Theory and Practice of Science and Technology
2025, VOL. 6, NO. 9, 85-89
DOI: 10.47297/taposatWSP2633-456917.20250609

A Survey of Recent Advances in Artificial Intelligence for Detecting Balance-Breaking Behaviors in FPS Games

Jingchen Yang

Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, 215000, China

ABSTRACT
First-person shooter games have become a cornerstone of esports due to their competitiveness and commercial value. However, balance-breaking behaviors such as cheating and smurfing severely undermine fairness, degrade player experience, and threaten the sustainability of the industry. Traditional rule-based or manual inspection approaches have proven inadequate against the growing sophistication of such behaviors. In recent years, artificial intelligence, with its strengths in pattern recognition, temporal modeling, and anomaly detection, has emerged as a promising solution for both research and industrial applications. This paper reviews major advances in AI-driven detection of balance-breaking behaviors in FPS games over the past decade, covering methods based on behavioral features, input device dynamics, vision-based analysis, and unsupervised learning, while comparing their strengths and limitations in terms of accuracy, efficiency, and robustness. Finally, the paper identifies critical challenges—including data scarcity, adversarial dynamics, and the lack of explainability—and proposes future directions focusing on large-scale open dataset construction, adversarial and adaptive modeling, and the development of explainable AI frameworks.

KEYWORDS
FPS games; Artificial intelligence; Cheat detection; Aimbot; Wallhack; Smurfing

1 Introduction

As one of the most popular game genres worldwide, first-person shooter (FPS) games not only attract a massive player base but also carry substantial market value and commercial significance in the field of esports [1]. The core appeal of FPS games lies in fair competition. However, various balance-breaking behaviors continue to threaten the integrity of the gaming ecosystem. Traditional anti-cheat mechanisms, which largely rely on rule-based detection or manual inspection, have proven inadequate in addressing the evolving sophistication of cheating strategies [2]. In recent years, artificial intelligence (AI), with its strengths in pattern recognition, temporal sequence modeling, and anomaly detection, has emerged as a promising approach for both academic research and industrial applications [3-4]. A systematic review of recent AI-driven advances in detecting balance-breaking behaviors in FPS games can not only summarize the effectiveness and limitations of existing methods but also provide valuable insights for the development of more intelligent and efficient anti-cheat systems in the future.

2 Types and impacts of balance-breaking behaviors

2.1 Game cheating

Aimbot and wallhack represent the most prevalent forms of cheating in FPS games. The former manipulates mouse inputs or memory variables through external programs, enabling players to instantly lock onto and hit targets, thereby significantly altering the natural distribution of aiming trajectories [2-3,5-8]. The latter modifies rendering processes or memory data to allow players to observe opponents through obstacles [9-10]. These types of cheats typically manifest in unnaturally high accuracy rates, superhuman reaction times, or abnormal attention to enemies that should otherwise be hidden from view. With their increasing stealth and adversarial sophistication, such behaviors have become central targets of AI-driven anti-cheat research.

2.2 Smurfing

Smurfing refers to the practice in which highly skilled players create alternate accounts to compete against lower-ranked opponents, thereby gaining disproportionate advantages [4,11-14]. The motivations behind smurfing include recreational dominance over inexperienced players, concealing true ranking status, or engaging in boosting for profit [15]. Unlike cheating tools such as aimbots or wallhacks, smurfing does not rely on unauthorized software but is instead reflected through anomalous behavioral patterns, such as rapid performance improvement within a short period or disproportionately high achievements compared to novice groups [3]. The difficulty lies in the fact that smurf behaviors may overlap with the natural performance fluctuations of genuine new players, making AI-based detection particularly

challenging.

2.3 Impacts of balance-breaking behaviors

Balance-breaking behaviors in FPS games exert significant negative impacts at both the player and industry levels. For players, cheating tools such as aimbots and wallhacks, as well as smurfing practices, place ordinary participants in inherently unequal competition, fostering frustration and discouragement that diminish perceptions of fairness and enjoyment, while also shortening players' overall engagement lifecycles [2,4-5,8,10,13-15]. At the industry level, the persistent presence of cheating and smurfing threatens the sustainable development of games. On the one hand, such behaviors reduce player retention and willingness to spend, while simultaneously increasing the costs of maintenance and customer support for developers [2,5,16]. On the other hand, they jeopardize the credibility of competitive games, undermining the legitimacy and commercial value of esports events [3,10]. Consequently, advancing AI-driven detection and governance mechanisms has become a critical priority for both academia and industry in order to safeguard fairness and ensure the long-term viability of FPS ecosystems.

3 Comparative analysis

To summarize the reviewed methods, this paper abstract a general AI-based anti-cheat framework (see Figure 1). This framework organizes existing research into a four-layer structure: data input, feature processing, AI modeling, and decision outputs. Representative studies at each layer are indicated in the figure.

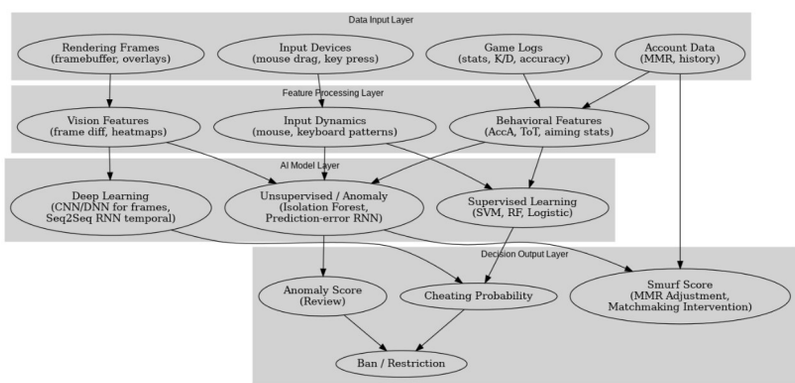

Figure 1 General AI anti-cheat framework for FPS games. The system consists of four layers: (1) Data input ; (2) Feature processing ; (3) AI models ; and (4) Decision outputs .

3.1 Game cheating

In recent years, artificial intelligence has made significant progress in detecting cheating behaviors in FPS games, particularly in identifying aimbots. For example, Yu et al. (2012) [8] proposed a heuristic method based on a distribution comparison matrix, combining statistical tests such as the T-test, F-test, and Kolmogorov – Smirnov (K-S) test to extract features from player behavior data. These features included Cursor Acceleration when acquiring Aim (AccA), absolute acceleration (abs(AccA)), and Time on Target (ToT). By constructing secondary features to capture deviations between sample distributions and training distributions, the study employed a linear classifier for binary classification. The method effectively addressed the limitations of traditional K-S testing when applied to high-dimensional data and intra-class distribution variance. Experimental results showed that it achieved a true positive rate of 98.21% and a true negative rate of 98.81% on real datasets, while offering superior computational efficiency compared to earlier voting-based K-S methods. Moreover, it supported multidimensional inputs and distribution-level comparisons, demonstrating strong potential for real-time detection.

Building on this foundation, Alayed et al. (2013) [5] advanced the practical utility and systematic application of behavior-based detection methods. Using a custom-built FPS game (*Trojan Battles*) as the experimental platform, the authors collected 18 features at the server side, including aiming accuracy, shooting behavior, and movement patterns. Multiple classifiers such as Logistic Regression and Support Vector Machines (SVMs) were systematically compared across time windows ranging from 10 to 90 seconds. The results revealed that adopting a category-specific detection strategy significantly improved accuracy for different aimbot variants, such as Lock, Auto-Switch, Auto-Miss, Slow-Aim, and Auto-Fire. For example, detecting "Lock" aimbots with an SVM-RBF model within a 60-second window achieved 98.9% accuracy, whereas multi-class detection often suffered from feature similarity that caused misclassifications. In addition,

the study proposed a threshold-based mechanism for real-time detection and emphasized the importance of feature interpretability and player privacy protection.

Compared with the work of Yu et al. (2012) [8], Alayed et al [5]. placed greater emphasis on the flexibility of practical deployment. On the one hand, the study defined a set of generic features applicable to multiple game genres as well as FPS-specific features, thereby enhancing the cross-game generalizability of the approach. On the other hand, by optimizing time windows and analyzing feature weights, the authors provided developers with a tunable parameter framework that could be adapted to different operational environments. However, both studies shared certain limitations: they relied heavily on supervised learning and high-quality labeled data, and neither fully addressed the problem of noisy data in high-latency environments. Taken together, these two works advanced the application of AI in game anti-cheat research from different perspectives—feature construction and systematic engineering practice—thus laying an important theoretical and practical foundation for subsequent studies.

Spijkerman and Ehlers (2020) [10] approached cheat detection from the perspective of input device dynamics, examining how mouse and keyboard interaction patterns could be leveraged to identify cheating behaviors. Their study compared the performance of several classifiers—including decision trees, SVMs, ensemble methods with feature combinations, and naïve Bayes—within the context of Counter-Strike: Global Offensive (CS: GO). Results indicated that decision trees performed strongly on unpruned keyboard dynamics with an accuracy of 95.45%, while pruning mouse dynamics data improved accuracy from 58.33% to 83.33%. In contrast, naïve Bayes proved effective only for categorical data derived from console logs, achieving an accuracy of 87.5%. Notably, mouse drag events were found to be highly discriminative due to their richness in behavioral information, although individual differences in hardware configurations introduced noise that could lead to false positives. Compared to Yu [8] and Alayed [5], this study highlighted the portability of device-level behavioral features across different games. However, its dataset was limited to single-player matches against bots, lacking validation in large-scale multiplayer environments, and the feature engineering remained relatively simple without fully exploiting temporal and contextual dependencies.

Jonnalagadda et al. (2021) [9] pioneered a vision-based approach to cheat detection by directly analyzing rendering frames to identify illegal overlays associated with wallhacks. The study constructed a large-scale dataset covering multiple games and various cheat software, and designed a dual-branch deep neural network (DNN): the global branch determined whether an entire frame had been tampered with, while the local branch produced pixel-level heatmaps indicating the likelihood of cheating, thereby enhancing interpretability and robustness. This work was among the first in the anti-cheat field to integrate confidence analysis, employing measures such as Variation Ratio and Uncertainty Loss to reduce false positives. In addition, the system adopted one-sided interval bound propagation (IBP) for adversarial defense, maintaining high detection rates even under strong adversarial attacks. Compared with the behavioral log-based approaches of Yu [8] and Alayed [5], this vision-based solution required no access to sensitive player data, relying solely on GPU framebuffer analysis. Furthermore, it incorporated adaptive retraining based on confidence monitoring, providing resilience against novel cheat software. However, its computational cost was substantial, particularly when processing 1080p frames, and the method depended on consistent rendering mechanisms across different games.

Unlike the server-side statistical analysis and supervised learning methods proposed by Yu [8] and Alayed [5], Choi et al. (2023) [3] introduced an innovative client-side unsupervised learning framework that effectively addressed two critical challenges: server computational bottlenecks and the scarcity of high-quality labeled cheat data. The core strength of this approach lies in its use of recurrent neural networks (RNNs) trained exclusively on normal player behavior. By modeling expected input patterns, the system identifies abnormal aiming behaviors through prediction errors, achieving an impressive detection accuracy of 97.64% on real CS:GO datasets. More importantly, the framework adopted a distributed architecture in which detection tasks were offloaded to individual clients, thereby alleviating server load. To ensure security and integrity, the system leveraged Intel SGX trusted execution environments to safeguard both detection logic and behavioral data, providing resistance against memory tampering and model theft attacks. Compared with device-level behavioral analysis by Spijkerman and Ehlers (2020) [10] —which is prone to individual variability—or the computationally expensive vision-based detection of Jonnalagadda et al. (2021) [9], Choi et al.' s [3] method demonstrated clear advantages in real-time efficiency, privacy preservation, and adversarial robustness. By integrating client-side detection, unsupervised learning, and hardware-based security, this work not only offered a feasible pathway for large-scale online games but also laid a solid foundation for the design of more general and efficient anti-cheat frameworks.

3.2 Smurfing

Academic research specifically addressing AI-based detection of smurfing remains relatively scarce, with existing studies often focusing more on player perceptions and sociopsychological perspectives—such as the impact of smurfing on user experience or community attitudes—rather than on technical detection models. For instance, McCauley (2023) [13] employed auto-netnography and interviews to qualitatively examine smurfing in FPS games, analyzing its diverse

motivations and its effects on players' prosumption experiences. Although "smurf detection" has not yet become a mainstream research topic, several recent studies and discussions are noteworthy.\

In the field of gameplay behavior modeling, Buckley et al. (2013) [11] collected raw keyboard and mouse input data from 34 participants across 212 matches, alongside background information such as weekly playtime and prior FPS experience. These variables were used as proxies for player skill. From the raw logs, the authors extracted multidimensional statistical features, including displacement in the x and y axes, absolute positions, key-hold durations, most frequently pressed keys, simultaneous keypresses, total event counts, and mean inter-event intervals. Using random forests as the classification model, they predicted players' skill categories and kill-death ratios. Experimental results demonstrated that solely based on players' keyboard and mouse interaction patterns, the model could classify player skill levels with an accuracy of up to 76%. Furthermore, comparable performance could be achieved using only the first 20 seconds of gameplay data. In addition, the study revealed that players' behavioral patterns shifted noticeably toward the end of matches, likely because experienced players strategically adjusted their actions in order to maximize their scores.

Similarly, Greige (2022) [12] proposed a two-stage unsupervised learning framework for anomaly detection. In the first stage, features such as kills per unit time, number of matches required to level up, average rank placement, and shooting accuracy were extracted from player logs. An Isolation Forest algorithm was then applied to compute anomaly scores for each player, with lower scores indicating stronger deviations from normal novice behavior. To further distinguish smurf accounts from those using explicit cheats, a second stage incorporated auxiliary features such as associated account count, duration of play, and ban or report history. Applied to real-world data from a major FPS title, the system flagged 182 anomalous accounts, of which 94% were later confirmed as violators through manual review, with smurfs comprising 65% of the cases. Statistical analysis revealed that these accounts consistently outperformed genuine novices in key indicators such as kill efficiency, ranking, and progression speed. By narrowing the review pool to only 1% of the original dataset, the method greatly improved the efficiency of manual investigations.

Beyond academic research, game companies have also deployed technical and systemic interventions. In response to the widespread smurfing problem in *VALORANT*, Riot Games [4] introduced an automated smurf detection system based on machine learning, which analyzes behavioral data to identify smurf accounts and rapidly adjusts their hidden matchmaking rating (MMR) to align with their true skill level. Field experiments demonstrated that adjusted smurf accounts could converge to their appropriate rank within just four matches, while the proportion of one-sided "stomp" games dropped from 32% to near-normal levels. Additionally, Riot modified its five-stack matchmaking restrictions, allowing players of any rank to team up while adjusting point gains for unbalanced squads, thereby reducing the incentive to create alternate accounts. As a result, the number of smurf accounts decreased by approximately 17%, and fairness in five-stack matches improved significantly, with over one-third of games ending in close scores. These measures provide a practical example of how machine learning – based interventions can mitigate smurfing and enhance the integrity of competitive play.

4 Challenges and future directions

Despite the significant progress that artificial intelligence has made in detecting cheating behaviors and smurfing in FPS games in recent years, the field still faces numerous challenges. First, difficulties in data acquisition and labeling remain the most pressing issue. The covert nature of both cheating and smurfing leads to data scarcity, and authentic cheating data are usually controlled by game developers, making them difficult for academic researchers to obtain. Public datasets are often limited in size or lack precise cheat annotations, which undermines the reliability of model training and evaluation. Furthermore, differences in game mechanics and data structures limit the transferability of models across titles. Second, adversarial and adaptive challenges are substantial. Cheat developers continuously update techniques to evade detection, such as simulating irregular human actions or randomizing behavioral patterns to confuse AI models. Similarly, smurf players may intentionally regulate their performance to conceal their true skill level. This adversarial arms race requires detection systems to maintain continuous updates and rapid responsiveness. However, achieving robustness while avoiding excessive punishment of falsely flagged players remains an unresolved problem. Third, fairness and explainability are increasingly emphasized. The black-box nature of many AI models makes it difficult for players to understand detection outcomes. If a player is wrongly banned due to misclassification, it may harm the developer's credibility and undermine player trust. Therefore, enhancing the interpretability of models—enabling systems to provide clear explanations for their decisions—will be crucial for ensuring both positive user experience and legal compliance.

In response to the aforementioned challenges, future research can be directed along several key avenues. First, the scarcity of datasets available for academic research remains a fundamental barrier. Constructing large-scale, open datasets and fostering collaboration between academia and industry will provide the foundation for subsequent advancements. Second, adversarial and adaptive modeling should be prioritized. Since cheating and smurfing are inherently dynamic and adversarial processes, traditional static models often experience severe performance degradation

when confronted with novel cheating patterns. Future work may therefore explore adversarial learning, transfer learning, and continual learning approaches to enhance the robustness and adaptability of detection systems in rapidly evolving environments. Finally, the development of explainability and trust mechanisms is essential. In particular, in sensitive scenarios such as account suspension, false positives can produce serious negative consequences. Developing explainable AI capable of providing transparent and verifiable detection outcomes will thus be critical to improving system transparency and fostering player trust.

References

[1] China Audio-Visual and Digital Publishing Association, E-sports Working Committee. *2024 China E-sports Industry Report - FX Report*. n.d. https://www.fxbaogao.com/view?id=4629699&requestId=&query=%7B%22keywords%22%3A%222024%E5%B9%B4%E4%B8%AD%E5%9B%BD%E7%94%B5%E5%AD% 90%E7%AB% 9E% E6%8A% 80%E4%BA% A7%E4%B8%9A% E6%8A% A5%E5%91%8A% 22%7D&index=0&pid= &xid= (accessed June 24, 2025).

[2] Collins S, Poulopoulos A, Muench M, Chothia T. Anti-Cheat: Attacks and the Effectiveness of Client-Side Defences. Proceedings of the 2024 Workshop on Research on offensive and defensive techniques in the context of Man At The End (MATE) attacks, Salt Lake City UT USA: ACM; 2024, p. 30–43.

[3] Choi M, Ko G, Cha SK. BotScreen: Trust Everybody, but Cut the Aimbots Yourself n.d.

[4] VALORANT Systems Health Series - Smurf Detection 2022. https://playvalorant. com/en-us/news/dev/valorant-systems-health-series-smurf-detection/ (accessed June 24, 2025).

[5] Alayed H, Frangoudes F, Neuman C. Behavioral-based cheating detection in online first person shooters using machine learning techniques. 2013 IEEE Conference on Computational Inteligence in Games (CIG), Niagara Falls, ON, Canada: IEEE; 2013, p. 1–8.

[6] Chen K-T, Pao H-KK, Chang H-C. Game bot identification based on manifold learning. Proceedings of the 7th ACM SIGCOMM Workshop on Network and System Support for Games, Worcester Massachusetts: ACM; 2008, p. 21–6.

[7] Liu D, Gao X, Zhang M, Wang H, Stavrou A. Detecting Passive Cheats in Online Games via Performance-Skillfulness Inconsistency. 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, USA: IEEE; 2017, p. 615–26.

[8] Yu S-Y, Hammerla N, Yan J, Andras P. Aimbot Detection in Online FPS Games Using a Heuristic Method Based on Distribution Comparison Matrix. In: Huang T, Zeng Z, Li C, Leung CS, editors. Neural Information Processing, vol. 7667, Berlin, Heidelberg: Springer Berlin Heidelberg; 2012, p. 654–61.

[9] Jonnalagadda A, Frosio I, Schneider S, McGuire M, Kim J. Robust Vision-Based Cheat Detection in Competitive Gaming 2021.

[10] Spijkerman R, Marie Ehlers E. Cheat Detection in a Multiplayer First-Person Shooter Using Artificial Intelligence Tools. Proceedings of the 2020 3rd International Conference on Computational Intelligence and Intelligent Systems, Tokyo Japan: ACM; 2020, p. 87–92.

[11] Buckley D, Chen K, Knowles J. Predicting skill from gameplay input to a first-person shooter. 2013 IEEE Conference on Computational Inteligence in Games (CIG), Niagara Falls, ON, Canada: IEEE; 2013, p. 1–8.

[12] Greige L. ANOMALY DETECTION IN COMPETITIVE MULTIPLAYER GAMES n.d.

[13] McCauley B. An Auto-netnographic Approach to Understanding Alternate Gaming Accounts: How Smurfing Impacts the Prosumer Experience in Counter-Strike:Global Offensive 2023.

[14] pmien996. The Hidden Crisis: How Smurfing is Undermining Competitive Integrity in Valorant. ReviewworldgameCom 2025. https://reviewworldgame. com/blog/news/the-hidden-crisis-how-smurfing-is-undermining-competitive-integrity-in-valorant/ (accessed June 24, 2025).

[15] published NC. The ethics of smurfing. PC Gamer 2019.

[16] Conroy E, Kowal M, Toth AJ, Campbell MJ. Boosting: Rank and skill deception in esports. Entertainment Computing 2021;36:100393.