Autonomous Road Navigation with Nvidia JetBot

Sasank Sribhashyam, Nikhil Sai Kondapaneni, Vineel Palla

Abstract:

This report delves into the cutting-edge techniques employed to enable autonomous navigation utilizing the powerful capabilities of the NVIDIA JetBot. Through a fusion of machine learning tools and neural networks, coupled with innovative data collection methodologies utilizing the CSI cam, our team has successfully engineered a system that not only traverses' lanes autonomously but also integrates state-of-the-art SLAM techniques for robust localization and mapping. Leveraging the flexibility and scalability of PyTorch, we have meticulously crafted a comprehensive solution that showcases the full potential of the NVIDIA JetBot platform.

Introduction:

The pursuit of safer and more efficient transportation systems has led to the development of autonomous navigation technologies. In this report, we document our efforts to enhance autonomous navigation using the NVIDIA JetBot platform. Our goal is to equip the JetBot with the ability to autonomously navigate lanes, avoid obstacles, and optimize speed, thereby contributing to the advancement of transportation safety and efficiency.

Methodology:

Data collection:

Our methodology commences with the crucial step of data collection. Utilizing the high-resolution camera mounted on the JetBot, we systematically capture real-world scenarios encompassing a diverse range of road conditions, lane configurations, and obstacles. Each image captured serves as a snapshot of the environment, providing invaluable insights into the intricacies of navigation.

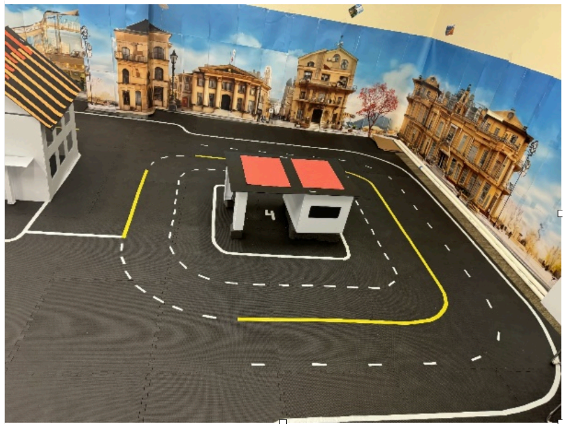


Figure:

To ensure the optimization of path following, we adopt a meticulous approach to data collection. Over the course of our data gathering efforts, we amassed approximately 350 images, meticulously covering every corner of the lane. This comprehensive dataset encompasses a wide spectrum of scenarios, ranging from straight stretches to sharp turns, thereby facilitating robust path optimization during model training.

Improvization:

Moreover, in our quest to enhance speed optimization, we introduce a novel speed equation derived from real-world data. By analyzing the relationship between lane curvature, obstacle proximity, and environmental conditions, we formulate a dynamic equation that governs the JetBot's speed adjustment mechanism. This equation serves as a cornerstone in our efforts to achieve seamless and efficient navigation.

$$\text{speed} = \text{average}\left(1 - \left|\frac{x - \frac{height}{2}}{\frac{height}{2}}\right|, 1 - \left|\frac{y - \frac{height}{2}}{\frac{height}{2}}\right|\right)$$

Fig: speed equation.

Model training:

Armed with a rich and diverse dataset, we transition to the model training phase. Leveraging the power of the PyTorch framework, we train our neural network models to effectively interpret and navigate the complexities of the captured environment. Through iterative training iterations, our models learn to recognize lane markings, anticipate obstacles, and optimize speed, thereby equipping JetBot with the intelligence necessary for autonomous navigation.

Equipped with a rich and diverse dataset, our journey transitions to the pivotal phase of model training. Harnessing the formidable capabilities of the PyTorch framework, we embark on the intricate task of training our neural network models to adeptly interpret and navigate the complexities of the captured environment. At the heart of our training process lies the ResNet architecture, a stalwart guide in our quest for robust autonomy.

ResNet, renowned for its depth and efficiency, serves as the backbone of our neural network model. Through iterative training iterations, we leverage ResNet's exceptional capacity to extract intricate features from the input data, enabling our models to discern subtle nuances in lane markings, anticipate the presence of obstacles, and dynamically adjust speed parameters.

Each training iteration serves as a crucible, where our models undergo refinement and enhancement, gradually sharpening their ability to comprehend and respond to the intricacies of real-world navigation scenarios. Guided by ResNet's architectural prowess, our models evolve into adept navigators, equipped with the intelligence necessary to navigate autonomously.

By leveraging the power of ResNet within the PyTorch framework, we imbue the JetBot with the cognitive capabilities required for seamless and efficient autonomous navigation. This fusion of cutting-edge technology and innovative methodologies paves the way for a new era of intelligent transportation systems, where safety, efficiency, and autonomy converge to redefine the future of mobility.

Experimental results:

Performance evaluation:

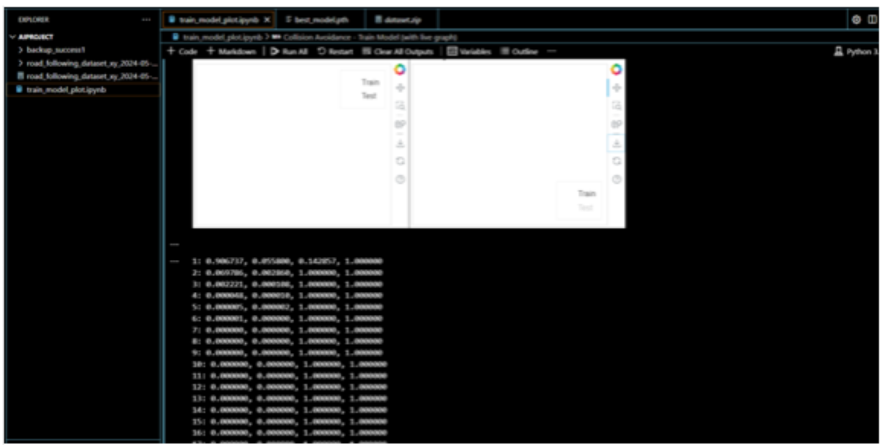
Rigorous experimentation forms the cornerstone of our evaluation process. We meticulously assess the performance of our autonomous navigation system across a myriad of metrics, including path following accuracy, obstacle detection rate, and speed optimization efficiency. By subjecting our system to diverse real-world scenarios, we validate its robustness and efficacy in navigating complex environments.

Metrics:

Performance metrics serve as vital indicators of the effectiveness and efficiency of our autonomous navigation system. By quantitatively evaluating key aspects of system performance, we gain valuable insights into its strengths and areas for improvement.

Lane Following Accuracy:

Lane following accuracy measures the system's ability to accurately track and adhere to lane markings. This metric is assessed by comparing the trajectory predicted by the JetBot with ground truth lane positions. A high lane following accuracy indicates that the JetBot is effectively navigating within the designated lanes, minimizing deviations and ensuring safe traversal.



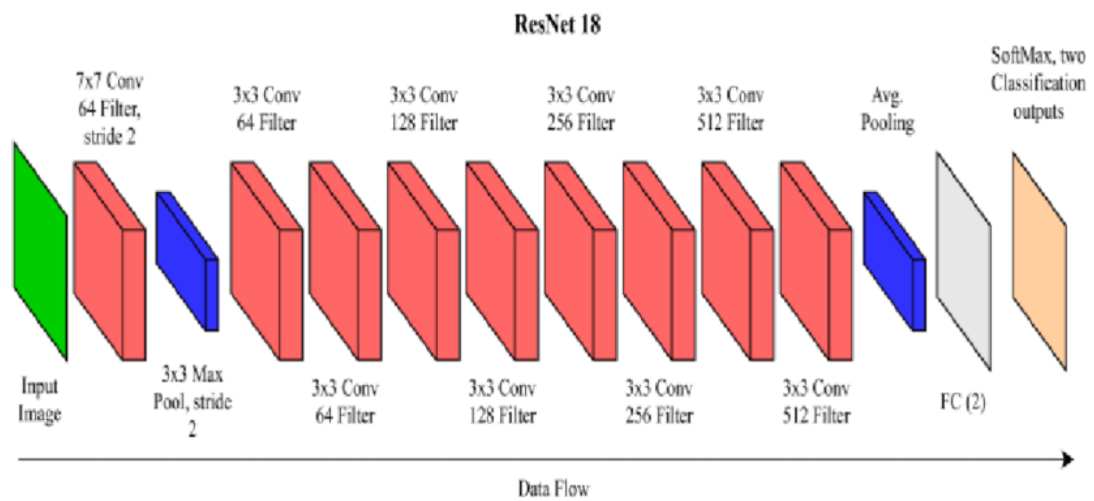
Obstacle Detection Rate:

Obstacle detection rate evaluates the system's proficiency in detecting and avoiding obstacles in its path. Utilizing ground truth obstacle positions, we quantify the system's ability to correctly identify and respond to potential hazards. A high obstacle detection rate signifies that the JetBot can effectively perceive and react to obstacles, mitigating collision risks and ensuring safe navigation.

Speed Optimization Efficiency:

Speed optimization efficiency measures the system's effectiveness in dynamically adjusting its speed to navigate through varying environmental conditions. By analyzing the correlation between speed adjustments and environmental factors such as lane curvature and obstacle proximity, we assess the system's ability to maintain an optimal speed profile. A high speed optimization efficiency indicates that the JetBot can adapt its speed to navigate efficiently while ensuring timely responses to changes in the environment.

By meticulously evaluating these performance metrics, we gain comprehensive insights into the capabilities and limitations of our autonomous navigation system. These quantitative assessments guide our efforts in refining and optimizing the system, driving continual improvements in safety, efficiency, and overall performance.



Conclusion:

In conclusion, our methodology, rooted in comprehensive data collection, rigorous model training, and innovative speed equation formulation, culminates in the development of a robust autonomous navigation system for the NVIDIA JetBot platform. By leveraging real-world data and advanced machine learning techniques, we empower the JetBot with the capability to autonomously navigate lanes, evade obstacles, and optimize speed. This achievement not only advances the field of autonomous navigation but also contributes to the broader goal of enhancing transportation safety and efficiency.

Future work:

Looking ahead, our future endeavours will focus on further refining and optimizing our autonomous navigation system. This includes exploring advanced neural network architectures, integrating additional sensors for enhanced perception, and fine-tuning the speed equation for optimal performance across diverse environments. Through continuous innovation and collaboration, we aim to push the boundaries of autonomous navigation technology and pave the way for a safer and more efficient transportation future.

References:

[1] JetBot (https://jetbot.org)

[2] Adventures with the Nvidia JetBot and JetRacer (https://www.diyrobocars.com)

[3] Examples · NVIDIA-AI-IOT/jetbot Wiki (https://github.com)

[4] JETBOT (https://www.netapp.com)

[5] ResNet — Torchvision main documentation (https://pytorch.org)