Final Project Report - PPO Agent for CarRacing

1. Objective

The goal of this project was to train a Proximal Policy Optimization (PPO) agent to play the CarRacing-v3 environment from OpenAI Gym using visual input. The target was to develop an agent capable of completing laps efficiently and achieving high cumulative rewards.

2. Environment Setup

Environment: CarRacing-v3 (continuous control, visual observations)

Observation Space: 96x96 RGB frames

Action Space: Continuous, 3 dimensions: [steering, gas, brake]

Libraries Used: gymnasium[box2d], stable-baselines3, opencv-python, matplotlib, imageio

3. Preprocessing Pipeline

We applied minimal preprocessing since PPO with CnnPolicy can handle raw image input.

Environment Wrapping: FloatActionWrapper to ensure float64 actions.

Vectorized with DummyVecEnv for compatibility.

Observations: Directly using 96x96x3 RGB frames.

4. PPO Architecture & Hyperparameters

Policy: CnnPolicy (CNN for image input)

Algorithm: PPO from Stable-Baselines3

Learning rate: 3e-4

Clip range: 0.2

Batch size: 64

Discount factor: 0.99

Total timesteps: 200,000

GPU acceleration when available.

5. Training Process

The agent interacts with the environment for 200K timesteps.

Final Project Report - PPO Agent for CarRacing

PPO learns using clipped surrogate objective, balancing exploration and exploitation.

Checkpoints saved every 10,000 steps.

6. Evaluation

Benchmark: Average reward over 10 episodes.

Early training: Random exploration, spinning off-track.

Later training: Smooth driving, better curve handling, avoiding off-road.

Expected score after 1M+ steps: >= 900 (considered solved).

7. Reward Shaping (Not Applied)

Default CarRacing rewards: Positive for forward movement, negative for off-road.

Future improvement: Penalize high steering oscillations, reward smooth driving.

8. Results & Observations

Strengths: PPO with CNN learns effective driving strategies.

Challenges: Long training time due to high-dimensional input.

Final Output: Trained PPO model and recorded gameplay videos.

9. Demo Video

Recorded a 30-second gameplay clip using VecVideoRecorder, showing the agent navigating the track.

10. Conclusion & Future Work

Successfully implemented PPO-based vision agent for CarRacing.

Future improvements: Longer training, frame stacking, reward shaping, recurrent PPO, and double PPO variants.