Dataset and AI in Detecting Player Cheats in FPS Games

Ming Khye Chong*

School of Computer Science, University of Nottingham, Ningbo, Zhejiang, China

Abstract. The evolution of cheating in first-person shooter (FPS) games has gone from simple memory editing to sophisticated uses of AI, which are major threats to the fairness of games and the player experience. As competitive gaming and esports grow rapidly, the social and economic impacts of cheating continue to increase, and studies show potential player churn rates of 78%. Cheating was typically addressed using traditional anti-cheat systems that used a signature-based detection method. However, the use of computer vision and hardware assistance by modern cheats allows them to bypass traditional detection methods. This review examines methods of artificial intelligence and machine learning to detect cheaters in FPS games as a result of a thorough review of literature articles that have explored various methods of detecting cheaters, which include XGBoost, random forest, deep learning architectures like CNN and LSTM, and transformer-based models. The findings show that space-temporal analysis hybridized models have detection accuracies of over 99%. Moreover, even the unsupervised forms of cheat detection are potential in detecting some new forms of cheats. The boundary between human and AI cheats will always be a challenge to the anti-cheat systems because of the possibility of humans being labeled as elite.

1 Introduction

The problem of cheating in first-person shooter (FPS) games makes gaming unfair and will force numerous players and the gaming industry developers to quit cheating. The studies indicate that 78 percent of those who play FPS leave games in which they are playing once they encounter an excessively large number of cheaters [1]. Cheating has a bad effect on the game society. Furthermore, FPS games have changed as compared to the early times when players used to have mini-bouts with their friends in their residences. The number of international competitions that people view online is high. The forms of cheating have evolved to suit the existing esports. The simple programs used by cheaters in the past to edit simple numbers included Cheat Engine. Cheaters have already advanced into next-level cheating like DMA cards and artificial intelligence, which makes it harder to detect, and consequently, the old method of detecting cheaters will not be a viable method anymore [2, 3]. The problem is that cheaters will be located somewhere outside the game; hence, the old way of identifying cheaters will not be relevant any longer. The game developers have to offer alternatives to put a check on the vice.

The existing anti-cheat systems fail to register the existing cheats; they only identify familiar cheat software. A case in point would be Valve Anti-Cheat, since it operates in User Mode and it does signature scanning [3]. However, cheats have evolved, and they can now see what is on the screen and can move the mouse as any normal person can; the previous method of scanning the software to detect cheats will not be able to see them. Additionally, new cheats use special methods to imitate human movements, which makes it difficult to determine if someone is skilled or using a cheat [4].

Several researchers are working to use AI to find players who cheat in FPS games, as, in recent years, the prevalence of players using cheats to win has caused other players immense anger at these players. In an example, Park et al. studied ESP cheats using adversarial reinforcement learning and found that transformer-based methods such as AntiCheatPT can achieve an AUC of 0.9336, which is a very high score, while other researchers have mixed LSTMs and CNNs and also obtained very good results, achieving an accuracy of up to 99.87% [5]. AI and machine learning are better than traditional methods of cheating detection due to the advanced ability of deep learning models to identify various patterns of player behavior, compared to rule-based systems that cannot identify these patterns. For this reason, many gaming companies today are exploring how to use AI in the fight against video game cheating.

This paper describes how AI and machine learning can detect cheaters in FPS games. It will describe traditional machine learning methods (e.g., XGBoost and Random Forest), which are quick and easy to apply, and deep learning models (e.g., CNN and LSTM). Apart from that, it will look at new methods like Transformers. It will also look at how good these methods are in detecting different kinds of cheaters (e.g., memory hacks, which are easy to detect, compared with more complex forms, such as aimbots based on AI, which are much harder to detect). Finally, it will address some

* Corresponding author: mingkhye@qq.com

problems that come with the detection of cheaters. E.g., it has to be fast enough so that players will not notice any delays while playing; conversely, as it will get harder and harder to detect cheaters in the future, it will become more difficult to tell whether a player is cheating, the more skillful the user of cheater software becomes. Finally, it will look at some possible future directions this research on the topic might take, namely federated learning, explainable AI, and new defence mechanisms, and why it is so important for the future of video games.

The paper will look at how AI or machine learning methods can spot cheating behaviour in FPS games. Afterwards, it will see how well some existing algorithms work against different behaviours of cheaters. This review will have three basic parts: analysis of traditional machine learning models, analysis of deep learning designs, and analysis of alternative approaches still under development. Challenges to implementing these solutions will also be discussed, as will future research opportunities.

2 The history of cheat development

In the 1980s and early 1990s, cheat codes were tools that made challenging games genuinely beatable. They weren't considered shameful [6]. For example, Game Genie can simply insert a cartridge-style device between the game and the console to play, and players can cheat by entering the code in a booklet to make character gains infinite lives or absurd powers [6]. It worked by intercepting the game's memory and changing specific values on the fly [6]. Then, in 2000, the golden age of PC gaming was ushered in, and the new cheat codes had been used by player widely. Those cheating codes are God Mode and No Clip, along with a variety of console commands [6]. God Mode can make the player invincible, and No Clip allows players to move through the walls and obstacles [6]. Console commands can alter game speed, spawn items, and more [6]. God Mode and No Clip were developed for internal testing or vulnerability detection purposes [6]. They were not designed as cheat tools but have been repurposed by users. Players use those cheats to enhance the gaming experience, test features, or simply for entertainment [6]. In 2000, the memory editors and trainers became popular; these tools can allows player directly modify the game data stored in memory [6]. Cheat Engine is a memory scanner and debugger created in 2000 [6]. Cheat Engine allows players to find and edit the value, view, and modify disassembled code [6]. Cheat Engine's hex editor forms the core foundation of these tools, enabling users to directly edit in-game memory addresses [6]. These abilities allow player to have complex cheats like infinite life and armour, visual hacks such as wallhacks or field of view adjustments [6]. With the surge in popularity of FPS games, multiplayer and team-based competitive modes have been introduced one after another, leading to an increasingly rampant proliferation of these cheats. Wallhacks, Aimbots, and ESP became prevalent in those FPS games. Wallhacks allow players to shoot through the wall, Aimbots can automatically aim the head of the enemy, and ESP can show critical information like resource location, player health, or more [6]. These cheats severely undermine the fairness of the game, which has a significant negative impact on its reputation. If there is no regulation for this, it will ruin the game.

This Cheaters vs. Anti Cheat battle gets serious when kernel-level is found [6].

There are two levels of privilege in the CPU called Ring 0 and Ring 3 [6]. Ring 3 is for normal players, which means it only has restricted access to system resources [6]. Ring 0 is Kernel-mode and it has the highest privilege level [6]. Cheaters can have all the hardware and memory by using Kernel-mode drivers [6]. Cheaters can have their own software modules in Ring 0 to cheat in the game [6]. This cheat is hard to detect because calls from its drivers intercept the system call, which means this driver is hiding the cheat in the anti-cheat, in other words [6].

However, the traditional way to develop anti-cheat is to focus on the way that only detects unauthorized software signatures inside the game, which is almost easy to bypass [2]. For instance, the Valve Anti-Cheat runs on User Mode Ring 3, so it almost totally depends on the signature to detect the known cheat program [3]. Many systems have changed rapidly in the last few years. Now, a lot of them use kernel-level detection, which is how Riot Vanguard works. Consequently, the cheat developers have now turned to designing cheats, which incorporate the outside hardware, like the Direct Memory Access cards and the computer vision cheats that are AI-related [2, 3]. The primary difference between these two forms of cheats is that in the new cheat codes, the system does not directly alter the game code or game memory, but instead reads what has been displayed on the screen and then simulates human action. These new cheat systems, therefore, are hard to detect through software-based cheat detection systems [2].

These sophisticated cheats bring about serious cheating. They provide a degree of competitive imbalance, the significance of which is too high in FPS games. They demotivate regular players. Research indicates that cheating helps to deter approximately 78 percent of the players from the game [1]. The effect of this is the possibility of billions of dollars being lost by game companies. Cheating also complicates the process of the developers distinguishing between a good, fair player and a cheater. Mature cheaters are able to render an aim fluid besides simulating a natural response time. An advanced cheat can be concealed with Bézier curves or Gaussian noise by the developers, and it will seem like the cheater is a genuine player [4]. It is due to this that cheat detection has to develop. Genuinely basic signature-based cheat recognition is no longer sufficient. Rather, developers are supposed to put their efforts into understanding their player behaviours. Machine learning can be useful in this undertaking. Machine learning can be used by the developer to spot recurring behaviours through time and detect anomalous behaviours. Machine learning would also assist the developers in detecting human-like advanced cheats.

3 The summary of the core machine learning and deep learning models in cheat detection

The modern anti-cheat systems use various machine learning and deep learning methods. The models are good at various tasks, e.g., one of the most popular models is called XGBoost, and it is a decision tree-based model, which is accurate, very fast, and works well on structured data. The other model is the Random Forest and is also a decision tree model; it follows the concept of using multiple decision trees together in an ensemble to improve its performance by reducing errors. Support Vector Machines are also developed to distinguish between normal and cheating players by identifying a line between them. Moreover, K-Nearest Neighbors is used to compare players, according to their similarity with other players, and detect cheaters. In comparison to these models, deep learning will automatically learn patterns on unstructured raw data and learn more complex patterns than other models. VADNet, a model that will be used specifically in anti-cheat applications, applies to visual information and player behaviour in order to differentiate the actions of a player and actions perceived. Lastly, LSTM and GRU are two that are best adapted to sequential data, given that they are capable of remembering the event sequence they have observed so far, hence can detect robotic behaviours that change over time.

4 Methodology, effect, and applicability of AI in FPS game security

Until the inception of neural networks, cheater detectors used to be produced through statistical testing. Machine learning also has many applications for real-time detection at the client side, and at a good speed. XGBoost is a model that performs well. It builds up decision trees sequentially to get an estimated output. Has shown up to 97.7% accuracy with structured data. Another common model is Random Forest, which creates a larger number of decision trees, which gives more stability to the classifier [7,8]. There are many other models that can be found in the literature that have been used; some of which include SVM and KNN. SVM uses the best line to separate normal players from cheaters, and KNN uses the closest data points to see if someone is cheating. These methods of clustering player behaviour can also be found in the literature [9].

Cheat detection is done using deep learning. Since game-play data can be represented as a time series, there are long-term dependencies. One type of deep learning model is a CNN, specifically VADNet, which processes video frames and detects visual anomalies such as ESP overlays [10]. Additionally, RNNs can also be used for cheat detection; LSTM and GRU are two examples. They are able to remember information in the long-term and analyze aiming patterns and other non-human-like movements [11,12].

Some new models are available as well, including transformer models, which can evaluate different time segments and determine which segments are most relevant for creating cheating patterns over extended periods, resulting with an AUC of 0.9336 [1]. There are also hybrid systems such as LSTM-CNN, which combines LSTM (for memory of temporal detail) and CNN (for extracting spatial features). Using these two models together allows for an accuracy level of approximately 99.87% on the prediction accuracy score [5].

Unsupervised learning can be a great tool for discovering cheats. The two most popular methods for using unsupervised learning to find cheats are through the Isolation Forest Method and LOF. Isolation Forest creates short paths, within the tree structure, through other randomly selected features of the data. LOF utilizes a density approach to determine how much denser the density is around that point compared to its neighbours. Both methods treat cheaters as outliers and therefore, do not need to know what type of cheaters they are going to find when utilizing unsupervised methods with cheaters [13].

5 Challenges and future directions

5.1 Current challenges in AI-driven cheat detection

It's tough to identify the talented player from the cheating player. Professional players have similar reaction times to aimbots [1]. Today's cheats use Bézier curves to replicate human movement, which makes their aim look just as good as high-level human play [4]. This means that it could accidentally ban talented players, damaging the trust that exists within the community [9]. Cheating developers are always evolving their cheats. They compete against detection systems [1]. They can now use adversarial machine learning to modify a cheat in tiny ways that fool detection models [14]. Technical issues also exist. Complex models lead to performance issues in games. This could be a problem for any real-time game where lag is an issue [7]. Finally, there are a lot of honest players, but too many cheaters for model training to be done correctly [15].

5.2 Promising future research directions

The emergence of proactive defence methods is not uncommon. The use of an Invisibility Cloak framework adds minor graphical changes to video game graphics that help lower visual confidence level of vision-based aimbots without having any negative impact on human players [16]. Federated Learning is promising. It can detect cheats but still allow users to maintain their privacy because models will be trained in a decentralized manner, which means only the model parameters will be sent to a central server, and the raw data from the user will remain on their local device [17]. Another emerging trend is the need for Explainable AI. Systems that use Explainable AI, such as XGuardian, use SHAP values to explain ban decisions. They help moderators and players to understand why another player has been banned, and also provide transparency [17,12]. One more area being investigated is hardware-

assisted security through using Trusted Execution Environments, such as Intel SGX, to isolate any anti-cheat logic, thus making it more difficult for kernel-based malware to interfere [4,10].

6 Conclusion

The cheat developers designed the external hardware cheats, like Direct Memory Access cards (DMA) and AI-based computer vision cheats, making the traditional anti-cheat software unable to detect the cheat. The traditional machine learning algorithms are useful for real-time and low-latency detection, such as XGBoost used to process structured telemetry data, Random Forest, which combines multiple decision trees, and Support Vector Machines, that apply for the behaviour clustering tasks. The Deep learning and Temporal modelling methods can be widely used because the gameplay data naturally forms time series with long term dependencies. For example, VADNet architecture can process raw video frames and detect visual anomalies like ESP overlays, and LSTM analysis aiming trajectories, and identify non-human micro adjustments. However, there are plenty of challenges and problems that have not been solved. The Modern cheats imitate natural human movement, making the distinction in elite human skill from humanized AI cheats. The low adversarial adaptation of the anti-cheat software will be unable to detect the cheats that the cheat developers add by making small changes using adversarial machine learning techniques. The anti-cheat software complex models will cause high latency, which can reduce game frame rates. For the future development direction, Explainable AI can clearly justify the ban decisions to moderators and players. Federated learning will help to protect sensitive behavioural information. Proactive defence methods can reduce the confidence of vision-based aimbots without affecting human players by using the Invisibility Cloak framework. Lastly, hardware-assisted security can make kernel-level malware hard to detect.

References

1. Park, J. G. Lee, T. Kwon, et al., Adversarial Reinforcement Learning Framework for ESP Cheater Simulation. arXiv preprint, arXiv:2509.24274 (2025)
2. SciTePress, Research on the Application of Artificial Intelligence in FPS Game Cheat Detection. SciTePress Digital Library (2024)
3. Canute, Technical Analysis of Cheating in Video Games: Methods, Detection and Countermeasures. Canute Great Blog (2025)
4. ResearchGate, GAN-Aimbots: Using Machine Learning for Cheating in First Person Shooters. Publication 360556375 (2022)
5. PMC, A high-performance hybrid LSTM CNN secure architecture. PubMed Central (2025)
6. Cheat Vault, The Evolution of Game Cheats: From Code to Kernel. Cheat Vault Blog (2025) https://cheatvault.net/blog/cheats/history-of-game-cheats/
7. NHSJS, A New Multi-Tier Anti-Cheat Approach in Online FPS Games. Natl. High Sch. J. Sci. (2025)
8. irijournals.ir, Predicting Cheating Behavior in Multiplayer Online Games Using Machine Learning. IRI Journals (2023)
9. MDPI, Evaluating the Effectiveness and Ethical Implications of AI Detection Tools. Information (2025)
10. LSEEE, Detecting aimbot Cheats in FPS games using Computer Vision and RNN. Technical Report (2026)
11. Moonlight, Identify As A Human Does: A Pathfinder of Next-Generation Anti-Cheat Framework. arXiv preprint, arXiv:2409.14830 (2024)
12. Emergent Mind, XGuardian: Explainable AI Anti-Cheat for FPS. Technical Analysis (2026)
13. Kaggle, Anomaly Detection using Unsupervised Techniques (Isolation Forest/LOF). Kaggle (2026)
14. CEUR-WS.org, A Game for Crowdsourcing Adversarial Examples for False Information Detection. CEUR Workshop Proc. 3275 (2022)
15. WJARR, AI-powered anti-cheat engines: Real-time behavior analysis. World J. Adv. Res. Rev. (2025)
16. USENIX, Invisibility Cloak: Proactive Defense Against Visual Game Cheats. Proc. USENIX Secur. '24 (2024)
17. Frontiers, Deep federated learning: a systematic review of methods and challenges. Front. Comput. Sci. (2025)