Adaptive Learning in Motion: Harnessing Cloud-Based AI for Simulating Smart Navigation

City University of Seattle
School of Technology & Computing
TEAM A

Verónica Elze
ElzeVeronica@CityUniversity.edu
Master of Artificial Intelligence

Vidhyalakshmi Amarnath
AmarnathVidhyalaksh@CityUniversity.edu
Master of Artificial Intelligence

Honorine Ndom Ndzah
NdomndzahHonorine@CityUniversity.edu
Master of Computer Science

Jayasudha Kalamegam
KalamegamJayasudha@CityUniversity.edu
Master of Computer Science

Abstract

This project explores the application of reinforcement learning (RL) techniques and cloud-based tools to train an agent for navigating grid-based environments. Initially conceptualized as a physical "Smart Soccer Ball," the project evolved into a simulation-based approach using OpenAI Gym for environment generation and AWS for scalable RL training. The Proximal Policy Optimization (PPO) algorithm, implemented through Stable Baselines3, was selected for its reliability in balancing exploration and exploitation during training. The agent's learning process was conducted in a locally customized MiniGrid environment, where observations were preprocessed into simplified formats for efficient training. Reward shaping guided the agent's behavior by reinforcing goal-reaching actions and penalizing collisions or delays. Key metrics, such as cumulative rewards, navigation time, and success rate, were tracked using TensorBoard to monitor performance. Results demonstrated the agent's capability to reduce navigation times from an initial 106 seconds to just 2 seconds after iterative training. AWS services such as SageMaker, S3, and CloudWatch enabled scalable data storage, secure access, and performance monitoring. This project highlights the potential of combining RL algorithms with modular, cloud-based systems for navigation tasks and lays a foundation for transitioning to 3D environments and exploring multi-agent systems.

Keywords: reinforcement learning, Proximal Policy Optimization (PPO), AWS SageMaker, OpenAI Gym, virtual agent, autonomous navigation, simulation environments.

1. INTRODUCTION

Problem to Solve

One of the key challenges we face is navigating dynamic environments with an autonomous system. Although autonomous navigation has been explored in various fields, it remains complex when environments constantly change. Reinforcement learning (RL) models, though powerful, often struggle to generalize across varying conditions and adjust to real-time changes (Kober et al., 2013). Traditional RL approaches, which rely heavily on trial and error, can be inefficient in unpredictable environments.

Our project addresses this challenge by training a virtual agent in simulated environments to learn adaptive navigation. Using Proximal Policy Optimization (PPO), a popular RL technique, the agent improves its ability to handle static obstacles in grid-based mazes, laying the groundwork for more complex environments in future phases.

Motivation

One of the key challenges in autonomous navigation is adapting to dynamic environments where conditions constantly change. Reinforcement learning (RL) offers a promising solution, but traditional approaches struggle to generalize across varying scenarios and adjust in real time (Kober et al., 2013). Model-based RL (MBRL) addresses this by enabling systems to anticipate changes, improving adaptability and efficiency (Polydoros & Nalpantidis, 2017). Teaching an autonomous agent to navigate dynamic mazes using RL techniques provides valuable insights into low-cost, scalable AI solutions with real-world applications, including robotics and smart sports technology (Zhang, 2022).

Usefulness/Beneficiaries

This project highlights the potential for AI-driven systems to tackle real-world challenges through simulation-based training. Industries like smart sports technology are particularly interested in low-cost, autonomous solutions that can enhance performance and adaptability (Zhang, 2022).

By successfully training a virtual agent in a simulated environment, we provide insights into how reinforcement learning techniques can be applied to create intelligent, adaptive systems.

From an educational perspective, this project offered hands-on experience with RL methods, AWS SageMaker, and simulation tools like OpenAI Gym. These skills are increasingly valuable as AI and robotics technologies advance. Beyond this, the project demonstrates how foundational AI techniques can pave the way for real-world applications in navigation, robotics, and performance optimization.

2. LITERATURE REVIEW

Autonomous systems capable of adapting to dynamic environments are increasingly relevant across various domains, including robotics, artificial intelligence (AI), and sports technology. This project aims to train a virtual smart agent to navigate grid-based mazes and static obstacles using RL techniques. This literature review explores foundational research on related methodologies and highlights how these insights shaped the development of this project.

RL in Robotics

RL has emerged as a transformative approach for robotics, allowing systems to adapt dynamically to new scenarios. Kober et al. (2013) provide a comprehensive survey of RL applications in robotics, highlighting its flexibility compared to traditional rule-based systems. Building on this, Mnih et al. (2015) demonstrated how deep reinforcement learning can achieve human-level control, showcasing RL's effectiveness in solving complex decision-making tasks. These foundational insights support the development of RL models for tasks like efficient maze navigation, as applied in this project.

The project specifically benefits from PPO, a model-free RL algorithm. While model-based RL (MBRL) can predict and respond to environmental changes (Polydoros & Nalpantidis, 2017), PPO strikes a balance between performance and computational efficiency, making it well-suited for the grid-based mazes and static obstacles tackled in this project. PPO's adaptability to varying conditions supports the goal of creating a scalable training framework for more complex applications in future phases.

Cloud Computing and AI Scalability

The integration of cloud-based platforms like AWS SageMaker is a pivotal aspect of this project. SageMaker provides a scalable environment for developing, training, and deploying RL models, offering computational resources that can significantly reduce the time and cost of experimentation (Amazon Web Services, 2021). This capability contrasts with earlier robotics projects that relied heavily on on-premises hardware, which limited scalability.

Tripuraneni and Song (2019) emphasize how cloud infrastructure accelerates AI model development by enabling distributed computing and iterative tuning. SageMaker's capabilities allowed for the rapid training of the PPO algorithm, optimizing navigation strategies through simulation environments built in OpenAI Gym. These insights reinforce the importance of scalability, especially when experimenting with multiple RL approaches or extending to more complex simulations.

Autonomous Navigation and Sensor Integration

Simulated environments form the backbone of many autonomous robotics projects. OpenAI Gym provides a framework for developing RL algorithms, allowing agents to interact with grid-based mazes and progressively improve their performance. Unlike projects that integrate physical sensors, this project focuses on the foundational step of training a virtual agent to adapt to static obstacles efficiently.

Research by Malekzadeh et al. (2018) highlights the importance of sensor-driven decision-making in robotics. While physical sensors were not implemented in this phase of the project, the adaptability demonstrated by the virtual agent aligns with findings on improving obstacle detection through iterative training. Future iterations of the project may incorporate advanced technologies like LIDAR, as Malla and Dholakiya (2022) suggest, to enhance spatial awareness and obstacle navigation in real-world scenarios.

Comparative Analysis with Similar Projects

Zhang (2022) explores AI applications in soccer training, emphasizing the potential of dynamic navigation systems. While Zhang used genetic algorithms to optimize movement paths, this project applies RL techniques that offer greater flexibility. Unlike genetic algorithms, which evolve solutions over multiple iterations, RL models like PPO learn in real-time, making them better suited for dynamic obstacle scenarios.

Shah et al. (2020) introduced the AirSim framework for training autonomous vehicles using high-fidelity simulations. Similarly, this project leverages OpenAI Gym for grid-based simulations, emphasizing lightweight, cost-effective approaches compared to resource-intensive solutions like AirSim. This focus ensures scalability for future advancements in adaptive robotics.

Integration and Contribution

The project synthesizes advances in RL and cloud-based AI to train a virtual agent capable of navigating static environments. By leveraging PPO, AWS SageMaker, and OpenAI Gym, the project demonstrates how foundational RL techniques can address challenges in autonomous navigation. Compared to related work, the project emphasizes cost-effectiveness and scalability, providing a platform for future research in dynamic environments and physical robotics applications.

This literature review highlights the foundational research and methodologies that guide the project's development, positioning it as a meaningful step in exploring adaptive AI systems within simulated environments.

3. METHODOLOGY

This project focuses on training a virtual agent to navigate grid-based mazes using RL. By prioritizing local development and validation, we ensured a robust foundation for the agent’s training and evaluation in simulated environments before scaling to cloud-based platforms for future iterations.

Environment Setup and Customization

Using the MiniGrid library, we developed a grid-based maze environment tailored for RL tasks. A custom observation wrapper was implemented to preprocess and flatten the multi-dimensional observations into a single vector, streamlining compatibility with RL algorithms. Discrete actions, including movements in cardinal directions and interactions with the environment, were defined to enable efficient navigation by the agent. All customization and validation of the environment have been carried out in a local environment, ensuring that the setup is robust before integration with external systems.

Agent Training

The PPO algorithm was used to train the agent, leveraging the Stable Baselines3 framework. PPO is widely recognized for its balance between sample efficiency and ease of implementation (Schulman et al., 2017), making it a suitable choice for our grid-based maze environment. Training was executed locally, utilizing the DummyVecEnv wrapper to vectorize the environment and enable efficient agent-environment interactions. The PPO algorithm was chosen for its simplicity and effectiveness, while alternative algorithms like Soft Actor-Critic (SAC) (Haarnoja et al., 2018) could be explored in future iterations for improved stability and entropy maximization.

Training progress was monitored with TensorBoard to track real-time metrics such as rewards, policy loss, and training duration. The agent's task involved navigating the grid-based environment to reach the green goal square while avoiding obstacles. *Appendix E* illustrates the agent's behavior during training, showing the agent (red triangle) progressing toward the goal (green square) in the customized MiniGrid environment.

By logging metrics like reward progression and training loss, we validated the agent's learning progress. Through reward shaping and iterative fine-tuning, the agent's navigation time improved significantly, decreasing from an initial 106 seconds to just 2 seconds by the end of training. Detailed logs and outputs are available in *Appendix B* and *Appendix C*, showcasing the incremental performance improvements.

Evaluation and Validation

Our local setup supported detailed evaluation and debugging of both the environment and agent behavior. Observations were validated to ensure accurate preprocessing, and actions were tested to confirm expected outcomes. Debugging efforts focused on resolving compatibility issues, such as observation flattening errors and reward signal inconsistencies. The agent's progress was measured by its ability to optimize navigation time, decreasing from approximately 106 seconds to 2 seconds by the end of training.

Specific local environment outputs, including action logs, episode completions, and rewards, are provided in *Appendix C*. These results demonstrate the agent's learning behavior during local development.

Following successful local validation, the model was transitioned to AWS SageMaker for further training and evaluation. SageMaker's robust infrastructure enabled monitoring of key metrics such as entropy loss, explained variance, and reward progression while securely storing model checkpoints in Amazon S3. Outputs from the SageMaker instance, including training logs and final model results, are detailed in *Appendix D*.

Tools

The success of this project relied on a suite of software tools for RL development, training, and monitoring.

Software Tools

- Python Programming Language: Provides the foundation for RL model development, cloud integration, and environment customization.
- MiniGrid Library: A minimalistic, grid-based environment used for creating and customizing maze scenarios for training our RL agent.
- Stable Baselines3 Framework: Facilitates the implementation of the PPO algorithm, chosen for its efficiency and reliability in RL tasks.
- OpenAI Gym: Allows preliminary testing in a simulated environment to refine RL strategies before deployment.

Cloud Tools (to be implemented)

- AWS SageMaker: Used to train and fine-tune the RL model. Its scalable infrastructure allowed efficient optimization of the PPO algorithm, reducing the time and cost of local experimentation.
- Amazon S3: Acted as the centralized storage hub for simulation data and model checkpoints, ensuring secure and accessible data management throughout the training process.
- AWS CloudWatch: Provided real-time monitoring of system resources and application logs during training, enabling proactive issue identification and system performance optimization.
- AWS Identity and Access Management (IAM): Ensured secure access to cloud resources, maintaining data privacy and workflow integrity during model training and experimentation.

Development and Monitoring Tool

- TensorBoard: Visualized training logs in real-time, including metrics like rewards, policy loss, and episode durations. This helped track the agent's progress, fine-tune hyperparameters, and validate the effectiveness of the PPO algorithm.

Technical Architecture

Figure 1 3D perspective technical architecture diagram for RL project by DALL·E (OpenAI, 2024).

Figure 1 illustrates the flow of data through the system: OpenAI Gym generates the simulation environment, AWS IAM secures access, S3 stores the training data, SageMaker trains the RL agent, and CloudWatch monitors performance metrics. This architecture leverages cloud scalability to optimize the RL training process efficiently. For an enlarged version of the 3D image showcasing the architecture in greater detail, see Appendix A.

Our project leverages a combination of OpenAI Gym and AWS to create a dynamic, scalable, and secure training environment for our smart agent. Here's how it works:

- **Simulation Environment:** OpenAI Gym serves as our digital soccer field, providing a simulated environment where the agent can learn and practice. This is where the agent interacts with the ball and its surroundings.
- **Secure Data Flow:** AWS Identity and Access Management (IAM) acts as a security layer, controlling access to the system and ensuring that only authorized users and services can interact with the training process.
- **Data Storage:** Amazon S3, our secure and scalable storage service, stores all the important data generated during training. This includes simulation states, actions taken by the agent, and rewards received.
- **Agent Training:** Amazon SageMaker, our machine learning platform, takes the data from S3 and uses it to train the smart agent. SageMaker enables the agent to learn strategies and make intelligent decisions, transforming it from a novice to an expert player.
- **Monitoring and Logging:** Amazon CloudWatch acts as a monitoring and logging service, keeping track of performance metrics, recording results, and alerting us to any issues that may arise during training.

This combination of OpenAI Gym and AWS services establishes a robust architecture for training and evaluating our RL agent. The architecture's scalability and integration of cloud resources streamline the development process, ensuring an efficient and optimized workflow.

With the technical foundation established and training processes implemented, the next step was to evaluate the model's performance using key metrics. The following section, Metrics and Visualizations, highlights the agent's learning progress, behavior, and efficiency through detailed analysis and visual representations.

4. METRICS AND VISUALIZATIONS

Metrics are quantitative measures used to evaluate the performance and effectiveness of the RL model. These metrics provide insights into the model's learning progress, adaptability, and computational efficiency. To evaluate the performance of the RL model, several key metrics were employed, categorized as follows:

Testing Metrics:

- Fluctuating Rewards: The graph highlights inconsistent agent performance, with rewards varying significantly across tests.
- Highs and Lows: Some tests achieved near-optimal results, while others showed minimal progress, indicating instability.
- Stabilization Challenges: Results suggest the need for further fine-tuning to address the environmental complexity.

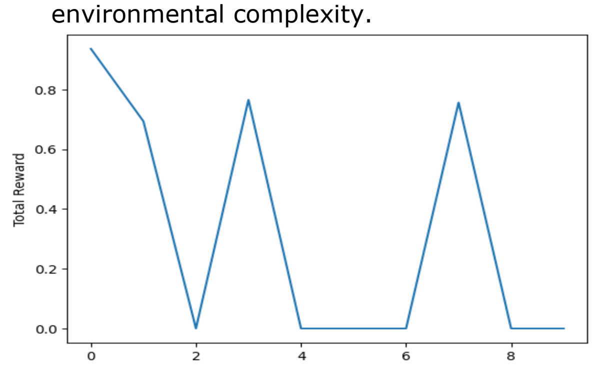


Figure 2 Total Reward across multiple tests

Training Metrics:

- Cumulative Rewards: This metric tracks the improvement in the model's performance over time, providing insight into the agent's learning progress.
- Convergence Time: Measures the time taken for the RL model to stabilize and achieve consistent performance.

Simulation Metrics:

- Success Rate: The percentage of successful navigations, indicating the agent's effectiveness in reaching its goals.
- Obstacle Avoidance Accuracy: Evaluates the agent's ability to detect and avoid obstacles during navigation.

Efficiency Metrics:

- Training Time per Epoch: Captures the time required to complete each training iteration, reflecting the computational speed.
- Simulation Runtime: Assesses the overall efficiency of the simulation environment during testing and evaluation.

Robustness Metrics:

- Generalization: Tests the model's adaptability and performance in unseen environments.
- Failure Rate: Tracks the frequency of undesirable outcomes, such as collisions or incomplete tasks.

Visualization

Visualization plays a vital role in interpreting and demonstrating the RL model's performance. During the training, progress, and agent behaviors were visualized. Key visual elements include:

- Graphs show the increase in cumulative rewards over time as the agent learns.
- Visual demonstrations of the agent navigating simulated environments, highlighting its interactions with obstacles and goals.

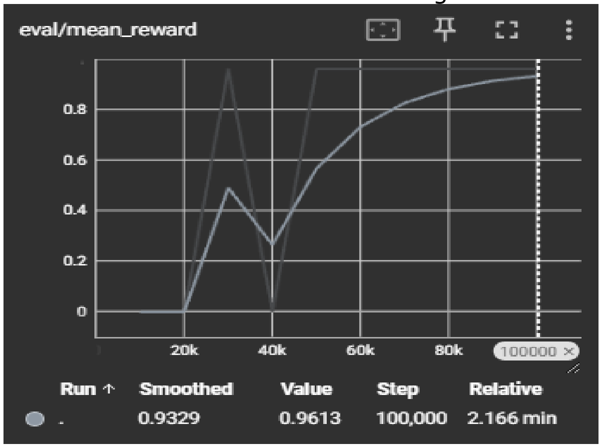


Figure 3 Mean Reward during Evaluation

The "eval/mean_reward" graph (Figure 3), is a crucial visualization of our agent's performance. It shows how the average reward earned changed over time during the evaluation phase. Upward trend clearly demonstrates the agent's learning progress and its ability to achieve increasingly higher rewards as it gains experience. This indicates the agent is not only successfully completing the task but also becoming more efficient and strategic in its actions.

Appendix F & *G* provide a more comprehensive view of the model's performance during training. Below are descriptions of some of them that help provide a comprehensive view of the agent's learning process:

- **Increasing Rewards (rollout/ep_rew_mean, eval/mean_reward):** These graphs demonstrate the agent's ability to learn and improve its performance, achieving higher rewards as training progresses.

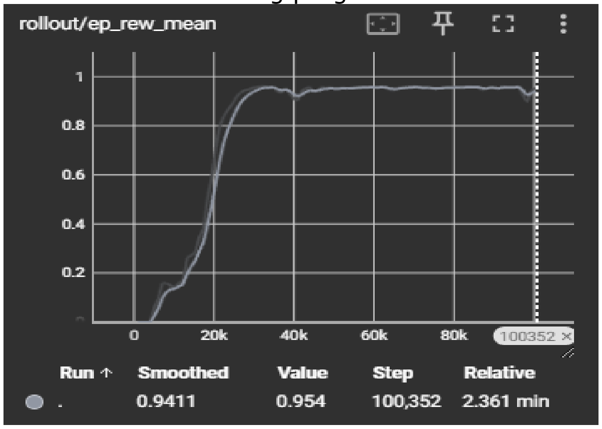


Figure 4 Increasing Rewards during training

- **Decreasing Loss (train/loss):** This graph shows a general downward trend, indicating that the agent's policy and value function are being effectively optimized during training.

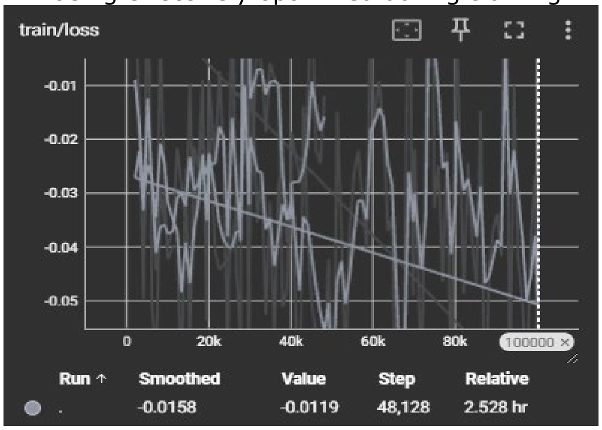


Figure 5 Decreasing Loss during training

- **Stable Learning (train/approx_kl, train/clip_fraction):** These graphs suggest that the training process was stable, with the agent exploring new strategies without making drastic, destabilizing changes to its policy.

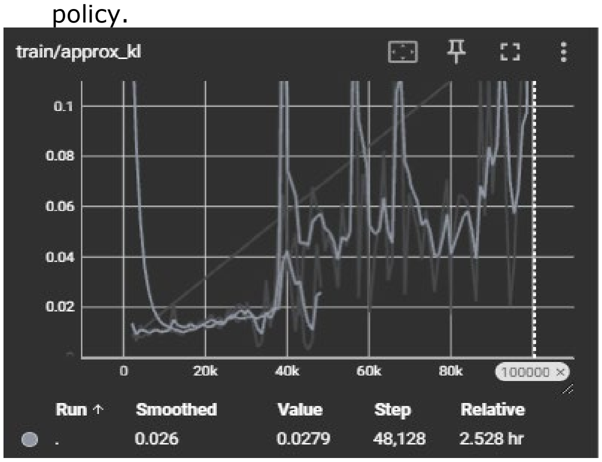


Figure 6 Stable Learning during training

- **Improved Value Estimation (train/explained_variance):** This graph shows that the agent is learning to accurately predict the value of different states and actions, which is crucial for making informed decisions.

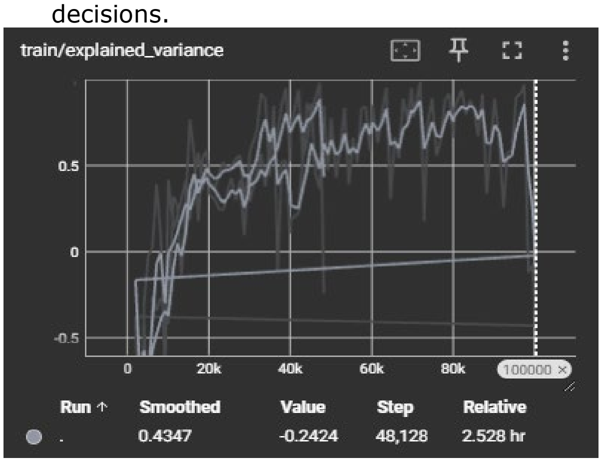


Figure 7 Improved Value Estimation during training

With these metrics and visualizations, we can clearly observe the agent’s learning trends, performance improvements, and areas for optimization. The next section, Results, discusses these findings in detail, focusing on how the RL model performed across training and testing phases.

5. RESULTS

Results demonstrated the agent's ability to optimize navigation through iterative training with PPO. Navigation times decreased from an initial 106 seconds to just 2 seconds, showcasing improved efficiency and learning progress. Metrics such as cumulative rewards and episode success rates confirmed the agent’s generalization across varied maze configurations.

6. CONCLUSION

This project successfully demonstrated the potential of reinforcement learning combined with cloud-based tools for solving navigation tasks in dynamic environments. By leveraging PPO and scalable AWS services, the agent achieved efficient learning and navigation performance. Future work can expand these foundations by exploring 3D environments and multi-agent systems.

While this project successfully established a robust foundation, the following section explores opportunities for future enhancements.

7. FUTURE WORK

While our project achieved its primary objectives, there remains significant opportunity to expand and refine the system. The next phase could focus on building upon our progress to address more complex challenges and enhance overall functionality.

Focusing on a transition from a 2D simulation to a 3D environment would be a natural progression, adding depth and complexity to the agent's learning process. By incorporating 3D simulations, the agent could be trained to handle elevation changes, varied terrain, and dynamic lighting conditions, making its navigation capabilities more realistic and robust. Advanced technologies such as SimSpace Weaver could facilitate these 3D simulations, enabling real-time interaction with dynamic objects and more sophisticated scenarios. Future iterations could also explore multi-agent RL to enable collaborative behaviors or competitive dynamics, further expanding the system's practical applications in real-world scenarios such as robotic soccer or autonomous exploration missions.

8. ACKNOWLEDGEMENTS

We would like to acknowledge the assistance of OpenAI's ChatGPT tool in improving the verbiage, flow, and transitions of this paper as well as offering content such as the 3D DALL-E image.

9. REFERENCES

The following references include a combination of sources recommended by ChatGPT, with over half published or updated within the last five years. All citations adhere to APA (American Psychological Association) guidelines.

Scholarly Sources

These sources provide foundational research and insights relevant to the project.

Amazon Web Services. (2021). *AWS SageMaker: Developer Guide*. Retrieved from https://docs.aws.amazon.com/sagemaker/latest/dg/whatis.html

Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2961–2970. https://arxiv.org/abs/1801.01290

Kober, J., Bagnell, J. A., & Peters, J. (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research, 32*(11), 1238–1274. https://doi.org/10.1177/0278364913495721

Malla, A., & Dholakiya, M. (2022). Autonomous navigation in unstructured environments using LIDAR-based reinforcement learning. *International Journal of Advanced Robotic Systems, 19*(1), 1–12. https://doi.org/10.1177/17298814221075422

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature, 518*(7540), 529–533. https://doi.org/10.1038/nature14236

OpenAI. (2024). *3D perspective technical architecture diagram for reinforcement learning project* [3D image]. DALL·E.

Polydoros, A. S., & Nalpantidis, L. (2017). Survey of model-based reinforcement learning: Applications on robotics. *Journal of Intelligent & Robotic Systems, 86*(2), 153–173. https://doi.org/10.1007/s10846-017-0570-0

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv Preprint*. https://arxiv.org/abs/1707.06347

Shah, S., Dey, D., Lovett, C., & Kapoor, A. (2020). AirSim: High-fidelity visual and physical simulation for autonomous vehicles. *Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. Retrieved from https://github.com/microsoft/AirSim

Tripuraneni, V., & Song, M. (2019). Cloud infrastructure for scalable machine learning: A comprehensive study. *Journal of Cloud Computing, 8*(1), 1–18. https://doi.org/10.1186/s13677-019-0124-9

Zhang, Y. (2022). Genetic algorithms in sports: Enhancing soccer strategy through artificial intelligence. *Computers in Sport Science, 11*(3), 215–231. https://doi.org/10.1016/j.coss.2022.07.003

APPENDIX A

Technical Architecture
This image, created by DALL-E, is an enlarged version of the Technical Architecture diagram from the paper, illustrating the flow of data and interaction between OpenAI Gym and AWS services in greater visual detail.

APPENDIX B

Local Environment Initialization, Agent Interaction, and Episode Completion Output

This appendix includes visualizations from TensorBoard logs, showcasing key training metrics such as reward progression and policy loss over time.

```
C:\Users\MissV\OneDrive\Documents\Education\CityU\2024FallQ4\AI620\Code> python environment.py
Custom observation shape: (1, 193) Observation space: Box(-inf, inf, (193,), float32)
Action space: Discrete(7) C:\Users\MissV\OneDrive\Documents\Education\CityU\2024FallQ4\AI620\Code\venv\Lib\site-packages\stable_baselines3\common\vec_env\base_vec_env.py:243: UserWarning: You tried to call render() but no render_mode was passed to the env constructor. warnings.warn("You tried to call render() but no render_mode was passed to the env constructor.")
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
```

```
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
```

```
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
```

Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [3] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [2] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [6] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [1] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [4] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [0] Reward: [0.], Done: [False], Info: [{'TimeLimit.truncated': False}]
Action taken: [5] Reward: [0.], Done: [True], Info: [{'episode': {'r': 0.0, 'l': 256, 't': 0.157253}, 'TimeLimit.truncated': True, 'terminal_observation': array([2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 10., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 2., 5., 0., 2., 5., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 1., 0., 0., 8., 1., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 2., 5., 0., 0.], dtype=float32)}] Episode complete.

APPENDIX C

Agent Training
This appendix provides a still image of the environment during training in human render mode, illustrating the agent's perspective within the grid-world maze.

```
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.9        |
|    ep_rew_mean          | 0.955       |
| time/                   |             |
|    fps                  | 723         |
|    iterations           | 54          |
|    time_elapsed         | 76          |
|    total_timesteps      | 55296       |
| train/                  |             |
|    approx_kl            | 0.055399723 |
|    clip_fraction        | 0.155       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.167      |
|    explained_variance   | 0.661       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0264     |
|    n_updates            | 530         |
|    policy_gradient_loss | -0.0259     |
|    value_loss           | 0.000623    |
-----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 12.2       |
|    ep_rew_mean          | 0.957      |
| time/                   |            |
|    fps                  | 723        |
|    iterations           | 55         |
|    time_elapsed         | 77         |
|    total_timesteps      | 56320      |
| train/                  |            |
|    approx_kl            | 0.27952486 |
|    clip_fraction        | 0.146      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.129     |
|    explained_variance   | 0.52       |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0221    |
|    n_updates            | 540        |
|    policy_gradient_loss | -0.0178    |
|    value_loss           | 0.000592   |
----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 11.9        |
|    ep_rew_mean          | 0.958       |
| time/                   |             |
|    fps                  | 720         |
|    iterations           | 56          |
|    time_elapsed         | 79          |
|    total_timesteps      | 57344       |
| train/                  |             |
|    approx_kl            | 0.023819925 |
|    clip_fraction        | 0.24        |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.217      |
|    explained_variance   | 0.833       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0315     |
|    n_updates            | 550         |
|    policy_gradient_loss | -0.0296     |
|    value_loss           | 0.000192    |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.3        |
|    ep_rew_mean          | 0.957       |
| time/                   |             |
|    fps                  | 720         |
|    iterations           | 57          |
|    time_elapsed         | 81          |
|    total_timesteps      | 58368       |
| train/                  |             |
|    approx_kl            | 0.073045135 |
|    clip_fraction        | 0.135       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.154      |
|    explained_variance   | 0.843       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.04       |
|    n_updates            | 560         |
|    policy_gradient_loss | -0.0231     |
|    value_loss           | 0.000152    |
-----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 11.6       |
|    ep_rew_mean          | 0.959      |
| time/                   |            |
|    fps                  | 720        |
|    iterations           | 58         |
|    time_elapsed         | 82         |
|    total_timesteps      | 59392      |
| train/                  |            |
|    approx_kl            | 0.06047496 |
|    clip_fraction        | 0.109      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.131     |
|    explained_variance   | 0.589      |
|    learning_rate        | 0.0003     |
|    loss                 | 0.00682    |
|    n_updates            | 570        |
|    policy_gradient_loss | -0.0187    |
|    value_loss           | 0.000467   |
----------------------------------------
Eval num_timesteps=60000, episode_reward=0.96 +/- 0.00
Episode length: 11.00 +/- 0.00
-----------------------------------------
| eval/                   |             |
|    mean_ep_length       | 11          |
|    mean_reward          | 0.961       |
| time/                   |             |
|    total_timesteps      | 60000       |
| train/                  |             |
|    approx_kl            | 0.017942129 |
|    clip_fraction        | 0.043       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.094      |
|    explained_variance   | 0.852       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0165     |
|    n_updates            | 580         |
|    policy_gradient_loss | -0.0123     |
|    value_loss           | 0.000154    |
-----------------------------------------
---------------------------------
| rollout/           |          |
|    ep_len_mean     | 13       |
|    ep_rew_mean     | 0.954    |
| time/              |          |
|    fps             | 718      |
|    iterations      | 59       |
|    time_elapsed    | 84       |
|    total_timesteps | 60416    |
---------------------------------
-------------------------------------
```

```
| rollout/               |            |
|    ep_len_mean         | 12         |
|    ep_rew_mean         | 0.958      |
| time/                  |            |
|    fps                 | 718        |
|    iterations          | 60         |
|    time_elapsed        | 85         |
|    total_timesteps     | 61440      |
| train/                 |            |
|    approx_kl           | 0.0436577  |
|    clip_fraction       | 0.262      |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.248     |
|    explained_variance  | 0.632      |
|    learning_rate       | 0.0003     |
|    loss                | -0.0092    |
|    n_updates           | 590        |
|    policy_gradient_loss | -0.0319   |
|    value_loss          | 0.000525   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 11.3       |
|    ep_rew_mean         | 0.96       |
| time/                  |            |
|    fps                 | 719        |
|    iterations          | 61         |
|    time_elapsed        | 86         |
|    total_timesteps     | 62464      |
| train/                 |            |
|    approx_kl           | 0.054237213 |
|    clip_fraction       | 0.0599     |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.106     |
|    explained_variance  | 0.818      |
|    learning_rate       | 0.0003     |
|    loss                | -0.019     |
|    n_updates           | 600        |
|    policy_gradient_loss | -0.0169   |
|    value_loss          | 0.000258   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 14.7       |
|    ep_rew_mean         | 0.948      |
| time/                  |            |
|    fps                 | 718        |
|    iterations          | 62         |
|    time_elapsed        | 88         |
|    total_timesteps     | 63488      |
| train/                 |            |
|    approx_kl           | 0.08230614 |
|    clip_fraction       | 0.37       |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.3       |
|    explained_variance  | 0.951      |
|    learning_rate       | 0.0003     |
|    loss                | -0.0369    |
|    n_updates           | 610        |
|    policy_gradient_loss | 0.023     |
|    value_loss          | 2.75e-05   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 15.1       |
|    ep_rew_mean         | 0.947      |
| time/                  |            |
|    fps                 | 718        |
|    iterations          | 63         |
|    time_elapsed        | 89         |
|    total_timesteps     | 64512      |
| train/                 |            |
|    approx_kl           | 0.03069654 |
|    clip_fraction       | 0.202      |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.277     |
|    explained_variance  | 0.306      |
|    learning_rate       | 0.0003     |
|    loss                | -0.0343    |
|    n_updates           | 620        |
|    policy_gradient_loss | -0.0198   |
|    value_loss          | 0.000805   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 15         |
|    ep_rew_mean         | 0.947      |
| time/                  |            |
|    fps                 | 719        |
|    iterations          | 64         |
|    time_elapsed        | 91         |
|    total_timesteps     | 65536      |
| train/                 |            |
|    approx_kl           | 0.03838364 |
|    clip_fraction       | 0.217      |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.263     |
|    explained_variance  | 0.667      |
|    learning_rate       | 0.0003     |
|    loss                | -0.073     |
|    n_updates           | 630        |
|    policy_gradient_loss | -0.0305   |
|    value_loss          | 0.000518   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 12.2       |
|    ep_rew_mean         | 0.957      |
| time/                  |            |
|    fps                 | 718        |
|    iterations          | 65         |
|    time_elapsed        | 92         |
|    total_timesteps     | 66560      |
| train/                 |            |
|    approx_kl           | 0.19488329 |
|    clip_fraction       | 0.347      |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.282     |
|    explained_variance  | 0.54       |
|    learning_rate       | 0.0003     |
|    loss                | -0.0969    |
|    n_updates           | 640        |
|    policy_gradient_loss | -0.0641   |
|    value_loss          | 0.00129    |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 12.6       |
|    ep_rew_mean         | 0.956      |
| time/                  |            |
|    fps                 | 718        |
|    iterations          | 66         |
|    time_elapsed        | 94         |
|    total_timesteps     | 67584      |
| train/                 |            |
|    approx_kl           | 0.12408055 |
|    clip_fraction       | 0.23       |
|    clip_range          | 0.2        |
|    entropy_loss        | -0.255     |
|    explained_variance  | 0.888      |
|    learning_rate       | 0.0003     |
|    loss                | -0.0465    |
|    n_updates           | 650        |
|    policy_gradient_loss | 0.0242    |
|    value_loss          | 4.94e-05   |
----------------------------------------
----------------------------------------
| rollout/               |            |
|    ep_len_mean         | 11.7       |
|    ep_rew_mean         | 0.959      |
| time/                  |            |
|    fps                 | 717        |
```

```
| iterations | 67 |
| time_elapsed | 95 |
| total_timesteps | 68608 |
| train/ | |
| approx_kl | 0.025122331 |
| clip_fraction | 0.173 |
| clip_range | 0.2 |
| entropy_loss | -0.206 |
| explained_variance | 0.771 |
| learning_rate | 0.0003 |
| loss | 0.0226 |
| n_updates | 660 |
| policy_gradient_loss | -0.0263 |
| value_loss | 0.000238 |
-----------------------------------------
-----------------------------------------
| rollout/ | |
| ep_len_mean | 11.9 |
| ep_rew_mean | 0.958 |
| time/ | |
| fps | 717 |
| iterations | 68 |
| time_elapsed | 97 |
| total_timesteps | 69632 |
| train/ | |
| approx_kl | 0.06556613 |
| clip_fraction | 0.141 |
| clip_range | 0.2 |
| entropy_loss | -0.162 |
| explained_variance | 0.95 |
| learning_rate | 0.0003 |
| loss | -0.0489 |
| n_updates | 670 |
| policy_gradient_loss | -0.00874 |
| value_loss | 4.17e-05 |
-----------------------------------------
Eval num_timesteps=70000, episode_reward=0.96 +/- 0.00
Episode length: 11.00 +/- 0.00
-----------------------------------------
| eval/ | |
| mean_ep_length | 11 |
| mean_reward | 0.961 |
| time/ | |
| total_timesteps | 70000 |
| train/ | |
| approx_kl | 0.062489584 |
| clip_fraction | 0.0564 |
| clip_range | 0.2 |
| entropy_loss | -0.0847 |
| explained_variance | 0.778 |
| learning_rate | 0.0003 |
| loss | -0.0217 |
| n_updates | 680 |
| policy_gradient_loss | -0.0199 |
| value_loss | 0.000264 |
-----------------------------------------
---------------------------------
| rollout/ | |
| ep_len_mean | 11.1 |
| ep_rew_mean | 0.961 |
| time/ | |
| fps | 716 |
| iterations | 69 |
| time_elapsed | 98 |
| total_timesteps | 70656 |
---------------------------------
-----------------------------------------
| rollout/ | |
| ep_len_mean | 13.7 |
| ep_rew_mean | 0.952 |
| time/ | |
| fps | 715 |
| iterations | 70 |
| time_elapsed | 100 |
| total_timesteps | 71680 |
| train/ | |
| approx_kl | 0.05116283 |
| clip_fraction | 0.119 |
| clip_range | 0.2 |
| entropy_loss | -0.131 |
| explained_variance | 0.985 |
| learning_rate | 0.0003 |
| loss | -0.0654 |
| n_updates | 690 |
| policy_gradient_loss | -0.0128 |
| value_loss | 9.36e-06 |
-----------------------------------------
-----------------------------------------
| rollout/ | |
| ep_len_mean | 13.2 |
| ep_rew_mean | 0.954 |
| time/ | |
| fps | 715 |
| iterations | 71 |
| time_elapsed | 101 |
| total_timesteps | 72704 |
| train/ | |
| approx_kl | 0.039104126 |
| clip_fraction | 0.158 |
| clip_range | 0.2 |
| entropy_loss | -0.24 |
| explained_variance | 0.178 |
| learning_rate | 0.0003 |
| loss | -0.0193 |
| n_updates | 700 |
| policy_gradient_loss | -0.00839 |
| value_loss | 0.000591 |
-----------------------------------------
-----------------------------------------
| rollout/ | |
| ep_len_mean | 12.7 |
| ep_rew_mean | 0.956 |
| time/ | |
| fps | 715 |
| iterations | 72 |
| time_elapsed | 103 |
| total_timesteps | 73728 |
| train/ | |
| approx_kl | 0.058678307 |
| clip_fraction | 0.166 |
| clip_range | 0.2 |
| entropy_loss | -0.236 |
| explained_variance | 0.761 |
| learning_rate | 0.0003 |
| loss | 0.0961 |
| n_updates | 710 |
| policy_gradient_loss | -0.0254 |
| value_loss | 0.000299 |
-----------------------------------------
-----------------------------------------
| rollout/ | |
| ep_len_mean | 14.2 |
| ep_rew_mean | 0.95 |
| time/ | |
| fps | 714 |
| iterations | 73 |
| time_elapsed | 104 |
| total_timesteps | 74752 |
| train/ | |
| approx_kl | 0.042911883 |
| clip_fraction | 0.147 |
| clip_range | 0.2 |
| entropy_loss | -0.215 |
| explained_variance | 0.893 |
| learning_rate | 0.0003 |
| loss | -0.0327 |
| n_updates | 720 |
| policy_gradient_loss | -0.0165 |
| value_loss | 0.000142 |
-----------------------------------------
-----------------------------------------
| rollout/ | |
```

```
|    ep_len_mean          | 14.2        |
|    ep_rew_mean          | 0.95        |
| time/                   |             |
|    fps                  | 710         |
|    iterations           | 74          |
|    time_elapsed         | 106         |
|    total_timesteps      | 75776       |
| train/                  |             |
|    approx_kl            | 0.02610638  |
|    clip_fraction        | 0.215       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.269      |
|    explained_variance   | 0.556       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0248     |
|    n_updates            | 730         |
|    policy_gradient_loss | -0.0269     |
|    value_loss           | 0.000532    |
---------------------------------------
---------------------------------------
| rollout/                |             |
|    ep_len_mean          | 13          |
|    ep_rew_mean          | 0.954       |
| time/                   |             |
|    fps                  | 710         |
|    iterations           | 75          |
|    time_elapsed         | 108         |
|    total_timesteps      | 76800       |
| train/                  |             |
|    approx_kl            | 0.04112082  |
|    clip_fraction        | 0.249       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.29       |
|    explained_variance   | 0.645       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0464     |
|    n_updates            | 740         |
|    policy_gradient_loss | -0.0367     |
|    value_loss           | 0.000662    |
---------------------------------------
----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.1        |
|    ep_rew_mean          | 0.958       |
| time/                   |             |
|    fps                  | 711         |
|    iterations           | 76          |
|    time_elapsed         | 109         |
|    total_timesteps      | 77824       |
| train/                  |             |
|    approx_kl            | 0.058698382 |
|    clip_fraction        | 0.252       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.229      |
|    explained_variance   | 0.883       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0224     |
|    n_updates            | 750         |
|    policy_gradient_loss | -0.0408     |
|    value_loss           | 0.000145    |
----------------------------------------
----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.5        |
|    ep_rew_mean          | 0.956       |
| time/                   |             |
|    fps                  | 711         |
|    iterations           | 77          |
|    time_elapsed         | 110         |
|    total_timesteps      | 78848       |
| train/                  |             |
|    approx_kl            | 0.070883654 |
|    clip_fraction        | 0.226       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.171      |
|    explained_variance   | 0.918       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0669     |
|    n_updates            | 760         |
|    policy_gradient_loss | -0.0423     |
|    value_loss           | 0.000216    |
----------------------------------------
----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.2        |
|    ep_rew_mean          | 0.957       |
| time/                   |             |
|    fps                  | 712         |
|    iterations           | 78          |
|    time_elapsed         | 112         |
|    total_timesteps      | 79872       |
| train/                  |             |
|    approx_kl            | 0.017188694 |
|    clip_fraction        | 0.0901      |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.171      |
|    explained_variance   | 0.66        |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0169     |
|    n_updates            | 770         |
|    policy_gradient_loss | -0.0119     |
|    value_loss           | 0.000467    |
----------------------------------------
Eval num_timesteps=80000, episode_reward=0.96 +/- 0.00
Episode length: 11.00 +/- 0.00
---------------------------------------
| eval/                   |            |
|    mean_ep_length       | 11         |
|    mean_reward          | 0.961      |
| time/                   |            |
|    total_timesteps      | 80000      |
| train/                  |            |
|    approx_kl            | 0.04313474 |
|    clip_fraction        | 0.107      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.213     |
|    explained_variance   | 0.829      |
|    learning_rate        | 0.0003     |
|    loss                 | 0.00571    |
|    n_updates            | 780        |
|    policy_gradient_loss | -0.0187    |
|    value_loss           | 0.000195   |
---------------------------------------
---------------------------------
| rollout/           |          |
|    ep_len_mean     | 12       |
|    ep_rew_mean     | 0.958    |
| time/              |          |
|    fps             | 712      |
|    iterations      | 79       |
|    time_elapsed    | 113      |
|    total_timesteps | 80896    |
---------------------------------
---------------------------------------
| rollout/                |            |
|    ep_len_mean          | 12.8       |
|    ep_rew_mean          | 0.955      |
| time/                   |            |
|    fps                  | 711        |
|    iterations           | 80         |
|    time_elapsed         | 115        |
|    total_timesteps      | 81920      |
| train/                  |            |
|    approx_kl            | 0.06471266 |
|    clip_fraction        | 0.314      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.294     |
|    explained_variance   | 0.938      |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0486    |
|    n_updates            | 790        |
|    policy_gradient_loss | -0.0427    |
```

```
|    value_loss           | 5.16e-05   |
-----------------------------------------
-----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 11.8       |
|    ep_rew_mean          | 0.958      |
| time/                   |            |
|    fps                  | 711        |
|    iterations           | 81         |
|    time_elapsed         | 116        |
|    total_timesteps      | 82944      |
| train/                  |            |
|    approx_kl            | 0.06405732 |
|    clip_fraction        | 0.211      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.176     |
|    explained_variance   | 0.795      |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0243    |
|    n_updates            | 800        |
|    policy_gradient_loss | -0.0288    |
|    value_loss           | 0.000221   |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 11.6        |
|    ep_rew_mean          | 0.959       |
| time/                   |             |
|    fps                  | 710         |
|    iterations           | 82          |
|    time_elapsed         | 118         |
|    total_timesteps      | 83968       |
| train/                  |             |
|    approx_kl            | 0.061312027 |
|    clip_fraction        | 0.0476      |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.0518     |
|    explained_variance   | 0.881       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0393     |
|    n_updates            | 810         |
|    policy_gradient_loss | -0.023      |
|    value_loss           | 0.000125    |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.2        |
|    ep_rew_mean          | 0.957       |
| time/                   |             |
|    fps                  | 708         |
|    iterations           | 83          |
|    time_elapsed         | 119         |
|    total_timesteps      | 84992       |
| train/                  |             |
|    approx_kl            | 0.039377097 |
|    clip_fraction        | 0.181       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.117      |
|    explained_variance   | 0.814       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0468     |
|    n_updates            | 820         |
|    policy_gradient_loss | 0.022       |
|    value_loss           | 0.000187    |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 11.8        |
|    ep_rew_mean          | 0.958       |
| time/                   |             |
|    fps                  | 708         |
|    iterations           | 84          |
|    time_elapsed         | 121         |
|    total_timesteps      | 86016       |
| train/                  |             |
|    approx_kl            | 0.022505693 |
|    clip_fraction        | 0.109       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.158      |
|    explained_variance   | 0.671       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0147     |
|    n_updates            | 830         |
|    policy_gradient_loss | -0.0127     |
|    value_loss           | 0.000268    |
-----------------------------------------
-----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 15.4       |
|    ep_rew_mean          | 0.946      |
| time/                   |            |
|    fps                  | 707        |
|    iterations           | 85         |
|    time_elapsed         | 123        |
|    total_timesteps      | 87040      |
| train/                  |            |
|    approx_kl            | 0.11171889 |
|    clip_fraction        | 0.142      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.131     |
|    explained_variance   | 0.914      |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0735    |
|    n_updates            | 840        |
|    policy_gradient_loss | -0.0356    |
|    value_loss           | 6.9e-05    |
-----------------------------------------
-----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 14.4       |
|    ep_rew_mean          | 0.95       |
| time/                   |            |
|    fps                  | 706        |
|    iterations           | 86         |
|    time_elapsed         | 124        |
|    total_timesteps      | 88064      |
| train/                  |            |
|    approx_kl            | 0.05634898 |
|    clip_fraction        | 0.322      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.321     |
|    explained_variance   | 0.337      |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0435    |
|    n_updates            | 850        |
|    policy_gradient_loss | -0.03      |
|    value_loss           | 0.00163    |
-----------------------------------------
-----------------------------------------
| rollout/                |           |
|    ep_len_mean          | 11.6      |
|    ep_rew_mean          | 0.959     |
| time/                   |           |
|    fps                  | 706       |
|    iterations           | 87        |
|    time_elapsed         | 126       |
|    total_timesteps      | 89088     |
| train/                  |           |
|    approx_kl            | 0.1144023 |
|    clip_fraction        | 0.191     |
|    clip_range           | 0.2       |
|    entropy_loss         | -0.188    |
|    explained_variance   | 0.632     |
|    learning_rate        | 0.0003    |
|    loss                 | -0.0365   |
|    n_updates            | 860       |
|    policy_gradient_loss | -0.0271   |
|    value_loss           | 0.000624  |
-----------------------------------------
Eval num_timesteps=90000, episode_reward=0.96 +/- 0.00
Episode length: 11.00 +/- 0.00
-----------------------------------------
```

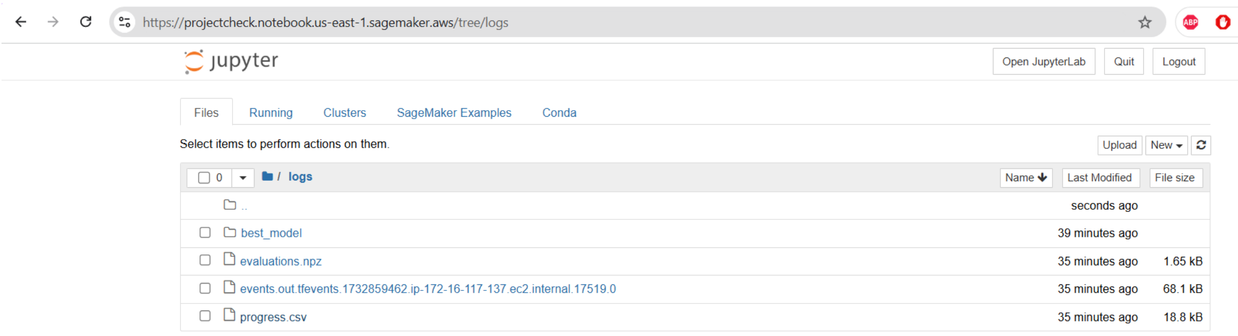
```
| eval/                   |              |
|    mean_ep_length       | 11           |
|    mean_reward          | 0.961        |
| time/                   |              |
|    total_timesteps      | 90000        |
| train/                  |              |
|    approx_kl            | 0.065284915  |
|    clip_fraction        | 0.063        |
|    clip_range           | 0.2          |
|    entropy_loss         | -0.0858      |
|    explained_variance   | 0.89         |
|    learning_rate        | 0.0003       |
|    loss                 | -0.0338      |
|    n_updates            | 870          |
|    policy_gradient_loss | 0.0276       |
|    value_loss           | 0.000171     |
-----------------------------------------
---------------------------------
| rollout/           |          |
|    ep_len_mean     | 12.8     |
|    ep_rew_mean     | 0.955    |
| time/              |          |
|    fps             | 705      |
|    iterations      | 88       |
|    time_elapsed    | 127      |
|    total_timesteps | 90112    |
---------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 14.6        |
|    ep_rew_mean          | 0.949       |
| time/                   |             |
|    fps                  | 704         |
|    iterations           | 89          |
|    time_elapsed         | 129         |
|    total_timesteps      | 91136       |
| train/                  |             |
|    approx_kl            | 0.096123055 |
|    clip_fraction        | 0.196       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.191      |
|    explained_variance   | 0.633       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0416     |
|    n_updates            | 880         |
|    policy_gradient_loss | -0.0249     |
|    value_loss           | 0.000407    |
-----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 11.1       |
|    ep_rew_mean          | 0.961      |
| time/                   |            |
|    fps                  | 704        |
|    iterations           | 90         |
|    time_elapsed         | 130        |
|    total_timesteps      | 92160      |
| train/                  |            |
|    approx_kl            | 0.19162205 |
|    clip_fraction        | 0.145      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.15      |
|    explained_variance   | 0.256      |
|    learning_rate        | 0.0003     |
|    loss                 | 0.0508     |
|    n_updates            | 890        |
|    policy_gradient_loss | -0.0202    |
|    value_loss           | 0.00348    |
----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 12.3       |
|    ep_rew_mean          | 0.957      |
| time/                   |            |
|    fps                  | 703        |
|    iterations           | 91         |
|    time_elapsed         | 132        |
|    total_timesteps      | 93184      |
| train/                  |            |
|    approx_kl            | 0.06836694 |
|    clip_fraction        | 0.347      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.264     |
|    explained_variance   | 0.581      |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0699    |
|    n_updates            | 900        |
|    policy_gradient_loss | 0.066      |
|    value_loss           | 0.000154   |
----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.4        |
|    ep_rew_mean          | 0.957       |
| time/                   |             |
|    fps                  | 703         |
|    iterations           | 92          |
|    time_elapsed         | 133         |
|    total_timesteps      | 94208       |
| train/                  |             |
|    approx_kl            | 0.020714343 |
|    clip_fraction        | 0.161       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.218      |
|    explained_variance   | 0.581       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.01       |
|    n_updates            | 910         |
|    policy_gradient_loss | -0.00672    |
|    value_loss           | 0.000254    |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 12.3        |
|    ep_rew_mean          | 0.957       |
| time/                   |             |
|    fps                  | 703         |
|    iterations           | 93          |
|    time_elapsed         | 135         |
|    total_timesteps      | 95232       |
| train/                  |             |
|    approx_kl            | 0.037652392 |
|    clip_fraction        | 0.184       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.199      |
|    explained_variance   | 0.896       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0423     |
|    n_updates            | 920         |
|    policy_gradient_loss | -0.0175     |
|    value_loss           | 0.000112    |
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 11.2        |
|    ep_rew_mean          | 0.961       |
| time/                   |             |
|    fps                  | 702         |
|    iterations           | 94          |
|    time_elapsed         | 136         |
|    total_timesteps      | 96256       |
| train/                  |             |
|    approx_kl            | 0.080989845 |
|    clip_fraction        | 0.205       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.133      |
|    explained_variance   | 0.918       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.05       |
|    n_updates            | 930         |
|    policy_gradient_loss | -0.0212     |
|    value_loss           | 8.25e-05    |
```

```
-----------------------------------------
-----------------------------------------
| rollout/                |             |
|    ep_len_mean          | 21          |
|    ep_rew_mean          | 0.923       |
| time/                   |             |
|    fps                  | 703         |
|    iterations           | 95          |
|    time_elapsed         | 138         |
|    total_timesteps      | 97280       |
| train/                  |             |
|    approx_kl            | 0.13105541  |
|    clip_fraction        | 0.316       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.254      |
|    explained_variance   | 0.971       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0671     |
|    n_updates            | 940         |
|    policy_gradient_loss | 0.197       |
|    value_loss           | 1.27e-05    |
-----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 27.9       |
|    ep_rew_mean          | 0.897      |
| time/                   |            |
|    fps                  | 703        |
|    iterations           | 96         |
|    time_elapsed         | 139        |
|    total_timesteps      | 98304      |
| train/                  |            |
|    approx_kl            | 0.1051268  |
|    clip_fraction        | 0.194      |
|    clip_range           | 0.2        |
|    entropy_loss         | -0.144     |
|    explained_variance   | -0.119     |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0391    |
|    n_updates            | 950        |
|    policy_gradient_loss | 0.162      |
|    value_loss           | 0.00478    |
----------------------------------------
----------------------------------------
| rollout/                |            |
|    ep_len_mean          | 15.9       |
|    ep_rew_mean          | 0.943      |
| time/                   |            |
|    fps                  | 704        |
|    iterations           | 97         |
|    time_elapsed         | 140        |
|    total_timesteps      | 99328      |
| train/                  |            |
|    approx_kl            | 0.5337625  |
|    clip_fraction        | 0.366      |
|    clip_range           | 0.2        |
|    explained_variance   | -0.0747    |
|    learning_rate        | 0.0003     |
|    learning_rate        | 0.0003     |
|    loss                 | -0.0265    |
|    n_updates            | 960        |
|    policy_gradient_loss | 0.308      |
|    value_loss           | 0.00451    |
----------------------------------------
Eval num_timesteps=100000, episode_reward=0.96 +/- 0.00
Episode length: 11.00 +/- 0.00
-----------------------------------------
| eval/                   |             |
|    mean_ep_length       | 11          |
|    mean_reward          | 0.961       |
| time/                   |             |
|    total_timesteps      | 100000      |
| train/                  |             |
|    approx_kl            | 0.13666381  |
|    clip_fraction        | 0.168       |
|    clip_range           | 0.2         |
|    entropy_loss         | -0.155      |
|    explained_variance   | -0.427      |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0696     |
|    n_updates            | 970         |
|    policy_gradient_loss | 0.0638      |
|    value_loss           | 0.00239     |
-----------------------------------------
---------------------------------
| rollout/           |          |
|    ep_len_mean     | 13.1     |
|    ep_rew_mean     | 0.954    |
| time/              |          |
|    fps             | 705      |
|    iterations      | 98       |
|    time_elapsed    | 142      |
|    total_timesteps | 100352   |
---------------------------------
Training complete.
Model saved as ppo_minigrid_model.
Testing trained agent...
```

APPENDIX D

Model Trained in SageMaker Instance

The images are captured during the training process in AWS SageMaker and logs are stored in the S3 bucket.

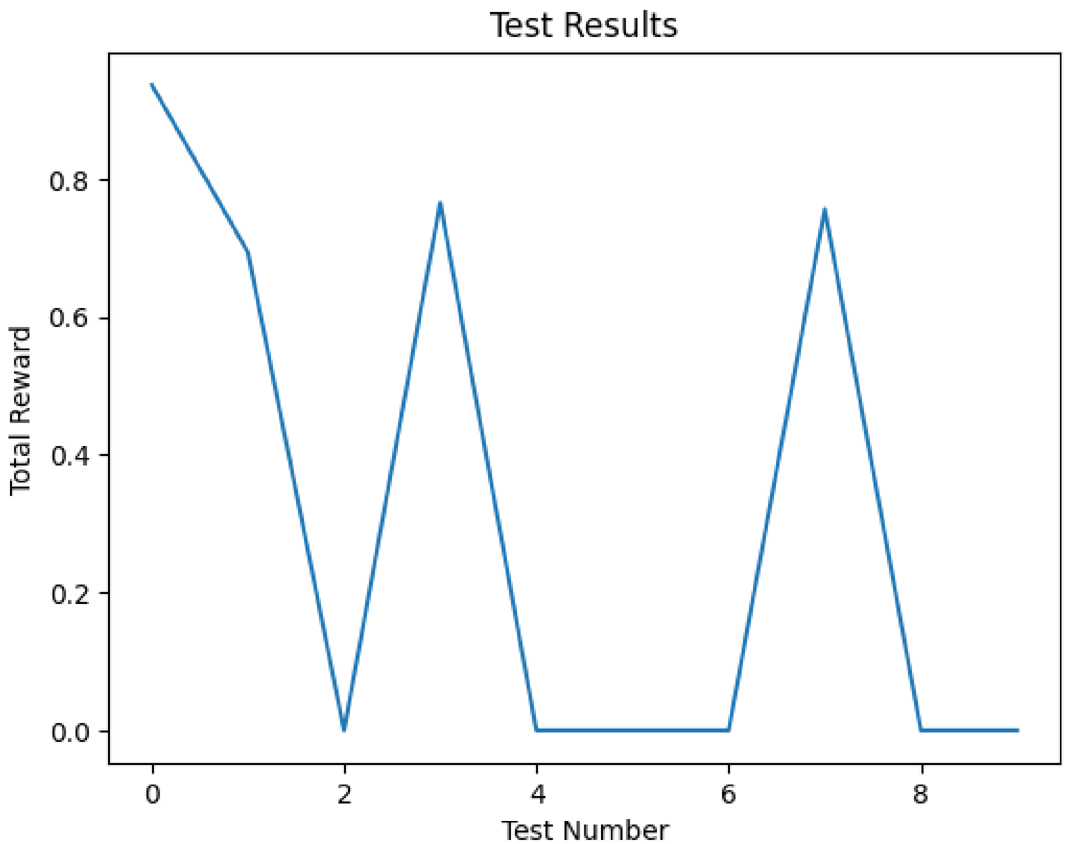


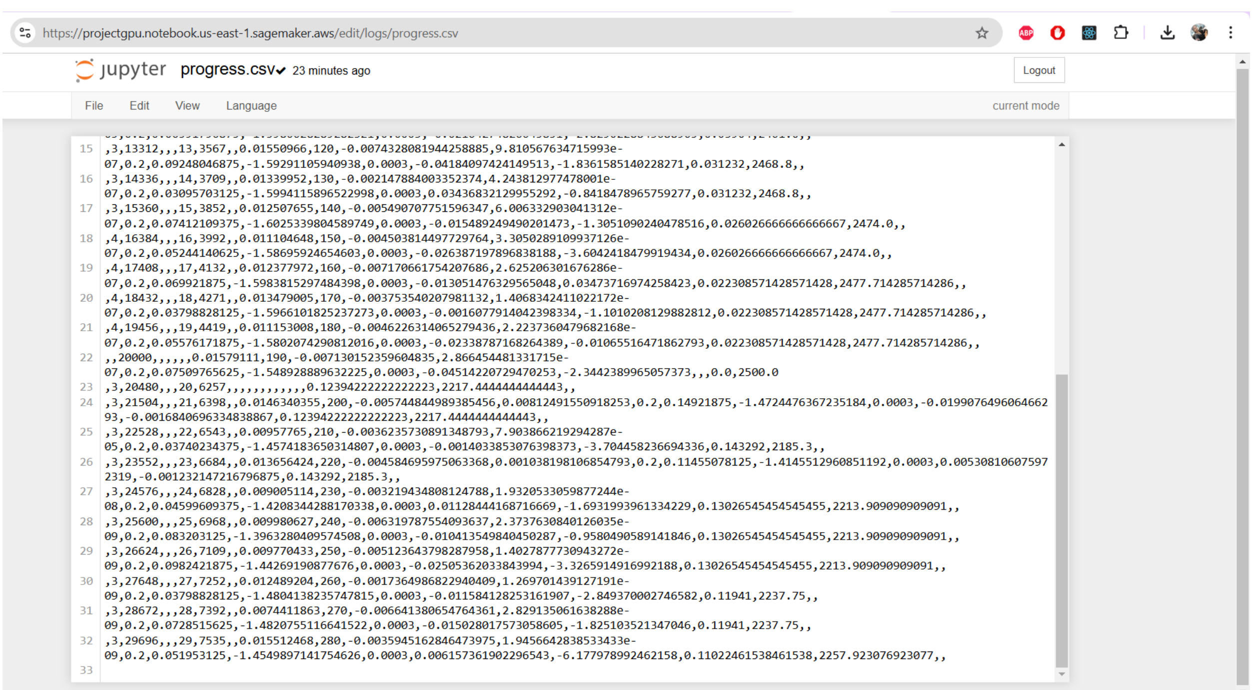

```
In [7]: def test_agent(model, env):
            """Tests a trained RL agent on the given environment."""
            obs, _ = env.reset()
            done = False
            while not done:
                env.render()
                action, _ = model.predict(obs)
                obs, reward, done, info = env.step(action)
                print(f"Reward: {reward}")
```

```
In [8]: env = create_environment()
        model = train_agent(env)
```

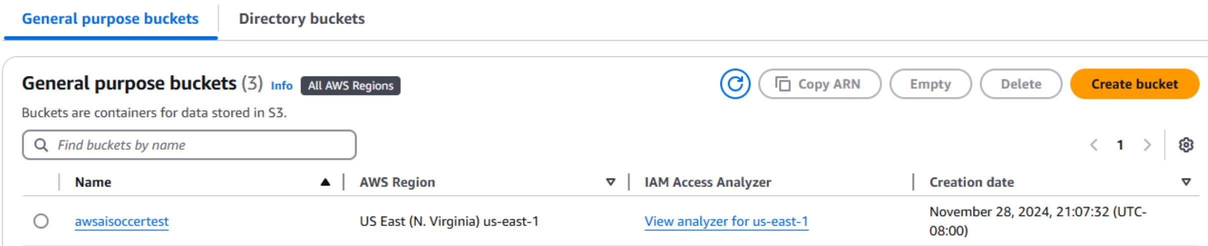
```
|    entropy_loss         | -0.134      |
|    explained_variance   | 0.615       |
|    learning_rate        | 0.0003      |
|    loss                 | -0.0335     |
|    n_updates            | 970         |
|    policy_gradient_loss | -0.0233     |
|    value_loss           | 0.000439    |
-----------------------------------------
---------------------------------
| rollout/           |          |
|    ep_len_mean     | 12.6     |
|    ep_rew_mean     | 0.956    |
| time/              |          |
|    fps             | 364      |
|    iterations      | 98       |
|    time_elapsed    | 275      |
|    total_timesteps | 100352   |
---------------------------------
Model saved to S3: s3://awsaisoccertest/models/ppo_minigrid_2024-11-29_05-51-01.zip
```

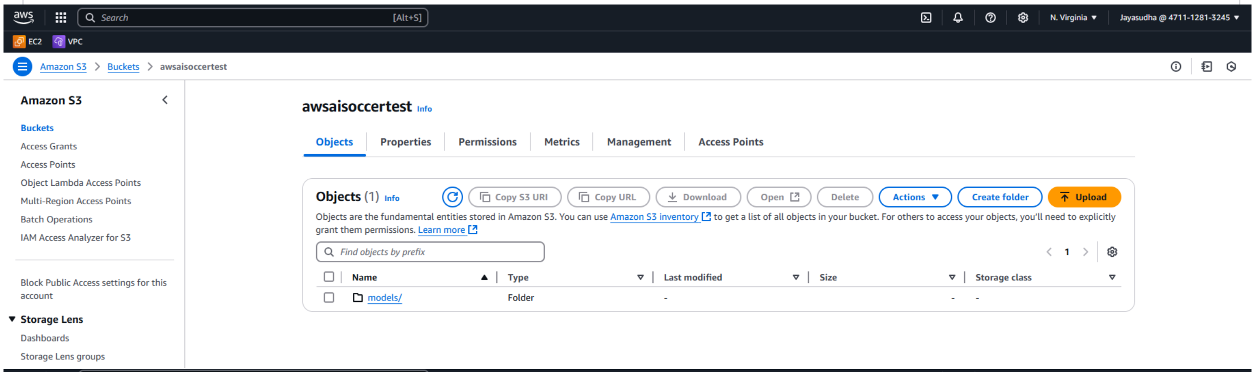
```
Episode finished after 7 steps with total reward: 0.9369999999999999
Episode finished after 34 steps with total reward: 0.694
Episode finished after 100 steps with total reward: 0
Episode finished after 26 steps with total reward: 0.766
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
Episode finished after 27 steps with total reward: 0.757
Episode finished after 100 steps with total reward: 0
Episode finished after 100 steps with total reward: 0
```

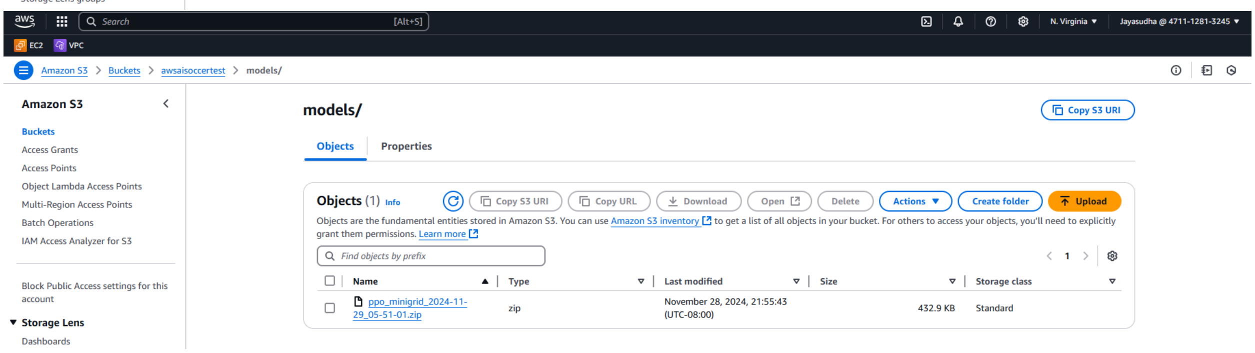





```
,3,13312,,,13,3567,,0.01550966,120,-0.0074328081944258885,9.810567634715993e-07,0.2,0.09248046875,-1.59291105940938,0.0003,-0.04184097424149513,-1.8361585140228271,0.031232,2468.8,,
,3,14336,,,14,3709,,0.01339952,130,-0.002147884003352374,4.243812977478001e-07,0.2,0.03095703125,-1.5994115896522998,0.0003,0.03436832129955292,-0.8418478965759277,0.031232,2468.8,,
,3,15360,,,15,3852,,0.012507655,140,-0.005490707751596347,6.006332903041312e-07,0.2,0.07412109375,-1.6025339804589749,0.0003,-0.015489249490201473,-1.3051090240478516,0.026026666666666667,2474.0,,
,4,16384,,,16,3992,,0.011104648,150,-0.004503814497729764,3.3050289109937126e-07,0.2,0.05244140625,-1.58695924654603,0.0003,-0.026387197896838188,-3.6042418479919434,0.026026666666666667,2474.0,,
,4,17408,,,17,4132,,0.012377972,160,-0.007170661754207686,2.625206301676286e-07,0.2,0.069921875,-1.5983815297484398,0.0003,-0.013051476329565048,0.03473716974258423,0.022308571428571428,2477.714285714286,,
,4,18432,,,18,4271,,0.013479005,170,-0.003753540207981132,1.4068342411022172e-07,0.2,0.03798828125,-1.5966101825237273,0.0003,-0.0016077914042398334,-1.1010208129882812,0.022308571428571428,2477.714285714286,,
,4,19456,,,19,4419,,0.011153008,180,-0.004622631406527943 6,2.2237360479682168e-07,0.2,0.05576171875,-1.5802074290812016,0.0003,-0.02338787168264389,-0.01065516471862793,0.022308571428571428,2477.714285714286,,
,,20000,,,,,,0.01579111,190,-0.007130152359604835,2.866454481331715e-07,0.2,0.07509765625,-1.548928889632225,0.0003,-0.04514220729470253,-2.3442389965057373,,,0.0,2500.0
,3,20480,,,20,6257,,,,,,,,,,,,0.12394222222222223,2217.4444444444443,,
,3,21504,,,21,6398,,0.0146340355,200,-0.005744844989385456,0.00812491550918253,0.2,0.14921875,-1.4724476367235184,0.0003,-0.019907649606466293,-0.0016840696334838867,0.12394222222222223,2217.4444444444443,,
,3,22528,,,22,6543,,0.00957765,210,-0.0036235730891348793,7.903866219294287e-05,0.2,0.03740234375,-1.4574183650314807,0.0003,-0.0014033853076398373,-3.704458236694336,0.143292,2185.3,,
,3,23552,,,23,6684,,0.013656424,220,-0.004584695975063368,0.0010381981068547 93,0.2,0.11455078125,-1.4145512960851192,0.0003,0.005308106075972319,-0.00123214721679 6875,0.143292,2185.3,,
,3,24576,,,24,6828,,0.009005114,230,-0.003219434808124788,1.9320533059877244e-08,0.2,0.04599609375,-1.4208344288170338,0.0003,0.01128444168716669,-1.6931993961334229,0.13026545454545455,2213.909090909091,,
,3,25600,,,25,6968,,0.009980627,240,-0.006319787554093637,2.3737630840126035e-09,0.2,0.083203125,-1.3963280409574508,0.0003,-0.010413549840450287,-0.9580490589141846,0.13026545454545455,2213.909090909091,,
,3,26624,,,26,7109,,0.009770433,250,-0.005123643798287958,1.4027877730943272e-09,0.2,0.0982421875,-1.44269190877676,0.0003,-0.02505362033843994,-3.3265914916992188,0.13026545454545455,2213.909090909091,,
,3,27648,,,27,7252,,0.012489204,260,-0.0017364986822940409,1.269701439127191e-09,0.2,0.03798828125,-1.4804138235747815,0.0003,-0.011584128253161907,-2.849370002746582,0.11941,2237.75,,
,3,28672,,,28,7392,,0.0074411863,270,-0.006641380654764361,2.829135061638288e-09,0.2,0.0728515625,-1.4820755116641522,0.0003,-0.015028017573058605,-1.825103521347046,0.11941,2237.75,,
,3,29696,,,29,7535,,0.015512468,280,-0.0035945162846473975,1.9456642838533433e-09,0.2,0.051953125,-1.4549897141754626,0.0003,0.006157361902296543,-6.177978992462158,0.1102246153846153 8,2257.923076923077,,
```

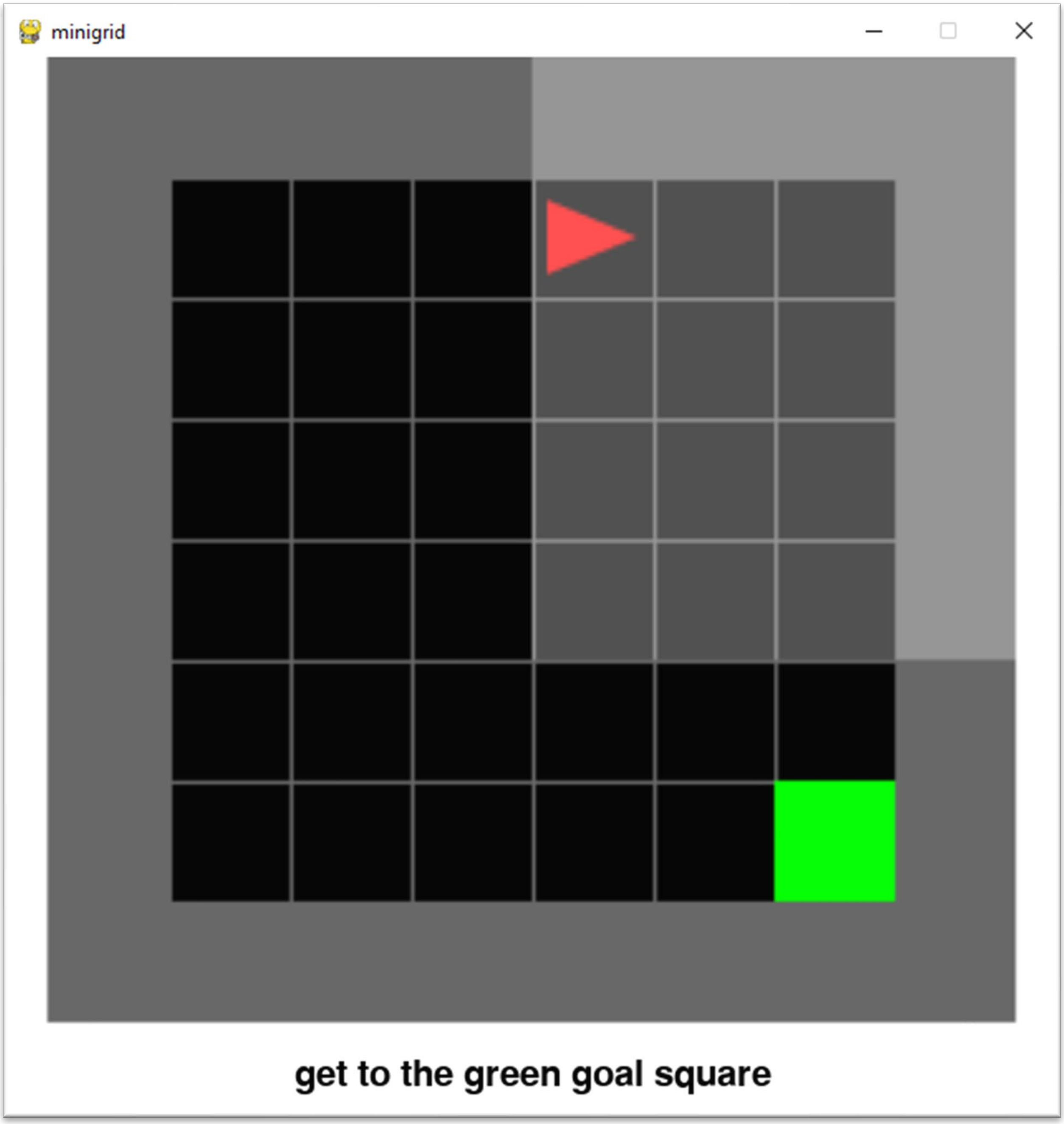

General purpose buckets
Directory buckets
General purpose buckets (3) Info All AWS Regions
Copy ARN
Empty
Delete
Create bucket
Buckets are containers for data stored in S3.
Find buckets by name
Name
AWS Region
IAM Access Analyzer
Creation date
awsaisoccertest
US East (N. Virginia) us-east-1
View analyzer for us-east-1
November 28, 2024, 21:07:32 (UTC-08:00)


Search [Alt+S]
N. Virginia
Jayasudha @ 4711-1281-3245
EC2
VPC
Amazon S3 > Buckets > awsaisoccertest
Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
Storage Lens groups
awsaisoccertest Info
Objects
Properties
Permissions
Metrics
Management
Access Points
Objects (1) Info
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more
Find objects by prefix
Name
Type
Last modified
Size
Storage class
models/
Folder


Search [Alt+S]
N. Virginia
Jayasudha @ 4711-1281-3245
EC2
VPC
Amazon S3 > Buckets > awsaisoccertest > models/
Amazon S3
Buckets
Access Grants
Access Points
Object Lambda Access Points
Multi-Region Access Points
Batch Operations
IAM Access Analyzer for S3
Block Public Access settings for this account
Storage Lens
Dashboards
models/
Copy S3 URI
Objects
Properties
Objects (1) Info
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder
Upload
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more
Find objects by prefix
Name
Type
Last modified
Size
Storage class
ppo_minigrid_2024-11-29_05-51-01.zip
zip
November 28, 2024, 21:55:43 (UTC-08:00)
432.9 KB
Standard

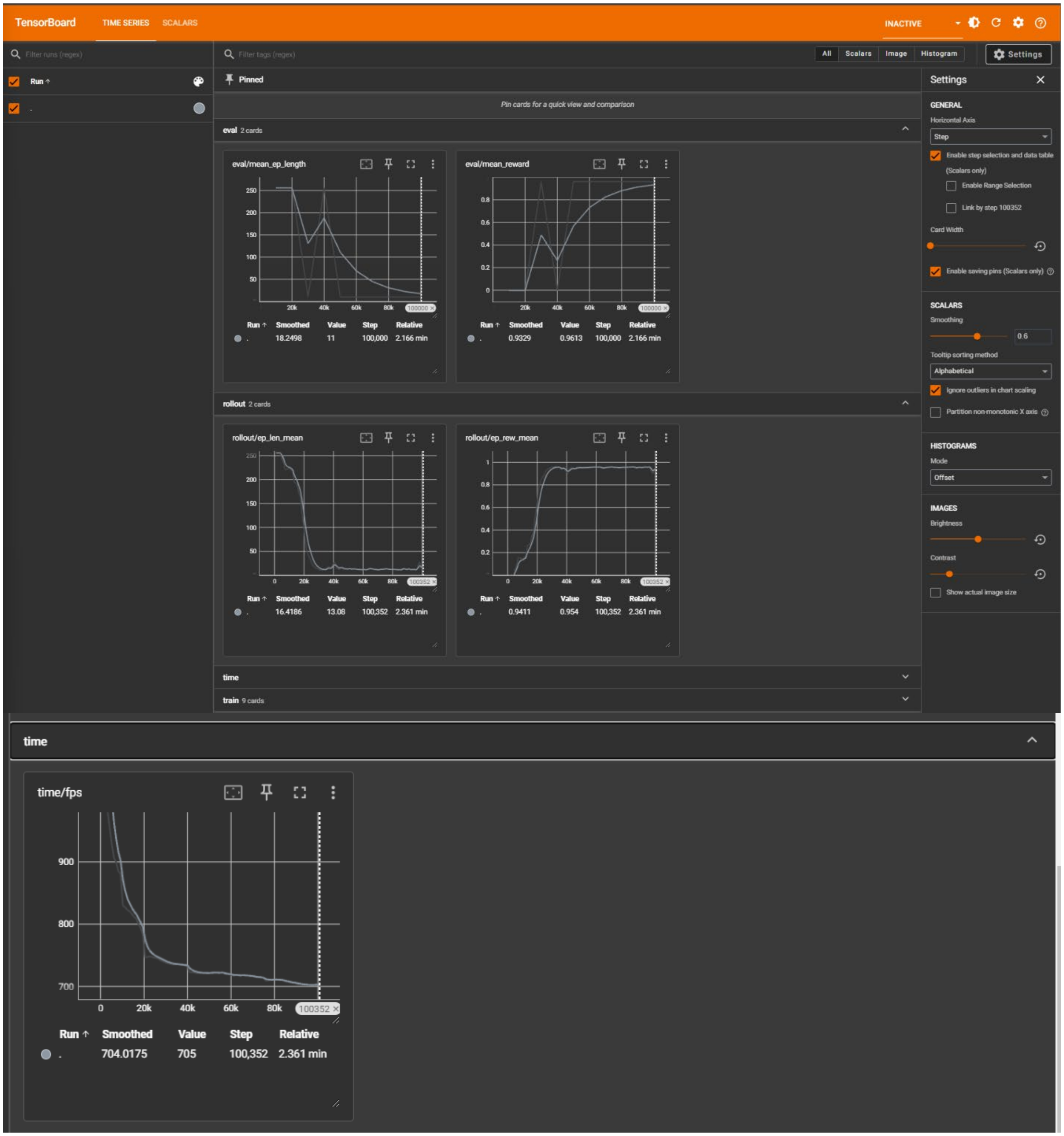
APPENDIX E

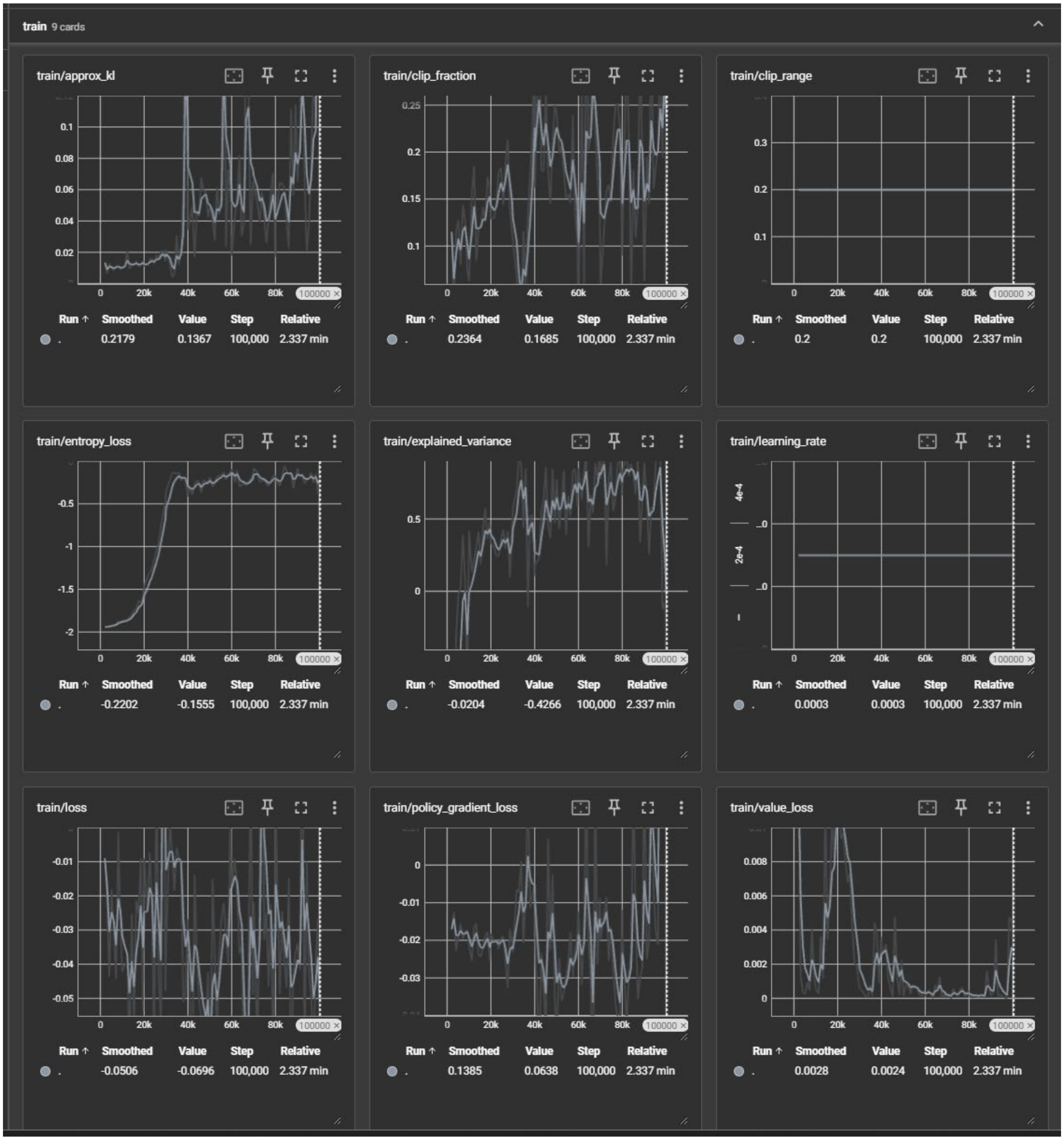
Render Mode = Human
This still image, captured during training in human render mode, illustrates the agent (red triangle) navigating a grid environment toward the green goal square as per the specified task.



APPENDIX F

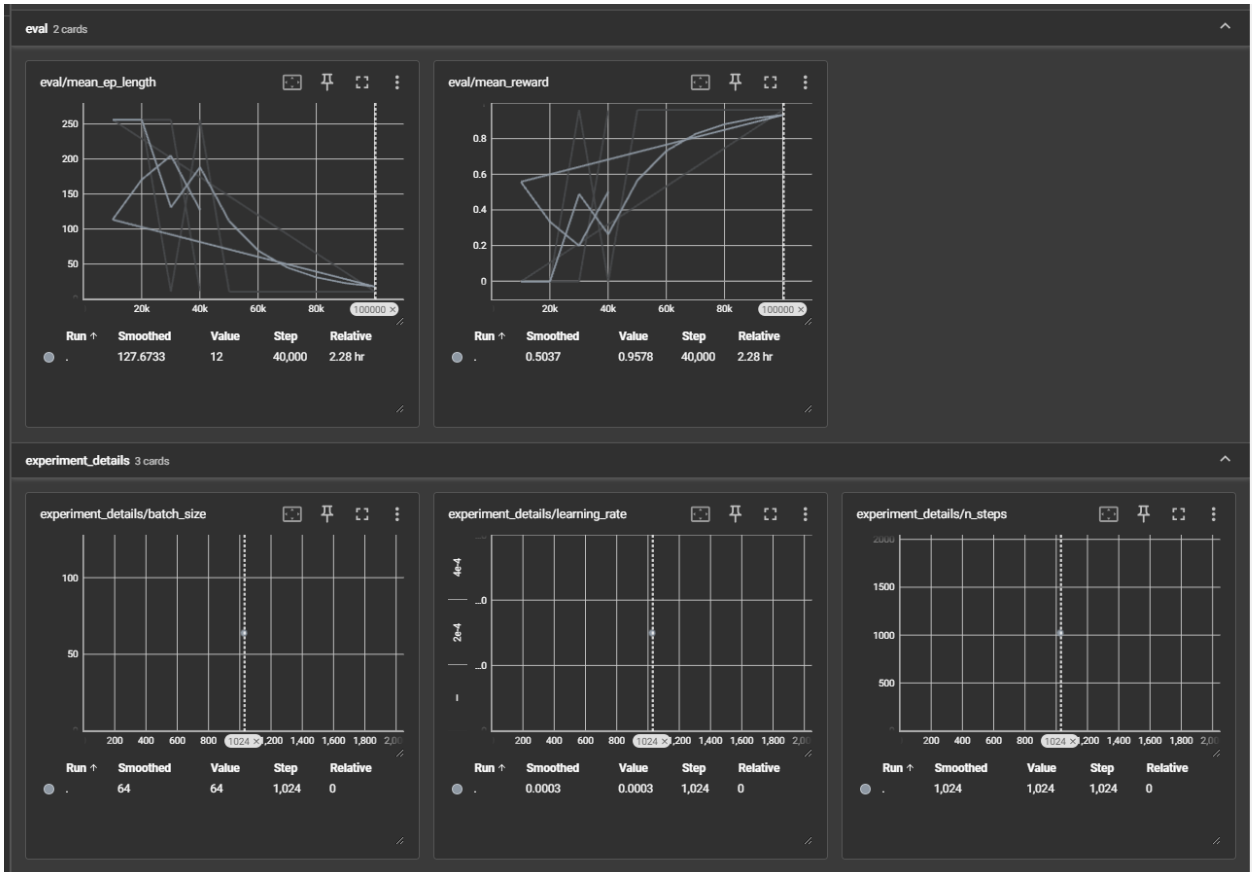
TensorBoard Logs – 1st Visualized Training Metrics
This appendix contains detailed logs from the initial evaluation phase, highlighting the agent's performance metrics, episode outcomes, and termination conditions during validation runs.

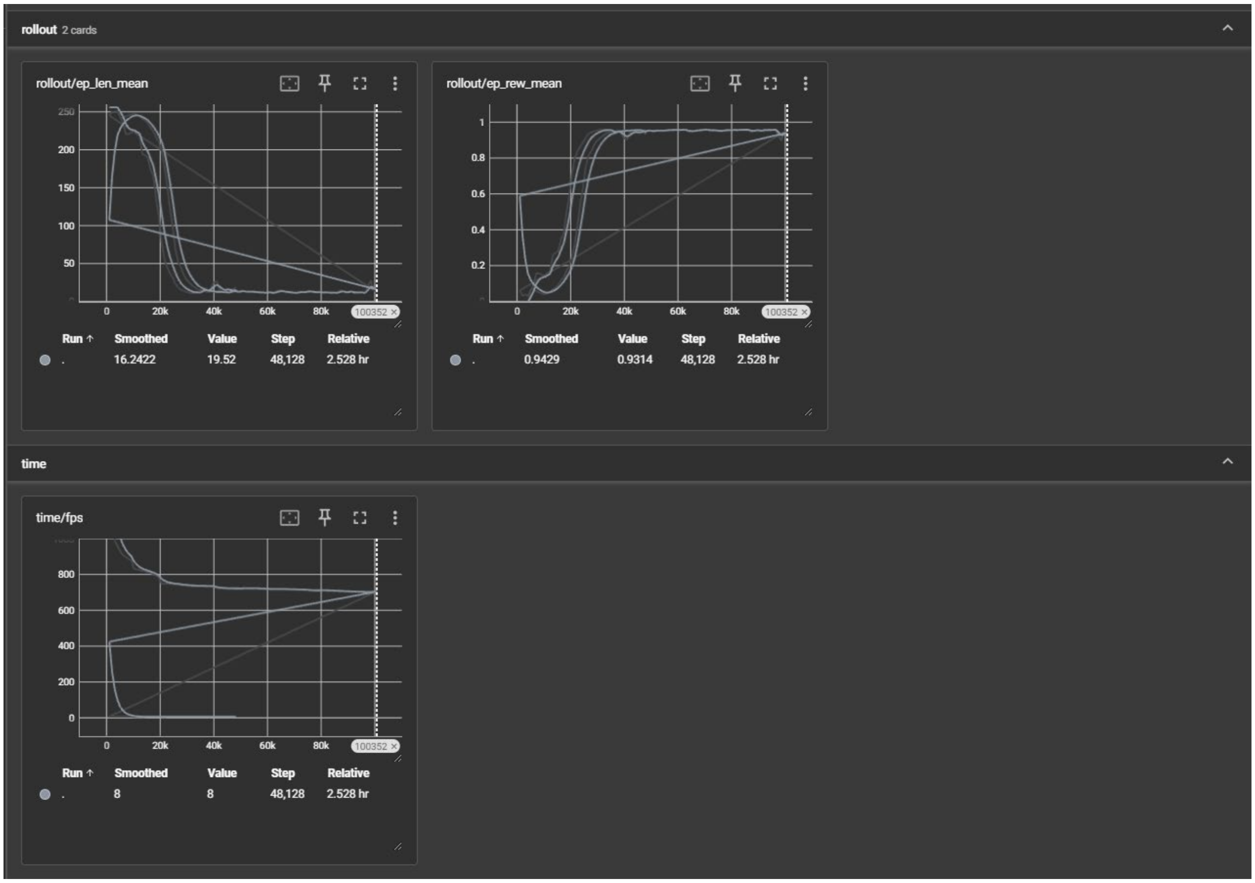


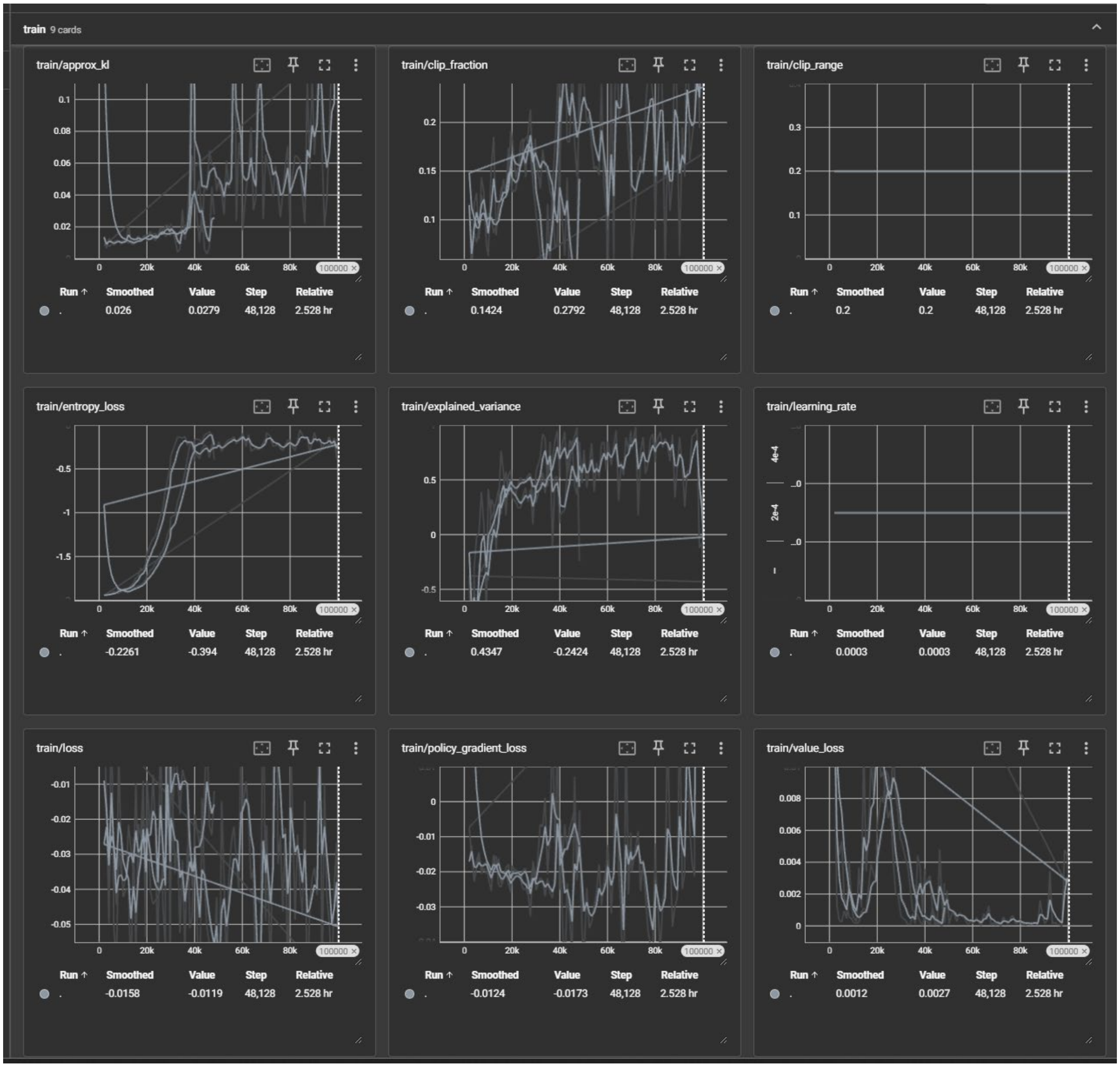

train 9 cards
train/approx_kl
Run Smoothed Value Step Relative
. 0.2179 0.1367 100,000 2.337 min
train/clip_fraction
. 0.2364 0.1685 100,000 2.337 min
train/clip_range
. 0.2 0.2 100,000 2.337 min
train/entropy_loss
. -0.2202 -0.1555 100,000 2.337 min
train/explained_variance
. -0.0204 -0.4266 100,000 2.337 min
train/learning_rate
. 0.0003 0.0003 100,000 2.337 min
train/loss
. -0.0506 -0.0696 100,000 2.337 min
train/policy_gradient_loss
. 0.1385 0.0638 100,000 2.337 min
train/value_loss
. 0.0028 0.0024 100,000 2.337 min

APPENDIX G

TensorBoard Logs – 2nd Visualized Training Metrics + Additional Details
This appendix provides comprehensive logs from the training runs, documenting the agent's progress, action sequences, rewards, and termination conditions to support the analysis of training performance.




rollout 2 cards
rollout/ep_len_mean
Run ↑ Smoothed Value Step Relative
. 16.2422 19.52 48,128 2.528 hr
rollout/ep_rew_mean
Run ↑ Smoothed Value Step Relative
. 0.9429 0.9314 48,128 2.528 hr
time
time/fps
Run ↑ Smoothed Value Step Relative
. 8 8 48,128 2.528 hr


train 9 cards
train/approx_kl
Run ↑ Smoothed Value Step Relative
. 0.026 0.0279 48,128 2.528 hr
train/clip_fraction
Run ↑ Smoothed Value Step Relative
. 0.1424 0.2792 48,128 2.528 hr
train/clip_range
Run ↑ Smoothed Value Step Relative
. 0.2 0.2 48,128 2.528 hr
train/entropy_loss
Run ↑ Smoothed Value Step Relative
. -0.2261 -0.394 48,128 2.528 hr
train/explained_variance
Run ↑ Smoothed Value Step Relative
. 0.4347 -0.2424 48,128 2.528 hr
train/learning_rate
Run ↑ Smoothed Value Step Relative
. 0.0003 0.0003 48,128 2.528 hr
train/loss
Run ↑ Smoothed Value Step Relative
. -0.0158 -0.0119 48,128 2.528 hr
train/policy_gradient_loss
Run ↑ Smoothed Value Step Relative
. -0.0124 -0.0173 48,128 2.528 hr
train/value_loss
Run ↑ Smoothed Value Step Relative
. 0.0012 0.0027 48,128 2.528 hr