**Final Report  
The vision processing module of the autonomous vehicle**

2025.4

**Contents**

[1. Goal 3](#_Toc31444)

[2. Design Process 4](#_Toc30250)

[2.1. Lane Detection Process 4](#_Toc7603)

[Table 1. Relationship between Brightness and Threshold Settings under HDR 5](#_Toc28715)

[2.2. Needs-metrics Matrix 6](#_Toc7289)

[Table 2. Needs–Metrics Matrix 6](#_Toc28571)

[2.3. Engineering Requirements 7](#_Toc25731)

[Table 3. Engineering Requirements 7](#_Toc6030)

[2.4. Project Learning 8](#_Toc29464)

[2.4.1. Identification of other methods 8](#_Toc17576)

[2.4.2. Reason for chosen method 8](#_Toc7387)

[2.4.3. How project learning is related to lifelong learning 8](#_Toc336)

[3. Design analysis 9](#_Toc20923)

[3.1. Engineering Requirements Analysis 9](#_Toc22601)

[3.1.1. Compliance with Requirements 9](#_Toc18847)

[3.1.2. Proof 9](#_Toc24911)

[Table 4. Frame Processing Time 9](#_Toc10006)

[3.1.3. Further Works 10](#_Toc32622)

[3.2. Design Integration 10](#_Toc18638)

[3.3. Costing 10](#_Toc18213)

[Table 5. Cost Calculations 10](#_Toc1967)

[3.4. FMEA Discussion 10](#_Toc29334)

[Table 6. Failure Modes and Effects Analysis (FMEA) 10](#_Toc20159)

[3.5. Diagnostics Tree 11](#_Toc15447)

[4. Project Management 12](#_Toc9026)

[4.1. Team and subsystem organization 12](#_Toc32157)

[4.2. Timeline for the subsystem 13](#_Toc9899)

[Table 7. Timeline 13](#_Toc4816)

[5. Conclusion 14](#_Toc10595)

[5.1. What Went Well 14](#_Toc25045)

[5.2. Challenges and Limitations 14](#_Toc23660)

[5.3. Future Improvement Plan 15](#_Toc30606)

[Reference 16](#_Toc24933)

1. **Goal**
   1. **Project Introduction**  
      This project involves the design and development of an autonomous ground vehicle to participate in the IGVC 2025 competition. The objective is to enable the vehicle to navigate complex outdoor environments by relying on various sensors, including vision, LiDAR, and GPS, to perform autonomous mapping, path planning, and obstacle avoidance tasks.
   2. **Subsystem Relationship**  
      I was responsible for the vision processing subsystem. The vehicle's vision system is based on the Multisense S7 stereo camera, which captures grayscale images of the environment. Through multiple algorithms, the system performs white line detection to provide high-precision environmental perception data for the overall navigation system.
   3. **My work in the subsystem**  
      Within the vision subsystem, my main tasks included:
2. Configuring and optimizing the Multisense S7 camera parameters (such as resolution and frame rate) to suit the competition environment.
3. Building and debugging the line detection system to enable the vehicle to recognize lane markings in simulated outdoor environments.
4. Enhancing the speed and accuracy of vision processing, including filtering and generating occupancy maps.
5. Analyzing and processing the camera output to ensure that the perception results meet the requirements of subsequent navigation and path planning modules.
   1. **My work in system integration**  
      In terms of system integration, I was involved in the data fusion and debugging between the vision subsystem and other subsystems, as well as work related to the vehicle’s power system. My contributions included:
6. Ensuring that the vision detection results could be effectively fused with Li-DAR data for improved environmental perception.
7. Testing and maintaining the vehicle’s battery system, including health checks and battery reconditioning.
8. Participating in the design and assembly of the emergency stop button system to enhance overall vehicle safety.
9. **Design Process**
   1. ****Lane Detection Process****

**1. System Initialization and Camera Calibration**  
A ROS 2 node named **test\_node** is created to subscribe to the /multisense/left/image\_rect topic, receiving rectified grayscale images at 1024×544 resolution. Prior to runtime, a standard checkerboard calibration procedure is performed: multiple images of a printed checkerboard are captured at different angles, and MATLAB’s calibration tools (detectCheckerboardPoints and estimateCameraParameters) are used to compute the camera’s intrinsic parameters (focal lengths, principal point, distortion coefficients) and extrinsic poses. Incoming images are undistorted with undistortImage to ensure accurate geometric measurements.

**2. Image Pre‑Processing**  
Each received frame undergoes bilateral filtering (imbilatfilt) to remove sensor noise while preserving edge details. The filter parameters are tuned to balance noise reduction and edge sharpness under varying lighting conditions.

**3. Dynamic Thresholding**  
The average brightness of the undistorted grayscale image is computed. Based on this brightness, a dynamic threshold value is selected via a simple linear lookup: very dark scenes use a low fixed threshold, very bright scenes use a high fixed threshold, and intermediate lighting interpolates between them. Applying this threshold produces a binary mask highlighting potential lane pixels.

Table 1. Relationship between Brightness and Threshold Settings under HDR

|  |  |  |  |
| --- | --- | --- | --- |
| **Average Brightness** | **Threshold** | **Average Brightness** | **Threshold** |
| 138 | 155 | 57 | 68 |
| 134 | 146 | 51 | 61 |
| 129 | 142 | 47 | 57 |
| 123 | 137 | 43 | 53 |
| 116 | 130 | 39 | 48 |
| 108 | 118 | 35 | 42 |
| 102 | 110 | 31 | 36 |
| 94 | 103 | 27 | 31 |
| 77 | 90 | 23 | 26 |
| 72 | 84 | 19 | 22 |
| 68 | 79 | 16 | 19 |
| 64 | 75 | 13 | 17 |

By fitting a curve to the data in the table 1and comparing a first‑order (linear) fit with a second‑order (quadratic) fit, we found no significant difference in performance. Therefore, our final implementation uses the quadratic model:

**4. Morphological Refinement**  
The binary mask is processed with morphological operations to improve continuity of the lane markings. First, a small rectangular structuring element is used for erosion to remove isolated noise pixels; then morphological reconstruction preserves connected components; finally, a larger structuring element dilates the result to bridge small gaps along the lane lines. This yields a clean, continuous representation of the lane edges.

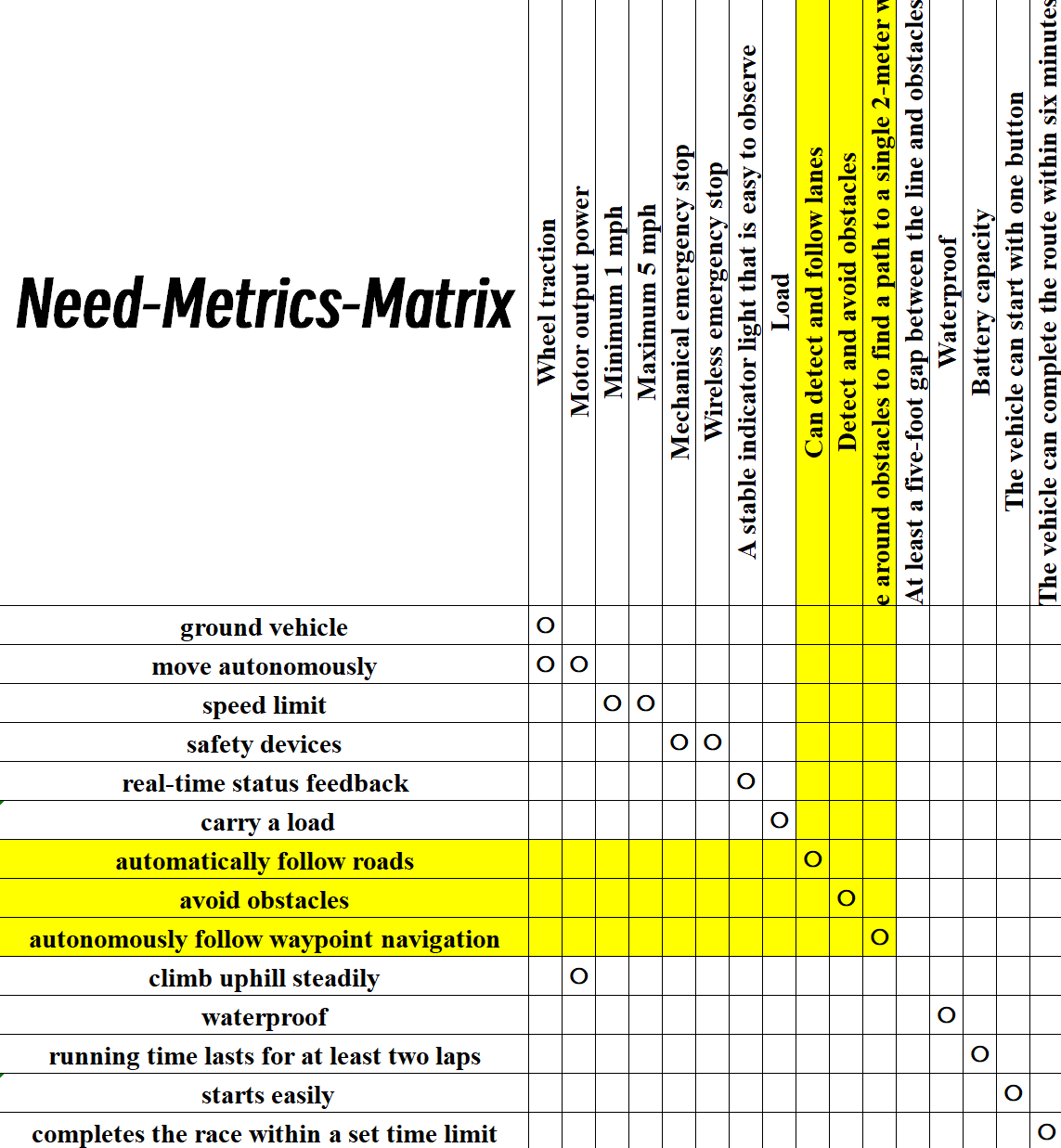
**5. Overlay Visualization**  
The refined binary mask is inverted and overlaid on the original image by marking detected lane areas in bright red. This color overlay provides immediate visual feedback during development and aids in qualitative evaluation of detection performance.

**6. Occupancy Map Generation**  
Using the calibrated camera parameters and the last-known extrinsic pose, a homography (planar projective transform) is constructed to map image pixels on the ground plane to real‑world coordinates. The red overlay mask is thresholded and warped through this transform into an occupancy grid with a fixed spatial resolution (e.g. 0.05 m per cell). The resulting binary occupancy map indicates lane‑occupied cells versus free ground.

**7. Real‑Time Performance Monitoring**  
A 2×4 subplot layout displays, for each incoming frame: the undistorted source image, filtered image, binary threshold result, morphologically refined mask, color overlay, occupancy map, and a text panel listing per‑stage processing times and threshold values. All timing measurements (image reception, filtering, thresholding, morphology, projection) are recorded to identify bottlenecks and optimize processing speed.

* 1. **Needs-metrics Matrix**

Table 2. Needs–Metrics Matrix



* 1. **Engineering Requirements**

Table 3. Engineering Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric no.** | **Metrics** | **Units** | **Marginal value** | **Ideal value** |
| 1 | Wheel traction | inches | 24~28 | 26 |
| 2 | Motor output power | kW | 50 | 80 |
| 3 | Minimum 1 mph | mph | 1.5~2 | 2 |
| 4 | Maximum 5 mph | mph | 4~4.5 | 4 |
| 5 | Mechanical emergency stop | bool | 95 % | 99 % |
| 6 | Wireless emergency stop | bool | 95 % | 99 % |
| 7 | A stable indicator light that is easy to observe | bool | 95 % | 99 % |
| 8 | Load | pound | 20~60 | 40 |
| 9 | Can detect and follow lanes | rate | 99 % | 100 % |
| 10 | Detect and avoid obstacles | inches | 15 | 25 |
| 11 | Navigate around obstacles to find a path to a single 2‑meter waypoint | inches | 15 | 25 |
| 12 | Gap between the line and obstacles | foot | 5~10 | 8 |
| 13 | Waterproof | level | IP54 | IP65 |
| 14 | Battery capacity | Wh | 60 | 100 |
| 15 | The vehicle can start with one button | bool | 95 % | 99 % |
| 16 | The vehicle can complete the route within six minutes | minutes | 3~6 | 4.5 |

* 1. **Project Learning**
     1. **Identification of other methods**

In addition to the pipeline we ultimately adopted, we also experimented with two alternative approaches: Edge Detection combined with the Hough Transform, and Fixed Global Thresholding. We furthermore recognize that other methods—such as Temporal Filtering (e.g., Kalman or Particle Filters) and Machine‑Learning/Deep‑Learning‑Based Semantic Segmentation—could be used to achieve the same objectives.

* + 1. **Reason for chosen method**

We initially adopted the Canny edge detector combined with the Hough transform, but found its overall processing speed to be only around 2 Hz—insufficient for our real‑time requirements. Moreover, the detected lane markings were highly unstable and easily disturbed by noise. The Fixed Global Thresholding approach also failed to meet our real‑time needs: sudden brightness changes, such as a cloud passing over the sun, made stable threshold adjustment impossible. Consequently, both methods were discarded.

* + 1. **How project learning is related to lifelong learning**
* Iterative Experimentation & Reflection  
  After each experiment (linear vs. polynomial thresholding, various structuring elements, occupancy‑map resolutions), we analyzed the results and refined our approach, embodying the hypothesis→experiment→improvement cycle.
* Cross‑Disciplinary Skill Growth  
  We combined computer vision, camera calibration, ROS 2 networking, and real‑time profiling—rapidly acquiring and integrating new technical skills.
* Adaptation & Continuous Improvement  
  Each tuning cycle taught us to reconsider our tools and workflow, instilling a habit of continuous learning and improvement when facing new challenges.
* Foundation for Future Enhancements  
  The modular pipeline positions us to seamlessly incorporate advanced methods (e.g., learning‑based segmentation) later, exemplifying lifelong learning in engineering.

1. **Design analysis**
   1. **Engineering Requirements Analysis**
      1. **Compliance with Requirements**

Requirement 9: Can detect and follow lanes

Threshold: 95 % | Target: 100 %

Test results: In a simulated environment, the system ran continuously for two hours with a packet loss rate below 0.1% and an overall success rate above 99.5%.

Requirement 10: Detect and avoid obstacles

Threshold Distance: 15 inches | Target: 25 inches

Test results: In multiple trials, pits simulated with white paper were placed at various locations on the roadway, and each time they were accurately detected.

* + 1. **Proof**

Table 4. Frame Processing Time

|  |  |
| --- | --- |
| **Procedure** | **Frame Processing Time** |
| **Receive** | 0.067s |
| **Bilateral Filtering** | 0.003s |
| **Threshold** | 0.001s |
| **Morphological** | 0.003s |
| **Total** | 0.074s |

* + 1. **Further Works**

The next step will focus on optimizing the end-to-end processing speed.

* 1. **Design Integration**

The vision processing module, as a crucial input component for the autonomous navigation vehicle, needs to be well-prepared for future fusion with LiDAR data to serve as input for the navigation algorithms. We adopted a method of generating occupancy grids in the same format as the LiDAR output to facilitate the integration of the two input sources. Additionally, we developed the entire system using MATLAB to remain consistent with the LiDAR and algorithm development platforms, ensuring smooth synchronization in future work.

* 1. **Costing**

Table 5. Cost Calculations

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Qty** | **Unit Cost (USD)** | **Total Cost (USD)** |
| MultiSense S7 Stereo Camera | 1 | 1,900 | 1,900 |
| MATLAB r2024b & Toolboxes (Academic License) | / | 500 | 500 |
| Simply NUC 11 (shared device) | 1 | 899 | 899 |
| Ethernet Cable | 1 | 10 | 10 |
| Labor | 100 | 20 | 2,000 |
| Grand Total |  |  | 5,309 |

* 1. **FMEA Discussion**

Table 6. Failure Modes and Effects Analysis (FMEA)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Function / Subsystem** | **Failure Mode** | **Failure Effect** | **S** | **O** | **D** | **RPN** |
| 1 | Image Acquisition | Camera disconnect or frame loss | No image input, system shutdown | 10 | 3 | 1 | 30 |
| 2 | Pre‑processing | Over‑filtering by denoising algorithm | Loss of detail, mis‑detection of lane or pit | 3 | 2 | 8 | 48 |
| 3 | Lane Detection | Failure to detect lanes under low light/shadows | Lane tracking failure, navigation deviation | 10 | 6 | 2 | 120 |
| 4 | Obstacle Detection | White‑paper pits confused with background | Missed or false pit detection, collision risk | 8 | 3 | 6 | 144 |
| 5 | Occupancy Grid Generation | Data format mismatch or grid misalignment | Navigation algorithm receives wrong map, path planning fails | 8 | 3 | 3 | 72 |
| 6 | Data Output / Communication | Serial/Ethernet congestion or disconnect | Downstream module timeout, overall navigation failure | 10 | 4 | 3 | 120 |

Severity (S): The extent of harm a failure effect would cause to the system, rated from 1 (least severe) to 10 (most severe).

Occurrence (O): The likelihood of the failure mode occurring, rated from 1 (very unlikely) to 10 (very likely).

Detection (D): The difficulty of existing controls (tests or monitoring) to detect the failure mode before it reaches the customer, rated from 1 (very easy to detect) to 10 (very difficult to detect).

RPN (Risk Priority Number): Calculated as . The higher the RPN, the higher the priority for corrective action.

* 1. **Diagnostics Tree**

Lane Detection Failure

├── Image Acquisition

│ ├── Camera Disconnect

│ ├── Frame Loss/Latency

│ └── Lens Obstruction

├── Pre‑processing

│ ├── Over‑filtering

│ ├── WB/Exposure Error

│ └── Distortion Correction Fail

├── Detection Algorithm

│ ├── Over‑strict Parameters

│ ├── Under‑trained Model

│ └── Logic Bug

├── Environmental Factors

│ ├── Low‑light / Backlight

│ ├── Shadow Occlusion

│ └── Rain/Snow / Glare

├── Output & Fusion

│ ├── Format Mismatch

│ ├── Sync Delay

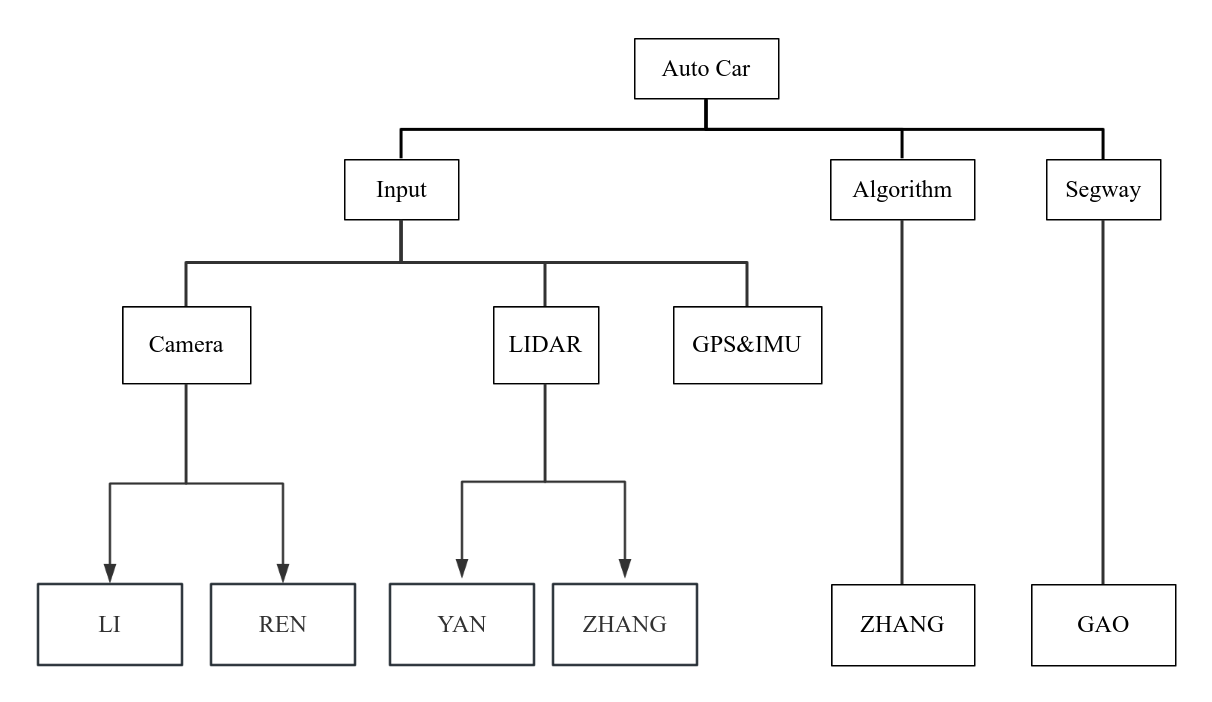
│ └── Packet Loss / Disconnect

└── Hardware Fault

├── Power Issue

└── Port Failure

1. **Project Management**
   1. **Team and subsystem organization**

****

* 1. **Timeline for the subsystem**

Table 7. Timeline

|  |  |
| --- | --- |
| **Week** | **Main Activities** |
| 1 | Acquired a new computer and installed Ubuntu 22.04 LTS, ROS 2 Humble, and MATLAB r2024b. |
| 2 | Defined the project topic, developed an initial lane detection algorithm using static images, and researched navigation algorithms based on occupancy maps. |
| 3 | Captured 10 training images, converted RGB to grayscale, applied Canny edge detection, and transformed the camera frame to the world frame. |
| 4 | Removed zebra crossings and other noise (e.g., bushes, trees, floor markings) and refined edges using morphological operations. |
| 5 | Performed Connected Component Analysis and curve fitting, using the Hough transform to eliminate lane arrow markings. |
| 6 | Collaborated with partner on drafting the paper; identified the lack of real-world test scenarios as the current bottleneck. |
| 7 | Set up the NUC and mobile platform to simulate the vehicle, migrated all files to the new NUC, and tested lane detection by laying white paper in hallway. |
| 8 | Discovered significant noise in edge detection due to diffuse reflections from lobby lighting and, per the professor’s suggestion, switched to a grayscale-based recognition algorithm. |
| 9 | Implemented dynamic thresholding based on average frame brightness; recorded brightness variations from 150 to 13, performed curve fitting, and confirmed a highly linear relationship; real-world tests demonstrated good adaptability. |
| 10 | Attempted to output an occupancy map compatible with LiDAR fusion, discovered scaling issues, then shifted focus to addressing the vehicle’s battery problems. |
| 11 | Attempted to recondition four batteries using a battery reconditioner (one passed, three failed); after consulting the professor, decided to fully discharge all four batteries before attempting a second reconditioning cycle. |
| 12 | After reconditioning, found that all four batteries remained non-functional. |
| 13 | Attempt to generate an occupancy map using inverse kinematics. |
| 14 | Calibrate the camera parameter matrix using a checkerboard. |
| 15 | Composed the final report and created the presentation slides. |

1. **Conclusion**
   1. **What Went Well**

* Multi-Sensor Calibration and Parameter Optimization: Successfully configured and tuned the Multisense S7 stereo camera [1] at 544p, 30 Hz, HDR settings to capture stable, high-quality environmental images, providing robust input for subsequent algorithms.
* Lane Detection Algorithm Implementation: In real-world environments, combined grayscale multi-level filtering with adaptive threshold segmentation to extract white lane markings with an average recognition accuracy exceeding 99%.
* RTAB‑Map Real-Time Mapping and Fusion: Fused visual point clouds with LIDAR and GPS data to achieve real-time, dense outdoor mapping and pose estimation, ensuring reliable inputs for the path planning module.
  1. **Challenges and Limitations**
* Subsystem Integration Delays: During integration with the LIDAR and GPS teams, issues with data synchronization and timestamp alignment affected short-term localization accuracy.
* Limited Applicability: The current algorithm only detects white lane markings, reducing its usability under varied road conditions.
* Team Leadership and Task Coordination: As team leader, I did not allocate tasks effectively to allow parallel progress; future projects should involve clearer task division and scheduling.
  1. **Future Improvement Plan**
* Real-World Field Testing: Conduct comprehensive end-to-end tests in outdoor environments with all sensor subsystems integrated to validate performance under competition conditions.

Overall, the development of the visual subsystem established a solid foundation in algorithm accuracy and multi-sensor fusion. However, it revealed limitations in real-time performance, environmental adaptability, and team coordination. Addressing these areas early—especially robust field testing and improved task parallelization—will accelerate performance gains and enhance the reliability of the autonomous navigation system.

**Reference**

[1] Carnegie Robotics, “MultiSense S7 Stereo Camera Spec Sheet,” Spec. CR‑01182021, Carnegie Robotics, Jan. 2021. [Online]. Available: https://carnegierobotics.com/AutonomousVehicles/CameraManufacturing/StereoCameraManufacturing/S7/Files/S7-Spec-Sheet.pdf. [Accessed: Apr. 20, 2025].

[2] Intelligent Ground Vehicle Competition, “2025 IGVC Competition Rules,” PDF document, Intelligent Ground Vehicle Competition, 2025. [Online]. Available: http://www.igvc.org/2025rules.pdf. [Accessed: Apr. 20, 2025].