**Deep Learning Implementation Of Self Driving Car**

**Team Members:**

***15BCE0205(Radhika Garodia)***

***15BCE0311( Kanav Sethi)***

**Report submitted for the**

**First Project Review of**

**Course Code: CSE4020 – Machine Learning**

**Slot: E2**

**Professor: Jaisakthi S M**

****

**Abstract:**

Reinforcement learning is considered to be a strong Machine learning paradigm which can be used to teach machines through interaction with the environment and learning from their mistakes. We propose a framework for autonomous driving using deep reinforcement learning.

It incorporates Recurrent Neural Networks for information integration, enabling the car to handle partially observable scenarios. It also integrates the recent work on attention models to focus on relevant information, thereby reducing the computational complexity for deployment on embedded hardware. Our simulation results demonstrate learning of autonomous manoeuvring in a scenario of complex road curvatures and simple interaction of other vehicles. An example is in the auto industry; many companies are trying to make their cars smarter. The cars can self-drive, avoid obstacles; find destinations without controls from human. The approach uses Machine Learning, Neural Network and a Deep Q learning network to train the car by making the car learn after each time it fails to finish the track.

1. **Introduction**:

**What is reinforcement learning?**

Reinforcement Learning is a type of machine learning that allows you to create AI agents that learn from the environment by interacting with it. Just like how we learn to ride a bicycle, this kind of AI learns by trial and error. As seen in the picture, the brain represents the AI agent, which acts on the environment. After each action, the agent receives the feedback. The feedback consists of the reward and next state of the environment. The reward is usually defined by a human. If we use the analogy of the bicycle, we can define reward as the distance from the original starting point.

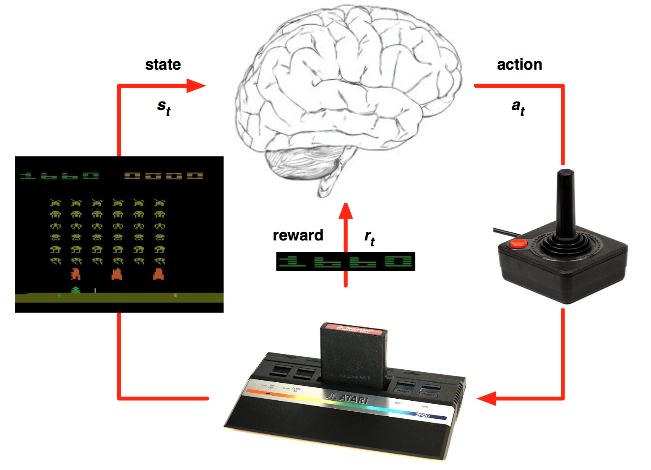


fig 1. – how reinforcement learning works.

A robot car that drives autonomously is a longstanding goal of Machine Learning. The tasks involved in creating an autonomous driving agent can be divided into 3 categories.

1) Recognition: Identifying components of the surrounding environment. Examples of this are pedestrian detection, traffic sign recognition, etc. Deep learning models are able to learn complex feature representations from raw input data, omitting the need for handcrafted features.

2) Prediction: It is not enough for an autonomous driving agent to recognize its environment; it must also be able to build internal models that predict the future states of the environment. To be able to predict the future, it is important to integrate past information. As such, Recurrent Neural Networks (RNNs) are essential to this class of problem.

3) Planning: The generation of an efficient model that incorporates recognition and prediction to plan the future sequence of driving actions that will enable the vehicle to navigate successfully. Integrating the ability of the model to understand the environment (recognition) and its dynamics (prediction) in a way that enables it to plan the future actions such that it avoids unwanted situations (penalties) and drives safely to its destination (rewards).

1. **Literature Review Summary Table**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *Authors and Year (Reference)* | *Title (Study)* | *Concept / Theoretical model/ Framework* | *Methodology used/ Implementation* | *Dataset details/ Analysis* | *Relevant Finding* | *Limitations/ Future Research/ Gaps identified* |
| Yuchi Tian, Kexin Pew, Suman Jana, Baishakhi Ray, 2017 | DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars | Developing a systematic tool for automatically detecting erroneous behaviour of Deep Neural Network driven vehicles. | Systematic exploring of different parts of  the Deep Neural Network logic by generating test inputs that maximize the numbers  of activated neurons. | Generating realistic synthetic images by applying image transformations on seed images and mimic different real-world phenomena like camera lens distortions, object movements, different weather conditions, etc. | Accuracy of a DNN can be improved by up to 46% by retraining the DNN with synthetic data generated by DeepTest. | Transformations are not designed to be exhaustive and therefore may not cover all realistic cases. |
| Lex Fridman∗, Daniel E. Brown, Michael Glazer et al, 2017 | Large-Scale Deep Learning Based Analysis of  Driver Behavior and Interaction with Automation | Understanding of how human beings interact with vehicle automation technology by integrating video data with vehicle state data, driver characteristics, mental models,  and self-reported experiences with technology | Large-scale  computer vision based analysis of human behavior | Study months to-date: 21  Participant days: 7,146  Drivers: 78  Vehicles: 25  Miles driven: 275,589  Video frames: 3.48 billion | Break new ground in  offering insights into how human and autonomous vehicles  interact in the rapidly changing transportation system. | Defines and inspires the next generation of naturalistic driving studies. |
| Mariusz Bojarski, Davide Del Testa et al, 2016 | End to End Learning for Self-Driving Cars | With minimum training data from humans the system learns to drive in traffic on local roads with or without lane markings and on highways. | Convolutional neural network (CNN) to map raw pixels from a sin-  gle front-facing camera directly to steering commands | Training data was collected by driving on a wide variety of roads and in a diverse set of lighting  and weather conditions. Most road data was collected in central New Jersey, although highway data  was also collected from Illinois, Michigan, Pennsylvania, and New York. | A small amount of training data from less than a hundred hours of driving  was sufficient to train the car to operate in diverse conditions, on highways, local and residential  roads in sunny, cloudy, and rainy conditions. | More work is needed to improve the robustness of the network, to find methods to verify the robust-  ness, and to improve visualization of the network-internal processing steps. |
| Chenyi Chen, Ari Seff, Alain Kornhauser, Jianxiong Xiao, 2015 | DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving | A direct perception approach  to estimate the affordance for driving. | Mapping an input image to a small number of key perception indicators that directly relate to the affordance of a road/ traffic state for driving. This representation provides a set of compact yet complete descriptions of the scene to enable a simple controller to drive autonomously. | Training  a deep Convolutional Neural Network using recording from  12 hours of human driving in a video game | Results show that  The direct perception approach can generalize well to real  driving images. | Regressors turn out to be far less accurate than the projection. |
| Mohammed Al-Qizwini, Iman Barjasteh, Hothaifa Al-Qassab and Hayder Radha, 2017 | Deep Learning Algorithm for Autonomous Driving using GoogLeNet | Introducing a new, more robust, and more realistic Direct Perception framework and corresponding algorithm for autonomous driving. | No unrealistic assumptions about the autonomous vehicle or its surroundings, and it uses only five affordance parameters to control the vehicle as compared to the 14 parameters used by prior efforts. | Trained each CNN for 300,000 iterations and selected the  model with minimum error. During the test phase, we present  25,000 random “unseen” images from the TORCS simulator  and have each model predict the affordance parameters. | Model is significantly  improved relative to the previous effort, which struggles to  navigate the whole track. The reason for this improvement is  due to removing the overlapped and redundant affordance  parameters | For future directions, we are planning to include other  parameters to the driving equations, such as human driver  fault tolerance for accident avoidance. We are also looking into adapting the TORCS environment to represent city  environment driving and also modifying our algorithm to work with pedestrians, bikes, motorcycles and animals  detection. |
| Ramon Iglesias,Federico Rossi,Kevin Wang, David Hallac,Jure Leskovec,Marco Pavone | Data-Driven Model Predictive Control of  Autonomous Mobility-on-Demand Systems | This paper is presents an end-to-  end, data-driven framework to control Autonomous Mobility-  on-Demand systems (AMoD, i.e. fleets of self-driving vehicles). | MPC algorithm that relies on forecasted demand to control  AMoD systems in real-time. | Numerical simulations based on  real-world data show that the algorithm scales well to largesystems and outperforms state-of-the-art rebalancing strate-  gies. | Results show that the incorporatingforecasted demand in the synthesis of a rebalancing algorithm can yield very significant improvements in customer  satisfaction, with 89.6% shorter customer wait times. | Extend this approach to incorporate other  relevant and promising aspects of AMoD systems, |
| Shuyang Du  Shuyangd,  Haoli Guo  Haoliguo,  Andrew Simpson | Self-Driving Car Steering Angle Prediction Based on Image Recognition | Predict steering angle based on only the provided images. | Two different models to perform high quality  prediction of steering angles based on images using differ-  ent deep learning techniques including Transfer Learning,  3D CNN, LSTM and ResNet. | The motivation of the project is to eliminate the need for  hand-coding rules and instead create a system that learns  how to drive by observing. | The paper have shown that  both transfer learning and a more advanced architecture  have promise in the field of autonomous vehicles. | Limited by computational resources, |
| April Yu, Raphael Palefsky-Smith, Rishi Bedi | Deep Reinforcement Learning for Simulated Autonomous Vehicle Control | Use of Deep Q-Learning to control a  simulated car via reinforcement learning. | Various reward functions to induce specific driving be-  havior, double Q-learning, gradient update rules, and other  hyperparameters. | Train an agent to con-  trol the simulated car in JavaScript Racer | Learned the turning oper-  ation, progressively gaining the ability to navigate larger  sections of the simulated raceway without crashing. | More improvements to Q-learning, techniques in  the same vein as Double Q-Learning that are relatively sim-  ple to implement |
| Xinlei Pan, Yurong You, Ziyan Wang zy- Cewu Lu2 lu | Virtual to Real Reinforcement Learning for Autonomous Driving | The proposed network can convert non-realistic virtual image  input into a realistic one with similar scene structure. | Given realistic frames as input, driving policy trained by reinforcement learning can nicely adapt to real world driving. | Experiments show that our proposed virtual to real (VR) reinforcement learning (RL) works pretty well | By using synthetic real images as training data in reinforcement learning, the agent generalizess better in a real environment than pure training with virtual data or training with domain randomization | The next step is to design a better image-to-image translation network and a better reinforcement learning framework to surpass the performance of supervised learning |
| Lu Chi, Yadong Mu | Deep Steering: Learning End-to-End Driving Model from Spatial and Temporal Visual Cues | Experimental study is based on about 6 hours of human driving data provided by Udacity. | Autonomous driving, convolutional LSTM, deep networks, deep steering | Data provided by Udacity. | Comprehensive quantitative evaluations demonstrate the effectiveness and robustness of our model, even under scenarios like drastic lighting changes and abrupt turning | Autonomous steering is still in its very early days and there are a number of challenges to be ironed out before the technique is employed on real cars |

1. **Objective of the project:**

Machine Learning impacts human life in many ways nowadays. An example is in the auto industry; many companies are trying to make their cars smarter. The cars can self-drive, avoid obstacles; find destinations without controls from human. The approach use Machine Learning, Neural Network and a Deep Q learning network to train the car by making the car learn after each time it fails to finish the track.

**4. Innovation component in the project:**

Deep Q Network: We present the first deep learning model to successfully learn control policies directly from high-dimensional sensory input using reinforcement learning. The model is a convolutional neural network, trained with a variant of Q-learning, whose input is raw pixels and whose output is a value function estimating future.

Monte-Carlo Search: The focus of Monte Carlo tree search is on the analysis of the most promising moves, expanding the [search tree](https://en.wikipedia.org/wiki/Search_tree) based on random sampling of the search space. The application of Monte Carlo tree search in games is based on many *playouts*. In each playout, the game is played out to the very end by selecting moves at random. The final game result of each playout is then used to weight the nodes in the game tree so that better nodes are more likely to be chosen in future playouts.

**5. Work to be done and implementation**

**Methodology to be adapted:**

Autonomous driving requires the integration of information from multiple sensors. The useful information needed to achieve the autonomous driving task is of much lower dimension. If this relevant information can be extracted, while the other non-relevant parts are filtered out, it would improve both the accuracy and efficiency of autonomous driving systems. Moreover, this would reduce the computation and memory requirements of the system.

In the proposed model, we consider a car, which is given a particular path. The car can move in right, left, forward and backward direction. So, the car will first try a particular move and if it encounters an obstacle it changes its path until it runs smoothly without any obstacle as a normal human would have done. It would continue doing this for all the moves. So, in our simulation of self-driving car for information retrieval we are going to use three camera angles i.e. left, centre and right while training the car how to run in the given environment. From the retrieved dataset the car would get to learn how to act on its own in the provided scenario and so as a bot would not need to get a command from others to move.

**Sofware requirements:**

Anaconda- Python 2.7  
Scikit

**Dataset used:**

The dataset will be generated by the program using reinforced learning. We will be first training the program and during this training process a dataset will be generated that will later drive the Machine Learning code.

**Tools used**

Keras  
Kivy  
Scikit  
Numpy

**6. Expected Results**

Using Machine Learning and Neural Network the car will overcome all the obstacles that it confronts throughout the travel and will memorize the correct moves that it follows to reach the destination just like a normal human.Introducing a framework for end-end autonomous driving using deep reinforcement learning to the automotive community.

**7. References**

1. Tian, Y., Pei, K., Jana, S., & Ray, B. (2017). DeepTest: Automated testing of deep-neural-network-driven autonomous cars. arXiv preprint arXiv:1708.08559.

2. Fridman, L., Brown, D. E., Glazer, M., Angell, W., Dodd, S., Jenik, B., ... & Abraham, H. (2017). MIT Autonomous Vehicle Technology Study: Large-Scale Deep Learning Based Analysis of Driver Behavior and Interaction with Automation. *arXiv preprint arXiv:1711.06976*.

3. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., ... & Zhang, X. (2016). End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*.

4. Chen, C., Seff, A., Kornhauser, A., & Xiao, J. (2015). Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2722-2730).

5. Al-Qizwini, M., Barjasteh, I., Al-Qassab, H., & Radha, H. (2017, June). Deep learning algorithm for autonomous driving using GoogLeNet. In *Intelligent Vehicles Symposium (IV), 2017 IEEE* (pp. 89-96). IEEE.

6. Iglesias, R., Rossi, F., Wang, K., Hallac, D., Leskovec, J., & Pavone, M. (2017). Data-Driven Model Predictive Control of Autonomous Mobility-on-Demand Systems. *arXiv preprint arXiv:1709.07032*.

7. Du, S., Guo, H., & Simpson, A. Self-Driving Car Steering Angle Prediction Based on Image Recognition.

8. Yu, A., Palefsky-Smith, R., & Bedi, R. (2016). Deep Reinforcement Learning for Simulated Autonomous Vehicle Control. *Course Project Reports: Winter*, 1-7.

9. You, Y., Pan, X., Wang, Z., & Lu, C. (2017). Virtual to Real Reinforcement Learning for Autonomous Driving. *arXiv preprint arXiv:1704.03952*.

10. Chi, L., & Mu, Y. (2017). Deep Steering: Learning End-to-End Driving Model from Spatial and Temporal Visual Cues. *arXiv preprint arXiv:1708.03798*.