DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Simulation Based Autonomous Driving Project Report

Efficient Autonomous Driving Using Hierarchical Representation Learning

Huzeyfe Ayaz

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Simulation Based Autonomous Driving Project Report

Efficient Autonomous Driving Using Hierarchical Representation Learning

Author:	Huzeyfe Ayaz
Supervisor:	Liguo Zhou
Submission Date:	09/07/2023

Contents

1 Motivation & Introduction

Autonomous driving represents a transformative paradigm shift in the automotive industry, promising safer and more efficient transportation. It promises not only convenience but also a significant reduction in traffic accidents and congestion. However, achieving the level of safety and reliability required for widespread adoption is an ongoing challenge. The goal of this research is to contribute to the advancement of autonomous driving technology by leveraging simulation-based approaches. Simulation provides a controlled and cost-effective environment to develop, test, and validate autonomous driving algorithms, ultimately accelerating the deployment of autonomous vehicles.

Simulating real-life scenarios is crucial for the development and validation of autonomous driving systems. For this reason, there exist various simulators such as Udacity [1] simulator and CARLA [2] that are designed to faithfully replicate the intricacies of traffic and driving environments. As researchers, we leverage these simulation tools to facilitate the generation of more efficient and robust algorithms and machine learning (ML) models for autonomous driving. Throughout this study, we conducted experiments involving a range of Deep Learning-based approaches, including Convolutional Neural Networks (CNN), AutoEncoders, and Semantic Segmentation Models, within a new simulated environment developed by Technical University of Munich (TUM). This report provides a comprehensive overview of the fundamental aspects of our research and explores its potential contributions to the advancement of autonomous driving technology.

2 Related Works

In autonomous driving systems, 4 different system architecture are mainly used which are Ego-only Systems, Modular Systems, End-to-end Driving, and Connected Systems [3].

In the ego-only approach, a self-sufficient vehicle autonomously handles all automated driving operations, whereas the connected Automated Driving System (ADS) approach may depend on other vehicles and infrastructure as needed. The ego-only approach is more common, primarily due to its practicality and the increased complexities associated with connected systems.

Modular automated driving systems are composed of distinct elements organized in a sequential structure, encompassing tasks such as localization, perception, planning, vehicle control, and human-machine interaction. Creating these modules separately serves to streamline the intricacies of automated driving, benefiting from knowledge in fields like robotics, computer vision, and vehicle dynamics. While modular systems offer benefits such as seamless function integration and redundancy, they are also vulnerable to issues like error spread and excessive complexity.

End-to-end driving, also known as direct perception, involves generating ego-motion directly from sensory inputs, with three main approaches: direct supervised deep learning, neuroevolution, and deep reinforcement learning. These methods use neural networks to predict driving actions, with direct supervised learning relying on human driver imitation, deep reinforcement learning seeking optimal driving behavior, and neuroevolution employing evolutionary algorithms to train networks. While promising, real-world implementation of end-to-end driving in urban settings remains limited, primarily due to concerns about safety, interpretability, and the need for online interaction and repeated failures in some approaches like deep reinforcement learning and neuroevolution.

The future of automated driving systems (ADS) lies in connected systems that rely on Vehicular Ad hoc NETworks (VANETs) and V2X communication to overcome the constraints of self-contained ego-only platforms. VANETs can be established through traditional IP-based networking or Information-Centric Networking (ICN) [4], where ICN offers advantages in terms of mobility and data exchange. Despite the significant advantages that connected systems promise, they also present challenges like security, routing, and reliability, complicating the development of fully functional connected systems.

In our research, we primarily focused on imitation learning, in which deep learning models learn human interactions with vehicles and mimic these behaviors.

3 Methodology

In the initial phase of our project, we employed the Udacity simulator as our primary platform for developing a comprehensive automated driving system. In this subsequent stage, our objective is to construct a highly efficient Convolutional Neural Network (CNN) model that can operate seamlessly on a dual-core Intel i5 laptop while also possessing the requisite computational prowess to effectively steer the vehicle in question. This entails a careful balance of computational efficiency and functional competence, demanding a judicious selection of network architecture and optimization techniques.

In the subsequent phase of our project, we transitioned to a new simulator environment, where we undertook a multifaceted approach to enhance our automated driving system. This involved the exploration of various strategies, with a primary focus on the utilization of auto-encoder models for scene analysis and parameter prediction.

Our first approach encompassed the training of an auto-encoder model using the simulator-generated scenes. Following successful training, we leveraged the encoder component of this model as a feature extractor. Subsequently, we constructed a fully connected layer atop this feature extractor to predict crucial parameters such as throttle, brake, and steering-angle values. This approach aimed to harness the encoded scene representations to facilitate precise vehicle control.

In parallel, we also investigated an alternative avenue by employing the backbone portions of semantic segmentation models. Fast-SCNN [5] model from MMSegmentation module [6] were employed to generate embeddings, serving as an alternative feature representation due to it's fast inference time. Following this, we applied a similar strategy as employed with auto-encoders, constructing fully connected layers on top of these embeddings for parameter prediction, thereby enabling effective vehicle control.

These dual approaches represent a comprehensive effort to enhance the performance and adaptability of our automated driving system, underpinned by advanced feature extraction and predictive modeling techniques.

4 Results

In the initial phase of our project, we embarked on the development of a streamlined and agile model architecture. This model was designed to strike a balance between computational efficiency and task performance, adhering to the following specifications:

- The model consisted of four convolutional blocks, each characterized by a filter size of 3, a stride of 1, and zero-padding.
- These convolutional layers were subsequently followed by Rectified Linear Unit (ReLU) activation functions and an additional average pooling layer with 2x2 kernel and a stride of 2, was incorporated into the model.

This architecture was meticulously trained on data from the first track, resulting in a model that exhibited a smooth and reliable performance on this track. Remarkably, even though the model was not explicitly trained on data from the second track, it demonstrated the ability to operate the vehicle on the second track, albeit with occasional human intervention. This adaptability underscores the model's potential for generalization and highlights its robustness in an unforeseen environment.

During the next stage of our project, we encountered some difficulties that forced us to make changes to our computer setup. Unfortunately, our original dual-core Intel i5 computer was not up to the task of efficiently running the new simulator. As a result, we acquired a new Mac computer with more powerful computing capabilities to better support our project.

Despite the increased computing power of the new machine, we faced several technical challenges as we progressed in our work. These challenges included occurrences of a black screen and issues with connecting the server to the simulator. These setbacks were notable, but they did not dampen our determination to move forward.

In the midst of these challenges, we embarked on the process of training an auto-encoder model, and the result can be seen in Figure 4.1. Additionally, we presented an example of semantic segmentation achieved by the Fast-SCNN model, which is visualized in Figure 4.2.

After setting up a Windows virtual machine on our Mac system, we initiated the execution of the model. While this effort did set the vehicle in motion, it is crucial to mention that due to the connectivity issues previously mentioned, we were unable to conduct thorough testing of the models. Consequently, the initial results produced by the models were sub-optimal and will require further evaluation and refinement.

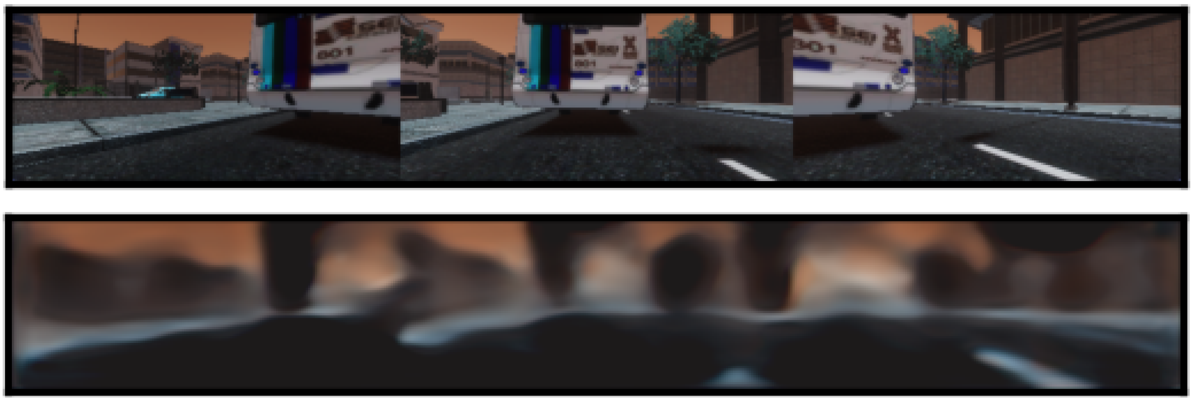

Figure 4.1: Auto-Encoder Model Result: Original image on top, generated image on the bottom.

Figure 4.2: Fast-SCNN model semantic segmentation result for a particular scene

5 Further Work

In a forthcoming project, we hold the view that semantic segmentation models possess significant potential for extracting image embeddings. These models are trained to identify and emphasize objects within images. Consequently, they have the capacity to capture intricate image details, potentially resulting in improved image vector representations.

Bibliography

[1] I. Udacity. *Udacity's Self-Driving Car Simulator*. https://github.com/udacity/self-driving-car-sim/tree/Unity_2019_4 [Accessed: 07.09]. 2023.

[2] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. "CARLA: An Open Urban Driving Simulator". In: *Proceedings of the 1st Annual Conference on Robot Learning*. 2017, pp. 1–16.

[3] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda. "A survey of autonomous driving: Common practices and emerging technologies". In: *IEEE access* 8 (2020), pp. 58443–58469.

[4] M. Gerla, E.-K. Lee, G. Pau, and U. Lee. "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds". In: *2014 IEEE world forum on internet of things (WF-IoT)*. IEEE. 2014, pp. 241–246.

[5] R. P. Poudel, S. Liwicki, and R. Cipolla. "Fast-scnn: Fast semantic segmentation network". In: *arXiv preprint arXiv:1902.04502* (2019).

[6] M. Contributors. *MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark*. https://github.com/open-mmlab/mmsegmentation. 2020.