Mouse Sensitivity In First-person Targeting Tasks

Ben Boudaoud *NVIDIA*
bboudaoud@nvidia.com
Josef Spjut *NVIDIA*
jspjut@nvidia.com
Joohwan Kim *NVIDIA*
sckim@nvidia.com

***Abstract*—Mouse sensitivity in first-person targeting tasks is a highly debated issue. Recommendations within a single game can vary by a factor of 10x or more and are an active topic of experimentation in both competitive and recreational esports communities. Inspired by work in pointer-based gain optimization and extending our previous results from the first user study focused on mouse sensitivity in first-person targeting tasks [1], we describe a range of optimal mouse sensitivity wherein players perform statistically significantly better in task completion time and throughput. For tasks involving first-person view control, mouse sensitivity is best described using the ratio between an in-game rotation of the view and corresponding physical displacement of the mouse. We discuss how this displacement-to-rotation sensitivity is incompatible with the Control-Display (CD) gain reported in traditional pointer-based gain studies as well as other rotational gains reported in head-controlled interface studies. We provide additional details regarding impacts of mouse dots per inch (DPI), on reported sensitivity, the distribution of spatial difficulty in our experiment, our submovement parsing algorithm, and relationships between measured parameters, further demonstrating optimal sensitivity arising from a speed-precision trade-off. We conclude our work by updating and improving our suggestions for mouse sensitivity selection and refining directions for future work.**

***Index Terms*—pointing devices, mouse, mouse sensitivity, first-person targeting, first-person games**

I. Introduction

Mouse sensitivity is the relationship between physical motion of a mouse and the virtual motion this induces in a computer application. This virtual motion can be translational (e.g., pointer-based tasks), rotational (e.g., first-person targeting), or even categorical (e.g., directional selection). Additionally this relationship can be 0th order (positional), 1st order (velocity-based), or even higher-order depending on the interaction in question. In modern computer systems mouse sensitivity is the product of several factors, including the mouse sensor resolution (a hardware limitation of the mouse), mouse firmware or driver settings that scale reported motion, as well as the operating system's and application's sensitivity settings.

The competitive gaming community considers mouse sensitivity a significant factor impacting player performance in targeting tasks [2]. Yet, there is wide variation in individual professional player settings. To demonstrate this, we analyze sensitivity settings data from prosettings.com for professional esports athletes across multiple First Person Shooter (FPS) gaming titles. Fig. 1 shows that these sensitivities vary across professional gamers by up to 8x within a single title and by up to 12x across titles. Furthermore, the default sensitivities for some of these games, when scaled by today's common mouse DPIs, lie beyond even the 12x range commonly used by professionals (see Table I). The combination of mouse hardware (e.g., DPI), operating system scaling, and application sensitivity settings make it hard to control mouse sensitivity and provide reasonable defaults to new users.

TABLE I
Default mouse sensitivities (in °/mm on a log x-axis) by game and DPI. The game only sets the in-game multiplier, ignoring DPI, which depends on the mouse sensor settings. Many gaming mice use 3200 DPI as a default while standard consumer mice tend to use 800 DPI by default.

DPI	Valorant	Fortnite	CS:GO	Overwatch	Apex Legends
800	2.20	4.90	1.73	3.12	3.46
3200	8.81	19.60	6.93	12.47	13.86

Academic research on mouse sensitivity predominantly focuses on productivity-oriented, pointer-based tasks, leaving a gap in research around first-person targeting. Many pointer-based studies recommend control-display (CD) gains of 2-8 [3], [4], [5], [6], [7], [8], or over a factor of 4x in sensitivity range. This limited range makes sense, as pointer-based interaction is a reasonably uniform and repeatable task given similar user input/output devices and content mapping. Unlike pointer-based interfaces, designed for ease of selection, FPS games present dynamic targets of various sizes at different virtual distances, deliberately challenging the player and rewarding high skill. In this way, competitive FPS games eschew strict adherence to traditional user interface design patterns, as skilled gamers attempt to optimize mouse sensitivity, balancing aiming speed and precision.

FPS targeting tasks have been demonstrated to conform to Fitts' law [9], but this does not imply that they share a common set of optimal sensitivities with pointer-based targeting tasks. In fact, the mouse sensitivities used to characterize pointer-based targeting sensitivity are not directly convertible to first-person targeting sensitivities. This work presents a deeper analysis of the first study of mouse sensitivity's impacts on FPS targeting task performance, reprising Boudaoud et al. [1], providing data and analysis further connecting and comparing pointer-based and first-person targeting mouse sensitivity.

II. Background

Prior art has explored various aspects of spatio-temporal task difficulty as well as the (orthogonal) effects of mouse

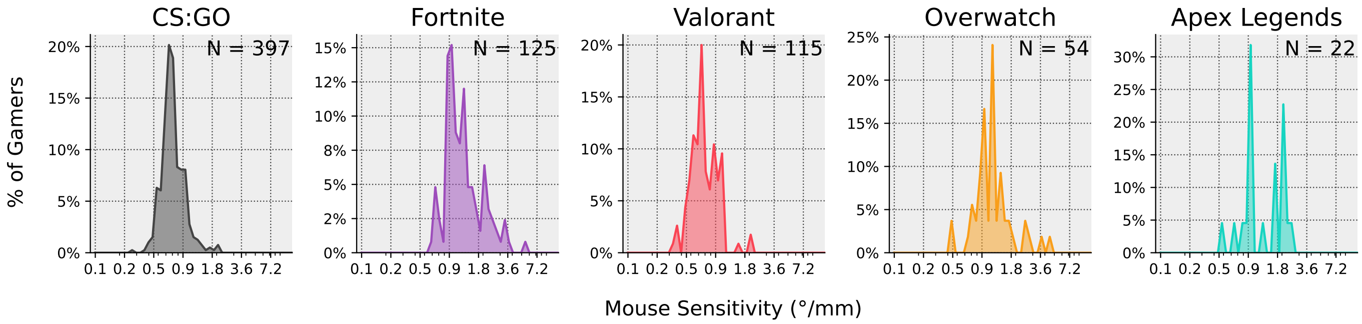

Fig. 1. Professional player mouse sensitivities by game played (normalized per game). All games use the mouse for rotational camera motions. Note the logarithmic X-axis. The sensitivities are provided in cm/360°, data sourced from prosettings.com in Sept. 2020.

sensitivity on pointer-based targeting tasks. In this section we describe this related work foundational to our motivation and analysis.

A. Sensitivity and Fitts' Law

Fitts' law [10] is a widely accepted model explaining targeting task difficulty as a function of target distance and size. Nearly all formulations of Fitts' law and its variants [10], [11], [12] normalize the distance to a target (D) by the target width (W) in common units, producing a unitless ratio. Index of Difficulty (ID), the measure of spatial task difficulty, is a base-2 logarithm of the D/W ratio (measured in bits), and is commonly linearly fit to task completion time. For ID computation in this work we use the Shannon formulation of Fitts' law as proposed by MacKenzie et al. [13], provided in Eq. 1. This formulation adds one to the D/W ratio inside of the log, akin to the formulation of Shannon's channel capacity with signal-to-noise ratio replaced by D/W, avoiding negative IDs when the width is greater than the distance to the target.

$$ID = \log_2\left(\frac{D}{W}+1\right) \text{ bits} \tag{1}$$

Task *throughput* (TP) [14], also referred to as index of performance (IP) [11], is the ratio of ID to task movement time (MT) (measured in bits/s) (see Eq. 2).

$$TP = \frac{ID}{MT} \tag{2}$$

Throughput approximates the *inverse slope* of the linear fit of task ID to movement time when the y-intercept of this fit is small as demonstrated in Eq. 3.

$$MT = m \times ID + b \approx \frac{1}{TP} \times ID \tag{3}$$

In doing so, it attempts to account for changes in movement time resulting from motor task difficulty by normalizing the two into a positive metric (the higher the throughput the higher the performance).

However, Fig. 2 demonstrates that changing mouse sensitivity either scales both D and W, in physical mouse movement space, or neither, in in-game rotation space, leaving the task ID unmodified. Instead, trends in optimal mouse sensitivity appear as changes in task throughput, an intuitive trend as

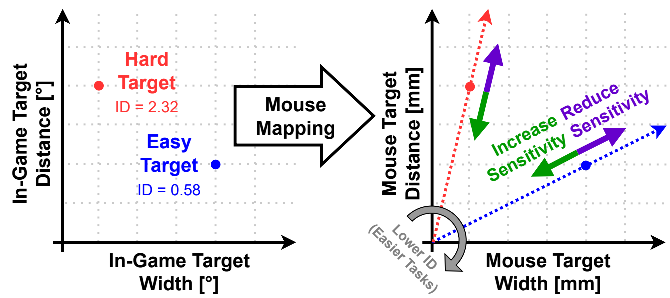

Fig. 2. Plots demonstrating the mapping of an in-game, virtual rotational task to the (physical) mouse motion space under varying sensitivity, together with the orthogonality of sensitivity and task difficulty (ID).

the sensitivity of the mouse certainly impacts the efficacy of motor actions. Mouse sensitivity has been demonstrated to impact movement time, with previous work showing significant interaction with target size and task ID [4], [7]. We (partially) explain the effect of mouse sensitivity on task performance using kinematic characteristics observed in our data in Section IV-D.

B. Mouse Sensitivity in Pointer-based Targeting

Prior work in optimal mouse sensitivity has focused on pointer-based, Fitts' law targeting tasks. In such tasks users move the mouse, translating an on-screen virtual proxy (cursor), towards a target, then click to indicate a selection. The control-display (CD) gain, or ratio between the travel distance or speed of a controller (e.g., mouse) and the corresponding cursor movement on a display, has been the conventional choice of sensitivity metric in this work. Reported optimal CD gains vary from 2 to 16 (an 8x range) [6], [15], [4], [7], [5] with some studies reporting higher optimal CD gain for tracking tasks [16], as shown in Tab. II.

Older studies [6], [15] tend to suggest lower CD gain, concluding mouse sensitivity does not significantly impact task completion time, with the quantization error of lower DPI mouse sensors limiting the range of sensitivity interesting for study. Later work [8], [4], [7], [3] implies that movement time does depend on mouse sensitivity, and highlights interactions with target size. Most recently, these results have been extended by monitoring limb usage range and rate of motion to show users do in fact scale their kinematic interaction with CD gain [5]. Additional studies indicate that tracking task completion time benefits from higher sensitivities or

TABLE II
A COMPARISON OF PREVIOUS POINTER-BASED CD GAIN STUDIES FOR THE OPTIMAL SENSITIVITY WITH VARYING CONDITIONS. NOTE 2-6X RANGE WITHIN A STUDY, AND LARGER DIFFERENCES BETWEEN THEM.

Study	Year	Tested Gains	Reported Best
Jellinek [6]	1990	1, 2, 4, 8, 16, 32	2
Trankle [15]	1991	1, 2	None
Lin [8]	1992	0.5, 1, 2, 4	2
Bohan [4]	2003	1, 2, 4	2
Sandfeld [7]	2005	2, 4, 8	4
Casiez 1 [5]	2008	1, 2, 4, 6, 8, 12	4, 8
Casiez 2 [5]	2008	2, 5, 8, 12, 16, 20	16
Senanayake [16]	2016	2.3, 5, 10, 15	9, 12
Pang [3]	2019	1.2, 2.4, 14.5, 38.6	2.4, 14.5

more direct methods of interaction like touch interfaces [16]. Broadly speaking all surveys tend towards globally optimal CD gains in the range of 2-8, which aligns well with the default gains applied by operating systems and pointer-based user interfaces at common DPI values [17].

a) CD Gain and Display Size/Distance: Conventional CD gain, though a commonly reported sensitivity metric for pointer-based tasks, has a notable shortcoming in that the gain ratio takes into account the physical *display*-space motion of a cursor. A large display viewed at a correspondingly large distance, such that it subtends the same visual angle as a common desktop monitor, will have a substantially higher CD gain than a perceptually equivalent desktop display. That is, though the observed angular displacement of a cursor resulting from the same user hand motion may be nearly identical between these cases, the CD gain is higher for larger displays as the cursor travels farther in display space [5]. In spite of CD gain being incompatible with rotational FPS mouse sensitivity (see the following section), this display-contingent scaling of the metric means it is best applied to a relatively narrow range of display size, viewed at a consistent distance. We suggest CD gain is a reasonable metric when looking at a small/common range of display sizes and viewing distances, but that the metric be avoided when considering broader ranges of display viewing conditions.

C. Mouse Sensitivity in First Person Targeting

First-person targeting interfaces, such as those used in FPS games, often provide an aim point indicator, referred to as the crosshair or *reticle*, fixed at the center of the screen. Users rotate the view direction in azimuth and/or elevation by translating the mouse horizontally and/or vertically respectively, aligning a (movable) target with the (fixed) reticle in screen-space. Since the view direction rotates, the entire virtual world, including the target, moves in the direction opposite the mouse motion (e.g., mouse moves left, target and scene move right). Similarly to pointer-based interfaces, the user clicks to perform selection. Despite obvious differences from pointer-based targeting, both approaches share similar spatial task difficulty characteristics explained well by Fitts' law [9]. The difference between pointer-based CD gain and first-person rotational mouse sensitivity is further demonstrated in Fig. 3.

In the FPS genre, many games provide settings sliders with the label "mouse sensitivity". Such sliders are typically

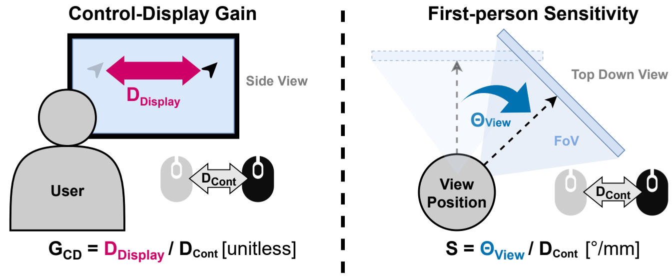

Fig. 3. A comparison of CD gain and displacement-to-rotation sensitivity in their respective contexts (pointer-based and first-person targeting). CD gain uses common units of displacement (or velocity) to create a unit-less gain comparing mouse to pointer movement. Displacement-to-rotation sensitivity converts the same mouse movement into rotation of a view frustum about the current view position in the application's world space, resulting in a ratio that must have units.

TABLE III
CONVERSION BETWEEN CM/360°, CM/°, AND °/MM MOUSE SENSITIVITY SETTINGS SELECTED FOR OUR STUDY. THE RELATIONSHIP BETWEEN CM/360° AND °/MM IS GIVEN BY $y = 36/x$ WHERE y IS IN °/MM AND x IS IN CM/360°.

cm / 360°	160	80	40	20	10	5
cm / °	0.444	0.222	0.111	0.056	0.028	0.014
° / mm	0.225	0.45	0.9	1.8	3.6	7.2

unit-less, with higher values corresponding to faster view rotation per unit of physical displacement. The slider's values (when reported) do not generalize across games and require a conversion factor to provide common units for gamers to communicate or transfer them between games. FPS gamers use a variety of metrics to discuss mouse sensitivity, the most universal of which is a displacement-to-rotation ratio, commonly measured in cm/360° (cm per full turn).[1] The primary motivation for this metric is ease of in-game measurement through moving the mouse in one direction until a full rotation has been completed then measuring the real-world distance the mouse traveled to complete that turn. We use an inverse metric, °/mm (degrees per millimeter) to match the intuition that higher sensitivity means the aim moves more quickly per unit displacement. For those who may be more familiar with the cm/360° metric, Table III maps the two for the settings used in this study.

a) CD Gain and First Person Targeting: In addition to the challenge of CD gain scaling with display size, there is no direct conversion between a CD gain and the cm/360° or °/mm sensitivity settings used in FPS targeting applications. One could attempt to convert historical CD gains into displacement-to-rotation sensitivities by translating display space pointer motion (at an assumed distance) into angular motion. However, if we were to convert a CD gain to an assumed °/mm ratio, we would confound degrees of user focal point translation or head rotation when tracking a pointer with degrees of camera rotation in mouse controlled cameras. This challenge is further complicated by common FPS practices, such as projecting a large rendered field of view (typically

[1]Note that this metric assumes that the mapping of mouse displacement to view rotation is purely linear, or that no mouse acceleration is applied. For more information on mouse acceleration refer to the paragraph on non-linear gain (acceleration) in first person targeting below.

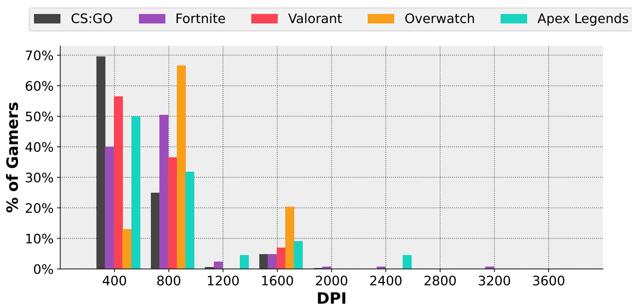

Fig. 4. Distribution of DPI settings for the pro settings data used in Fig. 1 above. Note the continued preference for low(er) DPI settings in spite of much higher DPI mice being available. Data sourced from prosettings.com in Sept. 2020

$\geq 100°$) onto a display that subtends a much smaller physical field of view (typically $< 50°$). As well as the user fixating at the center of the display instead of tracking a mouse cursor as it moves within the display area.

In an attempt to validate our results based on prior art measuring optimal CD gain in pointer-based tasks, we compare our experimental results to reported CD gains using self-referential multiplicative ranges (i.e., the max to min ratio of 4x between a reported CD gain of 2 and 8 can be compared to that of a first-person sensitivity range of 0.5 and 2 °/mm). We avoid comparing reported CD gains between studies as different display size or viewing conditions may change the reported optimal values.

b) Impacts of Mouse DPI: An additional point of confusion is that mice and mouse sensors report a parameter called dots per inch (DPI) or counts per inch (CPI). These terms tend to be used interchangeably with higher values increasing the CD gain and effective mouse sensitivity in game as demonstrated in Tab. I. Gamers sometimes use an effective DPI (eDPI) metric, the product of the mouse DPI and the game sensitivity value, to report sensitivity. However, since games use different sensitivity scales, these eDPI values are really only comparable within a single game and not useful more broadly. We anticipate that the FPS esports community will benefit from standardizing on a metric such as °/mm, which we use in this work, as it disambiguates differences in game-specific sensitivity scales and additional OS-applied sensitivity not captured by eDPI.

While the academic community has largely agreed that lower DPI values limited the optimal reported CD gains from early work (see Tab. II) it is worth noting many professional gamers still prefer 400 and 800 DPI settings, as demonstrated in Fig. 4. These lower DPI mouse settings trade off increased quantization error in movement with increased stability of the view when not moving (i.e., ease of keeping motor noise to less than a quantization step). This allows gamers to avoid small, undesirable, noisy view direction changes more easily at the cost of slightly higher quantization error in controlling the exact position of the aim direction [18]. We used a fixed DPI value of 3200 dots/inch to avoid quantization effects in our study, but acknowledge this represents a departure from the DPI values most commonly selected by professional gamers. This choice was deliberate as we designed our study to make sure single pixel aim displacement was possible in-game, if we chose to adopt a more common DPI value of 400-800 to reduce noise, we would have also introduced quantization effects wherein one dot of mouse displacement would result in a view rotation exceeding one pixel at the center of the screen, particularly at the highest tested sensitivity values. Effectively we opt for small mouse motions to create noticeable view direction change ahead of quantizing large(r) movements into single dots of mouse motion, mapping larger action spaces to a single outcome in-game.

c) Non-linear Gain (Acceleration) in First Person Targeting: Though common in pointer-based interfaces [5], [17], [19], velocity-dependent sensitivity, often referred to as cursor/pointer acceleration, enhanced pointer precision, or non-linear CD gain, has been historically avoided in competitive FPS game play [20], [21], [22]. When applying nonlinear gain, mouse sensitivity is no longer a constant, but rather a (user-specified) function of the velocity at which the mouse is currently moving, supplanting use of a single CD gain or °/mm sensitivity with a curve mapping the choice of sensitivity value to the mouse speed.

We suggest that competitive FPS gamers' aversion to non-linear sensitivity is sourced largely from the added challenge it presents in actuation space, where skilled gamers often operate using an open loop targeting model to minimize task completion time. This skill is often referred to as the *flick shot* technique or simply *flicking*. In the flicking actuation model, players learn the spatial relationship between screen and cursor space, and *flick* the mouse to align with a target and click as quickly as possible, without performing additional visual verification.

When flicking, controlling the speed of view rotation is more difficult as the movement takes place over a very short time period, with repeat-ability of net displacement being of paramount importance. We believe this is likely due to the cost of an additional perception-action cycle (to correct the flick) effectively negating any value of the technique over more traditional tracking approaches. For this reason, we suggest players prefer to simply learn to target their flick shots to the *displacement* space of the mouse pad, explicitly avoiding the more precise speed control required to get benefit from nonlinear mouse sensitivity.

It is worth noting that some subsets of the gaming community have, and continue to, attempt to optimize various nonlinear sensitivity curves for various games of their choosing [23], [24]. However, the nonlinear sensitivity curves arrived at for these applications do not typically conform to those suggested in prior academic literature or modern operating systems as they are deliberately intended to be easier to internalize and adapt to quickly for open loop actions (i.e., 2-tier sensitivity structures).

d) Rotation To Displacement Sensitivity in Head-controlled Interfaces: The sole prior result we are aware of in rotation-to-displacement sensitivity [25] focuses on head-controlled computer input devices, where a user's real world head rotation drives an on-screen pointer's translation, the inverse of translating a mouse to drive an virtual view rotation. This work did find significant differences between different rotation-to-displacement ratios, but did not explore wide enough a range of sensitivities to identify a clear optimal region. The range of sensitivity studied was likely relevant to the particular interface used in the demo. The authors did observe that movement time decreased as they increased the rotation-to-displacement ratio over the range of 0.1-0.2°/mm. In addition, the authors note that while user's preference for a displacement-to-displacement gain changed with viewing distance, per the suggestion in the *CD Gain and Display Size/Distance* paragraph above, subjects preference for rotation-to-displacement sensitivity did not significantly change with viewing distance.

III. Experiment Design

We designed an experiment to better understand the interaction of target width and distance with mouse sensitivity in First Person Science (FPSci), a publicly available, open-source FPS experimentation platform [26], [27]. We provide our experiment configuration files along with this work to enable others to repeat the study, or develop related experiments.

A. Questions and Hypotheses

The following questions motivate our experimental design:

a) Does a single optimal sensitivity (region) exist among competitive FPS gamers?: We hypothesize that the optimal sensitivity regions observed in prior art arise from humans' innate nonlinear kinematic characteristics, particularly those around small targets and/or long travel distances (higher spatial difficulty tasks). If this is true, we should observe similar trends among competitive FPS gamers and more conventional interface users despite years of aim training experience. Namely, that an optimal mouse sensitivity (range) arises from a speed-error trade-off which interacts with task spatial difficulty, implying that simply minimizing human motion by maximizing sensitivity is not a practical solution. To test this hypothesis, we included 6 fixed sensitivity conditions exponentially spaced over the set of sensitivities reported from our survey of esports athletes (0.225, 0.45, 0.9, 1.8, 3.6, 7.2 °/mm). We also widely varied target distance and size, into ID ranges at which nonlinear motion characteristics have been observed in prior art.

b) Does an individual competitive gamer's preferred sensitivity setting approximate their generally optimal setting(s) for FPS targeting tasks?: We hypothesize individual gamers' preferred sensitivity settings do tend to lie within regions of (near) optimal performance given sufficient time to adapt to in-game conditions. To validate this hypothesis, we allocated a preferred sensitivity condition as the first session for all the participants. We re-used subjects' preferred sensitivity setting, selected during practice in a previous (similar) aiming task wherein subjects could select and refine their choice of sensitivity. Subjects were not made aware that these previous preferred sensitivities were among the tested conditions. We compare subjects' performance at their preferred sensitivity to the surrounding sensitivities from our pre-selected set to evaluate this claim.

c) Does long(er) term muscle memory play a crucial role in aiming performance?: Competitive gamers commonly assert that muscle memory is "hard-wired" and thus their performance suffers when changing sensitivity too frequently. We hypothesize that changing sensitivity (at least within the duration of this experiment) does not have a significant impact on overall performance. To test whether this is true, we also allocate a preferred sensitivity condition as the final session of data collection, comparing these results to those from the first session played at the same preferred sensitivity. By the time they start the last session, participants have experienced 6 widely varied sensitivity conditions, over a 32x range. This should considerably confuse muscle memory and, if it plays a crucial role, performance in the last session should be significantly worse than the first.

B. Subjects

15 gamers (age 11-34, 1 female and 14 males) voluntarily participated in the experiment. All subjects were experienced FPS players with minimum play times of 10-20 hours per week. Subjects were not made aware of the experimental hypothesis. All subjects gave informed consent (for the two under-aged subjects, age 11 and 14 respectively, their parent gave informed consent). The experiment was conducted in accordance with the Declaration of Helsinki with each subject household provided their own dedicated system to minimize the need to clean/sanitize hardware resources between users.

One subject's system experienced a degradation in performance (frame rate) during data collection, as a result we excluded this user from our data set. Two other subjects skipped a significant portion of (difficult) trials due to fatigue. Of these two subjects, one subject voluntarily repeated the experiment without skipping trials. We chose to include this re-run subject data (shown in orange in all per-user plots) as much of our analysis was within subjects and, as the subject was already familiar with the experimentation platform, we did not observe evidence of significant long-term training effects. We excluded the data from the other subject who skipped trials. In total, 13 out of 15 participating subjects' data are included in our analysis (age 11-33, 1 female and 12 males).

C. Task and Stimulus

Subjects performed a first-person perspective aiming task with a stationary target (as shown in Fig. 5). We avoid rendering a weapon or complex background scene to minimize target view occlusion or possible distractions while subjects performed the task. We targeted a 240 fps rendering rate to match our display's 240 Hz update rate. FPSci is designed to run at high frame rates on a wide range of systems, but nonetheless, we validate that the application was performant by checking its reported measured frame times during analyzed trials. Overall more than 99.8% of all frames were rendered within 1 ms of the 4.167 ms target frame time, with the worst-case user having just 1.5% of frames lasting longer than 5.167 ms. This is visualized in Fig. 6, the linear y-axis plot demonstrates the small percentage of frames exceeding our target frame time and the log y-axis plot reveals that for

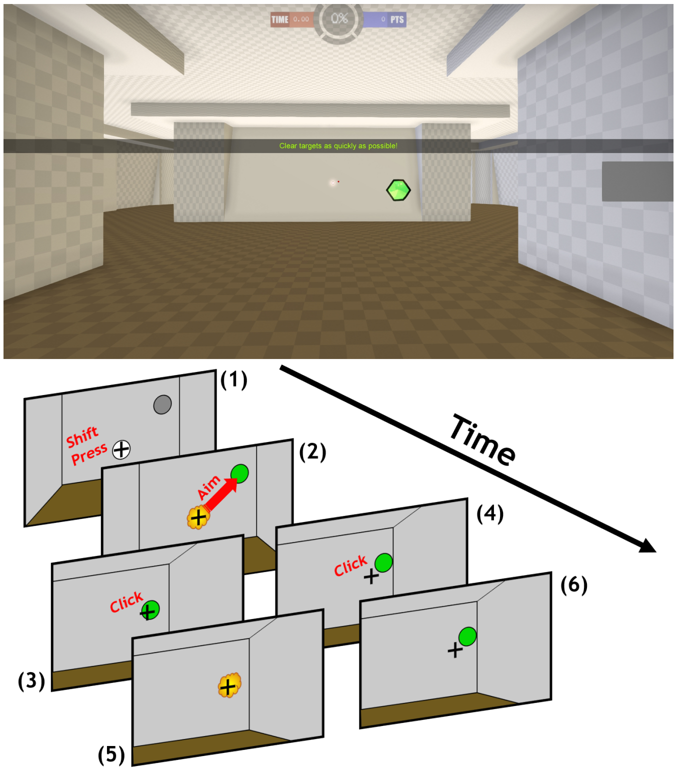

Fig. 5. (Top) A screenshot from our application, the gray/white square at right was used to monitor system latency during the experiment. (Bottom) Flow of a single trial for our FPS task. (1) a reference target, shown in white, is used to recenter the user's view direction at the start of the trial, with the trial's test target displayed in gray to familiarize the subject with it's position. (2) after destroying the reference target the user tries to move their aim to the, now green, task target and click as quickly as possible. If the target is hit (3) the trial ends successfully (5), otherwise on a miss (4) the trial ends in failure (6).

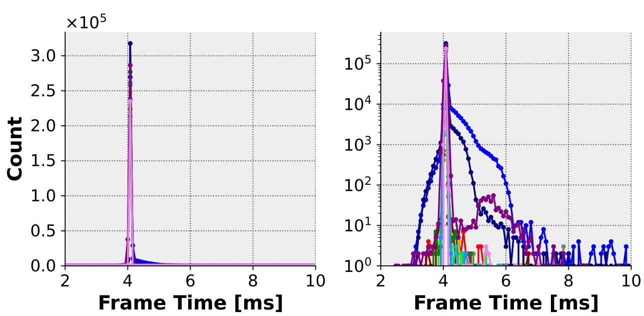

Fig. 6. The distribution of frame rate for our application across all users (1 color per user matching results below) using both a linear (left) and log (right) y-axis. It is interesting to note that our poorest performing users did appear to have worse system performance. Fortunately our analysis is within subjects and we predict the impact of these frame time variations to be relatively minor.

some subjects frame time varied more significantly than others. We do not consider this variance significant enough to impact the validity of our results.

A trial began with presenting a stationary, white reference target 0.57° in diameter (8 pixels at the center of the screen) and a stationary, but inactive, grey test target preview. Subjects aimed by translating the mouse to align their reticle with the reference target by rotating the view. The task timer started as the subject cleared the reference target by pressing the left shift key on the keyboard, activating the test target (indicated visually by changing its color to green). We chose to use left shift to begin the trial as this avoided effects from multi-purposing the left mouse button for both starting and ending the trial, including accidentally skipping trials by clicking twice. The subject then quickly changed their aim by moving the mouse toward the test target. The task timer stopped when the subject clicked the left mouse button, with only a single click allowed per trial. If the reticle was aligned with the target when this click occurred the task is completed successfully, otherwise the task ends in failure. Because the test target was visible before the timer started, the measured time consisted only of motor response time (visual search time and initial reaction time were excluded).

Subjects were instructed to clear the test targets as quickly and accurately as possible, and provided with a score indicative of this metric that was updated per trial. If the reticle was within the target bounds at the time of click, then subjects were scored by the amount of remaining task completion time in seconds (faster completion implies higher score). In the case of a failure, or the reticle being outside of the target at the time of click, the maximum completion time (1.5 s) was assumed for scoring purposes, resulting in a score of 0. The score penalty of the maximum completion time was substantially larger than that of the average task completion time (typically 0.4 - 0.8 s). This scoring system is intended to motivate the subjects to maintain relatively high accuracy levels throughout the experiment (average hit rate of >85%).

D. Conditions

Each subject completed 8 sessions of 500 trials, with one mouse sensitivity level assigned to each session. The first and last sessions used the subject's preferred sensitivity setting. We varied target distance and size within each session to provide targets with a wide range of index of difficulties (see Table IV).

1) Sensitivities: Sensitivity was the main variable of interest in our study. We selected 7 sensitivity levels for exploration: 0.225, 0.45, 0.9, 1.8, 3.6, 7.2 °/mm selected for their round number in the cm/360° scale (Table III), as well as the preferred sensitivity of each subject. We controlled sensitivity in FPSci by changing the scaling multiplier between the movement of the mouse and the rotation of the rendering camera in the experimentation platform.

2) Target Width and Eccentricity: In each condition we provided targets that varied both in width and eccentricity (i.e., distance from the central view direction, where the reference target was located). The targets were all icosahedra with (in-game/virtual) diameters of: 0.57°, 1.15°, 2.29°, 4.59°, and 9.15°, expressed in perspective angle from the player view camera, which used a 103° rendered horizontal field of view. When the targets are projected near the center of the screen (25 inch or 635 mm, 1920x1080 monitor), these targets were 8, 15, 31, 61, and 122 pixels wide respectively.

The target distance from the central view direction randomly varied between 7-25° horizontally and 0-2.8° vertically, with

TABLE IV
INDEX OF DIFFICULTY RANGE (IN BITS) AS COMPUTED BY EQUATION 1) FOR OUR SELECTED TARGET WIDTH AND DISTANCE PAIRINGS. ALL TARGETS WERE WITHIN THE RANGE OF 0.83-5.50 BITS OF DIFFICULTY WITH AN EQUAL NUMBER OF TARGETS DRAWN FROM EACH RANGE OF CONDITIONS DESCRIBED BELOW.

Target Distance (Azimuth x Elevation)	**Target Width** 9.15°	4.59°	2.29°	1.15°	0.57°
7-9° x 0-1°	0.83-0.99	1.35-1.57	2.03-2.31	2.84-3.15	3.74-4.08
9-11.7° x 0-1.3°	0.99-1.19	1.57-1.84	2.30-2.62	3.14-3.49	4.07-4.44
11.7-15° x 0-1.7°	1.19-1.41	1.83-2.10	2.61-2.92	3.48-3.82	4.43-4.78
15-19.4° x 0-2.2°	1.40-1.65	2.10-2.40	2.92-3.25	3.81-4.17	4.77-5.14
19.4-25° x 0-2.8°	1.64-1.91	2.39-2.70	3.24-3.58	4.16-4.52	5.13-5.50

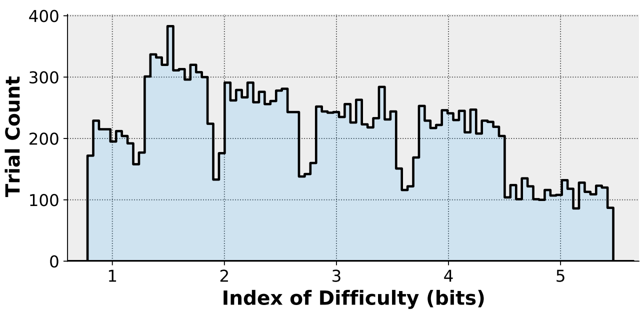

Fig. 7. Distribution of index of difficulty (ID) over the ∼26k total trials analyzed in this experiment. Note that the distribution is not uniform due to the overlap of width and distance ranges in Tab. IV and the uniform sampling of ID ranges of different width. Bins are drawn with flat tops to indicate histogram quantization effects.

this distance range logarithmically divided into five sub-regions for a more uniform distribution of index of difficulty (Table IV). Each trial's target distance was drawn uniformly from the sub-region designated for that trial, note this does not imply the overall distribution of ID was uniform as demonstrated in Fig. 7. This distributional shape arises from the random sampling within ID ranges/plot binning (producing small amounts non-uniformity across all values) and the imperfect alignment of ID across target widths, in spite of the target distances being selected for a continuous range of ID for any one target width (producing larger-scale binning effects). The total distance variation corresponded to 93-339 pixels from the central view direction. The combination of selected target distance and width resulted in a range of ID of 0.83-5.50 bits.

3) Equipment: All subjects used their own, identical PC setup (Intel Core i7-9700k @ 3.6 GHz, 32 GB of RAM, RTX 2080 Ti). Subjects were seated about 24 inches (610 mm) from a 240 Hz, 1920×1080 25 inch (635 mm), LCD monitor. All subjects completed the experiment with a Logitech G203 mouse set to a 3200 DPI sensor resolution, which was sufficient to provide pixel-level rotations in all tested sensitivity conditions, preventing any impact(s) of quantization.

E. Procedure

All subjects had experience with the experimentation platform prior to this study, acquainting them to the visual environment, controls, and high-level targeting task. No further orientation was needed beyond clarification on the scoring system used in this experiment. To encourage all subjects to put in their best effort, we issued a $25 gift card to the subject with the highest score after completing all 8 sessions.

Each session consisted of 500 trials (4000 trials/subject total), which were further divided into 10 blocks of 50 trials. Subjects were encouraged to take breaks between blocks to refresh themselves as needed. Subjects could complete as many sessions as they liked in a single sitting, but were strongly encouraged to stop if they felt any mental or physical fatigue. Each session took about 10 minutes to complete, with the entire experiment lasting about 80 minutes per subject. All participants completed the experiment within a single week.

IV. RESULTS AND ANALYSIS

We analyze our results from a mean and variation-based perspective, as all of these metrics impact player performance in the competitive gaming context. We consider hit rate, task completion time, and task throughput (using a Fitts' based ID formulation). Additionally, we segment submovements of players within trials to help better decompose task completion time into its (non-overlapping) constituent parts, illuminating how mouse sensitivity impacts different phases of human motion.

We discard the first 50% of trials from each session as an adaptation period for the new sensitivity condition, leaving 250 trials per session, or 2,000 trials per user, for analysis (see Section V and Fig. 15 for more information on average adaptation rate). Not all trials were successfully completed, in some cases the user clicked with the reticle outside of the target bounds. We choose to analyze unsuccessful trials along with successful ones based on the assumption these trials were still valid aiming attempts. Accordingly, we discard trials in which the aim error (distance from the center of the target) at the time of click was greater than 50% of the initial distance to the center of the target. These are trials where the user translated the view less than halfway to the target center before clicking. 45 total trials were removed based on this criteria, accounting for <0.2% of all data and <0.5% (10/2000) of any given user's trials.

A. Hit Rate

We analyze our data for hits as a percentage of total shots or trials, where each trial has a single shot that is either a hit or miss. As expected, our subjects maintain a hit rate above 85% for almost all sensitivity conditions. At the highest sensitivity setting nearly all users' hit rate decreased. Hit rate does not demonstrate a strong optimal range overall or per user in the

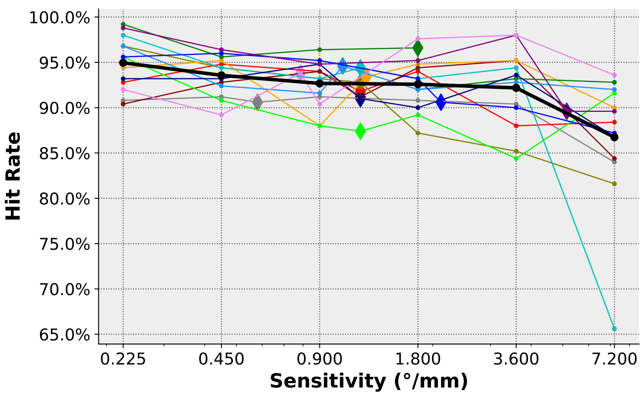

Fig. 8. Subject trial hit (success) rate vs sensitivity per user (colors) and overall (thicker black line). Sensitivity preferences are displayed as diamonds. Note the X-axis is logarithmic. Generally speaking subjects maintained a high (>85%) hit rate across all conditions.

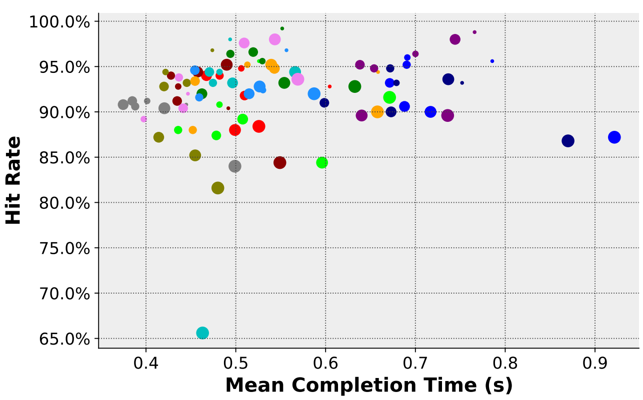

Fig. 9. Scatter plot of hit rate vs completion time per user (colors), with sensitivity indicated by marker size (larger marker implies higher °/mm sensitivity) demonstrating a lack of strong correlation between accuracy and task completion time, implying trials where the target was missed did not end significantly sooner or later than those where the target was hit.

range tested. Generally speaking hit rate was highest for the lowest sensitivities, with minimal change across all but the highest sensitivity tested. Though ANOVA reports a significant main effect ($F(5, 72) = 6.12$, $p < 0.0001$) of sensitivity on hit rate for the data in Fig. 8, pairwise t-testing reveals differences of the means to be significant only when comparing the highest sensitivity setting (7.2°/mm or 5 cm/360°) to the remainder of sensitivities ($p = 0.0146$). Additionally Fig. 9 demonstrates that hit rate was not significantly correlated with task completion time in our experiment. This makes sense as we expect our subjects always attempted valid shots and thus trials where the shot was missed did not incur substantially different task completion times than those in which the shot was hit. The only notable exception to this trend was the single high sensitivity teal marker from User 8, which does demonstrate some correlation of a lower completion time with lower hit rate. We did not choose to discard this data, as though the user did appear to be impacted by the high sensitivity, very few of their trials failed to meet the 50% displacement criteria for rejection.

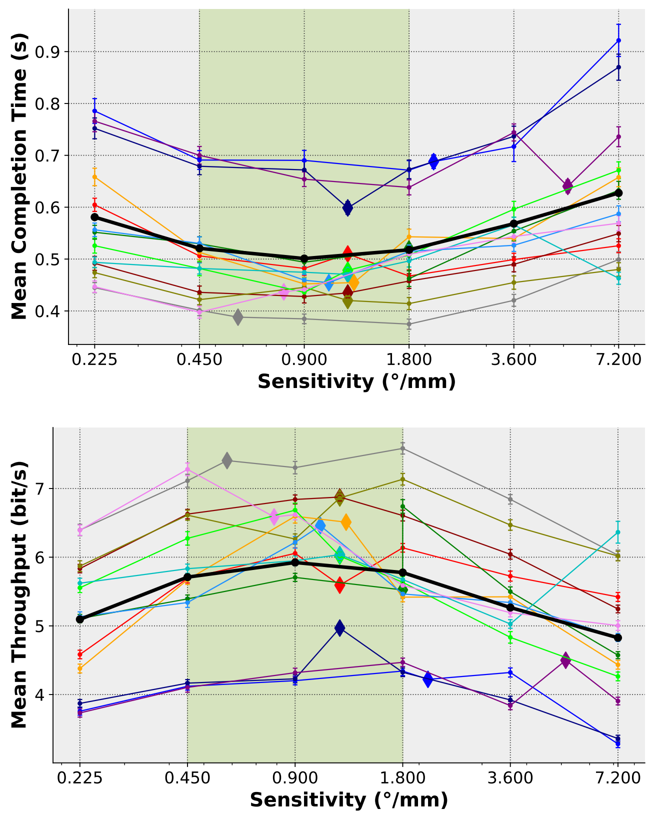

Fig. 10. Mean task completion time (top) and throughput (bottom) per user (colors) and overall (thicker black line) versus sensitivity. Per-user preferred sensitivity conditions are indicated by the diamond markers. Error bars represent the standard error metric. The X-axis is logarithmic and the green shaded region represents the zone of optimal performance.

B. Task Completion Time and Throughput

Within and between subjects we observed significant effects of mouse sensitivity on task completion time and throughput. Mean task completion time and throughput versus sensitivity are plotted per user (one per color) and overall (thicker black line) in Fig. 10. Each user's preferred sensitivity is indicated by a diamond marker, with many of the users selecting an overlapping setting of 1.2°/mm (30 cm/360°).

ANOVA indicates a significant main effect of sensitivity on task completion time ($F(5, 19955) = 111.72$, $p < 0.0001$), with pairwise t-testing validating these differences ($p < 0.001$) for all adjacent sensitivity pairings except those in the 0.45-1.8 °/mm (20-80 cm/360°) range. This zone of optimal sensitivity mirrors the 4$\times$ multiplicative range of optimal mouse sensitivities (CD gains) reported for pointer-based tasks in prior art.

In addition to completion time, we analyze mean task throughput (see Eq. 2) per individual and over all subjects based on our experiment data. This metric decorrelates the spatial difficulty of a task from the completion time measured for that task by taking the ratio of the index of difficulty (in bits) to the completion time, accounting for the non-uniform ID distribution in Fig. 7. ANOVA confirms a significant main effect of mouse sensitivity on throughput ($F(5, 19955) = 275.59$, $p < 0.0001$). With additional pair-wise t-testing between adjacent sensitivities confirming significant differences

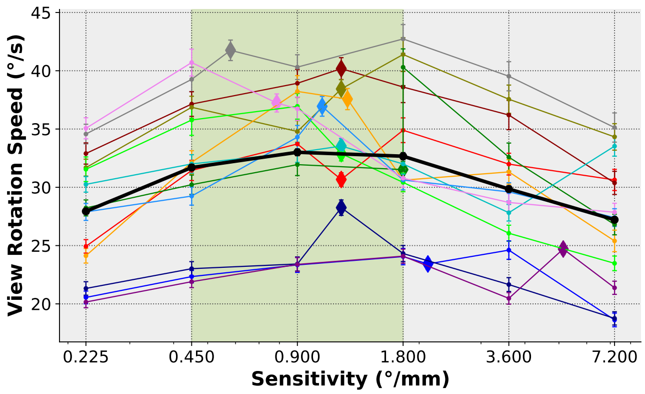

Fig. 11. The average speed (target distance divided by total task time) per player (color) and overall, shown as the thick black line. Diamond markers represent preferred sensitivity conditions. The X-axis is logarithmic and the green shaded region represents the zone of optimal performance.

of mean ($p < 0.001$) for all tested sensitivities. The similar, but inverted, shape of curves in Fig 10 and identical range of significant sensitivity differences between throughput and task completion time imply that differences in index of difficulty did not impact our measured results.

Interestingly, many subjects performed better at a pre-selected sensitivity than at their preferred sensitivity setting, both in completion time and task throughput, though generally speaking this preferred sensitivity was within their range of optimal performance. All users performed optimally in mean task completion time and throughput in the range of 0.45-1.8°/mm (20-80 cm/360°). Additionally, it is worth noting that the 2 users who selected preferred sensitivities outside of this optimal range (plotted in blue and purple in Fig. 10) were 2/3 of the worst performers in our experiment. This may indicate that, though they were regular gamers, they were not as familiar with the type of aiming tasks we provided in our experiment. One user (shown in dark green) selected a preferred sensitivity setting that overlapped with a selected tested sensitivity (1.8°/mm or 20 cm/360°). This accounts for the jump in the mean task completion time and throughput for this user at the setting. This discontinuity likely occurred due to fatigue impacting the session that overlapped with their preferred sensitivity conditions (the final session of that day for the user), but not the preferred sensitivity sessions themselves, as these were completed on other days.

C. Average Movement Speed

We analyzed our subjects' movement speed within each trial to assess whether increased mouse sensitivity resulted in increased average speed of view rotation in the application, an observation indicating human kinematics are limiting view rotation. We performed this analysis by dividing the (angular) displacement of the target, or distance in the Fitts' law formulation, by the task completion time to produce an average angular velocity for the trial. Our results are plotted in Fig. 11.

The trend in this data strongly mirrors that of average throughput above, which makes sense as their formulation

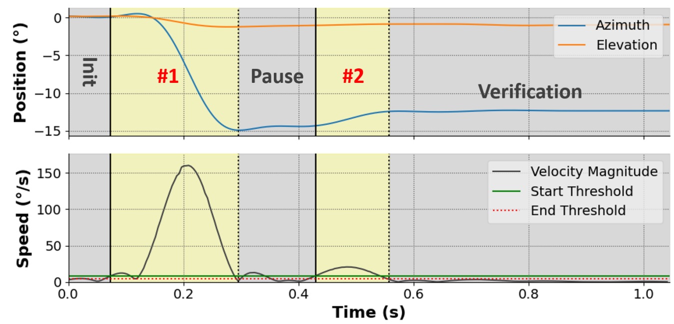

Fig. 12. An example of a partitioned set of submovements with initialization, pause, and verification periods (shown in gray), together with 2 distinct submovements (shown in yellow). The bottom plot shows the velocity vector magnitude of the azimuth/elevation time series reported in the top plot along with thresholds for detection.

is similar. The degradation of rotational view velocity at low sensitivity kinetically makes sense, at the lowest tested sensitivities users were not able to move the mouse quickly enough to maintain their view direction velocity. However, at the highest tested sensitivities this should not be the case. We suggest this velocity degradation at high sensitivities comes from time spent verifying the mouse was over the target and making corrective motions if required, and thus may not represent an actual reduction subject motion velocity.

D. Submovement Analysis

The previous analysis in this section, specifically that of task completion time and throughput, begs an interesting kinematic question. What was the effect of sensitivity on physical user motion? Did users exploit higher sensitivities to turn faster in-game at the same real-world motion velocity, and if so, why does performance often degrade more at the highest sensitivities than the lowest ones? To answer these questions we turn to submovement analysis.

Submovements are short, (pseudo)ballistic motion trajectories that arise from the human motor response planning process [28], [29], [30], [31]. We employ a submovement segmentation technique inspired by that of Meyer [28]. We extend this work by considering periods of initialization, pauses, and verification [29] and online regulation of movements [30], [31].

Algorithm 1: Submovement Parsing Algorithm

Data: $time$, azimuth (az), elevation (el) series
Result: list of submovement start/end times ($move_{start}, move_{end}$)

$az_{filt}, el_{filt} = filter_{LP}(time, az, el)$;
$v = |(\delta(az_{filt}), \delta(el_{filt}))|/\delta(time)$;
for *t in time* **do**
  if *previously in a submovement* **then**
    if $v[t] < V_{end}$ ***and*** $t - t_{start} > T_{min}$ **then**
      $move_{end}.add(t)$;
  else if $v[t] > V_{start}$ **then**
    $move_{start}.add(t)$; $t_{start} = t$;
end

Algorithm 1 outlines the approach used for parsing submovements for this work. A 5th order, 7 Hz low pass Butterworth filter is used to smooth the azimuth/elevation data prior to submovement processing. Initialization, pauses, and verification periods are distinguished from periods of motion using the start velocity threshold (V_{start}). Multiple submovements detected within a single psychological refractory period (T_{min}) are considered one submovement. A submovement ends whenever the minimum duration of a submovement (T_{min}) has elapsed and the combined angular velocity falls below the end threshold (V_{end}). For our analysis we select $V_{start} = 8°/\text{s}$, $V_{end} = 4°/\text{s}$, and $T_{min} = 80\text{ms}$. These values are selected empirically based on manual validation of the segmentation they produced, with T_{min} approximating lower bound estimates of the psychological refractory period. An example of submovements segmented from a single trial's azimuth/elevation aim time series using these values is provided in Fig. 12, demonstrating an initialization, pause, and verification window together with 2 segmented submovements.

Using this submovement segmentation algorithm we can fully partition the time a user spends in any given trial into 4 different classes. *Initialization* time, or the time before the first submovement begins, *movement* time (not to be confused with the movement time predicted by Fitts' law, which is more similar to task completion time) or the total time spent in active movements, *pause* time or the total time spent pausing between movements, and *verification* time, the time spent verifying the reticle is above a target prior to making a click.

Fig. 13 provides averages of per-user and overall time spent in each of these phases and confirms what we suspect drives degradation of overall average speed in Fig. 11. Namely that increasing trends in initialization, pause, and verification times across sensitivity make high sensitivities appear to negatively impact view rotation velocity without actually impacting movement time. Initialization time is mostly flat as expected, as this represents the time from destroying the reference target to beginning to aim at the task target, a time which is significantly reduced by providing a task preview with the reference target. Pause time demonstrates the strongest increasing trend, but does so by summing the duration(s) of all pauses, the number of which is strongly dependent on submovement counts for the trial. For example, trials with a single submovement always report a pause time of 0 s, as only initialization and verification times were reported. Verification time shows a stronger increase over mouse sensitivity than initialization time, an interesting result. We hypothesize this is due to participants understanding the greater risk of undershoot or overshoot when using high sensitivities, and implicitly devoting more time to verifying alignment in these conditions. Notably we can potentially account for some of user 8's strong decrease in hit rate in the 7.2°/mm condition with their strong decrease in verification time at that condition, while other users largely affirmed the trend of increasing verification time with mouse sensitivity.

By only analyzing the first submovement of the series (typically the largest motion) we can better separate slower performance due to long initialization, pause, or verification windows from slower average player motion. This analysis

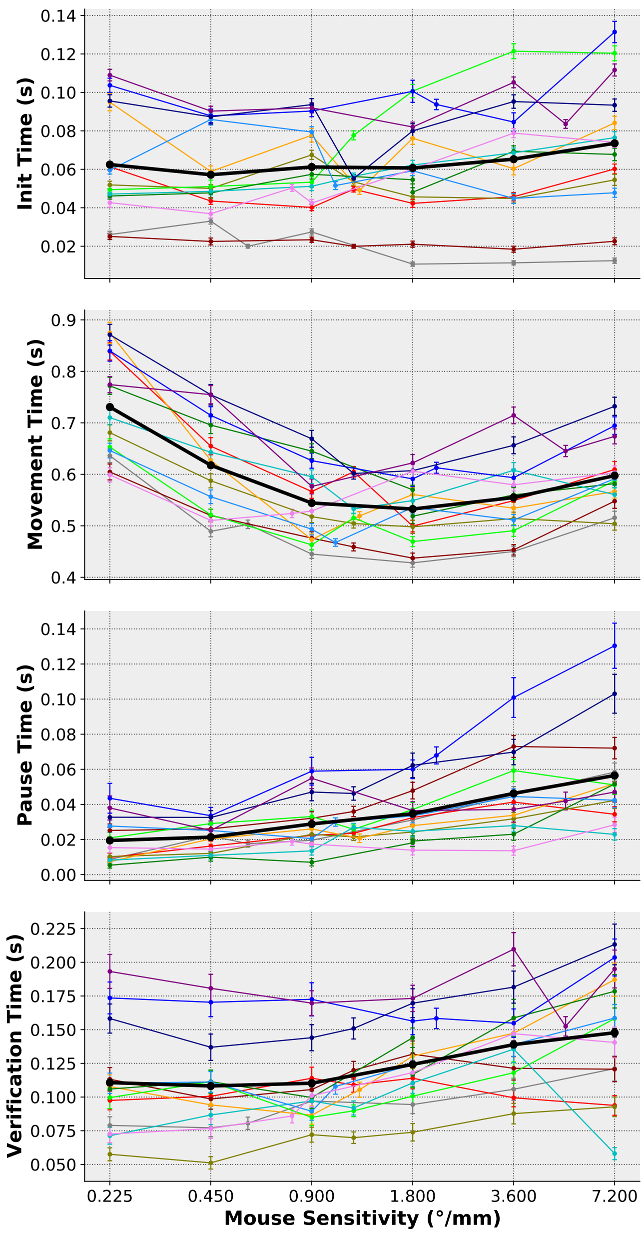

Fig. 13. Breakdown of per user (colors) and overall (thicker black line) of (top) initialization, (2nd from top) movement, (3rd from top) pause, and (bottom) verification time. Error bars represent standard error. Note that initialization, pause, and verification times all show a positive correlation with mouse sensitivity, though slopes vary significantly.

demonstrates that for the lower, and in some cases the highest, tested sensitivities, players rotated their view more slowly in game. However, the degradation of movement speed at high sensitivities is far less than that of the high sensitivities in Fig. 11. Instead, the average number of submovements monotonically increases with sensitivity (Fig. 14 center plot). This indicates that targeting motion precision degrades with mouse sensitivity, requiring more corrective submovements before successfully completing a targeting action, accounting for what appears to be a super-linear trend in pause times in Fig. 13. This increase in submovement count drives an increase in mean task completion time in turn, as demonstrated in the rightmost plot of Fig. 14. Therefore, the optimal region in task

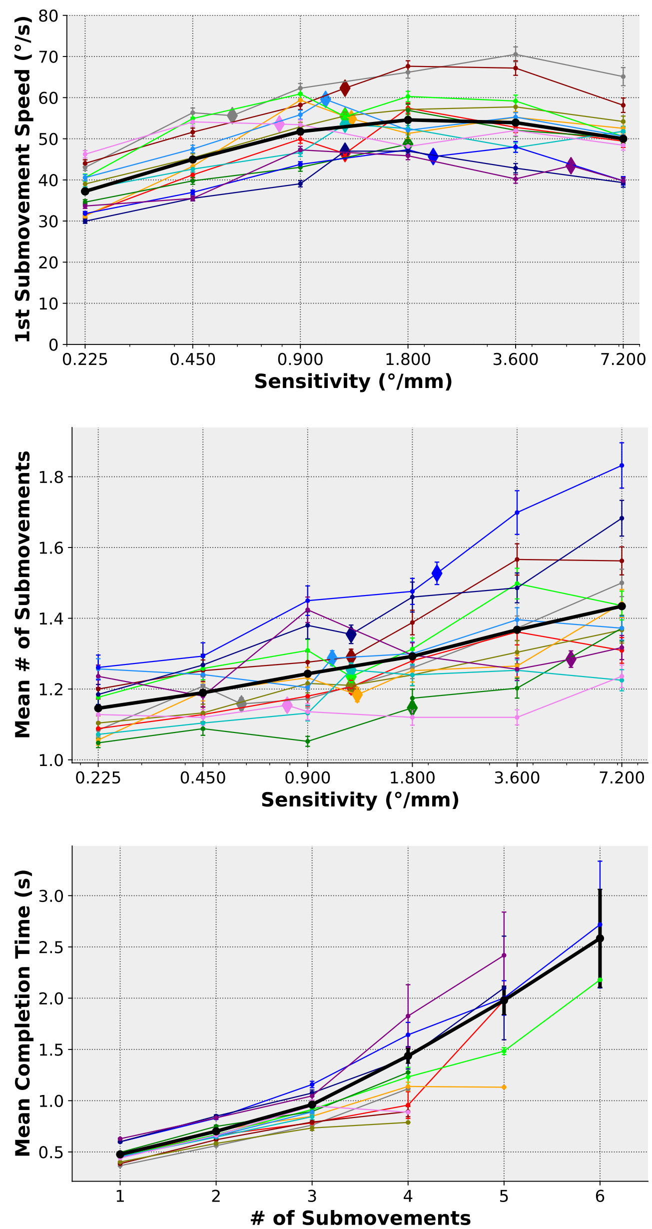

Fig. 14. (Top) average velocity of the first submovement versus mouse sensitivity, demonstrating that though high sensitivity did degrade overall task time, it did not as substantially effect first submovement velocity. (Middle) positive correlation between sensitivity and submovement count. (Bottom) super-linear correlation between submovement count and average task completion time. Note that since subjects were less likely to make higher numbers of submovements (SEM) error bars grow substantially with number of submovements.

completion time, throughput, and average view rotation speed in the plots in Fig. 10 are not solely the result of increased player motion speed, but rather the trade-off of movement speed impacts at low mouse sensitivity, and increased time for small, corrective movements at high sensitivities.

Additionally, we observe that spatial task difficulty (as measured by ID) is positively correlated with average number of submovements and that, across all IDs, the mean number of submovements increases monotonically with mouse sensitivity. This implies that task ID and mouse sensitivity drive not just the duration of submovements, but also their count, super-linearly impacting the average completion time of the task.

V. Discussion

Generally speaking, the self-normalized range of optimal suggested displacement-to-rotation mouse sensitivity for FPS aiming tasks in this work conforms to the range suggested by previous work in CD gain. The 4x range of near optimal sensitivities arrived at in our study matches the multiplicative range of optimality arrived at in prior art, but specifies this range in units of °/mm useful to FPS players. With this in mind we revisit our 3 experimental design questions.

A. *Experimental Questions*

Our experiment design asked three questions about competitive gamers' mouse sensitivity settings:

- Does an optimal sensitivity (range) exist for FPS targeting tasks?
- Are competitive FPS gamers preferred sensitivities well aligned with their optimal performance?
- Does (long term) muscle memory play a crucial role in aiming performance? If so, does playing with varied sensitivity significantly degrade performance?

Based on our measured and modeled results we answer these questions as follows.

a) Optimal Sensitivity Range: Our competitive FPS gamers demonstrated a preference for an overall optimal *range* of mouse sensitivity, a 4x window from 0.45-1.8°/mm or 20-80 cm/360°. Though the two cannot be directly converted, this 4x zone of near optimal sensitivity mirrors the range found by previous, pointer-based CD gain studies, suggesting a possible common basis for this range of near optimal performance. We conclude that FPS mouse sensitivity is driven by a speed-error relationship similar to optimal pointer-based CD gain, driving users away from lower mouse sensitivity in the name of speed and high mouse sensitivity to avoid large movement errors. However, whether the shape of this trend is consistent across different FPS aiming tasks is not addressed by our study. Our static targets likely have the most in common with pointer-based buttons in that they do not require tracking of target velocity as they do not move. However, it's possible even more interesting mouse sensitivity effects are present when targets are in motion, a common case for FPS aiming but relative rarity in pointer-based interactions.

b) Individually Optimal Sensitivity: The gamers we studied did not demonstrate significantly *better* choice of preferred sensitivity than generally optimal ones; however, all but two did use preferred sensitivities within the globally optimal region. The two users who selected sensitivities outside of (higher than) the optimal range represented 2 of the 3 poorest performers, but did demonstrate less degradation in performance near these choices than other users.

c) Long-term Effects of Sensitivity Changes: While short-term adaptation was demonstrated by a strong within-session training effect as demonstrated in Fig. 15 (controlled for by only analyzing the last 50% of trials from any given

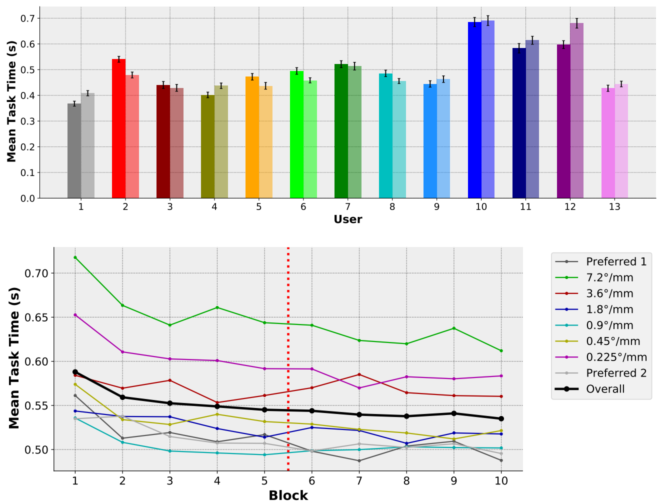

Fig. 15. (Top) comparison of average task completion time between the first (left bar) and final (right bar) session, both at the preferred sensitivity, by user. Error bars are standard error. Some users show statistically significant degradation of performance (increase from left to right bar), while other show no significant change, or even small significant improvement. (Bottom) per-session average completion time over blocks of 50 trials demonstrating the short-term training effect. The dashed vertical red line represents the threshold for discarding trials due to short-term training effects.

condition), longer-term (i.e., session-to-session) adaptation effects were not as pronounced in our data. On average subjects did not demonstrate a significant degradation of performance between their first and final (preferred) sensitivity sessions. Individually, about half of subjects demonstrate degradation of performance, or an increase in mean task time, from the first trial to the last, but nearly as many showed *improvement* between these same sessions. Generally, this degradation/improvement was small in magnitude ($<$35% of the standard deviation of completion time, and $<$15% of average completion time) and statistically insignificant. Interestingly the subjects with the highest and lowest average completion times, or the least and most skilled users, tended to experience degradation, while the intermediate performers were more likely to see an improvement between their first and final session. This may imply that the most skilled subjects did experience a performance detriment from changing sensitivities, while intermediate players benefited more generally from time spent in the experimentation platform. We do not interpret the lowest skill players results as highly significant as 2/3 of these players were the only two to select habitual sensitivities outside of our identified optimal range. Overall we conclude that the wide (32$\times$) range of sensitivities experienced by our subjects did not appear to either consistently or significantly impact their performance between their first and final (preferred) sensitivity sessions.

B. Mouse Sensitivity Selection

When choosing a mouse sensitivity for FPS gaming, identifying a range of typical spatial task difficulty and co-optimizing for desired view velocities are necessary first steps. This entails identifying the common range(s) of (angular) target size and distance (and thus their corresponding IDs) then selecting a mouse sensitivity that balances the trade-off of rotational velocity to number of submovements (i.e. chance of over/undershoot) for this sensitivity. From the user's perspective this means trading off mouse sensitivity high enough to avoid limiting in-game movement speed, but low enough to reduce the probability of over/undershooting targets. In our study we attempted to cover a wide but representative range of IDs common in FPS gaming (see Fig. 7), but in reality individual gamers may choose to further tailor sensitivity to their role (i.e. tasks) in-game.

As an example consider two roles common within a single competitive FPS game: a player who interacts with a target at short range and a player who interacts with the same (sized) target at a much greater (in-game) distance, with the target distributed over a similar range of view eccentricity in both cases. Since the target remains the same size in virtual world-space but is presented at different depths, the first player experiences a larger target size relative to the aim displacement required to hit it, while the second player must select a smaller target at a similar aim displacement. This means the first player experiences tasks with lower ID, while the second player experiences tasks with consistently higher ID. This is demonstrated visually in Fig. 16.

The player interacting with low ID (nearby) targets should select a higher mouse sensitivity, allowing them to rotate the view more quickly towards these targets, and make rapid changes in view direction to track nearby targets. This player's risk of under/overshooting is greatly reduced by the increased target size they experience. The player experiencing higher ID (more distant) targets stands to benefit from lower mouse sensitivity, as since the need to make sudden, dramatic changes in view direction is lower this player benefits more from the increased precision of lower sensitivity. We demonstrate this claim by plotting our task completion time and throughput data broken out by range of IDs in Fig. 17. The results demonstrate subjects performed better at lower sensitivities for high ID tasks, and higher sensitivities for low ID tasks, in accordance with our suggestion and prior art. Additionally, we find the the *sharpness* of the completion time/throughput vs sensitivity curve generally increases at higher difficulty. This implies that task ID drives not just the optimal mouse sensitivity value, but also the penalty a user pays for selecting a sub-optimal sensitivity. This conforms to our intuition as a smaller target (higher ID) requires more precise alignment of the reticle and tends to result in greater numbers of additional submovements to properly align the target.

Additionally, if these targets move at equal virtual world-space velocities, as is common, the larger target moves faster when projected to screen space, requiring faster view velocities to track it. This is caused by the same phenomenon as the change in ID, wherein the displacement of a nearby object projected to screen-space is much greater than that of a distant object over the same period of time. This creates a two-fold motivation for higher sensitivities when aiming at nearby targets. First task IDs shift towards lower difficulty, prompting faster movements, easier at higher sensitivities. Second target motion increases in velocity, requiring faster view rotation when attempting to track a target.

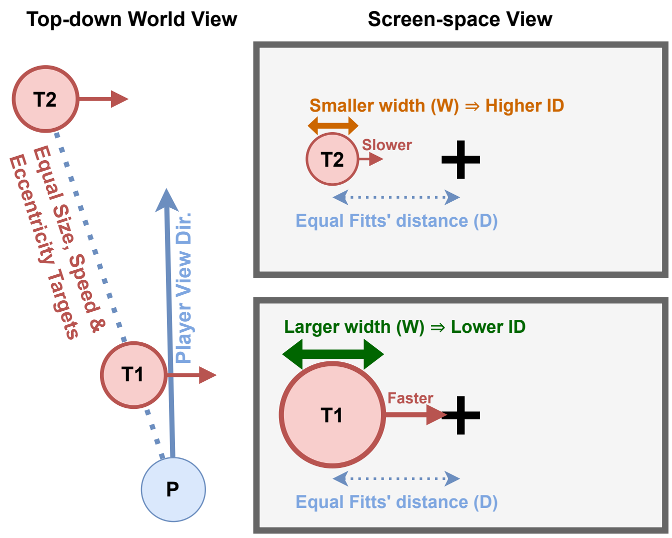

Fig. 16. A visualization of a player (P) interacting with a nearby (T1) and distant (T2) target, and their projections to screen-space. Note that though both targets have the same world-space size and speed the nearby target appears as larger (lower ID) and faster moving, while the distant target appears as smaller (higher ID) and slower moving.

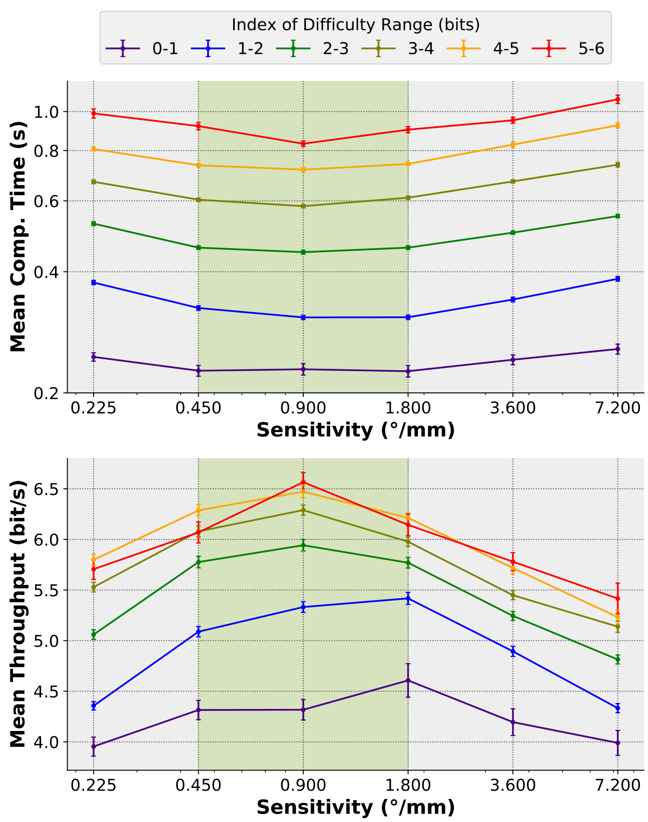

Fig. 17. Average task completion time (top) and throughput (bottom) for all participants plotted by various ID ranges (colors). A trend towards higher sensitivity at lower spatial task difficulty is observable. The mean completion time is plotted with log X and Y axes. Throughput is plotted with a log X axis. Error bars represent standard error. Our previously identified optimal range is again shaded in green.

Exceptions to this principle arise when display-space boundaries (such as display borders) are used to "stop" mouse travel at some extent. This example non-linearity is utilized in many OS-level interfaces (e.g., the start menu button) which allows easier selection of a small but distant (high ID) target by limiting pointer travel in the corner of the display. Such a limitation applies in FPS aiming as the view elevation (vertical-displacement) angle is often limited to $\pm 90°$. This implies targets located directly above or below the player view position may be substantially easier to target even if they have a high ID.

C. Limitations and Future Work

The range of target widths and distances studied in this work represents a reasonable sampling of ID, but a subset of targeting tasks presented in real FPS games. Studying the impacts of sensitivity in large angular rotations (i.e. 180° turns in game) may present an added benefit to lower mouse sensitivities not represented in this work. This may be more relevant to some game conditions than others.

This study does not include any moving targets, a common condition in real FPS gameplay. It is somewhat obvious that the need to track a sufficiently quickly moving target will also have impacts on optimal mouse sensitivity. We suggest that a future study on the effects of moving target speed on mouse sensitivity might demonstrate additional interesting relationships historically absent from CD gain based research.

In addition, many games implement separate sensitivities for "scoped" or "zoomed" modes that modify perception and sensitivity of an aiming task without impacting task ID, particularly when trying to hit small targets at reasonable distances. Typically this scoped view modifies both rendered field of view and the °/mm sensitivity, with some games allowing players to control this scoped sensitivity setting independently effectively enabling a 2-tier sensitivity model. Our study did not include a scoped view mode, but a future study could include this dynamic and study how the ability to select between two different target sizes/difficulties changes the optimal behavior, at the cost of field of view while zoomed in.

Additionally, some players further modify sensitivity by dynamically changing mouse DPI with hardware supported buttons, linearly scaling °/mm mouse sensitivity. This enables something similar to scoped sensitivity without modifying the rendered field of view or requiring use of a weapon that supports a scope. While we are aware that this option is widely supported in gaming mice, we are now aware of how pervasive its use is in professional esports gameplay.

Typically gamers learn the bounds of their mouse pad or surface and lift (or clutch) the mouse prior to reaching these limits, creating additional non-linearity in task completion time data. The mouse motion studied in this work was controlled to be limited to a range of 110 mm (the maximum target distance at the lowest sensitivity corresponds to 111.8 mm of mouse travel) to discourage subjects from lifting the mouse during data collection. However, gaming mouse pads can be as large as 400-500 mm [32]. Future work exploring larger target distances (i.e. user motion >500 mm) would encourage more lifting actions, improving understanding of how their speed and duration is related to target task difficulty and sensitivity. We expect lifting actions will significantly increase pause durations in submovement decomposition and further penalize multi-submovement trials in task completion time.

VI. Conclusions

We extend results from the first performance-oriented study of mouse sensitivity in FPS targeting tasks [1], further validating and explaining a range of broadly optimal FPS mouse sensitivities from 0.45-1.8°/mm. This range agrees well with the 4x range of CD gain reported by prior art in pointer-based mouse sensitivity literature, and includes the preferred sensitivities selected by most of our experienced FPS gamer subjects. We demonstrate that sufficiently low sensitivities reduce view rotation speed, linearly increasing task completion time, while sufficiently high sensitivities or difficult tasks result in increased count of corrective motions, super-linearly impacting overall task completion time. We validate these results by segmenting trial submovements to confirm that trends in overall view velocity can be attributed both to motion and non-motion periods of the view direction time series. The resulting optimal sensitivity range trades off speed of view velocity to submovement count, and is likely to change based on the type and difficulty of task(s) users commonly perform. We demonstrate that easier (lower ID) spatial tasks benefit from higher sensitivities, while more spatially difficult (higher ID) tasks benefit from lower sensitivities, conforming to traditional pointer-based results. Finally we indicate how the coupled increased screen-space velocity of nearby targets compounds this effect by requiring players to track larger targets moving at higher speeds. We do not observe long-term "hard wired" muscle memory of mouse sensitivity for all players, and based on this we encourage FPS gamers to experiment with changing mouse sensitivity to continue to optimize their setting(s) to their role and individual play style.

VII. Acknowledgements

The authors acknowledge Ben Watson and Rachel Brown for comments on the ideas behind this work. Thanks to Justin Thomas for administering the user study.

References

[1] B. Boudaoud, J. Spjut, and J. Kim, "Mouse sensitivity effects in first-person targeting tasks," in *2021 IEEE Conference on Games (CoG)*. IEEE, 2022.

[2] /u/StruthGaming. (2019, Feb) Your aim isn't terrible, your mouse sensitivity is (pc). Reddit. [Online]. Available: https://www.reddit.com/r/apexlegends/comments/apbo8z/your_aim_isnt_terrible_your_mouse_sensitivity_is/

[3] Y. H. Pang, E. R. Hoffmann, and R. S. Goonetilleke, "Effects of gain and index of difficulty on mouse movement time and fitts' law," *IEEE Transactions on Human-Machine Systems*, vol. 49, no. 6, pp. 684–691, 2019.

[4] M. Bohan, S. Thompson, D. Scarlett, and A. Chaparro, "Gain and target size effects on cursor-positioning time with a mouse," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 47, no. 4, SAGE Publications Sage CA: Los Angeles, CA. Los Angeles, CA: SAGE Publications, 2003, pp. 737–740.

[5] G. Casiez, D. Vogel, R. Balakrishnan, and A. Cockburn, "The impact of control-display gain on user performance in pointing tasks," *Human–computer interaction*, vol. 23, no. 3, pp. 215–250, 2008.

[6] H. D. Jellinek and S. K. Card, "Powermice and user performance," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '90. New York, NY, USA: Association for Computing Machinery, 1990, p. 213–220. [Online]. Available: https://doi.org/10.1145/97243.97276

[7] J. Sandfeld and B. R. Jensen, "Effect of computer mouse gain and visual demand on mouse clicking performance and muscle activation in a young and elderly group of experienced computer users," *Applied Ergonomics*, vol. 36, no. 5, pp. 547–555, 2005.

[8] M. L. Lin, R. G. Radwin, and G. C. Vanderheiden, "Gain effects on performance using a head-controlled computer input device," *Ergonomics*, vol. 35, no. 2, pp. 159–175, 1992.

[9] J. Looser, A. Cockburn, and J. Savage, "On the validity of using first-person shooters for fitts' law studies," *People and Computers XIX*, vol. 2, pp. 33–36, 2005.

[10] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement." *Journal of experimental psychology*, vol. 47, no. 6, p. 381, 1954.

[11] P. M. Fitts and J. R. Peterson, "Information capacity of discrete motor responses." *Journal of experimental psychology*, vol. 67, no. 2, p. 103, 1964.

[12] I. S. MacKenzie and W. Buxton, "Extending fitts' law to two-dimensional tasks," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '92. New York, NY, USA: Association for Computing Machinery, 1992, p. 219–226. [Online]. Available: https://doi.org/10.1145/142750.142794

[13] I. S. MacKenzie, "Fitts' law as a research and design tool in human-computer interaction," *Human-computer interaction*, vol. 7, no. 1, pp. 91–139, 1992.

[14] I. S. MacKenzie and P. Isokoski, "Fitts' throughput and the speed-accuracy tradeoff," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2008, pp. 1633–1636.

[15] U. Tränkle and D. Deutschmann, "Factors influencing speed and precision of cursor positioning using a mouse," *Ergonomics*, vol. 34, no. 2, pp. 161–174, 1991.

[16] R. Senanayake and R. S. Goonetilleke, "Pointing device performance in steering tasks," *Perceptual and motor skills*, vol. 122, no. 3, pp. 886–910, 2016.

[17] G. Casiez and N. Roussel, "No more bricolage! methods and tools to characterize, replicate and compare pointing transfer functions," in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, 2011, pp. 603–614.

[18] O. Green, "What is the best dpi for fps games?" May 2022. [Online]. Available: https://aimtrainer.com/best-dpi-for-fps-games/

[19] R. Balakrishnan, ""beating" fitts' law: virtual enhancements for pointing facilitation," *International Journal of Human-Computer Studies*, vol. 61, no. 6, pp. 857–874, 2004.

[20] Max, "What is mouse acceleration?" Nov 2019. [Online]. Available: https://prosettings.net/library/what-is-mouse-acceleration/

[21] "What is mouse acceleration?" Sep 2019. [Online]. Available: https://www.bestsettings.com/library/what-is-mouse-acceleration/

[22] F. Beaufort, "Disable mouse acceleration to provide a better fps gaming experience," Nov 2020. [Online]. Available: https://web.dev/disable-mouse-acceleration/

[23] M. Mamerow, "Is mouse acceleration good for gaming? [aiming myth uncovered]," Jun 2020. [Online]. Available: https://raiseyourskillz.com/is-mouse-acceleration-good-or-bad-for-first-person-shooters/

[24] N. Johnson, "This is why you might want to use mouse acceleration in csgo," Mar 2021. [Online]. Available: https://win.gg/news/7539/this-is-why-you-might-want-to-use-mouse-acceleration-in-csgo

[25] J. A. Schaab, R. G. Radwin, G. C. Vanderheiden, and P. K. Hansen, "A comparison of two control-display gain measures for head-controlled computer input devices," *Human Factors*, vol. 38, no. 3, pp. 390–403, 1996.

[26] J. Spjut, B. Boudaoud, K. Binaee, A. Majercik, M. McGuire, and J. Kim, "Firstpersonscience: Quantifying psychophysics for first person shooter tasks," in *UCI Esports Conference*, 2019.

[27] B. Boudaoud, J. Spjut, and J. Kim, "Firstpersonscience: An open source tool for studying fps esports aiming," in *ACM SIGGRAPH 2022 Talks*, 2022, pp. 1–2.

[28] D. E. Meyer, R. A. Abrams, S. Kornblum, C. E. Wright, and J. Keith Smith, "Optimality in human motor performance: ideal control of rapid aimed movements." *Psychological review*, vol. 95, no. 3, p. 340, 1988.

[29] Y. Chen, E. R. Hoffmann, and R. S. Goonetilleke, "Structure of hand/mouse movements," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 6, pp. 790–798, 2015.

[30] R. Chua and D. Elliott, "Visual regulation of manual aiming," *Human Movement Science*, vol. 12, no. 4, pp. 365–401, 1993.

[31] T.-Y. Hsieh, Y.-T. Liu, and K. M. Newell, "Submovement control processes in discrete aiming as a function of space-time constraints," *Plos one*, vol. 12, no. 12, p. e0189328, 2017.

[32] prosettings.net. (2021) Best gaming mousepad - the ultimate guide. Accessed: 2021-02-11. [Online]. Available: https://prosettings.net/best-gaming-mousepad/