Lower is Better? The Effects of Local Latencies on Competitive First-Person Shooter Game Players

Shengmei Liu, Mark Claypool
sliu7,claypool@wpi.edu
Worcester Polytechnic Institute, Worcester, MA, USA

Atsuo Kuwahara, James Scovell, Jamie Sherman
atsuo.kuwahara,james.j.scovell,jamie.sherman@intel.com
Intel Corporation, Hillsboro, OG, USA

ABSTRACT

Video game play is among the most popular forms of entertainment in the world and eSports is a multi-billion dollar industry. Esports gamers, and competitive gamers more broadly, want fast game systems to maximize their chances of winning. In general, the faster the game system, the lower the latency between a player's action and the intended outcome. But how much small reductions in local latencies benefit competitive players is not known. This paper presents results from a 43-person user study that evaluates the impact of system latencies for high-end gaming systems (below 125 ms) on experienced Counter-strike: Global Offensive (CS:GO) players. Analysis of the results show pronounced benefits to CS:GO player performance (accuracy and score) for even small reductions in latency, with subjective opinions on Quality of Experience following suit.

CCS CONCEPTS

• **Applied computing** → **Computer games**; • **Human-centered computing** → *User studies*.

KEYWORDS

skill, gamer, FPS, user study, eSports, lag

1 INTRODUCTION

Computer games are one of the world's most popular forms of entertainment, with global sales increasing at an annual rate of 10% or more [38]. The largest esport prize pools are about $25 million USD [14], larger even than traditional sports which range from about $2 to $20 million USD [32]. By 2023, there are expected to be about 300 million frequent viewers of esports worldwide, an increase from 173 million in 2018 [17].

Latency – delay between a player's input and the game responding with audio or visual output – makes games less responsive, degrading player performance and hurting the quality of experience. To reduce latencies, serious gamers typically want fast computers with upgraded processing, memory and graphics capabilities. This includes specialized computer mice that reduce input latencies with frequent polling and high-performance monitors that reduce output latencies by minimizing buffering and increasing frame rates, all designed to speed up the responsiveness of the system.

Despite the conventional wisdom that "faster is better", the degree to which the reduced latencies from better game systems benefit competitive game players is not well-known. There have been numerous studies on latency and network games [15, 21, 33], especially latency and first-person shooter (FPS) games [1, 2, 22, 34], owing to the sensitivity of FPS games to network latency and the prevalence of FPS games in the competitive and esports scenes. However, studies of network latency and games, particularly for commercial games, include confounding latency compensation techniques [3] that try to mitigate the effects of network latency, often by modifying how the player experience the game. Moreover, the latency experienced by the player for online games includes the network latency in addition to the local system latency, and therefore does not necessarily represent the effects of just the local latency on the game system.

There have also been many studies on latency and input actions and devices [24, 29]. While studies such as these have been helpful for better understanding the effects of latency on basic user input [35] and even fundamental game actions [28], the degree to which those effects translate to game performance is not well known. In particular, it is known that high-skilled game players perceive and adapt to latency differently than low-skilled game players [1, 8], but whether this translates to the same perception and adaptation for competitive gamers is not known.

In general, there have not been studies of local latencies on complete games. This means little is known of the effects of latencies at the low-end of the latency spectrum – less than 100 milliseconds – common latencies incurred by many local computing devices. Of particular interest are competitive game players, since they have more interest in high-performance game systems and the low latencies they bring, and first-person shooter games given their prevalence in esports.

This paper presents the results from a user study that measures the impact of latency on experienced first-person shooter game players. The user study was designed and conducted to adhere to constraints imposed by the COVID-19 pandemic – bringing carefully configured and instrumented gaming laptops to the homes of users in a contact-less fashion, instead of having players come into a lab as normally happens in a user study. Users were screened for

their skill at the esports FPS game *Counter-strike: Global Offensive (CS:GO)* (Valve, 2012), obtaining a pool of 43 qualified participants.

Analysis of the results shows 10 ms of latency reduction results in 0.8 percent improvement to accuracy and 1.2 additional points to score. The latter translates to about an extra kill per game – significant since matches are decided by the first team to win 16 rounds out of 30, and that a single kill can create huge advantages in a game round. In addition, this same latency reduction results in an improvement in subjective opinion of the game (frustration, annoyance, responsiveness) by about 5 percent. This increase is despite the fact that participants anecdotally said that they often could not really tell the difference in latency between many of the rounds.

The rest of this paper is organized as follows: Section 2 describes previous work on latency and games related to this paper; Section 3 describes our methodology, including CS:GO configuration and user study design and execution; Section 4 analyzes the results from the user study; Section 5 mentions some limitations of our user study; and Section 6 summarizes our conclusions and possible future work.

2 RELATED WORK

This section provides references to related work with brief descriptions of findings in several relevant areas: Counter-strike: Global Offensive, Configurations, Network Latency, Cloud-based Games, and Local Latency.

2.1 Counter-strike: Global Offensive

The longevity of *Counter-strike Global Offensive (CS:GO)* (Valve, 2012) and its prevalence in competitive gaming has motivated CS:GO use and focus in related research. Past studies have included both the social aspects of CS:GO [16, 25] and the prevalence of CS:GO in the esports scene [30, 31].

Frostling-Henningsson [16] show young, online Counter-strike players are foremost motivated by social reasons, with online games providing opportunities for cooperation and communication that single player games do not. Jansz and Tanis [25] have complementary findings in that social interaction motives are the strongest predictor of the time spent gaming, even for gamers that are also motivated by competition and challenge.

Lux et al. [30] describe how automatically summarizing CS:GO matches is a challenge, but that approaches focusing on events around opponent kills are promising. Makarov et al. [31] find ranking CS:GO players based on their impact to team victories is useful for predicting winning and losing.

While helpful to better understand CS:GO players and their interactions and performance in the game, these papers do not delve into the effects of latency on competitive CS:GO players, as does the work in our paper.

2.2 Configurations

Previous studies have analyzed the effects of system and game configurations on player performance and Quality of Experience (QoE), generally focusing on frame rate and resolution as independent variables in their analysis [5, 7, 40].

Spjut et al. [40] show a reduction in 30 ms of latency benefits first-person targeting tasks more than frame rates above 60 f/s. Claypool and Claypool [5] show that player actions that require precise, rapid response, such as shooting, are greatly impacted by degradation in frame rates below 30 f/s for a First-Person Shooter game. As a comparison, Claypool et al. [7] find frame rate has a marked impact on both player performance and game enjoyment while frame resolution has little impact on performance and some impact on enjoyment for a First-Person shooter game.

While helpful for ascertaining the impact of system level configurations on game players, these papers do not specifically deal with latency and/or do not deal with competitive game players.

2.3 Network Latency

Numerous studies have detailed the effects of network latency and games [1, 2, 6, 12, 15, 21, 22, 33, 34]. Most of these studies utilize commercial games with controlled amounts of network latency in a laboratory environment, similar to our approach in this paper, rather than observing players during normal play.

Dick et al. [12] show via a survey that players generally think about 120 ms is the maximum tolerable latency for a network game, regardless of game genre, but their user study shows players find 150 ms acceptable for the two First-Person Shooter games and racing game tested. Amin et al [1] show player experience defines and determines the sensitivity to latency for the First-Person Shooter game Call of Duty, with competitive gamers more adept at compensating for impaired conditions. Armitage et al. [2] estimate the latency tolerance threshold for the First-Person Shooter game Quake 3 to be about 150-180 ms. Pantel and Wolf [33] show latencies of about 100 ms can affect car racing games. Fritsch et al. [15] find players of the role-playing game Everquest 2 can tolerate hundreds of milliseconds of network latency, while Hoßfeld et al. [21] find players of the casual game Minecraft are insensitive to network latencies of up to 1 second. Howard et al. [22] indicate that for online cooperative games, a player can be affected by latency for a teammate due to cascading effects on the game outcome. Quax et al. [34] show players for the First-Person Shooter game UT2003 that latency and latency jitter under 100 ms can degrade player performance and quality of experience. Based on this body of work, Claypool and Claypool [6] suggest game action sensitivity to latency can be classified by precision and deadline – higher precision and tighter deadlines mean more sensitivity to latency.

However, while such works have been instrumental in better understanding the effects of network latency on players of online games, they do not necessarily represent the effects of local latencies on player performance since they combine local latency with network latency. As such, they typically deal with a much higher range of latencies than does our paper. Moreover, most network games have latency compensation techniques that help mitigate the effects of network latency [3]. In contrast, local latencies, such as those studied in this paper, cannot be overcome by traditional latency compensation techniques.

2.4 Cloud-based Games

Similar studies of latency have been done in cloud-based game streaming systems, where all player input is delayed by the round-trip latency from the client to the server [4, 9, 10, 39].

Claypool and Finkel [9] find both quality of experience and player performance degrade linearly with an increase in latency for Crazy Taxi (a driving game) and Neverball (an arcade game) streamed from the cloud. Clincy and Wilgor [10] show network latency and packet loss can significantly degrade subjective quality of experience for the First-Person Shooter game Borderlands played on the OnLive cloud-based game system. Sabet et al. [39] identify game characteristics that may be helpful for designers in determining if a cloud-based game is sensitive to latency or not.

As with the local latency that is the focus of the study in our paper, cloud-based game systems often cannot apply the latency compensation techniques used in traditional network games. That said, due to the generally higher latency in a cloud-based game streaming network, the extra workloads required for streaming video, and the remote processing demands, these works typically study much larger latencies than the low-end local latencies that are the focus of our paper.

2.5 Local Latency

There are several works that have characterized just the effects of local latency on game player performance [8, 24, 28, 29, 37] – so, not including the effects of network latency – in ways that are similar to the study in our paper. These papers have generally focused on a single player action, isolating and analyzing the effects of latency without the broader set of interactions of a typical game.

Claypool et al. [8] show local latency and target speed exponentially impact target selection time. Ivkovic et al. [24] find significant main effects for local latency on target tracking and acquisition tasks, both with and without latency compensation (aim assistance), and with a greater effect for higher target speeds. Long and Gutwin [28] find target speed directly affects target acquisition with latency, with fast targets affected by latencies as low as 50 ms but slower targets resilient to latencies as high as 150 ms. Eg et al. [37] show local latencies from 40 ms to 400 ms negatively affect player performance for moving target selection, but that performance does not co-vary with self-reported game skill. Long and Gutwin [29] compare the effects of local latency across 4 different gaming devices, demonstrating that latency affects each device differently for moving target selection.

While useful for understanding and even modeling the effects of latencies, in some cases even low-end local latencies, on player actions, these works do not necessarily represent player performance in an actual game where typically a player does more than one action, and must play under competitive game conditions. In contrast, our work studies play in a complete, commercial game (CS:GO) where users do multiple player actions (e.g., moving, shooting), often simultaneously, while competing against opponents.

3 METHODOLOGY

To investigate how low amounts of system latency affect competitive First-Person Shooter (FPS) gamers, the following methodology

was deployed: 1) Find a competitive FPS game that provides gameplay data in the form of log files and allows for customization of game settings; 2) Measure base system latency for our game system to get the minimum latency values; 3) Design and conduct a user study with the customized game and added delays to evaluate the impact of system latency on competitive FPS gamers; and 4) Analyze the user study results in terms of player performance and quality of experience with latency.

The COVID-19 pandemic created many challenges for conducting user studies. In particular, the traditional user study, where participants are brought into a laboratory environment with controlled experimental conditions, was impossible with campuses closed and social distancing rules in place. To overcome this challenge, we designed a user study that, instead of bringing participants to the lab, brought the lab to the participants. The study was reviewed and approved by our University Institute Review Board (IRB).

A high-end gaming laptop and low latency gaming mouse were selected and benchmarked together in order to assess "best case" latency conditions while allowing room for emulating slightly higher latencies by adding delay to user input. The selected laptop was the Gigabyte Aero 15, accompanied by a Logitech G502 mouse. The Aero has an 8-core i7 9750H / 2.6 GHz processor, 16 GB RAM, an NVidia GF RTX 2070 graphics card, and a screen resolution of 1920x1080 pixels at 240 Hz. The G502 is a laser mouse with 12k DPI, 300 IPS, and a polling rate of 1 kHz. The laptop was configured with Ubuntu 20.04 LTS, with Linux kernel version 5.4. For this study, we assembled a set of 11 identical laptops and mice, all configured exactly the same way.

To provide for accurate assessment of the latencies user experienced in the study described here, the base system latency was measured on the test system. The measurement method is depicted in Figure 2. A high-frame rate camera (a Casio EX-ZR100) was setup completely external to the game system and filmed a user at 1000 f/s, capturing the moment the mouse button was clicked. By manually examining the video frames, the frame number when the mouse was clicked (finger bent, frame number 214 in Figure 2) is subtracted from the frame number when the output was visible based on the user input click (frame number 239 in Figure 2), giving the base system latency (25 milliseconds in Figure 2). The measurement method was repeated 10 times on our system, resulting in an average base latency of 25.2 milliseconds (ms), with a standard deviation of 2.8 milliseconds. Note, these base latency measurements are the lowest of 5 gaming systems we tested, and are also the lowest in the range of local latencies that others have tested [24, 36]. Hereafter in this paper, 25 ms is added to all latency analysis, unless otherwise specified.

In order to test the effects of latencies above the baseline 25 ms, additional latency was added to all keyboard and mouse user input with a custom C program we wrote called *EvLag*. EvLag is a standalone executable that adds a constant amount of latency to any input device in Linux using evdev, interfaces that generalize raw input events from device drivers as character devices. EvLag accesses the devices via libevdev, a user-space library that abstracts I/O calls through a type-safe interface. When enabled, EvLag intercepts and enqueues all input events from a selected device and, after the specified delay, dequeues and delivers the events. Timing in EvLag is maintained via the real-time clock drivers for Linux, accessed
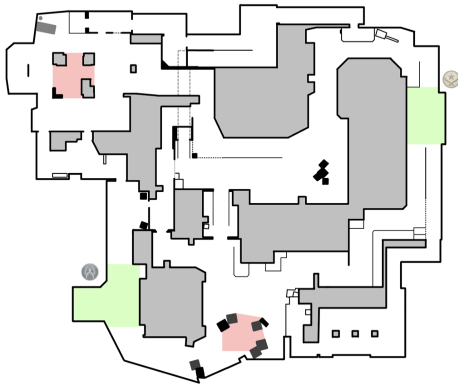
Figure 1: User study CS:GO map – Mirage

Figure 2: Measuring local latency

through /dev/rtc, giving fine-grained time resolution (less than a millisecond) for control.

The added latencies for EvLag in our user study were 0, 25, 50, 75, 100 ms. So, with the 25 ms base latency, the resulting total system latencies experienced by our users are shown in Table 1.

Table 1: Total latencies for the user study.

Latency values (ms)				
25	50	75	100	125

After considering a number of possible games for testing, *Counter-strike Global Offensive (CS:GO)* (Valve, 2012) was selected as it is: a) used for esports, and b) has a large player base, but c) allows options for single-player play in stand-alone mode without an Internet connection so as to remove the confounding effect of uncontrollable network latencies from the study. The selected CS:GO options were invoked via configuration files launched from the command line – this meant when starting CS:GO in the study, users immediately launched into the game, bypassing normal game lobbies and weapon selection phases.

While CS:GO matches often include team strategy, the focus of this study is on the effects of latency on individual player tactics. As such, a death match free-for-all game mode (no teams) was chosen. Thus, each round had open combat for the user and 20 AI-controlled bots, where everyone fights everyone and the goal was to kill as many opponents as possible. CSGO has 4 bot options: 0 - easy, 1 - normal, 2 - hard, and 3 - expert. The bots in our study are at the third (of four) difficulty level, 2 - hard. While it is likely that the absolute scores would indeed differ for users pitted against human players, the relative effects should be similar since the latency affects the ability to aim and shoot (thus, score and accuracy). The AI-controlled avatars move with the same game physics as do human-controlled avatars, with the primary difference aiming accuracy and firing speed, impacting player deaths only, not player accuracy nor score (kills, assists).

There was no upper limit on player score – the game terminated after a fixed 4 minutes of time.

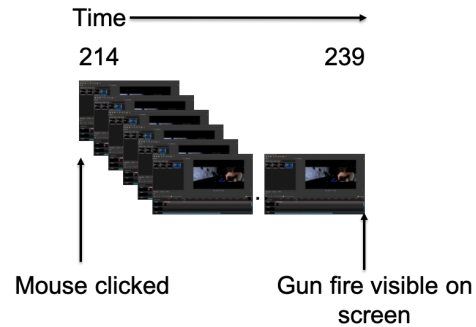The size of CS:GO game maps range from the smallest map "Train" ($5.4\ kHu^2$) to the largest map "Subzero" ($9.7\ kHu^2$), where $1\ kHu^2$ equals about $0.02\ km^2$. A small map was desired to limit the need for the player to explore and wander and to maximize combat. The map chosen, "Mirage", depicted in Figure 1, is the most popular [19] and third smallest map ($5.9\ kHu^2$) [42]. The user and the bots spawned at random locations on the map that were not currently in view of anyone else.

The laptops were configured to start up and launch the user study sessions with minimal instructions so as to reduce user error. The laptops automatically booted to the test user account upon powering on and, effectively, double-clicking a custom test harness script labeled "Play" on the desktop launched each session. Once launched, the test harness: 1) started EvLag with the appropriate latency, 2) started CS:GO with the pre-determined map, weapon, bots and game mode, 3) allowed the game to run for 4 minutes, 4) stopped the game, 5) launched a quality of experience survey, 6) gathered and archived all game logs, input logs and survey results, and 7) repeated the process for each weapon and latency condition. The script provided for robust error checking in the event of user error (e.g., closing the game early) or system malfunction (e.g., software crashing).

The software configuration was replicated across 11 identical laptops. These laptops were hand-delivered to each user in a contactless fashion, enabling users to participate in the study from their homes. A pickup time was arranged a few days after drop-off. Upon pickup, each laptop and mouse was carefully sanitized, the study data extracted, and the laptop reset for the next participant.

A user study proctor was available (by messaging and phone) for questions and trouble-shooting at all hours of the day during the study.

Before any sessions started, users first completed a reaction-time test written in Javascript and launched via a Chrome Web browser on the laptop. In the test, users waited for a screen color change then clicked the mouse as quickly as possible. Users did this 10 times (the total test time was about 30 seconds) before starting the CS:GO game sessions.

Users then played 5 sessions of CS:GO, each session being 5 rounds, each round having a different total latency selected from Table 1, randomly shuffled.

Table 2: Weapon attributes

Weapon	Firing mode	Firing rate	Clip size	Reload time	Damage	Accuracy range
AK-47	Automatic	600 per min	30	2.43 s	36	21.74 m
AWP	Sniper	41 per min	1	3.7 s	115	69 m

Table 3: Subjective questions per round

	Question	Source
Q1	I was frustrated by the round	GEQ [23]
Q2	The delayed reactions of the round annoyed me	TLX [18]
Q3	How well I did was completely due to me	Attribution [11]
Q4	Rate the responsiveness of the round	Survey [37]

In order to assess the effects of latency for weapons with different skill requirements (e.g., high precision versus low precision), users were equipped with only one of two weapons at a time: A) the AK-47 (the most popular automatic rifle), or B) the AWP (the most popular sniper rifle) [20]. Detailed specifications on the two weapons are shown in Table 2. Both weapons had unlimited ammunition, but still needed to be reloaded when their magazine clips were empty (after 30 bullets for the automatic rifle and after each bullet for the sniper rifle). Since the firing rate for the sniper rifle is lower than that of the automatic rifle, we had 3 sniper sessions as opposed to the 2 automatic rifle sessions to get more action data (shots fired) for the sniper. The first 2 sessions were with the automatic rifle as the only weapon and the final 3 sessions were with the sniper rifle as the only weapon.

After each round, users were presented with a subjective survey consisting of four questions on a 5-point, discrete Likert scale about the game experience in the preceding round. The questions are shown in Table 3.

After completing the survey, the next round would commence. However, users could take as much time as they desired before starting the following round, and users were encouraged to take at least an hour break between sessions to avoid fatigue.

To allow users to get familiar with the map and the weapon, the first session with each weapon started with an additional practice round without any induced latency. Data from the practice round was discarded.

Before the launch of the formal user study, two pilot studies were conducted with volunteers (first 3 and then 4 friends/family, with 4 of them experienced CS:GO players) in order to test the viability of the procedure and tune the study settings. The pilot study results helped adjust weapon choices, latency values, number of rounds, number of sessions, system settings and user instructions.

Study participants were solicited via University email lists. Interested participants first filled out a screener questionnaire to provide basic demographic information (e.g., age and gender), but also for screening for appropriate CS:GO experience. Qualification criteria included: 1) extensive experience playing games on a PC, 2) prior experience playing CS:GO, 3) at least 100 hours playing CS:GO and/or a high self-rating in FPS games, and 4) residence within 40 miles (65 km) of our university (to allow us to hand-deliver a laptop). Users were rewarded with a $75 USD Visa gift card upon completion of the study.

Forty-three (43) users were selected to participate in the user study out of 248 initial responses.

After returning the laptops, users were given a demographics questionnaire with additional questions about average time spent playing games and self-rated expertise with different CS:GO weapons.

In summary, the procedure each user followed was:

(1) Fill out the screener questionnaire to ensure sufficient CS:GO experience.
(2) Receive a pre-configured laptop and instructions regarding setup and game controls.
(3) When ready to start, setup the laptop on a desk, plug in the power supply, connect the external mouse and place it on the included mouse pad for use by whichever hand is preferred. Plug in external headphones, if those are preferred over the laptop speakers.
(4) Adjust the computer chair height and laptop angle/tilt so as to be comfortably looking at the center of the screen.
(5) After powering on the computer, start the study by double-clicking on the "Play" icon on the desktop.
(6) Complete the reaction-time test (takes about 30 seconds).
(7) Complete the first session (1 practice round and 5 rounds with shuffled latencies), including the QoE surveys after each round (each session takes about 25 minutes).
(8) After a break of at least an hour, repeat the previous step for the remaining 4 sessions.
(9) Return the laptop and receive remuneration.
(10) Complete final demographics questionnaire online.

4 ANALYSIS

This section first summarizes participant demographics (Section 4.1) then presents the core results – user performance (Section 4.3) and Quality of Experience (Section 4.4) in the presence of local latency. Additional analysis examines user actions by weapon type in the presence of local latency (Section 4.5) and comparative performance by weapon type (Section 4.6).

Table 4: Demographic information

Users	Age (yrs)	Gender	FPS Self-rating	CS:GO Self-rating	CS:GO Hours	Reaction-time (ms)
43	21.1 (5.0)	42 ♂ 1 ♀	4.5 (0.7)	4.2 (0.8)	664 (827)	217.2 (59.5)
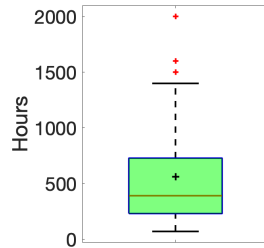
Figure 3: CS:GO hours played
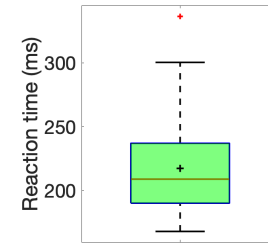
Figure 4: Reaction times

4.1 Demographics

Table 4 summarizes the demographic information for the user study participants. FPS self-rating and CS:GO self-rating are on a five-point scale, 1-low to 5-high. For age, FPS self-rating, CS:GO self-rating, CS:GO hours played, and reaction times, the mean values are given with standard deviations in parentheses. Our user study had 43 participants, ranging from 12-45 years old but with the large majority of typical college-age. Gender breakdown is predominantly male (42 males versus 1 female). We were slightly disappointed by the low number of female participants, but it should be noted that this is not atypical of esports players (about 5% are women) and reflects the gender breakdown of FPS game players, specifically [41]. User self-rating as FPS and CS:GO gamers both skew towards "high" (mean 4.5 and 4.2 out of 5, respectively). Half of the users played 10 or more hours of computer games per week and most users majored in Robotics Engineering, Computer Science, or Game Development (not shown in the table).

Figure 3 depicts the distribution of users' hours playing CS:GO, and Figure 4 depicts the distribution of users' reaction times as box-plots. Each box depicts quartiles and median for the distribution. Points higher or lower than 1.4 × the inter-quartile range are out-liers, shown by red pluses. The whiskers span from the minimum non-outlier to the maximum non-outlier. The black pluses shows the mean values. Users played from 70-5000 hours of CS:GO, with a mean of 664 hours and a large standard deviation of 827. Reaction times are mostly fast (just over 200 ms), typical of experienced computer game players [13].

4.2 Data Cleaning

For the reaction times, out of the 430 reaction time trials, 3 were extremely long – over 700 milliseconds – perhaps because the user's attention wandered. These three trials are removed from the reaction time analysis (and from Figure 4).

For the game data, out of 1075 rounds, 17 had mouse or key-board log files that were considerably shorter than the 4 minute round time, possibly because the user stopped playing or a program crashed. The game data and the QoE data from these 17 rounds are removed for analysis.

4.3 Player Performance

We measure user performance by effectiveness with each weapon: accuracy (shots hit divided by shots fired) and score (CS:GO computes score as $score = 2 \times kills + assists$). The CS:GO log files are mined to determine number of hits, kills and assists by each user for each round, and the EvLag log files are used to determine the number of shots fired based on the number of left mouse-button clicks.

Table 5 shows performance results averaged over all users and all game rounds, broken down by rifle type: automatic and sniper. The table has mean values, with standard deviations in parentheses.

Table 5: Performance summary

Weapon	Shots fired		Accuracy (%)		Score	
Automatic	385.0	(86.4)	17.8	(4.8)	45.8	(12.6)
Sniper	50.8	(13.0)	56.9	(11.8)	55.9	(14.9)

Section 4.3.1 analyzes weapon accuracy versus latency and Section 4.3.2 analyzes player score versus latency.

4.3.1 Weapon Accuracy. Figure 5 depicts weapon accuracy versus latency for the automatic rifle (the AK-47). The x axis is total system latency in milliseconds. The right y axis is the weapon accuracy (percent) and the left y axis is the percent increase from the 125 ms latency condition. For example, an accuracy of 20 percent at 25 ms of latency compared to an accuracy of 15 percent at 125 ms of latency would be a 5 percent improvement on the left y axis. The circles are the means for all users for that weapon and latency condition, bounded by 95% confidence intervals. The dashed line shows a linear regression for the mean values. The regression fits
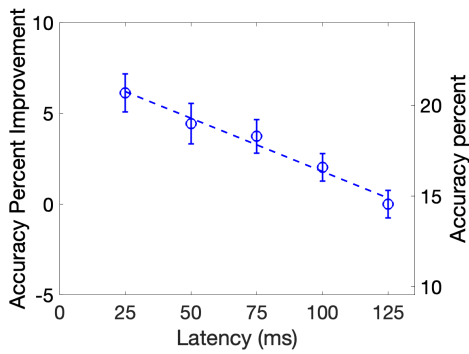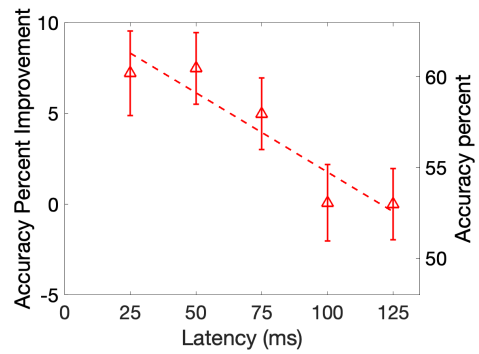
Figure 5: Accuracy – Automatic
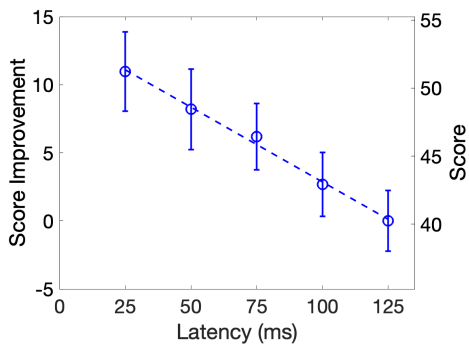

Figure 6: Accuracy – Sniper
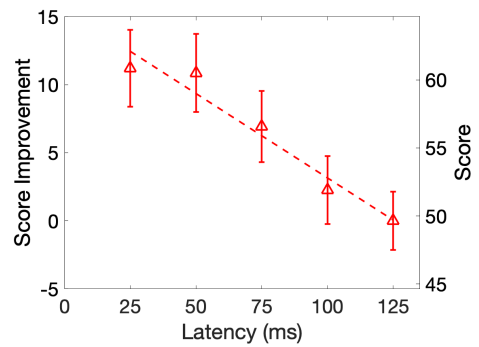

Figure 7: Score – Automatic
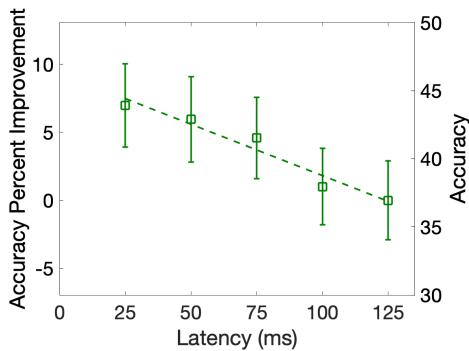

Figure 8: Score – Sniper
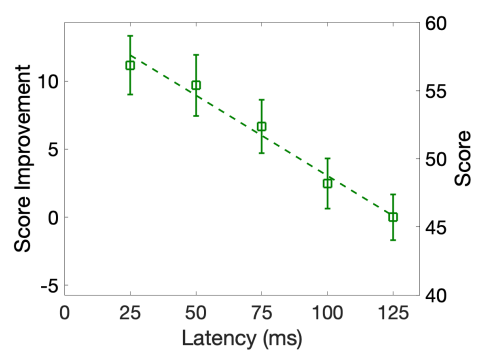

Figure 9: Accuracy – Combined

Figure 10: Score – Combined

the mean values well, with an R^2 of 0.98 and $p = 0.001$. As a take-away, for an automatic rifle, a decrease in latency by 10 ms improves accuracy by 0.6 percent on average.

Figure 6 depicts the same data as in Figure 5, but for the sniper rifle. The linear regression also fits the mean values well with an R^2 of 0.87 and $p = 0.02$. As a take-away, for a sniper rifle, a decrease in latency by 10 ms improves accuracy by 0.9 percent. However, from the figure, the sniper rifle accuracy trend may not follow this same linear trend from 50 ms to 25 ms latency and from 125 ms to 100 ms latency.

Considering the slopes of the regression lines in both Figure 5 and Figure 6, latency has a slightly larger effect on accuracy for sniper rifles than for automatic rifles.

4.3.2 Player Score. Figure 7 depicts player score versus latency for the automatic rifle. The axes and points are as in Figure 5, but the data is the CS:GO score ($2 \times kills + assists$) instead of accuracy. The liner regression fits the mean values well, with an R^2 of 0.99 and $p < .001$. As a take-away, a decrease in latency by 10 ms improves player score by 1.1 points per 4 minutes of gameplay. For reference,
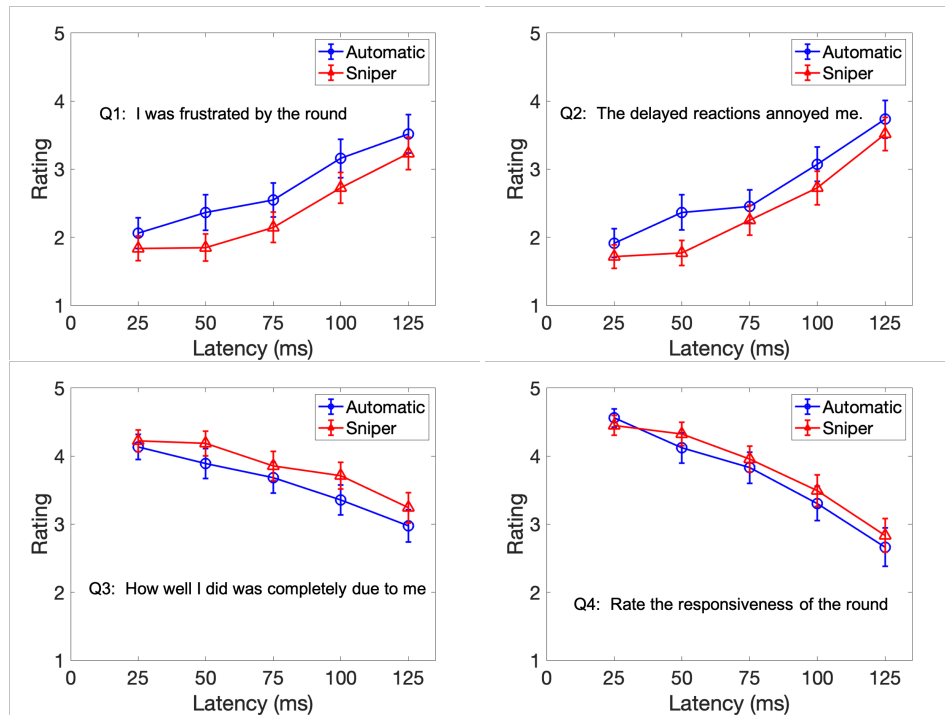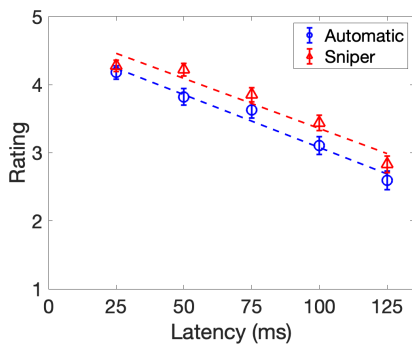
Figure 11: QoE – separate questions

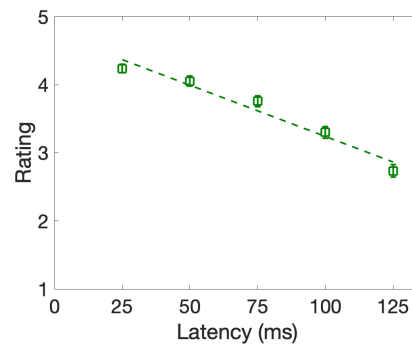**Figure 12: QoE – combined questions**

Figure 13: QoE – combined questions and weapons

often less than a single point separates the scores of top CS:GO players.

Figure 8 depicts the same data as Figure 7, but for the sniper rifle. The linear regression fits the mean values well, here, too, with an R^2 of 0.95 and $p = 0.005$. As a take-away, a decrease in latency by 10 ms improves player score by 1.2 points per 4 minutes of game play.

Considering the regression lines in both Figure 7 and Figure 8, latency has a similar impact on score for both sniper rifles and automatic rifles, with slightly more impact on the former.

To study how latency affects performance overall, the combined automatic rifle and sniper rifle data was analyzed. Figure 9 depicts the results for accuracy, with axes and data as for Figures 5 and 6. The linear regression fits the mean values for the combined data

well with an R^2 of 0.95 and $p = 0.005$. As a take-away, a decrease in latency by 10 ms improves overall accuracy by 0.8 percent. Figure 10 depicts the results for score, with axes and data as for Figures 7 and 8. The linear regression fits the mean combined score values well with an R^2 of 0.98 and $p = 0.001$. As a take-away, a decrease in latency by 10 ms improves score by 1.2 points per 4 minutes of gameplay.

Tables 6 and 7 summarize the results in tabular form, providing the slope, y-intercept, adjusted coefficient of determination (R^2) and statistical significance (*pvalue*).

Table 6: Analysis Summary – Accuracy

Weapon	Slope	y-intercept	R^2	P value
Automatic	-0.06	22.23	0.98	0.001
Sniper	-0.09	63.49	0.87	0.02
Combined	-0.08	46.32	0.95	0.005

Table 7: Analysis Summary – Score

Weapon	Slope	y-intercept	R^2	P value
Automatic	-0.11	54.09	0.99	< 0.001
Sniper	-0.12	65.20	0.95	0.005
Combined	-0.12	60.57	0.98	0.001

4.4 Quality of Experience

Quality of Experience (QoE) was assessed from user responses to 4 survey questions filled out at the end of each round. Responses are on a 5 point scale, from 1-low to 5-high.

Figure 11 depicts ratings for each question versus latency. The x axis is total system latency in milliseconds and the y axis is the rating. The circles are the means for all users for that weapon and latency condition, bounded by 95% confidence intervals. The blue circles, bars and lines are for the automatic rifle and the red triangles, bars and lines are for the sniper rifle. The top left graph is for question 1: "I was frustrated by the round", the top right graph is for question 2: "The delayed reactions of the round annoyed me", the bottom left graph is for question 3: "How well I did was completely due to me", and the bottom right graph is for question 4: "Please rate the responsiveness of the round." For questions 1 and 2, lower is better and for questions 3 and 4, higher is better. In general, mean user perceptions get worse with latency, roughly the same for both weapons. Mean values for user perceptions when using the automatic rifle are slightly worse than those for the sniper rifle, but most confidence intervals overlap for the same latency values.

For an overall measure of QoE, we flip the ratings of question 1 and 2 (for example, a rating at 5 would be converted to 1, a 4 would be a 2, etc.), and compute an overall average (i.e., all questions are weighted equally) – here, higher is better.

Figure 12 depicts the results, with axes as for the graphs in Figure 10. The circles are mean values for all users across all latency conditions, shown with 95% confidence intervals. The dashed lines are linear regression fits through the automatic rifle (blue) and sniper rifle (red), separately. Figure 13 shows the same data, but combines the automatic and rifle data. The linear regressions fit the means well in all cases, with R^2 values of 0.97, 0.93 and 0.96 for automatic, sniper and combined, respectively. All values are statistically significant ($p = 0.002$, $p = 0.008$, and $p = 0.004$). However, the sniper rifle QoE values may not follow the same linear trend from 25 to 50 ms latency, similar to sniper accuracy values (Figure 6). As a take-away, a decrease in latency by 10 ms improves QoE by 0.15 points out of 5.

4.5 Play Style

In addition to performance and perception, latency may impact how a player interacts with the game. For CS:GO, this may manifest in a different firing rate or different ratio of avatar movement to shooting.

Figure 14 and Figure 15 depict shots fired per minute versus latency for the automatic rifle and the sniper rifle, respectively. The x axes are the total latency and the y axes are shots per minute – the right y axis is the number of shots per minute and the left y axis is the shot per minute increase over the 125 ms condition. Points are mean values across all users for that latency condition and weapon, shown with 95% confidence intervals. From Figure 14, users generally fire automatic rifles more often for higher latencies, possibly trying to compensate for the decreased responsiveness. However, from Figure 15, the reverse is true for sniper rifles as users fire less often for higher latencies, possible because it is more difficult to align the gun reticle with the target before pulling the trigger.

Figure 16 and Figure 17 depict avatar movement per minute versus latency for the automatic rifle and sniper rifle, respectively. Movement is computed from the number of times the 'w', 'a', 's', and 'd' keys on the keyboard are pressed, divided by the length of the round (4 minutes). The axes are as for the graph in Figure 14. In the case of movement, users with both types of rifles move less often with higher latencies, possibly because the lower responsiveness requires more deliberate movement actions by the players.

4.6 Comparative Rifle Performance

Lastly, this section provides a brief analysis of user performance with each weapon type to asses whether good players are skilled with both weapons or if, instead, players tend to specialize and be better at one weapon versus the other. In order to compare performance across latency conditions and weapons, we normalize the data by computing the overall mean for each weapon across all conditions and dividing a user's mean by this total. So, a user with a value of 1.2 is 20% better than average with that weapon and a user with a 0.5 is 50% worse than average.

Figure 18 and Figure 19 depict scatter plots of the results – sniper rifle versus automatic rifle – for accuracy and score, respectively. Each point is the normalized mean value for one user across all latency conditions. From Figure 18, there are few visual patterns for accuracy with the sniper rifle and accuracy with the automatic rifle ($R = 0.21$). However, from Figure 19, there is a visual correlation in score for sniper rifles versus score for automatic rifles ($R = 0.86$), with a lack of points in the second and fourth quadrants. These graphs suggest specialization may show up in differences in accuracy, but when it comes to score, players that are good with one weapon are good with another, and vice-versa, regardless of specialization.

5 LIMITATIONS

While our methodology described in Section 3 is designed to minimize the differences in the test conditions across participants (e.g., identical laptops, no network connection), the home environment where each user played was not controlled. Users were asked to choose a place with a desk where they could play each session
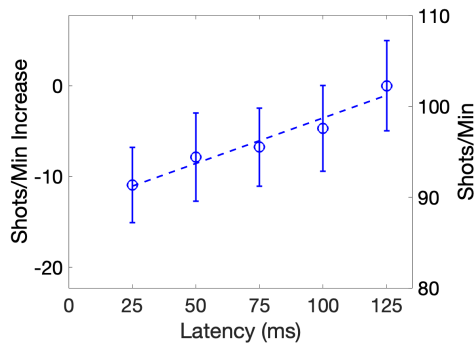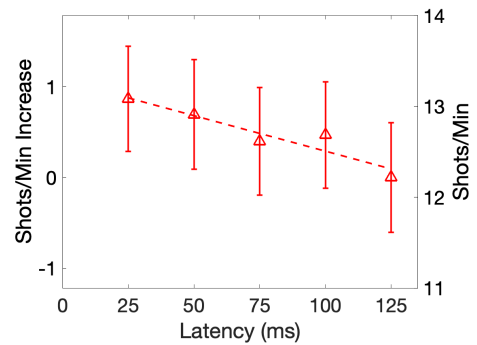
Figure 14: Shots fired – Automatic
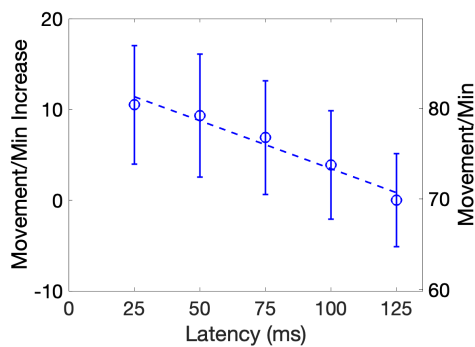

Figure 15: Shots fired – Sniper

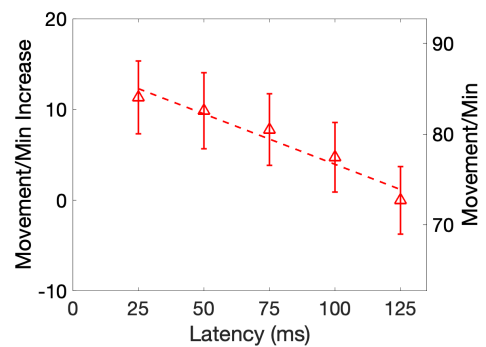**Figure 16: Movement – Automatic**
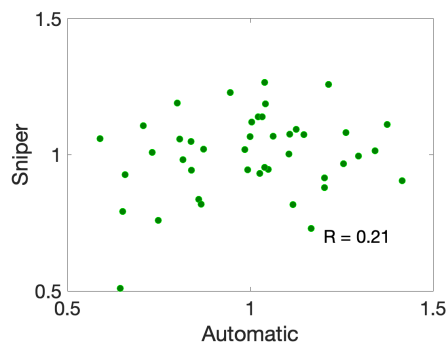

Figure 17: Movement – Sniper

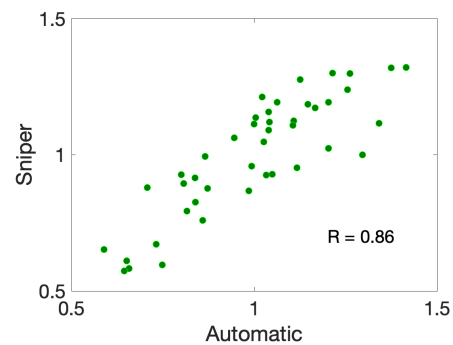**Figure 18: Accuracy – Sniper versus Automatic**

Figure 19: Score – Sniper versus Automatic

undistributed for the time required for one session (30 minutes), but whether those guidelines were adhered to could not be determined. As such, differences in seating, lighting and noise levels across test locations may have added unknown confounding effects.

Our user study intentionally focused on the effects of latency on individual player actions. However, as noted in Section 3, CS:GO is often a team game, where groups of players (typically 5 per team in esports) must work together to defeat the opposing team. The impact of latency on CS:GO team efforts, perhaps even team strategies, was not assessed.

As noted in Section 4.1, our study is considerably skewed towards males (only 1 female participated). While this may reflect the gender breakdown present in First-Person Shooter games and esports today, the results reported may not be indicative of female performance.

Serious game players often customize the software settings on their computers and games to suit their personal play preferences. For example, players may alter the mouse sensitivity or change the graphics resolution from the system defaults. These custom changes presumably improve the specific player's experience and

may improve the player's performance. However, since customizations that deviated from our settings create a difference in test conditions between users, we did not allow any changes to the computer or game settings.

6 CONCLUSION

With the growth in the game and esports industries, people are increasingly turning to games for entertainment, perhaps especially more so during the COVID-19 pandemic. Competitive game players generally seek fast systems in order to minimize delay between player input and the game response as output. While previous studies have assessed the impact of network latency on online games and local system latencies on player actions, the degree to which experienced game players may benefit from low-end local latencies is not well-known. Understanding the impact of local latencies may help better inform competitive gamers about system purchases and motivate system developers to minimize latency in their products, as well as guide researchers on devising systems and software to mitigate latency for games and game-like applications. Moreover, data from user studies can be helpful in modeling and simulating game systems.

This paper presents results from a COVID-era user study designed to provide for controlled latency conditions with limited confounding effects while still adhering to social distancing rules. Identical game systems were configured and distributed to qualified users in order for them to provide comparable game data for five levels of local system latency (25, 50, 75, 100 and 125 milliseconds), representing high-end through mid-range game systems. Forty-three people participated in the study, all highly experienced with the game under test: *Counter-strike: Global Offensive (CS:GO)* (Valve, 2012), a popular First-Person Shooter game used in esports. These users each played CS:GO for 25 rounds across 5 different latency conditions with two different weapons (100 total minutes of gameplay), providing objective player performance data (accuracy and score) via logs and subjective opinion data (Quality of Experience, QoE) via surveys.

Analysis of the results shows that across the range of local latencies studied, player performance and quality of experience both improve linearly as latencies decrease from 125 ms to 25 ms. Specifically, player scores at 25 ms average 20% higher than player scores at 125 ms, an equivalent of 5 additional kills or 10 additional assists in a 4 minute game. Over this same range, Quality of Experience (QoE) increases by about the same amount (20%), with the QoE at 125 ms being about 3 (out of 5) and the QoE at 25 ms being about

a point better at 4. These same latency reductions impact play with sniper rifles more than automatic rifles, the former weapon requiring more precision than the latter. Latency differences result in different impacts on play characteristics, too, with sniper rifle players shooting less at higher latencies and automatic rifle players shooting more. The results apply to local latencies, typical of high-end to mid-range personal computers. They also pertain to cloud-based game streaming systems with low latencies (e.g., due to edge-clustering), an increasingly important area for commercial game systems and game development.

Future work may investigate additional aspects of First-Person Shooter games. Specifically, the low-end of our latency range (25 ms and 50 ms) can be investigated to better understand the differences for the sniper rifle versus the automatic rifle – sniper rifle performance appears to be the same from 50 ms to 25 ms while the automatic rifle performance is not. We intentionally sought out experienced CS:GO players, but further analysis of how experience with a game alters the impact of latency may help fine-tune latency models [26] and the understanding of latency impact. Related to this, our skill criteria for participation was primarily time spent playing CS:GO, but assessing player skill in a game is more nuanced. Analysis of our self-rating data, shown previously to correlate with performance [27], including self-rating with a weapon, and external CS:GO rankings in relation to player performance and QoE are warranted. Other future work can explore latency's impact on players over a broader range of player skills (i.e., non-expert gamers) and player versus player, rather than player versus bots as in our study. While we studied the two most popular weapon types – automatic rifles and sniper rifles – other weapon types (e.g., shotguns) have different characteristics, such as an area of effect, that likely influence the impact of latency.

Other future work could apply the same methodology used in our paper to other game genres that are popular with competitive, esports players – e.g., Multiplayer Online Battle Arena (MOBA) games like DOTA 2 (Valve, 2013) and League of Legends (Riot Games, 2019) and Real-Time Strategy (RTS) games like Starcraft (Blizzard, 1998). Possible extensions include independent variables of network latencies and latency compensation techniques, and evaluation of latency's impact on different input and output devices – e.g., game controllers, TVs, touchscreens, and virtual reality.

REFERENCES

[1] Rahul Amin, France Jackson, Juan E. Gilbert, Jim Martin, and Terry Shaw. 2013. Assessing the Impact of Latency and Jitter on the Perceived Quality of Call of Duty Modern Warfare 2. In *Proceedings of HCI – Users and Contexts of Use*. Springer-Verlag, Berlin, Heidelberg, 97–106.

[2] Grenville Armitage. 2003. An Experimental Estimation of Latency Sensitivity in Multiplayer Quake 3. In *Proceedings of the 11th IEEE International Conference on Networks (ICON)*. IEEE, Sydney, Australia, 137–141. https://doi.org/10.1109/ICON.2003.1266180

[3] Yahn W. Bernier. 2001. Latency Compensating Methods in Client/Server In-game Protocol Design and Optimization. In *Proceedings of the Game Developers Conference (GDC)*. CMP Media, LLC, San Francisco, CA, USA, 13.

[4] Kuan-Ta Chen, Yu-Chun Chang, Hwai-Jung Hsu, De-Yu Chen, Chun-Ying Huang, and Ching-Hsien Hsu. 2014. On the Quality of Service of Cloud Gaming Systems. *IEEE Transactions on Multimedia* 16, 2 (2014), 480–495. https://doi.org/10.1109/TMM.2013.2291532

[5] Kajal Claypool and Mark Claypool. 2007. On Frame Rate and Player Performance in First Person Shooter Games. *Springer Multimedia Systems* 13, 1 (2007), 3–17.

[6] Mark Claypool and Kajal Claypool. 2006. Latency and Player Actions in Online Games. *Communications of the ACM* 49, 11 (2006), 40–45.

[7] Mark Claypool, Kajal Claypool, and Feissal Damaa. 2006. The Effects of Frame Rate and Resolution on Users Playing First Person Shooter Games. In *Proceedings of the ACM/SPIE Multimedia Computing and Networking Conference (MMCN)*, Vol. 6071. Association for Computing Machinery/Society of Photo-optical Instrumentation Engineers, San Jose, CA, USA.

[8] Mark Claypool, Ragnhild Eg, and Kjetil Raaen. 2017. Modeling User Performance for Moving Target Selection with a Delayed Mouse. In *Proceedings of the 23rd International Conference on MultiMedia Modeling (MMM)*. Springer International Publishing, Reykjavik, Iceland, 226–237.

[9] Mark Claypool and David Finkel. 2014. The Effects of Latency on Player Performance in Cloud-Based Games. In *Proceedings of the 13th Annual Workshop on Network and Systems Support for Games (NetGames)*. IEEE Press, Nagoya, Japan, Article 2, 6 pages.

[10] Victor Clincy and Brandon Wilgor. 2013. Subjective Evaluation of Latency and Packet Loss in a Cloud-based Game. In *Proceedings of the 10th International Conference on Information Technology: New Generations (ITNG)*. IEEE Computer Society, Las Vegas, NV, USA, 473–476.

[11] Ansgar Depping and Regan Mandryk. 2017. Why is This Happening to Me?: How Player Attribution Can Broaden Our Understanding of Player Experience. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, Denver, CO, USA, 1040–1052.

[12] Matthias Dick, Oliver Wellnitz, and Lars Wolf. 2005. Analysis of Factors Affecting Players' Performance and Perception in Multiplayer Games. In *Proceedings of 4th ACM Workshop on Network and Systems Support for Games (NetGames)*. Association for Computing Machinery, Hawthorn, NY, USA, 1–7.

[13] Matthew Dye, C. Shawn Green, and Daphne Bavelier. 2009. Increasing Speed of Processing with Action Video Games. *Current Directions in Psychological Science* 18, 6 (Dec. 2009), 321–326.

[14] E$ports Earning. 2019. Prize Money, Results, History, Statistics. Online: https://www.esportsearnings.com/. (Accessed January 5, 2021).

[15] Tobias Fritsch, Hartmut Ritter, and Jochen H. Schiller. 2005. The Effect of Latency and Network Limitations on MMORPGs: a Field Study of Everquest 2. In *Proceedings of the 4th ACM Network and System Support for Games (NetGames)*. Association for Computing Machinery, Hawthorne, NY, USA, 1–9.

[16] Maria Frostling-Henningsson. 2009. First-person Shooter Games as a Way of Connecting to People: 'Brothers in Blood'. *CyberPsychology & Behavior* 12, 5 (2009), 557–562.

[17] Christina Gough. 2020. eSports Audience Size Worldwide from 2018 to 2023. Statista. Online: https://tinyurl.com/y3tffxzo. (Accessed September 17, 2020).

[18] Sandra Hart and Lowell Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Human Mental Workload (Editors Peter A Hancock and Najmedin Meshkati)* 52 (1988), 139–183.

[19] HLTV. 2020. CS:GO Statistics Database – Distribution of Maps Played. hltv.org. https://www.hltv.org/stats/maps (Accessed September 17, 2020).

[20] HLTV. 2020. CS:GO Statistics Database – Top Weapons. hltv.org. https://www.hltv.org/stats?startDate=all (Accessed September 17, 2020).

[21] Oliver Hoßfeld, Hannes Fiedler, Enric Pujol, and Dennis Guse. 2016. Insensitivity to Network Delay: Minecraft Gaming Experience of Casual Gamers. In *Proceedings of the International Teletraffic Congress (ITC)*. IEEE, Würzburg, Germany, 31–33.

[22] Eben Howard, Clint Cooper, Mike Wittie, Steven Swinford, and Qing Yang. 2014. Cascading Impact of Lag on Quality of Experience in Cooperative Multiplayer Games. In *Proceedings of the 13th Annual Workshop on Network and Systems Support for Games (NetGames)*. IEEE Press, Nagoya, Japan, 1–6.

[23] Wijnand Ijsselsteijn, Yvonne de Kort, and Karolien Poels. 2013. *The Game Experience Questionnaire*. Technical Report. Technische Universiteit Eindhoven.

[24] Zenja Ivkovic, Ian Stavness, Carl Gutwin, and Steven Sutcliffe. 2015. Quantifying and Mitigating the Negative Effects of Local Latencies on Aiming in 3d Shooter Games. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, Seoul, Republic of Korea, 135–144.

[25] Jeroen Jansz and Martin Tanis. 2007. Appeal of Playing Online First Person Shooter Games. *Cyberpsychology & Behavior* 10, 1 (2007), 133–136.

[26] Shengmei Liu and Mark Claypool. 2021. Game Input with Delay - A Model of the Time Distribution for Selecting a Moving Target with a Mouse. In *Proceedings of the 27th International Conference on MultiMedia Modeling (MMM)*. Springer International Publishing, Prague, Czech Republic.

[27] Shengmei Liu, Mark Claypool, Bhuvana Devigere, Atsuo Kuwahara, and Jamie Sherman. 2020. 'Git Gud!' - Evaluation of Self-Rated Player Skill Compared to Actual Player Performance. In *Proceedings of the ACM Annual Symposium on Computer-Human Interaction in Play (CHI PLAY)*. Association for Computing Machinery, Online (Virtual Conference), 306–310. Work In Progress.

[28] Michael Long and Carl Gutwin. 2018. Characterizing and Modeling the Effects of Local Latency on Game Performance and Experience. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI Play)*. Association for Computing Machinery, New York, NY, USA, 285–297.

[29] Michael Long and Carl Gutwin. 2019. Effects of Local Latency on Game Pointing Devices and Game Pointing Tasks. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, Glasgow Scotland, UK, 1–12.

[30] Mathias Lux, Pål Halvorsen, Duc-Tien Dang-Nguyen, Håkon Stensland, Manoj Kesavulu, Martin Potthast, and Michael Riegler. 2019. Summarizing E-sports Matches and Tournaments: the Example of Counter-strike: Global Offensive. In *Proceedings of the 11th ACM Workshop on Immersive Mixed and Virtual Environment Systems (MMVE)*. Association for Computing Machinery, Amherst, MA, USA, 13–18.

[31] Ilya Makarov, Dmitry Savostyanov, Boris Litvyakov, and Dmitry I Ignatov. 2017. Predicting Winning Team and Probabilistic Ratings in "Dota 2" and "Counter-Strike: Global Offensive" Video Games. In *International Conference on Analysis of Images, Social Networks and Texts (AIST)*. Springer, Springer International Publishing, Moscow, Russia, 183–196.

[32] Emmie Martin. 2018. Super Bowl Champs Will Win Thousands - but They'd Earn 130 Percent More If They Played Baseball. CNBC - Money Online: https://www.cnbc.com/2018/02/02/how-much-super-bowl-winners-get-paid-compared-to-world-series.html. (Accessed January 5, 2021).

[33] Lothar Pantel and Lars C. Wolf. 2002. On the Impact of Delay on Real-Time Multiplayer Games. In *Proceedings of the Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*. Association for Computing Machinery, Miami, FL, USA, 23–29.

[34] Peter Quax, Patrick Monsieurs, Wim Lamotte, Danny De Vleeschauwer, and Natalie Degrande. 2004. Objective and Subjective Evaluation of the Influence of Small Amounts of Delay and Jitter on a Recent First Person Shooter Game. In *Proceedings of 3rd ACM Workshop on Network and Systems Support for Games (NetGames)*. Association for Computing Machinery, Portland, OG, USA, 152–156.

[35] Kjetil Raaen and Ragnhild Eg. 2015. Instantaneous Human-Computer Interactions: Button Causes and Screen Effects. In *Proceedings of the 17th HCI International Conference*. Springer International Publishing, Los Angeles, CA, USA, 492–502.

[36] Kjetil Raaen and Andreas Petlund. 2015. How Much Delay is There Really in Current Games?. In *Proceedings of ACM Multimedia Systems (MMSys)*. Association for Computing Machinery, Portland, OG, USA, 89–92.

[37] Kjetil Raaen Ragnhild Eg and Mark Claypool. 2018. Playing with Delay: With Poor Timing Comes Poor Performance, and Experience Follows Suit. In *Proceedings of the 10th International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, Sardinia, Italy, 1–6.

[38] Reuters.com. 2018. Investing in the Soaring Popularity of Gaming. Online: https://www.reuters.com/sponsored/article/popularity-of-gaming. (Accessed September 17, 2020).

[39] Saeed Shafiee Sabet, Steven Schmidt, Saman Zadtootaghaj, Carsten Griwodz, and Sebastian Möller. 2020. Delay Sensitivity Classification of Cloud Gaming Content. In *Proceedings of the 12th ACM Workshop on Immersive Mixed and Virtual Environment Systems (MMVE)*. Association for Computing Machinery, Istanbul, Turkey, 25–30. https://doi.org/10.1145/3386293.3397116

[40] Josef Spjut, Ben Boudaoud, Kamran Binaee, Jonghyun Kim, Alexander Majercik, Morgan McGuire, David Luebke, and Joohwan Kim. 2019. Latency of 30 ms Benefits First Person Targeting Tasks More Than Refresh Rate Above 60 Hz. In *SIGGRAPH Asia Technical Briefs*. ACM, New York, NY, USA, 110–113.

[41] Statista. 2020. Distribution of Gamers Playing Selected Game Genres Worldwide as of January 2017, by Gender. Online: https://tinyurl.com/yytrbj4d. (Accessed September 17, 2020).

[42] u/khaniage. 2019. Counter-strike: Global Offensive – Map Sizes. Reddit. https://www.reddit.com/r/GlobalOffensive/comments/a94jba/map_sizes/ (Accessed September 17, 2020).