# Benchmarking Autonomous Driving Agents with DeepDrive

## Harvard CS 249 (Fall 2019)

*Michael S. Emanuel*
*Hari Kothapalli*
*Bryan Lee*
*Tasha Schoenstein*

## Introduction

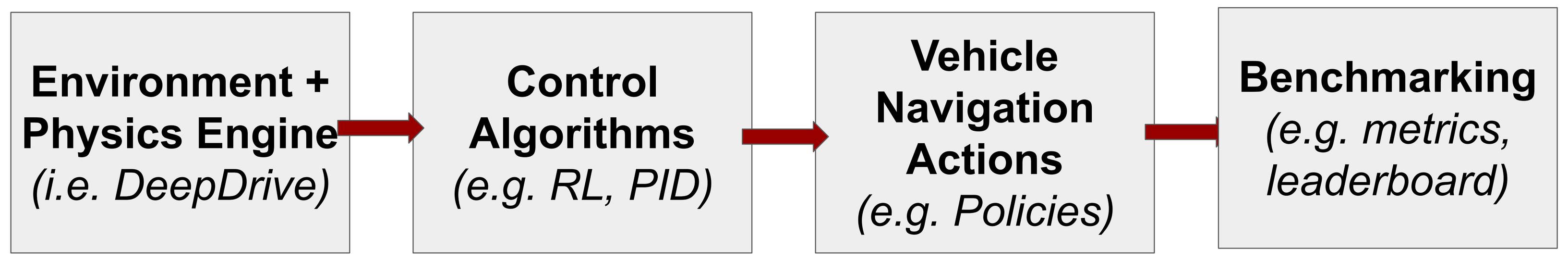
- Autonomous vehicles are an important challenge in both academia and industry
- Benchmarks for performance however are very limited
- The closest the industry has to unified performance metrics is the average miles per disengagement
- Previous work has been conducted to benchmark the performance of UAVs (drones), measuring task performance; compute speed; energy use
- **Our goal:** a first pass at creating infrastructure to benchmark autonomous driving agents

## Algorithms / System Description

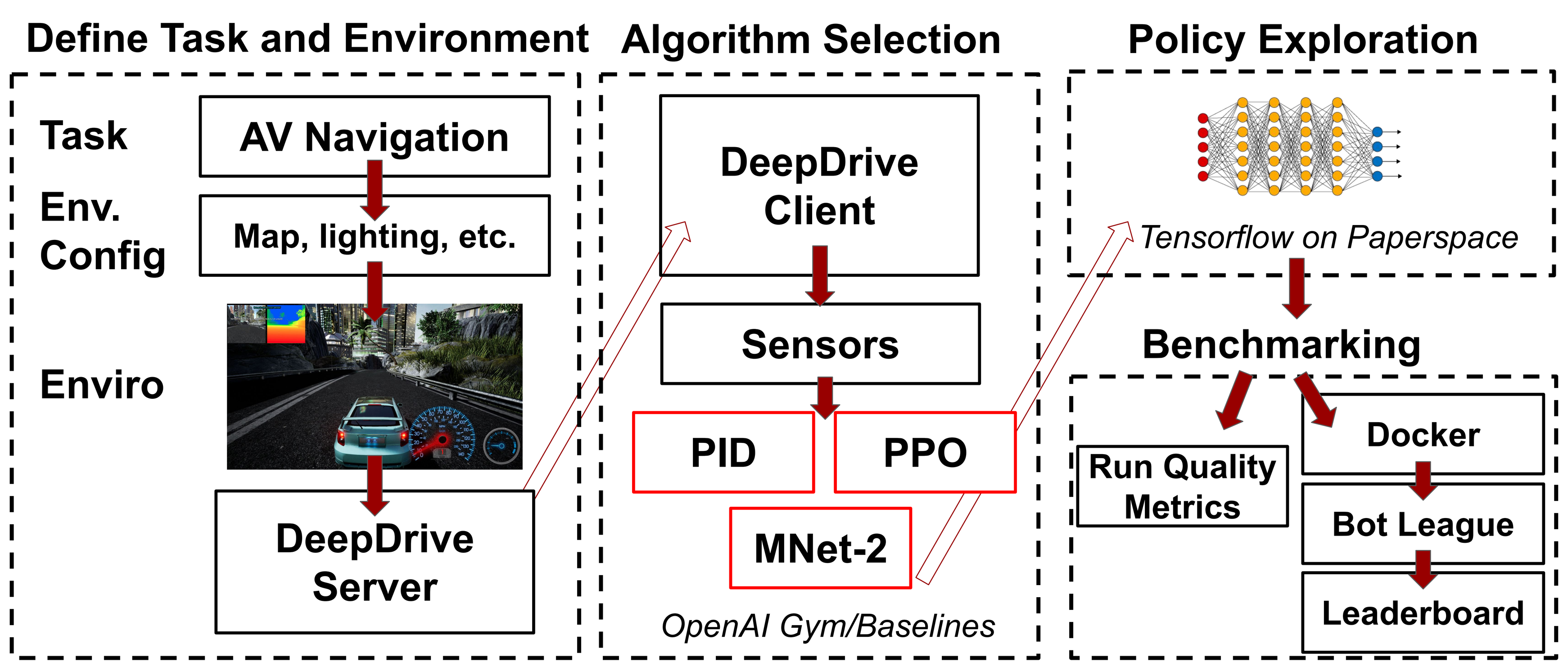
- We used the **DeepDrive** environment from **Voyage**
- It includes a high quality driving simulator that renders scenes in **UE4** (Unreal Engine) with simulated camera & LIDAR input
- ML agents can be trained using **OpenAI gym**
- Agents can run as clients against an environment server
- Voyage is running a competition on unprotected left turns
- Entries are evaluated using **Docker** containers

*Screenshot of DeepDrive left turn problem*

*Screenshot of Kevindale map used for training*




*Components of our current AV benchmarking stack*



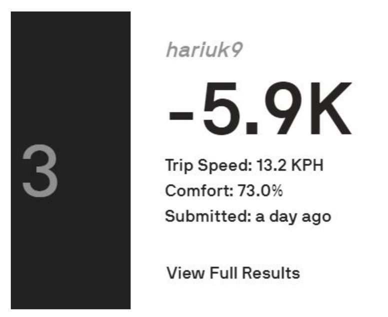

*Benchmarking Infrastructure with DeepDrive*

## Experiments

- **Agents:** forward (always on gas); lazy-driver (hand tuned left turn); PID controller; MobileNet v2; PPO agent
- **Contest:** we attempted to get a good score in the contest
- **Benchmarking:** we simulated 20 episodes for each agent and evaluated task performance and key metrics
- **Task Performance:** score from the RL environment. Key metrics: avoiding collision, progress, comfort (low g-force)
- **Computing Time:** average time required to take an action
- **Training:** we trained MNet2 and PPO agents from scratch

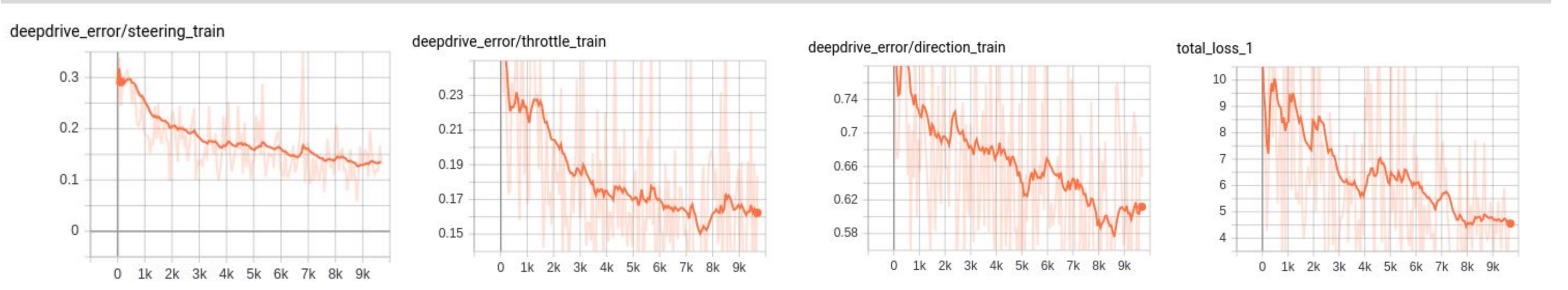
## DeepDrive Leaderboard Contest

- **Result:** our agent (trained MNet-2) achieved 3rd place and outperforms DeepDrive's baseline agent. Agent only loses to two brute force left-turn agents.
- **Scoring:** most leaderboard rankings are not generalizable (i.e. default forward agents)




*Leaderboard Ranking*

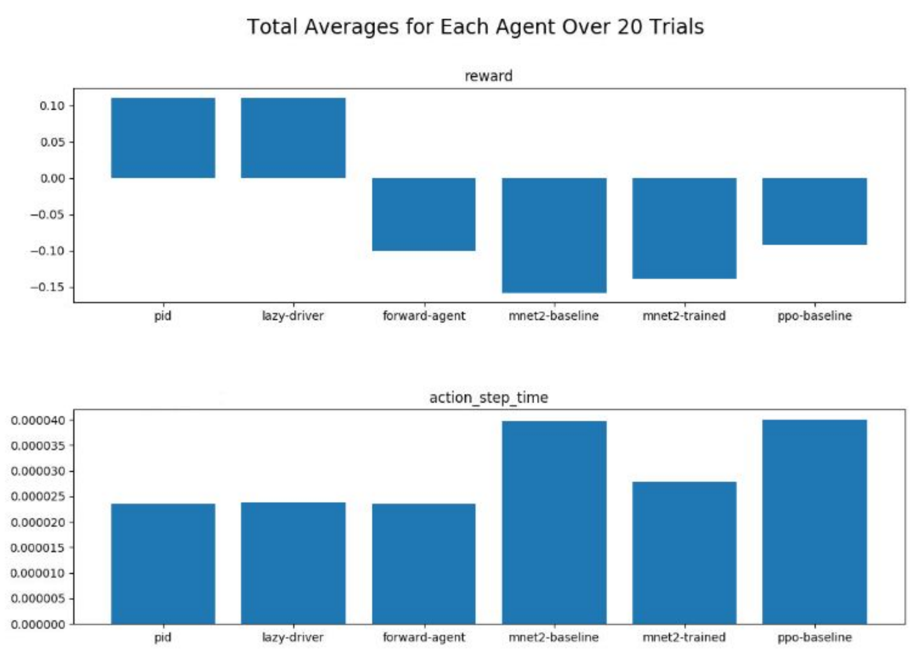
## Training




***Learning Curves:** steering, throttle, direction, and total loss on a training MNet2 Agent*

- Training shown was 10,000 steps / 2:45 (165 min)
- We did multiple training runs on mnet2 and ppo; we modestly outperformed the baseline after 3 GPU hours
- Training was done on 100 GB / 8.2 hours of driving by an "oracle" agent supplied by Voyage

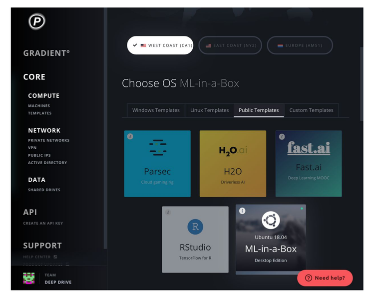
## Results




- **Six agents** were benchmarked on average reward and action time per step across 20 iterations for the left-turn task
- **Takeaway:** PID agent and the brute-force left turn agents obtained the highest average reward while the NN based agents (PPO and MNet) had the highest average action time

## Discussion

- **Computing:** Shared access to GPU servers for collaboration was a major challenge for our team: we selected cloud computing (via Paperspace, figure right)



- **Docker Containers:** The contest uses containers to separate the agents from the environment and allow competitors to design agents in any language / platform
- **Domain Randomization:** Contest winner was a hand tuned left turn, specific to the timing of the contest scenario; this shows the importance of domain randomization
- **PID Controller:** Strongest of the agents supplied by Voyage; although the scoring penalized it for high low comfort (high acceleration) it actually achieves the task
- **Mobilenet-v2:** Following multiple training attempts, we incrementally improved the baseline; however, the trained model did not reliably complete the ULT task
- **PPO**: This agent performed worse and was harder to train than the MNet2 agent, potentially due to sample efficiency

## Conclusions & Future Work

**Conclusions**

- **Preliminary Infrastructure:** We developed an initial infrastructure for benchmarking AV algorithms including environment simulation, algorithm selection, and benchmarking metrics
- **Algorithm Comparison**: Preliminary benchmarking indicates PID outperforms other agents in the left turn task in average reward. Average action time for PPO and MNet2 are highest due to their model complexity, as expected
- **Challenges of ROS:** AirLearning is built on an older version of ROS (Indigo) that only runs on Ubuntu 16.04. We failed to run AirLearning on a modern Ubuntu GPU server—a cloud-hosted VM would have worked better

**Future Work**

- **Infrastructure Extension:** Preliminary infrastructure can be extended to incorporate different environment configs (e.g. domain randomization), new OpenAI baselines (e.g. DDPG, DQN), and new benchmarking metrics (e.g. fuel consumption)
- **Hardware in the Loop (HIL):** To extend the work in MAVBench and AirLearning, HIL must be implemented, which requires more resources than were accessible to us
- **Study Goal: RL Agents or System Architecture?**
  - To benchmark system architecture and hardware, ROS and a simulator like AirSim is preferable
  - To benchmark algorithms sharing the same sensor suite of cameras and LIDAR, DeepDrive environment is preferable