Modeling Throwing Selection in Virtual Reality

Jae-Yeop Jeong, Wolfgang Stuerzlinger, and Jin-Woo Jeong

Abstract— Throwing is a natural and embodied interaction for selecting distant targets in virtual reality (VR). Unlike conventional pointing, throwing is ballistic: once released, the trajectory can no longer be corrected. Despite its practical importance, throwing-based target selection has received relatively little attention in human behavior and performance modeling, making systematic design and evaluation still difficult. In this paper, we investigate throwing-based target selection in VR and propose predictive models for its behavior and performance. Through a controlled user study manipulating target width (W) and movement amplitude (A), we measured planning time (PT), operationalized as the pre-release interval between grabbing and releasing the ball, and endpoint distribution. Our results showed that PT increased systematically with task difficulty, following a pattern similar to movement time in Fitts' law. Throwing endpoints were also generally well approximated by a bivariate Gaussian distribution, and target parameters significantly affected both endpoint bias and variability. Based on these findings, we derived a probabilistic accuracy model that predicts target acquisition performance. The proposed models showed high goodness-of-fit, and the accuracy model was robust under leave-one-condition-out cross-validation. Our findings provide a quantitative understanding of throwing-based target selection within the tested VR setup and support evidence-based design of throwing interactions in immersive environments.

Index Terms—Throwing, Human Performance Modeling, Target Selection, Virtual Reality

1 Introduction

Computational models of human motor behavior support the design and optimization of interactive systems [37]. Many efforts have sought to quantify user behavior and performance, particularly in classical graphical user interfaces (GUIs) [1–3,7,16,26,30,39,55]. These models focus on fundamental operations such as pointing, crossing, and steering, as these underlie more advanced interaction techniques, for example, text entry [28,40,62] and lassoing [44,54,56]. Beyond traditional screen-based computing environments (e.g., desktop and smartphone), recent research has increasingly focused on three-dimensional (3D) environments, particularly virtual reality (VR) [4,29,46,58,63]. VR supports interaction techniques that leverage users' direct motor actions [57]. Unlike traditional cursor-based input (e.g., a computer mouse), these interactions depend on users' physical movements and motor dynamics. In practice, users typically employ handheld controllers or bare-hand tracking and move their arms and hands to select and manipulate virtual objects [33]. Consequently, significant motor noise and additional factors in VR, such as the design of the ray casting technique [6,25] and the vergence-accommodation conflict [5], might be reflected in interaction outcomes; in turn, existing computational models and their theoretical background have been examined and validated in such environments [4,29,49]. Thus, a critical next step is to examine their generalizability to newly enabled interaction techniques.

In this context, our study focuses on the throwing operation in VR, which is utilized in various applications, including games, sports, military training, and rehabilitation [12,17,24,32,43,64]. There, throwing is employed as a target selection mechanism, and successful interaction depends on accurately delivering a virtual object (e.g., a dart) to a specified target (e.g., a scoreboard). Existing motor behavior studies in target selection have primarily focused on two dependent variables (DVs): completion time and endpoint distribution. Completion time reflects task efficiency and plays a central role in evaluating usability in target selection tasks. The predominant model is Fitts' law, which explains movement time (MT) as a function of target width (W) and amplitude (A) [16]. In throwing, however, a corrective phase (e.g., intermittent cursor movements [39]) is impossible after release. Consequently, movement time cannot be defined in the same manner as in pointing selection, where cursor-based movements typically involve iterative corrective adjustments [13,14]. Instead, temporal performance in throwing is primarily determined by preplanning an accurate release [45]. We introduce planning time (PT) as a novel performance metric to quantify the cognitive and motor preparation required for accurate release; see Section 3.1 for details.

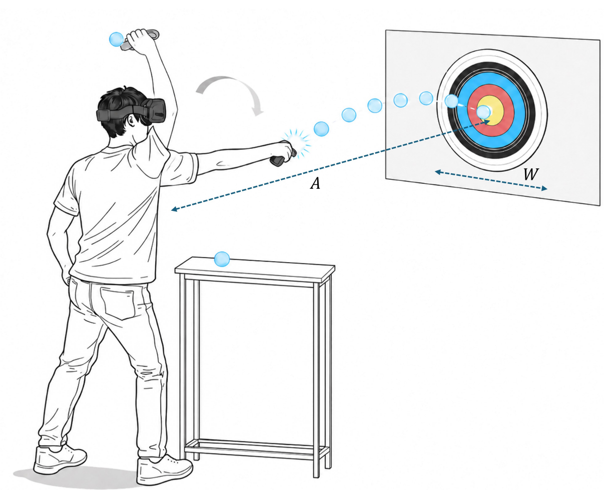

Figure 1: Illustration of our task scenario: throwing selection. A user is asked to pick up and throw a ball toward a target (i.e., a scoreboard). The ball is spawned on the table next to the user. W and A denote the target width and the distance between the user and the target, respectively. User behavior and performance change depending on the parameters of the target.

Throwing-based target selection necessarily produces an endpoint due to the final collision. The endpoint distribution captures the underlying structure of selection errors and motor behavior [58]. Given an endpoint distribution, selection accuracy can be predicted from target parameters, which is closely related to interaction effectiveness [22,50]. Endpoint distribution modeling has increasingly been adopted to uncover user behavior relative to time performance, particularly in immersive and spatial interaction contexts [48,58,63]. However, although endpoint distribution models have been established for conventional pointing selection, comparable probabilistic formulations for throwing

• Jae-Yeop Jeong and Jin-Woo Jeong (corresponding author) are with Seoul National University of Science and Technology. E-mail: {jaey.jeong, jinw.jeong}@seoultech.ac.kr.
• Wolfgang Stuerzlinger is with Simon Fraser University. E-mail: w.s@sfu.ca.

Manuscript received xx xxx. 202x; accepted xx xxx. 202x. Date of Publication xx xxx. 202x; date of current version xx xxx. 202x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.202x.xxxxxxx

selection remain largely unexplored. Most throwing research in VR has explored throwing performance and experience comparing real-world vs. VR [27,64], the point of release [17,52], and the effects of visual cues [53]. Without an explicit probabilistic model of throwing, designers and practitioners lack principled guidance for determining target size, distance, and difficulty in VR applications that employ throwing interactions. This gap is critical, as real-world user communities (e.g., Reddit) have raised recurring issues around throwing interactions in VR, highlighting the practical demand for design guidance [17,64].

To address this gap, this study establishes a predictive model for throwing-based target selection in VR. We examine throwing as a pre-release preparation-oriented selection task by introducing *PT* as an observable task-relevant temporal measure. We empirically develop a probabilistic model of the throwing endpoint distribution based on target parameters. This enables the prediction of selection accuracy, providing scientific evidence for the design and evaluation of throwing interactions in immersive environments. Our contributions are threefold:

- Characterization of user behavior in throwing-based target selection using *PT*, endpoint visualization, and ball trajectories.
- Identification of a significant relationship between *PT* and *ID*.
- Development of an endpoint distribution model and an accuracy prediction model for throwing-based selection.

2 Related Work

This paper covers throwing selection in VR and its modeling. In this section, we review the relevant literature on both topics.

2.1 Throwing Operation in VR

Throwing is a 3D embodied interaction paradigm that is difficult to realize in screen-based environments. VR systems, which support physical body movements as input, naturally enable throwing-based interaction [24]. Commercial platforms such as HTC VIVE and Meta provide SDKs that readily implement throwing mechanics in Unity3D [20,36], indicating its importance as a practical interaction technique. Throwing is widely adopted in VR applications. For example, users practice grenade throwing or baseball pitching in training scenarios [64]. Despite its prevalence, few studies have examined throwing as a target selection task. Prior work has primarily focused on performance and experience comparisons [17,27,52,53,64]. Real-world throwing has been shown to outperform VR in terms of accuracy, precision, and kinematic stability. To improve VR throwing performance, several studies investigated release mechanisms [17,52]. Wearable trackers (e.g., HTC VIVE trackers) achieved the highest accuracy, whereas handheld controllers with manual release were evaluated as more closely resembling natural throwing motions, with an average performance gap of approximately 12 centimeters (cm). Additionally, many works employed handheld controllers for hand tracking [10,41,64]. Another core design aspect is a visual cue. During throwing in VR, the availability of visual cues significantly affected throwing performance [53]. In their study, participants threw a virtual ball toward targets under two feedback conditions: Full, where the ball remained visible throughout its trajectory, and Minimal, where the ball became invisible immediately after release and only outcome feedback was provided. Their results showed that reduced visual feedback significantly increased throw error and decreased release velocity. Hand visibility also affected user performance [10].

Our study investigates user behavior in throwing-based target selection. Prior target-selection studies have shown that target parameters systematically affect user behavior and performance [16,48,58]. In contrast, previous VR throwing studies have primarily examined real-world comparisons, release mechanisms, and visual feedback rather than predictive modeling based on target parameters [17,53,64]. Therefore, empirical investigation and quantitative modeling are required.

2.2 Modeling Target Selection

Modeling efforts investigated what factors affect the performance of target selection. Two DVs are often employed: completion time (i.e., *MT* in Fitts' law) and endpoint distribution. For the pointing task, Fitts' law [16,42] can explain the relationship between *MT* and target parameters (i.e., the independent variables (IVs)):

$$MT = a + b\log_2(A/W + 1) \tag{1}$$

MT denotes movement time, and the logarithmic term represents the *ID* defined by *W* and *A*, the target width and task amplitude. Lowercase letters denote empirically determined regression coefficients. Across target conditions, *MT* follows the speed–accuracy trade-off, although the regression coefficients may vary depending on the task constraints [61]. Based on this theoretical background, we assume that *PT* also follows Fitts' law implications according to *ID*. In other words, accurate throwing requires users to adopt a posture that minimizes motor noise [14], which may demand substantial *PT*. Accordingly, higher *ID* values are expected to result in longer *PT*.

Another line of research has focused on modeling endpoint distributions and predicting error rates. If endpoints (x and y) are bivariate-Gaussian, then $(x,y) \sim \mathcal{N}(\mu, \Sigma)$ with

$$\mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \Sigma = \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix} \tag{2}$$

(μ_x, μ_y) and (σ_x, σ_y) denote the means and the standard deviations along the x and y axes, respectively. ρ is the correlation between the x and y axes. Wobbrock et al. [50] proposed an error-rate prediction model grounded in Fitts' law, later extending it to two-dimensional pointing tasks [51]. Their work demonstrated that the error rate can be predicted from target parameters across spatial dimensions. Several studies have directly modeled endpoint distributions to capture the spatial characteristics of target selection [7,21–23,31,38,39,48,59]. For example, Bi et al. [7] analyzed finger-touch input for small targets and showed that endpoint distributions deviate from classical assumptions, motivating refined distributional models. In VR contexts, Yu et al. [58] modeled endpoint distributions under varying task conditions and identified significant relationships between target parameters and distribution parameters. In conclusion, these studies indicated that both error rate and spatial endpoint distribution are systematically influenced by *W* and *A*, providing a principled foundation for predicting selection performance and informing interface design. Building on these foundations, we model throwing performance from two DVs: (1) *PT* as a temporal measure related to *ID*, and (2) endpoint distribution as a spatial measure reflecting accuracy.

3 Problem Space

Our central focus is to explore and model human behavior in throwing selection in VR. Depending on system specifications, such as VR platform, SDK, and throwing physics, there are numerous environments to throw an object toward a target. Additionally, accurate throwing in VR is known to be challenging [24,64]. To define our problem space, we used a Meta Quest 3 head-mounted display with its corresponding controllers since it is a widely used commercial VR device. The detailed implementation of this controller-based throwing setup is described in Section 4.2.

3.1 Hypotheses

We aim to (re-)analyze the differences between the most common selection method (i.e., pointing) and throwing selection. Both involve fundamentally different selection mechanics (see Figure 2). In conventional pointing tasks, movements can be explained by the *multiple process model* [13,14]. Users first perceive task parameters, *W* and *A*, and initiate a rapid movement toward the target, and then enter a corrective phase for accurate selection. Movements are continuously calibrated through feedback [39], resulting in a speed–accuracy trade-off [34]. In contrast, throwing does not allow corrective control after release. Once users perceive the task parameters, they must determine all movement configurations, such as release angle, force, and timing, during motor planning [45]. No correction is possible after release.

Here, we can derive the following hypotheses based on the difference between the pointing and throwing selection and the literature:

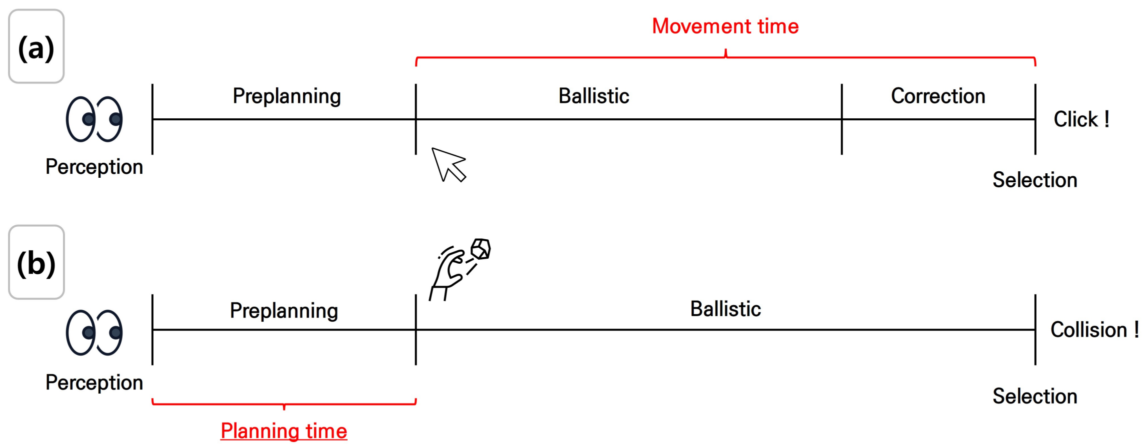

Figure 2: Illustration of the difference in the target selection procedure between pointing and throwing selection. In contrast to conventional pointing, which allows continuous feedback-based corrections during movement, throwing selection is a ballistic open-loop action in which all movement parameters (angle, force, and timing) must be determined during the preplanning phase before release. In this context, we assume the planning time acts like movement time in a conventional pointing task.

- **H1.** *Target parameters (W and A) significantly affect the planning time for throwing selection.* Because throwing performance depends on release parameters such as angle, force, and timing [45], we assume that more difficult target conditions may require users to spend additional time before release. Specifically, easier targets (short A and large W) allow faster planning, whereas more difficult target conditions require longer planning time, analogous to the role of MT in Fitts' law [35]. Furthermore, we identified performance differences (i.e., the rate of successful hits) in throwing in real-world settings between different IDs [60]. Accordingly, we hypothesize a systematic relationship between PT and ID in throwing-based selection.
- **H2.** *The endpoint distribution of throwing selection in VR is bivariate-Gaussian.* Previous research on 2-dimensional (2D) pointing has revealed that endpoint distributions follow a bivariate Gaussian across input modalities, including finger and gaze [7, 47, 48, 58]. Similarly, temporal pointing exhibits corresponding results in input timing [30, 59]. This commonality is primarily attributed to motor noise: variability in motor control leads to stochastic movement outcomes [19], producing Gaussian distributions for spatial and temporal dimensions. Throwing likewise depends on release parameters such as angle, velocity, and timing [45]. Based on the general effect of motor noise on movement outcomes, we hypothesize that variability in these release parameters produces an approximately Gaussian endpoint distribution in throwing selection.
- **H3.** *For throwing, the two axes (i.e., x and y axes in endpoint distribution) are correlated ($\rho \neq 0$).* In pointing selection, prior studies have reported negligible correlation between the two spatial axes (i.e., $\rho \approx 0$ in the bivariate Gaussian), across ray casting [58], touch pointing [9], and gaze input [48]. This independence arises because one task axis is typically dominant [58]. In contrast, throwing might show a different pattern. Release parameters jointly determine projectile motion in both horizontal and vertical directions, and a single release noise propagates into both axes simultaneously through projectile motion. The shared release event is characterized by angle, velocity, and timing; thus, a single release can simultaneously affect both spatial dimensions. Such release dynamics suggest that $\rho \neq 0$. We expect the magnitude of ρ to increase with A, as a longer travel distance amplifies the effect of release errors across both axes.
- **H4.** *The W and A parameters affect the endpoint distribution significantly.* In pointing selection, prior studies have examined the influence of parameters of the target on endpoint distribution. W has consistently shown a significant influence on this distribution [7, 58]. A, however, remains arguable [18, 22]. In throwing, we anticipate the following effects. A larger W reduces the need for precise release control, leading to increased endpoint variability. Similarly, a longer A amplifies release noise through projectile motion. Small errors in release angle or velocity propagate over a longer flight path, resulting in larger spatial deviations at collision. Accordingly, we hypothesize that both W and A significantly affect endpoint distribution in throwing-based selection.

4 EXPERIMENT

We conducted a study involving a throwing-based target selection task in VR, where participants threw a virtual ball toward a target positioned in front of them. The experiment was conducted with approval from the university's ethics board.

4.1 Participants and Apparatus

Sixteen paid participants were recruited: thirteen males and three females (mean age = 23.19 $\pm$ 2.40 years). All participants were right-handed and had normal or corrected-to-normal vision. None reported difficulty wearing the VR device or using the handheld controller. None had prior experience with throwing interactions in VR. All experiments were conducted with a Meta Quest 3 connected to a desktop PC equipped with an Intel Core i7-6900K, an NVIDIA GeForce RTX 4090, and 16 GB of RAM. Consistent with prior VR throwing studies [10, 41, 64], participants used the right-hand Meta Quest 3 controller. A 24-inch (1920 $\times$ 1080) commercial monitor was used for streaming the participants' point of view via Meta Horizon Link. The experimental application was implemented in Unity3D using C#.

4.2 Throwing Implementation in VR

Consistent with prior work on VR throwing [17, 64], we employed controller-based grab-and-release interaction. Participants used the grip button located on the side of the controller to grasp the virtual object (hereafter, the ball). They aimed while holding the button and initiated the throw by releasing it.

Our throwing implementation was based on the state-of-the-art throwing mechanic provided in the `Unity Interaction SDK` published by Meta (Meta XR version 83 [36]) at the time of the study. Specifically, we used the `ThrowTuner` component (`Oculus.Interaction.Throw`), which applies a configurable `ThrowPhysicsProfile` when an object is released. When the object is thrown, `Grabbable.VelocityThrow` triggers the `WhenThrown` event, providing an initial linear velocity and angular velocity, which are then applied to the rigidbody. We used the default velocity mapping in the profile (`Velocity Scale` = (1,1,1), `Velocity Add` = (0,0,0), `Max Speed` = -1), meaning that the release velocity from the SDK was applied without additional scaling, offset, or speed clamping. For rotation, `Spin Scale` was set to (1,1,1), `Spin Add` to (0,0,0), and `Max Spin` to 150. We did not enable built-in continuous flight forces (`EnableBuiltIns` = false); thus, no additional forces (e.g., modified gravity, constant force/torque, or velocity damping) were applied during flight beyond Unity's default physics. `Gravity scaling` was set to 1.0. We also disabled alignment options (`AlignForwardOnce` = false; `KeepForwardToVelocity` = false). Finally, we applied light drag during flight (`Linear Drag` = 0.008; `Angular Drag` = 0.008). For more details on reproducing the throw behavior, please refer to [36].

4.3 Experimental Design, Task, and Procedure

4.3.1 Experimental Design

To examine user behavior and performance in throwing selection, we designed an application based on previous VR target selection [8, 18, 48, 58] and throwing work [17, 27, 53, 64]. For throwing in VR, target widths are generally larger than those used in conventional pointing studies, and movement amplitudes are also extended due to the larger interaction space. Based on the parameter ranges reported in prior work (i.e., W ranging from 29 to 150 centimeters (cm) and A ranging from 110 to 1100 cm), we determined our experimental levels to cover comparable scales while maintaining systematic variation. Although our selected amplitudes did not cover the full range reported in prior work, we limited A to 300 cm to keep the required release velocity and physical effort within a comfortable range across repeated trials. We also included smaller target widths to create sufficiently challenging target-selection conditions rather than general throwing conditions. Accordingly, we selected four target widths and five movement amplitudes. These are categorized into the following levels in cm: $W \in \{18, 36, 72, 120\}$ and $A \in \{100, 150, 200, 250, 300\}$. Therefore, our study used a 4×5 within-subjects design. The experimental application included two interactable objects: a ball and a scoreboard (i.e., the target). The ball was placed at the participant's lower-right and could be thrown. The target had a fixed position in the (x, y) coordinates and was positioned directly in front of the participant (i.e., at eye level). The distance between the participant and the target was determined by A, while the target width varied according to W. For more details, please refer to Figure 1 and the supplementary video.

As the target was placed at a fixed position, we can ignore the z-axis and define the endpoint as (x, y), the collision point. Due to the limited collision area of the target, we placed an invisible object (i.e., a wall) at the same position as the target. This allowed us to collect the endpoints of missed trials (i.e., collision points outside the target). Participants threw the ball while standing, and no movement of position was allowed. Moreover, we accepted valid throwing selection inputs only when the ball completely traveled the amplitude A. In other words, if the ball did not reach the predefined distance A, i.e., the wall, participants had to repeat the same trial. This criterion follows previous work [64].

Our experiment consisted of eight blocks. We always ran two blocks consecutively to maintain motor performance. W and A were provided in randomized order. Each block had 40 trials, 20 (conditions) $\times$ 2 (repetitions), and in total, there were sixteen repetitions for each $W \times A$. In total, we recorded 16 participants $\times$ 40 trials $\times$ 8 blocks $=$ 5,120 data points.

4.3.2 Experimental Task and Procedure

The primary task was to hit the scoreboard by throwing the ball at it in VR. Initially, participants ($p_p = (0,0,0)$) were required to grab the ball placed on the desk at a position to the participant's lower-right ($p_b = (0.53, 1.41, -0.01)$) using the handheld controller. After grabbing the ball, a target with predefined W and A appeared in front of them at eye level ($p_t = (0, 1.7, A)$). Thus, participants first encountered the corresponding target condition only after picking up the ball. If the ball did not reach the target distance A, participants needed to grab the ball again and repeat the same trial. Although the target was designed as a scoreboard, there was no scoring system. We asked participants to hit the target as accurately as possible. We also told them that it was not necessary to hit the center of the target. We did not enforce a posture for throwing, i.e., both overhand and underhand throws were permitted, and participants could choose a comfortable and optimal posture for accurate throwing. A successful hit was calculated if the collision endpoint fell within the target area (W); otherwise, it was deemed a miss, meaning the ball landed outside the target after traveling A. The above process was repeated until the end of all blocks. We provided audio feedback on grabbing and throwing the ball. We further presented a visual cue when the ball hit the target or wall by bouncing the ball after the collision. During the task, both the virtual hand and the ball remained visible to participants throughout the throwing motion.

After greeting a participant, the experimenter explained the background and objectives of the study, followed by a brief introduction to the experimental task. Once verbal consent was acquired, we collected demographic information and formal consent forms. Initially, participants performed a practice block (i.e., 40 trials) identical to the main block. During practice and experiments, randomized target conditions were presented. If a participant failed to throw the ball to a distance of A, we deemed this trial invalid. This trial was thus repeated. The entire experiment lasted approximately 45 minutes per participant. After two consecutive blocks, participants received a mandatory 90-second break. Participants were welcome to take voluntary breaks as needed.

4.4 Measures

We used two primary measures for user behavior and performance in throwing selection: PT and endpoint distribution. As mentioned in Section 3.1, throwing-based selection involves different selection procedures compared to conventional pointing selection (see Figure 2). Therefore, we employ PT as an observable measure of pre-release preparation after the target appears. PT was automatically calculated as the time between grabbing the ball and throwing it in our experimental application, operationalized as the interval between pressing and releasing the grip button. The target appeared only after participants grabbed the ball; therefore, they could begin preparing for the throw only after the grab. Because this interval may include perceptual processing, target identification, hesitation, and motor preparation, we interpret PT as an observable pre-release preparation interval rather than pure motor-planning time. The second metric is the endpoint distribution. We used a 2D notation for the endpoint (x, y): the ball's collision point with the target. To characterize this distribution, we use five parameters (i.e., DVs): μ_x, σ_x, μ_y, σ_y, and ρ. More details are discussed in Section 2.2 and Equation 2. Figure 4 shows the empirical throwing endpoint distribution.

5 Results

All data from our sixteen participants were used for data analysis. We analyzed the data via repeated-measures ANOVA (RM-ANOVA). For post hoc comparison, a Bonferroni-adjusted pairwise test was used. For the normality test, we ran a Shapiro-Wilk (SW) test across each group of 16 participants $\times$ 20 conditions. To assess statistical significance, we used $p < .05$ and generalized eta-squared (η_g^2) to determine the impact of each task parameter, where $\eta_g^2 > .14$ indicates a large effect size [11]. To ensure reliable data analysis, we removed outliers. We found 97 outliers via the three-σ rule across the 320 data groups according to PT and endpoints (i.e., (x, y) coordinates). Figure 3 demonstrates the overall results of all DVs. In this section, we only report significant results from RM-ANOVA. For all results, please refer to Tables 1 and 2. All data analysis was conducted using MATLAB and R.

5.1 Planning Time

Participants took an average of 1.25$\pm$0.43 (mean$\pm$standard deviation) seconds for preplanning their throwing. We found that PT decreased sequentially from smaller W to larger values. Specifically, $W \in \{18, 36, 72, 120\}$ cm resulted in 1.48$\pm$0.62, 1.25$\pm$0.36, 1.15$\pm$0.28, and 1.10$\pm$0.28 seconds, respectively. In contrast, for A, we observed the opposite pattern: $A \in \{100, 150, 200, 250, 300\}$ cm resulted in 1.06$\pm$0.24, 1.16$\pm$0.34, 1.24$\pm$0.39, 1.34$\pm$0.50, and 1.43$\pm$0.53 seconds, respectively. The SW test revealed normality

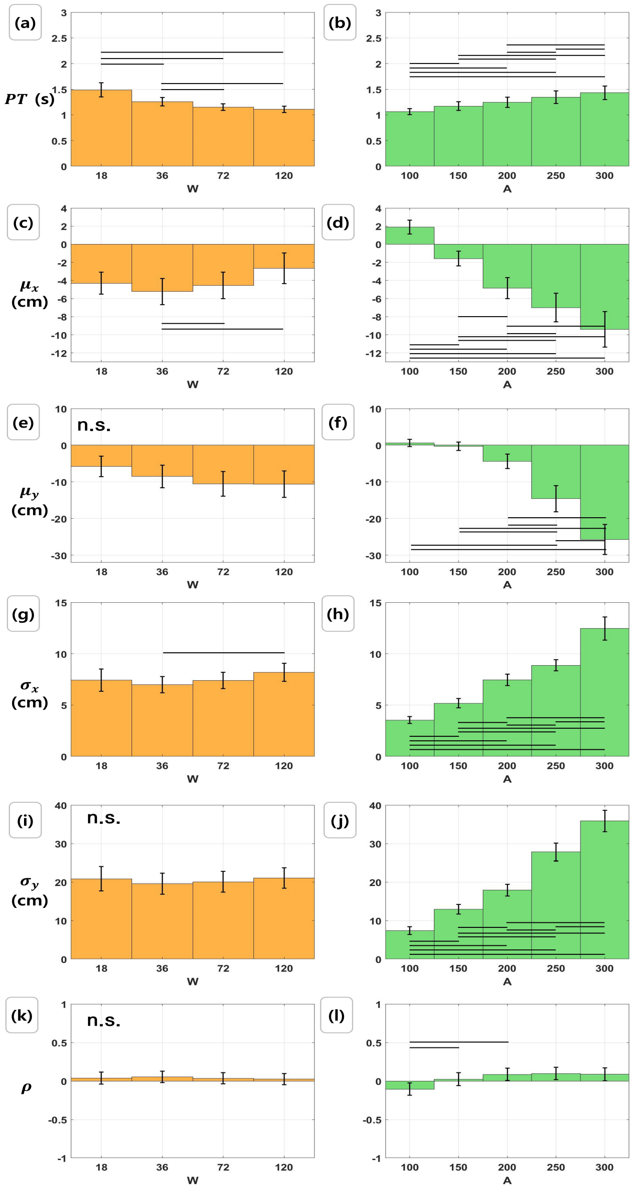

Figure 3: Results for all DVs in the study. The horizontal black lines are post hoc comparisons (at least $p < .05$). Error bars show 95% CIs.

across 70% conditions (222/320). RM-ANOVA showed a significant effect of W ($F_{1.107,16.615} = 17.805, p < .001, \eta_g^2 = .126$), A ($F_{1.237,18.556} = 21.251, p < .001, \eta_g^2 = .100$), and interaction ($W \times A$; $F_{2.297,34.464} = 4.324, p < .050, \eta_g^2 = .014$). The results of the post hoc comparisons are illustrated in Figure 3.

As anticipated, PT increased with task difficulty, similar to the increase of MT with ID in Fitts' law. Under more difficult conditions, participants spent more time before release, suggesting increased perceptual-motor preparation demands. Because PT may include perceptual processing as well as motor preparation, this effect should not be interpreted as direct evidence of pure motor planning. Nevertheless, these results support **H1** by showing a systematic relationship between PT and task difficulty. More details are discussed in Section 7.1.

5.2 Endpoint Distribution

We first ran the SW test to examine the normality of the throwing selection endpoint distribution for each x- and y-axis. The results indicated that both the x and y axes showed normality in 86% (278/320) and 82% (264/320) of all cases, respectively. Furthermore, bivariate normality based on multivariate skewness and kurtosis tests was satisfied in 96.9% (310/320) of cases.

Table 1: RM-ANOVA results for PT. Grey rows indicate significant results.

Factor	DV	d_{effect}	d_{error}	F	p	η_g^2
W	PT	1.107	16.615	17.805	$< .001$	.126
A	PT	1.237	18.556	21.251	$< .001$	.100
$W \times A$	PT	2.297	34.464	4.324	$< .050$	.014

Table 2: RM-ANOVA results for endpoint distribution. Grey rows indicate significant results.

Factor	DV	d_{effect}	d_{error}	F	p	η_g^2
W	μ_x	1.511	22.677	6.169	$< .050$	.031
A	μ_x	1.370	20.556	36.493	$< .001$	.361
$W \times A$	μ_x	4.462	66.930	2.512	$< .050$	.022
W	σ_x	2.440	36.604	5.165	$< .010$	.026
A	σ_x	1.703	25.547	120.970	$< .001$	.587
$W \times A$	σ_x	4.169	62.545	2.453	$= .052$	.055
W	μ_y	1.813	27.207	5.244	$< .050$	.034
A	μ_y	1.548	23.227	38.043	$< .001$	.474
$W \times A$	μ_y	4.495	67.436	1.979	$= .099$	.024
W	σ_y	1.854	27.817	0.773	$= .462$	.005
A	σ_y	1.848	27.728	130.900	$< .001$	.646
$W \times A$	σ_y	4.969	74.538	1.395	$= .236$	.025
W	ρ	2.694	40.416	0.179	$= .892$	.001
A	ρ	2.709	40.649	4.988	$< .010$	.051
$W \times A$	ρ	6.569	98.545	0.688	$= .674$	.018

For μ_x, RM-ANOVA resulted in W ($F_{1.511,22.677} = 6.169, p < .050, \eta_g^2 = .031$), A ($F_{1.370,20.556} = 36.493, p < .001, \eta_g^2 = .361$), and their interaction ($F_{4.462,66.930} = 2.512, p < .050, \eta_g^2 = .022$) being significant. Remarkable differences were observed in W ($F_{2.440,36.604} = 5.165, p < .010, \eta_g^2 = .026$) and A ($F_{1.703,25.547} = 120.97, p < .001, \eta_g^2 = .587$) for σ_x. In the case of the y-axis, μ_y exhibited significant differences in both W ($F_{1.813,27.207} = 5.244, p < .050, \eta_g^2 = .034$) and A ($F_{1.548,23.227} = 38.043, p < .001, \eta_g^2 = .474$). In terms of σ_y, we identified the following result from RM-ANOVA: A ($F_{1.848,27.728} = 130.90, p < .001, \eta_g^2 = .646$). Finally, ρ was significantly different across As ($F_{2.709,40.649} = 4.988, p < .010, \eta_g^2 = .051$). The results of the post hoc comparisons are illustrated in Figure 3. Figure 4 demonstrates the endpoint distributions across all conditions. As A increased, the variability also increased, while the impact of W did not appear visually significant. Overall, the endpoints were spread more in the vertical direction than in the horizontal direction. Additionally, as A increased, the effective target size also increased, and the endpoint distributions tended to form slightly rotated ellipses. The overall shapes from the endpoint distribution correspond to previous work [64].

Our results support accepting **H2** since we observed that most throwing endpoints formed a bivariate Gaussian based on the results of multivariate skewness and kurtosis tests. Each axis also showed a mostly normal distribution.

H3 received limited support. As shown in the last row of Figure 3, ρ varied significantly with A, yet the value remained small across conditions with no substantial patterns. Our findings showed ρ of 0.03±0.35, 0.05±0.34, 0.03±0.32, and 0.02±0.33 for W of 18, 36, 72, and 120, respectively. For $A \in \{100, 150, 200, 250, 300\}$, we found ρ of -0.10±0.32, 0.02±0.32, 0.08±0.32, 0.09±0.32, and 0.08±0.34, respectively. The estimated correlations were small ($|\rho| < 0.2$), suggesting a weak relationship between the two axes according to conventional correlation effect-size guidelines [15]. Looking at previous pointing selection work [58], their ρ was nearly zero, too (e.g., -0.022). However, our findings should not be interpreted as evidence that the two axes are

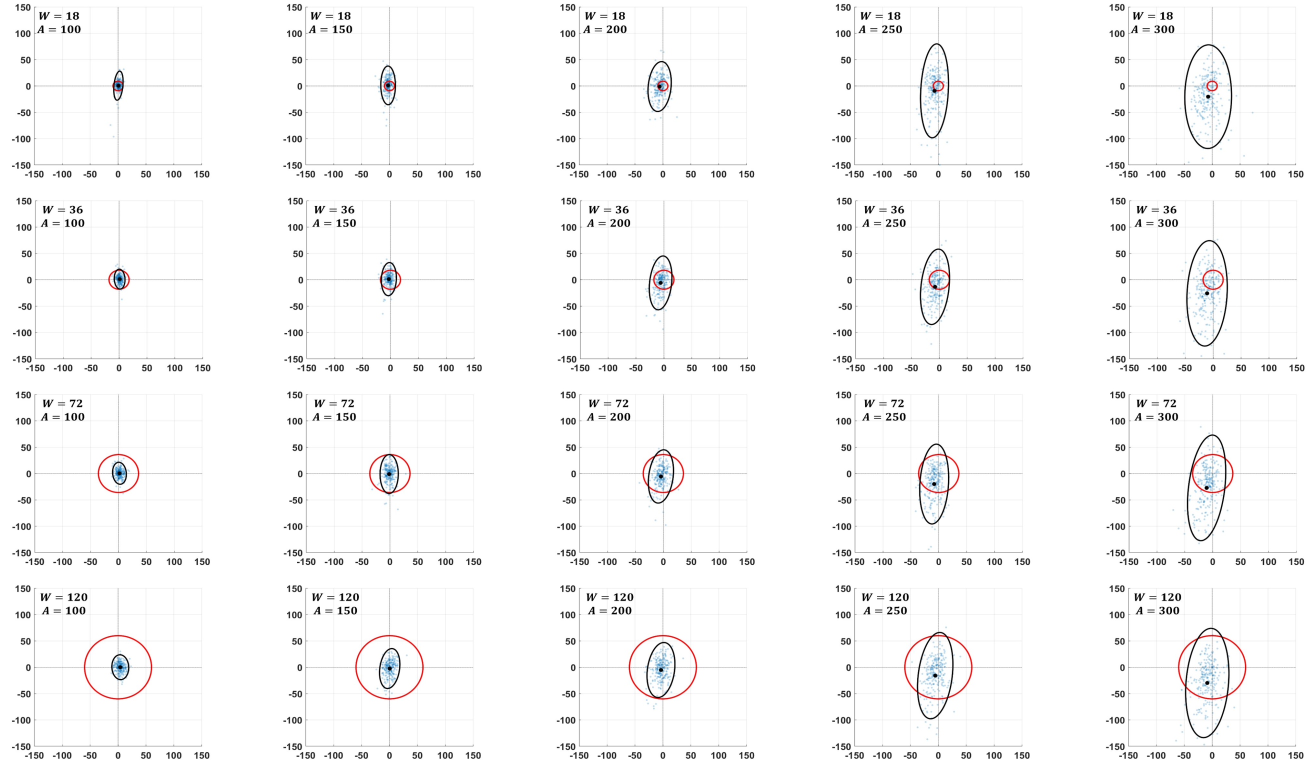

Figure 4: Distributions of endpoints across all task conditions. The red circle represents the nominal target, while the black ellipse represents the 95% CIs based on the actual endpoints (blue dots).

independent. Rather, they indicate a weak relationship between horizontal and vertical endpoint errors in the present setup. For modeling purposes, we therefore set ρ to zero as a practical simplification [48].

Finally, we argue that **H4** is supported. Based on our results, both W and A affected multiple endpoint-distribution parameters, with A showing broader and stronger effects than W. As A increased, the ball trajectories showed larger variability (see Figure 5), suggesting that motor noise at release was amplified during the longer flight of the ball [45]. In addition, μ_x was generally negative along the horizontal axis. This pattern is likely related to the fact that all participants were right-handed. When throwing the ball, participants tended to move their arm from right to left rather than the opposite. Therefore, given that the target center was defined as the origin on the x axis, the endpoints tended to appear at negative values. For μ_y, both W and A showed significant effects, although A had a substantially larger effect. This is likely because participants threw the ball along a parabolic trajectory toward the center, causing the endpoints to fall below the center more often at larger amplitudes. In terms of variability (i.e., σ_x and σ_y), W showed significance only for σ_x, but no clear pattern was observed. In contrast, A showed a significant linear relationship for both σ_x and σ_y. Overall, as A increased, the ball trajectory included more noise, which was amplified along the ball's flight.

5.3 Selection Accuracy

The grand success rate in our experiment was 67.58±33.76%. Specifically, $W \in \{18, 36, 72, 120\}$ cm resulted in 34.17±30.40, 60.31±31.16, 83.39±21.68, and 92.46±13.20%, respectively. In contrast, for A, we observed the opposite pattern: $A \in \{100, 150, 200, 250, 300\}$ cm resulted in 93.39±13.23, 81.52±25.61, 68.27±31.80, 52.98±32.98, and 41.75±32.77%, respectively. Due to the characteristics of throwing, overall accuracy was lower compared to conventional selection [7, 48, 58].

5.4 Ball Trajectory Variability

We further visualized ball trajectory variability across conditions (Figure 5). We visualized throwing trajectories for each (W, A) condition using the recorded 3D trajectory logs. For each trial, the raw (x, y, z) trajectory points were temporally normalized by resampling them to $N = 200$ evenly spaced samples over the normalized progress $[0, 1]$ using shape-preserving cubic interpolation. Trials with fewer than two trajectory points were discarded. The resampled trajectories were then averaged across trials within each (W, A) condition to obtain a mean trajectory.

Shorter amplitudes ($A \in \{100, 150, 200\}$ in cm) showed relatively straight trajectories. As A increased, the trajectories became more curved and showed larger variability across trials. This indicates that longer travel distances resulted in greater variability.

5.5 Summary

In summary, this section explored throwing selection behavior and performance. Similar to conventional pointing selection, we found that W and A significantly affected both selection behavior and performance. Throwing strategies also varied depending on the target parameters. We did not enforce posture (i.e., overhand vs. underhand) during the experiments, but all participants selected overhand throwing. This choice might be related to the fact that the target was situated at eye level. However, previous work [64] identified similar endpoint distributions for both postures. Therefore, we believe that there would be no substantial differences. In conclusion, we assessed our hypotheses based on empirically collected experimental data. While **H1**, **H2**, and **H4** were supported, **H3** received limited support. In the next section, based on these results, we model PT, the endpoint distribution of throwing selection, and the final user performance (i.e., selection accuracy).

6 Modeling

Here, we derive time performance and endpoint distribution models. For modeling, we used the empirical data points collected in Section 4. We employed several metrics to evaluate the goodness-of-fit: adjusted

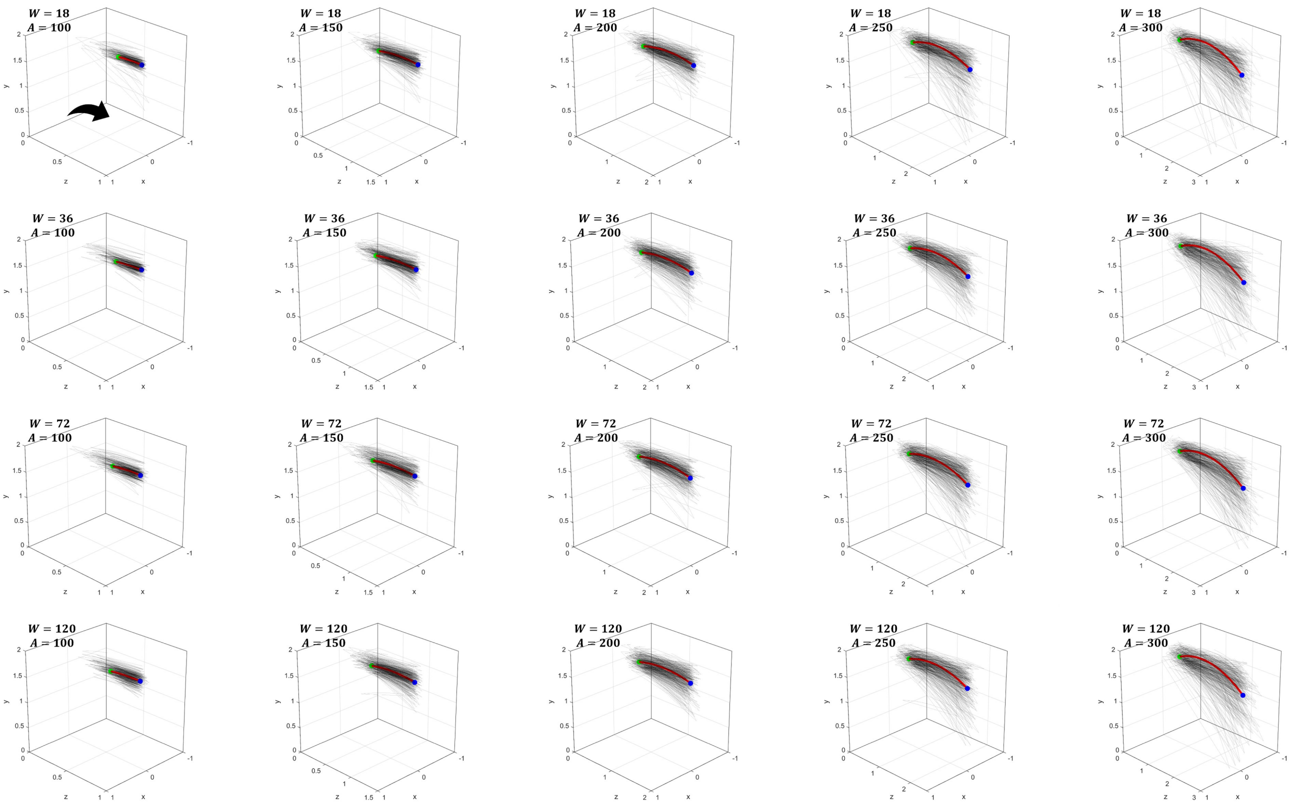

Figure 5: Averaged ball position for all conditions. The green dot indicates the mean ball position at release, the blue dot indicates the mean endpoint, and the red and black lines represent averaged and raw trajectory data, respectively.

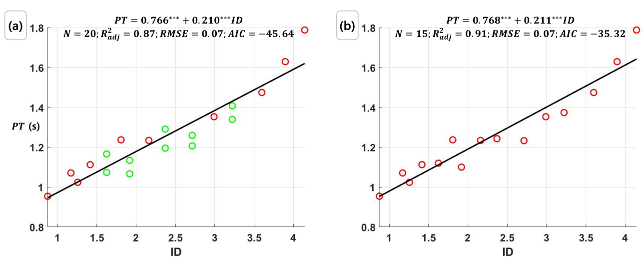

Figure 6: Model fitness for (a) data averaged by all conditions of W and A factors ($N = 20$) and (b) data averaged by IDs ($N = 15$). For (a), each data point for the ten conditions (five pairs) sharing the same ID is marked in green. For (b), the data for conditions with identical IDs were merged through averaging ($^* = p < .05$, $^{**} = p < .01$, $^{***} = p < .001$).

R^2 (R^2_{adj}), root mean squared error ($RMSE$), and Akaike's information criterion (AIC). Higher R^2_{adj} and lower $RMSE$ and AIC demonstrate better goodness-of-fit.

6.1 Modeling Planning Time

As reported in Section 5.1, our findings showed that PT increased for smaller W and longer A, similar to Fitts' law. Therefore, we replace the DV (MT) in Equation 1 with PT:

$$PT = a + b\log_2(A/W + 1) \tag{3}$$

In our experimental design, five pairs of conditions had identical IDs (ranging from 0.87 to 4.41) across different target factor combinations: 1.62, 1.91, 2.36, 2.71, and 3.22. Based on Fitts' law, the corresponding IDs should produce similar MTs despite differences in the target parameters. We found that PT also played a role similar to MT in throwing selection. As a result, we confirmed that similar PTs occur between shared ID conditions, and these results suggest that the ID-based PT model was consistent within our tested parameter range. Here, we fit two data groups: one averaged by W and A factors ($N = 20$) and the other averaged by IDs ($N = 15$).

The fitting results are provided in Figure 6. Both models showed strong performance. Modeling all data regardless of shared IDs ($N = 20$) achieved an R^2_{adj} of 0.87, while the model averaged by IDs ($N = 15$) showed an R^2_{adj} of 0.91. All parameters of the linear regression were significant ($p < .001$). For $RMSE$, both models produced comparable results of 0.07 seconds. Finally, in terms of model efficiency measured by AIC, the model averaged by W and A factors ($N = 20$) showed a lower AIC of -45.64 than the model averaged by IDs ($N = 15; AIC = -35.32$). However, direct comparison is difficult because the two models are based on different numbers of data points. In conclusion, we found that the two models did not show substantial differences, and modeling PT in throwing selection was very similar to Fitts' law.

6.2 Modeling Endpoint Distribution

To build a selection accuracy model of throwing selection, we first need to model its endpoint distribution. Given that most endpoint distributions exhibited bivariate normality and that W and A significantly affected several distribution parameters (Section 5.2 and Table 2), we modeled the endpoints using the following bivariate Gaussian distribution:

$$\hat{\mu}_x = a + bW + cA + d(W \times A) \tag{4}$$
$$\hat{\sigma}_x = a + bW + cA \tag{5}$$
$$\hat{\mu}_y = a + bW + cA \tag{6}$$
$$\hat{\sigma}_y = a + bA \tag{7}$$

Lowercase letters denote empirically determined independent regression coefficients for each formulation. We selected the significant task

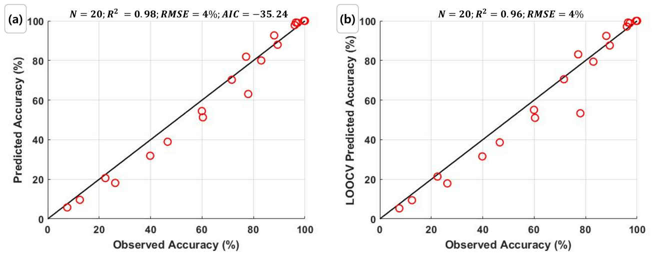

Figure 7: Model fitness for accuracy prediction: (a) in-sample evaluation and (b) LOOCV.

parameter that met our criteria, $p < .05$ and $\eta_g^2 > .01$ (small effects). Although **H3** received limited support, the small observed magnitude of ρ should not be interpreted as evidence that the two endpoint axes are statistically independent. Nevertheless, we set ρ to zero as a practical modeling simplification to reduce model complexity. Finally, we can derive the following bivariate-Gaussian distribution model based on the estimated parameters from the above models and Equation 2:

$$\mu = \begin{bmatrix} \hat{\mu}_x \\ \hat{\mu}_y \end{bmatrix}, \Sigma = \begin{bmatrix} \hat{\sigma}_x^2 & 0 \\ 0 & \hat{\sigma}_y^2 \end{bmatrix} \tag{8}$$

Table 3 summarizes the fitting results of all models. All models consistently achieved high fits in terms of all metrics. Except for one with $R^2_{adj} = 0.85$, they all resulted in R^2_{adj} larger than 0.91. It should be noted that some predictors retained from the RM-ANOVA results were not significant at the individual regression-coefficient level. For example, W was retained in the σ_x model because W showed a significant effect in the corresponding RM-ANOVA, but its fitted regression coefficient was not significant in Table 3. We therefore interpret this predictor cautiously and do not claim a strong isolated regression effect of W on σ_x. Similarly, W in the μ_y model should also be interpreted cautiously.

Another approach is to use only the parameters with strong effect sizes (i.e., $p < .05$ and $\eta_g^2 > .14$). In our case, A was the only parameter that satisfied this criterion (see Table 2). This approach also simplifies model formulation across different bivariate-Gaussian distribution parameters. All models can then be formulated as follows: $\hat{DV} = a + bA$. We also trained and tested this simplified model. We obtained the following fitting results ($N = 5$) for μ_x and σ_x: $\hat{\mu}_x = 7.001 - 0.055A; R^2_{adj} = 0.98; RMSE = 0.53; AIC = 9.31$ and $\hat{\sigma}_x = -1.108 + 0.043A; R^2_{adj} = 0.96; RMSE = 0.625; AIC = 10.93$, respectively. Additionally, the model achieved the following fitting performance for μ_y: $\hat{\mu}_y = 17.870 - 0.133A; R^2_{adj} = 0.86; RMSE = 4.15; AIC = 29.86$. Because this strong-effect-only formulation uses a different dataset aggregated by A only ($N = 5$), the results are not directly comparable with the full models in Table 3 ($N = 20$). Therefore, we report these results as a compact alternative formulation rather than as a direct efficiency improvement over the full models.

6.3 Predicting Throwing Selection Accuracy

One of the primary goals of this work is to predict the accuracy of throwing selection. As reported in Section 5.3, selection accuracy systematically changed with W and A. By predicting the selection accuracy of throwing, our results can assist in the design and optimization of GUIs, games, etc. For this, we employed well-known probability target acquisition models in 2D space [9, 58]:

$$y = \iint_D f\left(\begin{bmatrix} x \\ y \end{bmatrix}, \mu, \Sigma\right) dx\, dy, \tag{9}$$

where μ and Σ correspond to Equation 2. Given the small absolute magnitude of the observed ρ values and the practical simplification introduced in Section 6.2, Equation 9 can be approximated as:

$$y = \iint_D \frac{1}{2\pi\hat{\sigma}_x\hat{\sigma}_y} \exp\left(-\frac{(x - \hat{\mu}_x)^2}{2\hat{\sigma}_x^2} - \frac{(y - \hat{\mu}_y)^2}{2\hat{\sigma}_y^2}\right) dx\, dy \tag{10}$$

Based on the above method, we can calculate the selection accuracy of the target D. We applied the estimated values from Equation 4 to 7. The prediction results are provided in Figure 7(a). Fitness results showed 0.98, 4%, and -35.24 for R^2, *RMSE*, and *AIC*, respectively. Our model thus successfully predicted throwing selection accuracy. To verify the robustness of our model, we ran leave-one-condition-out cross-validation (LOOCV) across 20 conditions (Figure 7(b)), where R^2 and *RMSE* were 0.96 and 4%, respectively. This evaluation was performed at the condition level rather than the participant level. These values are comparable to those of the in-sample model. In summary, within the tested setup and parameter range, our model can reliably predict selection accuracy in throwing selection in VR.

7 General Discussion

7.1 Motor Control Strategies in Pointing and Throwing

We found that PT followed a pattern similar to MT in Fitts' law, with more time spent on targets with a high ID. While the goal of selecting a target remains the same, this time allocation suggests a fundamental difference in how selection behavior is managed in throwing versus pointing (Figure 2). Unlike pointing, throwing requires users to manage task difficulty primarily before release. Therefore, under difficult conditions, users may rely more heavily on preplanning processes before release to cope with motor variability and improve the likelihood of a successful hit. From this perspective, the increase in PT may reflect a behavioral adaptation to task difficulty. One possible interpretation is that participants allocated additional time to prepare for accurate throws under more difficult conditions. However, the present study did not directly measure the underlying motor-planning processes, and thus, this interpretation should be treated with caution. Moreover, this additional preparation may not have been sufficient to fully compensate for the increased motor variability observed in difficult throwing conditions. Overall, these results suggest that, unlike traditional pointing, throwing-based interaction places greater emphasis on the pre-release stage for achieving accuracy control.

7.2 Throwing Endpoint Distributions

Although we used a circular target (i.e., the scoreboard), the actual endpoints formed an elliptical distribution (see Figure 4). This indicates that horizontal and vertical controls are handled differently during throwing. First, horizontal error (σ_x) was affected by W. Horizontal control primarily depends on adjusting the arm's aiming direction. When the target was wider (large W), participants tolerated greater horizontal variability. Consequently, significant differences in W were observed only for σ_x (36 vs. 120 cm), but not for σ_y. The horizontal bias (μ_x) also varied significantly with W. Interestingly, as W increased, the endpoints moved closer to the target center (see Figure 3(c)). This contrasts with the lazy effect observed in cursor-based pointing, where users tend to terminate movements near the target boundary when the target is sufficiently large [48, 58]. In throwing selection, however, continuous feedback-based correction is not possible after release. Therefore, participants likely treated the center of the target as the primary reference point under all W conditions to minimize failure. In other words, the lazy effect observed in cursor-based studies does not seem to appear in throwing selection.

In contrast, vertical error (σ_y) was primarily determined by A rather than W. The distance the ball travels depends on the release angle, force, and timing [45]. Even small release noise accumulates during flight, amplifying trajectory deviations (see Figure 5). This explains why vertical errors were consistently larger than horizontal errors. In other words, horizontal control might require matching the release angle, whereas vertical control requires coordinating angle, force, and timing. Another notable pattern was a downward bias ($\mu_y < 0$) with increasing distance. Participants showed a systematic undershooting bias. Throwing the ball too high may cause it to fly over the target and leave the field of view. To avoid such a miss, participants tended to throw slightly shorter, even if it resulted in hitting the lower part of the target. This suggests that users prefer keeping the ball within a visible and controllable range. This explanation is also consistent with

Table 3: Model fitting results ($^* = p < .05$, $^{**} = p < .01$, $^{***} = p < .001$).

DV	Equation	N	a	b	c	d	R^2_{adj}	*RMSE* (cm)	*AIC*
μ_x	4	20	3.526*	0.056**	-0.044***	-0.0001*	0.94	1.05	62.09
σ_x	5	20	-1.667*	0.009	0.043***	-	0.92	0.90	55.30
μ_y	6	20	20.557***	-0.043	-0.133***	-	0.85	4.07	115.65
σ_y	7	5	-8.393*	0.143**	-	-	0.97	1.78	21.39

previous work [53], which showed that continuous visual cues improve throwing performance.

Finally, the near-zero correlation between horizontal and vertical errors ($|\rho| < 0.2$; **H3** limited support) can be explained based on our analysis in this section. Horizontal and vertical endpoint errors may be influenced by different components of the throwing movement. Horizontal error may be more closely associated with directional aiming, whereas vertical error may be more strongly influenced by release angle, force, and timing. However, the present results do not establish that the two axes are statistically independent or generated by separate control processes.

7.3 Contribution to HCI and Implications

Throwing is one of the fundamental movements in many sports and games. As VR interactive systems allow natural interactions with a high degree of freedom, throwing operations can be utilized in many applications. In particular, the throwing metaphor has often been employed for target selection. Nevertheless, it has received relatively little attention from the human behavior and performance modeling perspective in the HCI community compared to pointing-based interaction. In this study, we enhanced the understanding of throwing-based selection in VR and presented predictive models for time performance (PT) and endpoint distribution. We believe our work opens new opportunities for user modeling in single-shot interaction metaphors (e.g., throwing, kicking, punching).

Our findings and models could be applied to various throwing-based applications, including darts, baseball, rugby, and American football. By optimizing their design, these applications can improve throwing precision, control the level of difficulty, and enhance the overall user experience. Our prediction model for selection accuracy also demonstrated strong robustness through LOOCV evaluation. This indicates that the model can predict users' throwing performance even under previously untested conditions within the tested setup and parameter range. Designers and practitioners can extend these insights to develop throwing-based applications, allowing user interfaces to be optimized based on scientific evidence.

Finally, we present several implications: 1) We presented two predictive models for PT and accuracy and recommend using the two models simultaneously to achieve a time-accuracy balance, as a high PT does not necessarily guarantee high accuracy. 2) In situations where quick throwing is important, such as the timely throwing of a grenade, use the PT model. 3) Regardless of the target's size, users can aim their throws toward the center. In particular, since variability is greater in the y-axis than in the x-axis, using a vertically elongated target shape may further improve performance.

7.4 Limitations and Future Work

Although we presented several insights into modeling throwing selection, our study also has some limitations. Our results are naturally limited by the experimental setup, such as the Unity built-in physics system, the throwing interaction SDK, and the Rigidbody configuration, as mentioned in Section 4.2. Therefore, our findings should be interpreted within the context of our experimental design. PT was measured as the interval between pressing and releasing the grip button after the target appeared. Thus, it may include perceptual processing, hesitation, and posture adjustment in addition to motor preparation, potentially overestimating planning duration. Accordingly, the PT model should be interpreted as describing pre-release preparation rather than motor planning alone. The grand success rate also showed substantial variability across participants. Because our models were developed from data aggregated across participants, they characterize population-level tendencies rather than individual throwing ability or strategy. Future work should examine models that account for participant-specific differences. We used a commercial handheld controller from Meta Quest 3 to grab and release the ball. Different input devices and interaction mechanisms could lead to different performance in VR throwing [17]. The models should also not be assumed to generalize directly to real-world throwing because physical objects provide different haptic, release, and motion characteristics. For better generalizability, future work should thus examine other devices and interaction mechanisms and test whether our models hold under those conditions. Participants were asked to throw the ball naturally, and therefore, no posture constraint (e.g., overhand or underhand) was imposed. The target was placed only at eye level, which may have encouraged participants to adopt an overhand throwing posture. Still, previous work reported similar endpoint distributions for both postures [64], which is consistent with the elliptical distribution observed in our study. Therefore, the effect of posture may be negligible for modeling purposes. However, this assumption should be verified empirically in future studies.

8 Conclusion

In summary, this paper investigated throwing selection behavior and performance in VR. Similar to conventional pointing selection, we found that W and A significantly affected both behavior and performance, and that throwing strategies varied depending on the target parameters. We employed conventional target selection parameters as independent variables (W and A) and measured two main dependent variables: PT and endpoint distribution. For PT, we assumed that it might follow a pattern similar to MT in Fitts' law, and the results confirmed a significant linear relationship with selection difficulty (i.e., ID). For endpoint distribution, vertical error (σ_y) was significantly larger than horizontal error, resulting in an overall vertically elongated distribution. Based on the results, we established several new models using the significant task parameters and achieved high prediction accuracy. Based on these models, we could successfully predict throwing-based selection accuracy. Our findings have not been extensively addressed in the literature, and we believe our work supports a deeper understanding of throwing-based target selection in VR for designers, practitioners, and researchers. We hope these findings will help guide the design and optimization of comparable VR throwing interfaces and motivate validation under broader parameter conditions.

Acknowledgments

This study was financially supported by Seoul National University of Science and Technology (Project No. 2025-1500).

References

[1] J. Accot and S. Zhai. Beyond fitts' law: models for trajectory-based hci tasks. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, CHI '97, pp. 295–302. Association for Computing Machinery, New York, NY, USA, 1997. doi: 10.1145/258549.258760 1

[2] J. Accot and S. Zhai. More than dotting the i's — foundations for crossing-based interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '02, pp. 73–80. Association for Computing Machinery, New York, NY, USA, 2002. doi: 10.1145/503376.503390 1

[3] J. Accot and S. Zhai. Refining fitts' law models for bivariate pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pp. 193–200. Association for Computing Machinery, New York, NY, USA, 2003. doi: 10.1145/642611.642646 1

[4] M. Amini, W. Stuerzlinger, S. Yamanaka, H.-N. Liang, and A. U. Batmaz. Tunnels vs. wires: A comparative analysis of two 3d steering tasks in virtual environments. In *Proceedings of the 2025 31st ACM Symposium on Virtual Reality Software and Technology*, VRST '25, art. no. 15, 11 pp. Association for Computing Machinery, New York, NY, USA, 2025. doi: 10.1145/3756884.3766023 1

[5] M. R. Bashar, M. D. Barrera Machuca, W. Stuerzlinger, and A. U. Batmaz. The effect of visual depth on the vergence–accommodation conflict on 3d selection performance within virtual reality headsets: Mr bashar et al. *The Visual Computer*, 41(12):9645–9661, 2025. 1

[6] A. U. Batmaz and W. Stuerzlinger. Effect of fixed and infinite ray length on distal 3d pointing in virtual reality. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI EA '20, pp. 1–10. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3334480.3382796 1

[7] X. Bi, Y. Li, and S. Zhai. Ffitts law: modeling finger touch with fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pp. 1363–1372. Association for Computing Machinery, New York, NY, USA, 2013. doi: 10.1145/2470654.2466180 1, 2, 3, 6

[8] X. Bi and S. Zhai. Bayesian touch: a statistical criterion of target selection with finger touch. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, pp. 51–60, 2013. 4

[9] X. Bi and S. Zhai. Predicting finger-touch accuracy based on the dual gaussian distribution model. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 313–319, 2016. 3, 8

[10] M. Borgwardt, J. Boueke, M. F. Sanabria, M. Bonfert, and R. Porzel. Vrisbee: How hand visibility impacts throwing accuracy and experience in virtual reality. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–7, 2023. 2, 3

[11] J. Cohen. *Statistical power analysis for the behavioral sciences*. routledge, 2013. 4

[12] S. A. Drew, M. F. Awad, J. A. Armendariz, B. Gabay, I. J. Lachica, and J. W. Hinkel-Lipsker. The trade-off of virtual reality training for dart throwing: a facilitation of perceptual-motor learning with a detriment to performance. *Frontiers in Sports and Active Living*, 2:59, 2020. 1

[13] D. Elliott, S. Hansen, L. E. Grierson, J. Lyons, S. J. Bennett, and S. J. Hayes. Goal-directed aiming: two components but multiple processes. *Psychological bulletin*, 136(6):1023, 2010. 1, 2

[14] D. Elliott, J. Lyons, S. J. Hayes, J. J. Burkitt, J. W. Roberts, L. E. Grierson et al. The multiple process model of goal-directed reaching revisited. *Neuroscience & Biobehavioral Reviews*, 72:95–110, 2017. 1, 2

[15] J. D. Evans. *Straightforward statistics for the behavioral sciences.* Thomson Brooks/Cole Publishing Co, 1996. 5

[16] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954. 1, 2

[17] A. Ghasemaghaei, M. Maslych, Y. Hmaiti, E. S. Martinez, and J. J. LaViola. Towards better throwing: A comparison of performance and preferences across point of release mechanics in virtual reality. In *Proceedings of the 50th Graphics Interface Conference*, GI '24, art. no. 26, 11 pp. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3670947.3670963 1, 2, 3, 4, 9

[18] T. Grossman and R. Balakrishnan. A probabilistic approach to modeling two-dimensional pointing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(3):435–459, 2005. 3, 4

[19] C. M. Harris and D. M. Wolpert. Signal-dependent noise determines motor planning. *Nature*, 394(6695):780–784, 1998. 3

[20] HTC VIVE Developer. Unity: Realistic hand interaction (beta). https://developer.vive.com/resources/openxr/unity/tutorials/hand-tracking/realistic-hand-interaction/, 2025. Accessed: 2026-02-27. 2

[21] J. Huang, F. Tian, X. Fan, H. Tu, H. Zhang, X. Peng et al. Modeling the endpoint uncertainty in crossing-based moving target selection. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2020. 2

[22] J. Huang, F. Tian, X. Fan, X. Zhang, and S. Zhai. Understanding the uncertainty in 1d unidirectional moving target selection. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pp. 1–12, 2018. 1, 2, 3

[23] J. Huang, F. Tian, N. Li, and X. Fan. Modeling the uncertainty in 2d moving target selection. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, pp. 1031–1043, 2019. 2

[24] Y.-J. Huang, M.-Y. Tseng, S. Yaqub, T.-Y. Hsieh, Y.-C. Huang, S.-S. Lee et al. A systematic review on throwing in virtual environments. *Available at SSRN 5108309*. 1, 2

[25] J.-W. Jeong and J.-Y. Jeong. Effect of onset position of ray casting in virtual reality. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '24, art. no. 116, 7 pp. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3613905.3650905 1

[26] J.-Y. Jeong and J.-W. Jeong. Understanding user behavior in window selection using dragging for multiple targets. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, art. no. 855, 21 pp. Association for Computing Machinery, New York, NY, USA, 2025. doi: 10.1145/3706598.3713410 1

[27] S. Kelling and N. Kelling. Don't drop the ball: A comparison of throwing performance between vr and real life. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 67, pp. 1686–1692. SAGE Publications Sage CA: Los Angeles, CA, 2023. 2, 4

[28] P.-O. Kristensson and S. Zhai. Shark2: a large vocabulary shorthand writing system for pen-based computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, pp. 43–52. Association for Computing Machinery, New York, NY, USA, 2004. doi: 10.1145/1029632.1029640 1

[29] L. Lane, F. Lu, S. Davari, R. J. Teather, and D. A. Bowman. Revisiting performance models of distal pointing tasks in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 31(10):8283–8296, 2025. doi: 10.1109/TVCG.2025.3567078 1

[30] B. Lee and A. Oulasvirta. Modelling error rates in temporal pointing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 1857–1868. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2858036.2858143 1, 3

[31] N. Li, F. Tian, J. Huang, X. Fan, and H. Wang. 2d-bayespointer: An implicit moving target selection technique enabled by human performance modeling. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–6, 2018. 2

[32] J.-A. Lozano-Quilis, H. Gil-Gomez, J.-A. Gil-Gomez, S. Albiol-Perez, G. Palacios, H. M. Fardoum et al. Virtual reality system for multiple sclerosis rehabilitation using kinect. In *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops*, pp. 366–369. IEEE, 2013. 1

[33] T. Luong, Y. F. Cheng, M. Möbus, A. Fender, and C. Holz. Controllers or bare hands? a controlled evaluation of input techniques on interaction performance and exertion in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 29(11):4633–4643, 2023. doi: 10.1109/TVCG.2023.3320211 1

[34] 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*, pp. 1633–1636, 2008. 2

[35] 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*, 95(3):340, 1988. 3

[36] oculus-samples. Unity-interactionsdk-samples. https://github.com/oculus-samples/Unity-InteractionSDK-Samples, 2025. Sample projects to showcase the Unity Meta XR Interaction SDK. 2, 3, 4

[37] A. Oulasvirta, P. O. Kristensson, X. Bi, and A. Howes. *Computational interaction*. Oxford University Press, 2018. 1

[38] E. Park, H. Kim, and B. Lee. Button++ designing risk-aware smart buttons. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–6, 2018. 2

[39] E. Park and B. Lee. An intermittent click planning model. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pp. 1–13. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3313831.3376725 1, 2

[40] M. Silfverberg, I. S. MacKenzie, and P. Korhonen. Predicting text entry speed on mobile phones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, pp. 9–16. Association for Computing Machinery, New York, NY, USA, 2000. doi: 10.1145/332040.332044 1

[41] Y. Singh, S. K. Agrawal, and V. Vashista. Throwing strategy in a dual-motor-task of aiming at the bullseye while walking in virtual reality. *IEEE*

Robotics and Automation Letters, 7(4):9091–9098, 2022. 2, 3

[42] R. W. Soukoreff and I. S. MacKenzie. Towards a standard for pointing device evaluation, perspectives on 27 years of fitts' law research in hci. *International journal of human-computer studies*, 61(6):751–789, 2004. 2

[43] J. Tirp, C. Steingröver, N. Wattie, J. Baker, and J. Schorer. Virtual realities as optimal learning environments in sport-a transfer study of virtual and real dart throwing. *Psychological Test and Assessment Modeling*, 57(1):57, 2015. 1

[44] H. Usuba, S. Yamanaka, and H. Miyashita. Comparing lassoing criteria and modeling straight-line and one-loop lassoing motions considering criteria. In *Proceedings of the 2019 ACM International Conference on Interactive Surfaces and Spaces*, ISS '19, pp. 181–191. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3343055.3359707 1

[45] M. Venkadesan and L. Mahadevan. Optimal strategies for throwing accurately. *Royal Society Open Science*, 4(4), 2017. 1, 2, 3, 6, 8

[46] U. Wagner, M. N. Lystbæk, P. Manakhov, J. E. S. Grønbæk, K. Pfeuffer, and H. Gellersen. A fitts' law study of gaze-hand alignment for selection in 3d user interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, art. no. 252, 15 pp. Association for Computing Machinery, New York, NY, USA, 2023. doi: 10.1145/3544548.3581423 1

[47] F. Wang and X. Ren. Empirical evaluation for finger input properties in multi-touch interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1063–1072, 2009. 3

[48] Y. Wei, R. Shi, D. Yu, Y. Wang, Y. Li, L. Yu et al. Predicting gaze-based target selection in augmented reality headsets based on eye and head endpoint distributions. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pp. 1–14, 2023. 1, 2, 3, 4, 6, 8

[49] Y. Wei, K. Xu, Y. Li, L. Yu, and H.-N. Liang. Exploring and modeling directional effects on steering behavior in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 30(11):7107–7117, 2024. doi: 10.1109/TVCG.2024.3456166 1

[50] J. O. Wobbrock, E. Cutrell, S. Harada, and I. S. MacKenzie. An error model for pointing based on fitts' law. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1613–1622, 2008. 1, 2

[51] J. O. Wobbrock, A. Jansen, and K. Shinohara. Modeling and predicting pointing errors in two dimensions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1653–1656, 2011. 2

[52] G. Yamac, J. J. Chai, and C. O'Sullivan. Let it go! point of release prediction for virtual throwing. *Computers & Graphics*, 110:11–18, 2023. 2

[53] G. Yamac and C. O'Sullivan. Eye on the ball: The effect of visual cue on virtual throwing. In *SIGGRAPH Asia 2022 Posters*, pp. 1–2. 2022. 2, 4, 9

[54] S. Yamanaka and W. Stuerzlinger. Modeling fully and partially constrained lasso movements in a grid of icons. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 1–12. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300350 1

[55] S. Yamanaka, W. Stuerzlinger, and H. Miyashita. Steering through sequential linear path segments. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pp. 232–243. Association for Computing Machinery, New York, NY, USA, 2017. doi: 10.1145/3025453.3025836 1

[56] S. Yamanaka, H. Usuba, W. Stuerzlinger, and H. Miyashita. The effectiveness of path-segmentation for modeling lasso times in width-varying paths. *Proc. ACM Hum.-Comput. Interact.*, 6(ISS), art. no. 584, 20 pp., Nov. 2022. doi: 10.1145/3567737 1

[57] D. Yu, M. Cibulskis, E. S. Mortensen, M. S. Christensen, and J. Bergström. Metrics of motor learning for analyzing movement mapping in virtual reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, art. no. 724, 18 pp. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3613904.3642354 1

[58] D. Yu, H.-N. Liang, X. Lu, K. Fan, and B. Ens. Modeling endpoint distribution of pointing selection tasks in virtual reality environments. *ACM Trans. Graph.*, 38(6), art. no. 218, 13 pp., Nov. 2019. doi: 10.1145/3355089.3356544 1, 2, 3, 4, 5, 6, 8

[59] D. Yu, B. V. Syiem, A. Irlitti, T. Dingler, E. Velloso, and J. Goncalves. Modeling temporal target selection: A perspective from its spatial correspondence. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2023. 2, 3

[60] D. Zahradník, F. Vaverka, and V. Gajda. Optimisation of the size of a target and the throwing distance during a throw at a target for adults. *Acta Universitatis Palackianae Olomucensis. Gymnica*, 38(4), 2008. 3

[61] S. Zhai, J. Kong, and X. Ren. Speed–accuracy tradeoff in fitts' law tasks—on the equivalency of actual and nominal pointing precision. *International journal of human-computer studies*, 61(6):823–856, 2004. 2

[62] S. Zhai and P.-O. Kristensson. Shorthand writing on stylus keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pp. 97–104. Association for Computing Machinery, New York, NY, USA, 2003. doi: 10.1145/642611.642630 1

[63] Y. Zheng, J. Huang, H. Zhang, Y. Bian, J. Liu, C. Yang et al. 3d ternary-gaussian model: Modeling pointing uncertainty of 3d moving target selection in virtual reality. *International Journal of Human-Computer Studies*, 198:103454, 2025. doi: 10.1016/j.ijhcs.2025.103454 1

[64] T. Zindulka, M. Bachynskyi, and J. Müller. Performance and experience of throwing in virtual reality. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pp. 1–8, 2020. 1, 2, 3, 4, 5, 6, 9