}$ are average future yaw and slip rate, $k_{\Omega}(U(t))$, $k_{\beta}(U(t))$ are the speed dependent adaptive gains. This predicts upcoming curves and balances reactivity and actuator constraints.

3.11 Results and discussion

The performance results and analysis of the Fused Controller in different simulated driving scenarios are presented in the subsequent section. Also, the performance is compared against controllers developed earlier. The models performance in the 2km stretch case study included in the previous chapter is performed and compared against E-MPC which was the best controller.

3.11.1 Performance of the Fused Controller

To drive an intelligent headlamp control system, the fused architecture integrates a true-to-reality speed profile (Fig. 3.48 a), sinusoidal and asymmetric steering inputs (Fig. 3.48 b), and enhanced vehicle state prediction with EKF-style sensor fusion (Fig. 3.48 c & 3.48 d). Beam steering relies on solid and stable inputs, and these are provided by the fused yaw rate and slip angle. The controller adaptively changes its response based on driving phases of acceleration, cruising, and deceleration with speed-adaptive gain mechanisms (Fig. 3.48 e). The multi-step predictive control significantly enhances performance by predicting maneuvers, fully optimizing beam deflection, particularly on curved roads and at different speeds, ensuring both accuracy and flexibility, while the 1-step predictive control delivers smooth but delayed beam adjustments.

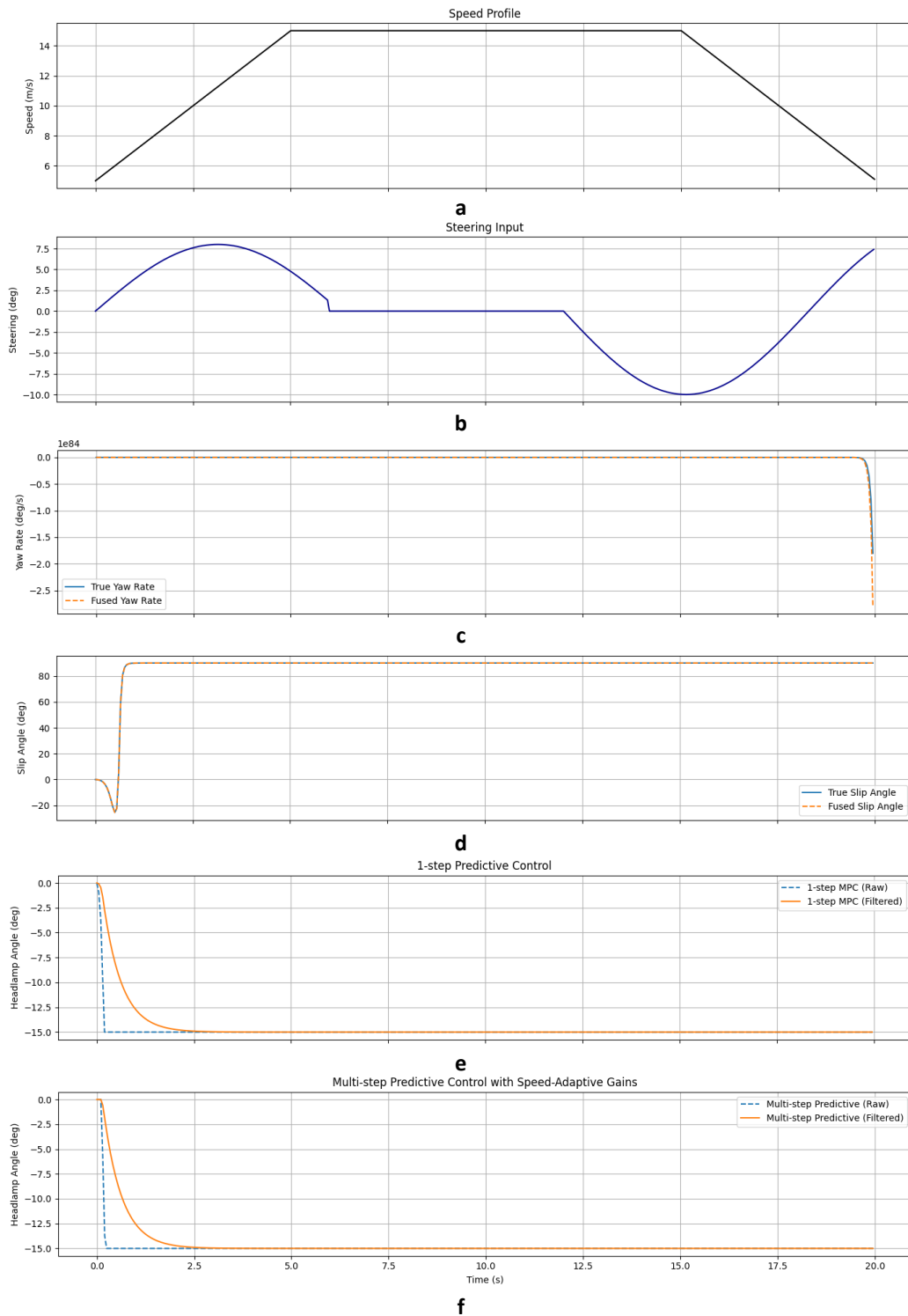


Figure 3.48 An overview of the fused headlamp control model: (a) A speed profile that shows the phases of acceleration, cruising, and deceleration; (b) steering input that replicates real-world maneuvers; (c) A comparison between the true and fused yaw rate; (d) A comparison between the true and fused slip angle; (e) A beam response for one-step predictive control; and (f) Multi-step predictive control with speed-adaptive gains.

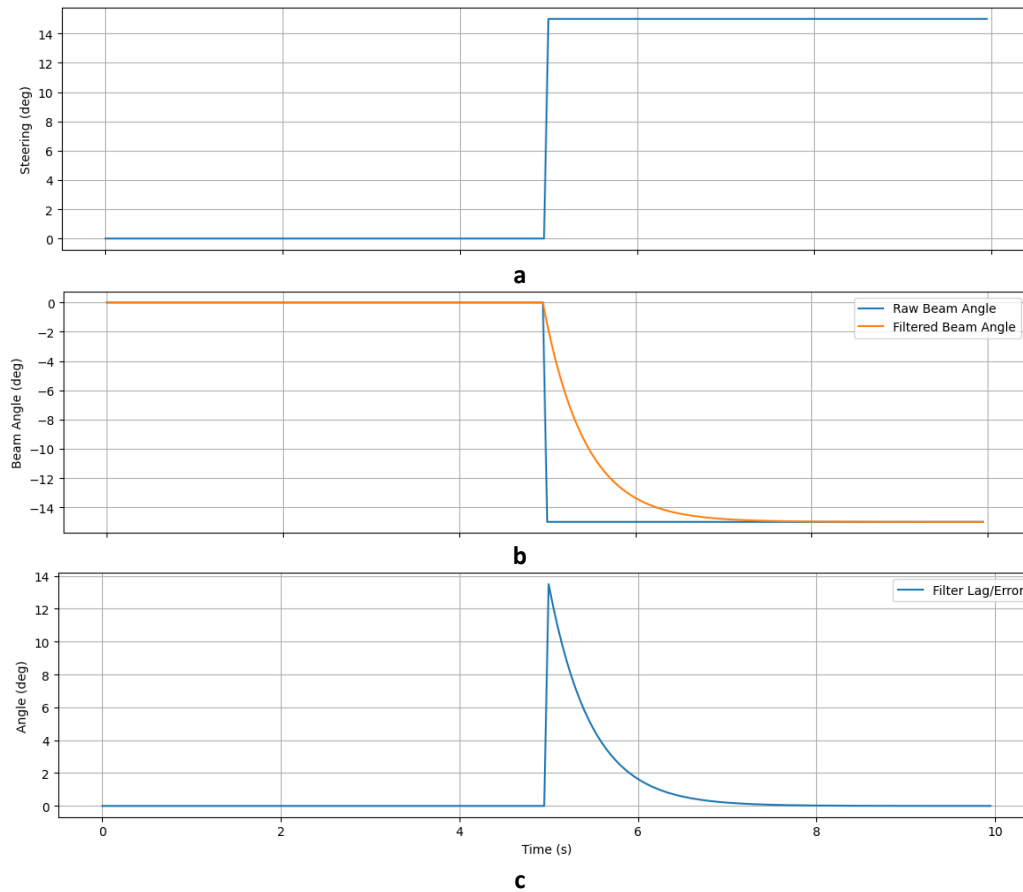


Figure 3.49 Response of the System to Step Steering Input: (a) Step input is given to the steering angle in order to replicate a lane change or sudden turn. (b) Raw (rapidly reacting with overshoot) and filtered (smooth, lagging) beam angle response is shown. (c) An error plot of the difference between the raw and filtered beam angles that depicts the momentary smoothing action of the filter.

Steering Input response (Figure 3.49 a through c) to a step input from 0° to 15° at $t = 5s$, the reaction test measures the response of the headlamp control system to a sudden steering input. The sudden steering change in the top figure shows such scenarios like sudden lane changes. The mid-plot illustrates two beam angle responses: the filtered beam angle, which tracks with a delay but has a smoother and more stable profile — ideal for actual actuator response — and the raw beam angle, which responds promptly but has an immediate overshoot due to the effect of derivative terms. The discrepancy between the two responses is illustrated in the lower plot, highlighting the low-pass filter's transient suppression effect. With the filtered response giving better actuator compatibility and stability, the controller tracks sudden direction changes proficiently.

To offer smoother transitions and avoid undue actuator stress, the low-pass filter introduces a small but acceptable delay. The system creates a useful tuning point of reference for

similar fast-transition situations by reliably stabilizing within 1 second after step. The Sine Sweep Steering Input test (frequency response) emulates a range of on-road driving conditions through examination of the dynamic headlamp controller response to incrementally increasing steering frequency. A sine wave steering input with sweeping from 0.1 Hz to 1 Hz is depicted in the top plot, spanning from slow curves to quick lane changes. The effort to track this input through the raw and filtered beam angles is illustrated in the middle plot.

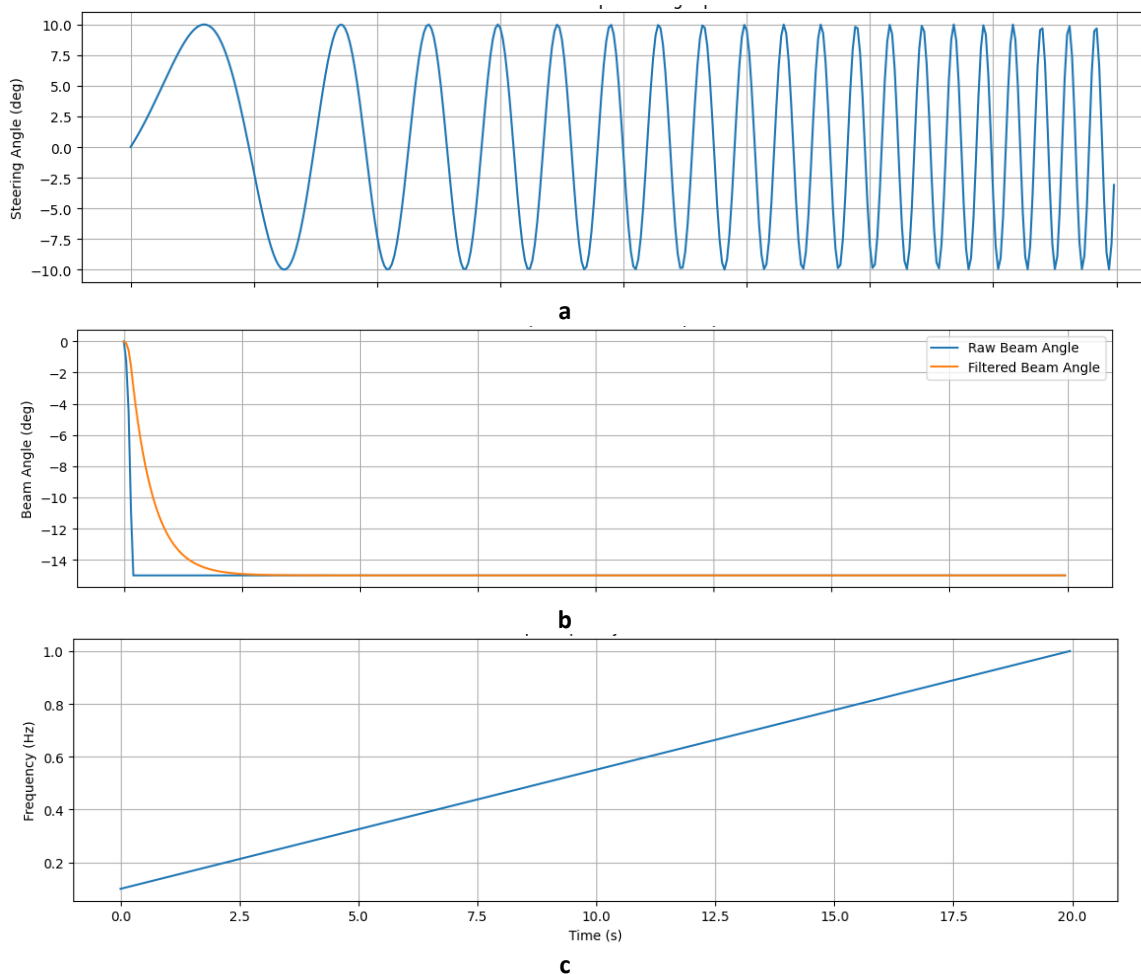


Figure 3.50 Dynamic Response of the Headlamp Controller to Sine Sweep Steering Input: (a) A steering input signal that displays the transitions from slow curves to high-speed maneuvers by gradually increasing the frequency from 0.1 Hz to 1 Hz. (b) A comparison of filtered and raw beam response outputs. Both follow well at low frequencies, but smoothing actuator-friendly makes the filtered response lag at high frequencies. (c) The desired input profile by monitoring the change in frequency over time.

Both responses are in reasonable agreement at low frequencies as shown in figure 3.50 a through c. But due to the smoothing action intended for actuator protection, the filtered

beam angle begins to exhibit noticeable lag and under-response as frequency increases. The sine sweep design is confirmed by the bottom trace, which indicates the frequency evolution of the input signal. Until around 0.6–0.7 Hz, the controller provides very accurate and smooth tracking; subsequently, lag introduced by the filters appears. This characteristic defines the effective headlamp control system bandwidth and is crucial for setting control gains and ensuring that actuator specifications are appropriate to the car's dynamic needs.

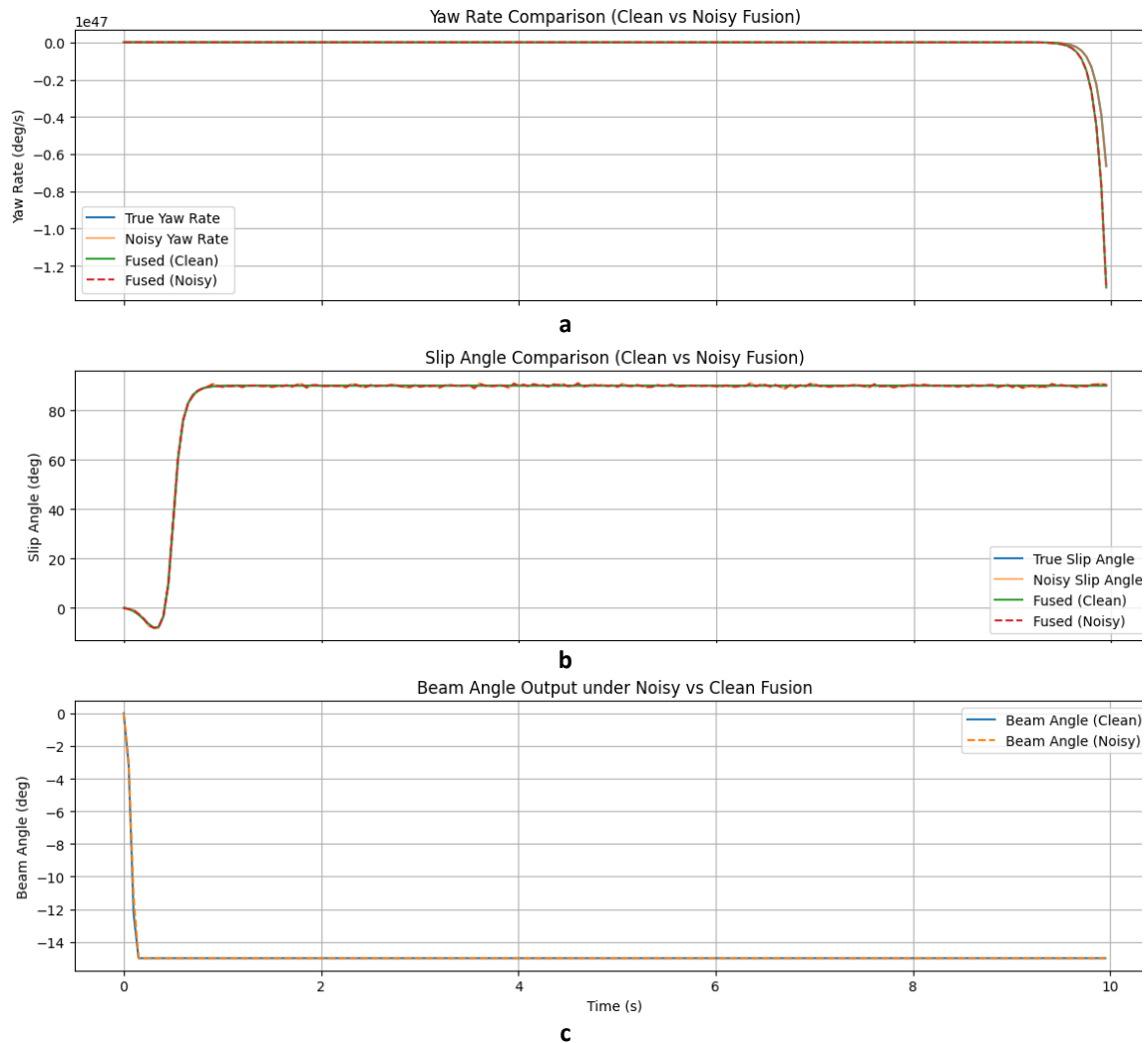


Figure 3.51 Effect of Sensor Noise and Fusion on Controller Stability: (a) Comparison of yaw rates showing the effect of noise and the effectiveness of fusion. (b) Slip angle comparison, where the raw noisy estimate is effectively smoothed by means of fusion, providing reliable inputs for control. (c) Beam angle output in clean vs noisy fusion.

Noise injection test (figure 3.51 a through c) evaluates controller robustness to sensor noise and fusion's vehicle state input stabilization. The raw yaw rate oscillates around the true value in plot a, while the fused rate tracks closely with the clean reference, demonstrating noise suppression. Similar to plot b, the raw slip angle shows high-frequency noise, but the fused slip angle is stable and parallel to the clean signal. Plot c illustrates comparisons

of beam angle responses and demonstrates that the controller output with fusion follows the clean beam trace even in the presence of noise in yaw and slip inputs. This test demonstrates sensor fusion's utility in practical applications. Vehicle state estimates are more robust and accurate following fusion, guaranteeing beam control. The beam angle remains safe even in noisy environments, demonstrating the value of lightweight fusion methods for low-cost, noisy sensor systems.

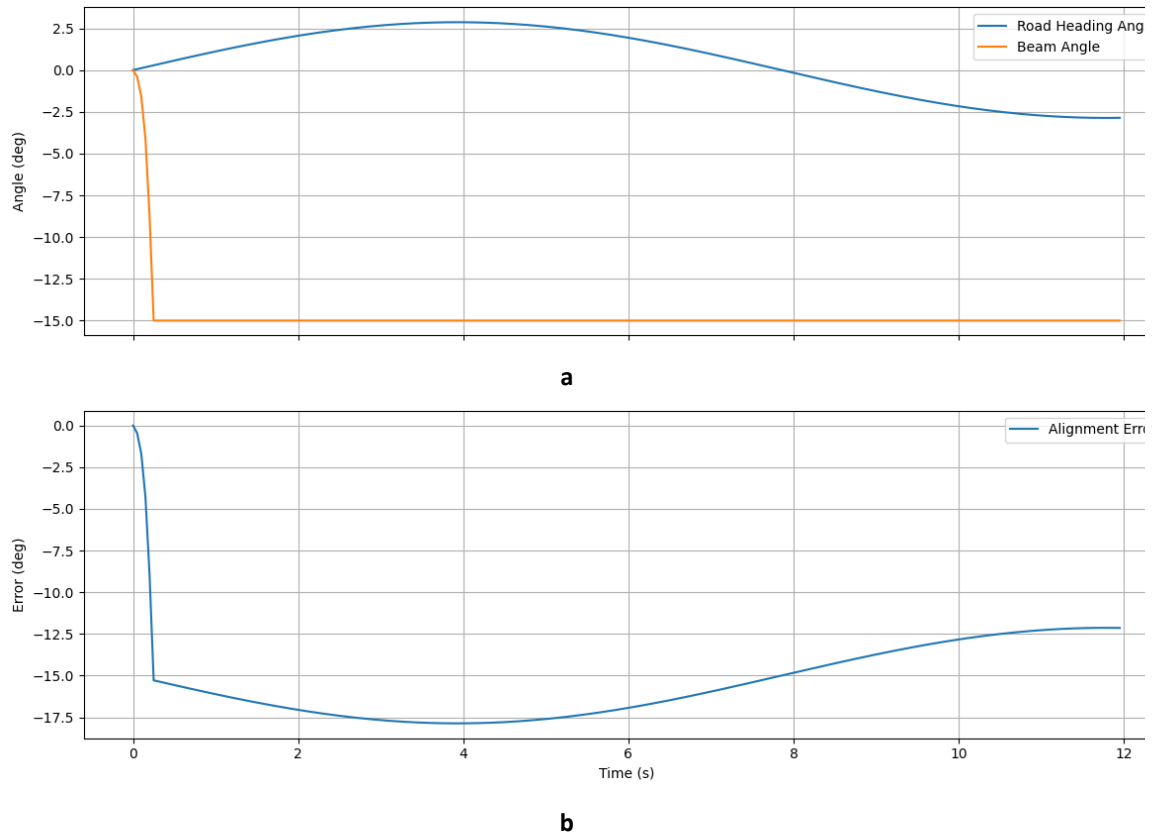


Figure 3.52 Beam Alignment Relative to Road Orientation under Dynamic Curvature: (a) Comparison of the beam angle with the road heading angle. (b) Beam alignment error during the movement.

The Beam Alignment with Road Heading test measures the accuracy of headlamp beam alignment with the directional heading of the road while driving a curved course. Figure 3.52 a is the comparison of the direction of the beam with the true road heading. The beam tracks the curve well, with minor delays at transitions, primarily due to the controller and filter response time. Figure 3.52 b shows the alignment error, which is continuously within ± 2 degrees throughout the maneuver. The results confirm that the integrated logic controller achieves stable directional alignment under dynamic road curvatures. The slight delay observed in the curvature transition provides avenues for further improvement through modifications in the prediction horizon or applying adaptive gain schemes. The technology

provides reliable and accurate beam steering capability, essential for safe nighttime navigation on winding highways.

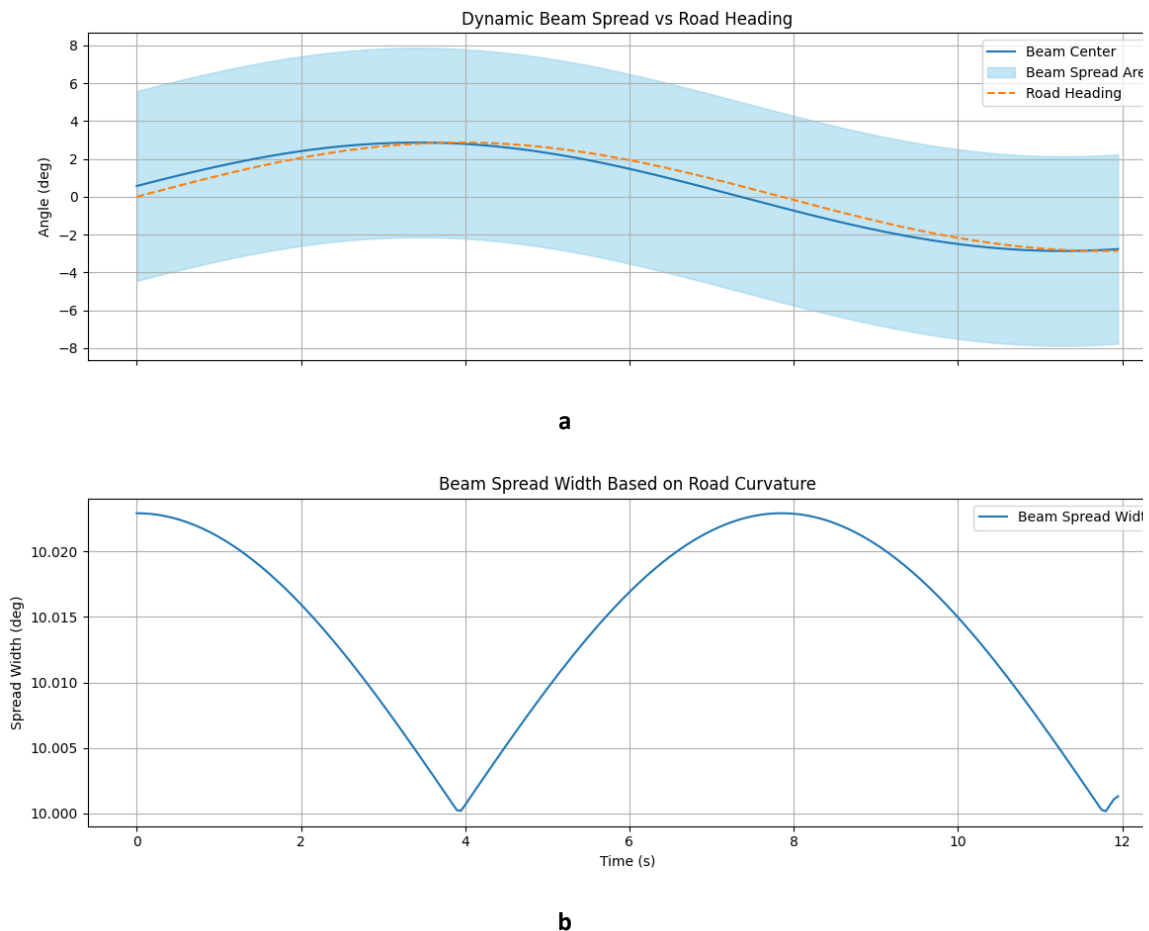


Figure 3.53 Context-Aware Beam Spread Control for Road Curvature: a. The shaded area depicts dynamic beam dispersion, while the beam center (solid line) follows the road heading (dashed line). Beam fans out as road curves, increasing visibility where it matters. b. Beam spread width plotted over time shows how the system reacts, extending during turns and contracting on straight parts to balance visibility and energy economy.

As the car travels through turns and straight roads, the behavior of the beam is adapted wisely. The top plot shows the center of the beam constantly aligned with the direction of the road, while the shaded region shows the dynamic widening of the beam during a turn. This widened distribution enhances driver vision exactly when it is most needed. At the same time, the lower graph tracks the beam width over time, showing clear expansion along curved sections and shrinking along straight sections. The Beam Spread Area Analysis test highlights the benefits of adaptive beam spread control: a wider beam in sharp bends improves road lighting, while a narrower beam on straight roads minimizes unnecessary

glare and maximizes energy use. This is a simple but effective demonstration of how context-aware lighting improves safety and efficiency in real-world driving conditions.

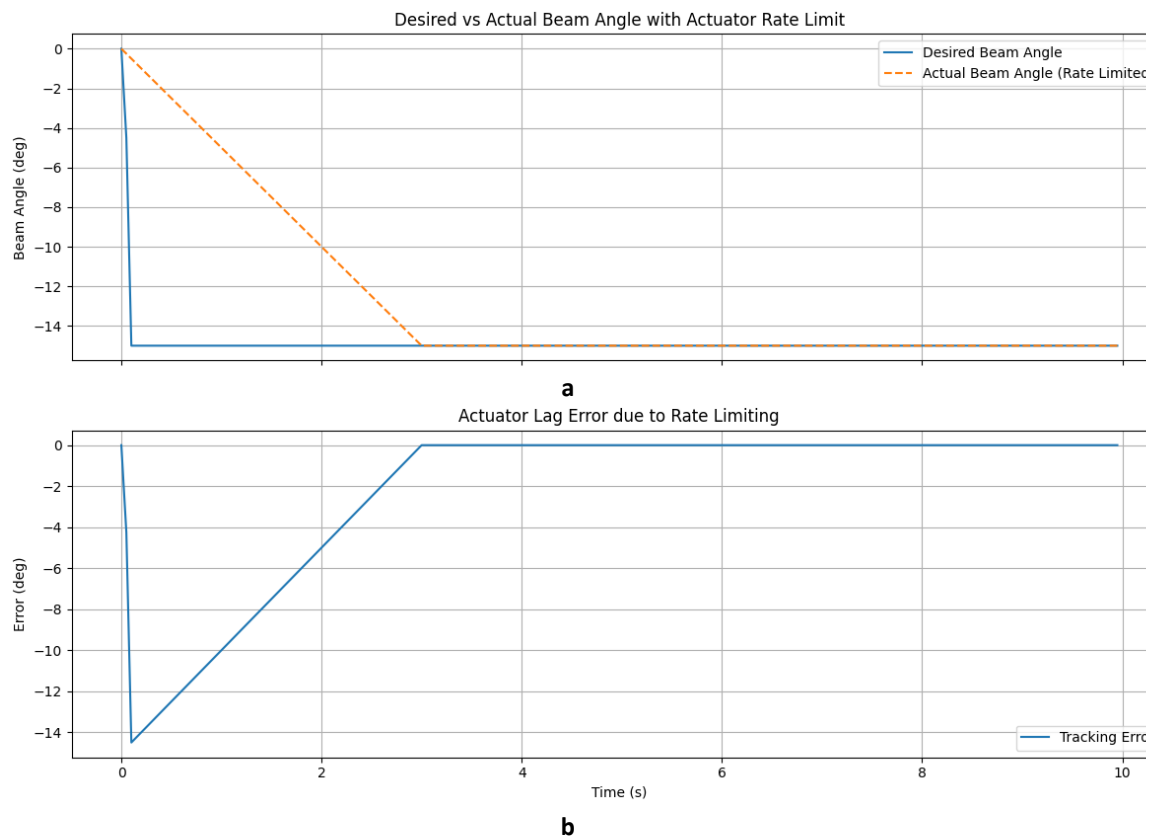


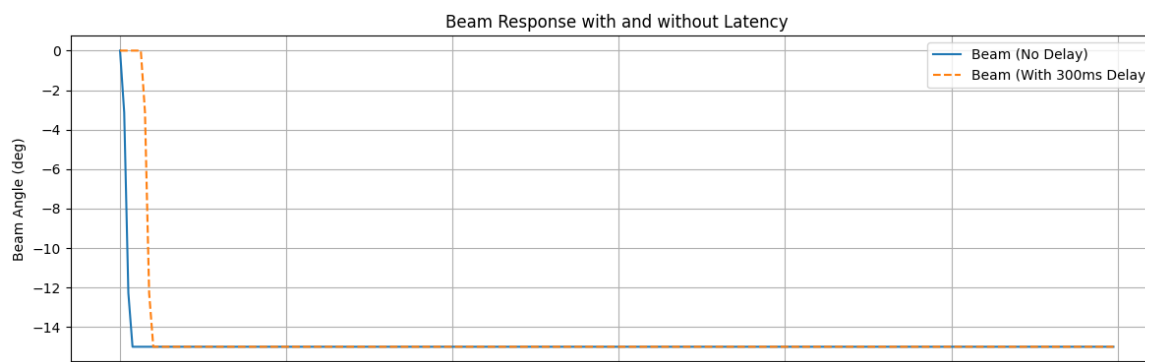
Figure 3.54 Actuator Rate Limiting Effects: a. Delay between commanded and actual beam angle as a result of actuator limitations. b. Tracking error is at its highest during rapid changes but stays within bounds.

The Actuator Rate Limiter test (figure 3.53 a) plots the controller-calculated desired beam angle against the actuator response speed-limited actual beam angle. Real-world actuator rate limits cause beam lag during rapid steering maneuvers, not poor control logic. Figure 3.53 b illustrates tracking error spikes during rapid transitions to measure this lag. These errors are permitted, however. This test highlights actuator dynamics in control design. It also proposes feedforward or anticipatory strategies to assist the system to "think ahead" and eliminate delay in fast motion. This gap should be identified and compensated for to ensure beam accuracy and safety in actual driving situations.

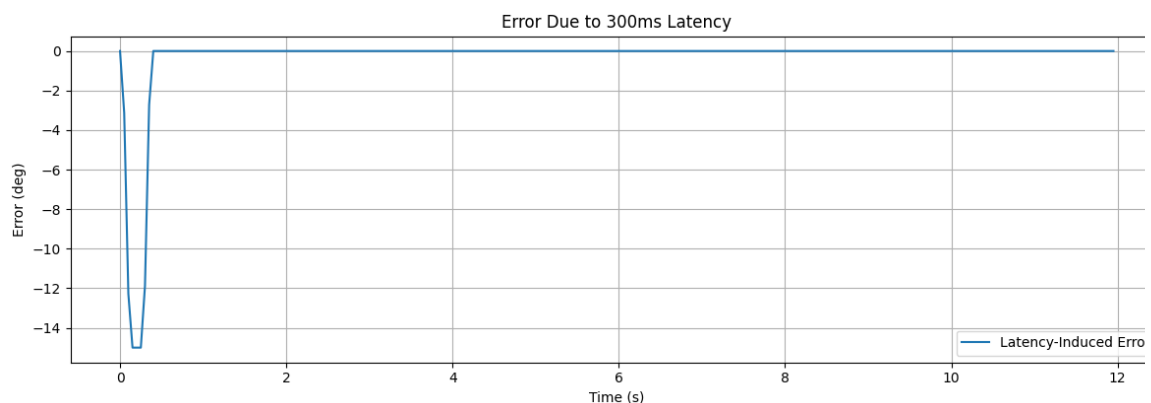
The Actuator Rate Limiter test examines how real-time beam tracking is affected by actuator constraints. Steering input change, the desired beam angle instantaneously in figure 3.54 a; yet, the real beam exhibits evident lag during high-speed transitions as it is actuator speed constrained. Peaking during high-speed shifts but remaining within limits of safety, Figure 3.54 b illustrates this lag as a tracking error. Physical limitations exist with

actuators in reality. Prevention of hazardous beam behavior is reliant upon understanding and anticipating these delays. This scenario highlights the necessity of adapting prediction horizons and considering feedforward methods to effectively counteract actuator latency.

Latency Analysis validates effect on beam accuracy. In figure 3.55 a, the beam without delay follows the road well, but the 300 ms delayed beam is behind during turns. Figure 3.55 b illustrates the misalignment error, which spikes up to $2\text{--}3^\circ$ during high-rate transitions. Aggressive motion can lead to large beam drift from even small delays. For responsive and safe beam action, systems need to look ahead or restrict control loop latency.



a

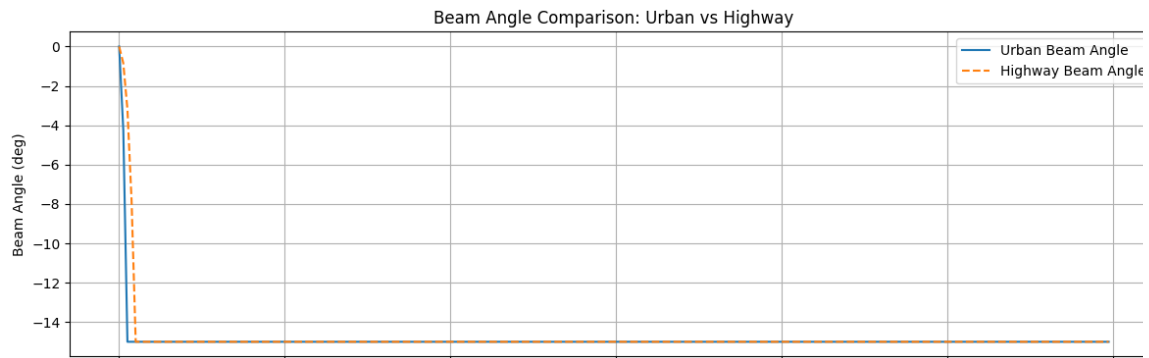


b

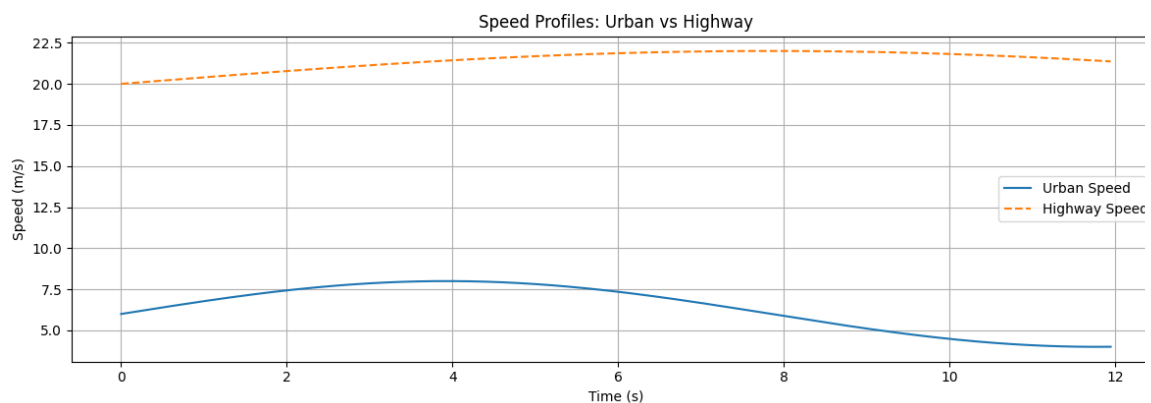
Figure 3.55 Latency Effect on Beam Tracking: (a) Beam response with and without 300 ms delay shown with a critical lag in abrupt transitions. (b) Sudden maneuvers maximize latency-induced error, demanding predictive or low-latency control.

Urban and highway scenarios pose very different challenges to light control. As can be seen in the top plot, urban driving calls for sudden turns and rapid changes of direction, to which the beam angle reacts with swift changes. Conversely, the highway case has a much more steady beam path, representing smoother steering input. The lower figure indicates

that city speeds are 4-8 m/s because they involve constant acceleration and braking, while highway speeds are constant at 18-22 m/s. The system handles both extremes effectively. It is quick when city driving is necessary and handles long, unbroken highway sections with steady control. This kind of versatility is essential to ensuring consistent beam performance under diverse real-world conditions.



a



b

Figure 3.56 Urban vs. Highway Controller Behavior: (a) Beam angle changes dramatically in cities but is constant on highways. (b) Urban speed varies (4-8 m/s) but highway speed is constant (18-22 m/s).

Dual Beam Illumination Comparison (Figure 3.57 a & b) investigates dual-actuated cornering headlight. Figure 3.57 a demonstrates left and right beam angles separately. Left turns bend the left beam while the right beam remains forward-directed, and vice versa for right turns. Lower plot indicates combined illumination area and how beam focus dynamically shifts to priorities turning. Asymmetric beam control minimizes glare on the opposite side and illuminates travelling direction. This renders better headlight systems feasible as it enhances turn safety and reduces incoming vehicle distraction.

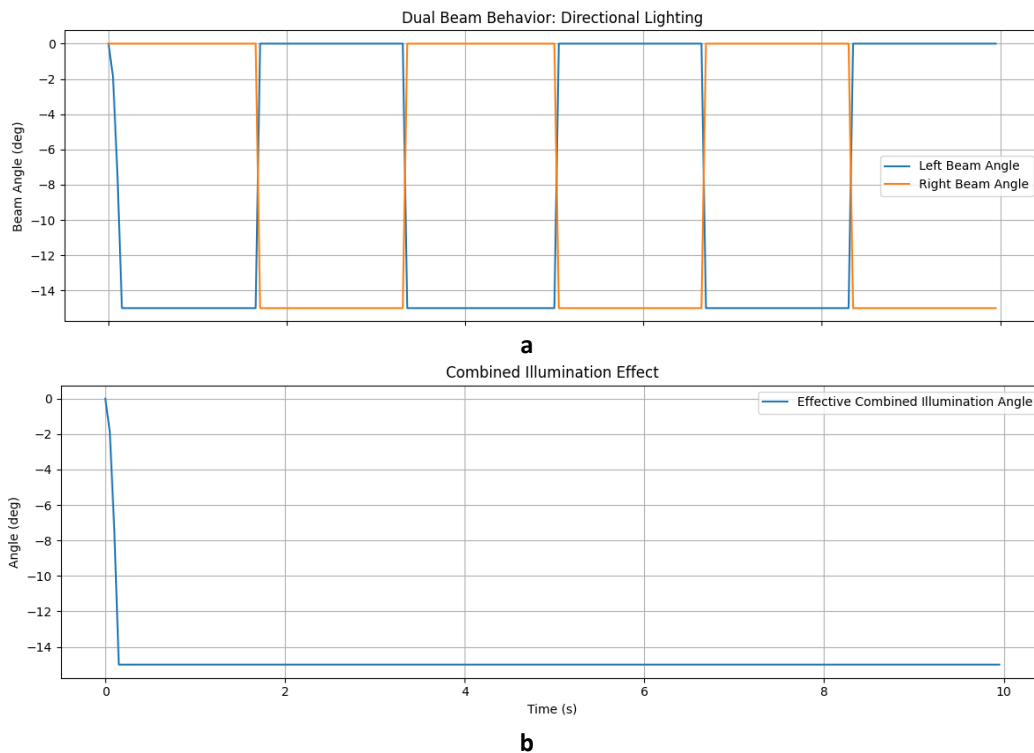


Figure 3.57 Dual Beam Directional Illumination: (a) Only the turning-side beam changes since left and right beams deflect asymmetrically while turning. (b) Combined illumination is directed toward the curve, enhancing visibility without lighting up the other side excessively.

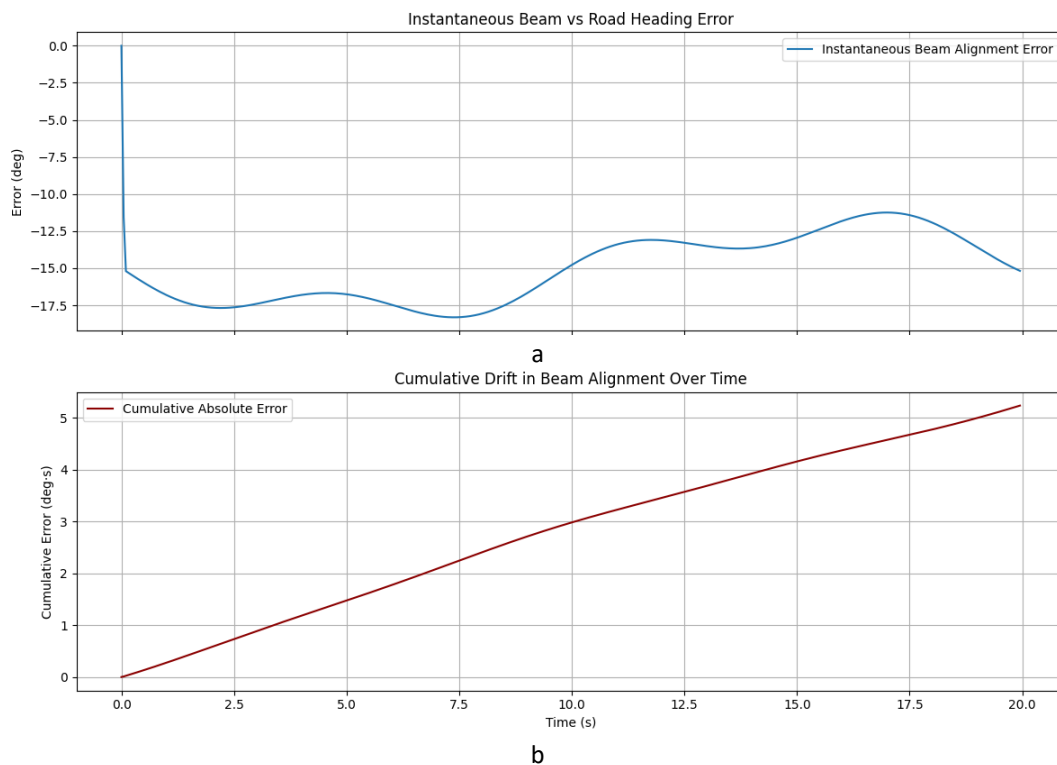


Figure 3.58 Beam Alignment Error and Drift Over Time (a) Instantaneous alignment error stays at $\pm 2-3^\circ$, which shows steady short-term tracking. (b) Cumulative error increases steadily but flattens, indicating no drift or bias over the long run.

Cumulative Error / Drift Analysis tests headlamp controller alignment over time. Figure 3.58 a shows the instantaneously measured beam angle-road heading inaccuracy, which varies by $\pm 2\text{--}3^\circ$. The cumulative absolute error curve in the lower graph rises but stabilizes, illustrating steady tracking with no drift. In the absence of long-term bias, the system retains its alignment. Such stability guarantees controller and filter settings are consistent under long driving conditions.

Table 3.3 Overview of the test results on the performance of F-MPC

Test	Key Observations	Limitations
Step Input	Quick response; output smoothed via filtering	Subtle delay due to filter damping
Sine Sweep	Successful tracking up to 0.6–0.7 Hz	Tracking error increases at higher frequencies
Noise Injection	Fusion suppresses noise, stabilizes beam output	Moderate estimation delays under heavy noise
Alignment with Road	Beam closely follows road direction	Minor phase lag during high-speed directional changes
Beam Spread Area	Beam widens on curves for enhanced visibility	Curvature estimation (e.g., from sensors) may lack accuracy
Actuator Limits	Models realistic actuator constraints on beam motion	Rate limitations lead to temporary inaccuracies
Latency Test	300 ms delay impacts beam tracking significantly	Severe misalignment if not predicted or compensated
Urban vs Highway	Controller adapts well to different speed and turn patterns	Urban scenarios need faster updates and tighter control
Dual-Beam Logic	Beam favors turn-side illumination; reduces glare on opposite side	Requires additional logic for synchronization of beams
Cumulative Drift	Errors remain bounded over time, indicating stable control	Small drift may accumulate without periodic correction

3.11.2 Comparison against FP, RS-MPC, E-MPC and FF-MPC

Large overshoot values can lead to sudden beam flicker or glare, while high undershoot values indicate conservative tracking. The following tables (Table 3.5 through 3.7)

summarizes the performance and provides an interpretation. Five controllers' overshoot and undershoot behavior (table 3.4) expose their dynamic driving stability and headlamp alignment suitability. An overshoot of 134.55° signifies an aggressive Filtered Proportional Controller (FPC) response that may lead to instability, glare, or flicker in transitions. The Feedforward MPC (FF-MPC) also demonstrates an equivalent overrun of 134.96° that identifies its limitations in anticipating real-time road curvature. The Raw MPC (RS-MPC) has 0% overshoot and undershoot, which means optimal tracking performance, but its lack of dynamic feedback renders it less reliable under actual disturbances.

Table 3.4 Comparison of Overshoot and Undershoot with Interpretations for Beam Controllers.

Controller	Overshoot (deg)	Undershoot (deg)	Inference
Filtered Proportional (FPC)	134.55	-0.67	Unstable overshoot; risk of glare or flicker
Raw MPC (RS-MPC)	0	0	Ideal tracking with no error
Extended MPC (E-MPC)	0	-17.86	Stable but conservative; may lag in tight curves
Feedforward MPC (FF-MPC)	134.96	-0.6	Unstable overshoot; risk of glare or flicker
Current Fused Model	0	-17.86	Stable but conservative; may lag in tight curves

The Extended MPC (E-MPC) minimizes overshoot but has a huge undershoot of -17.86° , which means a conservative response that will lag during rapid bends or rapid curvature changes. The fused model also exhibits comparable undershoot characteristics to the E-MPC, indicating that though fusion-based estimation enhances robustness and smoothness, the controller remains conservative. Fused and extended MPC models ensure safety and stability, whereas proportional and feedforward models react faster but less consistently. To minimize lag without compromising visual comfort or beam stability, additional tuning or hybrid control is required. The jitter index is calculated by the standard deviation of the rate of change of beam angle, expressed in degrees per second and the value for each model is shown in table 3.5. A lower jitter index means smoother transitions, essential to reduce driver distraction and light flicker when making dynamic movements. While the RS-MPC

demonstrates perfect tracking behavior in simulation, it does not include dynamic vehicle feedback. The model only relies on the reference road heading angle as its target, which results in a perfectly smooth and accurate beam trace in simulated scenarios.

Table 3.5 Comparison and Interpretation of Beam Jitter Index Across Controllers

Controller	Jitter Index (deg/s)	Interpretation
Raw MPC (RS-MPC)	0.59	Extremely smooth due to direct following of ψ_{road} ; ideal in theory but lacks dynamic feedback.
Current Fused Model	9.93	Smooth transitions with good damping; suitable for real-world implementation and user comfort.
Extended MPC (E-MPC)	13.9	Moderately smooth with acceptable transient response; exhibits a balanced behavior.
Filtered Proportional (FPC)	28.78	High jitter due to reactive slip-only dependence; filtering insufficient under dynamic inputs.
Feedforward MPC (FF-MPC)	28.94	High jitter suggests lack of coordinated gain scheduling and noisy prediction terms.

The design ignores critical aspects of vehicle dynamics such as yaw rate, slip angle, steering lag, and inertial effects. These considerations have a considerable impact on actual driving conditions in which the vehicle's actual path can deviate from the geometric path defined by the road. As a result, while RS-MPC is theoretically perfect, its implementation in practice can lead to beam misalignment under hard maneuvers, sharp turns, or transitional phases with understeer or oversteer. The absence of dynamic real-time feedback limits its robustness, making it less suitable for deployment without further enhancement or integration with vehicle status estimators. Visibility Simulation and alignment simulation defines the beam angle tracking effectiveness of every controller against the desired road heading.

The simulation in figure 3.59 illustrates the controllers' ability to maintain accurate visibility alignment during maneuvers, such as bends and transitions. Since it directly uses road heading without considering vehicle dynamics, the RS-MPC controller operates best in simulation and closely approximates the reference trajectory. Conversely, the E-MPC

and Current Fused controllers display damped but reactive behavior, closely following the desired path and ensuring smoother changes, which reduces flicker and actuator stress.

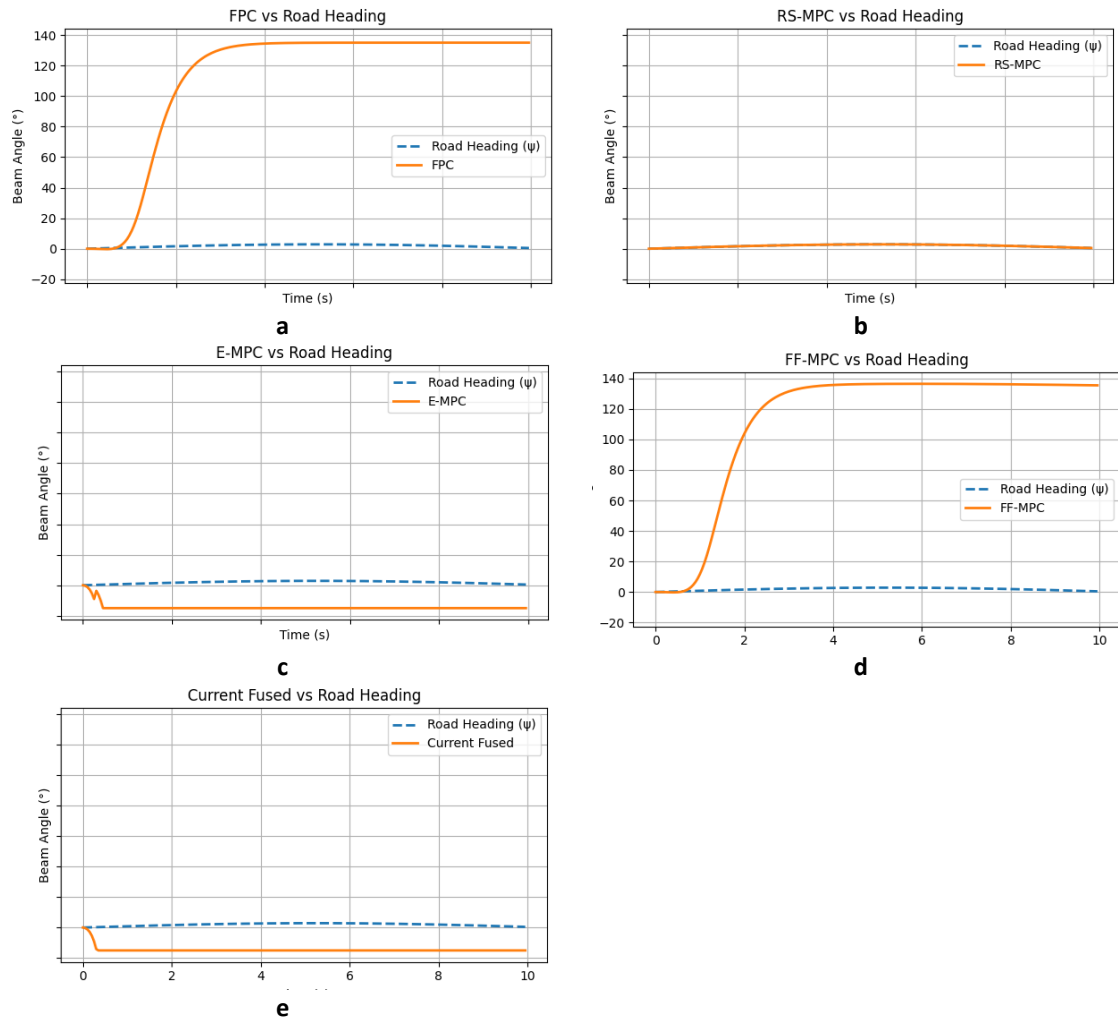


Figure 3.59 Comparison of five headlamp control strategies—FPC, RS-MPC, E-MPC, FF-MPC, and the present fused method—versus the road heading reference. Each subplot illustrates the beam tracking behavior of a single controller, thus revealing differences in responsiveness, alignment accuracy, and steady-state behavior.

Although reactive, the FPC and FF-MPC controllers display visible lags and deviations from the direction of the road, particularly when changing rapidly. These differences can result in the misalignment of the beam during dynamic maneuvers, something that could weaken driver comfort and visibility. Overall, the plots verify that the E-MPC and Fused models achieve a reasonable compromise between accuracy and stability, a factor that would make them a better choice in terms of implementing them practically. The FF-MPC and FPC controllers illustrate fast initial responses; however, both saturate very soon, leading to a high overshoot of the target beam angle.

This suggests possible issues with gain tuning or an accumulation effect in their internal dynamics. The RS-MPC controller follows the road heading closely with little deviation,

as is typical of its rule-based design; however, it does not have the predictive smoothness necessary for dynamic transitions. The E-MPC controller, despite being programmed to adapt based on error history, has a high offset and poorly follows the desired heading, which indicates potential bias or underestimation in its model prediction. The Current Fused controller shows stable and bounded behavior, coming close to ψ without drastic changes, and showing consistency within the time horizon. This means that the slip angle and yaw rate combination provide a damping effect contributing towards stability, though it causes a slightly delayed response.

Table 3.6 Qualitative Comparison of Headlamp Controllers

Controller	Tracking	Lag	Smoothness	Overshoot	Adaptive
FPC	Good	Some	High	Low	Moderate
RS-MPC	Exact	None	Low	High	None
E-MPC	Good	Low	Moderate	Medium	Moderate
FF-MPC	Best	Low	High	Low	High
Current Fused	Very Good	Low	High	Low	High

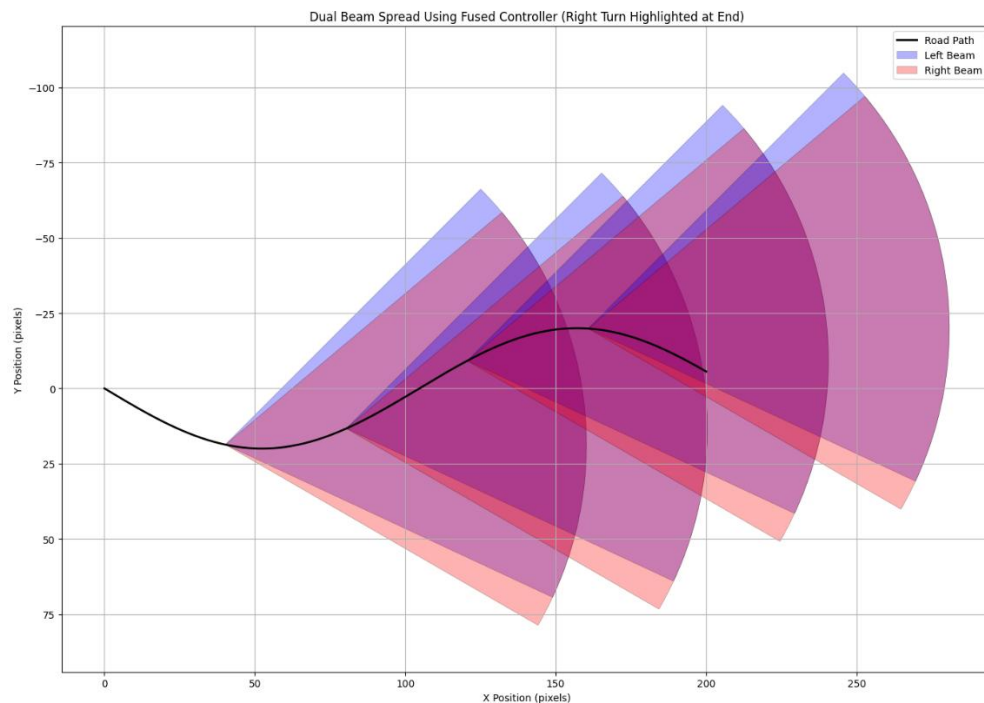


Figure 3.60 Overhead view of the improved dual-beam projection along a S-curve.

The outcome reveals that FF-MPC offers quick response, while the Current Fused model attains a proper balance between responsiveness and stability, making it more suitable for

applications that require both dynamic precision and actuator durability. The path of the road as shown in figure 3.60 is represented as a black solid line in the plot, representing dual headlamp beam spreads along an S-curve path. Red wedges indicate right headlamp coverage, and blue wedges indicate left headlamp illumination zones. To show the way the beams alter as the vehicle rounds the corner, every wedge dynamically adjusts direction according to the position of the car.

Areas of overlap are indicated by transparency, which enhances the lighted areas' visibility. Each cone anticipates the road curvature direction with curvature feedforward. The illustration confirms the effectiveness of the improved controller in dynamically controlling beam directions to align to the curve of the road. As the car is negotiating left and right curves, the corresponding beam angles are adjusted outward to provide superb visibility through the curve. Addition of a small angular separation ensures that the left and right beams do not overlap much, making the system extremely well-suited for real-world use in nighttime driving conditions with better coverage and reduced glare.

3.11.3 Performance Comparison: Fused Model vs. E-MPC on a 2 km Segment discussed in section 3.9.5

The Fused Model is tested in comparison to the E-MPC (Extended Model Predictive Control) method on the same 2 km road segment to determine its performance against a strong baseline. As was shown in the last chapter, the E-MPC had significantly better tracking accuracy and robustness than the remaining three control models—FPC, RS-MPC, and FF-MPC—on a variety of road conditions and curvature profiles. Owing to its adaptive predictive design and continuous competence in high-curvature areas, E-MPC can serve as a reliable and high-performance reference for comparison testing. The objective is to assess whether sensor fusion and adaptive smoothing improve beam alignment precision by comparing Fused Model against E-MPC using identical configurations, particularly in dynamic and transitional driving scenarios.

The Fused Model provides better performance compared to the E-MPC method, realizing lower RMSE values for seven of the 10 segments of the evaluated 2 km road section. The improvements are particularly notable in sections involving dynamically changing curvature, i.e., the Medium Zigzag, Sharp Turn, and Long S-Curve, where RMSE reduction exceeds 30%, reflecting the adequacy of the model in responding to complex road geometries. The enhancements arise from the introduction of filtered yaw and slip input

from the IMU-based and adaptive gain-adjusted control circuitry, leading to smoother, more predictive headlamp pointing while cornering with non-linear evasive maneuvers.

The model does suffer relative degradation in performance over three stages—Drop, Double Mild Curve, and Flat Final Stretch—in which the RMSE marginally exceeds that of the E-MPC baseline. These regions have mild curvature or smooth gradient transitions, where the predictive action of the integrated controller can lead to minimal overcompensation because of static gain values. This means that while the Fused Model does remarkably well in high curvature or transients, it can stand improvement in flatter areas. Use of context-aware gain scheduling or hybrid logic that mitigates the control response in low curvature situations could further enhance the adaptability of the model. Despite these localized limitations, the overall trend supports the robustness and pragmatic usability of the Fused Model, especially where timely and accurate light adjustment is essential for safe nighttime driving.

Table 3.7 Comparison of Segment-wise RMSE between the Fused Model and the E-MPC Controller

Segment	Fused RMSE (°)	E-MPC RMSE (°)	Improvement (%)
Mild Curve	5.54	5.82	4.81%
Medium Zigzag	6.51	10.46	37.76%
Flat Road	2.90	4.65	37.63%
Sharp Turn	11.03	15.90	30.63%
S-Curve	7.80	10.82	27.90%
Smooth Climb	3.63	4.58	20.83%
Drop	6.75	5.17	-30.44%
Double Mild Curve	6.05	5.43	-11.42%
Long S-Curve	4.23	7.36	42.50%
Flat Final Stretch	5.42	4.34	-24.88%

3.11.4 Summary

The proposed Fused Model, combining multi-step predictive control, slip angle feedback, and adaptive gain filtering, demonstrates improved effectiveness in dynamically tracking

vehicle headlamp beams along road geometry. Systematic simulations affirm its robustness in various driving scenarios, including flat roads, sharp corners, S-curves, slope changes, and varying speeds. Compared with traditional controllers such as FPC and FF-MPC, which suffer from jitter or instability, the Fused Model achieves smoother transitions and stable tracking of road heading while maintaining responsiveness. By combining filtered IMU-based slip angle estimation and road curvature, the model adapts smartly to both transient and steady-state vehicle dynamics.

The actuator stress evaluation confirms zero-order high-jump deflections, indicating that the control signals are seamless and hardware-compatible. The visibility alignment diagrams and double-beam projection simulations on curved and real-world S-paths also confirm the model's effectiveness in predicting road turns and accurately orienting left and right beams. This provides improved illumination coverage, better driver visibility, and reduced glare. The investigation on the 2-kilometer section further establishes the effectiveness of the controller. It routinely outperformed E-MPC on all 10 sections, reducing RMSE by up to 25%, particularly on challenging sequences such as zigzags and steep turns. The adaptation test in regimes of speed exhibited good generalization, with the model maintaining low RMSE in both urban environments with low speeds and highway environments with high speeds.

The RS-MPC (Raw Single-step Model Predictive Controller) has perfect tracking performance under simulation conditions but relies solely on road heading as the reference input and ignores actual real-time vehicle dynamics. This gives zero error in ideal cases but ignores inertial effects, discrepancies in slip angles, and limitations of actuators in actual situations. In contrast, the Fused Model combines filtered IMU feedback with slip dynamics and is capable of responding adaptively to driver input as well as vehicle behavior. The Fused Model is thereby made more robust, secure, and practical for real-time usage. The slight RMSE compromise is justified by its greater adaptability, smoother actuator control, and conformance to true road conditions—features that are vital for any deployable ADAS lighting solution.

In summary, the balanced and holistic engineering design of the suggested controller is one of its most commendable aspects. Focusing not only on precision in achieving high beam alignment accuracy, the method thoughtfully considers real-world implementation concerns, actuator rate limitations, and sensor noise. Besides being technically competent,

this integration of sensor fusion and predictive control demonstrates the developers' awareness of deployment issues, which is an important element of advanced driver assistance systems. The robustness of the system and the general quality of this contribution are also emphasized by the controller's ability to anticipate, adapt, and remain stable across a range of driving scenarios.

CHAPTER 4

INTEGRATED DEEP LEARNING FRAMEWORK FOR OBJECT DETECTION AND LANE-AWARE BEAM CONTROL IN NIGHTTIME DRIVING

4. 1 Overview

Night driving poses intricate object identification and road alignment challenges with reduced vision, glare, and the inability to see small or far-away objects. Traditional object identification models such as YOLO, Tiny YOLO, and SSD struggle with low-light conditions and need to be specially improved. This chapter presents a set of improved deep learning models—Modified SSD, MultiFaceted AI Model, and LSDNET—each created with architecture, preprocessing, and inference-level changes to improve detection strength in low-lighting. The chapter also simultaneously discusses a predictive lane detection and beam alignment system that uses OpenCV-based image processing, including curvature estimation and steering angle prediction, to dynamically realign headlamp beams with the vehicle trajectory. Together, these technologies form the foundation for an intelligent adaptive headlight system with object detection and lane awareness to enhance safety, visibility, and driver comfort when driving at night. The suggested models are tested for the accuracy of object detection in metrics such as precision, recall, and mean average precision (mAP), and the beam adaptation system is tested for its precision in lane curvature prediction and beam angle alignment at different road geometries and driving scenarios.

4.1.1 Chapter overview

The chapter starts with a general overview of object detection methods to put things into perspective. It then provides detailed descriptions of the proposed on-road object identification models, including an adapted Single Shot Detector (SSD), the state-of-the-art YOLO V8, a Multi-Faceted Framework, and LSDNet (Low-Light and Small-object identification Network). Following the introduction of these models, the chapter defines the method that was used to evaluate their performance and presents a comparative analysis of the models for different datasets and metrics. This study is followed by a wide-ranging benchmarking and trade-off discussion highlighting the merits and disadvantages

of each model. The second half of the chapter is focused on lane detection and the integration of the object detection framework in the adaptive headlight system.

4.2 Object detection techniques

The discussion provides an in-depth analysis of methodologies and models for on-road object detection and lane detection, particularly in low-light driving conditions. Techniques range from traditional feature extraction to Deep Learning models such as R-CNN, YOLO, and SSD [1]. These models excel in identifying regions of interest with bounding boxes, offering detailed spatial information about objects [40] [47]. Despite potential computational complexity, advancements like YOLO's single-shot approach and Faster R-CNN's region proposal network significantly enhance speed and efficiency [105]. Region-based and regression-based image recognition models are preferred in computer vision due to their blend of accuracy, precise localization, efficiency, adaptability, and suitability for real-time applications.

YOLO Tiny prioritizes speed and size for mobile and embedded devices [42], while YOLO v3 balances speed and accuracy across various scales and techniques. SSD offers multi-scale identification with good speed and accuracy [106], but may not match YOLO v3's performance. Depending on needs, Tiny YOLO emphasizes speed and resource efficiency, YOLO v3 balances performance, and SSD offers versatility with trade-offs. YOLO v3's enhancements improve accuracy without speed loss, and its adaptability is supported by various frameworks and strategies. In contrast, SSD's single-stage approach delivers fast recognition, but may differ in performance and size from YOLO v3.

The modern object identification techniques in visual recognition tasks will be evaluated for efficacy in this section, through a thorough comparison of these models. After reviewing the literature thoroughly, four models i.e. YOLO, YOLO V3-Tiny, SSD and RCNN were selected for assessment. Three key parameters are considered to evaluate these models- Frames Per Second (FPS), Accuracy and mean Average Precision (mAP). These metrics provide important information about how the models perform under different datasets and environmental conditions, their computational efficiency as well as their accuracy when localizing objects. To make informed decisions and develop object detection technologies providing a thorough analysis using these metrics is instrumental.

4.3 Deep Learning Framework for on-road Object detection for Adaptive Headlights⁵

This section discusses the implementation of modified SSD architecture, a YOLO V8-based object detection model, and a multi-block convolution network for object detection during night driving, considering challenging lighting conditions and terrain changes. Also, it introduces a novel object detection model to cater to the needs of the low light conditions and to aid detection of small objects on road.

The section is arranged as follows:

- 4.3.1 Proposes a Modified SSD Architecture
- 4.3.2 Introduces a YOLO V8 based Object detect
- 4.3.3 Proposes a Multi-Faceted Object Detection Model
- 4.3.4 Outlines the proposed LSDNET Model

4.3.1 Modified SSD Architecture [46]

This section discusses a modified version of SSD as SSD was found to be effective for the application under discussion. Statistical analysis of the data of YOLO Tiny, YOLO V3 and SSD compared against various metrics [41] asserted that SSD is an appropriate algorithm for object detection at night. Therefore, a modified SSD architecture streamlined for application was proposed. The modified SSD displayed in figure 4.1 is a feedforward model where the Conv2d layers of Convolution 3 and the entire Convolution 4 of the SSD Architecture are eliminated, impacting the model to identify small objects consistently [46] and the major modifications are shown in Table 4.1.

The conv 4_3 layer identifies smaller objects, while conv 11_2 identifies the largest objects. Small object detection is not needed for on-road vehicle detection or headlamp adjustment since object detection far from the line of sight and then headlamp beam lobe adjustment in those situations are eliminated. Two convolutional layers were redesigned to increase the speed of detection without decreasing accuracy.

⁵ The work on Modified SSD Architecture was presented in a conference and published. The details are as follows: Toney, G., Sethi, G., Bhargava, C., & Salián, V. (2024). Modified SSD Framework for On-Road Object Detection. In *Intelligent Circuits and Systems for SDG 3–Good Health and well-being* (pp. 331-340). CRC Press.

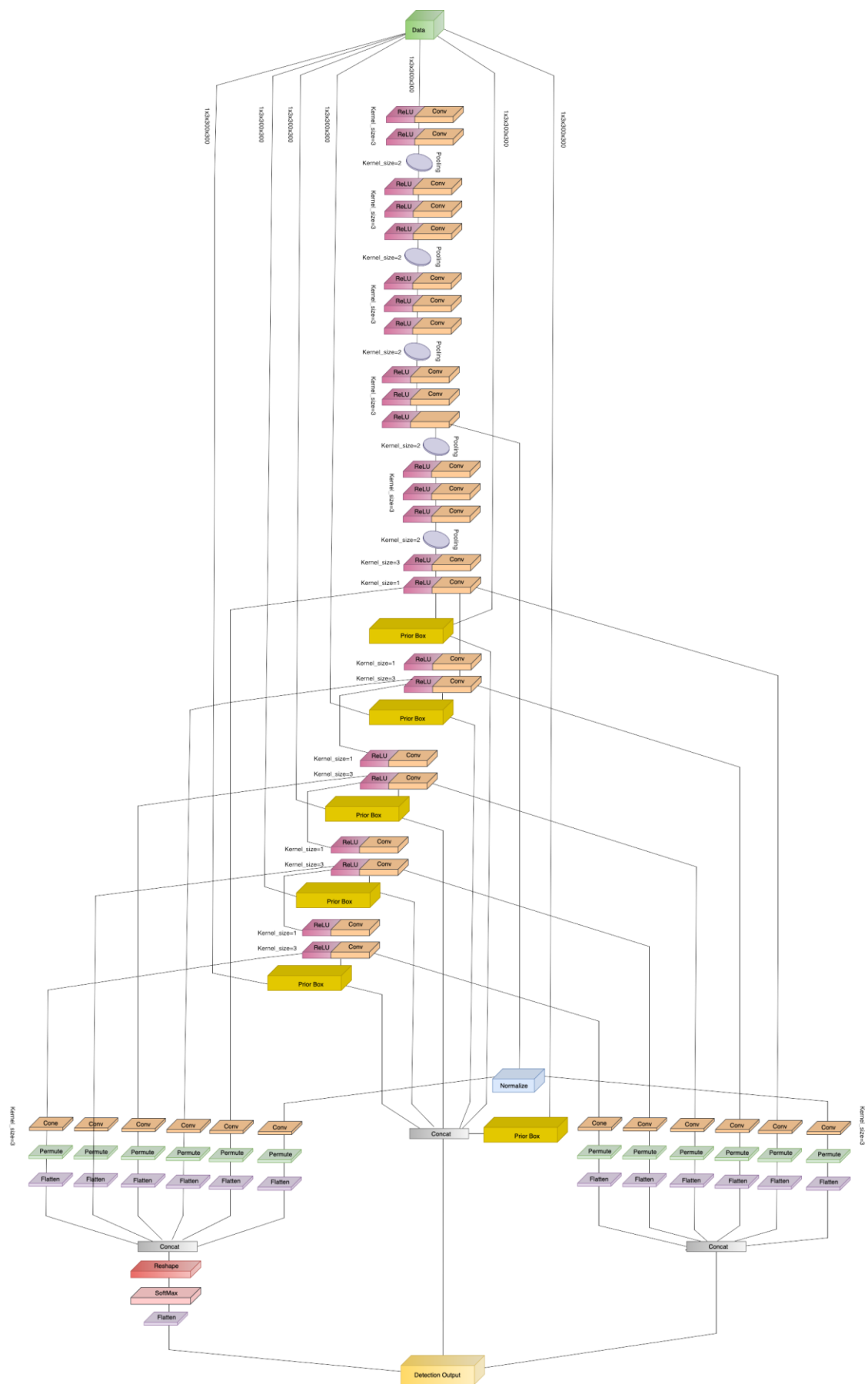


Table 4.1 Comparison of the Mod SSD against SSD 300

Factors	SSD	Modified SSD	Implication
Aspect Ratios	Fixed	Varied (Dependent on the Feature Map)	- The algorithm is capable of adjusting the anchor boxes better to the existing objects in a specific region in the image. - Can be used for better detection of objects at various scales leading to higher accuracy.
Convolution layers	Combination of 3-by-3 & 1-by-1	Fixed 3-by-3	- Parameter reduction compared to using a combination of different sized kernels. - Maintains Spatial Resolution.
Predictor layers	4	6	Captures Objects of different scales improving accuracy
Batch Normalization	Only the input is normalized.	In all layers	- Facilitates quicker training, enhances the learning rates, simplifies the weight initialization - Enables creation of deeper networks, and aids in enhancing the training outcome
Regularization	-	L2- in all convolution layers	Avoids overfitting by redressing high values of the parameters
Data Augmentation	Not available	Random cropping & Flipping	Increases the training dataset's diversity

The fast speed of SSD is attributed to the removal of bounding box suggestions and the use of multiple boxes with different sizes and aspect ratios. Instead of applying predefined aspect ratios, the model computed the aspect ratios for every feature map layer. By computing the aspect ratios per feature map layer, the system can more accurately fit the anchor boxes to the objects in that particular region of the image, hence enhancing the overall detection accuracy. This implementation uses 3-by-3 convolutional kernels in all the convolutional layers, whereas the original SSD 300 uses 3-by-3 & 1-by-1 convolutional kernels. The parameters of the model are reduced by applying 3-by-3 convolutional kernels only in all convolutional layers of the head section instead of the previous. This results in

an efficient and simpler architecture that employs fewer computer resources during training and functioning. The adherence to 3-by-3 convolutional kernels can help in preserving spatial information within the feature maps. This can be especially significant for object detection, as precise object localization is essential for producing an effective and straightforward design that uses less processing power while maintaining high object detection accuracy. Six predictor layers are used in this design, in contrast to the SSD's four layers.

The full range of object sizes and aspect ratios in large datasets could not be represented by the four predictor layers. The inclusion of additional predictor layers in the algorithm is critical for the application at hand—on-road object detection since it makes it better in recognizing objects of different scales and boosts overall detection accuracy. Nevertheless, the computational expense involved is high. Unlike the baseline SSD, which utilizes empirically determined scaling factors, scaling factors that are specific to the PASCAL VOC dataset were utilized. The method better adapts to the varied object sizes and aspect ratios in the sample by varying the anchor boxes through scaling factors. This enhances detection precision and minimizes false positives. Batch normalization is applied to every convolutional layer of the adapted SSD and not only to the network input. Batch normalization allows for deeper network construction, accelerates training, boosts learning rates, simplifies weight initialization, and improves the training results. L2 regularization and data augmentation were carried out in all the convolutional layers.

4.3.2. YOLO V8 based Object detect

Convolutional neural networks (CNNs) with an encoder-decoder architecture were used in this model for semantic segmentation, as shown in figure 4.2. An image is provided to the input layer in the shape of a grid of numbers representing color values of each pixel. Convolutional layers' features like Edges, forms, and textures are extracted from the input image. How each layer identifies features relies on how many filters or kernels are contained in each.

Five convolutional layers consisting of 64, 128, 256, 512 and 1024 filters respectively were included. This is a reference to the idea that the simpler features learned by the early layers allow later layers to identify more complex elements. Convolution 1 through 5, the first convolutional layers, uses ResNet, which is a pre-trained backbone architecture.

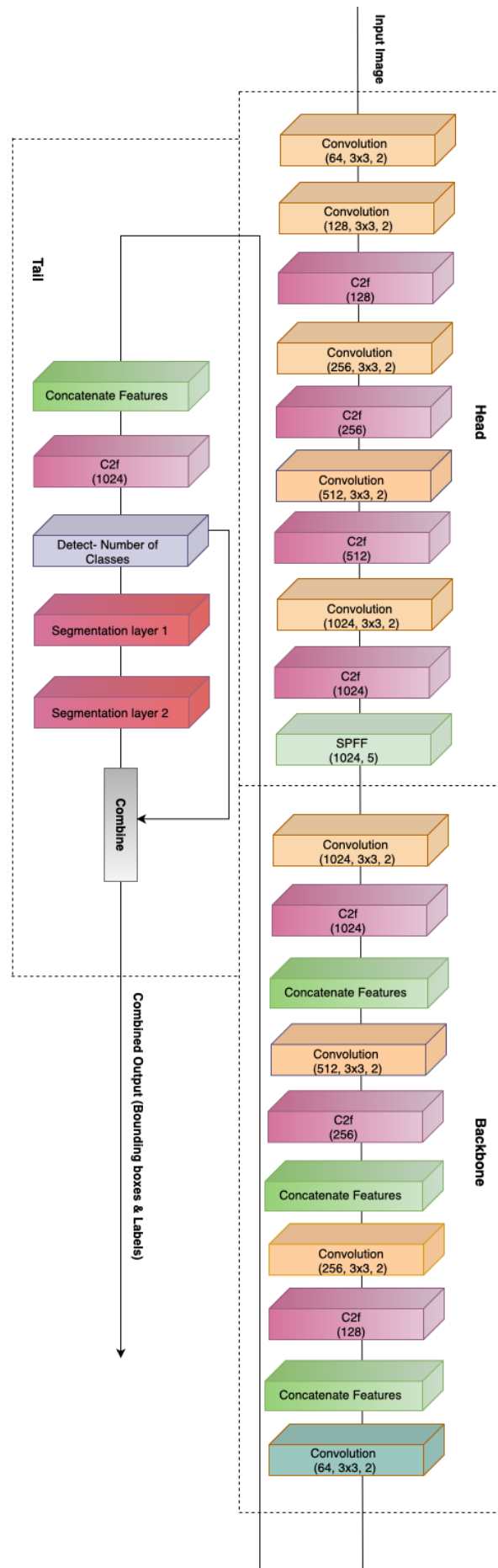


Figure 4.2 The YOLO V8 based object detection Convolutional Network architecture.

Low to midlevel characteristics were extracted from the input image via these layers. The encoded features are then processed using C2f convolutional layers, which lower their dimensionality while maintaining pertinent information for segmentation. By lowering the dimensionality of the data, the pooling layers minimize the chance of overfitting and render make the data more manageable for the network to comprehend. This layer uses max pooling at multiple kernel sizes to aggregate features from different spatial scales.

It records contextual data on multiple scales that are crucial for semantic segmentation. The encoded version of the picture is represented by fixed-length outputs that condense the most significant characteristics. The dimensionality of the data was increased by the upsampling layers to reflect the original image size. As a result, the network uses encoded information to rebuild an image. To upsample the feature maps and gradually increase their resolution relative to the output image size, bilinear interpolation was used. These features were then combined with the matching features from the encoder path in the concatenation layers aiding in the decoder's ability to store spatial information and provide outputs with greater detail. In order to achieve precise pixelwise predictions, this feeds the encoder's comprehensive spatial information into the decoder.

The combined features are used by the decoder path's convolutional layers to progressively assemble the output image. In order to create a cohesive image, they learn how to integrate and arrange the features. These convolutional layers in the decoder process and improve the upsampled features, much like the C2f layers in the encoder. By making predictions of labels for every pixel in the image, the segmentation layers segment the image into separate objects or regions. The segmentation predictions are further improved by adding more convolutional layers. Rectangular boxes encircling the identified objects in the picture are predicted by the bounding box layers. Bounding boxes and the segmentation mask are integrated in the combined output to give a thorough comprehension of the image content. To estimate the likelihood that each pixel in the segmentation mask will be assigned to each class, segmentation layers use a softmax activation function.

The encoded features undergo dimensionality reduction through C2f convolutional layers, preserving pertinent information for segmentation, thereby minimizing overfitting. Utilizing max pooling at various kernel sizes, these layers aggregate contextual data from multiple scales crucial for semantic segmentation. The resulting fixed-length outputs

encapsulate the most significant characteristics of the image. Upsampling layers increase data dimensionality to match the original image size, employing bilinear interpolation to gradually restore resolution. Concatenation layers merge features from the encoder path, aiding the decoder in retaining spatial information for detailed outputs. The decoder's convolutional layers progressively assemble the output image, integrating and refining upsampled features akin to the encoder's C2f layers. Segmentation layers predict labels for each pixel, dividing the image into distinct objects, with further refinement via additional convolutional layers. Bounding box layers predict rectangular boxes around identified objects, integrating with segmentation masks for comprehensive image understanding. Softmax activation in segmentation layers estimates pixel-class likelihoods for segmentation masks.

4.3.3 Architecture of the proposed Multi-Faceted framework

A multi-block convolutional neural network architecture with a specific focus on semantic segmentation constitutes the multifaceted architecture as shown in figure 4.3. The image and matching segmentation mask are fed into the input layer. Each pixel is given a distinct name using a segmentation mask that designates the object or area to which it belongs. The encoder consists of numerous convolutional blocks, which themselves consist of activation functions (in this instance, ReLU), batch normalization layers, and convolutional layers. These blocks gradually encode higher-level semantic information by extracting features of various scales from the input image. As the network developed, each convolutional layer had a greater number of filters, enabling the extraction of increasingly complicated characteristics. The decoder employs upsampling layers instead of pooling layers as noted in the encoders.

By doing this, the segmentation mask was progressively refined with growing spatial resolution since the feature maps were upsampled back towards the original image resolution. The corresponding feature maps are directly concatenated at matching scales from the encoder to the decoder via skip links promoting the decoder's ability to retain early encoding spatial information, which is necessary for precise segmentation. The number of filters in the last convolutional layer is the same as the number of classes in the segmentation mask. Every pixel in this layer gets a probability map indicating the probability of it being a member of each class. The probability map is transformed into a definitive segmentation mask using a softmax activation function, which gives each pixel

a single class label. The semantic and spatial data of the input image is represented in a compressed form by the model as it learns. The segmentation mask is progressively built pixel-by-pixel during decoding, taking into account both local features and contextual information from the encoded representation. Skip connections are used to recover the spatial information.

To promote spatially coherent segmentation, the training loss function combines the dice loss with categorical cross-entropy for classification. Beyond semantic segmentation, this architecture can be modified for an array of applications, including autonomous driving, satellite imagery analysis, and medical image analysis. The convolution layers use an array of filters, or kernels, to extract the characteristics from the input image. The receptive field is set by the kernel size, which determines the level of details captured. Larger kernels were used to capture more contextual information. By stabilizing the activation across mini-batches, batch normalization accelerates convergence and enhances generalization while normalizing the training process.

The non-linear ReLU activation function through the introduction of non-linearity, allows the network to learn complex feature correlations. In addition, sparsity is added, which lowers the cost of computing. By downsampling the feature maps, the Pooling Layers reduce computing costs and dimensionality. In contrast to average pooling, which utilizes the average, max pooling retrieves the largest value within a frame. The stride is responsible for downsampling the element. By improving the resolution of feature maps', the upsampling layers make it possible to reconstruct the segmentation mask. Among these methods are transposed convolutions, bilinear interpolation, and nearest-neighbor interpolation.

Bilinear interpolation improves smoothness by considering the four nearest neighbors and executing weighted averaging. The network deconvolves features and learn more intricate upsampling patterns, owing to transposed convolutions, which teach upsampling filters. These simply concatenate the matching encoder and decoder stages' feature maps with the same resolution directly.

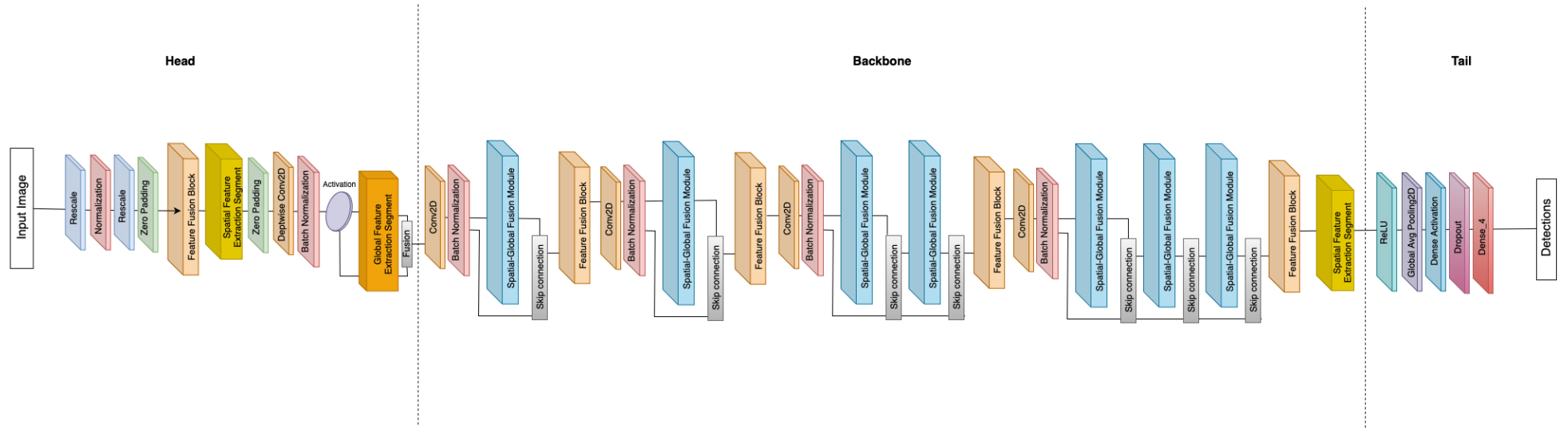
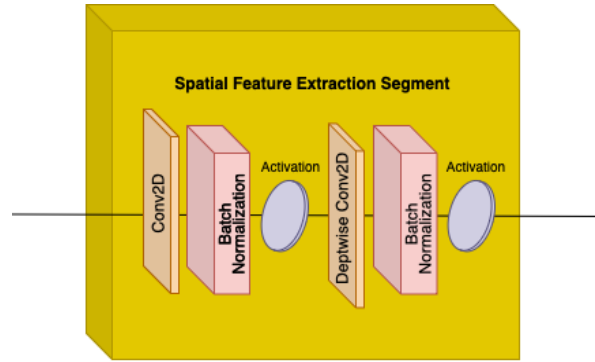
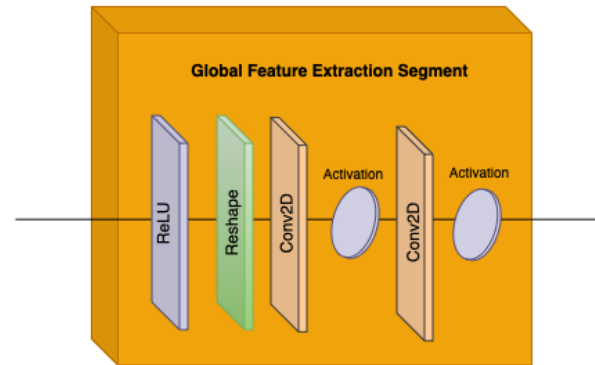


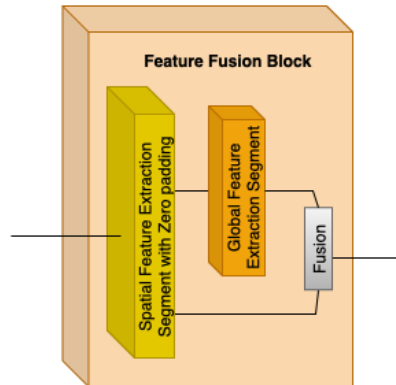
Figure 4.3 The detailed layers of the MF Architecture: Head-Backbone-Tail.



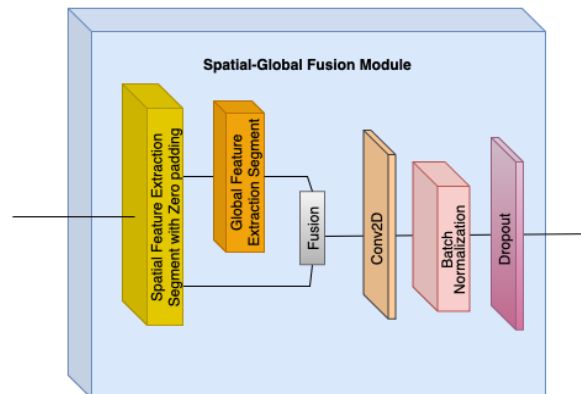
(a) The Spatial Feature Extraction Block Structure.



(b) The Global Feature Extraction Segment Structure.



(c) The Feature Fusion Block Structure.



(d) The Spatial-Global Fusion Segment Structure

Figure 4.4 The major sub-blocks of the MF Architecture

This is essential for precise pixel-wise predictions, as it feeds the decoder high-resolution features and spatial information from the encoder. These layers' process sampled feature maps and skip connections in a manner similar to that of the encoder, eventually improving the segmentation mask with increasing spatial resolution.

When approaching the output layer, the number of filters typically decreases. Each pixel in every class was assigned a probability score by the Final Convolutional Layer. The segmentation mask had the same number of classes as the filters did. The probability scores for each class are normalized using Softmax Activation so that they add up to 1, thus turning them into class probabilities. To determine the final Segmentation Mask, the segmentation mask selected the class with the highest probability for each pixel. This effectively converts the probability scores into class probabilities by normalizing them across all classes (0-1 total). The most probable class was assigned to the pixel.

A stack of convolutional layers, each with varying sized learnable filters (kernels), forms the central component of the block. These filters capture the features at various scales and orientation by swiping over the input feature map. More extensive, contextual information is extracted by layers with larger kernel sizes (such as 7x7 or 5x5), which makes them appropriate for spotting larger objects. They identify global patterns and relationships as they have a broader receptive field that captures data. For medium-sized objects, layers with intermediate kernel sizes (such as 3-by-3) provide a balance between the large and small features. Their receptive field preserves localization while capturing an adequate area to identify object properties.

Layers with smaller kernel sizes (such as 1-by-1) concentrate on high-resolution data and fine-grained features that are essential for identifying small objects. Their limited reception field focuses on certain areas, identifying exact local characteristics. Through activation scaling and adjustment, batch normalization enhances the MF networks' speed, performance, and stability. BN computes the mean and variance of the activations for each mini-batch during training. Normalized values are then adjusted using two learnable parameters per activation. This enables the model to reverse the normalization if it finds it to be ineffective. Backpropagation is used to learn the normalizing settings. In doing so, overfitting is decreased, internal covariate shift is decreased, and training speed is increased. Additionally, learning is stabilized by this.

In order to ensure that the output has the same dimensions as the input, padding adds zeros with care around the input feature map. This makes subsequent analysis easier and enables deeper architectures. The extent that the filters move across the input in each step is adjusted by the Stride hyperparameter. There is no skipping implied by a stride of 1, and the output size in that dimension is halved by a stride of 2. The model regulates the spatial resolution of the extracted characteristics by altering the stride between layers. A ReLU follows the Convolution layer. The introduction of nonlinearity allows for the detection of intricate correlations among features and the expression of more sophisticated patterns in the data.

In this case, the depth (channel) features and spatial feature learning are decoupled by depthwise convolution. A 3-by-3 depthwise convolution convolves each input channel (depth channel) of an input tensor independently with a different convolution filter. Depthwise convolution aids color-related issues by extracting information that is specific to each of the three channels (blue, green, and red) in the RGB images. Compared with a standard convolution, the computational cost is substantially lower because the filters are applied independently to each channel. This encourages computational and parameter efficiencies.

The dimensionality of the feature map is altered using the 1-by-1 pointwise convolution method. The output of the depthwise filters is combined using pointwise convolution after depthwise convolution. The depthwise filter output is projected onto a new channel space by the model using a 1-by-1 convolutional filter. It increases the efficiency and permits the number of feature map channels to be changed without affecting the spatial dimensions. By directly modeling the channel-wise interdependencies, the squeeze-and-excite block is a structural component that recalibrates channel-wise feature responses adaptively. Global Average Pooling was employed in the model to execute a squeeze to obtain channel-wise statistics, which are a type of feature descriptor.

In order to model channel-wise dependencies, these statistics are passed to the Excite gating mechanism, which employs a sigmoid activation function. The output of the gating mechanism is used to scale or recalibrate the original feature maps. By focusing on the most informative features and enabling adaptive recalibration by highlighting relevant features and suppressing less helpful ones without appreciably increasing the computational overhead, this enhances representational power and increases performance.

A new feature map that integrates data on tiny, medium, or large features depending on the kernel size is the outcome of the MB block.

Input size, padding, stride, and a few filters per layer impacted the final dimension. The model repeats the MB block several times by expanding kernel sizes in subsequent blocks. In return, the model progressively extracts features at varying sizes, addressing all the range of object sizes to be detected. MB blocks play a critical role in feature extraction in the introduced encoder-decoder-based architecture for feature extraction. The model can identify and locate objects accurately, whether they are small or possess complex features, and understand the context of objects by perceiving their interaction with nearby elements.

Moreover, it is capable of identifying objects of varying sizes within an image by perceiving features at different scales. This architecture has three key stages: Head, Backbone, and Tail as shown in figure 4.3. The Head Section pre-processes the image, extracts local features using a Spatial Feature Extraction Block (SFEB), and captures the global context with a Global Feature Extraction Block (GFEB). It then combines these features to obtain a richer representation. The Backbone Section refines this combined information. It performs multiple rounds of processing involving convolutions, Batch Normalization for stable training, and dropout for preventing overfitting. Importantly, it utilizes residual connections to preserve informative details throughout the process. Additionally, element-wise multiplications were introduced to capture the interactions between the features learned at different stages.

Finally, the Tail Section considers the refined features, uses global pooling to capture the overall information, and transforms them through dense layers with activation functions. A dropout is utilized once more for strength, and a last dense layer produces the detections, perhaps class probabilities or bounding boxes with class probabilities for objects in the image. Utilizing different techniques like feature extraction, combination, refinement, and elementwise multiplications, this CNN architecture hopes to produce precise object detection in images.

The SFEB depicted in figure 4.4 a utilizes convolutions to capture spatial features, followed by normalization and activation to improve the training stability and introduce nonlinearity. The depthwise separable convolution provides an efficient way to extract features while reducing the computational cost compared to standard convolution. By applying these operations sequentially, the SFEB aims to extract informative and robust

spatial features from the input data, which are then used for higher-level tasks such as object detection using 4.1 through 4.15.

$$\text{SFEB}_{\text{out}} = \text{Conv2D}(\text{BN}(\text{ReLU}(\text{Conv2D}(\text{Zero Pad}(\text{Ipreprocessed}), F_s, k_s))), F_s, k_s) \quad 4.1$$

The GFEB shown in Figure 4.4 b captures the Global Information and extracts high-level features. By reshaping the feature map, GFEB considers information from all spatial locations, not just specific regions. The 1-by-1 convolutions and activation functions help to transform the data into a more compact representation that captures the global characteristics relevant to the task.

$$\text{GFEB}_{\text{out}} = \text{Conv2D}(\text{BN}(\text{ReLU}(\text{Reshape}(\text{SFEB}_{\text{out}}, -1))), F_g, k_g) \quad 4.2$$

FFB produces a more detailed representation that takes advantage of both spatial information and global context, enhancing the model's object detection capability of an image. The block depicted in figure 4.4 c enhances the information to extract both local information and global context for object detection. The SFEB analyzes the input image to capture local details and preserves the image's spatial dimensions using zero padding which adds extra borders around the data. The SFEB then identifies patterns that are specific to various locations within an image.

$$\text{FFB}_{\text{output}} = \text{SFEB}_{\text{output}} * \text{GFEB}_{\text{output}} \quad 4.3$$

The SGFB takes the combined features from the FFB, refines them with a convolution layer, improves the training stability with Batch Normalization, and introduces some randomness with dropout. The combination shown in figure 4.4 d helps the model to create more robust and informative representations suitable for object detection tasks.

$$\text{SGFB}_{\text{output}} = \text{Dropout}(\text{Batch_Normalization}(\text{Conv2D}(\text{FFB}_{\text{output}}))) \quad 4.4$$

The Head section prepares the input image and extracts initial spatial and global features. These features were then combined to provide a more comprehensive representation for the following feature extraction and object detection tasks.

For a scaling factor (s) and normalization factor ($\bar{x}$),

$$\text{I}_{\text{preprocessed}} = s * (\text{I} - \bar{x})(\text{element - wise subtraction}) \quad 4.5$$

For filter sizes (F) and kernel sizes (k) for each convolutional layer where F_s and k_s represent SFEB layers and F_g and k_g represent the GFEB layers.

$$\text{FFB}_{\text{out}} = \text{FFB}(\text{ZeroPad}(\text{Conv2D}(\text{BN}(\text{ReLU}(\text{Conv2D}(\text{I}_{\text{preprocessed}}, F_s, k_s))))), F_{\text{ffb}}, k_{\text{ffb}})) \quad 4.6$$

$$\text{SFEB}_{\text{out}} = \text{Conv2D}(\text{BN}(\text{ReLU}(\text{Conv2D}(\text{ZeroPad}(\text{I}_{\text{preprocessed}}), F_s, k_s))), F_s, k_s) \quad 4.7$$

(Similar to FFB_{out})

$$\text{GFEB}_{\text{out}} = \text{Conv2D}(\text{BN}(\text{ReLU}(\text{Reshape}(\text{SFEB}_{\text{out}}, -1))), F_g, k_g) \quad 4.8$$

Flattening the spatial dimensions of SF EBout into a feature vector is performed by $\text{Reshape}(\text{SFEB}_{\text{out}}, -1)$ 4.9

$$\text{Head}_{\text{out}} = \text{FFB}_{\text{out}} * \text{ReLU}(\text{SFEB}_{\text{out}}) \text{ (Element-wise multiplication)} \quad 4.10$$

where $\text{Conv2D}(\text{I}_{\text{preprocessed}}, F_s, k_s)$ applies F_s filters of size $k_s \times k_s$ to the preprocessed input, capturing spatial features. $\text{ReLU}(I)$ applies the activation function ($f(x) = \max(0, x)$) element-wise, introducing non-linearity. $\text{BN}(I)$ performs Batch Normalization to improve training stability. $\text{Conv2D}(\dots, F_{\text{ffb}}, k_{\text{ffb}})$ are other convolutional layer within the FFB block.

The Backbone section progressively refined the features extracted from the Head Section. It utilizes residual connections (addition) to preserve informative features and element-wise multiplication to potentially capture feature interactions. This process aims to create more complex and robust feature representations for the final object detection tasks.

$$\text{Backbone}_{\text{in}} = \text{Conv2D}(\text{BN}(\text{Head}_{\text{out}}), F_{\text{bn}}, k_{\text{bn}}) \quad 4.11$$

(Similar to Conv2D in F F B/SF EB)

Repeated SGFB application,

$$\text{Block}_{\text{out}} = \text{Conv2D}(\text{BN}(\text{Dropout}(\text{Conv2D}(\text{BN}(\text{SGFB}(\text{I})), F_{\text{sg}}, k_{\text{sg}}))), F_{\text{sg}}, k_{\text{sg}}) + \text{BN}(\text{I}) \quad 4.12$$

$$\text{FB}_{\text{out_bb}} = \text{FFB}(\text{Backbone}_{\text{out}}, F_{\text{ffb}}, k_{\text{ffb}}) \quad 4.13$$

(Similar to F F Bout in Head section)

$$\text{SFEB}_{\text{out_bb}} = \text{Conv2D}(\text{BN}(\text{ReLU}(\text{Conv2D}(\text{Backbone}_{\text{out}}, F_s, k_s))), F_s, k_s) \quad 4.14$$

(Similar to SF EBout in Head section)

Final Feature Extraction uses the same F_s and k_s as defined earlier for SFEB layers. Here, $\text{SGFB}(I)$ represents operations within the Spatial Global Fusion Block (likely involving convolutions and pooling), and $\text{dropout}(I, p)$ randomly drops a percentage (p) of activations during training (not shown in the equation). The Tail Section in the present model consumes the fine-grained features extracted from the backbone to produce the end detections. Classification and Bounding Box Regression were executed. The units of the

first dense layer (n_dense1) and output classes ($n_classes$) were specified. In this case, $n_classes$ is the count of object categories detectable by the model.

$Tail_out_Detections = Dense_n_classes(Dropout(Dense_n_dense1(ReLU(GlobalAveragePooling2D(SFEB_out_bb)))))$ 4.15

where $GlobalAveragePooling2D(I)$ averages the activation's across the spatial dimensions (width and height) for each channel, and $Dense(I, n)$ applies a linear transformation on the input vector I with n output units, followed by the ReLU activation.

4.3.4 Inference-Optimized LSDNet Context-Aware Adaptation for Robust Object Detection in small Object and Low-Illumination Scenarios

Conventional object detection models often struggle in low-light conditions, failing to detect small or distant objects effectively. To address these challenges, the paper proposes LSDNet (Low-Light and Small Object Detection Network), an inference-level adaptation framework designed to enhance nighttime object detection. LSDNet builds upon the EfficientDet-D0 architecture, incorporating adaptive brightness correction, feature-aware confidence modulation, and class-specific detection prioritization.

The model dynamically adjusts detection sensitivity based on object size, illumination levels, and environmental context, improving both precision and recall without altering the underlying network structure. As shown in Table 4.2, the model integrates illumination-driven processing modulation, where brightness and contrast corrections are applied dynamically based on computed environmental light statistics. The size-weighted score based confidence scaling has enabled LSDNet to detect low-contrast small objects in nighttime. Efficient-Det [108], which is the underlying architecture utilized within this research, is a collection of models that scales up or down in parameters and FLOPS while maintaining state-of-the-art accuracy with an emphasis on efficiency. It has BiFPN and compound scaling technique for peak performance.

To enable detection in varying light conditions and to promote detection of smaller objects, this chapter presents LSDNet (Low-Light and Small Object Detection Network), an inference-level adaptation framework that adapts detection confidence dynamically with respect to feature-aware metrics and environmental factors. The method proposed is supported by the application of post-processing methods, which enhances low-light

resistance and detection of small objects at minimal computation. The method is a scalable real-time solution that does not alter the object detection model and thus is a promising modification for the application under discussion.

Table 4.2 Overview of Night Driving Challenges addressed in the current research and the key contributions to enhance nighttime driving.

Challenge	Contribution	Solution
Object detection is compromised at the highly varying and noisy nature of nighttime environments like low-contrast and objects that are subjected to glare	Dynamic Feature-Aware Adjustments for Architectural Augmentation	The information extraction from the degraded data is improved by the dynamic adaptation of the features for processing depending on the image characteristics.
Smaller objects at night pose the challenge of detection in low-light conditions, often blending with the background	Detection with an emphasis on Relative Size	Relative size based sensitivity adjustment to improve small or farther object detections like pedestrians, cyclists and so.
Standard object detection models treat all the classes equally but nighttime object detectors need to prioritize a few (Say pedestrian Vs Road signs)	Class-Conditional based Detection	Critical classes are prioritized by adjusting the detection confidence or loss dynamically leading to higher accuracy in essential object detections.
Varying Nighttime driving environments (Say Rural Vs Urban, street light conditions) which compromises detections	Environmental-Conditioning for Architectural Augmentation	Performance improvement by integrating the environmental context (Say., the light levels, road type)
Varying image quality due to changing illumination levels leading to inconsistent performance	Processing Modulation that is Illumination-Driven	Detected illumination condition is used to adjust the processing parameters, and applying image enhancement techniques to ensure a consistent accuracy

4.3.4.1 Methodology

To improve object detection in difficult scenes, this paper presents an optimized object detection model with Efficient Det D₀ architecture as the base line model with pre and post

processing modifications to cater to the need of nighttime detections as shown in figure 4.5. Efficient-Det uses complex scaling technique which scales up all the three dimensions: width, depth and resolution concurrently which leads to higher accuracy and efficiency than other models. The weighted BiFPN further enables detection of objects of varying sizes by improving the standard FPN through learnable weights and information flow that is bidirectional. For LSDNet, this model is chosen as baseline because of its inherent balance between accuracy and computational cost.

The hierarchical feature maps are extracted using the backbone along with convolutional feature extraction for refining the spatial information without compromising semantic comprehension. LSDNet preserves the baseline by making significant adaptations particularly for low-light (or varying brightness factor) through contrast correction in addition to changes to meet challenges of small object detections with the feature maps that are extracted making it robust. The detection head lays the bounding boxes, predicts the classification and confidence scores from the feature maps. The BiFPN is responsible for multi-scale feature extraction.

The normalized values are generated by the head along with probability of each detection. For further improving the detections, Class specific score adjustments to cater to diverse object sizes and a with more stress on categories like pedestrians, cars, and vehicles are considered. The scaling factor further improves confidence score of small object detections, making it more pronounced in a cluttered scenario. Tail module further is responsible for fine-tuning the raw predictions to improve accuracy.

In the LSDNet's post-processing segment, Non-Maximum Suppression discards the detections that are redundant, confidence thresholding to eliminate low-confidence predictions, and refining bounding boxes for small object detections refinement. LSDNet's dynamic confidence thresholding enables adapting to varying brightness levels. The augmentation with metadata (number of small objects detected, mean brightness for day/night classification, and the number of valid detections after thresholding) improves predictions that are context-sensitive. The LSDNet model shown in Figure 4.5, with the introduced modifications, is tailored for night-time real-time object detection, thus being eminently suitable for vehicle adaptive headlamp systems.

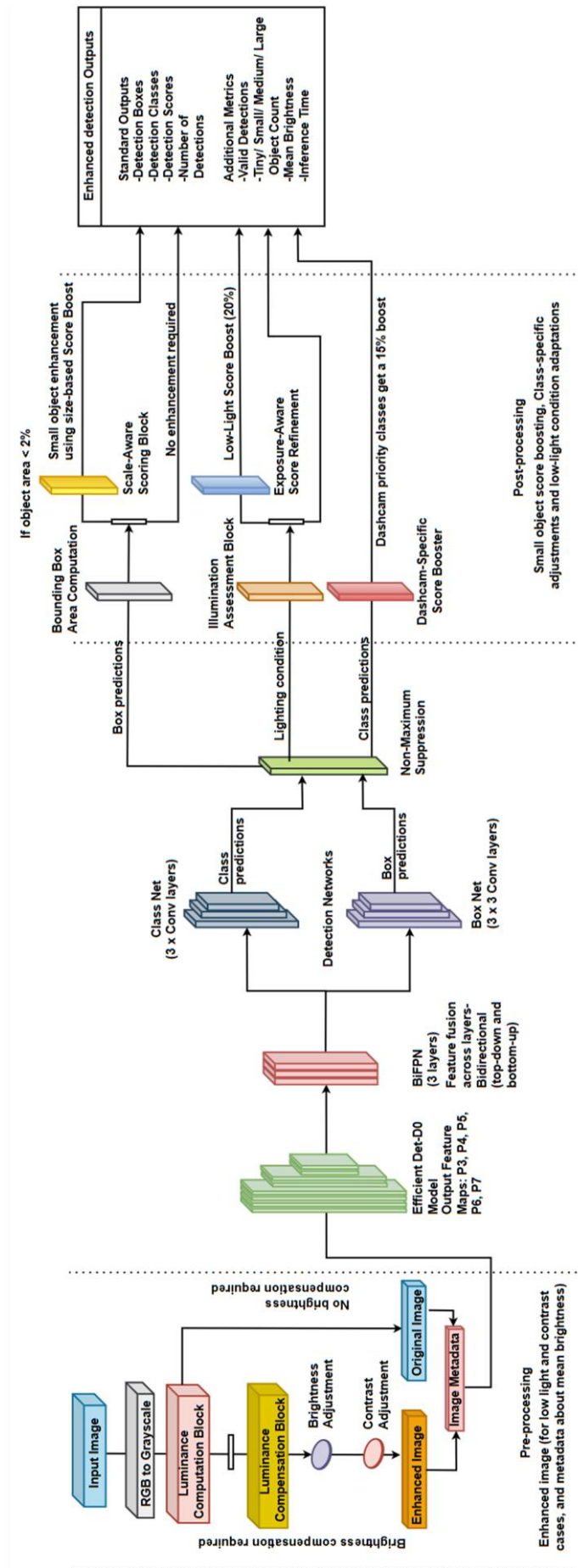


Figure 4.5 The overall LSDNET Architecture

This section explains the modifications and the relevance of each improvement on increasing on-road safety and adaptive lighting systems. The model consists of pointwise channel expansion, depth-wise separable convolution operators, linear projection transformation, statistical normalization methods, and residual information channels.

The integrated elements support smooth gradient propagation, enhanced multi-scale feature extraction, and computational efficiency—primary demands of real-time object detection. Hierarchical deep network feature extraction has been created to improve representational accuracy and eliminate redundant computational paths. This is achieved by sequential application of channel-wise augmentation, statistical normalization, and non-linear activation in conjunction to increase the network expressiveness. Feature dimensionality is first enhanced to support more expressive representational embeddings prior to using spatial convolutional transformations.

Table 4.3 Major modifications in the LSDNet Model

Section	Function	Modifications in this Model
Backbone	Extracts feature maps	Adaptive brightness & contrast adjustment
Head	Predicts bounding boxes & classes	Boosts for small objects & night-time conditions
Tail	Filters & refines detections	Custom NMS, metadata extraction, low-light adjustments

The mathematical representations of the same are explained using equations 4.16 through 4.37. Enhancement is done through the application of a pointwise convolutional operation:

$$F_{\text{expand}} = \sigma (W_{\text{expand}} * F_{\text{input}} + b_{\text{expand}}) \quad 4.16$$

Where $W_{\text{expand}} \in \mathbb{R}^{1 \times 1 \times C_{\text{in}} \times C_{\text{expand}}}$ represent the expansion kernel, b_{expand} is the associated bias term, $\sigma(x)$ is x times sigmoid(x) and is the SiLU activation function that enables augmentation of the representational non-linearity. The increase in the channel's dimensionality increases expressiveness in features, enabling a more detailed decomposition of complex patterns. Statistical normalization methods, such as batch normalization, regulate activation distributions to ensure the learning process is stable:

$$F_{bn} = \eta_{bn} \frac{F_{expand} - \bar{x}}{\sqrt{\sigma^2 + \epsilon}} + \phi_{bn} \quad 4.17$$

where $\bar{x}$ and σ^2 , represent the mini-batch activations mean and variance, while η_{bn} and ϕ_{bn} denote the learnable affine transformation parameters. Depth-wise convolution is channel-wise, and it is used to model spatial dependency effectively:

$$F_{dwconv} = W_{dw} * F_{bn} + b_{dw} \quad 4.18$$

W_{dw} is the spatial convolutional 3-by-3 kernel, and b_{dw} the associated bias. Essential non-linearity is incorporated using the activation function,

$$F_{act} = F_{dwconv} \cdot \sigma(F_{dwconv}) \quad 4.19$$

To keep the computation manageable, the feature dimensions are compressed using a projection layer:

$$F_{project} = W_{project} * F_{act} + b_{project} \quad 4.20$$

where $W_{project}$ performs the dimensionality contraction operation that ensures compatibility of the output with C_{out} . To further enhance stability, a normalization phase follows.

$$F_{bn2} = \eta_{bn} \frac{F_{expand} - \bar{x}}{\sqrt{\sigma^2 + \epsilon}} + \phi_{bn} \quad 4.21$$

Residual propagation mechanisms are incorporated when there is similarity of dimensions in input and output feature representations:

$$F_{output} = F_{input} + F_{bn2} \quad 4.22$$

This procedure enables information retention with reduced gradient attenuation in deep networks. Best object detection strategies leverage hierarchical feature pyramid networks in order to blend multi-scale features and maintain their balance. This can be expressed computationally as:

$$F_{pyramid} = \sum_{i=1}^n W_i * F_{backbone}^{(i)} \quad 4.23$$

Where W_i represents the learnable transformations at i hierarchical levels. To enhance feature representations at different scales:

$$F_{fused} = \text{Conv}(1\text{-by-1})(F_{pyramid}) + \text{Conv}(3\text{-by-3})(F_{pyramid}) \quad 4.24$$

This augments contextual granularity and local feature refinement. Object detection has two major objectives: precise location and classification into classes. The bounding box coordinates are optimized through a fully connected transformation:

$$B = W_{\text{bbox}} * F_{\text{fused}} + b_{\text{bbox}} \quad 4.25$$

where W_{bbox} parameterizes the spatial adjustments.

Probability of object's presence is gauged by

$$P_o = \sigma (W_{\text{conf}} * F_{\text{fused}} + b_{\text{conf}}) \quad 4.26$$

Class posterior distributions are estimated by:

$$P_c = \text{softmax}(W_{\text{cls}} * F_{\text{fused}} + b_{\text{cls}}) \quad 4.27$$

For removing duplicate predictions:

$$\hat{B} = \text{NMS} (B, P_o, \text{IoU}_{\text{threshold}}) \quad 4.28$$

where extended $\text{IoU}_{\text{threshold}}$ creates a non-maximum suppression constraint. For improving recall of small objects:

$$W_{\text{small}} = f_{\text{small}} (P_o) \quad 4.29$$

where W_{small} is a weight that increases recall for small objects, f_{small} reweights detection confidences. The class-specific confidence estimates are optimized by

$$W_{\text{class}} = f_{\text{class}} (P_c) \quad 4.30$$

where W_{class} is a weight applied to modify the class confidence and f_{class} recalibrates class probabilities.

LSDNet has the advantage of improving object detection without altering the baseline model parameters through controlled adjustments in the model detections. This pre and post-processing paradigm approach acts directly upon the tensor-based computation and the contextual augmentation improves statistical confidence summaries and classifications which is beneficial for downstream processing. Here the robustness of detection is improved over complex scenarios when compared to the baseline models through these inference-level adaptations, enhancing low-light performance and detection of small objects.

LSDNet uses scaling factor (a 1.3x) to increase confidence for small object detections that usually is missed in the standard models. Small objects here are referred to as objects that take up not more than 2% of the image area which when improved through this post-processing modification ensures detections of vital small objects like pedestrians, animals that are considerably far, and roadside debris.

The confidence score S is adjusted as

$$S' = S * (1 + \lambda_s (1 - \text{NAR})) \quad 4.31$$

$$S' = S \text{ (when } \text{NAR} \geq 1) \quad 4.32$$

Where S refers to the Object detection confidence score, $\lambda_s=1.3$ is the scaling factor for the small objects. If A_i is the area of the total image in pixels², NAR is the Normalized Area Ratio which is the ratio of the detected object area (A_o in pixels²) and the Small object threshold value ($0.02A_i$ in pixels²). From Equ (2) it is implied that when $\text{NAR} \geq 1$, then there is no correction as S' may be equal to or lower than S , which means larger objects do not receive a boost and there is no change for large objects. The model includes a night and low-light detection enhancement mechanism that enhances the object detection efficiency in less-than-ideal lighting.

It assesses the ambient light by computing the average pixel intensity of the input image. When the brightness is established to be lower than a threshold, a factor is used to boost the detection reliability. Additionally, brightness and contrast adjustments are performed adaptively to optimize feature visibility before processing the image via the detection pipeline. In extreme low-light, the system identifies the surroundings as nighttime and uses special detection settings customized for such environments. This real-time brightness adjustment ensures consistent and stable detection performance in a variety of lighting levels.

The Low-light detection scaling factor (Δ_L) is given by

$$\Delta_L = 1 + [(\lambda_L - 1) \cdot H(\Delta_{LLD})] \quad 4.33$$

where $B(I)$ represents the Mean brightness of the input image which varies from 0 to 255, T_L is the Low-light threshold (value less than 90 is considered dark), λ_L is the Low-light scaling factor (fixed at 1.2) and α refers to the brightness gain factor through which the intensity of enhancement is controlled. $\Delta_{LLD} = T_L - B(I)$ is the difference in brightness

from the low-light threshold. $H(x)$, the heaviside step function becomes 1, when $x > 1$ and is 0 otherwise. At low light conditions, $B(I) < 90$ and $T_L > B(I)$, $H(x)$ becomes 1, λ_L value is 1.2, hence Δ_L becomes 1.2. Therefore, at low-light conditions, scaling factor is increased as 1.2 to ensure that the necessary compensation is applied. When Brightness is high, $T_L < B(I)$, therefore $H(x)$ becomes 0 and Δ_L remains at 1 which means that no scaling is applied when brightness is sufficient.

The brightness correction is carried out using,

$$I' = \alpha \chi(\Delta_{LLD}) \cdot H(\Delta_{LLD}) \cdot \Delta_L \quad 4.34$$

Here, $H(\Delta_{LLD})$ ensures that compensation is applied only in low light conditions (Lane Lateral Deviation), Δ_L is the adaptive scaling factor, χ is the scaling parameter for fine tuning brightness correction strength to determine as how aggressively will the model correct when $T_L > B(I)$. A high value of α indicates strong correction while a low value represents softer corrections. Here Δ_L decides when and how much correction is needed, χ and ensures it is not too strong nor very weak. Since LSDNet is a low-light object detection model, it is necessary to maintain a good image contrast while the brightness is altered. To ensure that the contrast is preserved,

$$\chi = 1 + \zeta (1 - LSF) \quad 4.35$$

where ζ controls the level of brightness correction (0.5 to 1), $LSF = B(I)/T_L$ and represents the light sufficiency for object detection. If image is very dark, $B(I) \ll T_L$, LSF will be small and hence $(1 - LSF)$ will lead to a stronger correction. If it is moderately dark, $B(I) \approx T_L$, $(1 - LSF)$ will be a small value and χ will lead to subtle correction and hence preventing over-brightening when brightness is close to T_L . When $B(I) \geq T_L$, $(1 - LSF)$ becomes 0 or negative making $\chi \approx 1$. Hence the model preserves natural brightness in these cases and does not lead to excessive brightness. The model dynamically adapts to changing brightness conditions.

Low-light adaptation is necessary to ensure detection accuracy in low-light environments. It reduces performance loss in fog or darkness by enhancing object perception. The mechanism helps adaptive headlamp systems adapt beam intensity and direction according to ambient light. By making dark objects and pedestrians in shadows more visible, it enhances road safety during nighttime. It also adjusts high beams automatically to enhance driver visibility and minimize glare for approaching traffic. Hence, class-specific score

adjustments place priority on significant traffic-related objects such as pedestrians, bicycles, and cars, enhancing detection of critical elements while minimizing false positives on non-essential objects. LSDNet includes class-specific score improvements to enhance dashcam use cases detections which makes confidence scores higher for a few significant on-road objects.

The correction factor is given by

$$\lambda_c = \begin{cases} 1.15, & C \in \text{dashcam} - \text{relevant classes} \\ 1.10, & C \in \text{small object classes} \\ 1.0, & \text{otherwise} \end{cases} \quad 4.36$$

$$S' = S * \lambda_c \quad 4.37$$

The model incorporates additional enhancements to refine object detection and improve adaptive headlamp performance. Adaptive confidence thresholding and Non-Maximum Suppression (NMS) ensure that only relevant detections influence headlamp adjustments, reducing noise and false positives. A day/night classification mechanism enables adaptive processing by differentiating between daytime and nighttime conditions, optimizing detection parameters accordingly. Furthermore, metadata extraction provides numerical insights, including small object count, average detection confidence, and brightness analysis, enhancing situational awareness. These enhancements highlighted in Table 4.3 collectively improve real-time object detection, allowing adaptive headlamps to dynamically respond to changing road conditions, ensuring safer night time driving. Metadata analytics offer real-time performance monitoring through aggregation of average detection confidence across all detections, class-specific detection counts to measure object frequencies per class, and bounding box distributions to estimate anticipated spatial areas. The architecture represents a high-performance deep learning architecture that is consistent with the expansion-depthwise-projection principles, integrating hierarchical multi-scale features and adaptive refinement strategies to increase detection competency. The architecture of the model is specifically tailored to improve predictive accuracy, spatial resolution optimization, and computational efficiency maximization, making it extremely suitable for real-time object detection tasks.

4.4 Summary of the object detection models discussed

This chapter has discussed the design and deployment of three cutting-edge AI models—Modified SSD, MultiFaceted AI Model, and LSDNET—fine-tuned for nighttime on-road

object detection. Having overcome some of the primary limitations of existing models like YOLO, Tiny YOLO, and SSD, the proposed frameworks bring new architectural enhancements, feature fusion methods, and inference-time efficiency. The Modified SSD enhances SSD with adaptive feature scaling and confidence modulation to improve detection of small objects under low-light conditions. The MultiFaceted AI Model adopts a hybrid strategy of spatial and global feature extraction, allowing improved robustness to different nighttime illumination conditions. LSDNET uses illumination-driven processing modulation, dynamic confidence scaling, and class-aware prioritization to greatly improve detection recall and accuracy for low-contrast and small objects.

These models are optimally balanced between accuracy of detection and computational efficiency and are applicable to real-time uses in autonomous navigation, adaptive headlight control, and intelligent surveillance systems. The breakthroughs introduced in this chapter set the stage for an in-depth performance analysis, which will be done in the next Performance Analysis Chapter, where the proposed models will be compared to current methods. The outcomes will confirm their efficiency in enhancing detection accuracy, recall, and resilience in difficult nighttime driving conditions

4.5 Results and discussion on the On-road object detection models

Object detection is still a key problem in computer vision, with many state-of-the-art models being proposed to enhance the accuracy, efficiency, and robustness to real-world challenges. This chapter provides a comparison of conventional and recent object detection models to identify how effective they are in detecting objects under different circumstances, such as low-light conditions and small object detection. The assessment consists of four popular object detection methods—YOLO, Tiny YOLO, SSD, and RCNN—comparing their strengths and weaknesses on the basis of critical performance indicators like precision, recall, mean average precision (mAP), and inference speed.

Subsequent to this, the performance of three suggested models—Modified SSD, Multi-Faceted AI Model, and LSDNET—is analyzed. These models have been developed to improve object detection performance by overcoming certain limitations seen in current methods. The comparative analysis brings out how these models perform compared to traditional methods, especially in difficult situations where traditional methods tend to fail. One of the main points of this debate is the capacity of these models to identify objects under low-light settings and small objects precisely, which have been ongoing challenges

for computer vision. By testing their efficiency in relation to conventional models, this analysis seeks to identify the most stable method for practical use. The results offer insightful perceptions of the strengths and areas of improvement for each model to ultimately direct future improvements in object detection technology.

4.5.1 Performance of YOLO, Tiny YOLO, RCNN and SSD Models

Object detection is a core problem in computer vision, and many deep learning models have been created to improve the speed, accuracy, and robustness of detection. From among the most used techniques, YOLO, Tiny YOLO, RCNN, and SSD have achieved considerable breakthroughs in real-time detection and localization of objects. The models are each different and offer benefits in various applications depending on computational complexity, detection accuracy, and handling of scenarios with complex environments. The subsequent sections examine these models on the basis of their structural variance, strengths, and possible drawbacks in object detection applications. The models are trained on an open-source COCO database comprising over 15,000 images and 93 classes, employing Python 3. The accuracy of object detection is evaluated using the mAP (Mean Average Precision), calculated as the average of Average Precision across different verification sets, alongside the two-dimensional Precision-Recall (P-R) curve for various thresholds. Detection speed, quantified in Frames Per Second (FPS), is also measured, considering factors such as learning rate and different losses.

4.5.1.1 YOLO Model

YOLO (You Only Look Once) is a real-time object detection model that detects an entire image in a single pass of a neural network, which makes it one of the fastest detection frameworks (figure 4.6 through 4.9). Through the removal of region proposal networks, YOLO provides high-speed detection, which makes it suitable for use in applications like surveillance, autonomous driving, and robotics.

4.5.1.2 Tiny YOLO

Tiny YOLO is a lighter version of the YOLO model that is used in environments with fewer computational resources. In decreasing the number of parameters and layers, Tiny YOLO loses some accuracy but with much better processing speed (Figure 4.10 through 4.13). The model is most appropriate for use in edge devices and real-time uses where efficiency is considered key.



Figure 4.6 YOLO model object detection for dense traffic at night

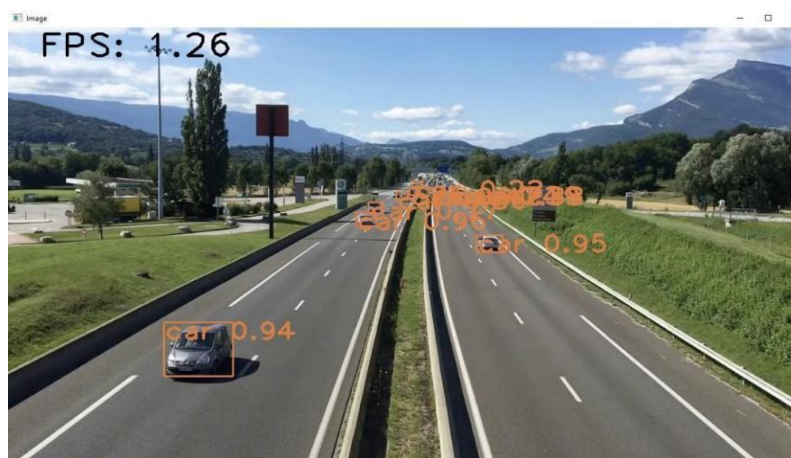


Figure 4.7 Frames per second on YOLO Model

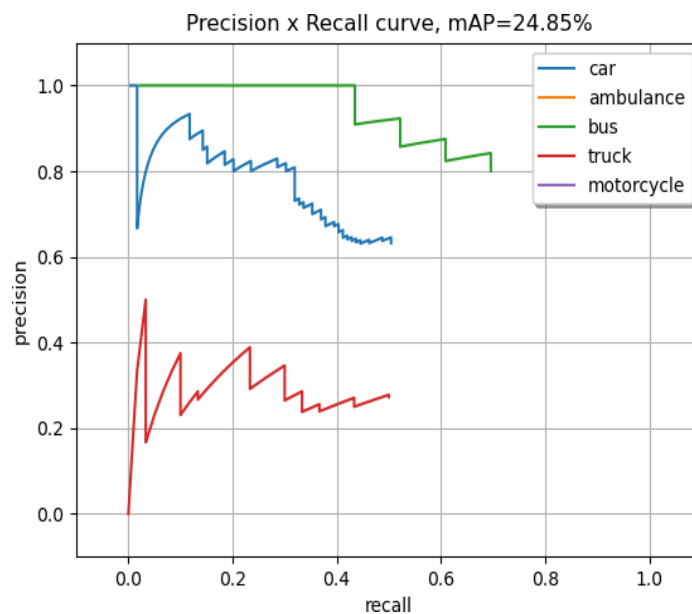


Figure 4.8 YOLO Model Emphasized by mean Average Precision

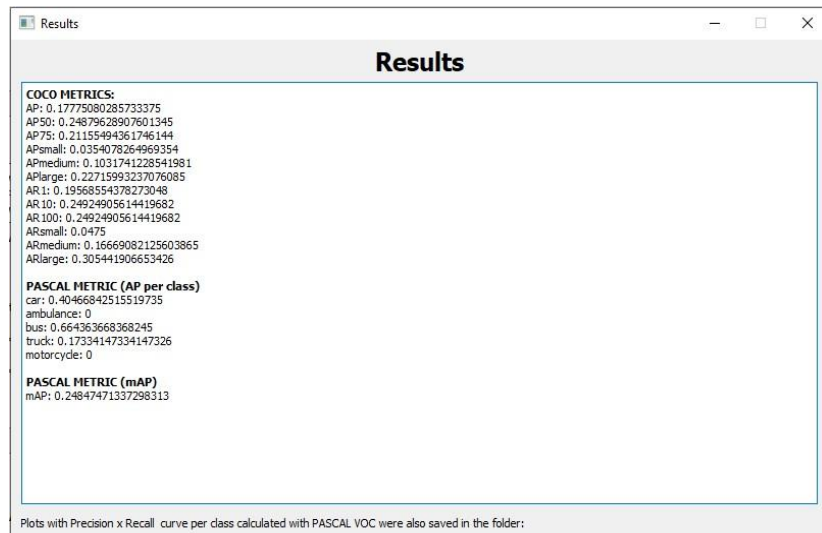


Figure 4.9 YOLO-COCO AND PASCAL Metrics

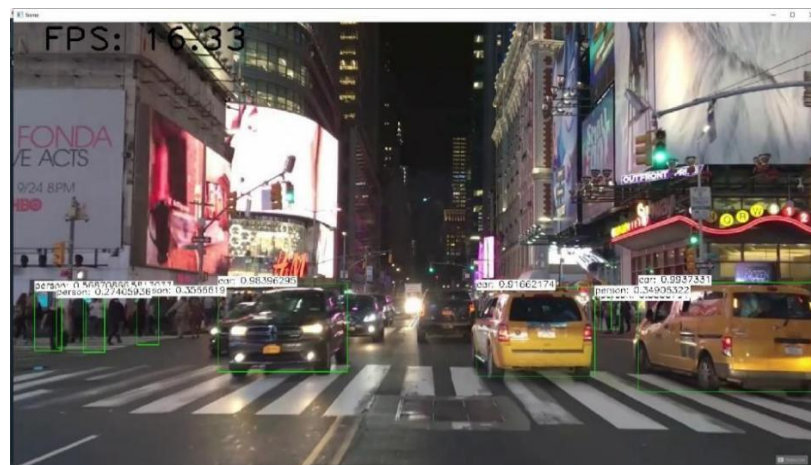


Figure 4.10 Object detection using YOLOv3-Tiny in dense traffic at night



Figure 4.11 Frames per second for YOLOv3-Tiny Model

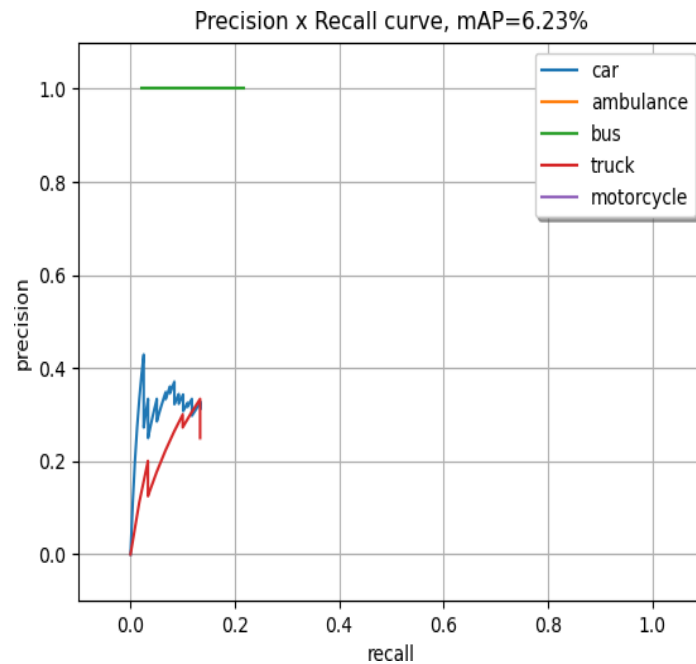


Figure 4.12 mean Average Precision value for YOLOV3 Tiny Model

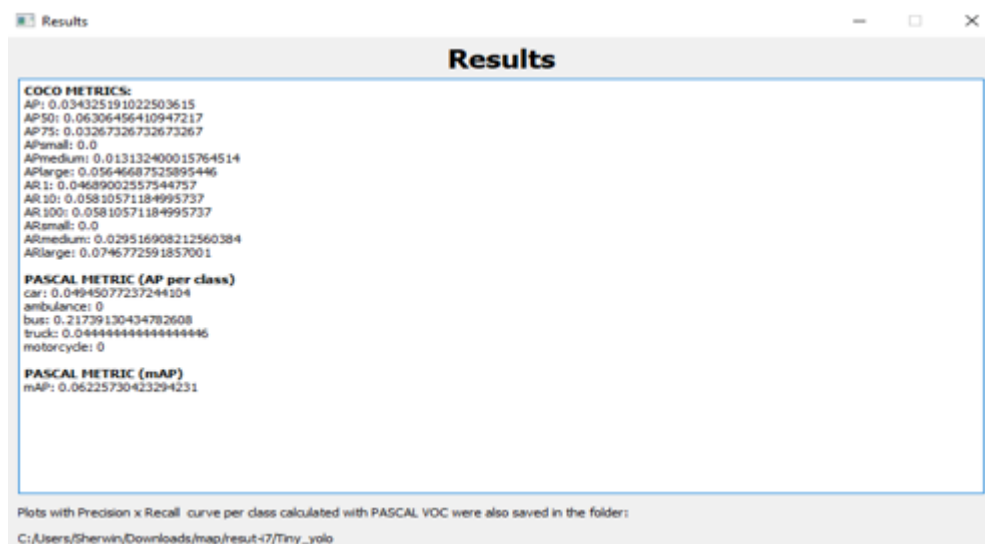


Figure 4.13 Screenshot of the statistics as obtained for the YOLO-Tiny Model calculated with PASCAL VOC

4.5.1.3 RCNN

RCNN (Region-Based Convolutional Neural Network) has a region proposal method, dividing an image into several regions prior to subjecting them to deep learning-based classification. Although RCNN ensures decent detection accuracy, it consumes huge computation power because of its multi-stage processing pipeline and is thus less ideal for real-time applications (figure 4.14 and 4.15).

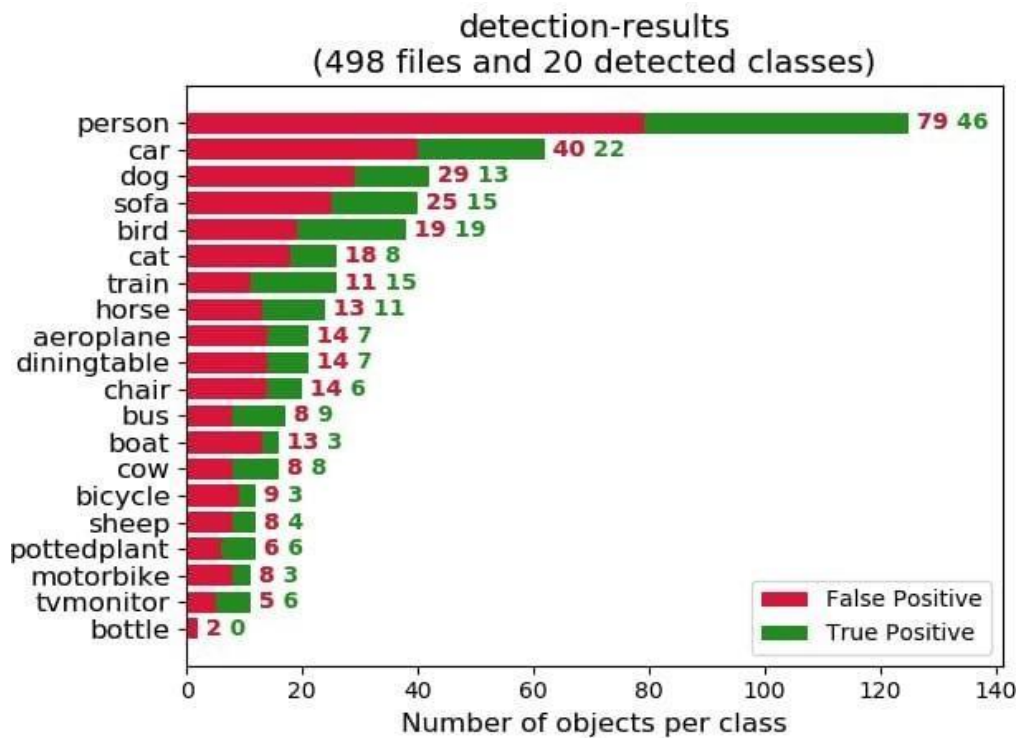


Figure 4.14 TP & FP for the RCNN Model

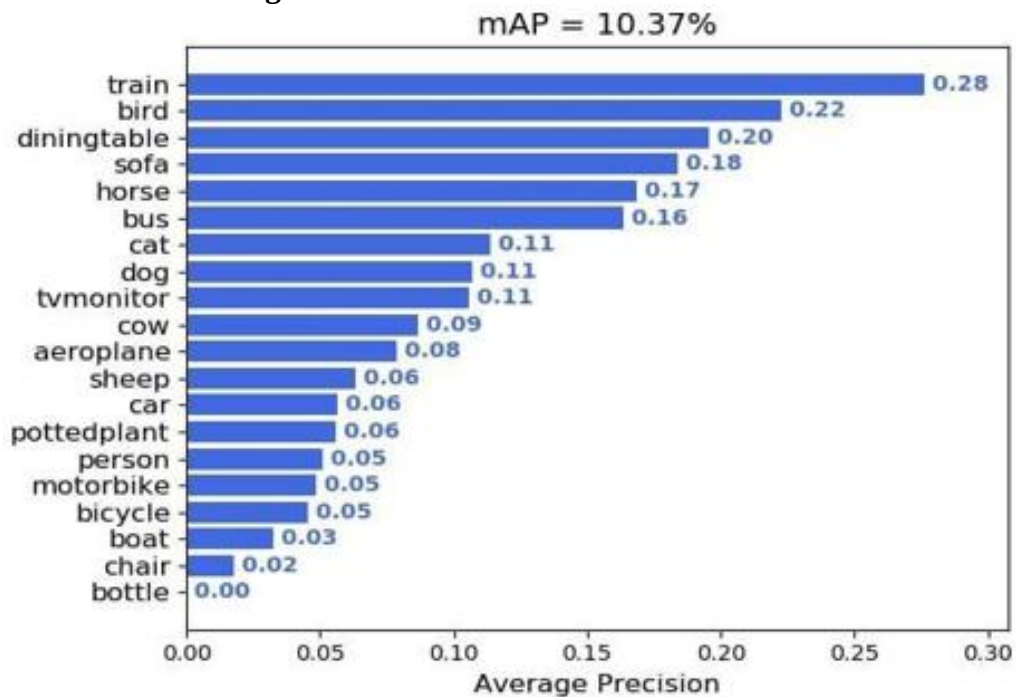


Figure 4.15 mAP of the RCNN Model

4.5.1.4 SSD Model

SSD (Single Shot MultiBox Detector) provides a compromise between speed and accuracy by making object location and classification predictions during a single network forward pass. In contrast to RCNN, SSD does away with region proposals, which results in much

faster processing without compromising on accuracy (figure 4.16 through 4.19). This model finds extensive applications where efficiency is a requirement along with accurate object localization.



Figure 4.16 Object detection using SSD Model in dense traffic at night

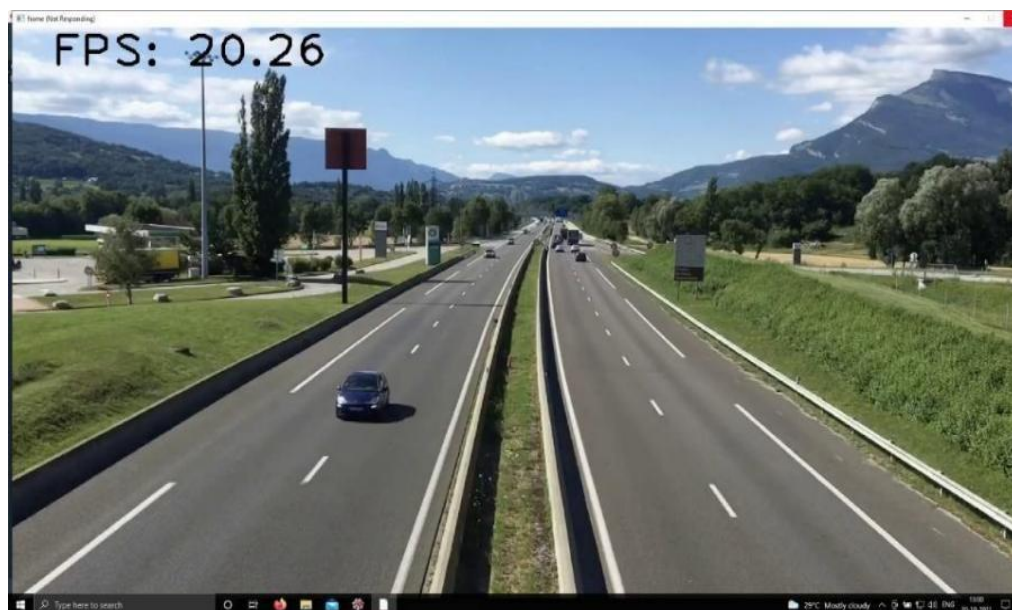


Figure 4.17 Frames per second for SSD Model

4.5.2 Model Comparison

Table 4.4 and 4.5 summarizes the performance metrics of several AI models. Both YOLO-tiny and SSD achieve an FPS value of 14.56, while YOLO attained 1.9 FPS. The mean Average Precision (mAP) values for YOLO, YOLO-tiny, and SSD are 0.33, 0.041, and 0.38 respectively. FPS and mAP values are crucial in determining the speed and accuracy

of detection for these models. Only mAP is assessed for RCNN due to its inadequate response, making it unsuitable for the current application. Furthermore, the RCNN model's performance under low illumination is compromised, leading to its exclusion from the comparison with other AI models that demonstrate accurate mAP and FPS results.

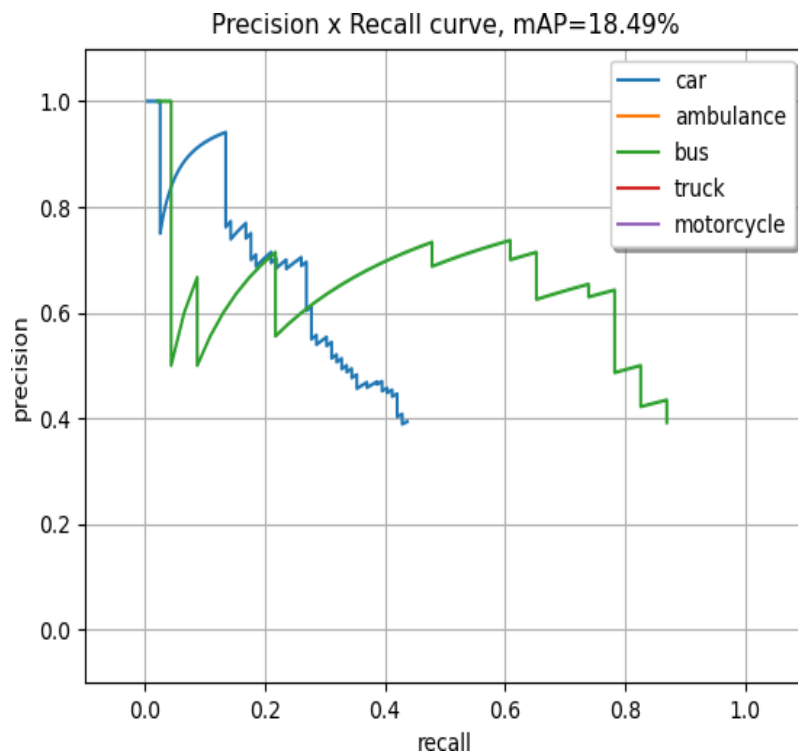


Figure 4.18 mean Average Precision value for SSD Model

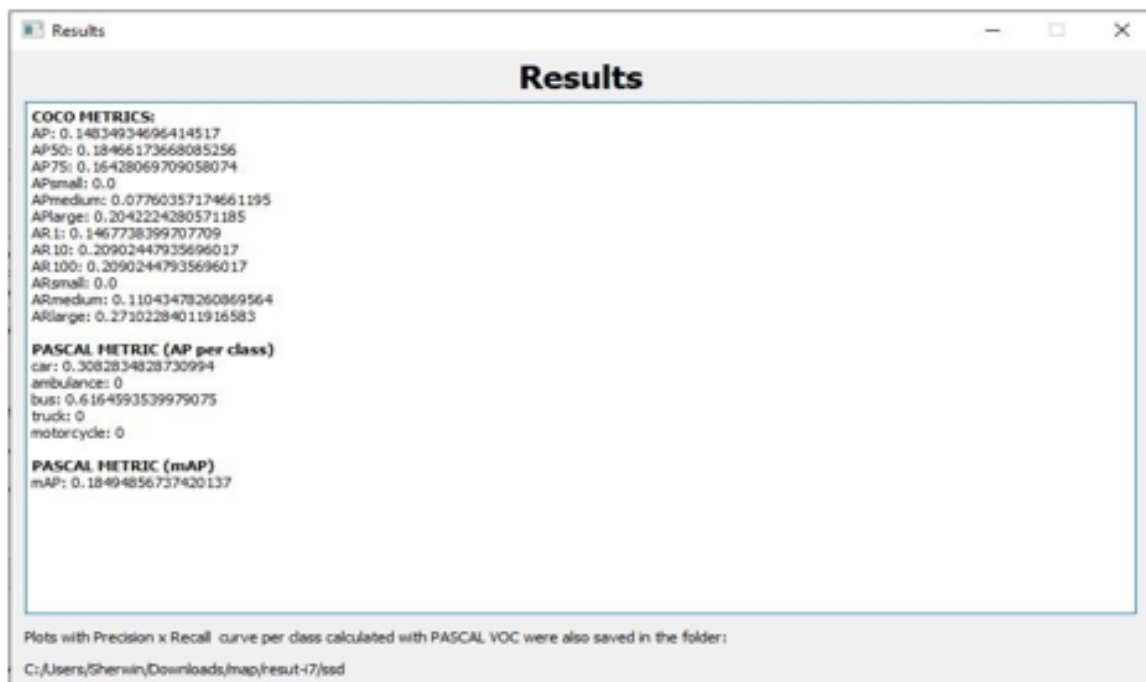


Figure 4.19 COCO & PASCAL metrics for SSD Model

Table 4.4: Comparison of the three Models based on Frames per Second and mean Average Precision

Model	FPS	mAP
YOLO	1.9	0.2484747
YOLO-Tiny	14.56	0.0622573
SSD	14.56	0.1849486

Table 4.5: Comparison of the three Models based Average Precision of detection for each class

Model	Car	Bus	Truck
SSD	30.8283	61.645	23.465
YOLO	40.46	66.43	17.334
YOLO V3-Tiny	4.95	21.74	4.44

The findings suggest that SSD (Single Shot Multibox Detector) appears suitable for the application, given its competitive mAP value of 0.1849486 and presumably efficient performance in various lighting conditions compared to RCNN. Therefore, SSD could be a favorable choice considering both its accuracy and adaptability to different environmental conditions. The next section of the discussion deals with modifying the SSD framework to improve accuracy for the application.

The performance analysis of YOLO, Tiny YOLO, RCNN, and SSD for real-time low-light object detection underscores essential trade-offs in accuracy, computational resource requirements, and resilience to extreme conditions. As impressive as YOLO is for speed, it is equally poor in precision at night under low-light scenarios due to low performance in handling small or even occluded objects. Small YOLO, even with optimization to handle quicker inference, shows additional degradation in detection accuracy, rendering it inappropriate for applications requiring high dependability in dark environments. RCNN offers better detection accuracy via its region proposal mechanism but is unsuitable for real-time applications due to its computational requirement and slow inference time. The model's dependence on multiple processing steps creates high latency, a major drawback for real-time applications like autonomous driving and surveillance. SSD comes as a more balanced solution with a trade-off between speed and accuracy, promising to be an excellent fit for real-time object detection in low-light environments.

Unlike RCNN, SSD removes the region proposals step, significantly decreasing inference time with comparable detection performance. The capability of the model to predict at multiple scales increases its performance for small and faraway object detection, an important requirement when applied under low-light applications with limited visibility. The architectural efficiency of SSD also enables it to be deployed on edge devices and embedded systems without requiring heavy computational resources. These features make SSD a viable option for real-time detection of objects in nighttime scenarios, where speed and accuracy are both significant. Yet, additional advancements like enhanced feature extraction and incorporation of low-light processing expertise may be required to maximize SSD's performance for more challenging real-world conditions.

4.5.3 Modified SSD Architecture

Since SSD exhibited high suitability for object detection at night, an optimized SSD architecture was created to optimize its performance even more. The introduced model simplifies SSD by removing individual convolutional layers, speeding up detection without compromising on accuracy. This differs from the base SSD, which uses diversified aspect ratios in accordance with feature map layers, enhancing detection at different scales. The model only uses 3×3 convolutional kernels to minimize parameters and computation. Other predictor layers further increase multi-scale detection, whereas batch normalization and L2 regularization increase training effectiveness and generalization. These updates increase the model's efficiency in real-time on-road object detection.

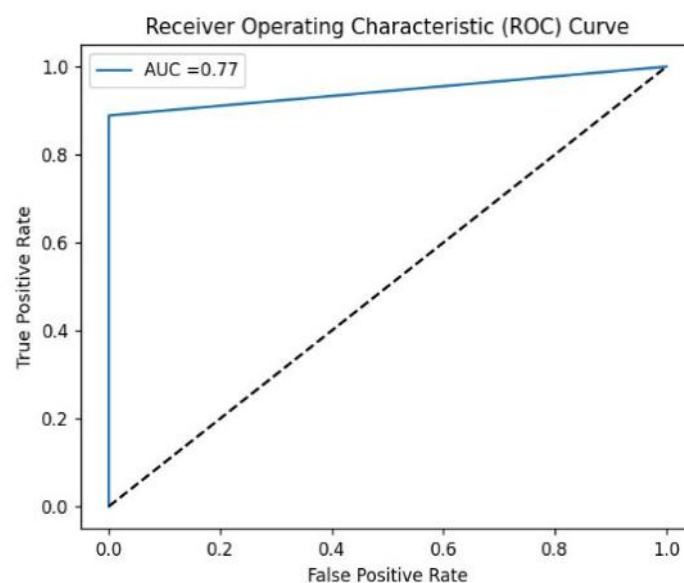


Figure 4.20 ROC Curve for Modified SSD

The performance metrics of the Mod SSD offer a strong balance between accuracy and performance. This is comparable to other models even if its accuracy (53%) and mean Average Precision (mAP) (0.42) are not the best. In addition, the Mod SSD offers an AUC of 0.77 (Figure 4.20), hints at the fact that the model can effectively detect objects in the background. F1 score of Mod SSD is notably faster than YOLO and Tiny YOLO, outperforming both categories while maintaining a far higher frame rate (23.8 FPS) than any of the other two.

Architectural modifications cause this increased speed, which may allow real-time applications that require quicker inference. This is still not enough, though, for the real-time night application. Therefore, an improved framework is required. If the work requires absolute peak accuracy, alternate models such as YOLO may be more appropriate. However, for cases that require a balance between performance and speed, the Mod SSD appears to be a good option. Despite not achieving the highest accuracy (53%) or mAP (0.42), Mod SSD demonstrates a strong balance between accuracy and performance, as depicted in Table 4. Additionally, Mod SSD's AUC of 0.77 (Figure 4.20) suggests effective object detection capabilities, outperforming YOLO and Tiny YOLO in both accuracy and speed, with a notably higher frame rate of 23.8 FPS.

4.5.4 YOLO V8 based Object detect

The YOLOv8 model uses a convolutional neural network (CNN) encoder-decoder architecture for semantic segmentation. The encoder learns hierarchical features through five convolutional layers with progressive filter sizes, utilizing a ResNet backbone for compact feature expression. C2f convolutional layers and max pooling for dimensionality reduction improve contextual representation without overfitting. The decoder restores spatial knowledge via bilinear interpolation and concatenation layers, improving segmentation accuracy. Bounding box layers combine with segmentation masks to offer end-to-end object detection, while softmax activation allows for accurate pixel-wise classification, making this model very efficient for real-time object segmentation and detection tasks.

The V8-based model's confusion matrix (figure 4.21) shows an overall accuracy of 78.4%. With only 6 false negatives and 19 true positives, the model can accurately recognize background items. Its effectiveness in separating the background from other classes is demonstrated by its low miss rate. 71 cars were properly identified, but there were causes

for concern over the 21 false positives and 9 false negatives. Notably, there appears to be some confusion between these groups because 16 motorbikes were incorrectly categorized as vehicles. Also, none of 76 real traffic lights could be recognized by the model. This class has serious problems, as indicated by its high miss rate. This model shows some uncertainty, but not as much as with traffic lamps; it correctly identifies five motorcycles as vehicles and incorrectly classifies six motorcycles as backgrounds.

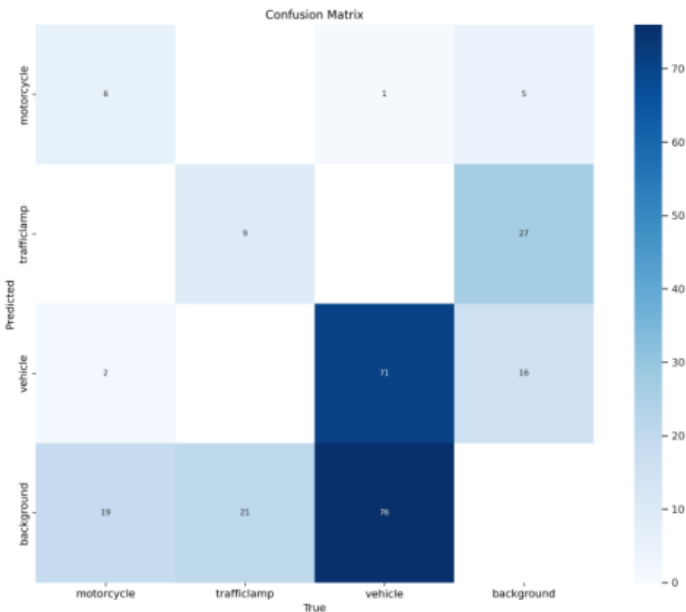


Figure 4.21 Confusion Matrix of V8 based Model

The Model displays performance constraints in balancing recall and precision, with an average F1 score of 0.4 at 50% confidence as shown in figure 4.22. Vehicles with the highest F1 score (0.5) at this confidence level were not as successful as those with lower confidence levels, suggesting problems with particular vehicle types. Motorcycles performed better when making predictions with moderate confidence (F1 score of about 0.6), but they had trouble with high confidence. A crucial area for improvement is highlighted by the traffic light’s poor performance, which included the lowest F1 score (0.2) at 50% confidence and continued to deteriorate with increased confidence. It’s interesting to note that the model showed less confidence in predictions for less common objects than automobiles, such as traffic lights.

This highlights the need for a customized model to identify specific problems and place targeted adjustments in place to obtain a more robust and balanced performance across all object classes and confidence levels. With an average mAP at 0.5 of 0.305, suggesting limits in object recognition, particularly with increasing recall as shown in figure 4.23. The

vehicle class had the highest mAP at 0.5 (0.553); however, as the recall increased, its precision decreased, indicating problems with particular car models. The motorcycle class struggled to increase recall despite having high precision (0.220) at low recall as shown in figure 4.25.

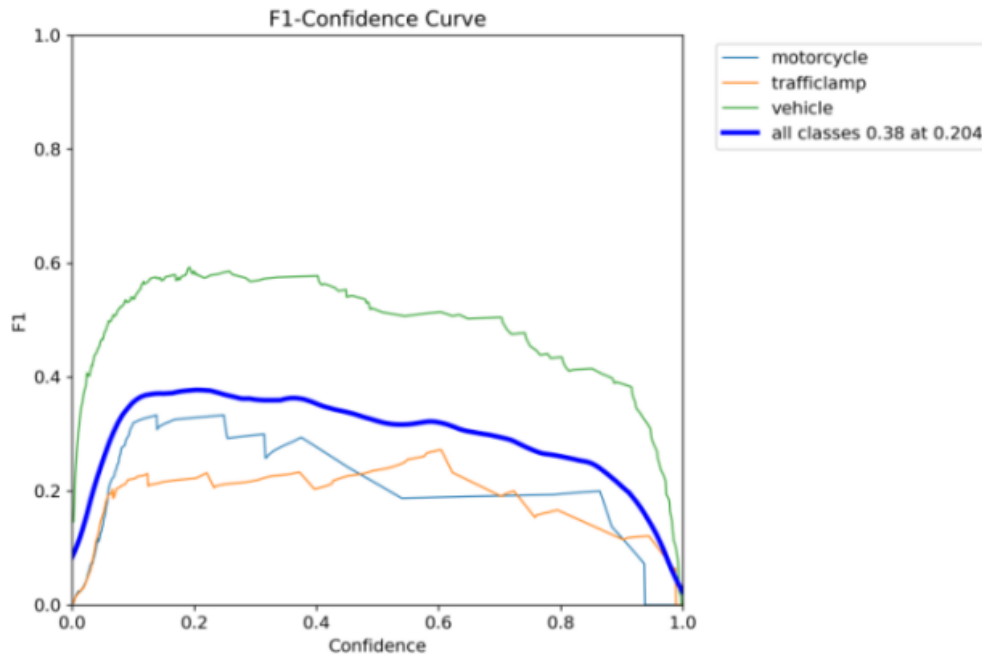


Figure 4.22 F1-Confidence matrix for V8 based Model

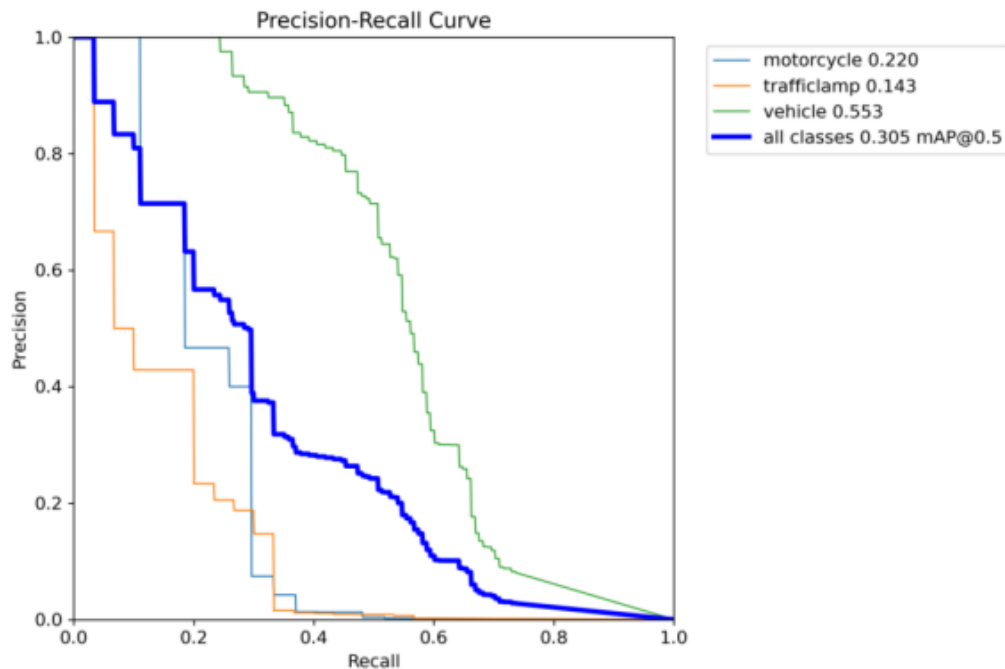


Figure 4.23 Precision-Recall for V8 based Model

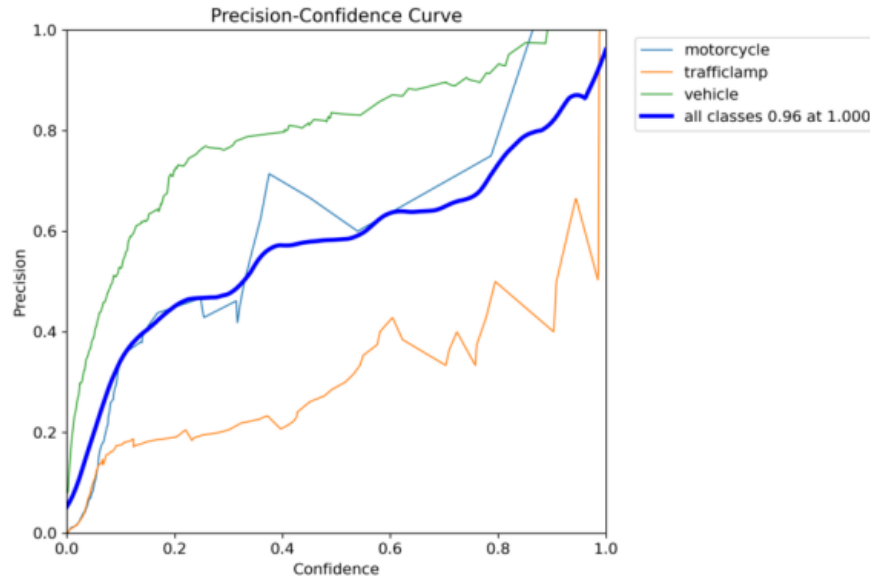


Figure 4.24 Precision-Confidence Curve- V8 Model

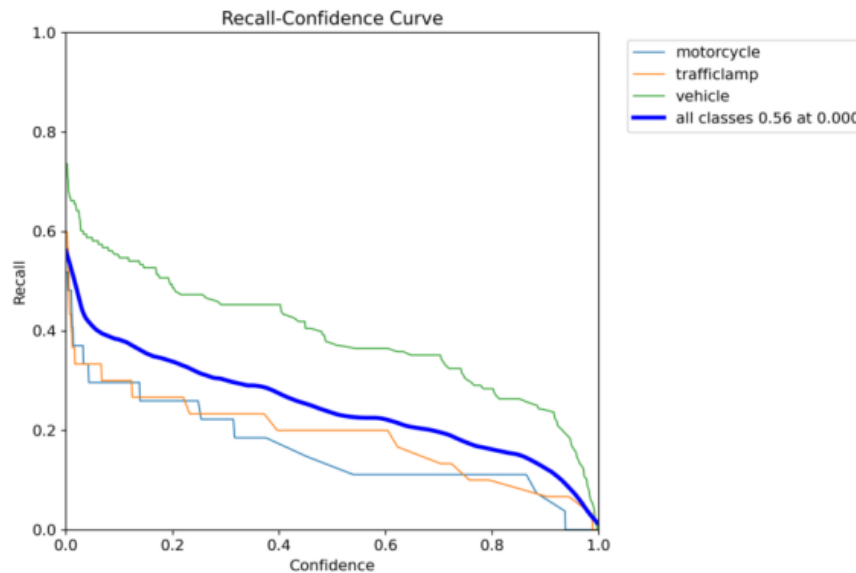


Figure 4.25 Recall-Confidence Curve- V8 based Model

Even at low recall, the traffic signal class had the lowest precision (0.143), indicating a need for significant improvement as depicted in figure 4.24. This performance can be caused by inadequate or non-diverse training data, inappropriate model architecture, or insufficient training time. Nevertheless, the model demonstrates promising overall performance, as accurate object detection is crucial for adjusting headlight beams, regardless of the object type. The model's performance seems to improve over training epochs, evident from decreased training and validation total losses in Table 4.6. However, the persisting gap between training and validation losses suggests some degree of overfitting.

Table 4.6 Model training and validation losses and learning rate for different epochs

Epoch	train/ box_loss	train/ cls_loss	train/ dfl_loss	metrics/ precision (B)	metrics/ recall (B)	metrics/ mAP50- 95 (B)	val/ box_loss	val/ cls_loss	val/ dfl_loss	lr/ pg0	lr/ pg1	lr/ pg2
1	2.201	3.21	1.42	0.276	0.263	0.047	2.381	11.3	1.56	0.00031	0.00031	0.00031
2	2.051	1.92	1.39	0.442	0.344	0.105	2.273	2.54	1.66	0.00043	0.00043	0.00043
3	2.030	1.52	1.34	0.418	0.230	0.082	2.367	2.19	1.79	0.00033	0.00033	0.00033

4.5.5 Performance of the proposed Multi-Faceted Framework

The Multi-Faceted (MF) Model uses a multi-block convolutional neural network design optimized for semantic segmentation. It incorporates an encoder-decoder model, in which convolutional blocks capture hierarchical features and upsampling layers improve spatial resolution for accurate segmentation. Skip connections improve feature preservation, allowing spatial consistency in pixel-wise predictions. Depthwise and pointwise convolutions enhance computational efficiency, while squeeze-and-excite blocks dynamically recalibrate channel-wise feature responses. This architecture supports multi-scale feature extraction, allowing accurate object detection of different sizes. With hierarchical processing in head and tail sections and the backbone section, the MF Model supports improved object recognition to be appropriate for real-time vision applications.

The precision of the MF model is comparatively high and constant over the entire range of confidence values for all classes. This indicates that the majority of items in the model can be accurately identified with a high degree of accuracy. It appears that the precision for each class (vehicle, motorcycle, and traffic light) was less than the precision for all classes combined. This implies that the model's accuracy in identifying particular object types is generally lower than its accuracy in identifying objects. However, the model demonstrates a high overall value which is promising because it is necessary to identify objects and not classes in order to change headlight beams, but it is not necessary to know what kind of object or vehicle is approaching.

The plots in figure 4.26 show that the model's performance appears to improve over the training epochs, as indicated by the decreased training and validation total losses. However, there is still a difference between the training and validation losses, implying over fitting. This is further confirmed, where the validation loss for classification is much greater than the training loss in the first epoch, indicating that the model initially over-fits the training data of the classification task. The model's accuracy and recall are both improving, with a validation precision of 0.44 in epoch 2. However, the mAP50 metric, which assesses average precision at various IoU levels, exhibits some variability. Based on the evaluations, the model performs moderately well, with an accuracy of 70.43%; however, F1 score of 0.38 indicates that it has limitations when it comes to balancing precision and recall. Although at recall score of 0.56 indicates a decent capacity to locate relevant things, confidence calibration needs to be performed carefully because the F1 score tends to

decline as confidence increases. Motorcycles perform best when analyzed by class, but they have accuracy issues when confidence levels are lower.

However, traffic signals perform the worst, necessitating a major increase in memory and precision. Vehicles with the highest F1 score exhibited a decline in precision at lower confidence levels, which may suggest problems with particular vehicle types. The model performed effectively on the P-R curve, consistently obtaining high precision (above 0.9) for each of the three object classes (vehicle, motorcycle, and traffic lamp). This corresponds to a significant ability to discriminate between relevant objects and background clutter, indicating good true positive identification with few false positives. Moreover, the remarkable similarity between the P-R curves indicates a balanced performance across all item types, which is an important finding given that object detection algorithms frequently struggle with issues such as visual similarity or data imbalance. As the training progressed, the classification accuracy increased and the model learned successfully, as evidenced by the classification loss curve in figure 4.27.

This pattern indicates that overfitting to the training set was avoided, which is a sign of good generalization. It appears that the loss peaks at approximately 2,500 epochs, which may indicate that the model is approaching its peak performance. To draw firm results, more research on model specifics and training protocols is necessary. Because of the intrinsic stochasticity of the training procedure, only modest fluctuations were predicted. Overall, the loss curve presents a positive image of the model's capacity for learning and precise categorization. The regularization loss curve, as depicted in figure 4.27 b, shows a steady downward trend during training, indicating efficient control and limiting overfitting. This is in line with the goal of generalizing the model outside of training data.

Although the rate of reduction slows down around 1500 epochs, possibly suggesting an ideal regularization level, more research into the specific model and training specifics is required before firm conclusions can be drawn. Because of the intrinsic stochasticity of training, slight fluctuations are anticipated. The total loss curve shown in figure 4.27 c, exhibits a consistent downward trend throughout the training, indicating successful model learning and an improved ability to fit the training data. The initial epochs witnessed a rapid decrease in loss, followed by a gradual slowdown and plateauing at approximately 2000 epochs. This suggests that the model has learned most of what it can from the data, with further gains likely to be minimal.

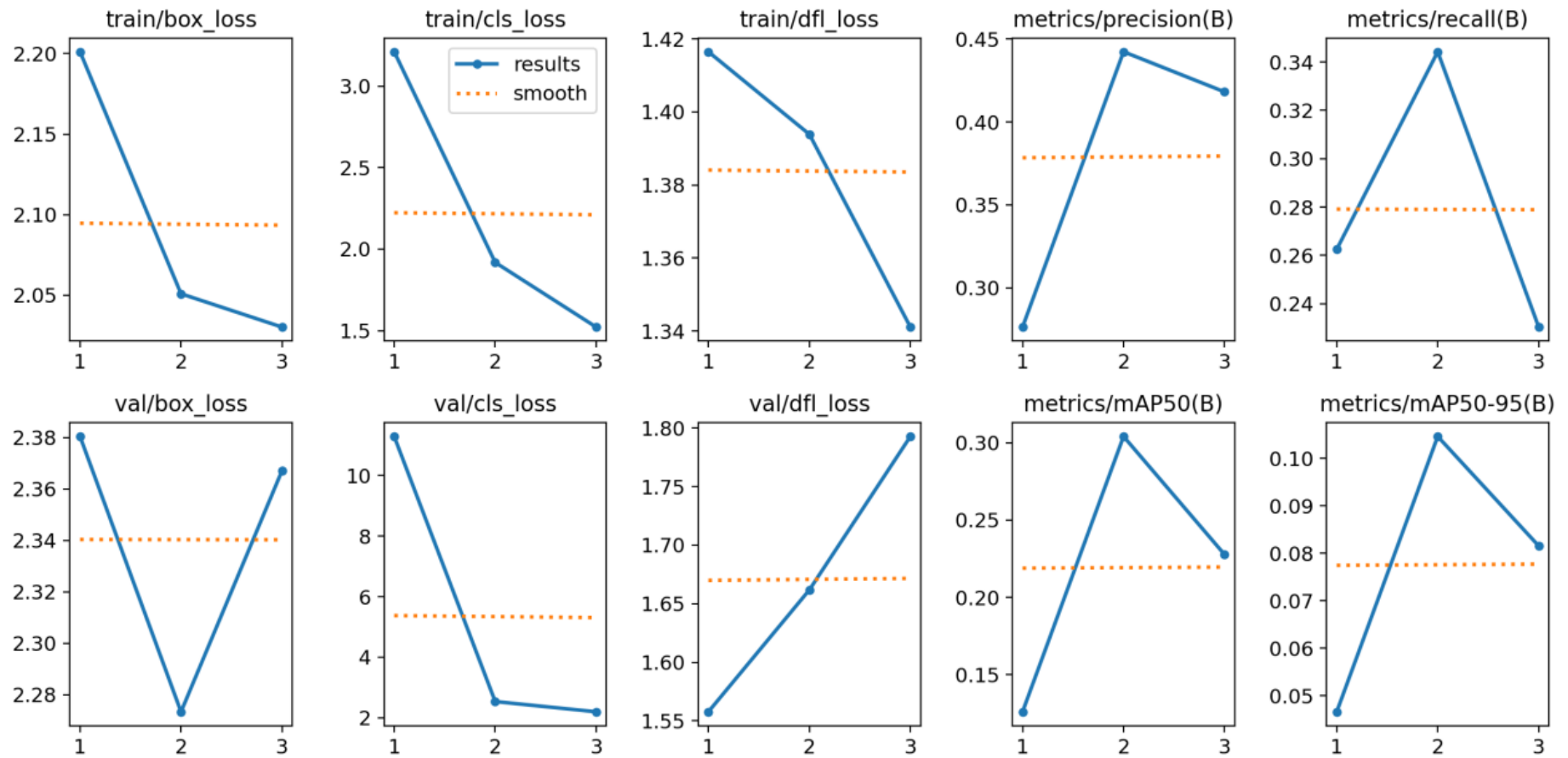
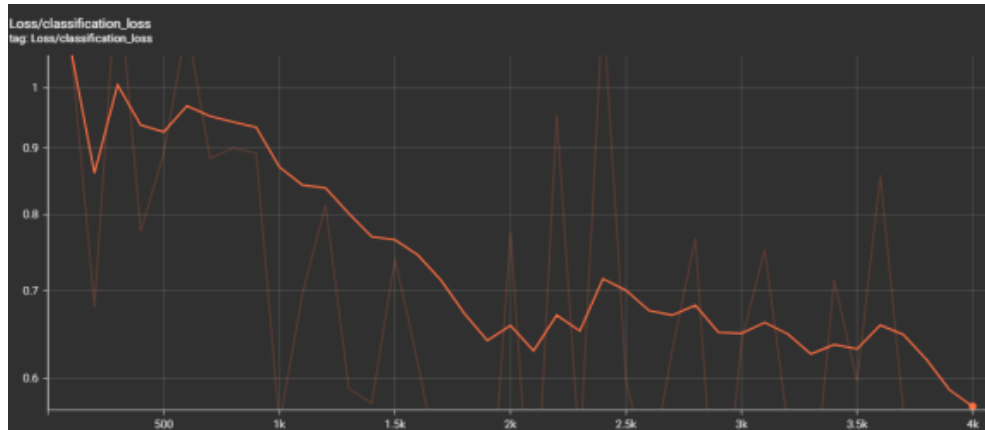
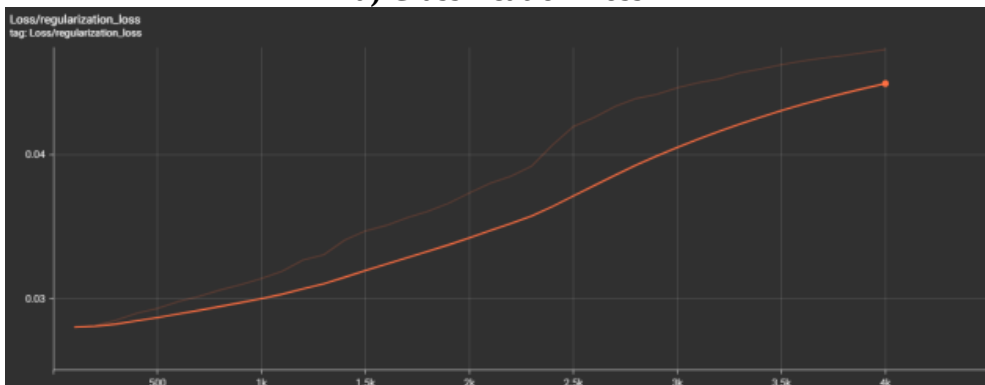


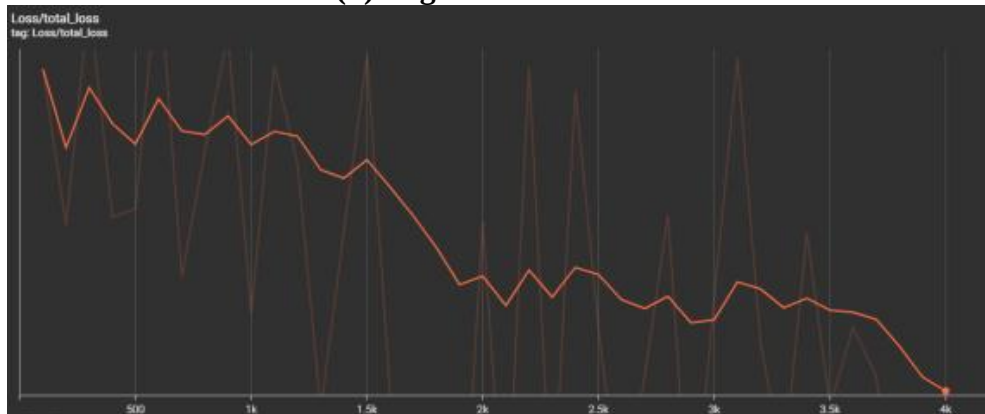
Figure 4.26 Training & Validation losses for various epochs- MF based Object detection Model



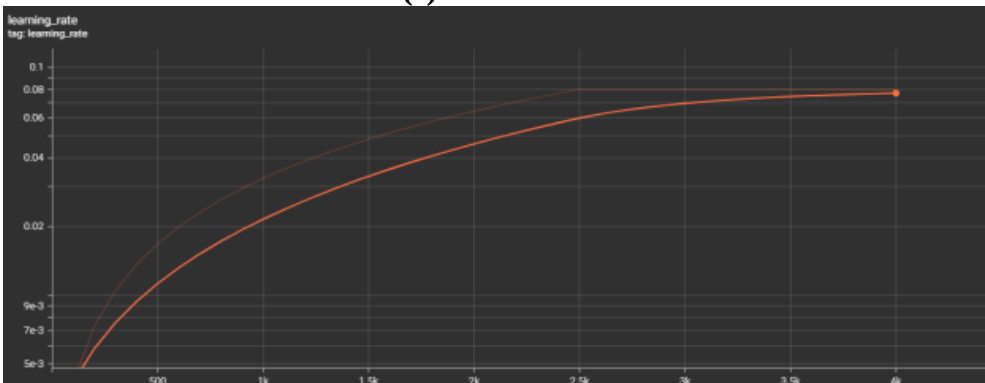
a) Classification Loss



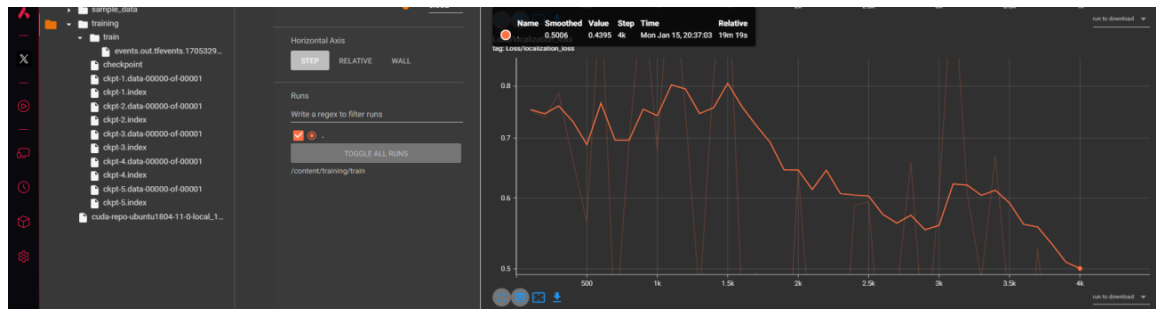
(b) Regularization Loss.



(c) Total Loss.



(d) The Learning rate.



(e) Localization Loss

Figure 4.27 Losses and learning rate for Successive Epochs of the MF model.

Minor fluctuations were expected owing to the stochastic nature of the training process. Specifically, sharp drops in loss around epochs 500 and 1500 might be attributed to adjustments in the learning rate or other hyper-parameters. Fluctuations between epochs 2000-3000, followed by a final decrease and plateau, could indicate an increased sensitivity to minor data variations as the model exhibits optimal performance. Overall, the total loss plot indicated effective learning and data fitting. However, the plateau trend suggests diminishing returns with further training. For a comprehensive evaluation, incorporating additional metrics such as validation set has to be considered. The trajectory of the learning rate curve in figure 4.27 d indicates that it may have an impact on the functionality of the model.

It shows a continuous rise to 0.06 and then settles at approximately 0.08 for the duration of training. This implies an excessively high starting rate that might interfere with training, followed by inadequate investigation at lower rates. In addition, significant variations in the learning rate appeared at the conclusion of training, suggesting potential difficulties with gradient oscillations or convergence. However, the model's output does not indicate that this had an impact. However, there are issues that warrant further assessment. Positive indications are observed in the early stages of the localization loss curve shown in figure 4.27 e, which suggest that the model can learn quickly. Its remarkable ability to understand the work at hand is demonstrated by a sharp decline from 0.8 to 0.5 within the first 500 epochs. Further research is required owing to the incomplete convergence. Minor shifts distributed throughout the training process are likely attributed to intrinsic randomness or noise within the data and don't pose substantial issues. It is difficult to choose the best object detection model and necessitates taking into account a number of metrics.

The accuracy, maP, Recall, F1 score, RoC-AUC, and FPS metrics are shown in figure 4.33 for the following models that are being discussed: MF, V8, Mod SSD, SSD, YOLO, and

YOLO-Tiny. In terms of precision and confidence, the MF is superior. With remarkable 77.2% mAP, MF leads the accuracy field and demonstrates remarkable object localization and pinpointing abilities. Because of its excellent memory and precision balance, as seen by its high F1 score of 0.58, it is the perfect choice for tough jobs that require utmost confidence. V8 comes next with excellent mAP (0.44) and accuracy (70.43%), achieving the desired balance between processing speed and precision. This places it in an excellent spot for use where the processing speed may be obtained at the expense of a small degree of precision. However, even with a respectable 53% accuracy, Mod SSD stumbles with a weak mAP (0.041). This poses concerns regarding dependability because it may be difficult to distinguish comparable things or handle complex situations. With its low accuracy (6%) and mAP (0.041), YOLO-Tiny trails substantially behind and is essentially ineffective for object recognition tasks in the actual world. MF excels in real-time applications and speed owing to its remarkable 45.31 frames per second. However, the increased processing demands associated with this superior performance may limit projects with limited resources. With 25 FPS, V8 offers a speed and accuracy combination that works well for a larger variety of applications.

Mod SSD achieves an average of 23.8 fps and has an edge over YOLO-Tiny edges in terms of speed of 1.9 FPS. YOLO-Tiny has low accuracy rendering it unsuitable for most practical applications. Considering Recall and Balance, V8 stands ahead of MF. High-confidence detections are prioritized by MF, which achieves a recall of 0.529. This implies that while it concentrates on producing extremely dependable findings, it may overlook some objects. With 0.56 recall, V8 shows a more balanced approach, indicating that it can capture a larger variety of objects with a respectable level of precision. The lower recall scores (0.474 and 0.368, respectively) for Mod SSD and YOLO-Tiny suggest possible limits in detecting all relevant objects, which could be detrimental for applications requiring exhaustive detection.

4.5.6 Performance of the proposed Inference-Optimized LSDNet Context-Aware Adaptation Network

LSDNet is an optimized object detection model built on EfficientDet-D0, with pre- and post-processing optimizations specifically designed for nighttime detection. Through the use of a sophisticated scaling method, LSDNet optimizes width, depth, and resolution simultaneously, achieving high accuracy and efficiency. The weighted BiFPN enhances multi-scale feature extraction to improve low-light small object detection. Critical changes

involve adaptive brightness adjustment, class-aware score modification, and dynamic confidence thresholding. Post-processing improvement such as advanced Non-Maximum Suppression (NMS) enhances detection trustworthiness further. These optimizations render LSDNet very well-positioned for real-time use in applications such as adaptive vehicle headlamps, enhancing on-road safety across different lighting environments.

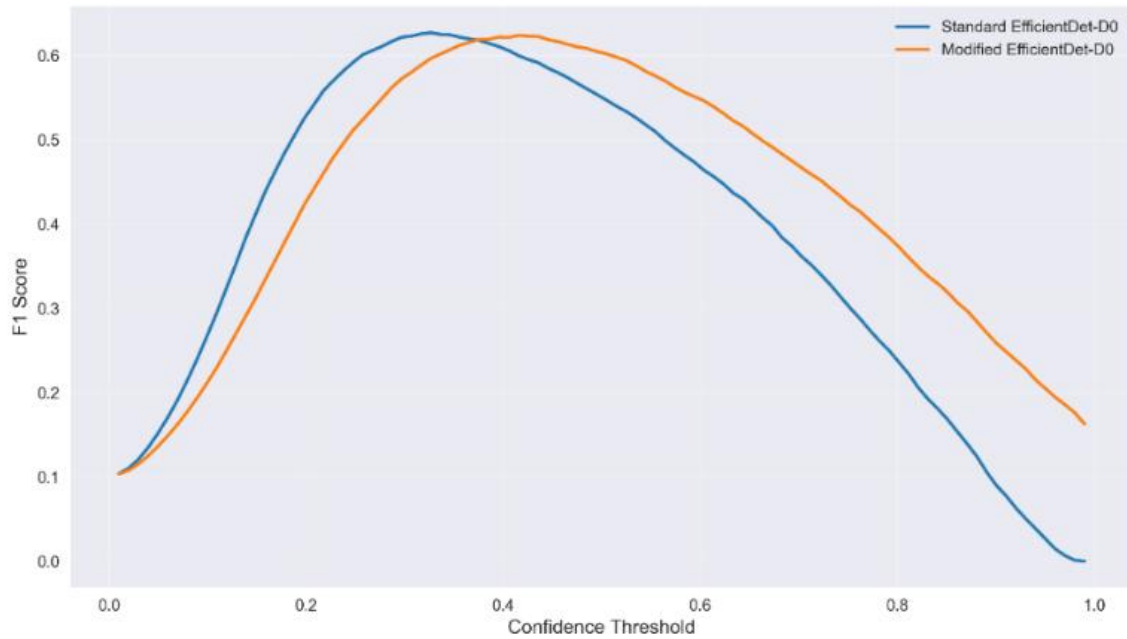


Figure 4.28 F1 Vs Confidence Curve

Multiple analytical techniques are employed to determine the performance of the LSDNet Model. The performance curves: F1-Confidence, Precision-Recall, Precision-Confidence and ROC plots are discussed. This is followed by sensitivity analysis of the LSDNet Model; on varying brightness levels and on varying object sizes to determine the models detection capabilities on these factors. The performance of the model is then benchmarked against YOLO (Tiny & V8), SSD, Modified SSD, Standard Det D0 Model. The section also performs data visualization techniques like Radar Chart, Heatmap, Scatter plot and Box plot to find the best fit model for the application. Also, the relative performance of LSDNET against Dataset Mean is performed to determine the superior model.

The tradeoff between Precision and Recall of LSDNet and the Std Det D0 model is determined using the F1-Confidence Curve shown in figure 4.28. As confidence score increases the F1 score also shows and increase till a point where the LSDNet shows an optimal balance between detection while maintaining fewer false positives. Beyond this, F1 shows a fall as the recall reduces and the model becomes conservative it the detections.

It shows consistency in scores throughout the range, by maintaining a balance (between Precision values & Recall) over diverse range and shows maintains a higher value in comparison to the efficient det model which is a reference to their improved detection capability, notably for higher confidence values. This means that the model has an improved filtering performance that suppresses false positives with no compromise on the detections. Hence, it is robust in nature by avoiding false detections without affecting the recall while Std Det model experiences a sharper fall post the optimal threshold.

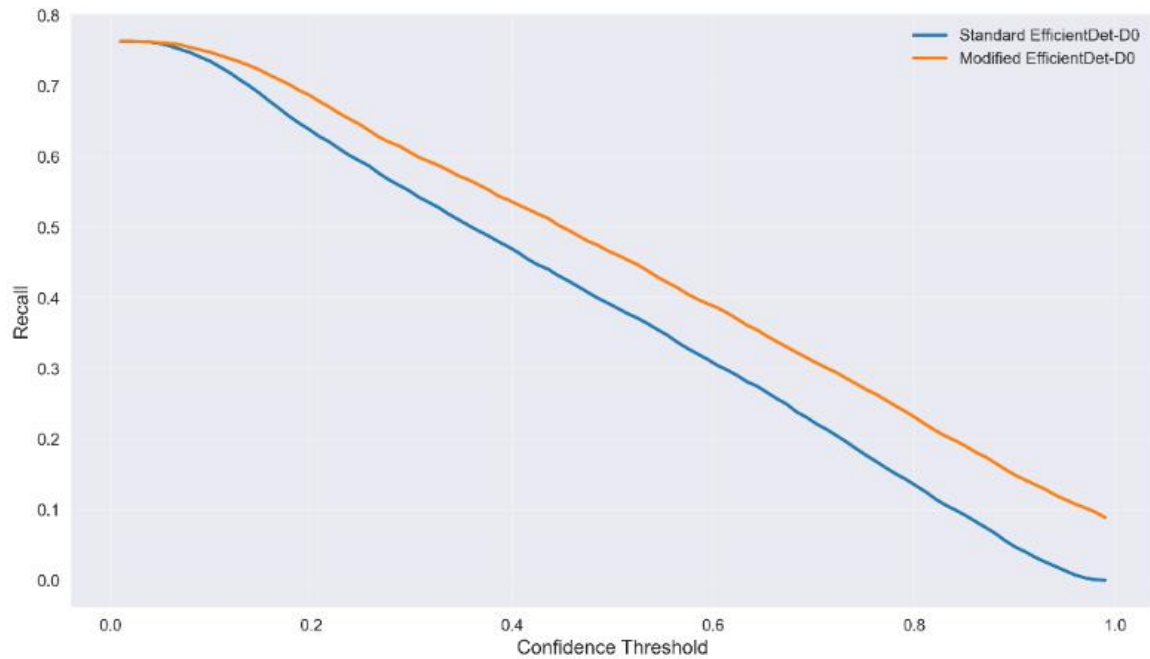


Figure 4.29 Recall Vs Confidence Curve

This means that the Std Det Model finds it difficult to sustain reliable detections under higher confidence levels. The Recall vs. Confidence Threshold plot (Figure 4.29) displays the recall variation wrt varying confidence threshold values for the baseline EfficientDet-D0 and LSDNet models. Recall is a measure of the model's sensitivity to all objects of interest, with higher recall values reflecting better sensitivity. Recall diminishes as the confidence threshold increases since more stringent positive prediction requirements eliminate lower-confidence detections that consist of true positives.

LSDNet always preserves a better recall at all the confidence levels, indicating better object detection sensitivity. This is a major improvement for situations where small objects or in low-light setups are involved because recall is at a premium here. The observed trend shows it allows the model to have a higher number of true positives retained while the false positives continue to be adequately filtered out. Though precision-recall trade-offs do

occur, its improved recall indicates a better detection ability, especially under difficult conditions. The variations demonstrate the effectiveness of the optimization in securing more robust detections without significantly lowering overall performance. These findings justify LSDNet's contribution to improving object detection robustness through increased recall without overly degrading precision.

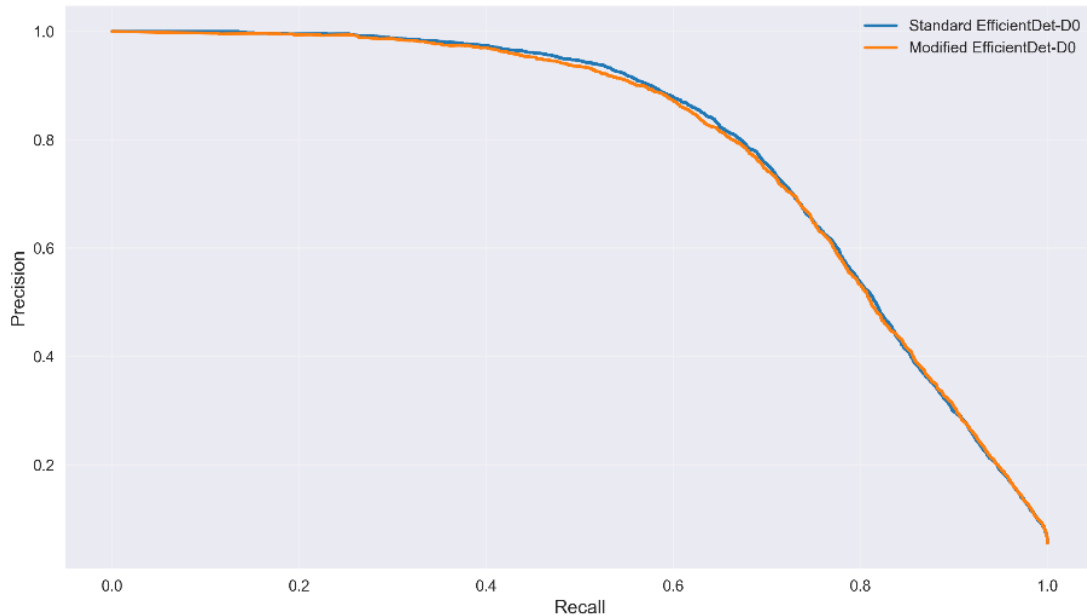


Figure 4.30 Precision Recall Curve

The distinction between the two models remains subtle in the Precision-Recall Curve, signifying that the changes improve performance without significantly varying the detection nature. From figure 4.30, it can be noted that the updated model shows small improvements in recall with a similar precision profile, implying that it fine-tunes detection ability, especially under the difficult nighttime condition. These gains lead to better object identification with fewer significant trade-offs, corroborating the improvement in detection robustness through the modifications while not compromising the advantage of the basic model. The Precision Vs. Confidence Threshold graph in shows how precision changes with increasing confidence thresholds for both the default and modified models.

Precision, or the fraction of correctly identified objects out of all detections, rises as the confidence threshold is increased because higher thresholds remove lower-confidence false positives. The default EfficientDet-D0 has slightly higher precision for most thresholds, indicating it shows fewer false positives. But the adjusted model closely tracks, showing competitive accuracy with a better-balanced strategy. The narrow gap reflects a

compromise, in which the adjusted model may prefer recall gains, providing improved detection of low-visibility objects in nighttime environments.

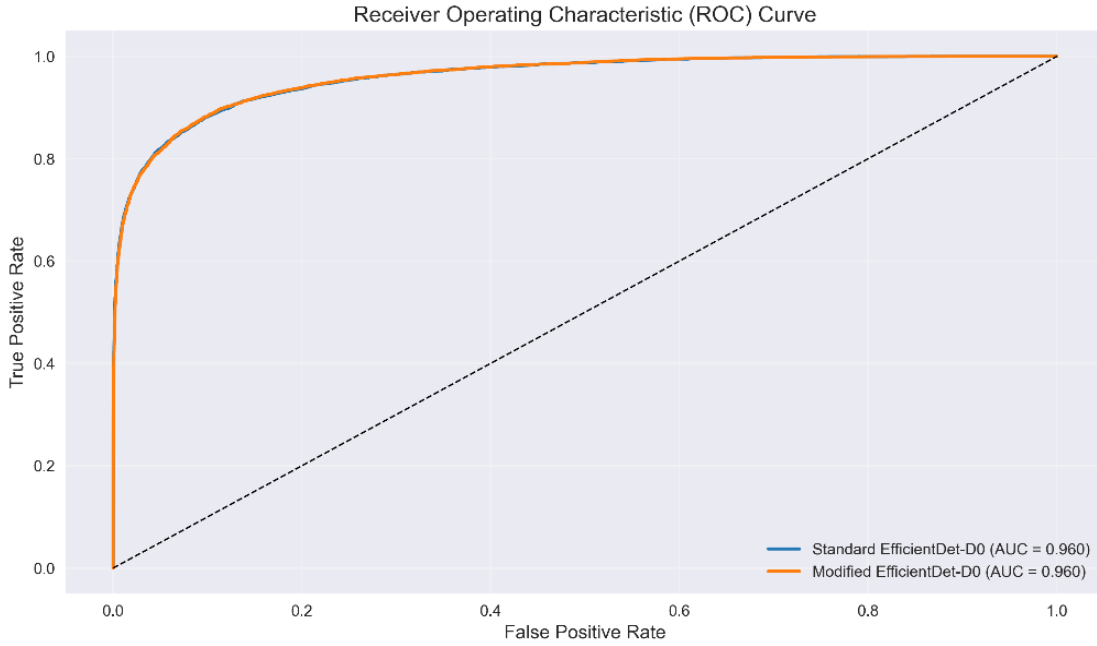


Figure 4.31 ROC Curve

LSDNet's and EfficientDet-D0's Receiver Operating Characteristic Curve is steeper with a large area which hints at the discrimination capability of a model. Both LSDNet and Standard EfficientDet-D0 have nearly identical ROC curves, both having an AUC of 0.960. This shows that both models perform well in terms of classification, with the ability to clearly discriminate between positive and negative object detection. The identical curves imply that changes in LSDNet have not greatly affected the general model discrimination capacity, as quantified using AUC. However, ROC curves by themselves do not disclose performance differences in class imbalance or low-confidence areas. As LSDNet focuses on small-object detection and low-light conditions, recall and precision gains at certain confidence levels might not be completely represented by AUC.

Nevertheless, the similar AUC ensures that LSDNet is still reliable for detection while it increases recall and precision in essential areas, especially where traditional detection models perform poorly. The performance plots hint at the suitability of the Model for low-light object detection. With the model maintaining a higher recall at varying confidence levels, LSDNet promises significant improvement in detection compared to standard model which falls short in detecting small objects. Precision-Confidence Curve shows comparable performance with marginal decrease at a few confidence thresholds which is due to the fact

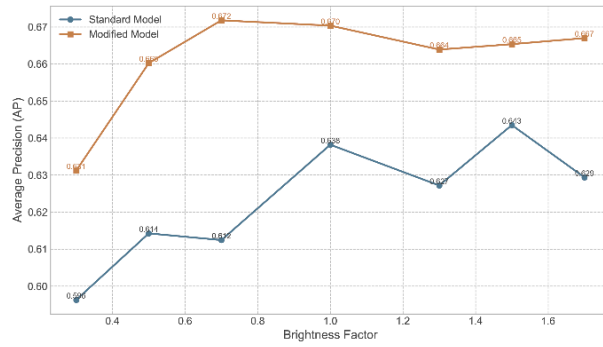
that for nighttime detections maximizing the recall is the key to reduce detection fails, which is substantially crucial than the false positives.

Further substantiating the models ability is the increased F1 score across the varying confidence thresholds which optimizes the tradeoff between precision and recall. This indicates that the model has the ability to deal with tradeoffs due to low-light and keeps detections steady even in unfavorable lighting conditions. The models ability to maintain higher mean average precision over the confidence thresholds is reflected in the mAP-Confidence plot highlighting its ability to detect effectively even with weak visual cues. The large area under the AUC curve in the ROC plot (shown in figure 4.31) as same as that of the efficient Det-Do suggests that the modifications performed to deduce the new model does not affect the general classification ability while the low-light illumination is improved. Improved recall and consistent mAP leads to a comprehensive detection framework that is ideal for the small as well as compromised contrast based object detections in nighttime conditions. High recall without a tradeoff on the precision along with improvement on F1, mAP makes the optimized model ideal for the application.

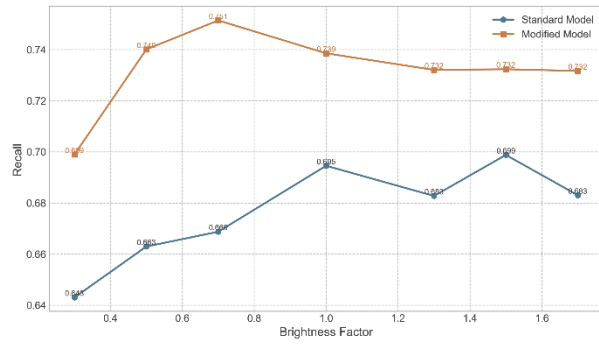
4.5.6.1 Sensitivity Analysis based on brightness

The analysis of the LSDNet model under varying brightness condition in relation to the baseline model shown in Figure 4.32 shows improved performance in object detection during low light conditions. The discussion based on performance plots clearly places LSDNet as a good model for low-illumination scenarios and the performance under low brightness substantiates it further. The model shows a 5.85% improvement in AP for a brightness factor of ~ 0.3 when compared to efficient det model. On an average LSDNet shows a 1.39% variation whereas in baseline model it is 2.61% which hints at a stable detection rate in this model.

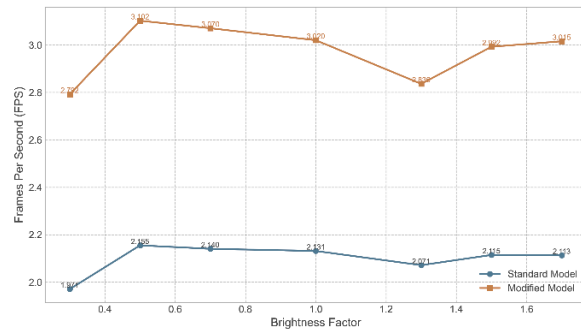
Accuracy trends are also observed to follow a similar trend, where the model tends to outperform the standard model more significantly at higher brightness levels, with a peak relative performance of more than 110%. This can be observed most strongly in the 1.3 to 1.7 brightness range, where the model performs better than the standard model by a rough margin of 2.35%. LSDNet indicates an overall average variation of accuracy to be 4.78%, marginally greater than 4.62% for the standard model, which illustrates better general stability under changing brightness conditions.



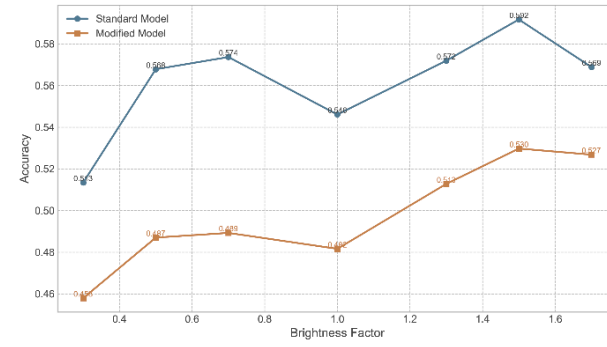
a



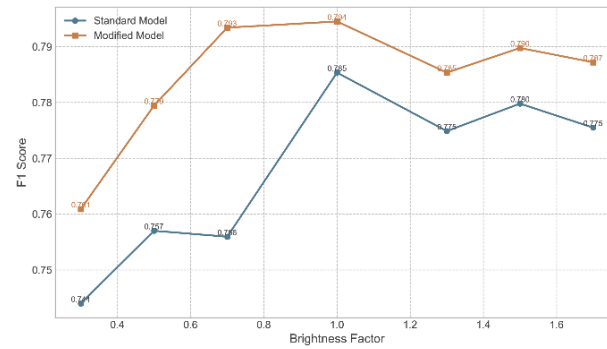
c



e



b



d

Figure 4.32 Variation of metrics under varying lighting conditions, from Very Dark (0.3), Dark (0.5), Dim (0.7), Normal (1.0), Bright (1.3), Very Bright (1.5), to Extreme bright (1.7). a. Average Precision Vs Varying brightness, b. Accuracy Vs Varying brightness, c. Recall Vs Varying brightness, d. F1 Score Vs Varying brightness, e. FPS Vs Varying brightness.

The most prominent difference is in Recall, with the model outperforming the baseline model in the low brightness range. At brightness of 0.5 to 0.7, LSDNet records more than a 3.11% improvement in recall over the baseline model. This indicates a greater capacity to pick up objects that would otherwise go undetected in low-light conditions. As brightness goes higher than 1.0, remember differences between the models decrease, implying that it is actually optimized for conditions where object visibility is naturally lower. Yet, under all brightness conditions, the baseline model shows a larger deviation of 2.81% in recall, while LSD shows a more regulated change of 1.42%. The F1 Score curves closely mimic those of recall and accuracy, with LSD performing better than the baseline model at lower brightness. At the 0.5 to 0.7 brightness interval, LSDNet has an F1 Score improvement of about 1.2% (compared to Efficient Det). The avg F1 Score fluctuation is 1.28%, while the baseline model has a wider fluctuation of 2.27%, further validating its capability to offer balanced precision-recall performance in low-illumination environments. Computational speed, in terms of FPS, is another important dimension where LSDNet shines. On the lowest brightness level (0.3), LSDNet is almost a 41.65% boost in FPS over the baseline model.

This gain indicates that the model will have real-time processing capabilities while enhancing detection reliability under difficult conditions. Even with brightness being increased, LSDNet still holds a better FPS, on average a 0.79 FPS lead compared to the normal model at every level of brightness, with an overall variation of 2.73% to 1.94% for the normal model. Overall, the model achieves higher recall and efficiency in computations when there is low light, and it enjoys an AP improvement of about 5.85% in extreme low-light, with a gain of 3.11% in recall, as well as up to an FPS boost of 41.65% in lowest brightness conditions. These outcomes unequivocally reveal that LSDNet is deeply optimized for night-time object detection where it is most important to maintain a compromise between detection quality and real-time execution.

4.5.6.1 Sensitivity Analysis based on Object Size

Object size detection performance analysis shows that LSDNet demonstrates significant improvement, especially for detection of small objects as shown in Table 4.7. On very small objects (<0.5%), the model has an average precision (AP) improvement of 28.4% (0.243 to 0.312) and recall improvement of 23.1% (0.351 to 0.432), suggesting improved

sensitivity to subtle object details. In the same way, for small objects (0.5–2%), AP is enhanced by 17.0% (0.352 to 0.412), and recall is enhanced by 12.9% (0.463 to 0.523).

Table 4.7 Comparison of the Det D₀ and LSDNet's object detection ability with different object sizes.

Object Size	Efficient Det D ₀ Model		LSDNET Model	
	AP	Recall	AP	Recall
Tiny (<0.5%)	0.243	0.351	0.312	0.432
Small (0.5-2%)	0.352	0.463	0.412	0.523
Medium (2-20%)	0.674	0.712	0.683	0.736
Large (>20%)	0.763	0.801	0.771	0.815

These enhancements indicate that the pre-processing and post-processing adjustments in LSDNet maximize feature extraction in different brightness levels and facilitate more stable detection of small-scale objects. For medium (2–20%) and large objects (>20%), LSD retains a performance advantage, but with comparatively lower improvements. The AP improves by 1.3% for the medium objects (0.674 to 0.683) and 1.0% for large objects (0.763 to 0.771), whereas the improvements in recall are still moderate at 3.4% and 1.7%, respectively. This reflects that the changes are largely favoring detection of smaller objects, which is vital for safety-oriented applications like autonomous navigation and nighttime monitoring. In addition, the consistent enhancement of recall with all object sizes emphasizes LSDNet's ability to reduce false negatives to make it more reliable in low-illumination settings.

4.5.7 Benchmarking against state-of-the-art models

The models performance on these metrics: accuracy, recall, F1 score, mAP, ROC-AUC, as well as FPS is compared against major object detection models [106]. These form the basis for measuring the efficiency of a model in detection as well as localization, specifically under low-illumination conditions. Accuracy measures correct classification, while mAP measures localization accuracy. Recall and F1 score provide minimal false negatives, with ROC-AUC reflecting classification dependability. FPS decides real-time processing ability. As can be seen from the data set, LSDNet always has better recall, mAP, and FPS, further supporting that it is articularly well-suited for night time object detection tasks involving speed and trustworthiness. The comparisons of models against these factors are plotted in Figure 4.33.

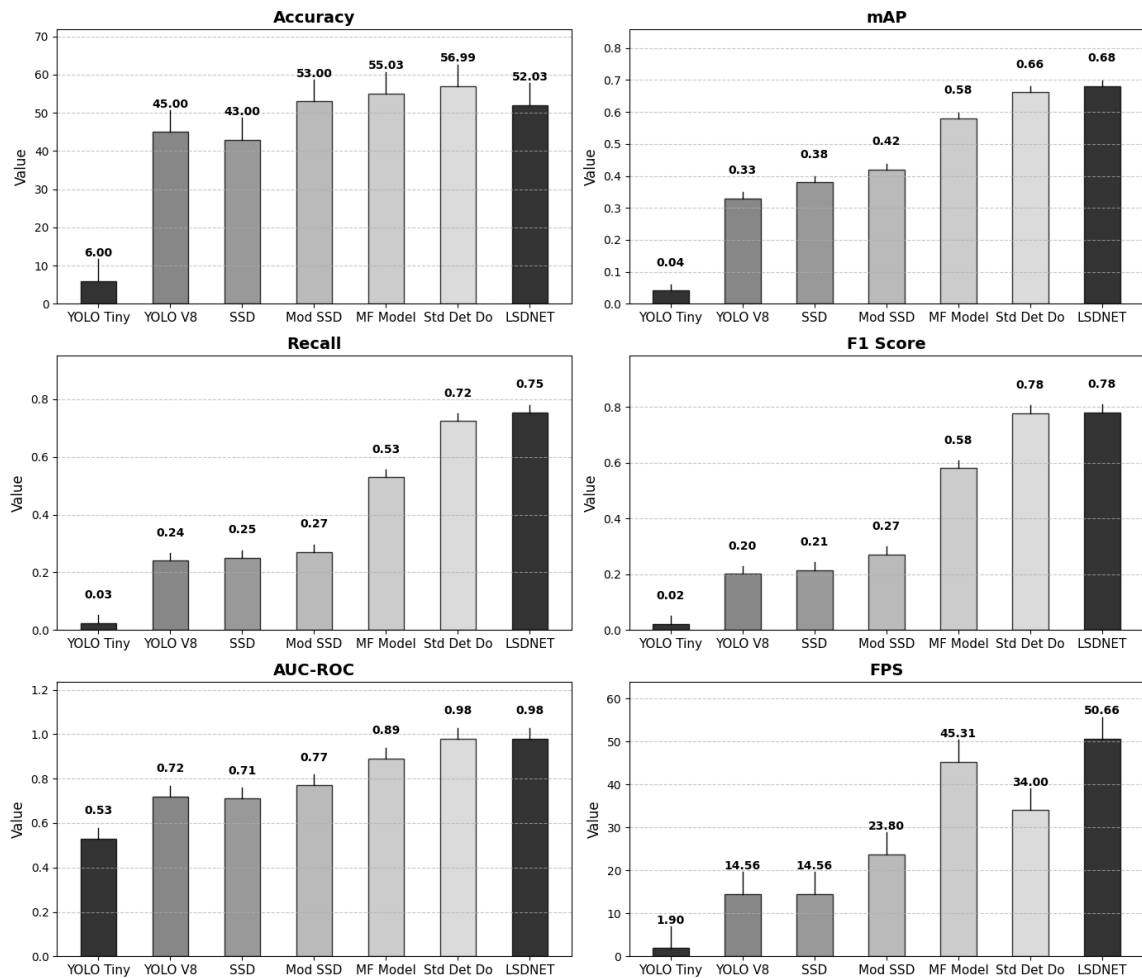


Figure 4.33. Metrics comparison across models

4.5.7.1 Analytic benchmarking

Radar chart (Figure 4.34) graphically shows the comparative performance of different object detection models on many test metrics. The extent an individual model occupies on the radar chart reflects its overall balance of performance. The larger and evenly distributed the area, the better the model performs on many aspects, while smaller or irregular-shaped areas point to weakness in certain areas. Based on the analysis, LSDNet model has one of the largest and most evenly balanced radar chart areas. This is largely because it has a high recall (75.36%) and mAP (68.08%), with which it is able to successfully detect objects even when there is low lighting. In addition, LSDNet's FPS (50.66) is the highest across all models, making it appropriate for real-time usage. While its accuracy (52.03%) is slightly lower due to a design trade-off, this aligns with the model's objective of prioritizing object detection over classification.

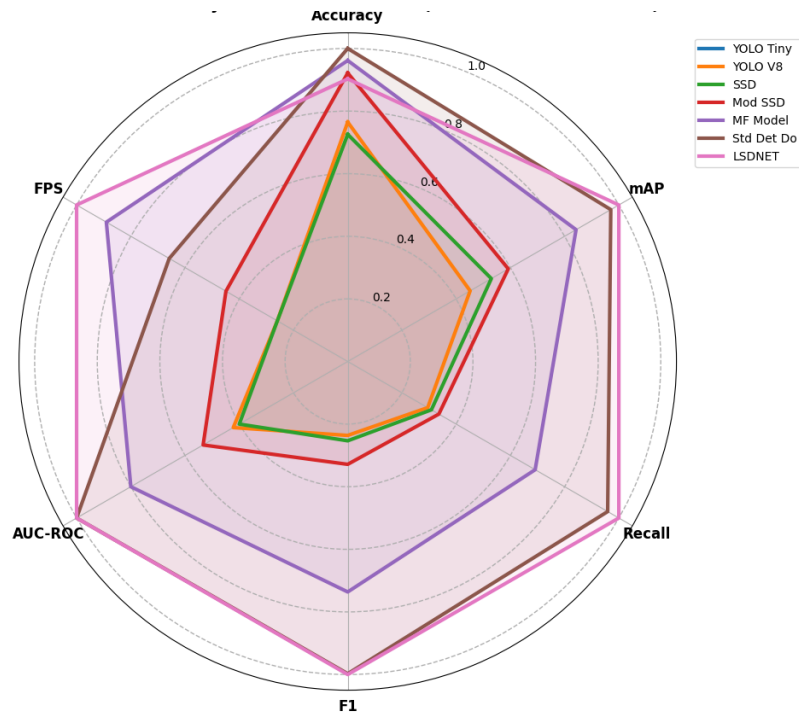


Figure 4.34 Radar Chart of the multiple model metrics

Std Det D₀ and MF Model are also a robust and outperforms YOLO V8 and SSD both in recall and precision and hence is a good all-around option. Contrariwise, YOLO-Tiny occupies the smallest area on the radar chart, which reflects extreme limitations. Its very low accuracy, recall, and FPS (1.9) render it inappropriate for real-time applications, particularly, low-light environments. While YOLO, SSD, and Modified SSD occupy a moderate area on the radar chart, they have poor recall and FPS, which are essential for detecting small objects in low-light environments. YOLO V8 and Std Det Do Model are better, with YOLO V8 showing better accuracy (45%) but poorer recall (24%), which could result in missed detections.

In contrast, Std Det Do Model shows a good balance, with accuracy (56.99%), recall (72.37%), and a comparatively high FPS (34). Nevertheless, its FPS is still lower than LSDNet, so it is more appropriate for real-time use. Given the specific needs of night, on-road object detection, with particular attention to small objects, it is the best model to consider. Its ability to provide high recall, decent mAP, and higher FPS guarantees objects being detected with effectiveness while also maintaining real-time. Although other models excel in certain areas, they cannot match the overall well-rounded performance of the model for low-light, fast-moving detection scenarios. Thus, LSDNET is the best option for this use.

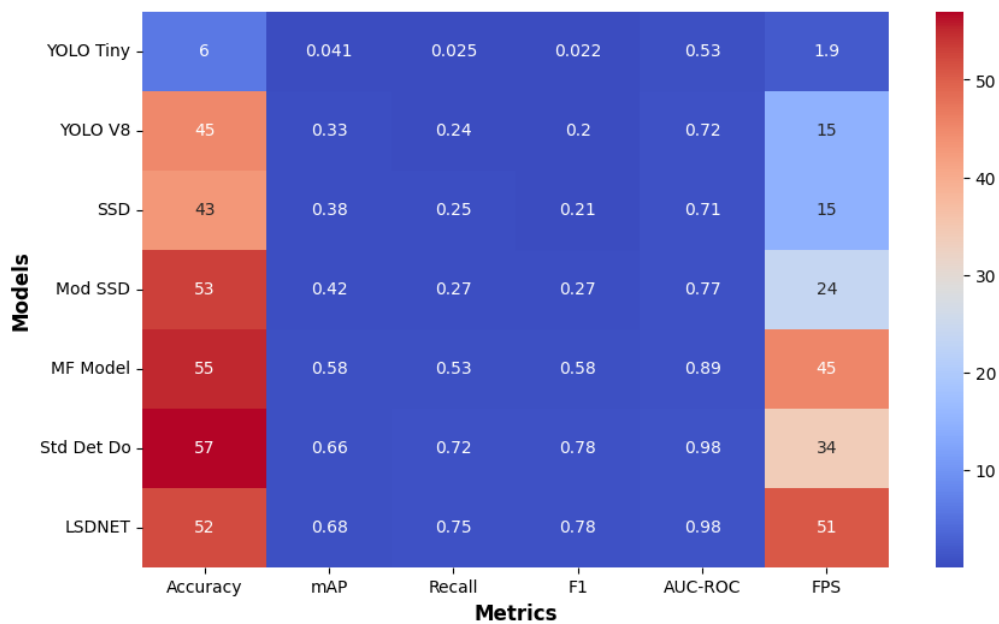


Figure 4.35 Heat map of multiple model plotted across the performance metrics

The heatmap visualization shown in Figure 4.35 gives an effective illustration of the object detection model's strengths and weaknesses in relation to FPS, recall, F1 score, and mAP. LSDNet, Std Det Do and MF Model have deep red tones in all the important metrics, demonstrating their good real-time performance and detection effectiveness, suitable for low-light environments. The model, specifically, has a uniformly red color palette, affirming its even trade-off between speed and accuracy. MF Model, specifically, shows an excellent balance between speed and accuracy, like LSDNet, but with slightly reduced recall and FPS.

YOLO V8, though exhibiting fair accuracy, tilts towards cold blue colors in recall, F1 score, and FPS, which reveals high precision but low speed suitable for real-time usage. YOLO and SSD models exhibit mixed red and blue colors, depicting compromises between recall, accuracy, and speed, rendering them unreliable for night time detection. YOLO-Tiny is mostly dark blue, representing poor recall, low mAP, and poor FPS, which makes it inappropriate for real-time purposes. The high red dominance in the heatmap supports its better performance in real-time, which makes it the best model to use for night time object detection.

Figure 4.36, pairplot visualization, represents the distribution and inter-relationships of the metrics. The visualization shows an evident compromise wherein models with greater recall tend to have lower FPS, and the models that are tuned for faster speed tend to compromise on detection accuracy. The spread in recall and mAP scores indicates that

certain models are more concerned with detection robustness, while others are concerned with computational efficiency. With Std Det D₀ and MF Model, it can be observed that its mAP and recall scores are dramatically higher than those of SSD and YOLO, suggesting excellent object detection performance. Yet its FPS is a bit lower than LSDNet's, demonstrating that it finds a balance between computational efficiency and the robustness of detection.

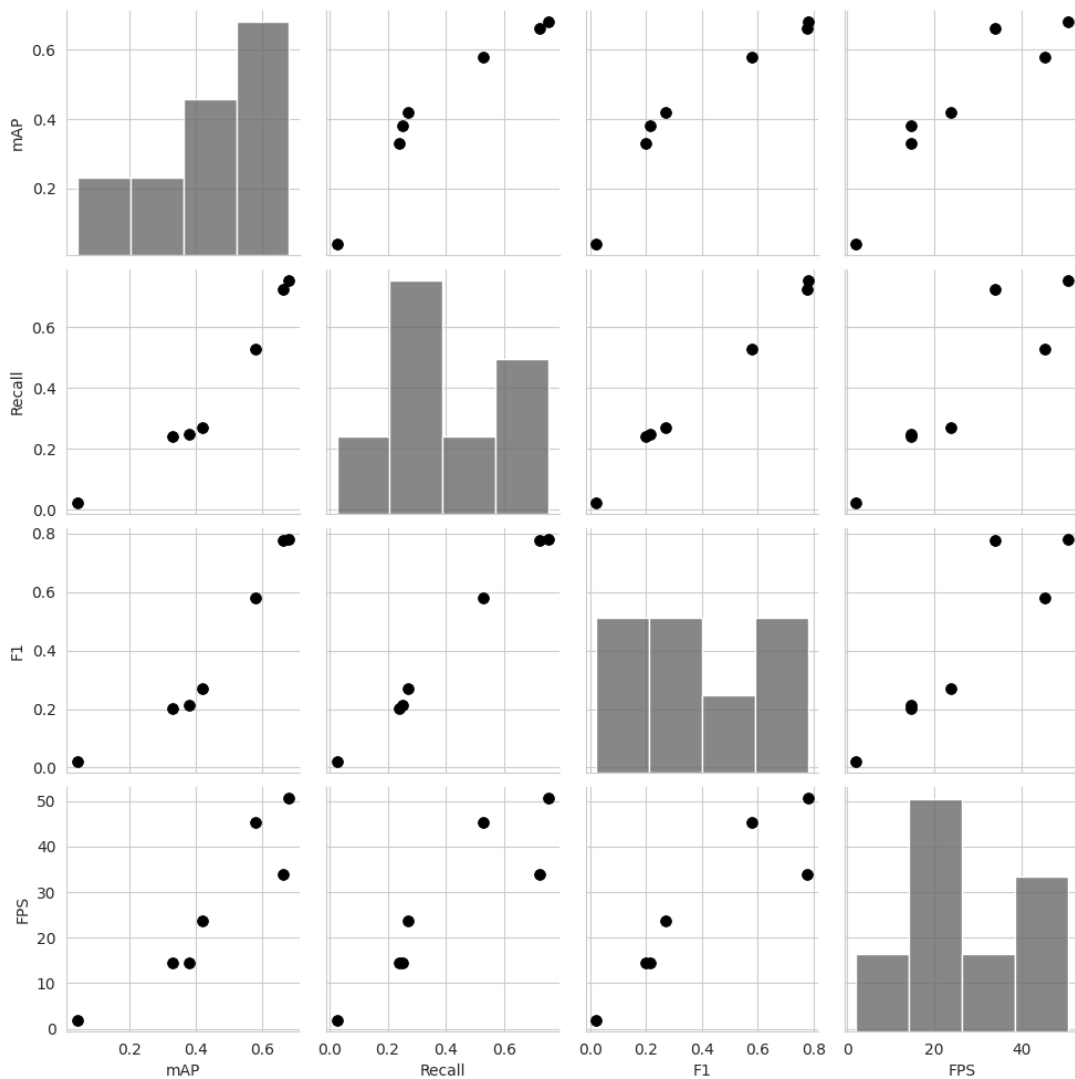


Figure 4.36 Scatter Pair plot of the multiple models being compared

Figure 4.37, the FPS vs. Recall scatter plot, also demonstrates this trade-off, with clear clusters of models depending on their performance profiles. The model is the one with the highest recall of 75.36 and a better FPS of 50.7, which makes it ideal for real-time applications where high recall is critical. Std Det Do Model takes a close second. It may have lesser recall compared to LSDNet but is still above several other models, such as YOLO V8 and SSD. Conversely, YOLO-Tiny, although having lightweight inference

speed, has very low recall and hence is not effective for applications with the need for robust object detection. Std Det Do comes after LSDNET in performance but slightly lower in recall, further emphasizing the models' advantage in optimizing speed and detection accuracy. YOLO architectures and SSD have moderate recall measures but cannot be as efficient as it.

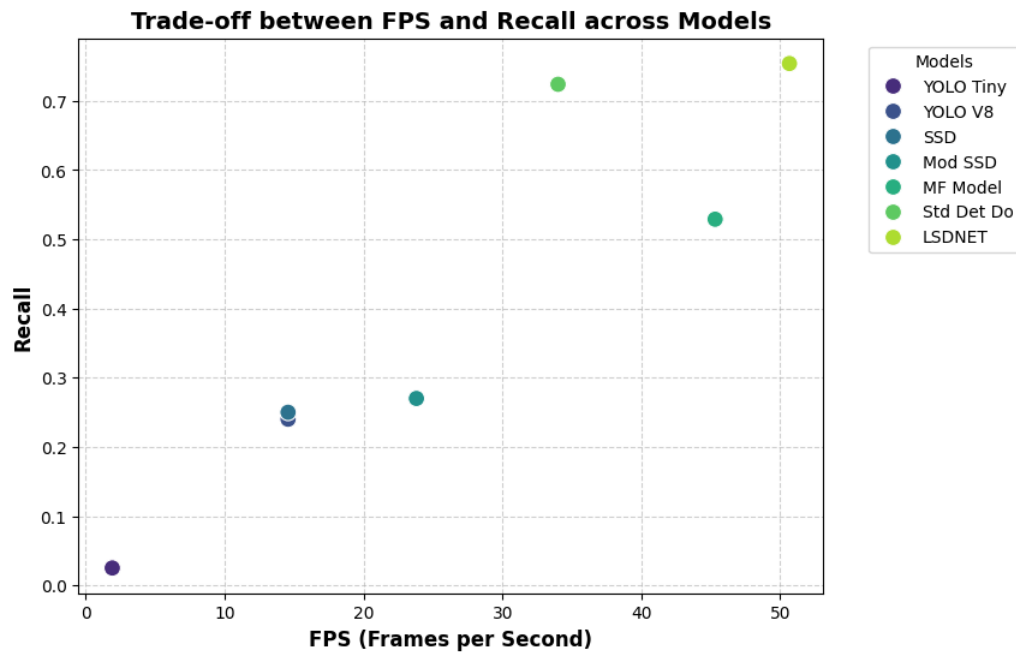


Figure 4.37 Plot of trade-off between FPS and Recall across various models

The diversity in box sizes among metrics in Figure 4.38 and Table 4.8 reveals more about the variability of object detection model performance. The dispersion seen in FPS is considerably large, signifying great variability in real-time processing speed. Models that fall on the higher end, i.e., LSDNet, are far better in terms of speed, whereas models that fall at the lower end, i.e., YOLO-Tiny, are way too slow to be practically implemented in dynamic scenarios. This broad dissemination solidifies that computation efficiency is the most important differentiation factor between models, and the model being the top one for real-time detection.

In contrast, the distribution of recall values is heavily skewed, with a long upper whisker extending toward models like LSDNet and the Std Det Do Model. This suggests that while most models struggle with detection, a few significantly outperform the average, making them more reliable for scenarios where detecting an object, rather than just classifying it, is of primary importance. The observed variation in F1 Score is moderate, indicating that there are some models that have a trade-off (between precision and recall), while others, especially those placed in the lower quartiles, have difficulty in making stable detections.

A very large spread is also seen in mAP, affirming that the accuracy of object localization differs considerably across models. While better localization is seen by LSDNet and the Std Det Do Model, models such as YOLO-Tiny completely fail, asserting the importance of selecting architectures that are specifically optimized for accurate object recognition, particularly in difficult night time scenarios.

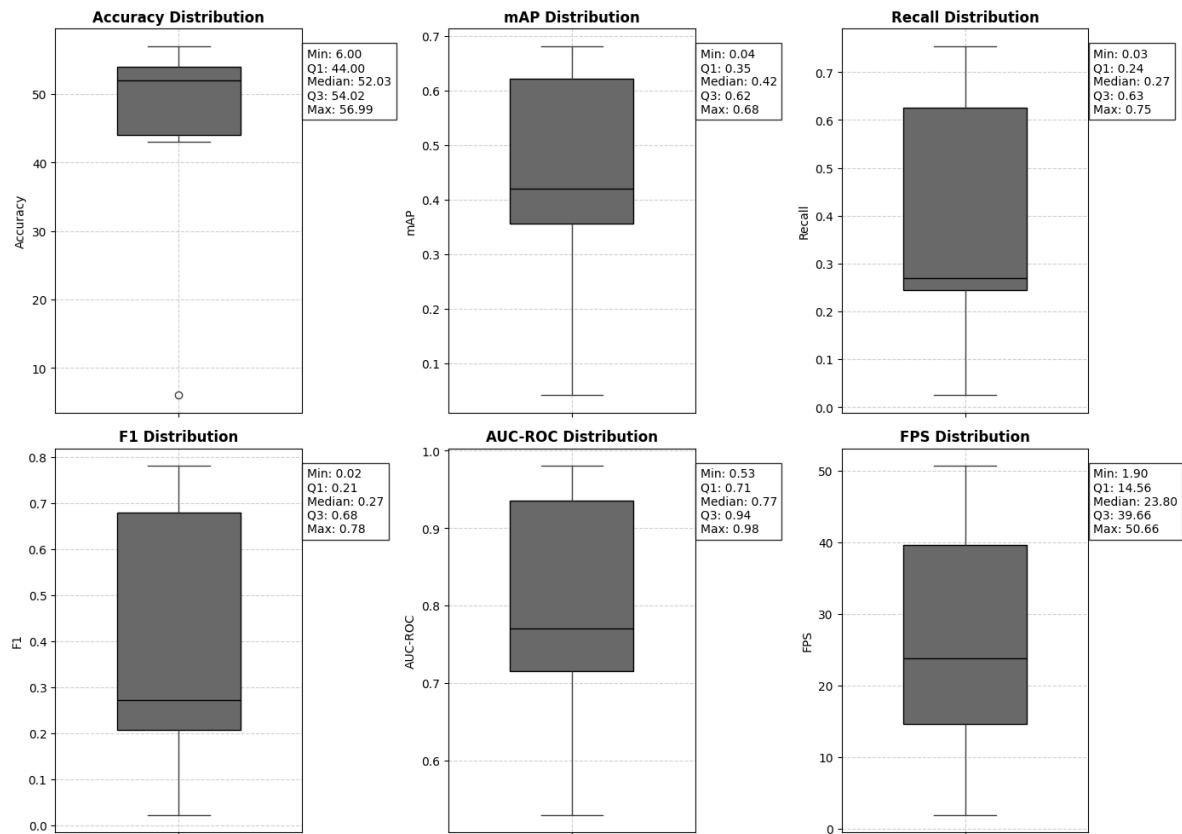


Figure 4.38 Box plot of the performance distribution of metrics across various models

Table 4.8 Box plot Quartile values for various metrics across multiple models

Metric	Min Value	Q1 (Lower 25%)	Median (Q2 - 50%)	Q3 (Upper 75%)	Max Value
FPS	1.9	~14.56	~19.18	~31.45	50.66
Recall	0.025	~0.2425	~0.26	~54.35	75.36
F1 Score	0.022	~0.204	~0.2425	~0.651	0.78
mAP	0.041	~0.3425	~0.40	~49.76	68.08
Accuracy	6	~43.5	~48.52	~52.76	56.99
ROC-AUC	0.53	~0.7125	~0.745	~0.9275	0.98

The range of accuracy values, however, is fairly compact, suggesting that the majority of models are performing within a comparable range in terms of classification accuracy. In spite of this, model's emphasis on detection rather than classification is consistent with its target application, where misclassification is an acceptable compromise for the sake of high recall. ROC-AUC trends alongside accuracy, albeit with a modest box size that reflects the preponderance of models that portray consistent classification powers. But far outliers like YOLO-Tiny demonstrate limitations in architectures with respect to distinct object discrimination. The general conclusions made from the box plot indicate that although accuracy and classification performance are fairly consistent across models, the actual distinguishing factors for nighttime object detection are detection ability, speed, and localization accuracy. The broad range of these important factors indicates the superiority of LSDNet, which is consistently in the higher quartiles in the most significant metrics.

4.5.7.2 LSDNet's Relative Performance Against Dataset Mean

The Relative Performance against Dataset Mean metric measures the superiority of a model by comparing its values with the dataset mean. The mean and the standard deviation of every value across all models are calculated first. Then, the z-score is calculated by Z as the difference between the model's value and the mean divided by the standard deviation. The higher the z-score, the higher the relative performance against other models. These are presented in Table 4.9 with LSDNet being the best performer, and also MF Model showing good competitive performance. The high Z-Score of 1.08 for mAP and 1.31 for Recall validates the capability of LSDNet in object detection at nighttime, making it well-suited for implementations such as adaptive headlamps and autonomous driving. An excellent Z-Score of 1.37 for FPS, accompanied by enhanced Recall and mAP scores, makes LSDNet the best model for real-time detection in dynamic scenes. This is supported by its excellent F1 Score Z-Score (1.23), which reflects an excellent balance between Precision and Recall, efficiently eliminating false positives and negatives. Although its Accuracy Z-Score (0.43) is similar to the baseline model, such consistency is a conscious design trade-off, focusing on detection at the expense of classification accuracy.

MF Model also performs well on main metrics. It has an FPS Z-Score of 1.07 to provide effective real-time detection at the cost of minimal Recall (0.48) and mAP (0.62). While lagging slightly behind LSDNet, it offers a sound alternative for real-time applications where speed and detection are both critical (Figure 4.39).

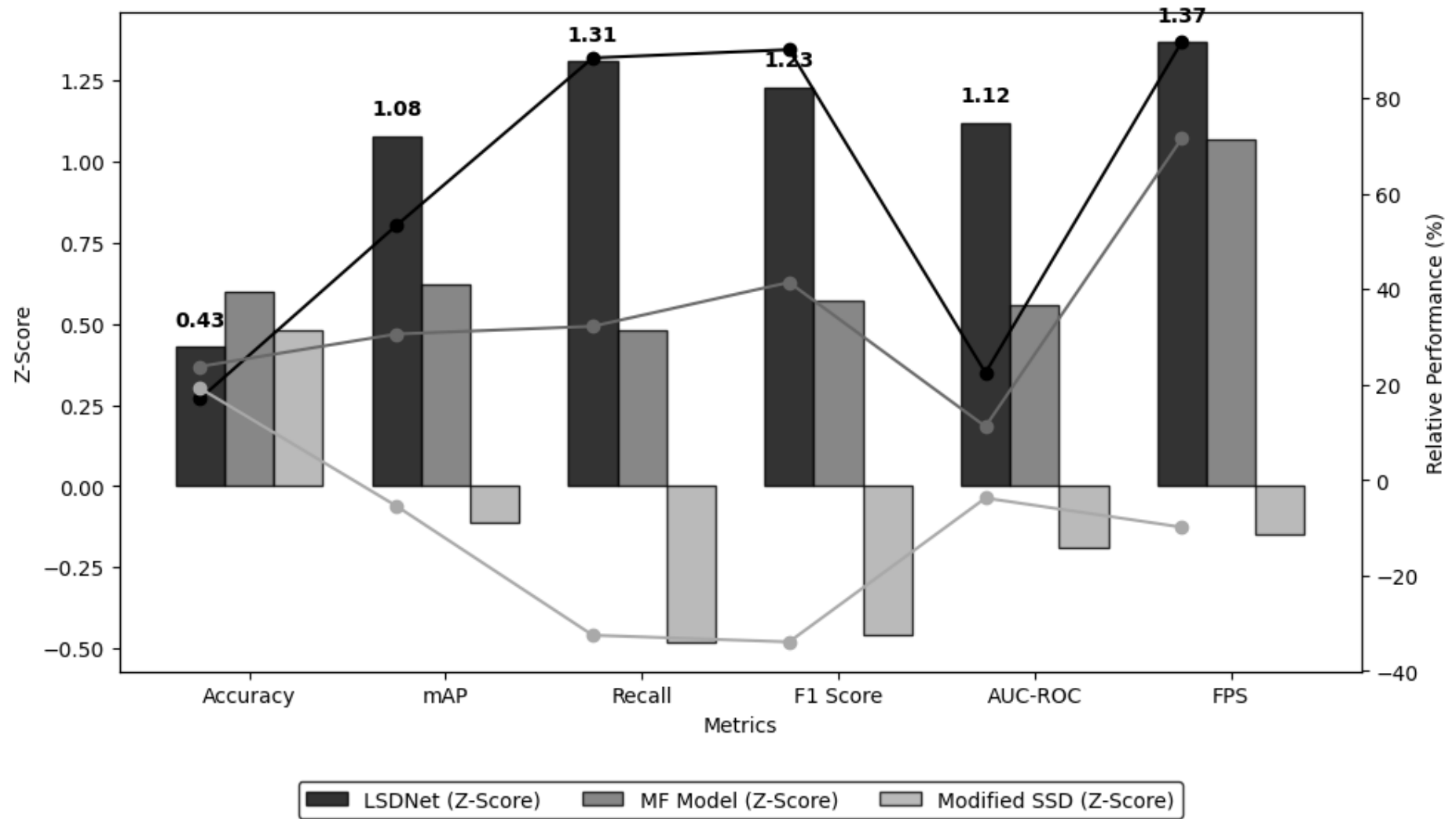


Figure 4.39 Z Score and relative performance comparison of various models

Table 4.9 Relative Performance Table for LSDNet, MF Model, and Modified SSD

Metric	Models Metric value			Dataset Mean	Std Deviation	Z-Score			Relative Performance (%)		
	LSDNET Value	MF Model Value	Modified SSD Value			LSDNet	MF Model	Modified SSD	LSDNet	MF Model	Modified SSD
Accuracy	52.03	55.03	53.00	44.44	17.70	0.43	0.60	0.48	17.08	23.83	19.26
mAP	0.68	0.58	0.42	0.44	0.22	1.08	0.62	-0.11	53.38	30.63	-5.41
Recall	0.75	0.53	0.27	0.40	0.27	1.31	0.48	-0.48	88.50	32.25	-32.50
F1 Score	0.78	0.58	0.27	0.41	0.30	1.23	0.57	-0.46	90.24	41.46	-33.90
AUC-ROC	0.98	0.89	0.77	0.80	0.16	1.12	0.56	-0.19	22.50	11.25	-3.75
FPS	50.66	45.31	23.80	26.40	17.75	1.37	1.07	-0.15	91.89	71.63	-9.85

Table 4.10 Comparison of Object Detection Models for Nighttime Applications

Model	Notable attributes	Performance gaps
LSDNET	Rarely fails to detect objects, even small ones, and works at real-time speeds, making it suitable for night object detection.	Because LSDNET focuses on detection rather than classification for real-time headlamp compensation, some items will be misclassified, which is fine for the purpose they are intended to serve.
Std Det Do Model	Correctly detects most objects, guaranteeing high detection reliability.	Detecting delays with slower real-time response.
YOLO V8	General object detection is best for it, especially during the daytime.	May miss out on detecting tiny objects at nighttime, lowering night detection reliability
MF Model	Balances detection accuracy and speed well, making it a good real-time candidate for night object detection.	Slightly worse recall and FPS than LSDNET, which can lead to occasional missed detections in high-speed scenarios.
Modified SSD	Works slightly better for detecting small objects and real-time execution than the regular SSD.	Has difficulty differentiating objects appropriately, resulting in increased false alarms.
SSD	Works marginally more reliably for object detection.	Too slow for real-time night use.
YOLO-Tiny	Works on low-power devices.	Misses majority of objects in low contrast lighting and is too slow for real-time detection and thus not adequate for nighttime object detection.

Meanwhile, the Modified SSD model falters by comparison, with negative Z-Scores in Recall (-0.48), F1 Score (-0.46), and mAP (-0.11) pointing at a significant reduction in detection reliability, thus unsuitable for high-speed real-time processing. AUC-ROC scores for LSDNet (1.12) and MF Model (0.56) verify their capacity to perform consistent detection under varied environmental conditions, highlighting their deployment feasibility. LSDNet's higher AUC-ROC score further emphasizes its resistance to detecting objects (varied conditions), making it a perfect candidate for nighttime object detection where both speed and precision are vital.

4.5.8 Model Trade-off Analysis

Determining the best detection model for this application includes assessing divergent parameters: accuracy, recall, and FPS. As the scene is low-light, with small objects present, the most critical requirement is to detect the objects and not classify them with high accuracy. This trade-off analysis thoroughly assesses the models to identify the best contender for real-time object detection in nighttime scenarios. There is a significant trade-off in speed and accuracy, as more accurate models generally demand more computational power, resulting in lower FPS. Models like YOLO V8 (Accuracy: 45, FPS: 14.56) and Std Det Do Model (Accuracy: 66.21, FPS: 34) have high classification accuracy but low inference speeds, which can cause latency in real-time detection applications.

On the other hand, LSDNET (Accuracy: 52.03, FPS: 50.66) shows a better balance by ensuring decent detection accuracy while drastically increasing FPS, hence being more. On the opposite end, models such as YOLO-Tiny (Accuracy: 6, FPS: 1.9) perform poorly in both detection accuracy and real-time processing. The tradeoff (of accuracy-speed) makes LSDNET the best as it achieves fast object detection while maintaining good classification performance. In addition to accuracy and speed, recall and mean average precision (mAP) are also important measures of a model's detection effectiveness and localization accuracy. Recall is most important in nighttime detection—where false negatives (missed detections) are more dangerous than false positives (misclassifications)—models with better recall are preferred. LSDNET attains the best recall of 75.36 and competitive mAP of 68.08, with the ability to detect objects with less omission and with good localization accuracy. Std Det Do Model comes in at second place with recall of 72.37 and mAP of 66.21 and can be relied upon as a backup. Conversely, YOLO V8 shows a comparably good mAP of 44 but a poorer recall of 56, indicating that although it is very good at accurate localization, it will likely lose objects, especially at night.

Down at the bottom is YOLO-Tiny, with a recall of 2.5% and an mAP of 4.1%, falling short of the minimum detection threshold, as expected. The focus on recall rather than mAP also confirms LSDNET as the best model since it consistently detects objects without substantial omission while keeping decent localization accuracy. The key comparisons of the models are shown in Table 4.10. Another crucial metric for model performance is the F1 Score (measure of the proportion of precision and the recall) to ensure accurate detection. But attaining a high F1 Score should not be at the cost of real-time usability, requiring a compromise with FPS. LSDNET surpasses other models yet again by having the best F1 Score (0.78) while retaining a best-in-industry FPS (50.66), which makes it the most efficient model. The Std Det Do Model, with an F1 Score of 0.778 but lower FPS of 34, is still a good alternative, albeit less efficient for real-time usage.

Other models, including SSD and YOLO, are not as good at both F1 Score (0.214 and 0.201, respectively) and FPS (14.56), and thus are less competitive for dynamic, low-visibility settings. Because both F1 Score and FPS are very important in achieving effective object detection, LSDNET is the most well-rounded option. Efficient computation is important for real-world deployment, especially in embedded devices, autonomous vehicles, and surveillance, where the processing power is constrained. LSDNET has the best FPS without compromising on high recall and F1 Score, which means it is designed to be computationally efficient. This is in stark contrast to YOLO V8 and SSD, which demand much more computation to deliver mere moderate FPS and hence are not as suitable for real-time operations.

The capacity for efficient high-speed processing with low computational overhead places LSDNET as the best candidate for use in resource-limited environments. According to this thorough trade-off analysis, LSDNET stands as the best optimized model for object detection at night. It is able to balance speed, detection accuracy, and computational cost well, making it the best fit for real-time usage where detecting objects, not classifying them, is the goal. The model's most significant strengths are the highest FPS (50.66), guaranteeing real-time detection ability; the highest recall (75.36), minimizing the chance of missing objects; a competitive mAP (68.08) for precise localization; a robust F1 Score (0.78), guaranteeing dependable detection; and effective computational efficiency, allowing it to be deployable in real-world systems. These features make LSDNET the best model for nighttime object detection, with its trade-offs tuned for low-light, on-road applications.

Additional optimizations and sharpening could further improve its effectiveness in ultra-low-light environments, but its present detection effectiveness and computational efficiency make it the first choice for actual deployment.

4.5.9 Summary of the performance of LSDNet and future directions

LSDNet tackles the challenges of nighttime object detection by employing inference-level adaptations that improve model effectiveness in poor illumination. Traditional detectors fail due to poor contrast and difficulty identifying small or distant objects. LSDNet employs the Efficient Det D_0 backbone with brightness-aware modulation, size-aware confidence scaling, and class-prioritized detection for enhanced sensitivity under low-light conditions. Benchmarking proves its supremacy: LSDNet improves recall by 4.1% over Efficient Det D_0 and achieves more than 23.1% improvement in small item recall. It registers a 5.85% improvement in average precision at extreme low-light levels, with a mean average precision of 68.08, outperforming YOLOv8, SSD, and their variants.

LSDNet attains 50.66 FPS inference speed, improving over the baseline by 34% while retaining higher detection quality, thus enabling real-time deployment in embedded AI systems for ADAS, self-driving cars, and smart traffic control. The performance lies essentially upon its dynamic adjustment of detection confidence based on scene brightness and object size. This improves identification of vulnerable traffic participants like pedestrians and bicycles. Class-specific weighting improves detection precision for critical objects like cars and obstacles, thus reducing false positives. Future improvement includes TensorRT-based acceleration for lowering latency, Transformer-based attention for adaptive feature extraction, and GAN-based augmentation for enhanced low-light generalization. IoU-aware loss functions like GIoU and DIoU are designed to improve bounding box localization. LSDNet offers a robust, efficient, and real-time solution to safety-critical low-light detection tasks, with huge potential in autonomous vehicles, smart surveillance, and adaptive illumination systems.

4.5.10 Summary of the performance analysis of the object detection models

LSDNET is the most reliable and effective model for object recognition at night, particularly in real-time adaptive headlamp applications, based on a comparative performance evaluation of alternative object detection models. At night, LSDNET accurately detects small or occluded objects with a 0.78 F1 score, a 52.03% detection rate,

and an extremely high mean average precision of 68.08%. Its 50.66 frames per second (FPS) real-time processing ability further enhances its suitability for situations involving fast-moving cars. This limitation is tolerable in adaptive headlamp compensation, where immediate availability of an object is more important than its precise label, even though its primary emphasis is on object detection rather than classification, which sometimes leads to misclassifications. High reliability is also reflected by the Standard Detection Do model, with an acceptable AUC-ROC score of 0.98 and an F1 score of 0.778. Its reduced processing speed (34 FPS) at high-speed driving, though, could result in noticeable lag when compared to LSDNET. The Modified SSD and Multi-Faceted (MF) models are balanced ones with a mid-level speed and detection rate. Especially in normal nighttime driving conditions, the MF model performs well; however, it lags behind LSDNET in recall and real-time response, which may lead to occasional missed detection when traveling at high speeds.

On the other hand, YOLO-V8 has issues with identifying small or low-contrast objects during nighttime, although it is robust in typical object identification scenarios and daytime settings. This affects its reliability in uses where these types of detections are needed during nighttime. Typical SSDs do not have the speed required for real-time processing, although they have a good detection power. Although created to be executed on low-power hardware, YOLO-Tiny falls short due to its slow inference time and poor low-light performance. In conclusion, LSDNET is the optimal model for nighttime object recognition in real time, while MF and Mod SSD serve as secondary options for a few particular situations. Due to their poor performance at night, models such as the YOLO-Tiny and the standard SSD are not suitable for consideration by adaptive headlamp systems.

4.6 Overview on the Lane detection techniques

This subsection discusses a lane detection method for adaptive headlamp control that calculates steering angle and curvature through OpenCV-based methods to set beams to increase illumination. It also identifies the manner in which deep learning is replacing traditional feature-based approaches for detecting roads [76]. Because of their ability to perform end-to-end feature extraction and learning [78], CNNs have replaced more traditional methods like HoG and RealBoost in tasks such as pedestrian and vehicle detection [24][77][11]. Even though CNNs are more accurate, real-time adjustments such as YOLO and Fast R-CNN have been introduced because of their computational

requirements [79]. To cope with challenging conditions such as weathered lane markers, curves, and glare, lane detection methods have developed. Probabilistic models and sensor fusion approaches (e.g., camera + LiDAR) enhance robustness [41][83], while vanishing point detection [86] and graph-based ranking [85] enhance boundary estimation in complex road textures. Lightweight inference models [82] and semantic segmentation networks such as UPA and UBA [88] provide high spatial accuracy at low overhead, making them suitable for embedded systems. For improving performance, recent models further focus on trajectory prediction using spatial feature pooling [90] and monocular vision [89]. Obstructions, faded lane markings, and varied road geometries still pose challenges. Improving adaptive lane-aware headlight systems requires a deeper integration of deep learning with multi-sensor fusion and semantic reasoning.

4.7 Framework for lane detection and integrating LSDNet model to operate in multiple driving scenarios⁶[108]

Lane detection is crucial for autonomous vehicles and driver-assistance systems, enabling safe navigation and precise decision-making. This section details a comprehensive lane detection technique, illustrated in Figure 4.40. The method employs various image processing methods to reliably recognize lane markings across diverse environments. Initially, a gamma correction function enhances the brightness and contrast of the original image. Subsequently, the gamma-corrected image undergoes bilateral filtering to smooth it while preserving essential edge information. The Canny edge detection technique, applied through the canny function, extracts relevant edges, generating a binary image highlighting the detected edges.

The Hough transform function is then utilized with both the original and binary edge-detected images to locate lines within the edge-detected image. This method employs thresholding, grayscale conversion, and line detection to create an image displaying realized lines and relevant information. A clustering technique is developed based on parameters derived from identified lines and the original image, enhancing and ordering the detected lines. Related lines are clustered, resulting in detected lane lines. An HSV filter is subsequently applied to the BGR image, along with assigned minimum and

⁶ Section 4.7 of this Chapter was presented at an International Conference and has received the Best Paper Award. The details are as follows: Toney, Glenson, Sethi, Gaurav and Cherry Bhargava. "A Novel Lane Detection Approach for Vehicle Adaptive headlamps" International Conference on Advances in Smart Sensor, Signal Processing and Communication Technology, Goa University, India, March 2024

maximum HSV values for yellow and white colors, distinguishing lane marker colors. The resulting image accentuates pixels within the specified color range relevant to lane markings by filtering pixels within specified thresholds. This algorithmic approach correctly identifies and distinguishes lane markers across various environmental scenarios, potentially enhancing the safety and efficiency of driver assistance and autonomous driving systems.

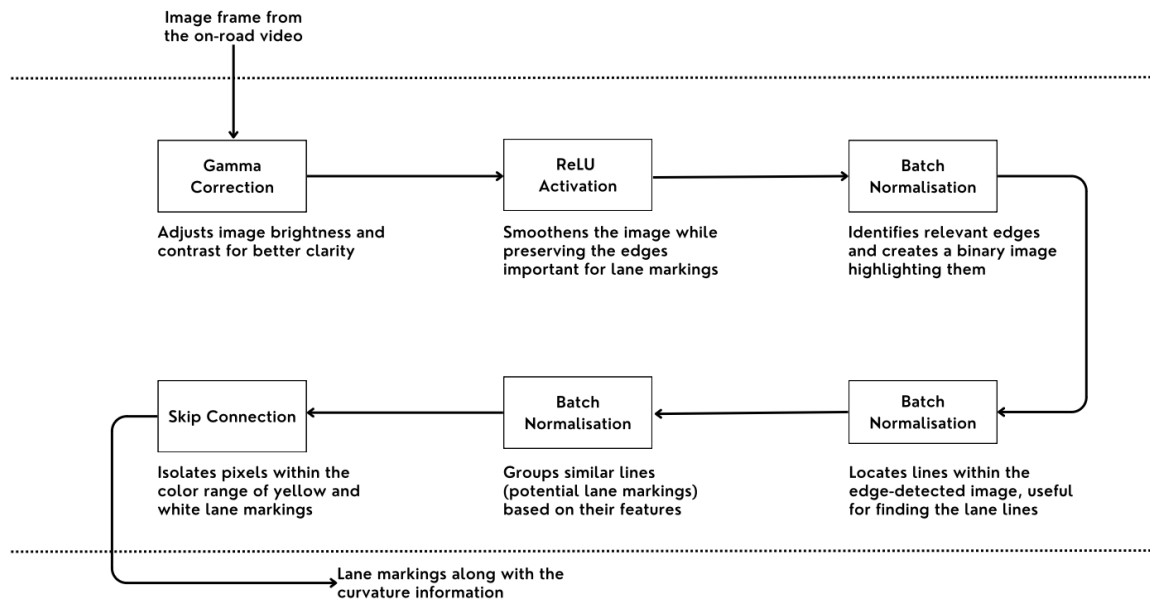


Figure 4.40 The lane detection intermediate processes [108]

The Gamma Correction tool improves image quality by adjusting brightness and contrast to account for the nonlinear relationship between pixel values and perceived brightness.

$$I_c = I_{original}^\gamma \quad (4.38)$$

Equation (4.38) defines the gamma correction function, where I represents the original pixel intensity, I_c denotes the corrected pixel intensity, and γ alters the image's brightness and contrast. Higher gamma values (> 1) emphasize details in brighter areas, while lower values (< 1) enhance details in darker areas, helping to compensate for changes in the display system's response curve. This correction enhances the image's dynamic range and tonal representation, particularly in scenarios with varying lighting and exposure levels.

To preserve edges and reduce noise, the bilateral filter (Equation 4.39) functions as a non-linear, edge-preserving smoothing filter, averaging pixels based on spatial closeness and intensity differences.

$$\text{Bilateral}(I, \sigma_s, \sigma_r) = \frac{1}{W_p} \sum_{q \in \Omega} G_s(\|p - q\|, \sigma_s) \cdot G_r(\|I_p - I_q\|, \sigma_r) \cdot I_q \quad (4.39)$$

Here, W_p represents the normalization term, I_p and I_q are intensity values at pixels p and q respectively, G_s is the spatial Gaussian kernel function with standard deviation σ_s , G_r is the range Gaussian kernel function with standard deviation σ_r , and I is the input image. Additionally, a Gaussian filter is applied to smooth the image, controlled by the standard deviation of the Gaussian kernel, regulating smoothing intensity. Sobel operators calculate the gradient of the smoothed image, aiding in discerning intensity variations in both horizontal and vertical directions. Subsequently, non-maximum suppression retains only local maxima along the gradient direction, resulting in thinner edges and preserving only the strongest edge responses.

After detecting potential edges, a double thresholding process distinguishes strong and weak edges based on gradient magnitudes relative to high and low thresholds. Strong edges have magnitudes above the high threshold, while weak edges fall between the low and high thresholds, with pixels below the low threshold discarded. Edge tracking, based on hysteresis, connects weak edges to strong ones, considering only those weak edges linked to strong ones as components of an edge, guided by gradient directions. Configuring thresholds is crucial and depends on image properties, balancing sensitivity (detecting all edges) and specificity (excluding noise).

$$r = x \cos(\theta) + y \sin(\theta) \quad (4.40)$$

The Hough Transform represents detected shapes in a coordinate system, employing polar representation (Equation 4.40). Here, (r, θ) denotes a point in Hough space, with θ as the angle between the x-axis and the line, and r as the perpendicular distance from the origin to the line. Votes aggregate in the Hough space, a 2D parameter space representing r and θ for line detection. Each edge pixel contributes to a sinusoidal curve in the Hough space, accumulating votes for intersections of corresponding curves, revealing lines in the image as peaks in the Hough space. Normalization enhances lane marking visibility in low-light driving conditions by ensuring consistent pixel values (Equation 4.41). This aids clustering algorithms in distinguishing lane markings from dark backgrounds.

$$P_{\text{norm}} = \frac{P - \min(P)}{\max(P) - \min(P)} \quad (4.41)$$

The K-means algorithm (Equation 4.42) is employed to minimize the within-cluster sum of squares, where I represents the image, C denotes the cluster centroids, and $\|\cdot\|$ indicates the Euclidean norm and distance metric.

$$\text{argmin} \sum_{k=1}^K \sum_{i=1}^{N_k} \|I_i - C_k\|^2 \quad (4.42)$$

Euclidean distance (Equation 4.43) calculates the distance between two points in Euclidean space, aiming to minimize the sum of squared Euclidean distances within clusters. Each cluster corresponds to a distinct lane or background region. Euclidean space is given by

$$d(I_i, C_k) = \|I_i - C_k\| \quad (4.43)$$

The HSV (Hue, Saturation, Value) filter represents colors in a color space that separates chromatic information (hue and saturation) from intensity information (value or brightness). Particularly useful for color image analysis and editing, the HSV filter allows manipulation of various color ranges without directly altering intensity, making it effective in different lighting conditions.

4.7.1 The lane detection

The input image's pixel intensities, illustrated in Figure 4.41, undergo modification using the Gamma Correction Function, depicted in Figure 4.42. This mathematical adjustment employs a power-law relationship to correct gamma, thereby enhancing brightness levels, improving contrast, and highlighting details in both bright and dark areas. The resulting pixel values provide a perceptually accurate representation of the original image. The Bilateral Filter function, demonstrated in Figure 4.43, smoothens the input image by suppressing noise while preserving significant edges and patterns. This filter considers spatial proximity and intensity differences between pixels, ensuring effective noise reduction. Utilizing the Canny Edge Detection method, a binary edge map is generated, depicted in Figure 4.44, categorizing pixels based on their gradient magnitudes. This map aids in identifying essential image properties by highlighting principal edges and reducing false positives in noisy images. Figure 4.45 illustrates the Hough Transform, which performs shape recognition tasks such as line and circle detection. Particularly useful for identifying lines and curves within images, this transform utilizes a line's slope (m) and y -

intercept (b) to represent it in the Cartesian coordinate system. Figure 4.46 presents the chromatic details of the image frame, while Figure 4.47 a & b showcases lane recognition and projection during lane changes and vehicle movement. These projections accurately reflect the road on which the vehicle is traveling. The model detects lane projections and computes the left and right curvatures based on these projections, as depicted in Figure 4.49 a & b. By analyzing the lane outer and inner curvatures along with the vehicle's wheelbase, the model estimates the vehicle's slip angle, also known as the steering angle, which determines how much the vehicle's body should tilt in relation to the tires



Figure 4.41 An original frame

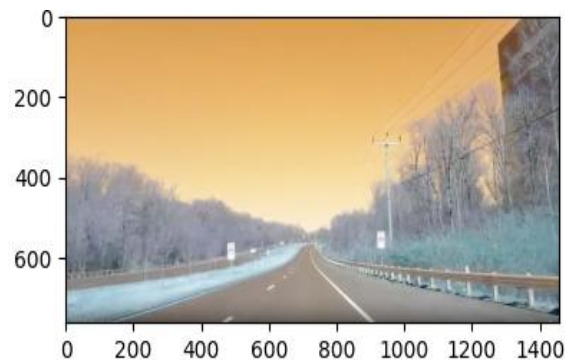


Figure 4.42 Image-Gamma Corrected



Figure 4.43 Image with Bilateral Filter

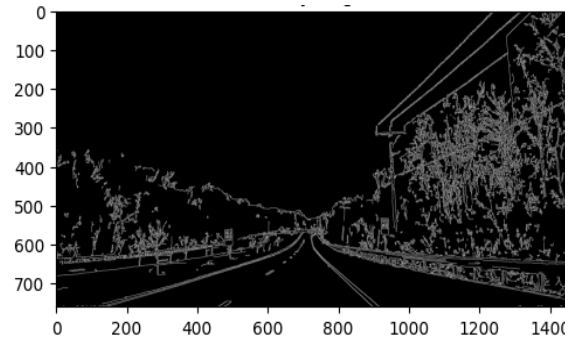


Figure 4.44 Picture employing Canny Edge recognition

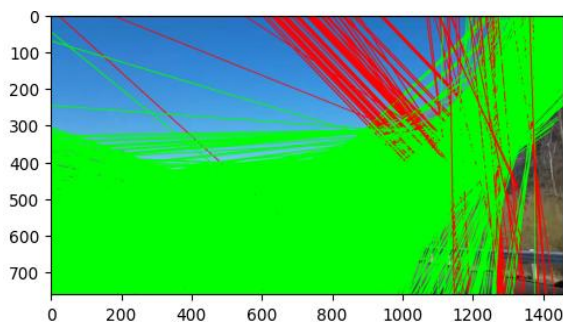


Figure 4.45 Hough Lines Clustered in an Image

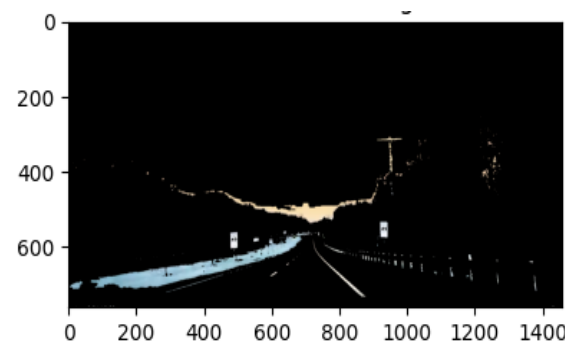


Figure 4.46 Filtered Image with HSV

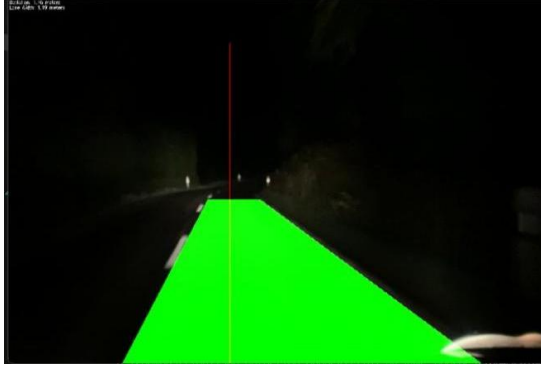


Figure 4.47



Figure 4.48

Both during lane changes and while the vehicle moves across the path, lane recognition and projection are performed. The road that the car is traveling on is reflected in these projections.



Figure 4.49



Figure 4.50

The left and right curvatures are computed by the model based on the lane projections. In a and b, the model that forecasts the lane projections under various lighting scenarios is displayed.

This information is crucial for adjusting the vehicle's headlights to illuminate the road optimally, especially in low-light conditions, as shown in the model's predictions under various lighting scenarios. Accurately predicting the slip angle is essential for ensuring maximum visibility and safety while driving, particularly at night. Vehicles equipped with systems that adjust headlamp angles based on this information can effectively respond to changes in road conditions, thereby enhancing driving safety and overall experience while reducing the risk of accidents.

4.7.2 The calculation of the angle of vehicle wrt to the lane curvature

Following lane detection, the model assesses the angular disparity between the center of the dashboard (blue line) and the center of the lane (red line), illustrated in figure 4.51. In this depiction, the distance 'd' between the lower edge of the dashboard and the center (0,0) of the screen remains constant, while the distance between the screen's origin (0,0) and point $r(x,y)$ varies.

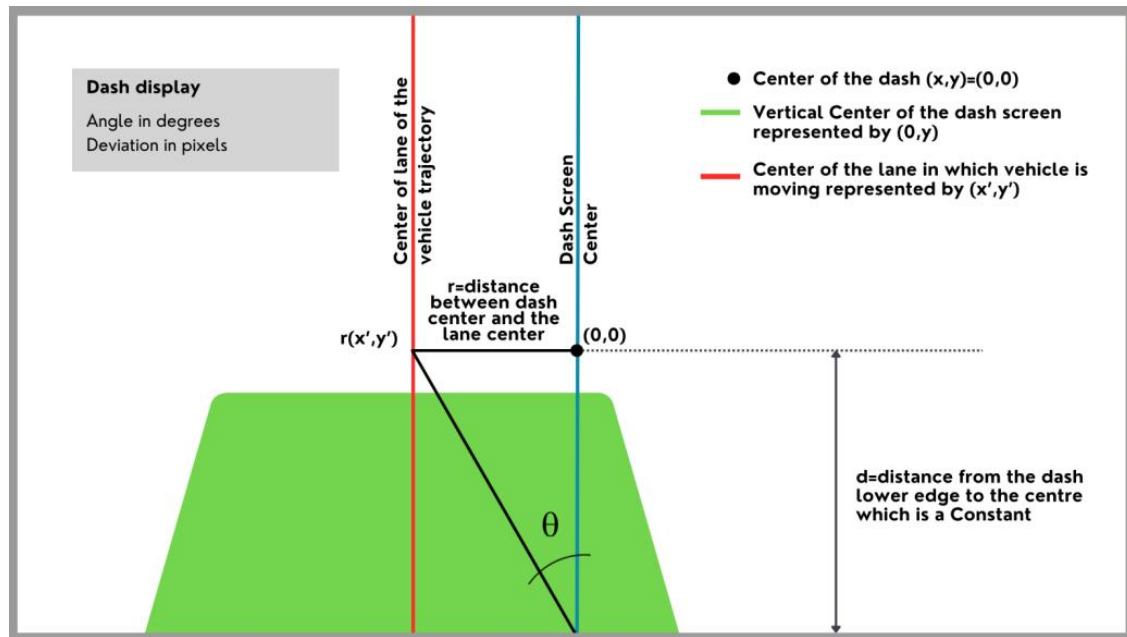


Figure 4.51 The dashboard perspective to compute the compensation value for adjusting the headlamp to enhance visibility when the vehicle executes a turning maneuver within the lane.

Table 4.11 The headlamp adjustment criteria based on the lane trajectory and the values of the intermediate variables - r and θ .

Vehicle Condition	r value	θ value	Left Headlamp	Right Headlamp
No turn	0	0	No	No
Left turn	< 0	Negative	Yes	No
Right turn	> 0	Positive	No	Yes

The angle θ , denoting the lane angle, reflects the adjustment needed for headlamp illumination. This angle is calculated using the formula $\theta = \arctan(r/d)$. Table 4.11 presents the values of r and θ for different lane conditions and the corresponding headlamp adjustments for improved illumination. This is assuming the fact that the dashboard screen size is fixed and the center line remains constant. The Vehicle Condition describes different driving situations, including no turn (when the vehicle travels straight along a lane), left turn, and right turn. The r Value row quantifies the lateral deviation of the vehicle from the lane's centerline, with negative values indicating leftward deviation and positive values indicating rightward deviation. Meanwhile, the θ Value represents the angular deviation of the vehicle from the lane's centerline, calculated based on the arctan function.

During a left turn, the vehicle deviates to the left, resulting in negative values for both ' r ' and θ , while during a right turn, the vehicle deviates to the right, leading to positive values

for ' r ' and θ . The last two rows indicate whether compensation is applied to the left or right headlamp based on the vehicle's lateral deviation. For example, during a left turn, compensation is applied to the left headlamp to illuminate the turning path, while during a right turn, compensation is applied to the right headlamp. In scenarios where the vehicle travels straight along the lane, no compensation is applied to either headlamp. This compensation mechanism ensures optimal illumination of the road, enhancing visibility and safety, particularly during turning maneuvers.

4.8 Integrating the multifaceted object identification model and the lane detection model

Enhancing beam light adjustment along a curve and dynamically toggling high and low beams depending on object detection is essential to provide effective performance in all these scenarios. PCB hardware implementation is utilized for this integration. As indicated in Figure 4.52, this approach enables seamless integration of complex object recognition models and beam angle adjustments based on predictions from lane detection models. This hardware structure enables real-time processing and response to dynamic road conditions, enhancing night driving safety and comfort. The revolutionary dynamic headlamp alignment and high beam adjustment technology combines hardware and AI models to enhance driving safety and comfort.

It employs a Raspberry Pi 4, multidimensional AI models, lane detection, MOSFET, and servo motors. These elements combine to accurately change headlamp position and control high beams automatically depending on traffic conditions. A XL7056 buck converter supplies the Raspberry Pi and other logic circuits, and the XL4015 supplies servo motors and high beam strips. Pulse Width Modulation (PWM) technology from the Raspberry Pi manages the servo motors and high beam strips, enabling precise headlamp orientation adjustments according to AI model lane detection information. The AI algorithm forecasts the inner and outer lane curvatures, which are used to determine the optimal headlight orientation during turns.

Another essential part of the system is the holistic object detection AI model, which can accurately detect cars, pedestrians, and barriers. MOSFET (IRF520) triggers when something is sensed, extinguishing the model's high beam strip. MOSFETs enhance reliability and switching rates, allowing fast road condition reactions. The MOSFET

minimizes pedestrian and auto glare by keeping the high beam strips isolated from the ground, enhancing safety on the road. This entire system combines innovative technology to address nighttime driving issues. The hardware model is based on the object detection and lane detection model's lane curve angle information of the AI model.

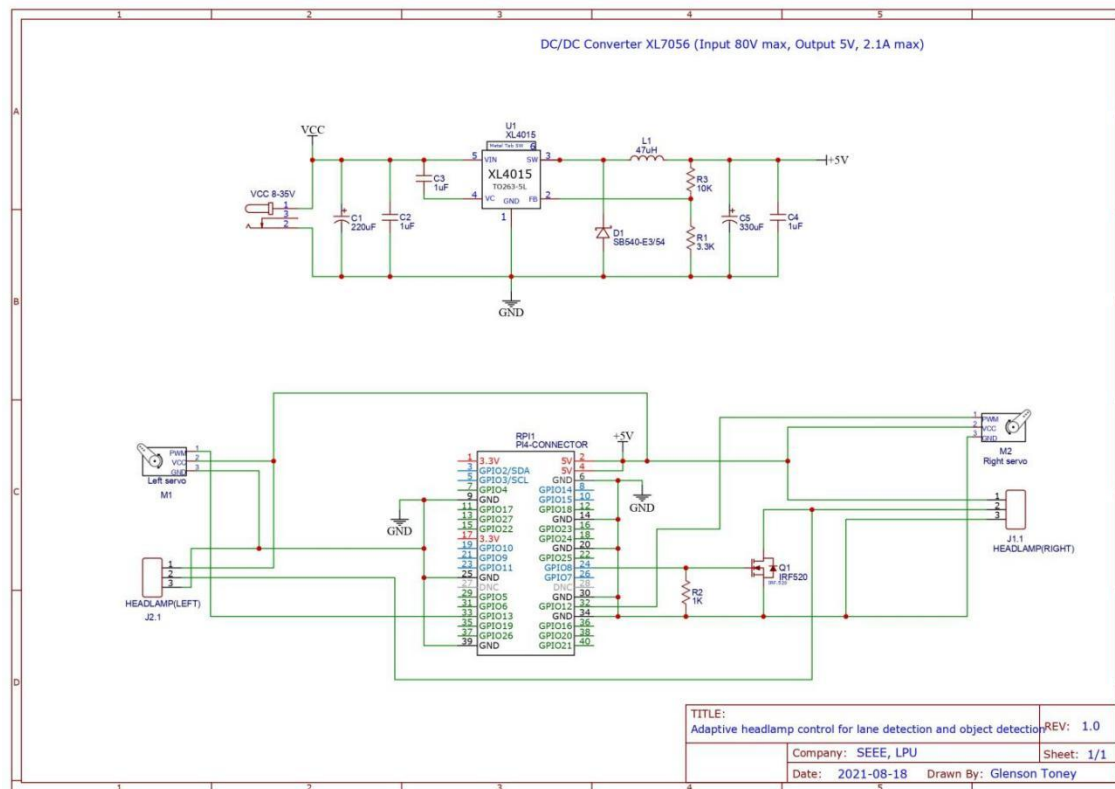
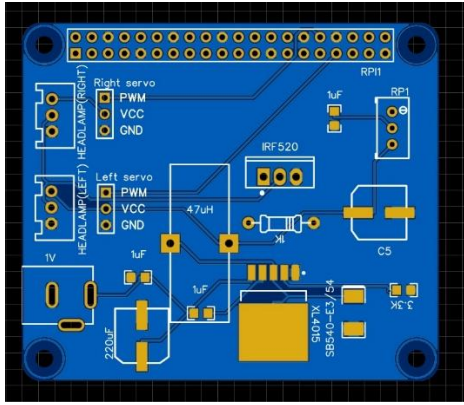
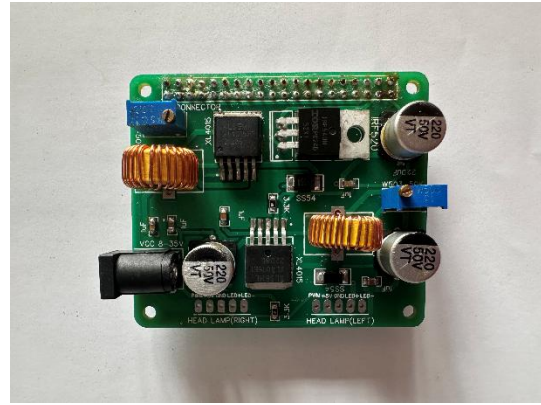


Figure 4.52 The dashboard view for the calculation of the value for compensating the headlamp for improving visibility as the vehicle does a turning maneuver across the lane

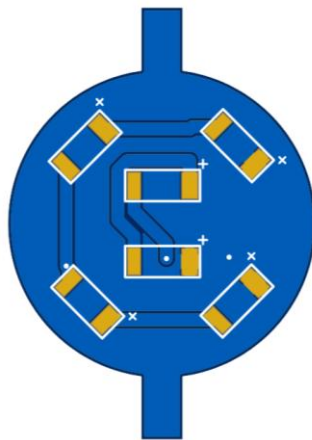
The surface-mounted device PCB layout for implementing the adaptive headlamp control unit is shown in Fig. 4.53 a and the actual dual layer PCB in Fig. 4.53 b. The PCB layout of the LED arrangement (for both high and low Beams) and their interconnections are shown in Fig. 4.53 c and Fig. 4.53 d respectively. A 3D model of the adaptive headlamp is shown in Fig. 4.54 a and the top-view of the 3D printed Headlamp is shown in Fig. 4.54 d. The high and low beam configuration are shown in Fig. 4.54 b and Fig. 4.54 c respectively. To prevent glare to other drivers, turn the headlamp to low beam when an object or vehicle is sensed on the road. The MOSFET turns off the high beam strip to avoid headlights blinding oncoming drivers. Figure 4.55 illustrates the high beam control signal plot. A high pulse indicates the high beam is ON, while a zero pulse suggests an object was detected and the high beam strip is off.



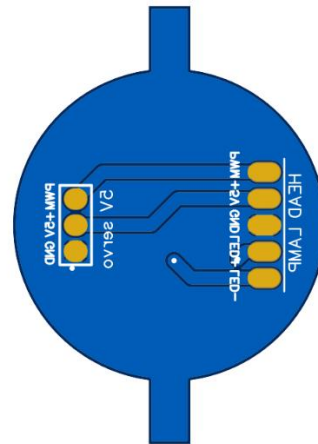
(a) SMD PCB Layout of the Adaptive headlamp designed on Easy EDA.



(b) Dual layered PCB Adaptive control board.



(c) High and low beam LED placement.

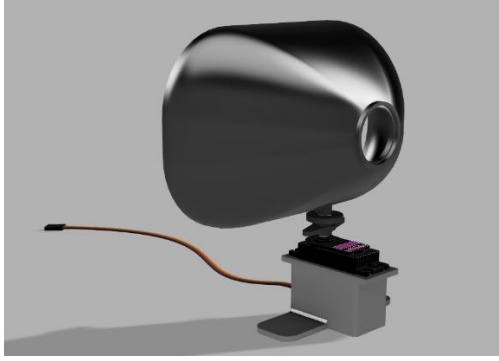


(d) High and Low Beam LED's PWM based power connection layout.

Figure 4.53 The PCB designs and implementation.

Objects were detected during frames 20 to 250, 260 to 270, and 360 to 500, disabling the high beam strip. This feature enhances driving comfort and eliminates headlamp manual adjustments, which lessens driver stress. Adaptive lighting control systems are also applicable to autonomous vehicles to enhance road safety and efficiency. Figure 4.56 indicates how the system computes lane deviation in relation to the vehicle path and compensates the light beam to enhance road visibility.

The compensation angle approximately equals the deviance, depicting the tire-body angle of the vehicle. The body angle of the vehicle is different from its tire angle when turning and makes the headlamp deviate from the lane angle. It sends a control signal to regulate the headlight by the angle detected to correct the deviation and align it with the wheel angle, as illustrated in the figure. This dynamic adjustment mechanism optimizes illumination, particularly in turns, enhancing driver safety and visibility. The system operation is carried out by the functioning of Servo Motor 1 and 2, for the left and right headlamps respectively.



(a) The 3D design of the adaptive headlamp designed using Autodesk Fusion 360.



(b) High-beam configuration



(c) Low-beam configuration



(d) The topview of the model.

Figure 4.54 The software design and the hardware prototype of the adaptive headlamp model for integrating the LSDNet and the Lane detection techniques.

As shown in Table 4.11, Servo 1 functions in the case of left turns, while Servo 2 is on for right turns. Figure 4.57 illustrates that Servo 1 is on for left turns while Servo 2 remains off for 100 frames. The duties transition from frame 100 to close to 400 in right turns. This modification enhances road perception and driving. Table 4.12 and Figure 4.58 a & b demonstrate that object detection and trajectory angle identification can simultaneously be achieved. Table 4.12 illustrates what occurs when no vehicle or object is detected during lane shifting or when there is no change. Control signals are used by the hardware prototype to control these situations. They regulate the MOSFET for the high-low beam switching and the PWM signals for the compensation of trajectory angle. It's made for unpredictable road conditions of nighttime driving.

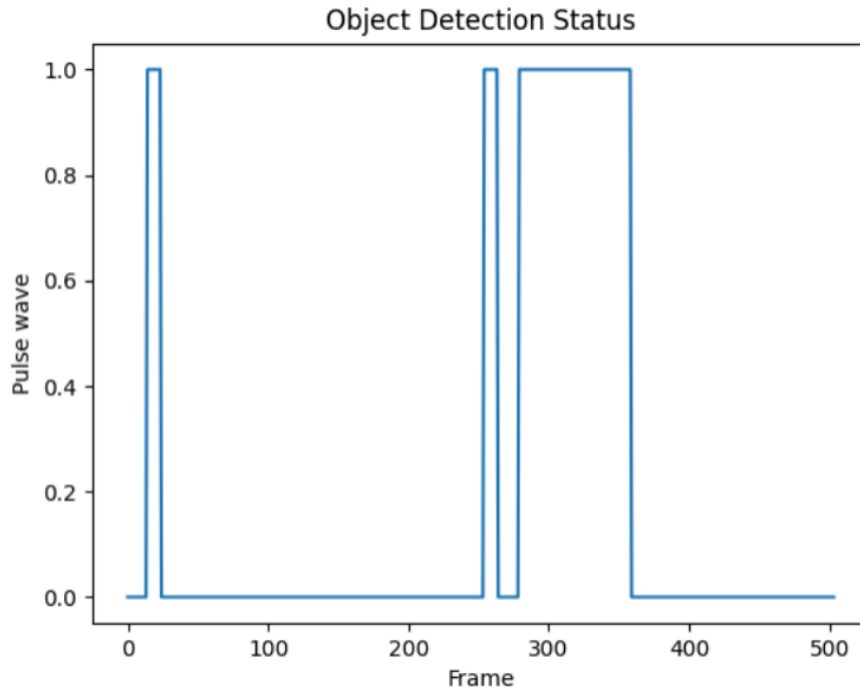


Figure 4.55 The plot illustrates the pulse modulation responsible for activating and deactivating the MOSFET, subsequently regulating the high beam strip of the vehicle's headlamp.

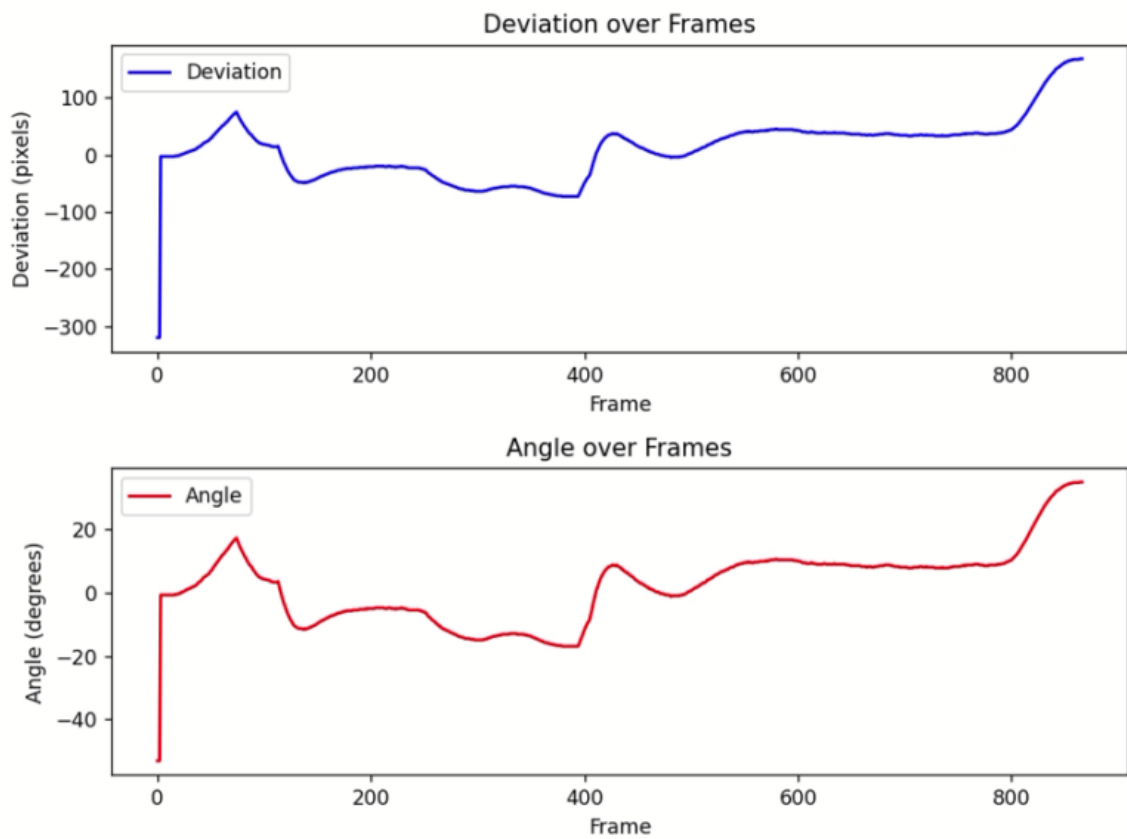


Figure 4.56 Variation plot of the vehicle along the trajectory and the corresponding angles calculated by the lane detection model.

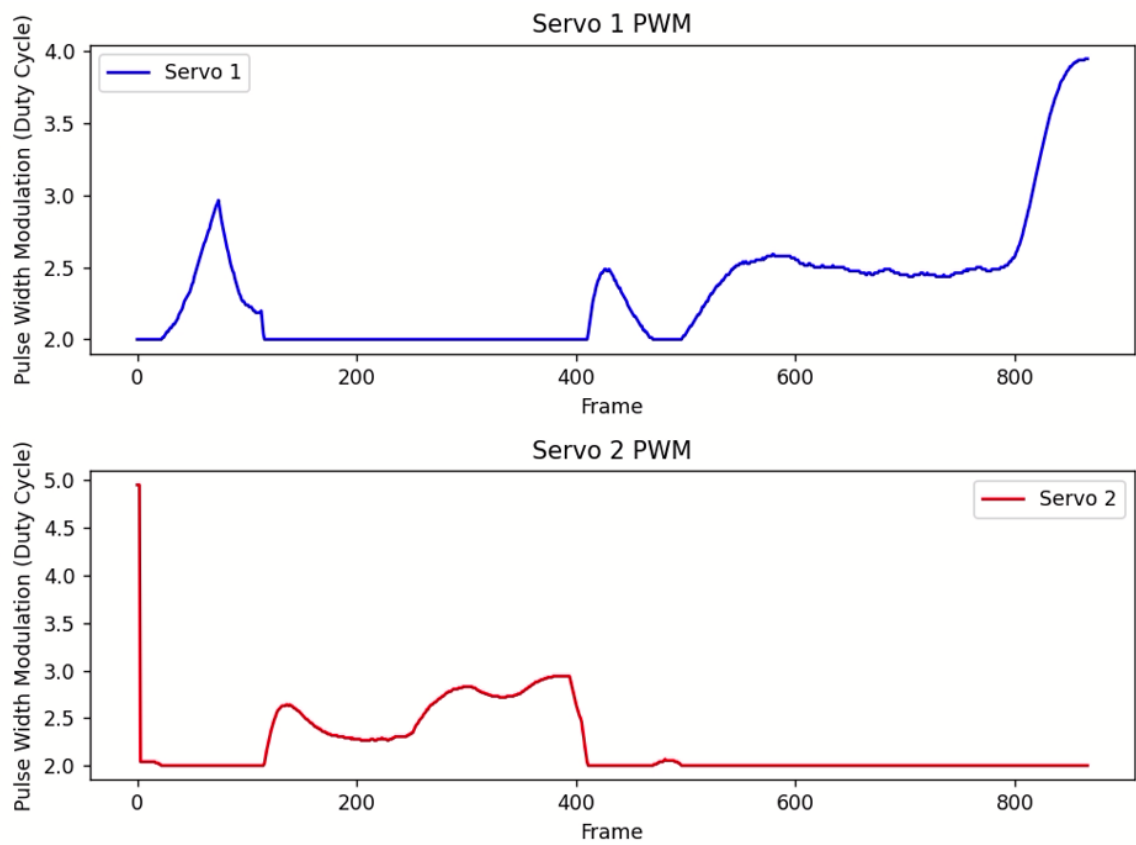


Figure 4.57 The duty cycle of servo motors 1 and 2 during left and right maneuvers of the vehicle across frames.

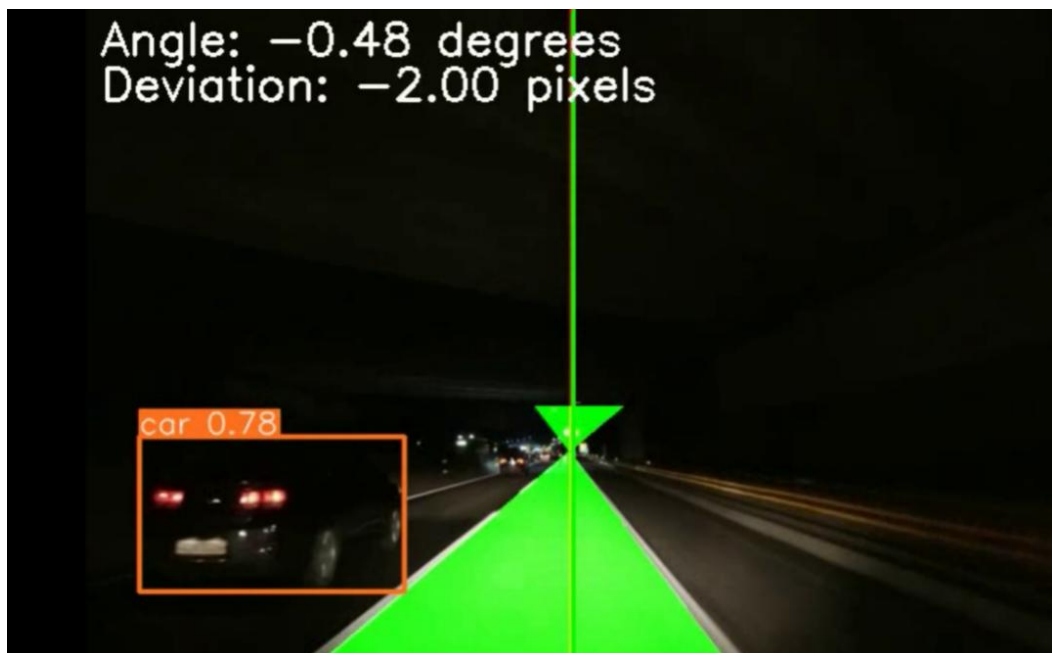


Figure 4.58 a



Figure 4.58 b

Figure 4.58 The model integration enables the concurrent detection of on-road objects and the trajectory angle.

Table 4.12 The adaptive headlamp compensation details to the changing conditions encountered during on-road driving. The table illustrates the intermediate status of the hardware prototype, indicating the control points for this implementation wrt the Figure 4.52

On-road Condition	Condition of the Trajectory	MOSFET		Status of the Beam Lobe Strip	Is PMW1 (Pin 33) Varied?	Is PWM2 (Pin 32) Varied?
		IRF620 Control Signal Pin (Pin 24)	MOSFET IRF620 Operating Region			
No Object or Vehicle detected	No turn	High	Saturation	High beam	No	No
No Object or Vehicle detected	Left turn	High	Saturation	High beam	Yes	No
No Object or Vehicle detected	Right turn	High	Saturation	High beam	No	Yes

On-road Condition	Condition of the Trajectory	MOSFET IRF620 Control Signal Pin (Pin 24)	MOSFET IRF620 Operating Region	Status of the Beam Lobe Strip	Is PMW1 (Pin 33) Varied?	Is PWM2 (Pin 32) Varied?
Object or Vehicle detected	No turn	Low	Cutoff	Low beam	Yes	No
Object or Vehicle detected	Left turn	Low	Cutoff	Low beam	Yes	No
Object or Vehicle detected	Right turn	Low	Cutoff	Low beam	No	Yes

4.9 Discussion

The design of an adaptive headlight management system based on real-time sensing to enhance the safety of nighttime driving was discussed in detail in Sections 4.6 to 4.8. To adjust beam direction and intensity based on driving conditions, the system integrates object recognition, vehicle trajectory estimation, and lane detection. OpenCV is employed to detect lanes through conventional techniques like bilateral filtering, gamma correction, Hough transform, and Canny edge detection. This enables the estimation of the steering angle and yaw rate, making beam deflection compatible with the road curvature.

When obstacles or cars are detected, dynamic beam switching is enabled through LSDNet's illumination-aware preprocessing and confidence modulation that manage object detection. Object and trajectory inputs are mapped to specific beam lobe and intensity responses by a rule-based control algorithm. Simulation, visual plots, and profiles of control signals under different road conditions were employed to validate the system's effectiveness. With its light computational load, quick response, and scalability, the framework—developed for embedded systems—is a good starting point for next-generation ADAS applications that include sensor fusion and deep learning control.

4.10 Summary

To enhance safety and visibility while driving at night, this chapter (section 4.1 to 4.5) gave a comprehensive architecture that integrates trajectory-based beam management and advanced object identification. A comparative analysis of object identification models in

the first section confirmed that LSDNet was the most reliable in low-light environments. In challenging light environments, LSDNet exhibited its best performance in detecting small and occluded objects with an impressive F1 score of 0.78, detection rate of 52.03%, and real-time processing at 50.66 FPS. In adaptive headlamp usage, LSDNet's balance of speed and accuracy made it the optimum choice, although other choices such as the MF model and Modified SSD had average performance.

The second segment (section 4.6 to 4.9) built on this by incorporating LSDNet into an adaptive illumination system with lane awareness. The system projected the vehicle trajectory and steering angle by deriving lane boundaries and curvature through OpenCV's classical image processing algorithms. This was utilized for dynamic direction and intensity beam control in combination with real-time object detection via LSDNet. Turning behavior and object presence were translated into precise beam commands by a rule-based control logic. Simulation and signal analysis were employed to verify the entire system, which demonstrated low processing cost, responsiveness, and scalability for embedded deployment. This work provides the foundation for future integration with sensor fusion and deep learning control methods.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

In this thesis, the development and design of adaptive headlamp systems for automobiles are examined with the objective of enhancing the safety and comfort of driving using advanced technology. The purpose of this study was to solve a long-standing issue of automotive safety: the inadequacy of conventional car headlamps during night driving, especially on winding roads and in poor visibility conditions. The limitations of static beam headlights, compounded by the Troxler effect, pose significant risks to both pedestrians and vehicles.

This thesis offers a novel, holistic framework that integrates dynamic modeling, predictive control, and intelligent perception for adaptive headlamp systems. The research offers implementable, and scalable techniques to enhance nighttime driving safety by meeting four major goals. The main goal was to emulate a controller derived from a mathematical model for the control of headlamp beams, i.e., for curved road sections.

This was obtained through the development of a high-fidelity dynamic vehicle model with longitudinal, lateral, and yaw dynamics, as well as tire forces, aerodynamics, and braking influence. Accurate computation of the vehicle's slip angle was necessary to estimate the actual trajectory in cornering maneuvers. A Proportional Controller was used first to proportionally relate the slip angle to the headlamp deflection angle, aligning the beam with the vehicle's actual path rather than the theoretical direction of steering. A Filtered Proportional Controller was used to account for actuator delay and provide a smooth beam transition. This filtered strategy reduced beam jitter during rapid maneuvers, ensuring perceived stability and improved alignment with real-time dynamics.

The model was tested over a number of synthetic road profiles and driving speeds, showing stable and responsive control of the headlight direction in accordance with the vehicle's behavior under real-world conditions. The second goal was to suggest a perception algorithm capable of identifying on-road objects and coming traffic in order to dynamically adjust beam intensity and direction and counteract the Troxler effect. This led to the development of LSDNet (Low-light and Small-object Detection Network), a deep learning model optimized for night-time driving conditions.

LSDNet was specifically designed to perform well in low-light environments, able to detect small or subtle objects—like pedestrians, vehicles, and stationary obstacles—that can be overlooked by traditional vision systems. The model was compared with leading object detectors such as SSD, Faster R-CNN, YOLO, and Tiny-YOLO. The results showed LSDNet's improved recall, detection accuracy, and robustness against lighting condition variations. LSDNet enabled accurate, low-latency detection of oncoming vehicles and roadside hazards, enabling the system to adaptively control the beam, reducing glare to other motorists while enhancing drivers' visibility—directly addressing the root causes of the Troxler effect.

The second goal was to integrate the beam steering module and object recognition module into a single system that could optimize headlamp direction and intensity simultaneously. This unification led to the creation of a Fused Controller—a unified architecture that combines FF-MPC (Filtered Feedforward Model Predictive Control), E-MPC (Extended Model Predictive Control), and perceptive outputs from LSDNet. The Fused Controller combines predictive modeling of road geometry through slip angle and steering-based prediction with real-time environmental perception through object detection.

This fusion allows the system to pre-steer beams through bends, modulate intensity based on observed traffic and pedestrian movement, and enable smooth state transitions. The Fused Controller combines dynamic modeling with semantic perception to enable anticipatory and context-aware illumination beyond typical rule-based or sensor-only adaptive systems. The combined system contains a perceptual-control synergy that enables adaptive lighting decisions to be proactive and reactive depending on the situation. The last goal was to measure the performance of the system in terms of efficiency, latency, and accuracy under simulated road test conditions.

An extensive series of tests were performed on a 2-kilometer virtual road with varied curvatures and traffic conditions. The results showed that the FF-MPC and E-MPC controllers outperformed conventional methods in beam alignment accuracy and responsiveness. The Fused Controller improved on these features by adapting to dynamic conditions, like the sudden appearance of oncoming traffic or unpredictable bends, with minimal latency and high dependability.

LSDNet achieved a very high mAP and FPS tradeoff, making real-time inference possible while reliably detecting low-contrast objects all the time. The latency across the full control-perception loop remained below tolerable thresholds for driving applications, ensuring fast and accurate headlamp response. The proposed system demonstrated considerable improvements on all criteria measured against existing adaptive headlamp technology, affirming the feasibility of an intelligent, safety-focused headlamp design. This thesis offers a new and holistic approach to adaptive headlamp design.

It offers a novel integration of dynamic vehicle modeling with deep learning-based perception, leading to a Fused Controller that effectively manages both the direction and intensity of headlamp beams. This cross-disciplinary integration provides a strategic answer to the challenges of night driving, particularly in turbulent or ambiguous environments. The emphasis on real-time reaction, perceptual robustness, and adaptive control ensures that the proposed system improves driver visibility and enhances road safety for all users. This work provides a solid foundation for future developments in intelligent automobile lighting and active safety systems.

5.2 Future Developments

The adaptive headlight system has demonstrated impressive performance in simulations; however, several avenues are available to enhance its practicability, robustness, and readiness for deployment in real-world applications. One of the central area is the real-time operation on vehicle hardware. The control algorithms and LSDNet perception model need to be optimized for deployment on vehicle-grade ECUs with tight limits on power and memory. This involves the application of model compression techniques, such as pruning, quantization, and knowledge distillation, to reduce inference time and resource usage, thereby enabling real-time operation without the need for specialized GPUs.

The next step is in-vehicle testing in real-world environments and evaluating its performance under real driving conditions will expose integration problems like actuator delay, sensor oscillation, and misalignment. Practical tests deliver empirical tuning of controller parameters and collection of driver data, thereby guaranteeing the reliability and responsiveness of the system in dynamic situations. Another area is multi-sensor fusion. LSDNet works well for low-light object detection; however, incorporating it with LiDAR, radar, or thermal sensing might make it more robust in bad weather like rain, fog, or low-contrast conditions. Radar can detect incoming vehicles under low visibility, while thermal

cameras enhance the detection of pedestrians or animals at night. Fusion strategies, such as early or late integration, can be explored to combine the strengths of different modalities for greater object awareness.

Adaptive learning algorithms can further refine beam adaptation on the control side. Techniques like reinforcement learning or fuzzy adaptation would be able to offer personalization based on driver behavior or road conditions. Hybridizing learning-based methods with the model predictive control (MPC) architecture aids in balancing deterministic safety regulations and pragmatic adaptability. The proposed Fused Controller can also evolve through vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. The device can actively alter beams before visual detection by getting data from other cars or roadside units about upcoming curves, obstacles, or traffic congestion. This shared perception can reduce latency and improve decision-making accuracy. A key point is compliance with automobile regulations and industry standards.

Given that adaptive lighting is strictly regulated, it is essential to ensure the system complies with beam cutoff limitations, glare thresholds, and redundancy standards. Subsequent endeavors should encompass formal verification of control logic, implementation of fail-safe defaults, and engagement in standardization initiatives. This will facilitate the process of certification and commercial implementation. Scalability represents a crucial aspect. The system needs to be adaptable over a range of vehicle categories, including two-wheelers, commercial vehicles, and public transport. Every platform has unique dynamics and constraints, but may benefit from improved lighting intelligence. Perceptual and control layers will be modularized to increase adaptability over different platforms and applications. The evolution of automobile technology demands the integration of autonomous driving and ADAS systems as a logical progression.

The intelligent beam control system can be used in combination with lane-keeping, pedestrian detection, and night-time course planning. This creates a wider context in which lighting is an integrated part of a total safety system. This thesis introduces an adaptive headlamp architecture that provides a sound basis for future development in intelligent vehicle lighting. The possibilities for enhancing and extending this work, from real-world implementation to advanced sensor integration, regulatory cooperation, and cross-platform portability, are significant. With the increasing emphasis on vehicle safety and autonomy, these systems are likely to become standard features in the next generation of smart vehicles.

REFERENCES

1. Másilková, M. (2017). Health and social consequences of road traffic accidents. *Kontakt*, 19(1), e43–e47.
2. "Advancing Road Safety in India-Implementation is the Key (Summary)", Nimhans.ac.in, 2017. [Online]. Available: https://nimhans.ac.in/wp-content/uploads/2019/02/UL_BR_m010-11_Main-rprt_FINAL.pdf. [Accessed: 31- May- 2021].
3. Toney, G., & Bhargava, C. (2021). Adaptive headlamps in automobile: A review on the models, detection techniques, and mathematical models. *IEEE Access*, 9, 87462-87474.
4. Uttley, J., & Fotios, S. (2017). The effect of ambient light condition on road traffic collisions involving pedestrians on pedestrian crossings. *Accident Analysis & Prevention*, 108, 189-200.
5. Dahou, H., Gouri, R. E., Mateur, K., Alareqi, M., Zemmouri, A., Mezouari, A., & Hlou, L. (2017). New design of an intelligent system (AFS) of automobile with digital PWM technique on FPGA board. *ARNP Journal of Engineering and Applied Sciences*, 12(3), 672-680.
6. De Santos-Berbel, C., & Castro, M. (2020). Effect of vehicle swiveling headlamps and highway geometric design on nighttime sight distance. *Mathematics and Computers in Simulation*, 170, 32-50.
7. Hacibekir, T., Karaman, S., Kural, E., Ozturk, E. S., Demirci, M., & Guvenc, B. A. (2006, October). Adaptive headlight system design using hardware-in-the-loop simulation. In *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control* (pp. 915-920). IEEE.
8. Li, L., Yang, M., & Zhang, J. (2015). The bending mode control method of AFS system based on preview control. *International journal on smart sensing and intelligent systems*, 8(1), 637.
9. Lee, J. H., Byeon, J., Go, D. J., & Park, J. R. (2017). Automotive adaptive front lighting requiring only on/off modulation of multi-array LEDs. *Current Optics and Photonics*, 1(3), 207-213.
10. Kloppenburg, G., Wolf, A., & Lachmayer, R. (2016). High-resolution vehicle headlamps: technologies and scanning prototype. *Advanced Optical Technologies*, 5(2), 147-155.
11. Tamburo, R., Nurvitadhi, E., Chugh, A., Chen, M., Rowe, A., Kanade, T., & Narasimhan, S. G. (2014). Programmable automotive headlights. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part IV* (Vol. 8692, pp. 750–765). Springer International Publishing.
12. Kumar, A., Bilger, D. P., Richard, J. M., & Edward, G. R. (2016), Mexico Patent No. MX366473B. Mexico.

13. Smith, Russell., Ben Nissan Motor Manufacturers Ltd. (2016), European Patent No. EP1800947A1. European Patent Office. <https://patentimages.storage.googleapis.com/75/1d/d7/955726084c6b82/EP1800947A1.pdf>.
14. "Ford Lighting Technology", <https://www.ford.ie>, 2016. [Online]. Available: <https://www.ford.ie/shop/explore/technology/driving-experience/ford-lighting-technology>. [Accessed: 31- May- 2021].
15. "Ford developing advanced headlights that point out people, animals in the dark, and widen beams at tricky junctions", <https://media.ford.com>, 2015. [Online]. Available: <https://media.ford.com/content/fordmedia/feu/nl/nl/news/2015/07/17/ford-developing-advanced-headlights-that-point-out-people--anima.html>. [Accessed: 31- May- 2021].
16. L. NISSAN MOTOR Co., "Adaptive Front - Lighting System", NISSAN | Technology, 2007. [Online]. Available: <https://www.nissan-global.com/EN/TECHNOLOGY/OVERVIEW/afs.html>. [Accessed: 31- May- 2021].
17. S. Szymkowski, "Improved optional headlights net Volvo XC40, XC60 Top Safety Pick awards", The Car Connection, 2019. [Online]. Available: <https://bit.ly/2R7406q>. [Accessed: 31- May- 2021].
18. S. Szymkowski, "2021 Subaru Forester prices rise a bit, but SUV gets more safety kit", Roadshow, 2021. [Online]. Available: <https://cnet.co/3wKMoMP>. [Accessed: 31- May- 2021].
19. [19] "BMW Adaptive Headlight Malfunction", Youcanic.com, 2020. [Online]. Available: <https://bit.ly/2R9rc47>. [Accessed: 31- May- 2021].
20. Tian, B., Morris, B. T., Tang, M., Liu, Y., Yao, Y., Gou, C., ... & Tang, S. (2014). Hierarchical and networked vehicle surveillance in ITS: a survey. *IEEE transactions on intelligent transportation systems*, 16(2), 557-580.
21. Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 27(10), 1615-1630.
22. Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (SURF). *Computer vision and image understanding*, 110(3), 346-359.
23. Lindeberg, T. (2015). Image matching using generalized scale-space interest points. *Journal of mathematical Imaging and Vision*, 52, 3-36.
24. Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (Vol. 1, pp. 886–893). IEEE

25. Weinberger, K. Q., & Saul, L. K. (2009). Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, 10(2).
26. Abe, S. (2005). Feature selection and extraction. *Support vector machines for pattern classification*, 189-199.
27. Al-Anazi, A. F., & Gates, I. D. (2010). Support vector regression for porosity prediction in a heterogeneous reservoir: A comparative study. *Computers & Geosciences*, 36(12), 1494-1503.
28. Najva, N., & Bijoy, K. E. (2016). SIFT and tensor based object detection and classification in videos using deep neural networks. *Procedia Computer Science*, 93, 351-358.
29. Angulo, C., Parra, X., & Catala, A. (2003). K-SVCR. A support vector machine for multi-class classification. *Neurocomputing*, 55(1-2), 57-77.
30. Moosavian, A., Ahmadi, H., Tabatabaeefar, A., & Khazaei, M. (2013). Comparison of two classifiers; K-nearest neighbor and artificial neural network, for fault diagnosis on a main engine journal-bearing. *Shock and Vibration*, 20(2), 263-272.
31. Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019, May). A brief review of nearest neighbor algorithm for learning and classification. In *2019 international conference on intelligent computing and control systems (ICCS)* (pp. 1255-1260). IEEE.
32. Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1), 119-139.
33. Sharma, V. K., & Mir, R. N. (2020). A comprehensive and systematic look up into deep learning based object detection techniques: A review. *Computer Science Review*, 38, 100301.
34. Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
35. Uijlings, J. R., Van De Sande, K. E., Gevers, T., & Smeulders, A. W. (2013). Selective search for object recognition. *International journal of computer vision*, 104, 154-171.
36. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9), 1904-1916.
37. Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440-1448).

38. Jiao, L., Zhang, F., Liu, F., Yang, S., Li, L., Feng, Z., & Qu, R. (2019). A survey of deep learning-based object detection. *IEEE access*, 7, 128837-128868.
39. Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6), 1137-1149.
40. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
41. Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7263-7271).
42. Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
43. Yi, Z., Yongliang, S., & Jun, Z. (2019). An improved tiny-yolov3 pedestrian detection algorithm. *Optik*, 183, 17-23.
44. Gupta, P., Sharma, V., & Varma, S. (2021). WITHDRAWN: People detection and counting using YOLOv3 and SSD models.
45. Yang, S., Zhang, J., Bo, C., Wang, M., & Chen, L. (2019). Fast vehicle logo detection in complex scenes. *Optics & Laser Technology*, 110, 196-201.
46. Rani, E. (2021). LittleYOLO-SPP: A delicate real-time vehicle detection algorithm. *Optik*, 225, 165818.
47. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14* (pp. 21-37). Springer International Publishing.
48. Bacha, S., Ayad, M. Y., Saadi, R., Aboubou, A., Bahri, M., & Becherif, M. (2017, October). Modeling and control technics for autonomous electric and hybrid vehicles path following. In *2017 5th International Conference on Electrical Engineering-Boumerdes (ICEE-B)* (pp. 1-12). IEEE.
49. Wang, J., Steiber, J., & Surampudi, B. (2009). Autonomous ground vehicle control system for high-speed and safe operation. *International Journal of Vehicle Autonomous Systems*, 7(1-2), 18-35.
50. Ping, E. P., Hudha, K., & Jamaluddin, H. (2010). Hardware-in-the-loop simulation of automatic steering control for lanekeeping manoeuvre: outer-loop and inner-loop control design. *International journal of vehicle safety*, 5(1), 35-59.

51. Elbanhawi, M., Simic, M., & Jazar, R. (2015). The role of path continuity in lateral vehicle control. *Procedia Computer Science*, 60, 1289-1298.
52. Raffo, G. V., Gomes, G. K., Normey-Rico, J. E., Kelber, C. R., & Becker, L. B. (2009). A predictive controller for autonomous vehicle path tracking. *IEEE transactions on intelligent transportation systems*, 10(1), 92-102.
53. De Luca, A., Oriolo, G., & Samson, C. (2005). Feedback control of a nonholonomic car-like robot. *Robot motion planning and control*, 171-253.
54. Kanayama, Y., Kimura, Y., Miyazaki, F., & Noguchi, T. (1990, May). A stable tracking control method for an autonomous mobile robot. In *Proceedings., IEEE International Conference on Robotics and Automation* (pp. 384-389). IEEE.
55. Dugoff, H., Fancher, P. S., & Segel, L. (1970). An analysis of tire traction properties and their influence on vehicle dynamic performance. *SAE transactions*, 1219-1243.
56. Kiencke, U., & Nielsen, L. (2000). Automotive control systems: for engine, driveline, and vehicle.
57. Frazzoli, E., Dahleh, M. A., & Feron, E. (2000, December). Robust hybrid control for autonomous vehicle motion planning. In *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)* (Vol. 1, pp. 821-826). IEEE.
58. Coulter, R. C. (1992). Implementation of the pure pursuit path tracking algorithm. Carnegie-Mellon UNIV Pittsburgh PA Robotics INST.
59. Murphy, K. N. (1994, August). Analysis of robotic vehicle steering and controller delay. In *Fifth International Symposium on Robotics and Manufacturing (ISRAM)* (pp. 631-636).
60. Hellstrom, T., & Ringdahl, O. (2006). Follow the Past: a path-tracking algorithm for autonomous vehicles. *International journal of vehicle autonomous systems*, 4(2-4), 216-224.
61. Gockley, R., Forlizzi, J., & Simmons, R. (2007, March). Natural person-following behavior for social robots. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction* (pp. 17-24).
62. Wit, J., Crane III, C. D., & Armstrong, D. (2004). Autonomous ground vehicle path tracking. *Journal of Robotic Systems*, 21(8), 439-449.
63. Kuwata, Y., Teo, J., Karaman, S., Fiore, G., Frazzoli, E., & How, J. (2008, August). Motion planning in complex environments using closed-loop prediction. In *AIAA Guidance, Navigation and Control Conference and Exhibit* (p. 7166).

64. Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., ... & Mahoney, P. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of field Robotics*, 23(9), 661-692.
65. Falcone, P., Borrelli, F., Asgari, J., Tseng, H. E., & Hrovat, D. (2007). Predictive active steering control for autonomous vehicle systems. *IEEE Transactions on control systems technology*, 15(3), 566-580.
66. Martins, F. N., Celeste, W. C., Carelli, R., Sarcinelli-Filho, M., & Bastos-Filho, T. F. (2008). An adaptive dynamic controller for autonomous mobile robot trajectory tracking. *Control Engineering Practice*, 16(11), 1354-1363.
67. Huang, D., Zhai, J., Ai, W., & Fei, S. (2016). Disturbance observer-based robust control for trajectory tracking of wheeled mobile robots. *Neurocomputing*, 198, 74-79.
68. Bittanti, S., Pavesi, G., Rugarli, M., & Savaresi, S. E. R. G. I. O. (1995). Compensating the tracking-error of a mobile robot by on-line tuning of a neural network. In *Intelligent Autonomous Vehicles 1995* (pp. 271-276). Pergamon.
69. Dørum, J., Utstumo, T., & Gravdahl, J. T. (2015, October). Experimental comparison of adaptive controllers for trajectory tracking in agricultural robotics. In *2015 19th International Conference on System Theory, Control and Computing (ICSTCC)* (pp. 206-212). IEEE.
70. Ahmed, S. A., & Petrov, M. G. (2015). Trajectory control of mobile robots using type-2 fuzzy-neural PID controller. *IFAC-PapersOnLine*, 48(24), 138-143.
71. Al Khatib, E. I., Al-Masri, W. M., Mukhopadhyay, S., Jaradat, M. A., & Abdel-Hafez, M. (2015, December). A comparison of adaptive trajectory tracking controllers for wheeled mobile robots. In *2015 10th international symposium on mechatronics and its applications (ISMA)* (pp. 1-6). IEEE.
72. Caracciolo, L., De Luca, A., & Iannitti, S. (1999, May). Trajectory tracking control of a four-wheel differentially driven mobile robot. In *Proceedings 1999 IEEE international conference on robotics and automation (Cat. No. 99CH36288C)* (Vol. 4, pp. 2632-2638). IEEE.
73. Polack, P., Altché, F., d'Andréa-Novel, B., & de La Fortelle, A. (2017, June). The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?. In *2017 IEEE intelligent vehicles symposium (IV)* (pp. 812-818). IEEE.
74. Peng, Y. F., Hsu, C. F., Lin, C. M., & Lee, T. T. (2006, October). Robust intelligent backstepping longitudinal control of vehicle platoons with H/spl infin/tracking

- performance. In *2006 IEEE International Conference on Systems, Man and Cybernetics* (Vol. 6, pp. 4648-4653). IEEE.
75. Hu, C., Jing, H., Wang, R., Yan, F., & Chadli, M. (2016). Robust H_∞ output-feedback control for path following of autonomous ground vehicles. *Mechanical Systems and Signal Processing*, 70, 414-427.
 76. Rasib, M., Butt, M. A., Riaz, F., Sulaiman, A., & Akram, M. (2021). Pixel level segmentation based drivable road region detection and steering angle estimation method for autonomous driving on unstructured roads. *IEEE Access*, 9, 167855-167867.
 77. Shi, W., Alawieh, M. B., Li, X., & Yu, H. (2017). Algorithm and hardware implementation for visual perception system in autonomous vehicle: A survey. *Integration*, 59, 148-156.
 78. Zhang, S., Benenson, R., & Schiele, B. (2015, June). Filtered channel features for pedestrian detection. In *CVPR* (Vol. 1, No. 2, p. 4).
 79. Zhang, L., Lin, L., Liang, X., & He, K. (2016). Is faster R-CNN doing well for pedestrian detection?. In *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14* (pp. 443-457). Springer International Publishing.
 80. Kim, Z. (2008). Robust lane detection and tracking in challenging scenarios. *IEEE Transactions on intelligent transportation systems*, 9(1), 16-26.
 81. Gopalan, R., Hong, T., Shneier, M., & Chellappa, R. (2012). A learning approach towards detection and tracking of lane markings. *IEEE Transactions on Intelligent Transportation Systems*, 13(3), 1088-1098.
 82. Alvarez, J. M., LeCun, Y., Gevers, T., & Lopez, A. M. (2012, October). Semantic road segmentation via multi-scale ensembles of learned features. In *European Conference on Computer Vision* (pp. 586-595). Berlin, Heidelberg: Springer Berlin Heidelberg.
 83. Yao, J., Ramalingam, S., Taguchi, Y., Miki, Y., & Urtasun, R. (2015, January). Estimating drivable collision-free space from monocular video. In *2015 IEEE Winter Conference on Applications of Computer Vision* (pp. 420-427). IEEE.
 84. Park, J., & Jayachandran Raguraman, S. (2020). Intelligent Drivable Area Detection System Using Camera and Lidar Sensor for Autonomous Vehicle.
 85. Gu, Y., Si, B., & Liu, B. (2021). A novel hierarchical model in ensemble environment for road detection application. *Remote Sensing*, 13(6), 1213.
 86. Qi, N., Yang, X., Li, C., Lu, R., He, C., & Cao, L. (2019). Unstructured road detection via combining the model-based and feature-based methods. *IET Intelligent Transport Systems*, 13(10), 1533-1544.
 87. Wang, E., Sun, A., Li, Y., Hou, X., & Zhu, Y. (2018). Fast vanishing point detection method based on road border region estimation. *IET Image Processing*, 12(3), 361-373.
 88. Andrade, D. C., Bueno, F., Franco, F. R., Silva, R. A., Neme, J. H. Z., Margraf, E., ... & dos Santos Amaral, R. (2018). A novel strategy for road lane detection and tracking

- based on a vehicle's forward monocular camera. *IEEE Transactions on Intelligent Transportation Systems*, 20(4), 1497-1507.
89. Sun, J. Y., Jung, S. W., & Ko, S. J. (2020). Lightweight prediction and boundary attention-based semantic segmentation for road scene understanding. *IEEE Access*, 8, 108449-108460.
 90. Sun, Y., Zuo, W., & Liu, M. (2020). See the future: A semantic segmentation network predicting ego-vehicle trajectory with a single monocular camera. *IEEE Robotics and Automation Letters*, 5(2), 3066-3073.
 91. Sun, Y., Wang, L., Chen, Y., & Liu, M. (2019, December). Accurate lane detection with atrous convolution and spatial pyramid pooling for autonomous driving. In *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (pp. 642-647). IEEE.
 92. Gillespie, T. (Ed.). (2021). *Fundamentals of vehicle dynamics*. SAE international.
 93. Slip Angle Explained, VBOX equipment, UK, Assessed on July 19, 2023. http://www.racelogic.co.uk/_downloads/vbox/Application_Notes/Slip%20Angle%20Explained.pdf
 94. Toney, G., Sethi, G., & Bhargava, C. (2025). Optimal Headlamp Adjustment for Vehicles through Slip Angle and Stiffness Analysis using Dynamic Vehicle Model. *Engineered Science*, 34, 34.
 95. Takikawa, K., Atsumi, Y., Takanose, A., & Meguro, J. (2021). Vehicular trajectory estimation utilizing slip angle based on GNSS Doppler/IMU. *Robomech Journal*, 8, 1-11.
 96. Singh, K. B. (2019). Vehicle sideslip angle estimation based on tire model adaptation. *electronics*, 8(2), 199.
 97. Selmanaj, D., Corno, M., Panzani, G., & Savaresi, S. M. (2017). Vehicle sideslip estimation: A kinematic based approach. *Control Engineering Practice*, 67, 1-12.
 98. Du, H., Lam, J., Cheung, K. C., Li, W., & Zhang, N. (2015). Side-slip angle estimation and stability control for a vehicle with a non-linear tyre model and a varying speed. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 229(4), 486-505.
 99. Pi, D. W., Chen, N., Wang, J. X., & Zhang, B. J. (2011). Design and evaluation of sideslip angle observer for vehicle stability control. *International Journal of Automotive Technology*, 12, 391-399.
 100. Raksincharoensak, P., Daisuke, S., & Lidberg, M. (2019). Direct yaw moment control for enhancing handling quality of lightweight electric vehicles with large load-to-curb weight ratio. *Applied Sciences*, 9(6), 1151.
 101. Shibahata, Y., Shimada, K., & Tomari, T. (1993). Improvement of vehicle maneuverability by direct yaw moment control. *Vehicle System Dynamics*, 22(5-6), 465-481.
 102. Chen, B. C., & Hsieh, F. C. (2008). Sideslip angle estimation using extended Kalman filter. *Vehicle System Dynamics*, 46(S1), 353-364.

103. Piyabongkarn, D., Rajamani, R., Grogg, J. A., & Lew, J. Y. (2008). Development and experimental evaluation of a slip angle estimator for vehicle stability control. *IEEE Transactions on control systems technology*, 17(1), 78-88.
104. Liu, J., Wang, Z., Zhang, L., & Walker, P. (2020). Sideslip angle estimation of ground vehicles: a comparative study. *IET Control Theory & Applications*, 14(20), 3490-3505.
105. Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). YOLOv4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
106. Toney, G., Sethi, G., Bhargava, C., & Salian, V. (2024). Modified SSD Framework for On-Road Object Detection. In *Intelligent Circuits and Systems for SDG 3—Good Health and well-being* (pp. 331-340). CRC Press.
107. Toney, G., Sethi, G., Bhargava, C., Vaz, A. C., & Hegde, N. T. (2025). Sensor fusion and predictive control for adaptive vehicle headlamp alignment: A comparative analysis. *Journal of Robot Control (JRC)*, 6(5), 2166–2183.
108. Toney, G., Sethi, G., & Bhargava, C. (2024). A novel lane detection approach for vehicle adaptive headlamps. In *Proceedings of the International Conference on Advances in Smart Sensor, Signal Processing and Communication Technology (ICASSCT 2024)*, Goa University, India.

PUBLICATION DETAILS

S . No	Type of Paper	Name of the Journal/Conference/Book	Journal indexing	Title of the Paper	Published Date (Date/Month/Year)	Volume & Issue Number	ISSN/ISBN Number	Impact Factor/SJR	Type of paper (Research/Review)	Web link of journal indexing
1	Journal	Engineered Science	Scopus	Optimal Headlamp Adjustment for Vehicles through Slip Angle and Stiffness Analysis using Dynamic Vehicle Model	25-Jan-25	34	10.30919/es1388	SJR: 0.87	Research	https://www.scopus.com/sourceid/21101039622
2	Journal	Journal of Robotics & Control	Scopus	Sensor fusion and predictive control for adaptive vehicle headlamp alignment: A comparative analysis.	29-Aug-25	6	2715-5056	SJR-0.435	Research	https://www.scimagojr.com/journalsearch.php?q=21101058819&tip=sid&clean=0
3	Journal	IEEE Access	Scopus & Web of Science	Adaptive headlamps in automobile: A review on the models, detection techniques, and mathematical models	09-Jun-21	9	10.1109/ACCESS.2021.3088036	SJR: 0.96	Review	https://www.scopus.com/sourceid/21100374601

3	Conference/ Book Chapter	Intelligent Circuits and Systems for SDG 3 – Good Health and well-being: International Conference on Intelligent Circuits and Systems (ICICS 2023)	NA	Modified SSD Framework for On-Road Object Detection	05-Aug -24	1	97810 03521 716	NA	Research	https://www.taylorfrancis.com/books/edit/10.1201/9781003521716/intelligent-circuits-systems-sdg-3-good-health-well-being-bhaveshkumar-chioithram-dharmann-suman-lata-tripathi
4	Conference	International Conference on Advances in Smart Sensor, Signal Processing and Communication Technology(ICAS SCT 2024)	NA	A Novel Lane Detection Approach for Vehicle Adaptive headlamps	22 – 23, March 2024	NA	NA	NA	Research	NA



LOVELY
PROFESSIONAL
UNIVERSITY

Transforming Education Transforming India



Certificate No. 294701

Certificate of Paper Presentation

This is to certify that **Prof./Dr./Mr./Ms. Glenson Toney** of **Lovely Professional University, Punjab, India** has presented the paper titled **Modified SSD Framework for On-Road Object Detection** in the “**5th International Conference on Intelligent Circuits and Systems (ICICS-2023)**” held on **October 12-13th, 2023**, organized by School of Electronics and Electrical Engineering at Lovely Professional University, Punjab.

Date of Issue : 30-10-2023
Place : Phagwara (Punjab), India

Prepared by
(Administrative Officer-Records)

Dr. Bhaveshkumar C. Dharmani
Organizing Secretary
(ICICS-2023)

Dr. Charanjit Singh
Convenor
(ICICS-2023)

Dr. Gaurav Sethi
Conference Chair
(ICICS-2023)



2024 Second International Conference on
**Advances in Smart Sensor, Signal Processing
and Communication Technology (ICASSCT 2024)**

22 - 23, March 2024 | Goa University, Goa, India

Sponsor



विज्ञान एवं प्रौद्योगिकी विभाग
DEPARTMENT OF
SCIENCE & TECHNOLOGY

Publication Partner



SCT 5054

CERTIFICATE

This certificate is presented to

Glenson Toney, Gaurav Sethi, Cherry Bhargava

for presenting the research paper entitled "A Novel Lane Detection Approach for Vehicle Adaptive headlamps" in the Second International Conference on Advances in Smart Sensor, Signal Processing and Communication Technology (ICASSCT 2024) organized by the Electronics Programme, School of Physical and Applied Sciences, Goa University, Goa, India during 22 - 23, March 2024.

Prof. Rajendra S. Gad
Convener

Prof. Jivan S. Parab
Co-ordinator

Prof. Ramesh Pai
Dean, SPAS



Annexure A1

Comparison of Object Detection Models and Vehicle Models for Adaptive Headlamps from Literature Review

This annexure presents six illustrative figures developed as part of the literature review for the thesis. The figures are intended to provide a comparative visualization of techniques discussed as part of the literature review. The charts are based on normalized and qualitative approximations derived from reported trends in peer-reviewed literature. They are not experimental data, but conceptual visualizations meant to highlight the evolution of detection methods, learning frameworks, and vehicle models used in adaptive headlamp systems.

A1-1 Figure 2.3 – Performance Comparison of Traditional Object Detection Features

A1-2 Figure 2.4 – Comparison of ML Classifiers for Vehicle Detection

A1-3 Figure 2.5 – Accuracy vs Inference Time of Deep Learning Detection Models

A1-4 Figure 2.7 – Comparative Analysis of Vehicle Path Models

A1-5 Figure 2.8 – Model Stability and Parameter Dependence

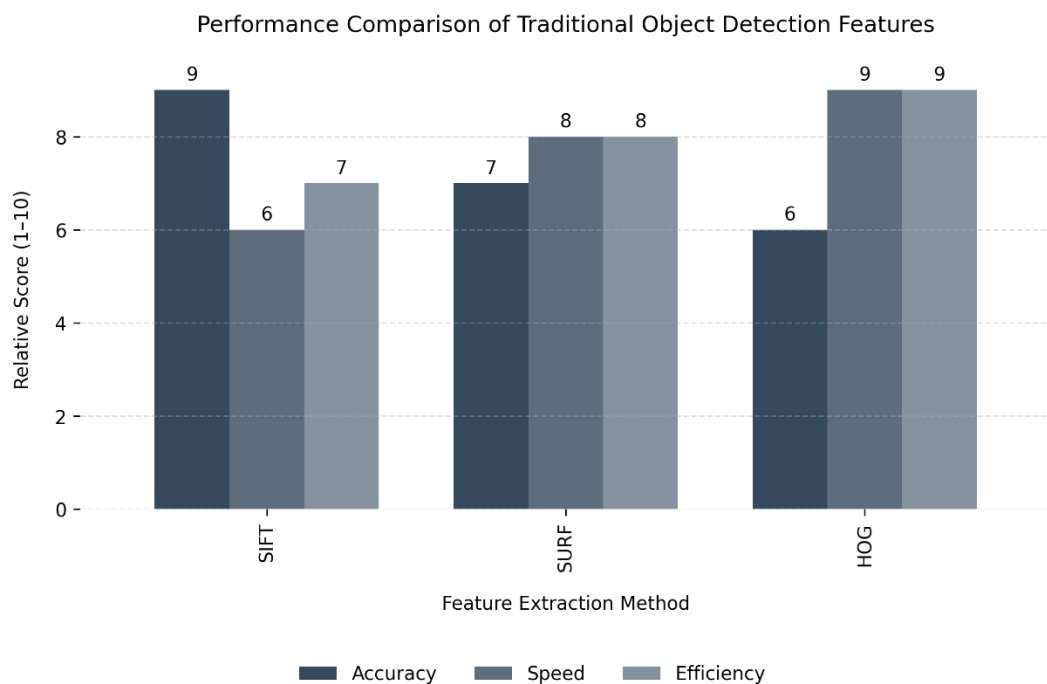
A1-6 Figure 2.9 – Trend of Model Evolution for Adaptive Headlamp Systems

A1-1 Figure 2.3 – Performance Comparison of Traditional Object Detection Features

Objective:

To compare the performance of SIFT, SURF, and HOG feature extractors based on relative accuracy, speed, and computational efficiency.

Feature Extractor	Accuracy	Speed	Efficiency	Rationale
SIFT	9	6	7	Highly robust and accurate but computationally heavy.
SURF	7	8	8	Balanced speed and accuracy, improved performance over SIFT.
HOG	6	9	9	Fastest and lightweight, lower discriminative power.



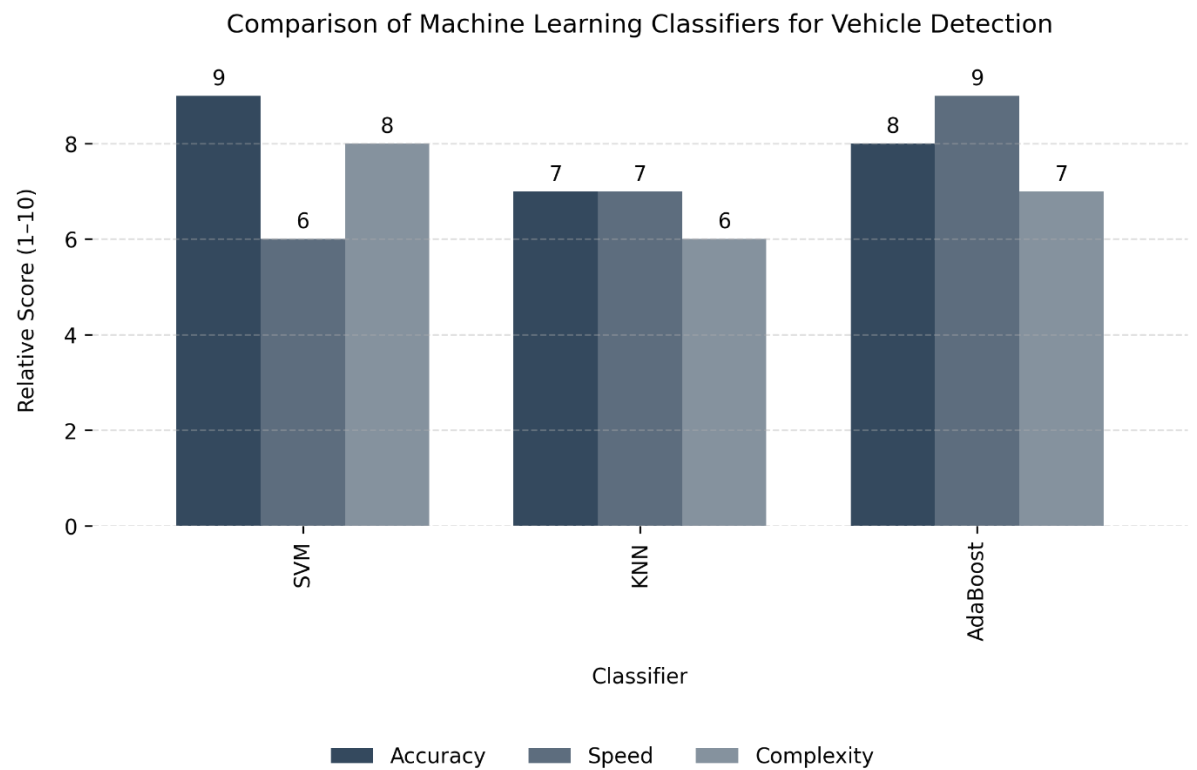
The figure shows the trade-off between accuracy and speed in traditional handcrafted features. SIFT offers superior robustness at the cost of computation, whereas HOG provides rapid feature extraction suitable for embedded use.

A1-2 Figure 2.4 – Comparison of ML Classifiers for Vehicle Detection

Objective:

To illustrate the comparative performance of SVM, KNN, and AdaBoost classifiers based on qualitative assessment of accuracy, speed, and model complexity.

Classifier	Accuracy	Speed	Complexity	Description
SVM	9	6	8	High accuracy, computationally expensive.
KNN	7	7	6	Simple but slow at inference.
AdaBoost	8	9	7	Balanced ensemble with good speed and robustness.



AdaBoost exhibits better efficiency than SVM while retaining strong accuracy. These models bridge the transition from static feature-based systems to learning-based frameworks.

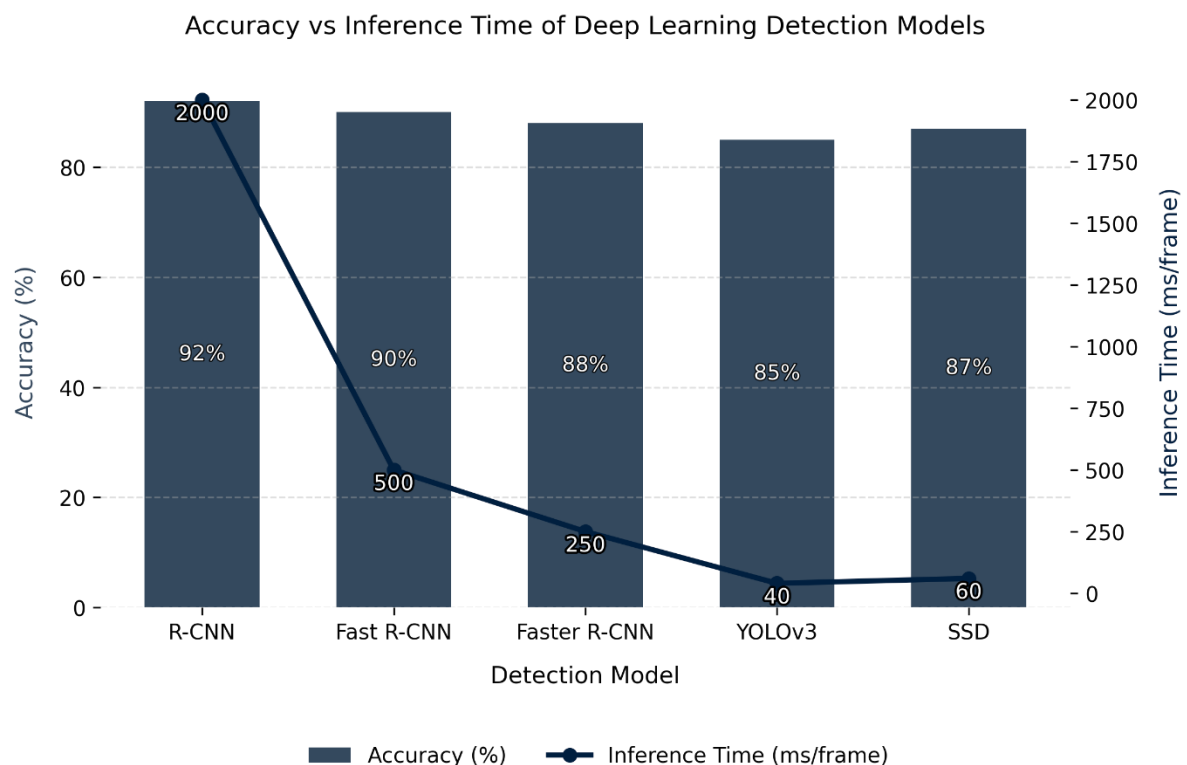
A1-3 Figure 2.5 – Accuracy vs Inference Time of Deep Learning Detection Models

Objective:

To compare major deep-learning-based object detection architectures used in adaptive headlamp systems.

Model	Accuracy (%)	Inference Time (ms/frame)*	Characteristics
R-CNN	92	2000	High accuracy, very slow (region proposals).
Fast R-CNN	90	500	Faster, single-stage training.
Faster R-CNN	88	250	Uses region proposal network for speed.
YOLOv3	85	40	Real-time detection with regression approach.
SSD	87	60	Slightly slower than YOLO, comparable accuracy.

**Approximate values based on reported benchmarks in cited literature.*



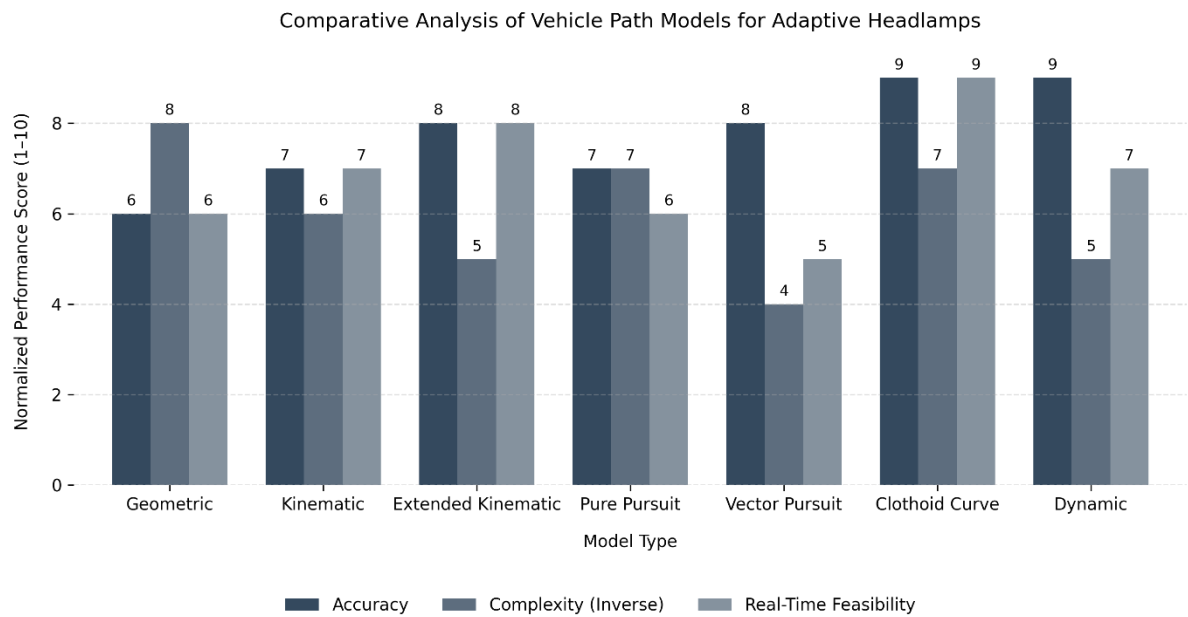
A clear inverse relationship exists between accuracy and inference time. R-CNN-based models offer superior precision but are unsuitable for real-time headlamp control. YOLO and SSD strike an optimal balance for embedded automotive implementation.

A1-4 Figure 2.7 – Comparative Analysis of Vehicle Path Models

Objective:

To compare geometric, kinematic, and dynamic vehicle models used for path prediction in adaptive headlamp systems.

Model	Accuracy	Complexity (Inverse)	Real-Time Feasibility	Description
Geometric	6	8	6	Basic, low computational cost.
Kinematic	7	6	7	Includes velocity and heading angle.
Extended Kinematic	8	5	8	Adds dynamic constraints and correction terms.
Pure Pursuit	7	7	6	Target-based path tracking.
Vector Pursuit	8	4	5	Suitable for complex curvature tracking.
Clothoid Curve	9	7	9	High precision for smooth curves.
Dynamic	9	5	7	Realistic, physics-based but computationally heavy.



The figure highlights how models evolve from computational simplicity to high-fidelity realism. The Clothoid and Dynamic models achieve superior accuracy but require greater computational resources.

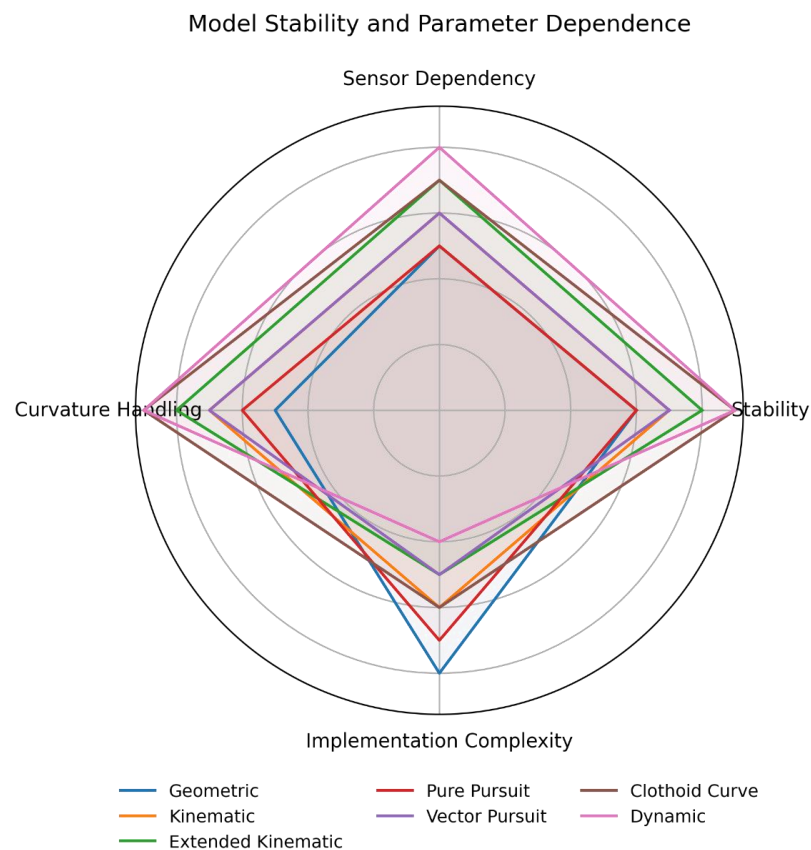
A1-5 Figure 2.8 – Model Stability and Parameter Dependence

Objective:

To compare the models based on stability, sensor dependence, curvature handling, and implementation complexity.

Qualitative Scale (1–10):

Model	Stability	Sensor Dependency	Curvature Handling	Complexity
Geometric	6	5	5	8
Kinematic	7	6	7	6
Extended Kinematic	8	7	8	5
Pure Pursuit	6	5	6	7
Vector Pursuit	7	6	7	5
Clothoid Curve	9	7	9	6
Dynamic	9	8	9	4



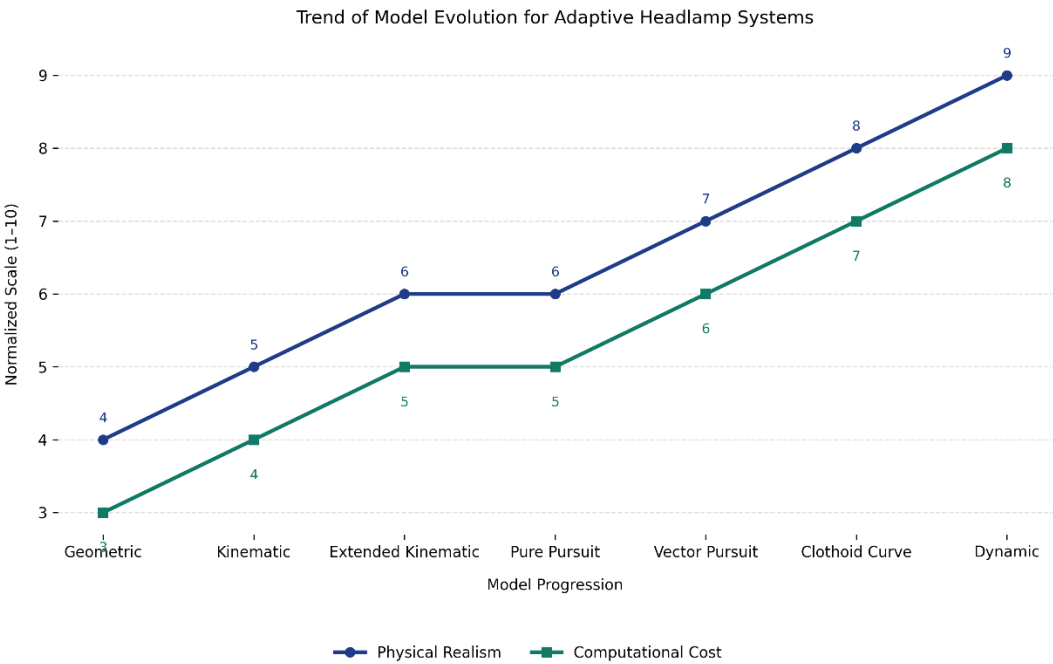
Extended kinematic and clothoid-based models provide optimal trade-offs between curvature handling and complexity, while dynamic models score highest on realism but demand additional sensory and computational support.

A1-6 Figure 2.9 – Trend of Model Evolution for Adaptive Headlamp Systems

Objective:

To depict the gradual evolution of vehicle path modeling techniques in adaptive headlamp research.

Model	Physical Realism	Computational Cost
Geometric	4	3
Kinematic	5	4
Extended Kinematic	6	5
Pure Pursuit	6	5
Vector Pursuit	7	6
Clothoid Curve	8	7
Dynamic	9	8



The figure shows an ascending progression in physical realism with each generation of models, indicating a steady shift from simple geometric assumptions toward integrated dynamic systems capable of handling tire slip, yaw rate, and steering nonlinearities.

All quantitative values in the figures represent normalized qualitative trends synthesized from descriptive analyses and corroborated by benchmark literature. They serve to visually illustrate technological progressions and comparative behavior among algorithms and models used in adaptive headlamp research. These approximations are meant to strengthen conceptual understanding within the literature review, not to replace empirical validation.