AI Enabled 2D Arcade Style Racing Game

Aaditya Trivedi[1], Priyanka Sharma[1], Ajay Bhardwaj[2]
[1]Department of Computer Science & Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur -302017 (India),
[2]Department of Electrical Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur -302017 (India)
Email: **myself.aaditya.trivedi@gmail.com, priyanka08291@gmail.com, ajayb23@gmail.com**
Received 03.09.2024 **received in revised form** 09.10.2024, **accepted** 23.10.2024
DOI: 10.47904/IJSKIT.14.2.2024.20-24

Abstract- This paper explores the integration of the Proximal Policy Optimization (PPO) algorithm in a 2D racing game created with Unity, aiming to demonstrate AI's transformative potential in gaming. By training an autonomous agent within the game's environment, the study addresses challenges in environment design and algorithm optimization. It focuses on how PPO can enhance performance and adaptability in dynamic racing scenarios, offering insights into the synergy between game development and reinforcement learning.

Keywords– Reinforcement learning, Unity, PPO Algorithm, Proximal Policy Optimization.

1. INTRODUCTION

In the dynamic landscape of artificial intelligence (AI) and gaming, the integration of advanced AI algorithms into the development of video games represents a cutting-edge intersection. This research project emerges within this context, aiming to fuse the Proximal Policy Optimization (PPO) algorithm with a 2D arcade-style racing game crafted from scratch using the Unity development platform [1]. Traditional gaming experiences often lack adaptability and dynamic interaction, limiting player engagement. This study addresses this issue by investigating the potential of the PPO algorithm to empower an autonomous agent within a 2D racing game [2]. The problem statement centers on the need for AI-driven enhancements in gaming to create more immersive and challenging environments for players. The significance of this research lies in its contribution to the broader discourse on the synergy between AI and gaming. By leveraging PPO in a custom-designed racing game, the study aims to demonstrate how AI can elevate player experiences, opening avenues for more sophisticated and engaging gameplay. Additionally, the research holds implications for the practical application of AI algorithms in game development, showcasing the potential for transformative innovations [3]-[4].

PPO uses an on-policy approach to train a stochastic policy. In other words, it investigates by sampling activities in accordance with the most recent iteration of its stochastic strategy. [5]. The training process and the initial conditions both affect how random the action selection process is. The policy usually becomes less random throughout training since the update rule pushes it to take advantage of rewards it has previously discovered. This could lead to the policy becoming stuck in local optima. [6]-[7].

The primary research objective is to evaluate the effectiveness of the PPO algorithm in training an AI agent to master the challenges presented in a 2D arcade-style racing game. The hypothesis posits that the integration of PPO will result in an autonomous agent capable of dynamic adaptation and optimal decision-making within the gaming environment [8]. The paper is structured to provide a comprehensive exploration of the research project. It begins with an introduction to the background and motivation behind integrating AI into gaming [9]-[10]. The subsequent sections delve into the methodology employed, key findings, and implications of the study. The research objectives guide the progression of the paper, culminating in a conclusive discussion on the transformative potential of AI in shaping the future of gaming experiences.

2. LITERATURE REVIEW

The integration of AI within the gaming industry represents a significant shift towards creating more dynamic and interactive gaming experiences [1]. Studies in the past have explored various AI techniques in video games, primarily focusing on enhancing non-player character (NPC) behavior and game environment dynamics [12]. A significant contribution to this area has been the adoption of machine learning algorithms to improve real-time decision-making and adaptability in games. Introduced by Schulman et al. (2017), PPO has gained prominence for its stability and effectiveness in training policies, particularly in the context of reinforcement learning. Its application in gaming is relatively recent but rapidly gaining interest for its ability to optimize complex decision-making processes [13]-[14]. While AI has the potential to revolutionize gaming experiences, challenges remain, particularly in terms of algorithmic efficiency and the design of adaptive game environments. Recent studies have focused on overcoming these obstacles by refining AI algorithms and enhancing their ability to learn from complex environments [15]. The literature suggests a growing interest in exploring hybrid AI models that combine multiple learning approaches to create more nuanced and engaging gaming experiences [16]. The potential for AI to adapt to player behaviors and alter game dynamics in real-time presents exciting opportunities for future research. In summary, the literature reveals a strong foundation for the use of AI in gaming, highlighted by the adoption of advanced algorithms like PPO. The ongoing developments [17]-[19] in this field suggest a promising future for the integration of more sophisticated AI technologies in game design, offering a pathway towards more immersive and responsive gaming environments [20]-[21].

3. METHODOLOGY

3.1 Environment Design
Since an agent cannot be trained on a game from which it cannot collect observations, we will need to create the game from scratch. For this reason, we will be using Unity 2022.3.5f1. The sprites were taken from a sprites archive. The scripts for the game manager and all the working of the game and game objects are written in C#.

3.2 Terrain Generation
Since the game that we are taking inspiration from is an endless runner, that implies that we cannot estimate how long to generate the terrain, we will need to generate it as we go. So, to procedurally generate the terrain, we will be using "Perlin Noise" which is widely accepted noise to generate terrain.

3.3 Death Handlers
As there is no objective for the game other than travelling as far as it can, we need to create a game over sequence when the player dies. This can be achieved if the player's fuel empties, or the head of the driver collides with the ground.

3.4 Scoring
The scores which would be the factor that would let us know if the model is effective or not is their score. Score is the amount of distance it has traveled from its starting point. For easier understanding, the scores will be in "meters" and represented by "m" in the game's UI.

3.5 Agent Configuration File
To Train the agent, we would need to define a configuration file and define some "Proximal Policy Optimization Algorithm" specific hyperparameters for the model. Some of these hyperparameters are:

- **Beta:** 5e-3, strength of the entropy regularization, which causes the policy to be "more random". This ensures that agents properly explore the whole action space during the training providing better learning.
- **Epsilon:** 0.2, affects the rate at which the policy can evolve during training.
- **Lambda:** 0.95, When determining the Generalized Advantage Estimate (GAE), the regularization parameter is utilized. This might be seen as the degree to which, in determining an update value estimate, the agent depends on its existing value estimate.
- **Num_epoch:** 3, The number of times to do gradient descent optimization through the experience buffer. The permissible size of the num_epoch is directly proportionate to the batch size.
- **Batch Size:** 32, number of experiences in each iteration of the gradient descent,
- **Buffer Size:** 4096, Number of experiences to collect before updating the policy model. It offers details on the number of experiences that ought to be gathered before the model is updated or learned from.

There is absolutely no constraint for PPO-Clip. This makes use of customized clipping in the objective function to eliminate motivation for the new policy to diverge much from the previous one. The equation that was used to update the policy by maximizing the PPO-Clip objective is:

$$\theta_{k+1} = \arg\max_{\theta} \frac{1}{|\mathcal{D}_k| T} \sum_{\tau \in \mathcal{D}_k} \sum_{t=0}^{T} \min\left(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)} A^{\pi_{\theta_k}}(s_t, a_t),\ \ g(\epsilon, A^{\pi_{\theta_k}}(s_t, a_t)) \right)$$

where,
- θ = Policy parameters
- $\mathcal{D}_k$ = Set of trajectories obtained by running policy $\pi_k = \pi(\theta_k)$ in the environment.
- $\hat{R}_t$ = Rewards-to-go
- $\hat{A}_t$ = Advantage estimates based on the current value function

3.6 Agent Interaction
Now that the environment where the agent will work is completed, we will focus on the agent now. We will be creating an "Agent" which would be a script called "AgentAI.cs". We will also be using Unity's ml-toolkit to be able to interact with the episode that the agent is on and reward or penalize it for its action.

3.7 Observations
To observe and sense the information form the environment, we will override the default function of ML Agents Toolkit called "Collect Observation" and we would be asking the "Actions buffer" for discrete values. As this is a 2D game, the only movement that the player is allowed is go forward or backward, which can be further encoded to 0 or 1. Therefore we know with certainty that the player's action would always be either of these values and not in between. Hence the use of discrete values

3.8 Action Performed
We have collected the observations for the agent, now it needs to be able to act on these actions. For this, we will override another function of the ML Agents Toolkit named "On Action Received", it

takes the Action buffer from the “Collect Observation” function and decodes them into 2 values (as we will only have 2 values from the Actions buffer). We will then map them to move forward or backward in the game with our movement script. This will take care of the movement of the agent but now let us work on the reward. The function that we overrode, as the name suggests, will be called every time the “Collect Observation” function would send actions. So, if we need to calculate something continuously, such as the reward for the agent, we will implement in this function.

For starters, we will take maximum distance travelled as 0, and every time the agent travels farther than the maximum distance travelled, it would receive a reward of 1. And for every movement that it goes towards the maximum distance travelled, it would receive a reward of 0.1 for every meter. The same is true for every meter that it would travel farther away from the maximum distance travelled.

4. RESULTS

The implementation of the Proximal Policy Optimization (PPO) algorithm within our custom 2D racing game has demonstrated notable success. Within just a few hours of training, the autonomous agent was capable of achieving distances exceeding 200 meters, a milestone that underscores the efficiency of PPO in rapid learning and adaptation. This performance surpasses that of comparable algorithms, such as the Soft Actor Critic (SAC), not only in the speed of learning but also in the overall resource economy. The PPO algorithm required fewer resources, less time, and fewer training steps compared to SAC, highlighting its suitability for complex game environments where quick adaptation and decision-making are crucial.

The faster learning rate of PPO has practical implications for real-world gaming applications, such as in the online multiplayer game "Hill Climb Racing 2". Here, PPO could be used to develop challenging AI-driven opponents. For instance, before a new custom map is released to the public, a model running PPO could be used to test and refine AI opponents, ensuring that players face highly competent and adaptive adversaries. This approach not only enhances player engagement but also ensures continuous evolution of game difficulty, aligning with players' increasing skill levels.

Moreover, the application of PPO has opened up new avenues for exploring more dynamic interaction models within game settings [22]. For example, the adaptability of the AI can be harnessed to modify game scenarios in real-time based on player behavior, creating a more personalized and engaging gaming experience [23]-[24]. This capacity for dynamic adaptation suggests that PPO could play a pivotal role in the future development of AI in gaming, where the focus is on creating responsive and evolving game environments that cater to the preferences and skills of individual players [25].

Below are some of the statistical data stored and visualized on Tensor Board:

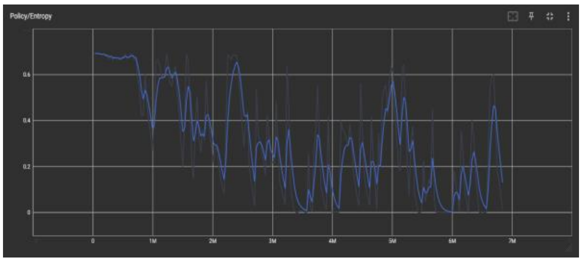

Figure 1: Policy/Entropy graph for the run

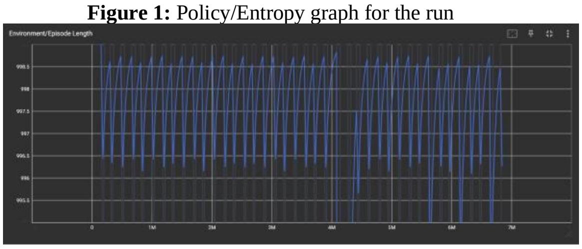

Figure 2: Environment/Episode Length for the run

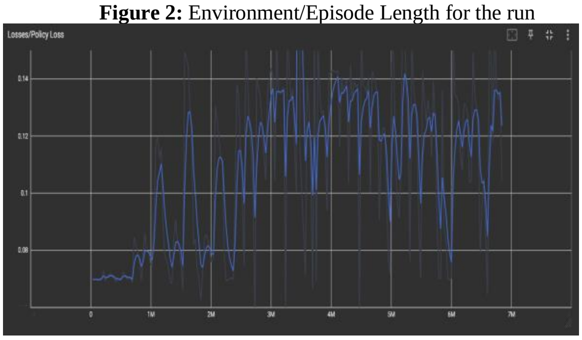

Figure 3: Losses/Policy Loss

Along with that, the running data is shown in Table 1 for the agent to travel a statically placed terrain of length 250m.

Table 1: Running Data for the Agent

Number of Steps	Distance Reached	Time Required
2,100,000	53m	1.319 hours
4.460,000	134m	2.349 hours
5,780,000	250m	3.152 hours

5. CONCLUSION

This study has successfully demonstrated the significant potential of the Proximal Policy Optimization (PPO) algorithm in enhancing the dynamics of game AI, particularly within a 2D arcade-style racing game developed using Unity. By achieving notable milestones in autonomous agent performance, with the agent quickly mastering a complex game environment and achieving distances exceeding 200 meters within a few hours, our research confirms PPO's efficacy over traditional reinforcement learning methods like the Soft Actor Critic (SAC) algorithm. The reduced resource requirement, faster learning rate, and fewer training steps required by PPO not only underline its efficiency but also its practical applicability in game development.

The implications of our findings extend beyond mere technical performance, suggesting a transformative potential for AI in gaming. By integrating PPO, game developers can create more engaging and challenging environments, enhancing player interaction through AI competitors that adapt dynamically to user-generated content. This adaptation ensures a consistently stimulating and evolving gameplay experience that could redefine player engagement standards. Particularly in multiplayer online games, the ability to refine and challenge player skills through sophisticated AI can lead to more immersive and lasting game experiences.

Looking forward, the application of PPO in game development holds promising avenues for further research and development. Future studies could explore the integration of PPO with other AI techniques to create hybrid models that further enhance learning efficiency and gameplay complexity. Additionally, examining the scalability of PPO in more complex 3D environments or its integration into VR gaming could unlock new dimensions of interactive entertainment. As AI continues to advance, its integration into gaming promises not only to enhance the realism and responsiveness of game environments but also to revolutionize the way players interact with and shape game worlds.

6. REFERENCES

[1] Lample, G., & Chaplot, D. S. (2017). Playing FPS Games with Deep Reinforcement Learning. Proceedings of the AAAI Conference on Artificial Intelligence, 31(1). https://doi.org/10.1609/aaai.v31i1.10827 Rukma Rekha Prasad Babu, International Journal on Computer Science and Engineering (2012), Volume 4, Pages 2-6.

[2] Model-based reinforcement learning for Atari, https://arxiv.org/abs/1903.00374v5

[3] Z. Xiao, N. Xie, G. Yang and Z. Du, "Fast-PPO: Proximal Policy Optimization with Optimal Baseline Method," 2020 IEEE International Conference on Progress in Informatics and Computing (PIC), Shanghai, China, 2020, pp. 22-29, doi: 10.1109/PIC50277.2020.9350833.

[4] Continuous-action Reinforcemtn Learning for Playing Racing Games: Comparing SPF to PPO, Mario S Holubar, Marco A Wiering, https://arxiv.org/abs/2001.05270v1

[5] Kim, B.; Kim, J. Optimizing Stage Construction and Level Balancing of Match-3 Puzzle Game with PPO Algorithm Machine Learning. Electronics 2023, 12, 4098. https://doi.org/10.3390/electronics12194098.

[6] D. Kapadiya, C. Shekhawat, and P. Sharma. 2023. A Study on Large Scale Applications of Big Data in Modern Era. In International Conference on Information Management & Machine Intelligence (ICIMMI2023), November 23–25, 2023, Jaipur, India. ACM, New York, NY, USA, https://doi.org/10.1145/3647444.3647880.

[7] P. Sharma and M. K. Jain, 2023. "Stock Market Trends Analysis using Extreme Gradient Boosting (XGBoost)," 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2023, pp. 317-322, doi: 10.1109/ICCCIS60361.2023.10425722.

[8] Merlin Mancy, A., Kumar, A. V., Latip, R., Jagadamba, G., Chakrabarti, P., Sharma, P., Musirin, I. B., Sharma, M., & Kanchan, B. G. (2024). Smart Healthcare System, Digital Health and Telemedicine, Management and Emergencies: Patient Emergency Application (PES) E-Governance Applications. In R. Kumar, A. Abdul Hamid, N. Binti Ya'akub, H. Sharan, & S. Kumar (Eds.), Sustainable Development in AI, Blockchain, and E-Governance Applications (pp. 124-151). IGI Global. https://doi.org/10.4018/979-8-3693-1722-8.ch008.

[9] P. Sharma, N. Bhatnagar, "Passenger Authentication and Ticket Verification at Airport Using QR Code Scanner", SKIT Research Journal, Vol. 13, Issue 2, pp. 10-13, 2023, doi: 10.47904/IJSKIT.13.2.2023.10-13.

[10] P. Rawat, A. Bhardwaj, N. Lamba, P. Sharma, Praveen Kumawat, Prateek Sharma, "Arduino Based IoT Mini Weather Station", SKIT Research Journal, Vol. 13, Issue 2, pp. 34-41, 2023, doi:10.47904/IJSKIT.13.2.2023.34-41.

[11] P. Sharma, D. Kapadiya, A. Bhardwaj, "Efficient Note Sharing Model for Collaborative Learning", SKIT Research Journal, Vol. 13, Issue 2, pp. 42-46, 2023, doi:10.47904/IJSKIT.13.2.2023.42-46.

[12] P. Sharma and P. Dadheech, "Modern-age Agriculture with Artificial Intelligence: A review emphasizing Crop Yield Prediction", EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy, Vol. 10, pp. 2570-2582, 2023, doi: https://doi.org/10.5109/7160906.

[13] P. Sharma, C. Sharma and P. Mathur, "Machine Learning-based Stock Market Forecasting using Recurrent Neural Network," 2023 9th International Conference on Smart Computing and Communications (ICSCC), Kochi, Kerala, India, 2023, pp. 600-605, 2023, doi: 10.1109/ICSCC59169.2023.10335083.

[14] P. Sharma, P. Dadheech, N. Aneja and S. Aneja, "Predicting Agriculture Yields Based on Machine Learning Using Regression and Deep Learning," in IEEE Access, vol. 11, pp. 111255-111264, 2023, doi: 10.1109/ACCESS.2023.3321861.

[15] Sethi, S.S., Sharma, P. New Developments in the Implementation of IoT in Agriculture. SN COMPUT. SCI. 4, 503 (2023). https://doi.org/10.1007/s42979-023-01896-w.

[16] A. Tanwar, P. Sharma, A. Pandey, S. Kumar. 2023. Intrusion Detection System Based Ameliorated Technique of Pattern Matching. In Proceedings of the 4th International Conference on Information Management & Machine Intelligence (ICIMMI '22). Association for Computing Machinery, New York, NY, USA, Article 110, 1–4. https://doi.org/10.1145/3590837.3590947.

[17] Sharma, P., & Rathi, Y. (2016, June 5). Efficient Density-Based Clustering Using Automatic Parameter Detection. Efficient Density-Based Clustering Using Automatic Parameter Detection | SpringerLink. https://doi.org/10.1007/ 978-981-10-0767-5_46.

[18] Sharma, P. (2023) Utilizing Explainable Artificial Intelligence to Address Deep Learning in Biomedical Domain, Medical Data Analysis and Processing using Explainable Artificial Intelligence, Taylor & Francis, pp. 19–38. https://doi.org/10.1201/9781003257721-2.

[19] Prasad G, A., Kumar, A. V., Sharma, P., Irawati, I. D., D. V., C., Musirin, I. B., Abdullah, H. M., & Rao L, M. (2023). Artificial Intelligence in Computer Science: An Overview of Current Trends and Future Directions. In S. Rajest, B. Singh, A. Obaid, R. Regin, & K. Chinnusamy (Eds.), Advances in Artificial and Human Intelligence in the Modern Era (pp. 43-60). IGI Global. https://doi.org/10.4018/979-8-3693-1301-5.ch002.

[20] Maharajan, K., Kumar, A. V., El Emary, I. M., Sharma, P., Latip, R., Mishra, N., Dutta, A., Manjunatha Rao, L., & Sharma, M. (2023). Blockchain Methods and Data-Driven Decision Making With Autonomous Transportation. In R. Kumar, A. Abdul Hamid, & N. Binti Ya'akub (Eds.), Effective AI, Blockchain, and E-Governance Applications for Knowledge Discovery and Management (pp. 176-194). IGI Global. https://doi.org/10.4018/978-1-6684-9151-5.ch012.

[21] V., M. V., Kumar, A. S., Sharma, P., Kaur, S., Saleh, O. S., Chennamma, H., & Chaturvedi, A. (2023). AI-Equipped IoT Applications in High-Tech Agriculture Using Machine Learning. In A. Khang (Ed.), Handbook of Research on AI-Equipped IoT Applications in High-Tech Agriculture (pp.

38-64). IGI Global. https://doi.org/10.4018/978-1-6684-9231-4.ch003.

[22] Sharma, P., Dadheech, P., & Senthil Kumar Senthil, A. V. (2023). AI-Enabled Crop Recommendation System Based on Soil and Weather Patterns. In R. Gupta, A. Jain, J. Wang, S. Bharti, & S. Patel (Eds.), Artificial Intelligence Tools and Technologies for Smart Farming and Agriculture Practices (pp. 184-199). IGI Global. https://doi.org/10.4018/978-1-6684-8516-3.ch010.

[23] Agarwal, Ayushi, et al. "Use Cases of ChatGPT and Other AI Tools With Security Concerns." In Applications, Challenges, and the Future of ChatGPT, edited by Priyanka Sharma, Monika Jyotiyana, and A.V. Senthil Kumar, 216-225. Hershey, PA: IGI Global, 2024. https://doi.org/10.4018/979-8-3693-6824-4.ch012.

[24] Patil, Pradnya, Kaustubh Kulkarni, and Priyanka Sharma. "Algorithmic Issues, Challenges, and Theoretical Concerns of ChatGPT." In Applications, Challenges, and the Future of ChatGPT, edited by Priyanka Sharma, Monika Jyotiyana, and A.V. Senthil Kumar, 56-74. Hershey, PA: IGI Global, 2024. https://doi.org/10.4018/979-8-3693-6824-4.ch003.

[25] Sharma, Priyanka, Monika Jyotiyana, and A.V. Senthil Kumar, eds. Applications, Challenges, and the Future of ChatGPT. Hershey, PA: IGI Global, 2024. https://doi.org/10.4018/979-8-3693-6824-4.