1 Introduction

Building upon the initial architectural setup, our team has now transitioned into the execution phase with a strong emphasis on System Systems Engineering (SSE**)**. To effectively manage the complexity of developing an autonomous vehicle from scratch, we have adopted the V-Model framework as our primary engineering methodology.

This strict development lifecycle guided our workflow through distinct phases: starting from high-level requirement definition and general system layout, proceeding to detailed component design, and finally to fabrication and implementation. Following the ascending branch of the V-Model, we conducted rigorous verification steps—validating the accuracy of individual components before advancing to full system integration testing. This structured approach has been our "strategic weapon," ensuring a synchronized workflow and facilitating efficient error tracing and rectification through the model's inherent feedback loops.

2 Planned activities

Our autonomous system architecture relies on a sensor fusion approach utilizing four primary inputs: an Encoder, an Inertial Measurement Unit (IMU), a GPS module, and a Camera. These sensors serve as the foundation for addressing two complementary control objectives:

Problem 1: Precision Localization and Path Tracking (Simulation Phase) This objective focuses on developing robust navigation algorithms within the Gazebo simulation environment before physical deployment. By utilizing simulated sensor data (GPS, IMU, Encoder), we plan to implement an Extended Kalman Filter (EKF) for precise pose estimation and a Stanley Controller to compute steering angles for accurate waypoint tracking on the digital map.

Problem 2: Visual Perception and Context-Aware Control (Real-world Implementation) This objective targets environmental awareness and traffic rule compliance using the onboard camera. We aim to deploy a YOLOv8 deep learning model for real-time traffic sign detection. This perception layer will be coupled with a Finite State Machine (FSM) to govern the vehicle's decision-making process, translating visual inputs into motor actuation commands for autonomous maneuvering.

3 Status of planned activities

Hardware lead:

Implementation: Successfully integrated and mounted the wheel encoders onto the car kit. Completed the 3D printing of standardized traffic signs. These are ready for deployment and will serve as the primary targets for the camera-based perception system.
Difficulties: The installation of the wheel encoders required the space originally allocated for the battery. To resolve this mounting conflict, we are designing and 3D printing a custom battery relocation bracket to securely house the power supply without interfering with the new odometry hardware.

Embedded lead:

Implementation: Successfully implemented the communication protocol to retrieve raw data from both the wheel encoders and the IMU via the I2C bus.

Difficulties: Identifying the cause of a consistent bias in the steering actuator. Despite sending neutral commands, the car drifts from the center line, requiring a calibration look-up table or software offset to correct the actual heading.

Planning lead:

Implementation: We designed a path-following algorithm that enables the vehicle to track consecutive nodes on the map. Along with that, a planning strategy using the Greedy algorithm was also designed to ensure the vehicle navigates through all designated target waypoints in the Gazebo simulation environment.

Difficulties: The vehicle sometimes loses its target node and moves erratically until it finds another valid one. To fix this, the algorithm needs a recovery mechanism so the vehicle can independently find its way back if it gets lost.

Perception lead:

Implementation: We curated a custom dataset and performed transfer learning on YOLOv8 with Non-Maximum Suppression (NMS) for robust object detection. This perception layer is now fully integrated with the control logic, enabling the vehicle to autonomously modulate steering and throttle in response to regulatory traffic signs.

Difficulties: Computational constraints on the embedded hardware cause latency and low FPS during simultaneous inference and control tasks. We are currently mitigating this bottleneck through model quantization and multi-threading optimizations.

Control lead:

Implementation: We have successfully deployed and validated our Extended Kalman Filter (EKF) and Stanley Control algorithms within the Gazebo simulation environment. The vehicle successfully navigated the designated waypoints, confirming the logic of our path-tracking and sensor fusion pipelines.

Difficulties: Sim-to-Real Transition & Sensor Characterization: Unlike the simulation, real-world sensors require precise measurement of their noise covariance and directional bias. Without these calibrated parameters as inputs for the EKF node, we cannot ensure stable pose estimation in a physical environment, currently delaying live track testing.

4 General status of the project

Completed: Integrated Hardware (Encoders, 3D signs); Validated Simulation (EKF, Stanley, Greedy); Integrated YOLOv8 for real-time response.

In Progress: Optimizing Mechanical (Battery bracket) and Performance (Quantization, Multi-threading); Calibrating Steering Offset.

Limitations: Handling Sim-to-Real sensor noise and developing a Path Recovery mechanism.

5 Upcoming activities

Hardware and Software Integration for Localization**:** Developing a robust system to track the vehicle's real-time position on the physical track.

Autonomous Lane Following**:** Implementing advanced control algorithms to ensure the car stays within lane boundaries while navigating.

Full Autonomous Deployment on Physical Track: Conducting comprehensive testing and fine-tuning of the autonomous system in a real-world environment.