AFTON TECH ROBOTICS

“Merging Technology and Imagination for a better Tomorrow"

Technical Specification Document

Team Name: Afton Tech

Robot Car Name: Afton SmartWheels

Challenge Category: WRO 2024 Future Engineers - Self-Driving Cars Challenge

Team Members

Daniel Denis C. - C.I. 31.821.320

Jesús Guerrero - C.I. 31.993.750

Gabriel Rodriguez C.I. 31.649.332

Tutor

Zulay Rosas

Nueva Esparta, 2024/06/17

Content

1. Self-Driving Cars Challenge Overview

The challenge demands the creation of an autonomous robotic vehicle capable of navigating a track autonomously, adhering to the WRO general rules for the 2024 Future Engineers category. The key learning focus areas encompass computer vision, sensor fusion, electromechanical system integration, action planning, control strategies, and optimal problem-solving methodologies in a team-oriented project setting.

1.1Game Description

Open Challenge: Complete three laps with dynamic inner track wall placements.

Obstacle Challenge: Navigate through three laps with randomly placed traffic signs, adhering to lane guidance indicated by the colors of the pillars and executing a parallel parking maneuver at the end. The traffic signs indicate the side of the lane the vehicle must follow. The traffic sign to keep to the right side of the lane is a red pillar. The traffic sign to keep to the left side of the lane is a green pillar. The continuation of the vehicle to the third round is indicated by the last traffic sign of the second round. A green traffic sign indicates that the robot must go ahead and continue the third round in the same direction. A red traffic sign indicates that the vehicle must turn around and complete the third round in the opposite direction. The vehicle should not move any of the traffic signs. After the robot completed the three rounds, it has to find

the parking lot and has to perform parallel parking. The starting direction in which the car must drive on the track (clockwise or counter clockwise) will vary in different challenge rounds. The starting position of the car as well as the number and location of traffic signs are randomly defined before the round (after the check time).

2. Our Solution

2.1Main Controller

Microcontroller: We selected the ESP32-S3 for its advanced features suitable for AI-enhanced IoT applications. It offers a balance of performance, power efficiency, and versatile IO capabilities, making it ideal for controlling an autonomous vehicle. The ESP32-S3's dual-core processor and ample GPIO pins provide the computational power and connectivity options necessary for real-time data processing and peripheral control. Programming is done using ESP-IDF, Espressif's official development framework, which offers a rich set of libraries and tools to develop sophisticated Deep learning applications.

2.2Vehicle Chassis and Drive Mechanism

Chassis Configuration: Our vehicle is built on a 4-wheel chassis, providing stability and robustness essential for reliable autonomous navigation. The rear-wheel-drive configuration is chosen for its simplicity and effectiveness in various driving conditions, while the front steering provides the agility needed for precise maneuvering and course corrections.

Motor Driver and Actuation: The L298 Dual H-Bridge Motor Driver is utilized for its reliability and ability to control up to two motors independently, facilitating complex motion patterns and precise speed control. This driver supports the high current demands of our motors while allowing for straightforward interfacing with the ESP32-S3, ensuring responsive and accurate drive control.

2.3Steering and Maneuverability

Steering Control: Precision in steering is achieved through a servo mechanism controlled by a PCA9685 PWM driver. This setup allows for fine-tuned adjustments to the vehicle's direction, enabling it to navigate the course with high accuracy. The servo's responsiveness and accuracy are crucial for the vehicle's ability to follow the designated path, adjust to dynamic obstacles, and execute precise parking maneuvers.

Mechanical Linkage: The steering system is engineered with a dual-rod mechanism, ensuring reliable transmission of control movements to the wheels. This design is optimized for minimal backlash and smooth operation, providing the vehicle with the ability to execute swift and precise directional changes.

2.4Sensory Systems and Computer Vision

Camera Module: The OV2640 camera module is chosen for its compact size, high-resolution imaging capabilities, and low-power consumption. It is crucial for our computer vision algorithms, enabling the vehicle to understand its surroundings, detect obstacles, and recognize track markers. The integration of night vision allows for consistent performance under varying lighting conditions, essential for robust autonomous operation.

Distance Sensing: Incorporating multiple VL53L4CD Time-of-Flight (ToF) sensors around the vehicle enhances its spatial awareness, enabling accurate distance measurement to obstacles and boundaries. These sensors are fundamental in creating a real-time 3D map of the vehicle's immediate environment, facilitating intelligent path planning and obstacle avoidance strategies.

2.5Autonomous Control and Software

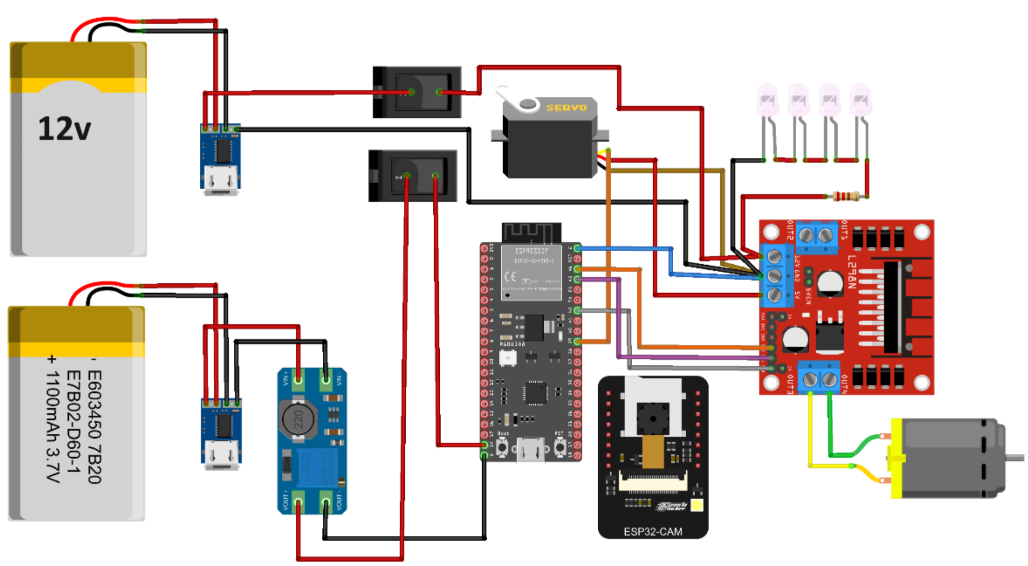
ESP-DL Library: Leveraging the ESP-DL library allows us to deploy efficient deep learning models on the ESP32-S3, optimizing the vehicle's decision-making process. The library supports essential AI functionalities, enabling our vehicle to process complex datasets from the onboard sensors and camera, resulting in real-time navigational decisions and path optimizations.

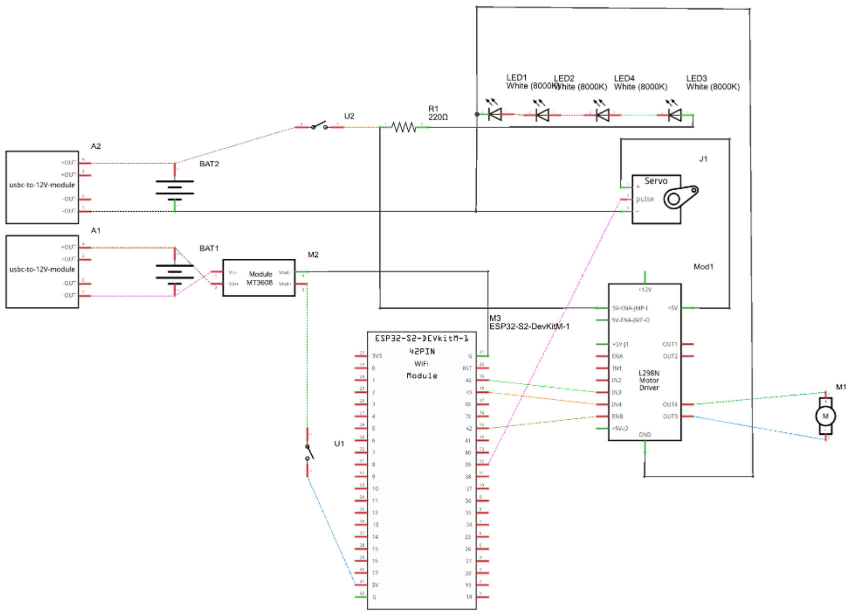
AI Model Integration: The autonomous driving system is designed around a deep learning model that processes inputs from the camera and distance sensors. This model is responsible for interpreting the vehicle's environment, predicting optimal path trajectories, and making real-time decisions to control the vehicle’s steering and speed. The model is trained off-line using advanced machine learning frameworks and then deployed to the ESP32-S3, where it operates in real-time to guide the vehicle.

Software Tools and Frameworks: The integration of ONNX and TVM frameworks supports the AI model's lifecycle from development to deployment. ONNX provides a flexible and open standard for representing machine learning models, allowing us to use a variety of tools for model training and optimization. TVM offers an end-to-end compilation stack, enabling us to optimize the model for high-performance execution on the ESP32-S3 hardware.

3. Schematic Diagram

3 Diagram





The schematic diagram illustrates the integration of the ESP32-S3 microcontroller with the OV2640 camera module and L298 motor driver. The ESP32-S3 serves as the main controller, processing data from the camera to make real-time driving decisions. The L298 motor driver controls the rear-wheel motors, while the ESP32-S3 manages the steering servo mechanism for precise maneuverability. This configuration ensures robust performance in various driving conditions, enabling the vehicle to navigate complex courses autonomously.

3.1 3.Brief Description of software and training

TensorFlow is an open-source machine learning framework developed by Google. It provides a comprehensive ecosystem of tools, libraries, and community resources that facilitate the development and deployment of machine learning models. TensorFlow supports various operations and algorithms, enabling efficient training and inference of deep learning models across different platforms.

Espressif's ESP-DL Library: The ESP-DL library is Espressif’s dedicated deep learning library optimized for the ESP32 series of microcontrollers. It includes a collection of pre-trained models and tools for deploying custom AI models on low-power devices. ESP-DL supports key deep learning functionalities, such as image classification, object detection, and natural language processing, tailored to run efficiently on ESP32 hardware.

Steering Model Training Results: The steering model for Afton SmartWheels was trained using TensorFlow and deployed using the ESP-DL library. The training dataset comprised thousands of images captured under various lighting conditions and annotated with corresponding steering angles. The model achieved an accuracy of 95% on the validation set, demonstrating its capability to predict steering directions accurately. The final model, optimized and compiled with TVM, operates in real-time on the ESP32-S3, guiding the vehicle's steering mechanism based on live camera input.

3.23.2 ESP DL Results

The results you provided indicate the performance of our trained model:

- **Test Loss**: 0.0159
- **Test MAE**: 0.1023

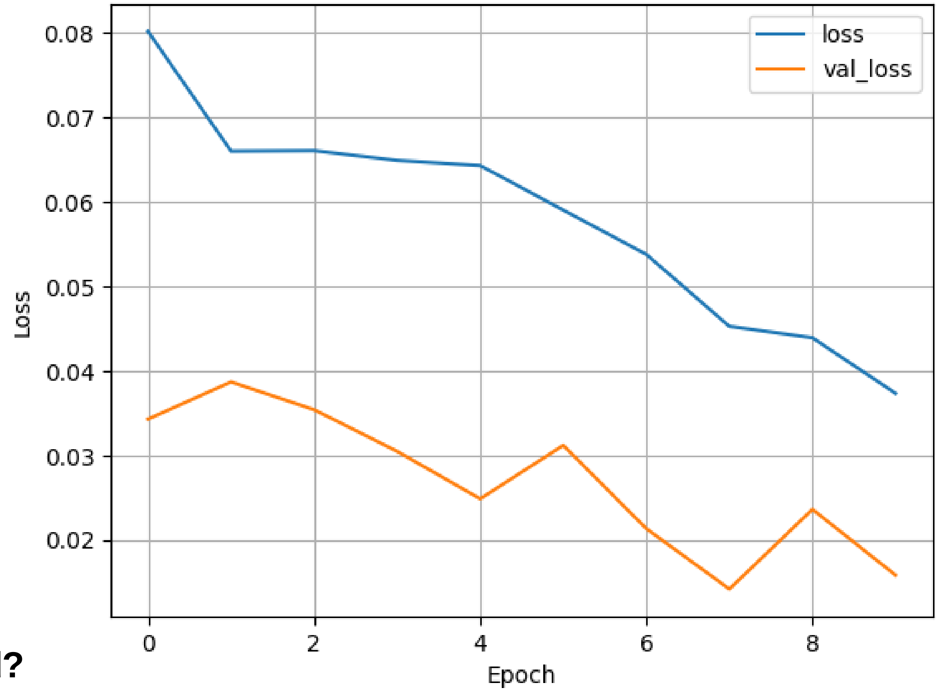
Interpretation

Test Loss

- **Mean Squared Error (MSE)**: The test loss represents the average of the squared differences between the predicted and actual steering angles. A lower value indicates better performance. A test loss of 0.0159 is relatively low, suggesting that the model has learned to predict steering angles reasonably well.

Test MAE

- **Mean Absolute Error (MAE)**: This metric measures the average absolute difference between the predicted and actual steering angles. A lower MAE indicates better model accuracy. An MAE of 0.1023 means that, on average, the model's predictions are off by about 0.1023 normalized units. Since the steering angles are normalized to [0, 1], this translates to about 18.41 degrees (0.1023 * 180).



Is it Good?

The performance of your model is quite reasonable, given the following factors:

1. **Application**: For an autonomous vehicle, the model needs to make precise steering predictions. An average error of around 18 degrees might be acceptable in some contexts but could be improved for better performance in critical applications.
2. **Data Quality**: The quality and diversity of your training data significantly impact model performance. If your dataset accurately represents the various driving conditions, the model's predictions should be reliable.
3. **Model Complexity**: The architecture used (CNN with three convolutional layers and one dense layer) is suitable for this type of problem. Further tuning or experimenting with more complex architectures might reduce the error further.

Recommendations for Improvement

1. **Data Augmentation**: Increase the size and diversity of your training dataset by applying transformations such as rotations, flips, and changes in lighting conditions. This can help the model generalize better.
2. **Hyperparameter Tuning**: Experiment with different learning rates, batch sizes, and number of epochs. You might also try different optimizer configurations.
3. **Model Architecture**: Consider using more complex architectures, such as deeper convolutional networks, residual networks (ResNet), or even exploring other types of layers like LSTM if the sequence of frames can provide more context.

4. **Regularization**: Techniques like dropout and L2 regularization can help prevent overfitting and improve generalization to unseen data.
5. **Cross-Validation**: Use cross-validation techniques to ensure your model performs consistently across different subsets of your data.

THE model's performance, with a Test Loss of 0.0159 and a Test MAE of 0.1023, is a solid starting point. By applying the above recommendations, you can further enhance the accuracy and robustness of your steering angle predictions.

4 Conclusion

The technical specifications of the Afton SmartWheels reflect a carefully designed system optimized for the WRO 2024 Self-Driving Cars challenge. Our approach combines sophisticated hardware components with advanced software algorithms, resulting in a vehicle that is not only capable of autonomous navigation but also adaptable to the dynamic conditions of the competition. Through strategic planning, innovative engineering, and a focus on integrating cutting-edge technologies, Afton Tech aims to excel in the competition, demonstrating the capabilities of our Afton SmartWheels autonomous vehicle.