Vision Based Autonomous Racing

Robin Davennes 49005102 **Martin Champion** 49005109 **Lucas Gorrec** 49005154 **Samuel Manchajm** 49005153

Abstract

This paper presents a vision-based approach to autonomous racing using the F1Tenth platform. We propose a pipeline that combines a U-Net-based convolutional neural network for real-time cost map prediction from dashcam images and Model Predictive Path Integral (MPPI) for trajectory optimization. The dataset was generated using the AutoDrive Simulator, enhancing model robustness through data augmentation. Our contributions include the design of the U-Net model for cost map generation and the integration of MPPI for control. Experimental results show promising performance in high-speed navigation. Further work focuses on reducing overfitting and optimizing inference time.

1 Introduction

Autonomous driving presents challenges, particularly in coupling perception and control for reliable navigation. Traditional methods relying on expensive sensors like LiDAR are not cost-effective or scalable for real-world applications. Vision-based approaches, using affordable cameras, offer a promising alternative. However, current implementations struggle with low speeds and fail to meet high-speed demands.

This paper focuses on a vision-based autonomous racing setup using the F1Tenth platform [2]. By employing a monocular dashcam and deep learning techniques, the goal is to achieve high-speed navigation through real-time perception and control. Specifically, we utilize a U-Net-based CNN for cost map generation and Model Predictive Path Integral (MPPI) for trajectory optimization.

Key contributions:

- Dataset generation using AutoDrive Simulator [4]
- Development of a U-Net architecture for transforming dashcam views into cost maps [1]
- Implementation of MPPI to translate cost maps into actionable trajectories [1]
- Evaluation of system performance and its potential for real-world applications

2 Method

2.1 Dataset Generation

Using AutoDrive Simulator [4], monocular dashcam images were collected and matched with cost maps in a bird's-eye view. Cost maps assign lower costs to track centers and higher costs to edges. Data augmentation techniques, such as brightness adjustment and cropping, were applied to enhance generalization.

2.2 U-Net Architecture

The U-Net-based CNN [3] predicts cost maps from dashcam images in real time. It comprises:

- **Contracting Path:** Reduces spatial dimensions while increasing feature depth with two convolutional blocks (16 and 32 filters) and max-pooling.
- **Bottleneck:** Extracts features with 64-filter convolutional layers.
- **Expanding Path:** Restores spatial resolution with transpose convolutions and merges encoder features via skip connections.
- **Output Layer:** A single convolution transforms features into cost maps.

Training used supervised learning with mean squared error (MSE) loss to minimize pixel intensity differences between predicted and target cost maps.

2.3 Control Strategy: Model Predictive Path Integral (MPPI)

MPPI, a derivative-free Model Predictive Control (MPC) method, generates candidate trajectories from cost maps and selects the optimal one. For trajectory simulation, the vehicle's initial speed and position are set to the cost map's bottom-center. Steering variations ($\Delta\theta$) are sampled from a Gaussian distribution, with steering angles limited to 30° as per simulator constraints. Each trajectory contains 50 points, and 200 trajectories are generated per cost map. The best trajectory minimizes cost by aligning points with low-cost regions (black areas).

Figure 1 illustrates the results of MPPI applied to a costmap. Once the best trajectory is selected, it is applied in the simulator using AutoDrive DevKit.

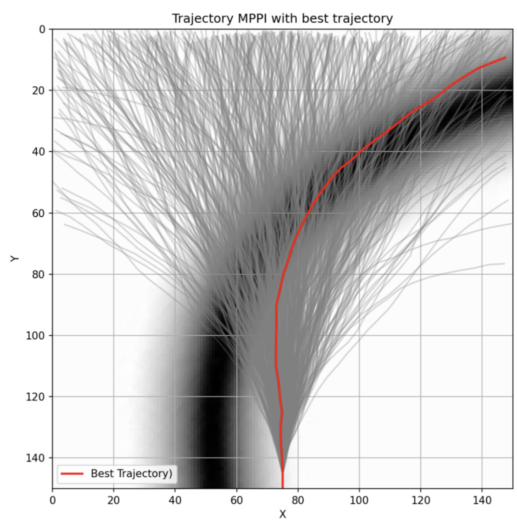

Figure 1: MPPI result for a costmap.

2.4 Pipeline Integration

The complete pipeline integrates:

1. Dashcam image capture in real time
2. U-Net processing to generate cost maps
3. MPPI computation for optimal trajectories
4. Control command translation for the F1Tenth car

2.5 Implementation Details

Implemented in Python using PyTorch for the U-Net model and NumPy for MPPI computations. Training was conducted on a GPU-enabled workstation, but real-time inference with the AutoDrive Simulator has not yet been tested.

3 Experiment

The training and validation loss curves for the U-Net model are presented in Figure 2. The training loss consistently decreased over the training iterations, indicating effective learning. However, the validation loss plateaued and slightly increased after a certain point, suggesting overfitting that will require further regularization or more data.

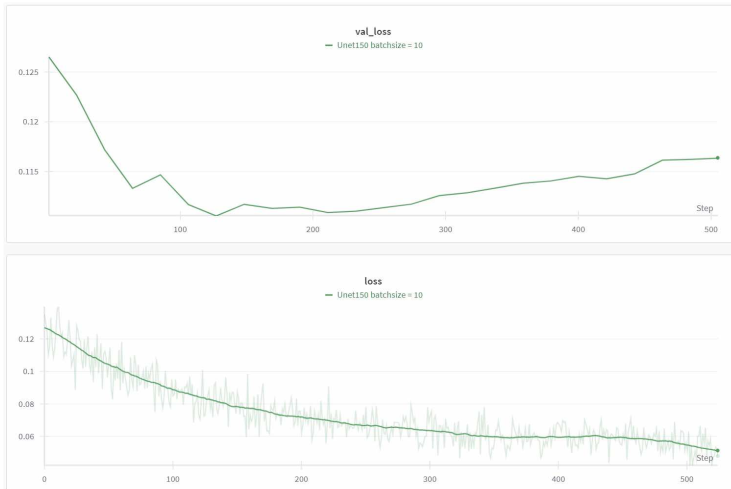

Figure 2: Training and validation loss for the U-Net model.

Figure 3 illustrates the output of the U-Net model compared to the ground truth cost map and the original dashcam input. The predicted cost maps match the desired output but show room for improvement in edge definition and noise reduction.

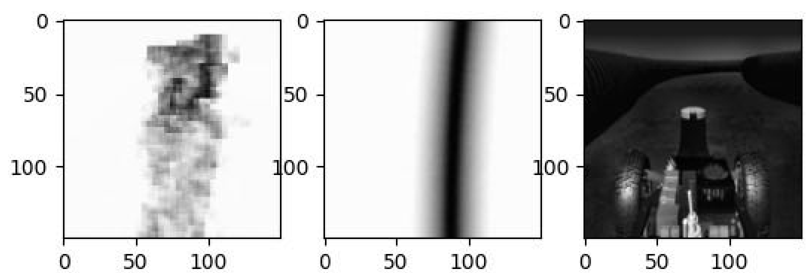

Figure 3: Left: Predicted cost map. Middle: Ground truth cost map. Right: Dashcam input.

4 Conclusion

The U-Net model successfully generated cost maps from dashcam images, enabling high-speed autonomous navigation in simulation. The integration of MPPI with the predicted cost maps is the next step to achieve real-time control. Preliminary results highlight the need for further data collection to improve model generalization and reduce overfitting. Additionally, integrating the perception and control pipelines will be critical for validating the system in real-world scenarios. Future work includes optimizing training processes and scaling the system for more complex environments.

References

[1] Drews, P., Williams, G., Goldfain, B., Theodorou, E.A., Rehg, J.M.: Aggressive deep driving: Model predictive control with a cnn cost model. arXiv preprint arXiv:1707.05303 (2017), http://arxiv.org/abs/1707.05303

[2] Evans, B.D., Trumpp, R., Caccamo, M., Jahncke, F., Betz, J., Jordaan, H.W., Engelbrecht, H.A.: Unifying f1tenth autonomous racing: Survey, methods and benchmarks (2024), https://arxiv.org/abs/2402.18558

[3] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation (2015), https://arxiv.org/abs/1505.04597

[4] Samak, T.V., Samak, C.V., Xie, M.: Autodrive simulator: A simulator for scaled autonomous vehicle research and education. 2021 2nd International Conference on Control, Robotics and Intelligent System (2021), http://arxiv.org/abs/2103.10030