Isabela State University Linker:
Journal of Engineering, Computing, and Technology
Volume 2, Issue 1
ISSN (Print): 3082-3676 ISSN (Online): 3082-3684
DOI: https://doi.org/10.65141/ject.v2i1.n4

HadouQen: Adaptive AI Agent Using Reinforcement Learning in *Street Fighter II: Special Champion Edition*

Isaiah Phil Pangilinan[1], **Neo Alaric B. Villanueva**[2], **Irish Paulo R. Tipay**[3], **Audrey Lyle D. Diego**[4]
College of Informatics and Computing Studies, New Era University, Quezon City, 1107, Philippines[1,2,3,4]
isaiahphil.pangilinan@neu.edu.ph; neoalaric.villanueva@neu.edu.ph; iprtipay@neu.edu.ph; alddiego@neu.edu.ph

RESEARCH ARTICLE INFORMATION

Received: April 13, 2025
Reviewed: May 07, 2025
Accepted: June 17, 2025
Published: June 30, 2025

ABSTRACT

This study presents the development of an AI agent trained using Proximal Policy Optimization (PPO) to compete in Street Fighter II: Special Champion Edition. The agent learned optimal combat strategies through reinforcement learning, processing visual input from frame-stacked grayscale observations (84 × 84 pixels) obtained through the OpenAI Gym Retro environment. Using a convolutional neural network architecture with carefully tuned hyperparameters, the model was trained across 16 parallel environments over 100 million timesteps. The agent was tested against M. Bison, the game's final boss and most challenging opponent, across 1,000 consecutive matches to evaluate performance. Results showed exceptional performance with a 96.7%-win rate and an average reward of 0.912. Training metrics revealed a healthy learning progression, showing steady improvement in average reward per episode, decreased episode length indicating more efficient victories, and stable policy convergence. The findings also demonstrate the effectiveness of PPO-based reinforcement learning in mastering complex fighting game environments and provide a foundation for future research in competitive game-playing agents capable of human-level performance in fast-paced interactive scenarios.

Keywords: *Proximal Policy Optimization, reinforcement learning, fighting games, Street Fighter II, adaptive AI*

Introduction

The gaming industry has evolved into a dynamic and influential sector, both culturally and economically, with diverse genres and platforms attracting a broad spectrum of players. Among these, fighting games like Street Fighter II: Special Champion Edition remain iconic for their fast-paced, competitive, and skill-based mechanics. As one of Capcom's flagship franchises, Street Fighter has played a pivotal role in popularizing the fighting game genre and setting industry standards (Osborn et al., 2023).

Parallel to the growth of gaming, artificial intelligence (AI) has emerged as a transformative force. AI refers to machines programmed to perform tasks typically requiring human cognition—such as visual recognition, decision-making, and strategic thinking (Wang et al., 2019). In gaming, AI enhances player engagement, simulates human-like opponents, and improves training scenarios. For example, OpenAI Five's success in DOTA 2 demonstrated the potential of reinforcement learning (RL)-based agents to achieve superhuman performance in complex multiplayer settings (Berner et al., 2019). The evolution of AI in video games has reached unprecedented levels with the integration of deep reinforcement learning (DRL), enabling agents to learn and adapt dynamically in complex environments. DRL has pushed the boundaries of AI gameplay by allowing bots to develop strategies, respond to environmental feedback, and adjust in real-time (Dong et al., 2021; Goldwaser & Thielscher, 2020).

Fighting games, however, pose unique and formidable challenges for AI compared to turn-based strategy games like chess or Go. These games demand frame-perfect decision-making at millisecond speeds, continuous adaptation to fast-evolving states, and management of partial observability under adversarial pressure (Hu et al., 2023; Yin et al., 2023). Unlike strategic games, where agents can deliberate over moves, fighting games feature sparse rewards, high-dimensional visual input, and vast, complex action spaces, making them one of the most demanding environments for AI research (Halina & Guzdial, 2022; Hazra & Anjaria, 2022). Agents must process visual cues, recognize attack patterns, and execute counter-strategies within tight windows of 1/60th of a second. The temporal complexity and multi-layered adversarial dynamics—such as predicting opponent actions, managing spacing, health, and resource allocation—require advanced DRL techniques that go beyond basic pattern recognition (Gallotta et al., 2024; Janiesch et al., 2021).

In addition, recent work underscores the value of fighting games as a fertile testbed for advancing real-time, adaptive, and generalizable AI systems. AI must respond reflexively and plan strategically, adjusting to different opponents, characters, and styles. Research in this domain helps expand understanding in areas like autonomous systems, robotics, and human-AI interaction (Ashktorab et al., 2020; Taherdoost, 2023).

This study focuses on developing an AI bot capable of defeating the final opponent in Street Fighter II: Special Champion Edition using Proximal Policy Optimization (PPO), a state-of-the-art reinforcement learning algorithm known for training stability and efficient handling of both discrete and continuous action spaces (Li, 2023). PPO was selected for its robustness in on-policy training and its proven effectiveness in complex environments with sparse rewards (Andrychowicz et al., 2021; Clifton & Laber, 2020).

Inspired by successful large-scale projects like OpenAI Five and applications of DRL in FPS and navigation tasks (Alonso et al., 2020; Almeida et al., 2024), the AI agent in this study learned directly from raw pixel data via the Gym Retro emulator environment. It uses vision-based perception and a reward-driven trial-and-error

mechanism to develop expert-level behavior. In contrast to supervised learning or behavior cloning, which rely heavily on human-labeled data, PPO enables agents to learn optimal strategies independently. The integration of computer vision allows the agent to perceive and interpret game states, such as how convolutional neural networks were used to train Atari-playing agents (Joo & Kim, 2019). The AI's performance was evaluated based on win rate, strategic variability, and consistency against the built-in game AI.

The findings of this study aimed to contribute to the reinforcement learning field and the development of intelligent NPCs in commercial games. More broadly, success in fighting game AI development demonstrates capabilities relevant to real-world decision-making systems, such as autonomous robotics and interactive digital environments. This research may also inform the design of personalized AI opponents that mimic human play styles, helping bridge the skill gap for amateur and casual players.

Furthermore, the study acknowledges the growing importance of explainable AI. Understanding and interpreting neural decision-making is crucial for user trust and transparency, especially in interactive settings (Jeyakumar et al., 2020; Samek et al., 2021). Future work may explore visualizing learned policies and mapping agent behaviors to human-understandable formats (Aamir et al., 2021).

Ultimately, this research contributes to developing adaptive, real-time, vision-based game-playing agents and explores the potential for human-AI collaboration in competitive environments. By showing how self-learned agents can master complex behaviors, it supports the broader vision of generalizable AI systems that extend beyond games into dynamic, high-stakes real-world applications (Gallotta et al., 2024; Janiesch et al., 2021; Shao et al., 2019).

Methods

Research Design

This study implemented a Proximal Policy Optimization (PPO)-based reinforcement learning (RL) agent to master Street Fighter II: Special Champion Edition using the OpenAI Gym Retro environment. The methodology is grounded in the Stable Baselines3 framework, with custom modifications for training stability and performance optimization. The researchers add further detail to the experimental setup, learning framework, and evaluation protocols, as shown below.

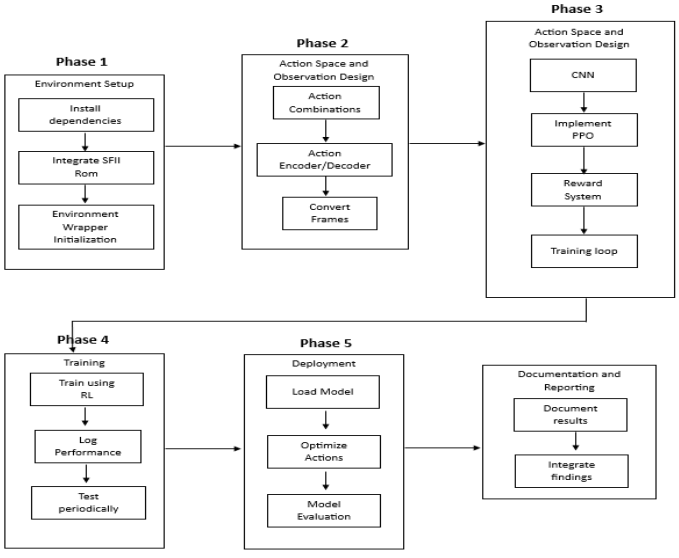


Figure 1. *Project Design of the Proposed System*

Environment Setup

The experimental environment was established using OpenAI Gym Retro, which provided a stable interface between the reinforcement learning algorithm and the Street Fighter II game engine. The game ROM was configured to initialize matches in the "Champion.Level12.RyuVsBison" state, with observations processed as grayscale images of 84×84 pixels. To capture temporal information, the system maintained a stack of four consecutive frames that were fed as input to the neural network. The action space was filtered to remove physically impossible move combinations, reducing the dimensionality of the action space from 12 to 8 valid actions.

Proximal Policy Optimization

The core learning algorithm implemented was Proximal Policy Optimization (PPO), chosen for its sample efficiency and training stability. The policy network architecture consisted of three convolutional layers for feature extraction, followed by two fully connected layers for policy and value estimation. Hyperparameters were carefully tuned, with the learning rate initialized at 2.5e-4 and gradually decayed to 2.5e-6 using a linear scheduler, while the clip range was similarly scheduled from 0.15 to 0.025 to ensure stable policy updates throughout the training process. This study implemented a Proximal Policy Optimization (PPO)-based reinforcement learning (RL) agent to master Street Fighter II: Special Champion Edition using the OpenAI Gym Retro environment. The methodology is grounded in the Stable Baselines3 framework, with custom modifications for training stability and performance optimization.

Model Training and Testing

Model Configuration

Training was conducted across 16 parallel environments to improve sample diversity and reduce wall-clock training time. Each training iteration collected 512 steps from every environment, resulting in batches of 8,192 steps for policy updates. The model underwent four (4) epochs of minibatch updates per iteration, with gradient steps computed using the Adam optimizer. Checkpoints were saved every 500,000 environment steps for progress monitoring and potential training resumption. The complete training regimen spanned 100 million timesteps, with performance evaluated periodically against the game's built-in AI opponents.

Evaluation

Agent performance was evaluated through 1,000 consecutive matches against the highest-difficulty M. Bison, the game's final boss character known for aggressive AI patterns and high damage output. Two quantitative metrics were prioritized: win rate and average reward points of the model.

$$WR_{Bison} = (\frac{Wins\ vs. Bison}{Total\ Matches\ vs. Bison}) \times 100\%$$

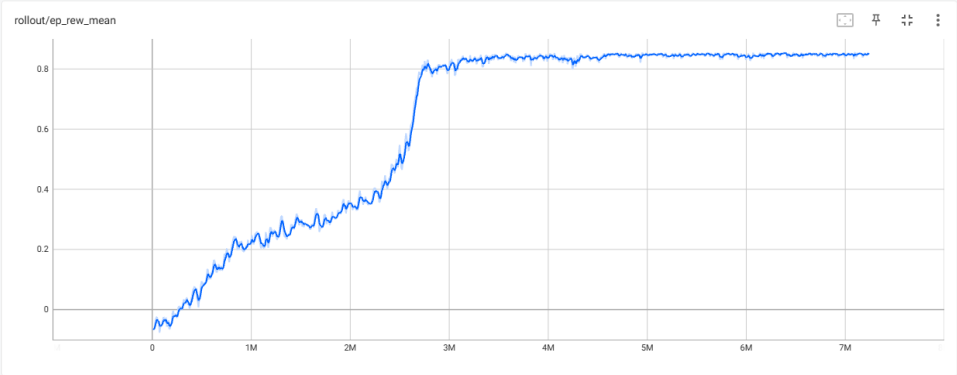
Equation 1. *Win-rate Formula*

$$AverageReward\backslash(= \frac{1}{N}\sum_{i=1}^{N} \quad (\sum_{t=0}^{T_i} \quad R_t^{(i)})$$

Equation 2. *Average Reward Formula*

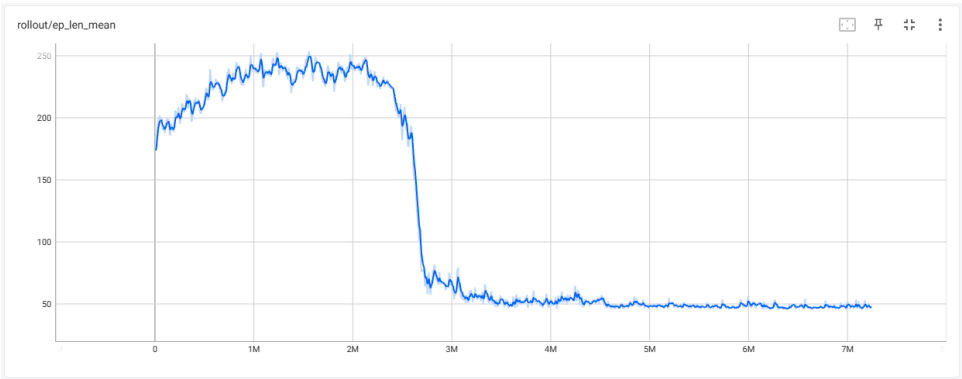
Results and Discussion

Graph 1 shows the mean reward per episode over time, and it directly tracks the agent's performance. A higher mean reward indicates the agent is learning and succeeding more often. The graph shows a clear upward trend, stabilizing around 0.8, indicating consistent performance gains and a higher win rate.



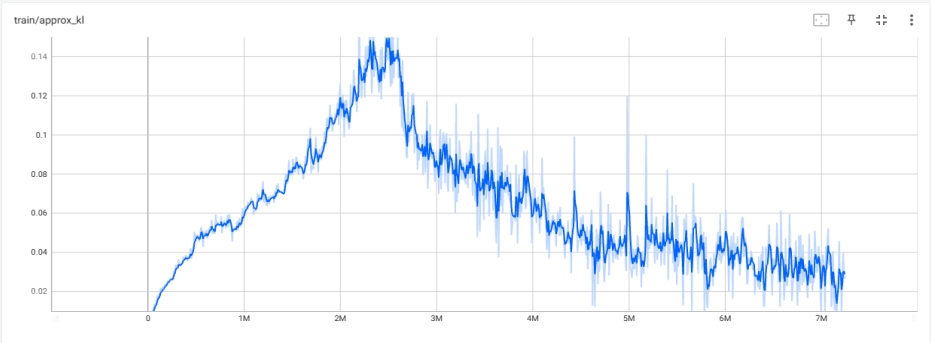
Graph 1. *Average Reward per Episode*

Graph 2 displays the average episode length over time. In fighting games, like Street Fighter, a shorter episode after training might indicate the agent is efficiently winning matches. The drop in the graph around 2.5M timesteps coinciding with the spike in rewards suggests the agent got better at winning fast and more efficiently.



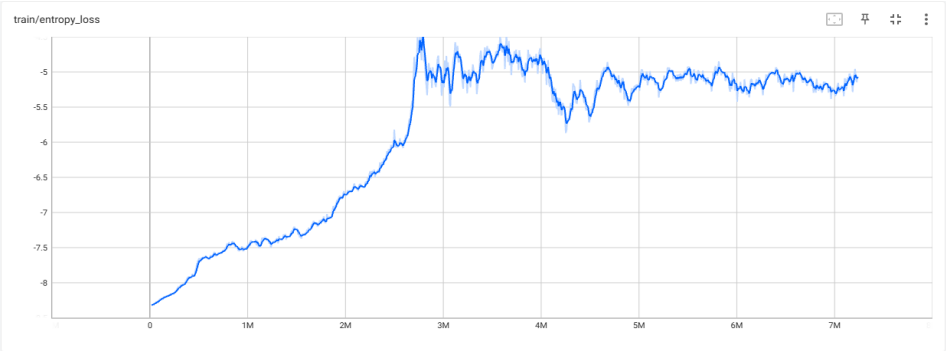
Graph 2. *Average Episode Length*

In Graph 3, it is rising steadily up to around 2.5M timesteps, peaking, and then gradually decreasing and stabilizing. This is expected in well-behaved PPO training. Early on, larger policy updates are common, but as the model improves and approaches optimal behavior, the updates become smaller. The drop after the 3M timesteps could be linked to the agent achieving good performance and the optimizer taking more cautious steps. This means that training is progressively fine-tuning the policy instead of wildly swinging back and forth.



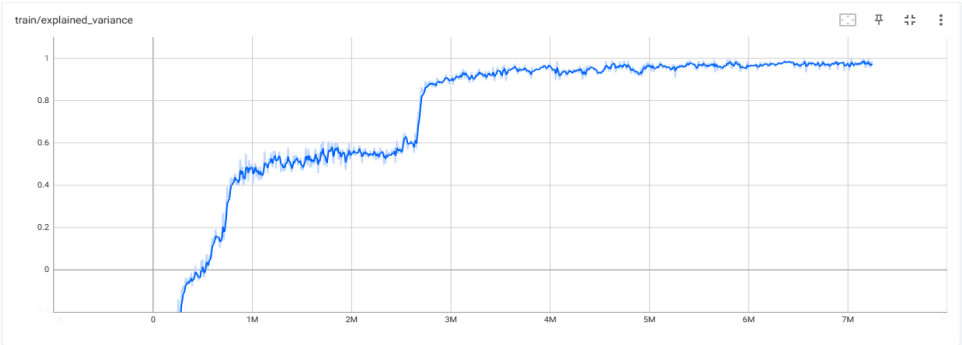
Graph 3. *Approximate KL Divergence*

The entropy loss rises (less negative) as training progresses, which seems counterintuitive because it is usually expected to decrease over time. However, note that entropy loss is typically a negative value. As it approaches -5 and flattens after around 3M, it suggests the agent's action choices are stabilizing, focusing more on high-probability actions. The initial rise in Graph 4 indicates the agent started very random and then reduced exploration as it learned, which shows a healthy transition from exploration to exploitation.



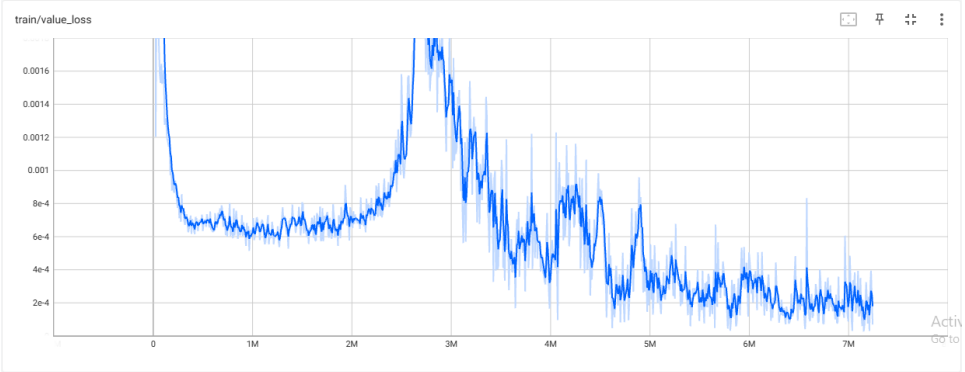
Graph 4. *Entropy Loss*

Graph 5 measures how well the value function predicts actual returns. A value close to 1 indicates perfect prediction, while a near 0 means no correlation. The explained variance rapidly increases to around 0.5 by 1M, then makes another leap to 0.9+ by 3M and holds steady. This pattern shows that early in training, the value function improved quickly and was fine-tuned to consistently explain a large portion of the returns.



Graph 5. *Explained Variance*

In Graph 6, the value loss starts high, drops quickly, and then has a couple of spikes around 2M to 3M before settling down. The spikes correlate with significant policy changes, as seen in Graph 3, which can momentarily destabilize the value estimates. After 4M, it steadily decreases and flattens at a low, stable value.



Graph 6. *Value Loss*

Figure 1 and 2 display the game's UI and the terminal, where the game logs are printed. The logs were separated into each round and each action. It logs the reward of that action, the changes in the player's or enemy's health, and the total reward of that round. At the end of the specific number of matches, it prints the win rate and average reward of the simulation.

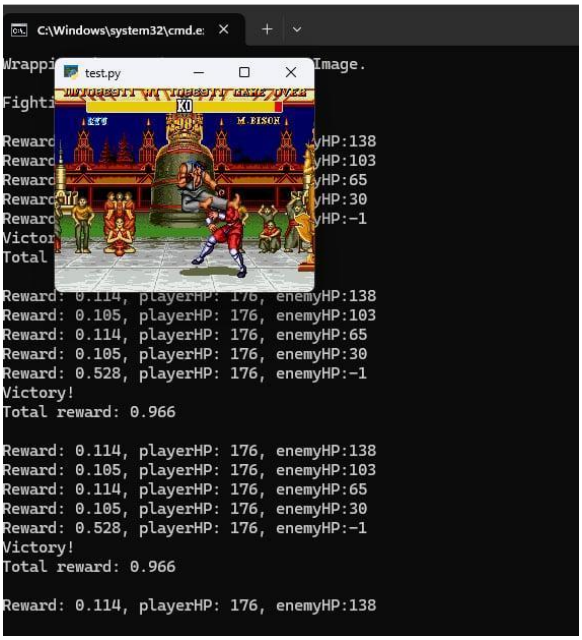


Figure 1. *Screenshot of Testing Phase*

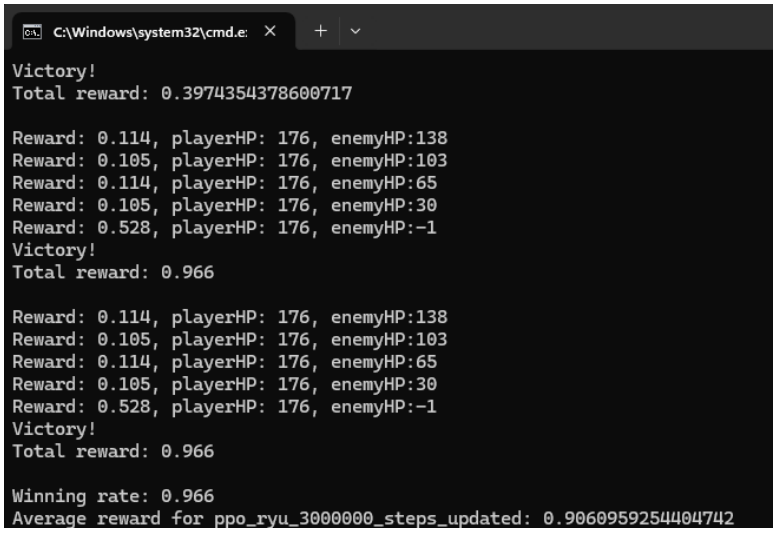
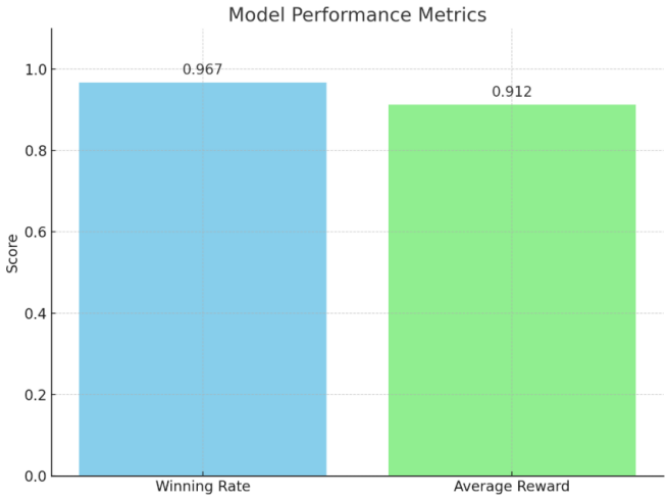


Figure 2. *Screenshot of Final Results*

After 1,000 matches, the model agent obtained a 96.7% win rate and an average reward of 0.912, as shown in Graph 7.



Graph 7. *Win Rate and Average Reward*

Conclusion and Future Works

This study successfully demonstrated the effectiveness of a reinforcement learning agent using the Proximal Policy Optimization (PPO) algorithm in mastering Street Fighter II: Special Champion Edition against the highest-difficulty opponent, M. Bison. The trained agent achieved a remarkable 96.7% win rate and an average reward of 0.912 after 1,000 consecutive matches. The training metrics supported these results, with the average reward per episode steadily increasing and stabilizing at a high level, indicating consistent performance improvements. The decreasing average episode length suggested the agent not only won more often but did so more efficiently as training progressed. Furthermore, the trends in approximate KL divergence and entropy loss confirmed a healthy transition from exploration to exploitation, while explained variance and value loss patterns indicated increasingly accurate and stable value function predictions. These findings confirm that reinforcement learning and carefully

monitored training metrics can produce highly effective agents in fast-paced competitive environments such as fighting games.

Beyond these technical achievements, the study contributes to the growing field of reinforcement learning in competitive gaming by demonstrating the viability of PPO in handling the dynamic, real-time demands of a fighting game environment. Unlike many RL applications focused on static or turn-based games, this research highlights the algorithm's ability to develop adaptive, reward-driven strategies under high-pressure conditions using only visual inputs. By achieving expert-level performance against a complex AI opponent, the study provides a strong case for RL's applicability in developing intelligent, responsive game agents. This is a foundation for future advancements in adaptive AI for gameplay enhancement and human-AI interaction in interactive entertainment and training scenarios.

For future work, several promising directions are proposed to enhance the agent's versatility and the scope of the study. One extension involves generalizing the agent to handle multiple characters and varying opponent strategies to increase adaptability. Additionally, integrating opponent modeling techniques could enable the agent to anticipate and counter diverse play styles more effectively. Evaluating the agent against human players of varying skill levels would also provide valuable insights into its real-world applicability. Exploring multi-agent reinforcement learning (MARL) setups, where the agent cooperatively or competitively interacts with other AI-controlled characters, presents another avenue for advancing strategic decision-making. Refining the reward function to consider additional gameplay aspects such as combos, defensive strategies, or flawless victories could further enhance the agent's tactical depth. Lastly, optimizing the training environment through parallelization, model compression, or accelerated simulations would support faster experimentation and more efficient deployment. Pursuing these future directions would not only strengthen reinforcement learning applications in gaming but also contribute to AI research in interactive and adversarial domains.

References

[1] Aamir, A., Tamosiunaite, M., & Wörgötter, F. (2021). Caffe2Unity: Immersive visualization and interpretation of deep neural networks. *Electronics, 11*(1), 83. https://doi.org/10.3390/electronics11010083

[2] Almeida, P., Carvalho, V., & Simões, A. (2024). Reinforcement learning as an approach to train multiplayer first-person shooter game agents. *Technologies, 12*(3), 34. https://doi.org/10.3390/technologies12030034

[3] Alonso, E., Peter, M., Goumard, D., & Romoff, J. (2021). Deep reinforcement learning for navigation in AAA video games. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)* (pp. 2133–2139). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2021/294

[4] Andrychowicz, M., Raichuk, A., Stańczyk, P., Orsini, M., Girgin, S., Marinier, R., Hussenot, L., Geist, M., Pietquin, O., Michalski, M., Gelly, S., & Bachem, O. (2021).

What matters for on-policy deep actor-critic methods? A large-scale study. *OpenReview.* https://openreview.net/forum?id=nIAxjsniDzg

[5] Ashktorab, Z., Liao, Q. V., Dugan, C., Johnson, J., Pan, Q., Zhang, W., Kumaravel, S., & Campbell, M. (2020). Human-AI collaboration in a cooperative game setting. *Proceedings of the ACM on Human-Computer Interaction, 4*(CSCW2), 1–20. https://doi.org/10.1145/3415167

[6] Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., Józefowicz, R., Gray, S., Olsson, C., Pachocki, J., Petrov, M., De Oliveira Pinto, H. P., Raiman, J., Salimans, T., Schlatter, J., & Zhang, S. (2019). Dota 2 with large scale deep reinforcement learning. *arXiv.* https://doi.org/10.48550/arxiv.1912.06680

[7] Clifton, J., & Laber, E. (2020). Q-Learning: Theory and applications. *Annual Review of Statistics and Its Application, 7*(1), 279–301. https://doi.org/10.1146/annurev-statistics-031219-041220

[8] Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review, 40.* https://doi.org/10.1016/j.cosrev.2021.100379

[9] Gallotta, R., Todd, G., Zammit, M., Earle, S., Liapis, A., Togelius, J., & Yannakakis, G. N. (2024). Large language models and games: A survey and roadmap. *arXiv.* https://arXiv.org/abs/2402.18659

[10] Goldwaser, A., & Thielscher, M. (2020). Deep reinforcement learning for general game playing. In *Proceedings of the AAAI Conference on Artificial Intelligence, 34*(2), 1701–1708. https://doi.org/10.1609/aaai.v34i02.5533

[11] Halina, E., & Guzdial, M. (2022). Diversity-based deep reinforcement learning towards multidimensional difficulty for fighting game AI. *arXiv.* https://arXiv.org/abs/2211.02759

[12] Hazra, T., & Anjaria, K. (2022). Applications of game theory in deep learning: A survey. *Multimedia Tools and Applications, 81*(6), 8963–8994. https://doi.org/10.1007/s11042-022-12153-2

[13] Hu, C., Zhao, Y., Wang, Z., Du, H., & Liu, J. (2023). Games for artificial intelligence research: A review and perspectives. *arXiv.* https://arXiv.org/abs/2304.13269

[14] Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets, 31*(3), 685–695. https://doi.org/10.1007/s12525-021-00475-2

[15] Jeyakumar, J. V., Noor, J., Cheng, Y., Garcia, L., & Srivastava, M. (2020). How can I explain this to you? An empirical study of deep neural network explanation methods. In *Advances in Neural Information Processing Systems (NeurIPS).*

[16] Joo, H., & Kim, K. (2019). Visualization of deep reinforcement learning using Grad-CAM: How AI plays Atari games? In *2019 IEEE Conference on Games (CoG)* (pp. 1–8). IEEE. https://doi.org/10.1109/cig.2019.8847950

[17] Li, S. E. (2023). Deep reinforcement learning. In *Reinforcement Learning for Sequential Decision and Optimal Control* (pp. 227–255). Springer. https://doi.org/10.1007/978-981-19-7784-8_10

[18] Osborn, J. C., Lederle-Ensign, D., Wardrip-Fruin, N., & Mateas, M. (2023). Combat in games. *eScholarship.* https://escholarship.org/uc/item/9zj6r5wz

[19] Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., & Müller, K. (2021). Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE, 109*(3), 247–278. https://doi.org/10.1109/jproc.2021.3060483

[20] Shao, K., Tang, Z., Zhu, Y., Li, N., & Zhao, D. (2019). A survey of deep reinforcement learning in video games. *arXiv.* https://arXiv.org/abs/1912.10944

[21] Simonov, A., Zagarskikh, A., & Fedorov, V. (2019). Applying behavior characteristics to decision-making process to create believable game AI. *Procedia Computer Science, 156*, 404–413. https://doi.org/10.1016/j.procs.2019.08.222

[22] Taherdoost, H. (2023). Deep learning and neural networks: Decision-making implications. *Symmetry, 15*(9), 1723. https://doi.org/10.3390/sym15091723

[23] Wang, D., Weisz, J. D., Muller, M., Ram, P., Geyer, W., Dugan, C., Tausczik, Y., Samulowitz, H., & Gray, A. (2019). Human-AI collaboration in data science: Exploring data scientists' perceptions of automated AI. *Proceedings of the ACM on Human-Computer Interaction, 3*(CSCW), Article 211, 1–24. https://doi.org/10.1145/3359313

[24] Yin, Q., Yang, J., Huang, K., Zhao, M., Ni, W., Liang, B., Huang, Y., Wu, S., & Wang, L. (2023). AI in human-computer gaming: Techniques, challenges and opportunities. *Machine Intelligence Research, 20*(3), 299–317. https://doi.org/10.1007/s11633-022-1384-6

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgements

The researchers would like to express their appreciation and gratitude to the New Era University professors for their instruction, without which this research would not have been feasible; to their parents for providing unconditional love and encouragement to always do their best; and to their friends for their assistance and support, whose practical suggestions significantly influenced the outcome of this study.

The researchers are most grateful to God, the source of all knowledge and wisdom, for His unending love and support.

Artificial Intelligence (AI) Declaration Statement

Artificial Intelligence (AI) tools were utilized in both the development of the code structure and the preparation of this manuscript. Specifically, OpenAI's ChatGPT was employed to assist in structuring and refining Python code related to reinforcement learning and gameplay automation and to support the writing, editing, and organization of the thesis text. The AI was used as a collaborative assistant to enhance clarity, coherence, and technical accuracy throughout the paper. All AI-generated content was thoroughly reviewed, edited, and verified by the author to ensure correctness, originality, and alignment with the research objectives. The final content reflects the author's critical evaluation and intellectual contribution.