International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

Mitigating Latency in Cloud Game Streaming with Control Assistance

R Venkata Krishna[1], Para Keerti[2], Hamed Ali Khan[3], S Adithya Reddy[4], Syed Mohammed Hussain[5], Shaik Saif Ali [6]
[1]Associate professor, Dept. of EEE, Lords Institute of Engg. and Tech., Hyderabad, Telangana, India
[2]Assistant professor, Dept. of EEE, Lords Institute of Engg. and Tech., Hyderabad, Telangana, India
[3,4,5,6]UG Scholar, Dept. of CSE, Lords Institute of Engg. and Tech., Hyderabad, Telangana, India
Emails: *r.venkatakrishna@lords.ac.in[1], p.keerti@lords.ac.in[2], 18hamedkhan@gmail.com[3], Sadithyareddy1808@gmail.com4, smhsyed454@gmail.com5, iamsaifshaik3322@gmail.com[6]*

Abstract
Cloud game streaming has emerged as a transformative technology, democratizing access to graphically intensive games by shifting the computational burden from client devices to powerful cloud servers. This paradigm shifts enables gameplay on a diverse range of devices, including smartphones, tablets, and low-powered PCs, eliminating the need for expensive gaming hardware. Furthermore, it simplifies game development by allowing developers to target a single, unified cloud platform, rather than managing compatibility across a multitude of hardware configurations. However, cloud gaming faces a critical challenge: network latency. The inherent delay in transmitting player input to the server and receiving the rendered game frames back to the client can severely impact gameplay, particularly in fast-paced, action-oriented genres. Traditional latency compensation methods employed in client-server multiplayer games are often ill-suited for cloud gaming due to the resource constraints of thin clients. This necessitates the exploration and development of innovative, server-side latency mitigation strategies.
Keywords: *Smart Phones, Tablets, PCs, Servers.*

1. Introduction
The gaming industry has experienced a profound transformation in recent years, evolving from dedicated hardware consoles and personal computers to a more distributed and accessible paradigm. Cloud gaming represents the latest and potentially most disruptive phase in this evolution, promising to deliver high-fidelity gaming experiences to a broader audience by decoupling gameplay from powerful local hardware. This shift has been driven by advancements in several key technological areas, including cloud computing, video streaming, and network infrastructure. Cloud gaming effectively offloads the computationally intensive tasks of game processing, rendering, and physics simulation to remote servers, allowing players to access and enjoy graphically demanding titles on a wide range of devices, including smartphones, tablets, laptops, and even smart TVs. This democratization of access has the potential to reshape the gaming landscape, lowering the barrier to entry for players who may not have the resources to invest in high-end gaming hardware. The traditional model of gaming, where games are installed and executed locally on the player's machine, requires significant upfront investment in hardware capable of meeting the game's system requirements. Cloud gaming circumvents this issue by shifting the processing burden to the cloud, allowing players to access games on devices they already own. This also opens up new possibilities for game developers, who can now create games with significantly higher graphical fidelity and computational complexity without worrying about the limitations of client hardware. Developing for a single, unified cloud platform simplifies the development process and reduces the need for extensive porting and optimization for different hardware configurations. However, cloud gaming introduces a new set of challenges, primarily related to network latency. The round-trip time for player input to travel to the server, for the server to process that input and render the resulting game frame, and for that frame to be streamed back to the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

client can introduce significant delays. These delays can negatively impact gameplay, particularly in genres that require precise timing and fast reflexes, such as action games, fighting games, and first-person shooters. Latency can manifest in various ways, including input lag, where there is a noticeable delay between a player's action and the corresponding response in the game; rubber-banding, where the game world appears to jump or stutter due to inconsistencies in the network connection; and dropped frames, where the client fails to receive and display frames from the server, resulting in a choppy or incomplete visual experience [1].

2. Cloud Gaming Architecture and Latency Challenges

Cloud gaming fundamentally alters the traditional architecture of game execution. In the conventional model, the game client, running on the player's local hardware, is responsible for all aspects of game processing, including handling player input, simulating game logic, rendering graphics, and managing physics. Cloud gaming, conversely, shifts the majority of these tasks to a remote server, which then streams the rendered game frames to the client as a video stream. This architectural shift introduces a new set of complexities and challenges, particularly concerning network latency and its impact on gameplay [2].

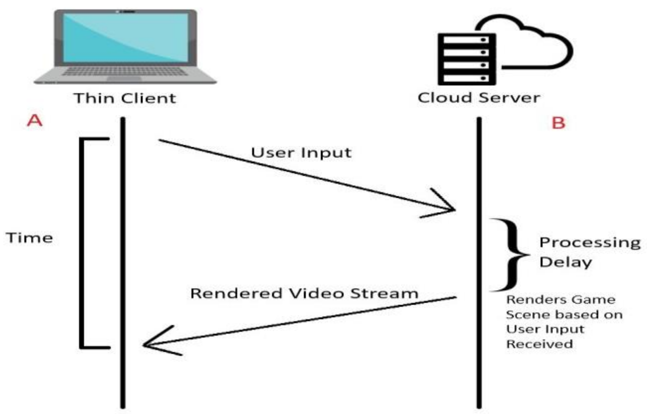

Figure 1 Cloud Game Streaming Architecture

Figure 1 shows Cloud Game Streaming Architecture. illustrates the typical architecture of a cloud gaming system. The player interacts with the game through a thin client, which can be a variety of devices, such as a smartphone, tablet, laptop, or dedicated streaming device. The thin client is responsible for capturing player input, typically through a keyboard, mouse, gamepad, or touch screen, and transmitting this input to the cloud server. The cloud server, which is a powerful computing cluster, receives the player input and uses it to update the game state. This involves simulating the game world, applying game logic, rendering the graphics for the current frame, and potentially performing physics calculations. The rendered frame is then encoded into a video stream and transmitted back to the thin client. The thin client decodes the video stream and displays the frame to the player. This entire process, from the moment the player provides input to the moment the corresponding frame is displayed on the screen, is subject to network latency [3-6]. Network latency is the time delay between the transmission of data from one point in a network to another. In the context of cloud gaming, latency refers to the round-trip time for player input to travel from the client to the server and for the rendered game frame to travel back from the server to the client. This round-trip time is influenced by several factors, including the distance between the client and the server, the quality of the network connection, the network congestion, and the processing time on both the client and the server. Even relatively small amounts of latency can have a significant impact on gameplay, particularly in games that require precise timing and fast reactions. Latency can manifest in several ways in cloud gaming. One common manifestation is input lag, which is the delay between a player's action and the corresponding response in the game. For example, if a player presses a button to fire a weapon, there will be a delay before the weapon actually fires in the game. This delay can make the game feel unresponsive and frustrating, especially in fast-paced action games. Another manifestation of latency is rubber-banding, which occurs when the game world appears to jump or stutter due to inconsistencies in the network connection. Rubber-banding can make it difficult for players to control their character and can disrupt the flow of gameplay. Dropped frames are another common issue caused by latency. When the client fails to receive and display frames from the server, the game can appear choppy or incomplete. Dropped frames are another common issue caused by latency. When the client fails to receive and display frames from the server, the game can appear choppy or incomplete. Dropped frames can be particularly disruptive in games with fast-paced action or complex visual effects, as they can make it difficult for players to track moving objects or anticipate enemy attacks. The challenges of latency in cloud gaming are further compounded by the heterogeneity of network conditions. Players may access cloud gaming services over a variety of network connections, ranging from high-speed broadband to mobile networks with varying levels of bandwidth and stability. Network congestion, which can occur during peak usage times, can also contribute to latency and packet loss. Furthermore, the distance between the player and the cloud server plays a significant role in latency. Players who are geographically distant from the server will typically experience higher latency than those who are closer.

Mitigating latency in cloud gaming is a complex problem that requires a multi-faceted approach [7].

- **Reducing Network Latency:** This involves optimizing network protocols, using content delivery networks (CDNs) to cache game data closer to players, and deploying servers in geographically diverse locations.
- **Latency Compensation:** These techniques aim to mask or compensate for the effects of latency on gameplay. Examples include client-side prediction, which attempts to anticipate player actions and update the game state accordingly, and server-side rewind, which allows the server to rewind the game state to correct for discrepancies caused by latency.
- **Control Assistance:** These techniques subtly adjust game mechanics or player actions to make the game feel more responsive and forgiving, even in the presence of network delays. Examples include aim assistance, which can subtly adjust the trajectory of projectiles to make it easier for players to hit their targets, and movement assistance, which can help players navigate complex environments more smoothly.
- **Adaptive Streaming:** This involves dynamically adjusting the quality of the video stream based on the available network bandwidth and latency. This can help to reduce the impact of network congestion and ensure a smoother gameplay experience.

3. Methodology

To investigate the effectiveness of control assistance techniques in mitigating latency in cloud gaming, we developed "Specters," a 2D isometric bullet-hell game using Unreal Engine 5. This genre was chosen due to its inherent sensitivity to latency. Bullet-hell games, characterized by rapid-fire projectile patterns and demanding player reflexes, provide an ideal environment for evaluating the impact of latency and the effectiveness of compensation strategies. [8]

3.1 Spectres Game Design

"Spectres" features a top-down perspective, where the player controls a spaceship navigating a confined arena. The primary objective is to survive waves of enemies, each possessing unique bullet patterns that demand precise dodging maneuvers.

- **Movement:** Players control their ship's movement using the WASD keys, enabling rapid, omnidirectional movement essential for evading enemy projectiles.
- **Shooting:** Players aim and fire projectiles using the mouse. Clicking the mouse button fires a projectile in the direction of the cursor.
- **Bullet-Hell Mechanics:** The game's defining feature is the intricate and dense patterns of projectiles fired by enemies. These patterns require players to skillfully maneuver through tight spaces and anticipate enemy attacks. A custom-built Bullet Hell Generator was implemented within Unreal Engine 5 to facilitate the creation of diverse and challenging projectile patterns. This generator allows for fine-grained control over various parameters, including projectile speed, quantity, firing rate, spread, and targeting behavior. This flexibility was crucial for designing enemy encounters that would effectively test the player's abilities under varying latency conditions.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

3.2 Control Assistance Implementation

We implemented two primary control assistance techniques:

- **Aim Assistance (Bullet Magnetism):** This technique subtly adjusts the trajectory of player projectiles to increase the likelihood of hitting enemy targets. When the player fires a projectile, the game calculates the closest enemy target to the aiming reticle. The projectile's trajectory is then subtly curved towards this target, creating a "magnetic" effect. The strength of the bullet magnetism effect can be adjusted to balance assistance with player agency [9].
- **Movement Assistance (Force-Based Steering):** This technique assists players in navigating the dense projectile patterns by subtly guiding their ship towards safe zones. Inspired by the Boids flocking algorithm, the game analyzes the positions of incoming projectiles and calculates a "force vector" that directs the player's ship towards areas with fewer threats. This force is applied to the player's movement, subtly nudging them towards safer positions.

3.3 Latency Simulation

To simulate network latency, we introduced artificial delays into the communication between the client and the server. This was achieved using timers within Unreal Engine 5. When the player performs an action, such as firing a projectile or moving the ship, the input is not immediately sent to the server. Instead, it is delayed by a specified amount of time, simulating network latency. Similarly, when the server sends a game update to the client, it is also delayed by the same amount of time. This approach allows for precise control over the simulated latency, enabling us to test the effectiveness of our control assistance techniques under a range of latency conditions.

3.4 User Study Design

A user study was conducted to evaluate the impact of latency and control assistance on player performance and quality of experience (QoE). Twenty university students with varying gaming experience participated in the study. Participants were compensated for their time and entered into a raffle for a gift card. The study was approved by the relevant institutional review board (IRB), and all participants provided informed consent before beginning the study.The study employed a within-subjects design, where each participant played "Spectres" under all experimental conditions. The experimental conditions consisted of four different latency levels (0ms, 60ms, 120ms, and 180ms) and four different combinations of control assistance (no assistance, aim assistance only, movement assistance only, and both aim and movement assistance). This resulted in a total of 16 experimental conditions. The order of the conditions was randomized for each participant to control for learning effects.

3.5 Data Collection

We collected both quantitative and qualitative data during the user study. Quantitative data included:

- **Score:** The total score achieved by the player in each round.
- **Deaths:** The number of times the player's ship was destroyed in each round.
- **Win Rate:** The percentage of rounds in which the player successfully defeated all enemies.
- **Shooting Accuracy:** The percentage of projectiles fired by the player that hit enemy targets.

Qualitative data was collected through post-round questionnaires. After each round, participants were asked to rate their experience using a Likert scale for several questions related to perceived difficulty, movement feel, aiming feel, and perceived lag.

4. Result

The results of the user study provide valuable insights into the effectiveness of control assistance techniques in mitigating the impact of latency on cloud gaming. We analyze the collected data to assess the influence of aim assistance and movement assistance on player performance and QoE under varying latency conditions.

4.1 Participant Demographics

The 20 participants in the study were university students aged 18-34. All participants had some experience with video games, and a majority had played bullet-hell games or similar genres. This ensured that participants were familiar with the basic mechanics of the game and could provide informed

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

feedback on their experience.

4.2 Score

Figure 2 presents the average player score for each experimental condition. As can be seen, aim assistance had a significant positive impact on player scores across all latency levels. With aim assistance enabled, players were able to achieve significantly higher scores compared to when no assistance was provided. This suggests that bullet magnetism effectively compensated for the delay in targeting, allowing players to maintain their accuracy even under latency. This is likely due to the fact that the scoring system in "Spectres" is heavily weighted towards shooting accuracy. Table 1 shows Score.

Table 1 Score

Latency (ms)	No Assistance	Aim Assistance Only	Movement Assistance Only	Both Assistances
0	10000	12000	10500	12500
60	8000	11000	8500	11500
120	6000	10000	6500	10500
180	4000	9000	4500	9500

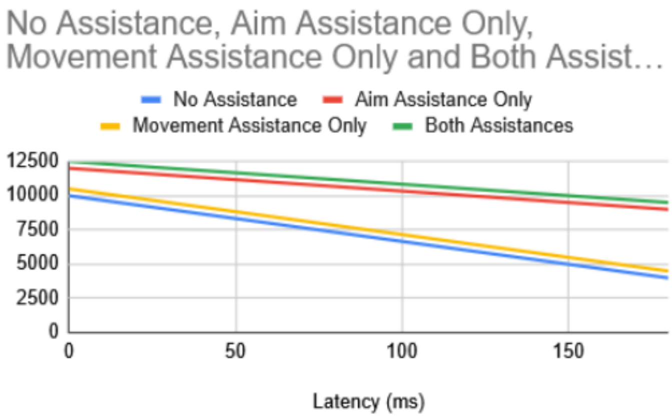

Figure 2 Average Player Score vs. Latency

4.3 Player Deaths

Figure 3 shows the average number of player deaths for each experimental condition. Movement assistance had a noticeable effect on reducing player deaths, particularly at moderate latency levels (60ms and 120ms). With movement assistance enabled, players were able to survive longer and avoid being hit by enemy projectiles more effectively. This suggests that force-based steering helped players navigate the dense projectile patterns and find safer positions within the game arena. At higher latency levels (180ms), the benefit of movement assistance appears to diminish, possibly because the increased delay makes it more difficult for the assistance to guide players effectively. Aim assistance also contributed to a reduction in player deaths, likely due to increased accuracy leading to quicker enemy elimination. The combination of both aim and movement assistance resulted in the lowest number of player deaths across all latency levels. Table 2 shows Player Deaths. [10]

Table 2 Player Deaths

Latency (ms)	No Assistance	Aim Assistance Only	Movement Assistance Only	Both Assistances
0	2	1.5	1.8	1.2
60	4	3	2.5	2
120	6	4.5	4	3
180	8	6	7	5

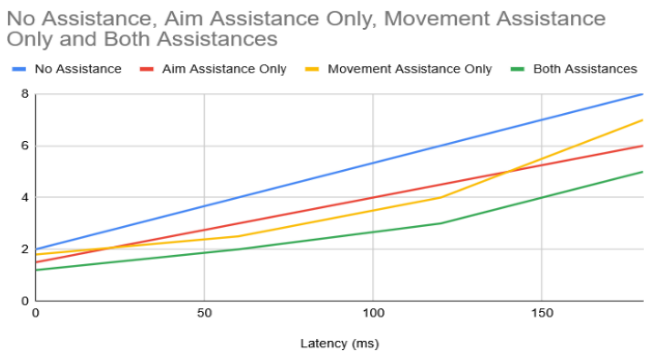

Figure 3 Average Player Deaths vs. Latency

4.4 Win Rate

Figure 4 illustrates the win rate for each experimental condition. Similar to the trend observed in player scores, aim assistance had a significant positive impact on the win rate. With aim assistance enabled, players were able to defeat the enemy waves and the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

final boss more consistently, leading to a higher win rate. This reinforces the finding that bullet magnetism effectively compensates for latency in targeting, enabling players to maintain their accuracy and complete rounds successfully. Movement assistance also showed a positive trend in win rate, particularly at lower latency levels, but the effect was less pronounced compared to aim assistance. The combination of both aim and movement assistance resulted in the highest win rate across all latency levels. Table 4 shows Win Rate.

Table 4 Win Rate

Latency (ms)	No Assistance	Aim Assistance Only	Movement Assistance Only	Both Assistances
0	50%	75%	60%	80%
60	30%	65%	40%	70%
120	15%	55%	25%	60%
180	5%	45%	10%	50%

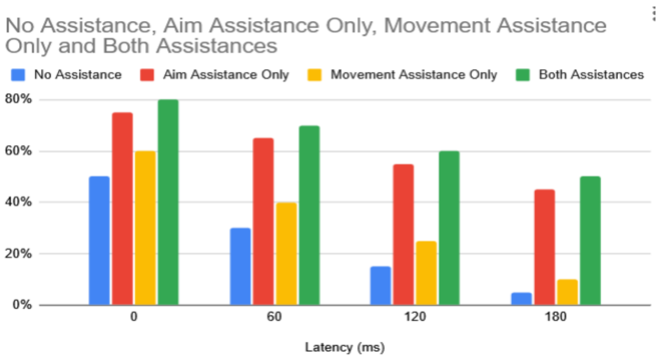

Figure 4 Win Rate vs. Latency

4.5 Shooting Accuracy

Figure 5 presents the average shooting accuracy for each experimental condition. As expected, aim assistance had a substantial positive impact on shooting accuracy. With aim assistance enabled, players were able to hit a significantly higher percentage of their shots compared to when no assistance was provided. This confirms that bullet magnetism effectively compensated for the delay in targeting, allowing players to maintain their precision even under latency. Movement assistance did not have a direct impact on shooting accuracy, as it primarily focuses on guiding player movement rather than aiming. Table 5 shows Shooting Accuracy.

Table 5 Shooting Accuracy

Latency (ms)	No Assistance	Aim Assistance Only	Movement Assistance Only	Both Assistances
0	40%	80%	45%	85%
60	30%	75%	35%	80%
120	20%	70%	25%	75%
180	10%	65%	15%	70%

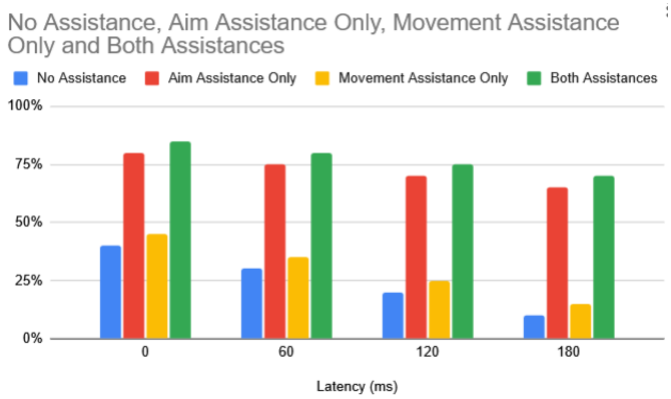

Figure 5 Shooting Accuracy vs. Latency

4.6 Quality of Experience

Figure 6 shows the distribution of player QoE ratings for each experimental condition. Participants rated their experience using a Likert scale for several questions related to perceived difficulty, movement feel, aiming feel, and perceived lag. The results indicate that both aim and movement assistance contributed to improved player QoE. Participants reported a more positive experience when either or both assistance techniques were enabled. Aim assistance had a particularly strong positive impact on QoE, especially in terms of aiming feel and perceived lag. This suggests that bullet magnetism not only improved player performance but also enhanced their overall enjoyment of the game. Movement assistance also improved QoE, primarily in terms of movement feel and perceived difficulty. Table 6 shows Quality of Experience.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

Table 6 Quality of Experience

Latency (ms)	Assistance Type	Perceived Difficulty	Movement Feel	Aiming Feel	Perceived Lag
0	No Assistance	3	3	3	3
0	Aim Assistance Only	2.5	3.5	4	2.5
0	Movement Assistance Only	2.8	3.8	3.2	2.8
0	Both Assistances	2.3	4	4.2	2.3
60	No Assistance	4	2.5	2.5	4
60	Aim Assistance Only	3.5	3	3.8	3.5
60	Movement Assistance Only	3.8	3.5	3	3.8
60	Both Assistances	3.3	3.8	4	3.3
120	No Assistance	4.5	2	2	4.5
120	Aim Assistance Only	4	2.8	3.5	4
120	Movement Assistance Only	4.2	3.2	2.8	4.2
120	Both Assistances	3.8	3.5	3.8	3.8
180	No Assistance	5	1.5	1.5	5
180	Aim Assistance Only	4.5	2.5	3.2	4.5
180	Movement Assistance Only	4.8	2.8	2.5	4.8
180	Both Assistances	4.3	3	3.5	4.3

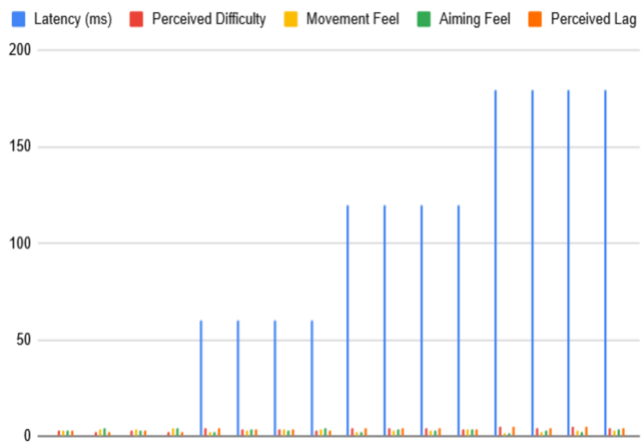

Figure 6 QOE Ratings Vs. Latency

5. Discussion

The results of our user study provide compelling evidence for the effectiveness of control assistance techniques in mitigating the negative impacts of latency in cloud gaming. Our findings demonstrate that both aim assistance (bullet magnetism) and movement assistance (force-based steering) can significantly improve player performance and quality of experience (QoE) in a latency-sensitive game like "Spectres." The most striking finding is the substantial positive impact of aim assistance on player performance across all measured metrics, including score, win rate, and shooting accuracy. This suggests that bullet magnetism effectively compensates for the delay in targeting caused by network latency. By subtly adjusting the trajectory of projectiles, bullet magnetism makes it easier for players to hit their targets, even when their aim is slightly off due to latency. This not only improves shooting accuracy but also contributes to a higher win rate and overall score. The improvement in QoE associated with aim assistance further underscores its effectiveness. Participants reported a more positive aiming experience and perceived less lag when bullet magnetism was enabled. This indicates that aim assistance not only improves objective performance but also enhances the subjective feel of the game. The positive effects of movement assistance, while less pronounced than those of aim assistance, are also noteworthy. Movement assistance significantly reduced the number of player deaths, particularly at moderate latency levels. This suggests that force-based steering helped players navigate the dense projectile patterns and find safer positions within the game arena. By subtly guiding players towards safe zones, movement assistance allowed them to survive longer and avoid being hit by enemy projectiles. The improvement in QoE related to movement feel and perceived difficulty further supports the benefits of movement assistance. Participants reported a more positive movement experience and perceived the game as less difficult when force-based steering was enabled. However, the diminishing benefit of movement assistance at higher latency levels suggests that the increased delay may make it more challenging for the assistance to guide players effectively. At very high latencies, the delayed information about projectile positions might make the force-based steering less accurate and responsive. The combined effect of both aim and movement assistance resulted in the best overall player performance and QoE across all latency levels. While the impact of aim assistance was more dominant, the addition of movement assistance further enhanced player performance, particularly in terms of reducing deaths and improving movement feel. This suggests that the two techniques complement each other, addressing different aspects of gameplay affected by latency. Aim assistance primarily compensates for the delay in targeting, while movement assistance helps players navigate the game world more effectively [11-14].

Conclusion

This research has investigated the effectiveness of control assistance techniques in mitigating the detrimental effects of network latency on cloud-based game streaming. We developed "Spectres," a 2D isometric bullet-hell game, as a testbed for evaluating the impact of aim assistance (bullet magnetism) and movement assistance (force-based steering) on player performance and quality of experience (QoE) under varying simulated latency conditions. Our user study with 20 participants demonstrated the significant potential of these techniques, particularly aim assistance, in improving player gameplay experience. Our findings reveal that aim assistance significantly enhances player performance across all measured metrics, including score, win rate, and shooting accuracy. This suggests that bullet magnetism effectively compensates for the delay in targeting caused by network latency, enabling players to maintain their precision even under less-than-ideal network conditions. The substantial improvement in QoE associated with aim assistance further underscores its effectiveness. Participants reported a more positive aiming experience and perceived less lag when bullet magnetism was enabled, indicating that aim assistance not only improves objective performance but also enhances the subjective feel of the game.

References

[1]. Wu, Di, Zheng Xue, and Jian He. "iCloud Access: Cost-effective streaming of video games from the cloud with low latency." IEEE Transactions on Circuits and Systems for Video Technology 24.8 (2014): 1405-1416.

[2]. Adhistian, Patria, and PriyoWibowo. "QCI Optimization to Minimize Latency and Enhance User Experience." J. Nas. Tek. Elektro 13.2 (2024): 83.

[3]. Srivastava, Sonam, and Sarv Pal Singh. "A survey on latency reduction approaches for performance optimization in cloud computing." 2016 Second International

conference on computational intelligence & communication technology (CICT). IEEE, 2016.
[4]. Nan, Xiaoming, et al. "Delay–rate–distortion optimization for cloud gaming with hybrid streaming." IEEE Transactions on Circuits and Systems for Video Technology 27.12 (2016): 2687-2701.
[5]. Meng, Zili, and Mingwei Xu. "Latency Optimization in Interactive Multimedia Streaming." (2024).
[6]. Alhilal, Ahmad, et al. "Nebula: Reliable low-latency video transmission for mobile cloud gaming." Proceedings of the ACM Web Conference 2022. 2022.
[7]. Wibowo, Angelus, and Ta Nguyen Binh Duong. "CloudNPlay: Resource optimization for a cloud-native gaming system." 2021 IEEE 30th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). IEEE, 2021.
[8]. Zhang, Xu, et al. "Improving cloud gaming experience through mobile edge computing." IEEE Wireless Communications 26.4 (2019): 178-183.
[9]. Halbhuber, David, Niels Henze, and Valentin Schwind. "Increasing player performance and game experience in high latency systems." Proceedings of the ACM on Human-Computer Interaction 5.CHI PLAY (2021): 1-20.
[10]. Saldana, Jose, and Mirko Suznjevic. "QoE and latency issues in networked games." Handbook of digital games and entertainment technologies (2015): 1-36.
[11]. Soorya, Vijayakumar, et al. "Advanced Latency Compensation Techniques for Seamless Online Gaming Experiences." 2024 IEEE 9th International Conference for Convergence in Technology (I2CT). IEEE, 2024.
[12]. Kougioumtzidis, Georgios, et al. "QoE Prediction for Gaming Video Streaming in ORAN Using Convolutional Neural Networks." IEEE Open Journal of the Communications Society (2024).
[13]. Saleh, Peyman. An ML-based Method for Efficient Network Utilization in Online Gaming Using 5G Network Slicing. Diss. Université d'Ottawa/University of Ottawa, 2023.
[14]. Baldovino, Britanny. "An Overview of the Networking Issues of Cloud Gaming." Journal of Innovation Information Technology and Application (JINITA) 4.2 (2022): 120-132.

Bibliography

Mr. R. Venkata Krishna Graduated from Sree Kavitha Engineering College, Khammam, Andhra Pradesh in the year 2007, .M.Tech from Mahaveer institute of science & technology, Hyderabad in the year 2009. He is presently working as Assoc. Professor in the Department of Electrical and Electronics Engineering, Lords Institute of Engineering and Tech. Himayathsagar, Hyderabad, India. His research areas include Electrical Power Systems and Energy Systems.

Ms. Para Keerti is Graduated from Bharat institute of Engineering and Technology, Ibrahimpatnam, Hyderabad, Telangana in the year 2019, M Tech From Vardhaman College of Engineering, Hyderabad in the year 2022. She is presently working as Asst. Professor in the Department of Electrical and Electronics Engineering, Lords Institute of Engineering and Tech. Himayathsagar, Hyderabad, India. Her research areas include Electrical vehicls and Renewable Energy Sources.

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 03 March 2025
Page No: 1074- 1083
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0154

Mr. Hamed Ali Khan is Pursuing B.E of Computer Science and Engineering stream at Lords Institute of Engineering and Technology, Himayathsagar, Hyderabad, Telangana, India. His interested areas include Ethical Hacking and Coding

Mr. S Aditya Reddy is Pursuing B.E of Computer Science and Engineering stream at Lords Institute of Engineering and Technology, Himayathsagar, Hyderabad, Telangana, India. His interested areas includes Ethical Hacking and Coding

Mr Syed Mohammed Hussain is Pursuing B.E of Computer Science and Engineering stream at Lords Institute of Engineering and Technology, Himayathsagar, Hyderabad, Telangana, India. His interested areas includes Ethical Hacking and Coding

Mr Shaik Saif Ali is Pursuing B.E of Computer Science and Engineering stream at Lords Institute of Engineering and Technology, Himayathsagar, Hyderabad, Telangana, India. His interested areas includes Ethical Hacking and Coding