Domain Background

Autonomous driving technology is gaining momentum as it helps in reducing accidents, and increasing the quality of experience for car users. Computer Vision based on deep learning is currently among the top techniques used in the field of autonomous driving. To have a functioning autonomous driving vehicle, the system needs to have different functional blocks, e.g, Automatic Emergency Braking (AEB), Lane Keeping Assist (LKA), Active Cruise Control (ACC), Traffic Jam Assist (TJA), and Crash Avoidance (CA).Traditional algorithms require difficult prerequisites: a) a well-defined map, b) sensor fusion, c) localization, and d) a controller. Unlike the traditional techniques, End-to-End solutions are proposed to solve the problem using a single block, instead of the complexities of the four previous blocks.

Fully Autonomous Driving is one of the most difficult problems facing the automotive applications. It is forbidden due to the presence of some restricted Laws that prevent cars from being autonomous for the fear of accidents occurring. However, researchers try to reach autonomous driving as a new area for research for the aim of having a strong push against these restricted Laws. Crash Avoidance functionality is considered as one of the most important features in Self-Driving Cars. Recently, it is partially integrated in the Self-Driving Car system.

A lot of research has been done in the End-to-End approach for the self driving cars,as the work which is proposed by NVIDIA in [1] put the basics of the end to end self driving car approach and prove that it is doable using imitation learning and also tested it on a real vehicle, INTEL also expanded the work to the conditional imitation learning for self driving cars in [2], which aims to let the driver able to interact with the vehicle by determining a desired destination or using navigation’s maps to determine the destination.

Problem Statement

This project aims to improve the end to end approach by providing a temporal information not only from the past but also using the predicted future.I do believe that the predicted future will improve the accuracy of the deep learning agent and will make it more robust.The future information will be provided to the model using GANS.I will use conditional imitation learning as described in [2], but I will feed the network not only by the current image but also with the past frames and the future predicted frames by GANS(using cycle gan, pix2pix,....etc).An ablation study will be made to compromise the best setup of the input channels, which means how many frames from the past and the predicted future should be stacked with the current frame to get the best accuracy.

Datasets and Inputs

I will use the dataset which is provided by INTEL[3],the training dataset comprises 2 hours of human driving in Town 1 of which only 10% (roughly 12 minutes) contain demonstrations with injected noise. Collecting training data with strong injected noise was quite exhausting for the human driver. However, a relatively small amount of such data proved very effective in stabilizing the learned policy. The data size is 24 GB, the RGB images stored at 200x88 resolution, and the dataset provides 28 different label which will be very useful while training the agent(e.g, Steer, Gas, Brake,...).The data is collected using CARLA simulator[4] so I will be able to create my own dataset if needed.But mainly I will use the provided data to be able to compare my results with their results.
The input representation will be as follows, the current frame captured from the camera in this timestamp, the previous frames which are captured at the previous timestamps and the predicted frames from the future(using GANS).The weighting of the percentage of the past, current and future frames will be discussed using the results.

Solution Statement

I will use the proposed architecture in [3] and change the input representation only, instead of using the current frame only or stacking it with the previous frames, I will stack them with frames predicted from the future using GANS.I do believe that stacking future frames will enhance the accuracy.

Benchmark Model

I will measure my work on CARLA benchmark[5].This benchmark put the agent in a fixed positions and let it control the vehicle for a period of time and record some measurements like how many times the agent make a collision, how many times the agent deviates from the road and intersection with the other lane.

Evaluation Metrics

As mentioned above, I will use CARLA benchmark, which records a lot of measurements, and I will follow the same evaluation metrics mentioned in the conditional imitation learning paper[2].They measure the percentage of successful episodes and the average distance (in km) driven between infractions. Higher is better in both cases.The episode will be considered as a successful one when the agent start from a start position and succeed in reaching the desired goal.The evaluation metric will be applied on CARLA town 1 and 2. Town 1 is the training town and town 2 the agent has never seen it so it is a good indication for generalization.

Project Design

Make a good analysis of the data before being used is very important, for now I will draw the histogram for the steering angle, throttle and the brake values or which algorithms will be considered for your implementation.

The workflow will be as follows:

1. Train a predictor to predict the future frames from the current one using paired Gans (like pix2pix [6])
2. Use the imitation learning to train an agent using previous, current and the predicted frames from step 1.
3. Evaluate different models on CARLA benchmark(Ablation study).

.

References

[1]https://images.nvidia.com/content/tegra/automotive/images/2016/solutions/pdf/end-to-end-dl-using-px.pdf

[2]https://arxiv.org/abs/1710.02410

[3]https://github.com/carla-simulator/imitation-learning

[4]https://github.com/carla-simulator/carla

[5]https://github.com/carla-simulator/driving-benchmarks

[6]https://phillipi.github.io/pix2pix/