Metrics of Motor Learning for Analyzing Movement Mapping in Virtual Reality

Difeng Yu
University of Copenhagen
Copenhagen, Denmark
diyu@di.ku.dk

Mantas Cibulskis
University of Copenhagen
Copenhagen, Denmark
maci@di.ku.dk

Erik Skjoldan Mortensen
University of Copenhagen
Copenhagen, Denmark
esm@psy.ku.dk

Mark Schram Christensen
University of Copenhagen
Copenhagen, Denmark
markc@psy.ku.dk

Joanna Bergström
University of Copenhagen
Copenhagen, Denmark
joanna@di.ku.dk

ABSTRACT

Virtual reality (VR) techniques can modify how physical body movements are mapped to the virtual body. However, it is unclear how users learn such mappings and, therefore, how the learning process may impede interaction. To understand and quantify the learning of the techniques, we design new metrics explicitly for VR interactions based on the motor learning literature. We evaluate the metrics in three object selection and manipulation tasks, employing linear-translational and nonlinear-rotational gains and finger-to-arm mapping. The study shows that the metrics demonstrate known characteristics of motor learning similar to task completion time, typically with faster initial learning followed by more gradual improvements over time. More importantly, the metrics capture learning behaviors that task completion time does not. We discuss how the metrics can provide new insights into how users adapt to movement mappings and how they can help analyze and improve such techniques.

CCS CONCEPTS

• **Human-centered computing** → **HCI theory, concepts and models**; **Virtual reality**; *Empirical studies in HCI*.

KEYWORDS

Interaction techniques, visual-motor mismatches, motor adaptation, beyond-real interactions

ACM Reference Format:
Difeng Yu, Mantas Cibulskis, Erik Skjoldan Mortensen, Mark Schram Christensen, and Joanna Bergström. 2024. Metrics of Motor Learning for Analyzing Movement Mapping in Virtual Reality. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3613904.3642354

CHI '24, May 11–16, 2024, Honolulu, HI, USA

ACM ISBN 979-8-4007-0330-0/24/05
https://doi.org/10.1145/3613904.3642354

1 INTRODUCTION

Virtual reality (VR) techniques can modify how physical body movements are mapped to the virtual body. Such techniques can enable a user to reach further in VR than their physical arm could reach [26, 55, 63] or impose supernatural body structures like an additional limb or tail [75, 88]. Many of these movement mappings have been shown to increase user performance and ergonomics in object selection or manipulation tasks [27, 68, 80, 86].

When a VR interaction technique introduces such a movement mapping, the user must learn it. Currently, task completion time is a primary metric to evaluate user performance and learning [7, 35, 90]. However, the time metric is abstract and oversimplified—it provides little understanding of why a user achieves such performance and does not capture crucial nuances of learning (e.g., what is being improved over the learning process and how [44, 87]). Therefore, it fails to inform how a movement mapping technique could be designed and improved.

The field of motor learning provides theories about how our bodies and central nervous systems learn movement mappings (e.g., adapting to slight changes such as mouse-to-cursor gains, or learning an entirely new control such as a reversing the cursor movement from that of the mouse) [16, 44, 52]. Therefore, motor learning may help to understand, design, and evaluate VR interaction techniques that introduce new movement mappings. However, the metrics from motor learning literature do not directly translate to measuring and understanding motor learning of VR interaction techniques.

One reason lies in the distinct objectives. The motor learning literature strives to introduce controlled perturbations that disrupt normal motions and induce errors systematically, allowing to describe the learning process [61, 62]. In contrast, VR interaction techniques aim to be fast to learn and effective to use, and only manipulate the mapping as much as is necessary to complete a task. Therefore, a good motor learning metric might not describe best the learning of a VR technique and point us to understand how the technique could be improved.

Another reason lies in the tasks. The accuracy of the movement in tasks without feedback is a sensitive measure for showing motor learning between repetitive trials, and thus modeling learning over time. In contrast, it is sufficient for the users to complete VR selection and manipulation tasks (or HCI tasks in general) under a certain accuracy threshold (e.g., hitting a target button anywhere

within its boundaries) and while receiving visual feedback continuously (e.g., of a cursor). Therefore, a good metric in motor learning may be less applicable for modeling learning of VR interaction techniques, in particular across varying tasks.

To quantify and understand how users learn movement mappings in VR, we design new metrics explicitly for VR interaction techniques. We use motor learning literature and analyze the characteristics of typical HCI tasks in VR to identify candidate metrics and establish criteria for a good metric in VR. We then evaluate the metrics in three common VR interaction tasks (object translation, rotation, and selection) employing one mapping type in each (a linear-translational gain [48], a non-linear-rotational gain [27], and a finger-to-arm mapping [80], respectively).

We find that the metrics capture such learning processes that the commonly used metric in HCI—task completion time—does not. We also validate that the metrics demonstrate known characteristics of motor learning, typically with rapid initial learning followed by more gradual improvements over time. We discuss how these metrics offer new insights into learning movement mappings in addition to task completion time. We also provide an open-source toolkit for using them[1]. The proposed metrics can be used to **(1)** analyze and compare movement mapping techniques across multiple motor learning processes and **(2)** inspire and derive strategies to improve motor learning of VR techniques.

2 RELATED WORK

In this section, we first relate VR techniques to motor learning problems by discussing how the techniques map movements and how they are currently evaluated in VR research. We then explain how the techniques fit into the two major paradigms in motor learning. Finally, we review the existing metrics to quantify motor learning.

2.1 Movement Mappings in VR

VR enables "beyond-real interactions" [1] that are impossible in the physical world. For instance, a user can have a third arm [88], a sixth finger [36], and a human tail [75]. They can use arms to control their leg movements [88] or embody themselves as an animal or even a chair [12, 71]. Their arms and fingers can also be extended from their original size/length [55]. These techniques are based on movement mappings that introduce incongruencies between the sensory feedback from the virtual and the physical body. Therefore, the user must learn the mapping.

Besides testing novel interaction concepts, many techniques have been shown to increase user performance in object selection or manipulation tasks [27, 68, 80], ergonomics [86], and to give haptic feedback by redirecting the user's hand to touch a physical object at another location without them noticing [4]. Translational or rotational gains (e.g., Go-Go [63], PRISM [26], and many others [18, 22, 27, 48, 86]) can help users reach faraway targets and enable faster or more accurate manipulations depending on the gain function. With a third arm [88] and multiple hands [68] users were able to increase target selection speed. Long fingers and arms also improved efficiency in selection tasks [55], a translational mapping of the hand enhanced target selection ergonomics [86], and a technique that maps finger movements to virtual arms allowed arm-based interactions in a constrained physical space [80].

As these examples show, the interaction techniques are often evaluated by completion time or accuracy in repetitive object selection or manipulation tasks, or by noticeability when movements are redirected. However, these metrics provide little insight into how users learn to use a technique, and thus why they perform as they do. Therefore, they also fail to inform how the techniques could be designed and improved.

2.2 Motor Learning

Learning VR interaction techniques is primarily relevant to two stages of motor learning: action selection and action execution [44] (the first stage, goal selection, is about the choice of a target that will require a movement, and hence is typically not in the scope of movement mapping techniques). *Action selection* is about identifying the movement that can achieve the goal, and *action execution* concerns performing quality movements to execute the selected action.

User performance evaluation in HCI usually emphasizes action execution. For example, repetitive Fitts' law tasks [74] is considered as *motor acuity* tasks in motor learning—an action, once selected, can be executed with greater speed and precision over practice [44]. When using a mouse cursor to point and select various target locations across the display, the component of action selection is negligible if a user is already used to the input controller (e.g., a mouse with a familiar gain). However, if the movement mapping is changing (e.g., a new gain function is introduced), the role of action selection suddenly increases in learning, as users need to determine how to move correctly, followed by the fine-tuning of the acuity. Despite these differences, action selection and action execution are mostly measured together as a whole, as it is hard to measure action selection directly.

The two major paradigms of motor learning are called motor adaptation and de novo learning. *Motor adaptation* refers to adjusting a well-practiced action to a novel perturbation [38, 44, 84]. For example, in common psychological tasks like visuomotor gain or rotation, participants need to learn to hit targets with a cursor that is sped up or rotated by an angle (e.g., 45°) on the movement direction [45, 62, 84]. The movement mappings in VR techniques corresponding to motor adaptation paradigm are, for instance, scaling the length of an arm [55, 76], or applying translational [48, 63, 86] or rotational movement gains (often called redirection) [4, 18, 27]. In these examples, the mapping is modified from a 1:1 mapping (i.e., the physical world) instead of completely changed.

De novo learning describes the procedure of establishing a new motor controller from scratch [44]. In contrast to motor adaptation, which only demands adjusting a previously mastered skill, de novo learning requires people to formulate appropriate actions based on novel information/feedback. In VR, manipulating a sixth finger [36, 82], a third arm [88], many hands with one [68], arm movements with a finger [80], and operating an animal body with human body motions [71, 75] are closer to de novo learning as the visuomotor associations are somewhat arbitrary/unfamiliar.

A third paradigm from motor learning, *Sequential learning*, stands for combining a set of actions in a temporally organized manner to

[1] https://github.com/Davin-Yu/MotorLearningMetrics4HCI

achieve a task [19, 41]. Examples of this category in VR are selecting an occluded target with multiple discrete steps of dis-occlusions and refinements [94] or swinging a light saber to cut flying cubes in Beat Saber with a continuous movement. Although sequential learning matches such HCI tasks in VR, in this study we focus on the more elementary components of motor learning to understand VR movement mapping.

The two major motor learning paradigms—motor adaptation and de novo learning—provide elementary tasks that are similar to many of the movement mappings in existing VR techniques, as reviewed above. Therefore, we depart from the metrics in motor learning within these paradigms.

2.3 Metrics of Motor Learning

A diverse range of metrics have been applied, especially in behavioral neuroscience and physiology, to quantify motor learning [2, 44, 54, 70, 72]. We roughly classify them into three main categories.

Time-based metrics like *movement time* and *task completion time* are straightforward approximations of learning [35, 72]. Intuitively, people become faster at tasks with practice until reaching the performance limit, which can be approximated by a power function [51, 59].

Error-based metrics are also prevalent in the related literature. *Endpoint error* describes the distance between the target position and a user's actual point of completion [8, 92]. It is often employed in motor adaptation tasks where no visual cursor is available during the movement until completion (i.e., no movement correction possible)—this is to estimate the "user prediction error" in every learning trial [61, 62]. In visuomotor rotation tasks, *Onset error*, the predicted movement direction without correction, can be a replacement when there is ongoing visual feedback [60, 67, 78, 84]. Moreover, other types of errors based on movement trajectory, including *deviation errors* from the desired movement vector [53, 67, 70, 77] and *maximum displacement* [8] have been applied in the studies.

Another set of metrics concerns velocity, acceleration, and jerk profiles throughout the movement. For example, *peak velocity* indicates the initial movement impulse and may correlate with how well a user learns a movement [54, 70, 89]. The overall "smoothness" of the movement can also be assessed via, for example, *numbers of velocity peaks* [14] and *normalized jerk* [10, 34, 70]. These measures evaluate the "goodness" of the whole movement, assuming that an ideal movement is stable (e.g., no sudden acceleration changes).

While metrics of motor learning have been developed in other research fields, many of them cannot be directly applied to HCI tasks, which have unique features such as bounded errors and constant visual feedback (detailed in Section 3.1). Therefore, we propose new metrics inspired by the literature and adjust some existing ones to make them more applicable in the context of HCI.

3 MEASUREMENT CANDIDATES

To develop metrics of motor learning explicit for VR interaction, we first looked into the typical task assumptions in HCI tasks. We then summarized a list of criteria for a good metric, which guides the formulation of five candidate metrics.

3.1 Task Assumptions

While measurements of motor learning have been proposed and applied in the field of motor learning, we identify three key features of a typical HCI task that may make the existing metrics not directly applicable to such a task.

3.1.1 Bounded errors. Unlike many behavioral neuroscience and physiology tasks where the goal is to "be faster and more accurate" over time, HCI tasks typically require users to complete the task under an accuracy threshold throughout the interaction process. For example, users may be required to select a spherical object or a key on a keyboard with a predetermined width [3, 7]. They may also need to move or rotate an object to a specific configuration with a tolerant threshold [56, 93]. Such a kind of binary output (i.e., *within the threshold = success* and *outside of the threshold = fail*) may influence movement behaviors significantly because the "reward" is different, thus affecting the methods for assessing them. For example, users may intentionally decrease their movement speed in favor of a more accurate action, or be only accurate enough (i.e., barely within the target area) to increase speed, depending on the cost of failure [6].

3.1.2 Constant visual feedback. When performing an HCI task with movement mapping techniques, users often receive continuous visual feedback from the environment to help them adjust their movement concurrently to complete the task. For example, they can see the movement of the virtual hand/controller, limbs, and the manipulated object during interaction (e.g., [1, 22, 27, 80]) to determine their next best move/correction. Such a task setting poses challenges in estimating the user's initial prediction error—it makes the motor learning metrics such as *endpoint error* unusable as they were designed for tasks where no visual cursor was available as feedback of the movement.

3.1.3 Variability across trials. In contrast to many motor adaptation tasks that train participants on a perturbation with limited target variations (e.g., single distances) [5, 45, 84], an HCI application may require users to complete tasks with different target configurations while learning a new movement mapping. For example, in the experiments evaluating object selection techniques in VR, the participants are often required to hit targets with different depths in consecutive trials with new mappings [32, 55, 68, 80]. Ideally, a metric should still be able to capture the learning progress even with changing task difficulties (e.g., distances and depths).

3.2 Criteria for a good metric

Following the three task assumptions, we determine five criteria for assessing the goodness of a metric for motor learning of movement mappings in VR.

- Validity: the metric complies with known characteristics of motor learning in a learning environment. The motor learning process typically starts with rapid improvements followed by diminished returns until reaching a performance plateau, which also suggests that "errors" during motor learning should decay over trials [32, 44, 73, 87]. We can approximate this process with

a power function[2]:

$$y = a \cdot x^{-k} + b \tag{1}$$

where y is the data based on the evaluation metric; x is the amount of practice (e.g., trials); k is the learning rate ($k > 0$); a is a scaling factor (as the power term x^{-k} is between 0 and 1); and b is the asymptotic, representing the performance limit ($b > 0$). A metric is invalid if it does not follow known learning characteristics (e.g., error increases over practice) or shows no learning process in a simple learning environment designed for motor learning.

- Informativeness: the metric provides detailed insights into how users learn/adapt to a movement mapping. For example, it would be ideal to understand the adaptation strategy in the learning process; the users may become quicker at task completion due to a more accurate first move, or due to a coarse first move but with fast correction movements. Understanding the strategies could help refine existing designs to offer dedicated support, such as giving augmented feedback for either effective ballistic movements (prediction) or the later correction movements [47].
- Generalizability: the metric remains applicable and informative within various task scenarios and new movement mappings. The evaluation can be accomplished by assessing metrics within more intricate task scenarios and movement mappings. A metric is less generalizable if it becomes invalid in scenarios with more complicated tasks and movement mappings. Additionally, an ideal metric should be sample size independent, meaning the metric should remain helpful regardless of the sample size (e.g., works for individual users).
- Resiliency: the metric is robust across different task difficulties. For example, task completion time may have poor resiliency as it is easily influenced by target widths and distances, as predicted by Fitts's law [24, 74]. Such a metric may still be helpful in a controlled experiment where difficulty levels are crossed, and participant performance is averaged. However, it might be more challenging to deploy it in scenarios where the difficulties during the learning process can differ dramatically (e.g., a user employs the Go-Go technique to retrieve objects situated at varying depths) to assess the learning progress in real-time.
- Autonomy: The metric assesses trial-by-trial learning independently, without the need for additional baseline information. For example, the metric is not supposed to estimate the future parts of the learning process (e.g., the mean peak velocity after enough practice [45]). It should also not rely on, for instance, the predicted task completion time, as this information is only sometimes available.

3.3 Candidates

We propose the following five candidate metrics to quantify the learning process of movement mappings. *Task completion time* is considered the baseline. *Refinement time proportion* and *refinement space* quantify the user prediction error. *Normalized path error* and *normalized jerk error* measure the closeness of the current movement to an ideal movement. The new metrics aim to provide additional insights into how users are learning (informativeness) and can quantify trial-by-trial learning (autonomy). In the following, we provide the rationale for building the metrics while we refer readers to Appendix A and our open-source toolkit for the technical aspects of metrics implementations.

[2]In the supplementary material, we also use an exponential function [33] and a dual-exponential function (representing a combined slower and faster learning process) [32, 73] to provide alternative views on how the learning process can be approximated.

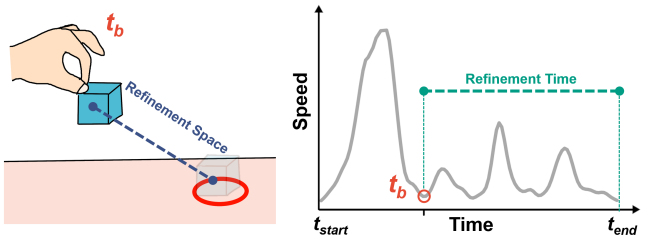

Figure 1: The refinement space and refinement time are determined by the correction movements observed in the entire movement.

3.3.1 Task completion time. Task completion time (*Time*) refers to the time taken to complete a specific task. The changes in task completion time over sessions/trials have been used as a key usability measure to quantify an interface's learnability in HCI [35, 40]. The metric provides limited understanding apart from the user getting quicker at the task over time. However, task completion time is known to be robust under different learning scenarios (i.e., the power law [51, 59])—we, therefore, consider it as a baseline.

3.3.2 Refinement time proportion. Refinement time proportion (*RTP*) is the proportion of correcting/refining movements from the total movement time. We aim to use this metric to quantify the correctness of a user's initial movement prediction.

In motor learning literature, one attractive error-based metric is *endpoint error* [61, 62], which assesses the "user prediction error" with no possible corrections (e.g., without any feedback during the movement). However, correction movements are inevitable in HCI tasks which typically provide constant visual feedback (e.g., in the form of a cursor or an avatar hand). Similar metrics that worked under persistent visual feedback—the *onset error* [60, 67, 78, 84]—is only usable for visuomotor rotation tasks to estimate the errors in the initial movement direction.

We thus formulated *RTP* because of the unique features in HCI tasks. Selection and manipulation movements are often composed of two phases: the ballistic phase which is the initial movement impulse to get closer to the target, and the refinement phase which consists of one or more slower adjustments for fine-grained selection or manipulation [25, 46, 49]. With *RTP*, we assume that the ballistic phase represents the initial user prediction without correction movements, and this prediction should become more accurate with learning and practice [65]. This view essentially corresponds to the Bayesian perspective of learning—a user's internal estimation can get more accurate over practice, given the inspected prediction errors in each trial [42, 45].

Suppose the ballistic phase ends in time t_b and the total completion time is t_{end} (see Figure 1), *RTP* is calculated as:

$$RTP = (t_{end} - t_b)/t_{end} \tag{2}$$

RTP takes the refinement proportion rather than the ballistic proportion because it is an error-based metric.

3.3.3 Refinement space. Refinement space (*RS*) is the Euclidean distance between where a user starts the refinement phase and where the user finishes the whole movement (see Figure 1). The concept is similar to *RTP* but considers the ballistic and refinement phases from a spatial perspective; *RS* captures a user's spatial prediction error in the ballistic movement (i.e., without correction).

RS aims to address a limitation of *RTP*: *RTP* might depend on the target size. Intuitively, users often have little incentive to fine-tune their movement if their action has already resulted in a correct execution, for instance when the pointer is within a large target [92]. In contrast, smaller targets may require more refinement to ensure accuracy, leading to a higher refinement proportion. To overcome this, *RS* assumes that users should be more precise at calibrating their ballistic action to reach as close as possible to the point they are targeting in space. Unlike *RTP*, *RS* gives an absolute distance error to eliminate the influence of the "proportion" of the correction movement. Additionally, the metric is intended to remain consistent irrespective of target sizes, as it assesses the proximity to the movement's endpoint rather than the target's center.

3.3.4 Normalized path error. Normalized path error (*NPE*) represents the percentage of the "extra movement", as compared to the ideal shortest path, in the whole movement (see Figure 2 left). It is based on the concept of quantifying the errors in the whole movement trajectory [8, 67], defined by the following equation:

$$NPE = (L_a - L_i)/L_a \tag{3}$$

where L_a is the actual movement distance, and L_i is the ideal shortest movement distance. Because HCI tasks typically have bounded errors, the shortest distance should be considered as if the user is aiming at the closest edge of the target boundary (not the target center), as users might not leverage the whole available width of the target [92]. Otherwise, the *NPE* might become negative. This metric quantifies if the user optimizes their movement over practice to align with the most optimal path. Essentially, this metric poses a simple optimal control assumption, where the goal is to minimize the distance to travel from the start to the target by avoiding unnecessary variances [32].

3.3.5 Normalized jerk error. Normalized jerk error (*NJE*) measures the "extra jerk" applied in the current movement, compared to the ideal movement governed by the minimum jerk assumption [23, 28].

Jerk is the change in acceleration, the third time derivative of position, which should ideally be minimized to ensure the smoothness of a movement. The following equation from Teulings et al. [10, 79] calculates *normalized jerk* of a movement trajectory, which is comparable across movement distances and durations:

$$N\mathcal{J} = \sqrt{\frac{1}{2} \cdot \int_{t_{start}}^{t_{end}} j^2(t)dt \cdot \frac{D^5}{L^2}} \tag{4}$$

where t_{start} is the movement starting time, t_{end} is the movement ending time, $j^2(t)$ is the squared jerk, D is the movement duration,

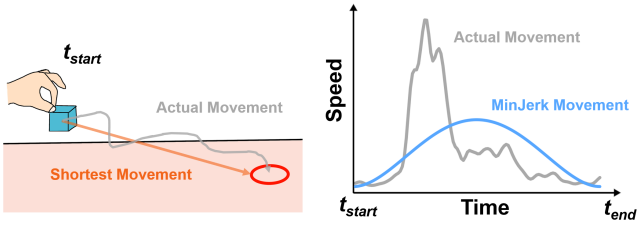

Figure 2: Normalized path error quantifies the "extra movement distance" compared to the ideal shortest path (left). A jerk-minimized movement, described by a 4^{th} order polynomial, covers the same distance within the same time as the actual movement (right).

and L is the movement length/distance. We used *normalized jerk* instead of the original jerk because it cancels out the effect of movement distances and durations, so it is ideally resilient across difficulty levels [79]. However, the absolute value of *normalized jerk* is hard to interpret (e.g., $N\mathcal{J} < 50$ in one study [79] while $N\mathcal{J} > 2{\times}10^4$ in another [81]). This motivated us to build a percentage-based metric that needs the ideal *normalized jerk* as a reference.

We assume an ideal movement covers the same distance within the same period of time as the actual movement; however, it contains only one velocity peak, and the jerk during the movement is minimized. In this case, we avoid inferring about ideal movement time for a given distance (AUTONOMY) but focus exclusively on the smoothness of the movement. Furthermore, our assumption posits that the optimal motion comprises solely a ballistic motion, so it imposes penalties for any jerk introduced during corrective movements.

We can describe an ideal, jerk-minimized movement using a symmetric, bell-shaped movement speed profile, described by a 4^{th} order polynomial [28] (see Figure 2 right):

$$v_i(t) = h \cdot (t - t_{start})^2 \cdot (t - t_{end})^2 \tag{5}$$

here, $v_i(t)$ is the speed of the ideal movement at timestamp t, and h is a free parameter to be fit. The parameter h can be calculated if we fit in the actual movement distance (L_a) and time period (from t_{start} and t_{end}), as according to our assumption, an integral of $v_i(t)$ equals to L_a. With t_{start}, t_{end}, and calculated h, we can then generate the speed profile (thus the jerk profile) for the ideal movement, and calculate the ideal *normalized jerk* $N\mathcal{J}_i$ with Equation 4. With the actual speed profile, we can also find the actual *normalized jerk* $N\mathcal{J}_a$. Finally, *NJE* is calculated as:

$$NJE = (N\mathcal{J}_a - N\mathcal{J}_i)/N\mathcal{J}_a \tag{6}$$

NJE assesses how closely the smoothness of the current movement matches the ideal condition of jerk minimization.

4 EVALUATION OF THE METRICS

We designed a user study to evaluate the VALIDITY, GENERALIZABILITY, and RESILIENCY of the proposed metrics in VR interaction tasks. In this section, we describe the task scenarios and the study protocol for evaluating the metrics.

4.1 Task Scenarios

This section presents a high-level overview of the task scenarios (more details in Appendix B.1). The scenarios are illustrated in Figure 3. They are:

- An object translation task with a linear-translational gain movement mapping (LT Task).
- An object rotation task with a non-linear-rotational gain movement mapping (NLR Task).
- An object selection task with a multi-joint finger-to-arm movement mapping (FA Task).

4.1.1 LT Task: Linear Translational Gain. The task requires the participants to move a 3D cube from a starting position to a target position on the table (see Figure 3 left). The target can appear in different depths and directions.

The participants complete the tasks in two mappings: one involves a virtual hand using a 1:1 physical to virtual mapping, while the other utilizes a virtual hand with a linear translational gain in the depth dimension (the forward z-axis) [48]. The new mapping requires motor adaptation.

We anticipate the LT task to be the easy to learn. Thus, we employ this task to evaluate the VALIDITY of the metrics, as the learning trend is expected to be evident in this straightforward learning environment.

4.1.2 NLR Task: Non-linear Rotational Gain. The task requires the participants to rotate a 3D bunny to a target orientation (see Figure 3 middle). The position of the bunny is fixed to eliminate the need for object translation, which is covered in the LT task. The required rotation angles and directions can be varied.

The task contains two mappings: a 1:1 mapping and a non-linear rotational gain mapping. We replicate a previous technique [27], where the virtual hand rotation can be amplified or attenuated based on the movement speed of the physical hand during manipulation. The new mapping necessitates motor adaptation.

We use this task to assess the GENERALIZABILITY of the metrics by representing a scenario where the task is difficult (requires mental rotation). This task will reveal whether the metrics are sensitive enough to capture non-instant learning.

4.1.3 FA Task: Finger-to-Arm Mapping. The task requires users to select a target from a starting position (see Figure 3 right). The target can appear at different depths, distances, and heights.

We apply two mappings: a 1:1 finger-to-finger mapping and a multi-joint finger-to-arm technique called *FingerMapper* [80]. *FingerMapper* determines the movement of the virtual arm by remapping index finger extension or retraction.

FingerMapper requires de novo learning, as a new motor controller of the virtual arm needs to be established. We use this task to assess the GENERALIZABILITY of the metrics to represent a scenario where the mapping is complex.

4.2 Study Protocol

4.2.1 Participants. We recruited 16 participants (6 women, 10 men) with a mean age of 27.3 (SD = 5.1). Their self-rated familiarity score with VR was 4.5 on average on a 7-point Likert scale. They all reported having normal or corrected-to-normal vision.

4.2.2 Devices. A standalone Oculus Quest 2 headset was used to immerse the participants in VR. The application was developed with Unity C#, using Oculus hand tracking and interaction modules.

4.2.3 Design. The study employed three interaction tasks requiring different movement mappings. Each task contained four learning phases that were presented in the same order to evaluate the metrics in different learning conditions.

(1) *Training*: 1:1 mapping was employed for the participant to get used to the interaction task.
(2) *Adaptation*[3]: the new movement mapping corresponding to each task was used for the first time.
(3) *De-adaptation*: the mapping returned to 1:1. This phase allowed us to identify if there was a need to de-adapt when returning to the standard mapping from the movement mappings (i.e., the aftereffects).
(4) *Re-adaptation*: the movement mapping technique was applied again. We aimed to use this phase to measure the "savings" of the adaptation phase and see whether it would make users to adapt to the mappings faster than when using them for the first time.

Within each phase, there were thirty learning trials varying across three *difficulty levels* and five *variance levels*. The difficulty levels constructed apparent task difficulties with different movement distances, and the variance levels produced deviations in movement directions. The trial order was semi-randomized in the way that all difficulty levels would appear in consecutive chunks (i.e., Trial 1-3, Trial 4-6, Trial 7-9, etc.), and the variance levels were randomly assigned to each trial at the same frequency. Such a design imposed fluctuations to tasks so that the participants could learn the new movement mappings rather than a particular movement itself with a fixed target configuration. The difficulty and variance levels were piloted with four participants who did not attend the formal study to ensure the tasks were not too easy nor too challenging. The detailed parameters are documented in Appendix B.2.

In sum, the study consisted of 360 trials (3 tasks × 4 phases × 3 difficulty levels × 5 variance levels × 2 repetitions) for each participant.

4.2.4 Procedure. The experiment lasted approximately 50 minutes. The participants were first invited to fill in a demographic questionnaire and a consent form and were informed that they would learn three new ways of interaction in VR. We then provided them with general instructions on conducting the experiment, such as completing the task as fast as possible and correctly performing a pinch gesture.

After that, they watched a video on how an expert finished LT Task with/without movement mapping. With this, we aimed to teach the participants the main concept of the movement mapping technique. They could ask questions about the technique and the task, if any. After the video and the clarification, the participant put on the VR headset, and re-centered and calibrated the table height to ensure comfortable positioning. The virtual table was also calibrated based on a physical table to support users during the tasks, which could decrease the interaction fatigue as compared to

[3]Note the "adaptation" phase here is different from "motor adaptation", which is a learning paradigm. Same for the phase names below.

Figure 3: Task scenarios: object translation with linear-translational gain (left), object rotation with non-linear-rotational gain (middle), and object selection with multi-joint finger-to-arm mapping (right). The blue outline indicates the physical hand position (not displayed during the task).

mid-air input [13, 83]. After the calibration, the first task started. The switch between the techniques (i.e., from the 1:1 mapping to a new mapping, or vice versa) was indicated by a short "ding" sound to mentally prepare participants for the adaptation. After completing the first task, the participants were required to take off the headset and to take a break. The participants were also allowed to take a break before any trial as long as they felt fatigued.

We then repeated video, clarification, and task-performing procedure for NLR Task and FA Task in a fixed order. The participants were compensated with drinks and snacks equivalent to 20€ after finishing the experiment.

4.3 Analysis Overview

The collected movement trajectory data were processed and fit into the power function (Equation 1) for evaluating VALIDITY and GENERALIZABILITY. We used both R^2 and *RMSE* to assess the goodness-of-fit. We also applied linear mixed models to evaluate RESILIENCY. Detailed data processing strategies can be found in Appendix C.

4.3.1 Result Interpretation. R^2 is scale-independent and can be compared across different metrics and phases. Generally, larger R^2 values indicate that the learning effect described by the power function in Equation 1 is better captured with the metric under evaluation. We refer to a rule of thumb 'large' effect threshold in behavioral science $R^2 > 0.25$ to determine whether the learning seems apparent [15]. When R^2 is between 0.09 and 0.25, denoted as 'medium' to 'large' effect threshold [15], we interpret the metric data as weakly indicating learning behavior. Within this range, we recognize that non-learning-related noises contribute significantly to data variances, which could imply that the learning component is relatively small. Finally, we interpret R^2 values smaller than 0.09 as the metric being unable to demonstrate motor learning. *RMSE* is scale-dependent, which provides supplementary information when comparing results within each metric and for phases with the same movement mapping (i.e., training vs. de-adaptation or adaptation vs. re-adaptation). Linear mixed models are used to determine whether task difficulties exhibit significant effects on metrics data ($\alpha = 0.05$). In the following, we describe the main results based on VALIDITY, GENERALIZABILITY, and RESILIENCY (Section 4.4-4.6). We will focus on INFORMATIVENESS in Discussion (Section 5).

4.4 Validity

A metric is invalid if error increases over practice (coefficient a in Equation 1 is smaller than 0) or shows no learning process in a simple learning environment (e.g., $R^2 < 0.09$ as in mentioned in Section 4.3.1). Overall, all the metrics exhibited a robust adherence to the known attribute of motor learning described by the power function—the process starts with rapid initial learning followed by more gradual improvements over time (see Figure 4).

Performance over *Time* is often treated as the standard metric for learning in HCI. While the metric had strong fitting results for phases with a 1:1 mapping ($R^2 = 0.89$ and 0.87), it produced moderate results for phases with new movement mappings ($R^2 = 0.52$ and 0.31). This indicated that the data variances in the adaptation and re-adaptation phases were high, covering up some parts of the learning process the metric was supposed to demonstrate.

RTP and *NJE* led to very similar performance pattern as *Time*: they both showed good fitting results for phases with a 1:1 mapping (R^2 between 0.72 and 0.91); however, the fitting performance diminished notably during phases involving novel movement mappings (R^2 between 0.41 and 0.53). The data variances in *Time*, *RTP*, and *NJE* could be due to the effective size of the target being smaller in the new movement mapping than the 1:1 mapping (given the same visual size but different movement gains). More difficult task conditions resulted in more noises in the metric data, indicating that the metrics could be susceptible to changing difficulties.

RS demonstrated the most robust performance across the four phases of learning with all $R^2 > 0.80$. The result indicated that the participants were indeed getting more accurate in matching their ballistic movement to their targeted position, and the noises that could impact the learning illustrated in this metric are relatively minor.

NPE achieved good fitting performance in training, adaptation, and re-adaptation phases ($R^2 = 0.81$, 0.91, and 0.75). The $R^2 = 0.14$ for de-adaptation was unexpectedly low. The error was indeed going down over practice but missed the rapid improvement part.

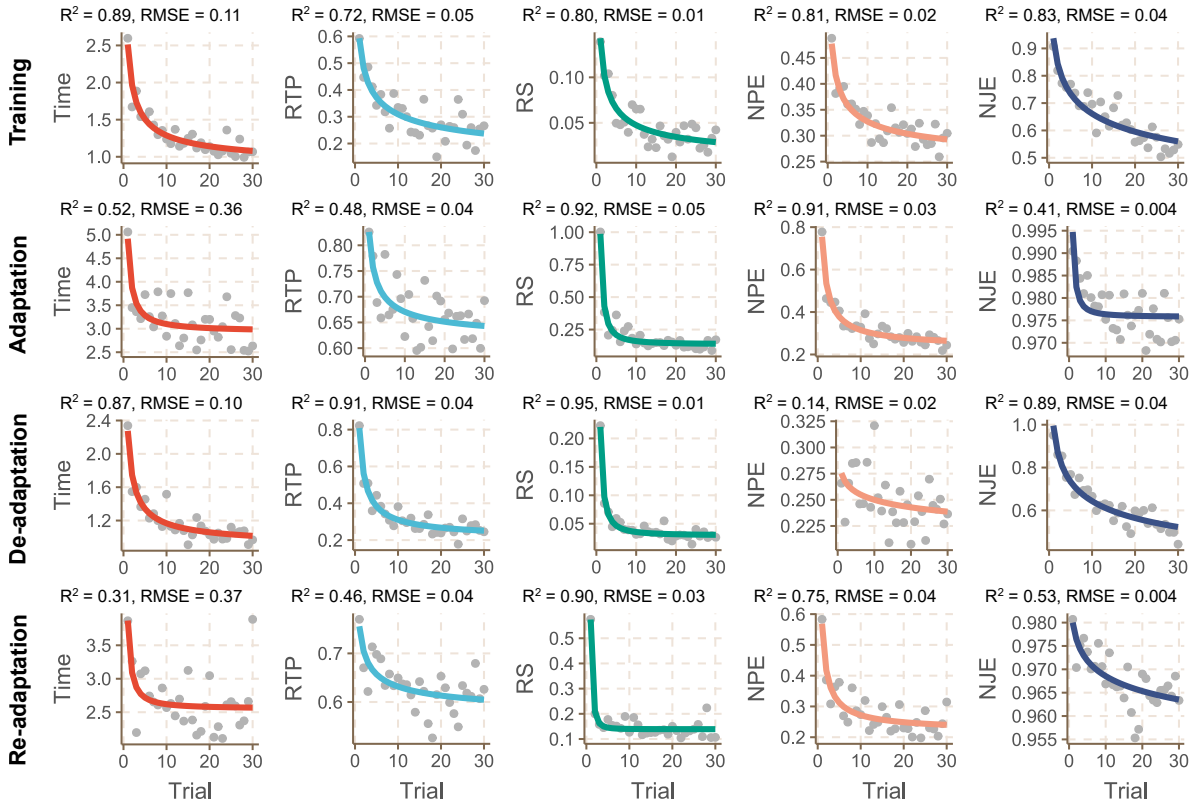

Figure 4: Fitting results of the power function in LT Task (linear translation). Data points are averaged across participants.

The *RMSE* was as low as the one in the training phase (both *RMSE* = 0.02). One possible explanation was the "saving" effect [20, 43]: participants had learned to move closely to the shortest path during training and did not need to completely re-learn it in de-adaptation. Because the learning was small and difficult to capture, the noises played a significant role in the data, leading to inferior R^2 (see Appendix D where we concatenated training and de-adaptation data).

In summary, all metrics were shown to be valid in LT Task (i.e., a simple learning environment). The learning process demonstrated by *RS* seemed most apparent. *Time*, *RTP*, and *NJE* also adhered to the power function closely in phases with 1:1 mapping and illustrated sufficiently clear learning trends when the new movement mapping was applied. *NPE* was helpful for both mappings and indicated potential savings in the de-adaptation phase.

4.5 Generalizability

We validated the metrics in a simple learning environment (LT Task). We then extended our analysis to scenarios with more complicated tasks and movement mappings, where the learning could be smaller and more challenging to capture. As expected, the overall fitting performance to the power function exhibited a significant decrease for all the metrics in NLR Task (non-linear rotation) and FA Task (finger-to-arm mapping), as shown in Figures 7 and 8 in Appendix D. We also evaluated the fitting results at the individual level in the adaptation phase of LT Task, as the data were supposed to be noisier than the averaged data. Figure 5 showcases two randomly selected individuals as samples.

4.5.1 NLR Task. *Time* was able to capture learning in training and adaptation (R^2 = 0.44 and 0.54), but not de-adaptation and re-adaptation (both $R^2 < 0.02$). *RTP* weakly disclosed the learning trend during training, adaptation, and de-adaptation (R^2 = 0.20, 0.15, and 0.12). Remarkably, *RS* was still able to illustrate rather clear learning processes in adaptation and re-adaptation (R^2 = 0.40 and 0.39), and revealed weak learning trend in training and de-adaptation (R^2 = 0.14 and 0.13). *NPE* provided weak fitting results in training, adaptation, and re-adaptation (R^2 = 0.16, 0.28, and 0.13), while became invalid in de-adaption (power function $a < 0$). *NJE* failed to produce meaningful learning information, perhaps because the noises were relatively too large across the four phases (all $R^2 \leq 0.09$).

4.5.2 FA Task. The fitting outcomes across the metrics were overall better for FA Task than NLR Task. The learning processes uncovered by *RS*, *NPE*, and *NJE* seemed to be relatively apparent in general. In particular, *RS* illustrated the learning especially well in training and de-adaptation (R^2 = 0.65 and 0.39). The data in adaptation and

re-adaptation resulted in a weaker fit ($R^2 = 0.22$ and 0.17). *NPE* appeared to be more consistent over the four phases ($R^2 \in [0.25, 0.50]$), so did *NJE* ($R^2 \in [0.20, 0.47]$). *Time* achieved better performance in training, adaptation, and de-adaptation ($R^2 \in [0.28, 0.49]$), similarly for *RTP* ($R^2 \in [0.21, 0.25]$). However, *Time* and *RTP* only captured the learning process in re-adaptation to a limited extent ($R^2 = 0.1$ and 0.13).

4.5.3 Learning in individuals. The mean values of R^2 and their corresponding standard errors at the individual level were 0.14 (s.e.=0.03) for *Time*, 0.08 (0.02) for *RTP*, 0.53 (0.06) for *RS*, 0.38 (0.04) for *NPE*, and 0.10 (0.02) for *NJE*. The results from two randomly selected individuals aligned with the summary statistics (Figure 5). *RS* and *NPE* demonstrated evident learning trends in both participants ($R^2 \in [0.24, 0.59]$). *Time* and *NJE* also captured learning for P9 ($R^2 = 0.27$ and 0.21) but not for P15 ($R^2 = 0.02$ and 0.01). The data in *RTP* were quite noisy for both participants to conclude learning (both $R^2 < 0.07$).

4.5.4 Summary. In contrast to a simple learning environment like LT Task, learning could be smaller and more difficult to capture in complex scenarios with a challenging task (NLR Task) and movement mapping (FA Task). Our results indicated that all metrics were, to some extent, obscured by noises that were not a part of the learning process described by the power function (i.e., lower R^2 fitting compared to the ones in LT Task).

We found *RS* was the most robust indicator of learning across all conditions, and its fitting results were often less influenced by data noises (i.e., high R^2). *NPE* also worked in most conditions, with one failure case (NLR Task, de-adaptation). The learning process demonstrated by *Time*, *RTP*, and *NJE* could be corrupted by large noises, and sometimes the learning trend was completely obscured. We recommend recruiting more participants to try to stabilize the noises in the data or administer fewer difficulty levels (see Section 4.6 below). More learning trials may also help uncover a more distinct learning pattern within a challenging learning environment. Additionally, our results indicated that *RS* and *NPE* could be promising for analyzing movement learning in individuals.

4.6 Resiliency

In LT Task, the linear mixed models showed that there was a significant effect of *difficulty levels* on *Time* ($p < .001$, 95% confidence interval of the estimated coefficient, i.e., CI: [3.10, 3.38]), *RTP* ($p < .001$, CI: [2.38, 2.72]), *RS* ($p < .001$, CI: [2.29, 2.65]), and *NJE* ($p < .001$, CI: [3.28, 3.52]). The impact of *difficulty levels* on *NPE* was not statistically significant, with a close-to-zero coefficient ($p = .11$, CI: [-0.10, 0.28]). In NLR Task, *difficulty level* was a significant determinant for all metrics (all $p < .001$) except *NPE* ($p = .40$, CI: [-0.21, 0.08]). In FA Task, the effect of *difficulty level* was significant on all metrics (all $p < .001$). We refer interested readers to more detailed statistical test results in the supplementary materials. Figure 9 in Appendix D also visually illustrates how metrics were influenced by *difficulty levels*.

Overall, our results indicated that *NPE* (the percentage of 'extra movement' compared to the ideal shortest path) remained relatively resilient in the LT Task and NLR Task, but not in the FA Task. Contrary to our hypothesis, *RS* and *NJE* did not demonstrate empirical resilience. Additionally, *Time* and *RTP* were found to change due to variations in difficulty levels. These findings suggest that the current metrics may not be suitable for scenarios where the difficulties during the learning process can vary dramatically. While they were shown to be helpful in a controlled experiment where difficulty levels were systematically varied and participant performance was averaged (in our study), one should anticipate variances in the averaged data resulting from different difficulty levels.

5 DISCUSSION

In the previous sections, we demonstrated that all the proposed metrics are valid for motor learning in a simple environment (VALIDITY). Several metrics, particularly *RS* and *NPE*, illustrated their generalizability to more complex task scenarios and movement mappings (GENERALIZABILITY). Additionally, we found all metrics, with some exceptions in *NPE*, were likely to be influenced by task difficulties and are thus more suitable in a controlled experiment (RESILIENCY).

In this section, we focus on INFORMATIVENESS by discussing new insights brought by these motor learning metrics in addition to task completion time and how they can help us analyze and design interaction techniques in VR and HCI.

5.1 Identifying Movement Learning with Metrics

The metrics demonstrated promise in revealing learning behaviors in user movement which task completion time—a common metric to quantify learning in HCI—could not capture. They provide two immediate benefits.

5.1.1 Identify learning behavior. In individual results (in Figure 5) and NLR Task results (in Figure 7), we found *Time* was not able to identify the learning process in the adaptation and re-adaptation phases. However, it was clear from *RS* and *NPE* that users were still learning the techniques to perform a more accurate initial movement within the first few repetitions and to follow a more optimal movement path over the whole practice session. As another example, in the de-adaptation phase of the NLR Task, while data on *Time* seemed converged, *RTP* and *RS* revealed that there was still a weak learning process going on over trials.

Because of such adaptive/learning behaviors, there must be associated mental or motor costs (e.g., cognitive load, memory resources, attention, physical fatigue) that were not encapsulated by task completion time but should not be ignored when designing or deploying new interaction techniques [16, 52, 91]. The new metrics can function as an evaluative tool to ascertain the presence of learning behavior.

5.1.2 Determine source of motor learning. The results suggested that multiple motor learning processes were happening simultaneously, as the learning rate could vary significantly with different metrics. For example, referring back to Figure 4, where *RS* showed abrupt shifts in the first few repetitions, the improvement on *RTP*, *NPE*, and *NJE* seemed more gradual in general. Motor learning literature provides potential explanations for these results: action

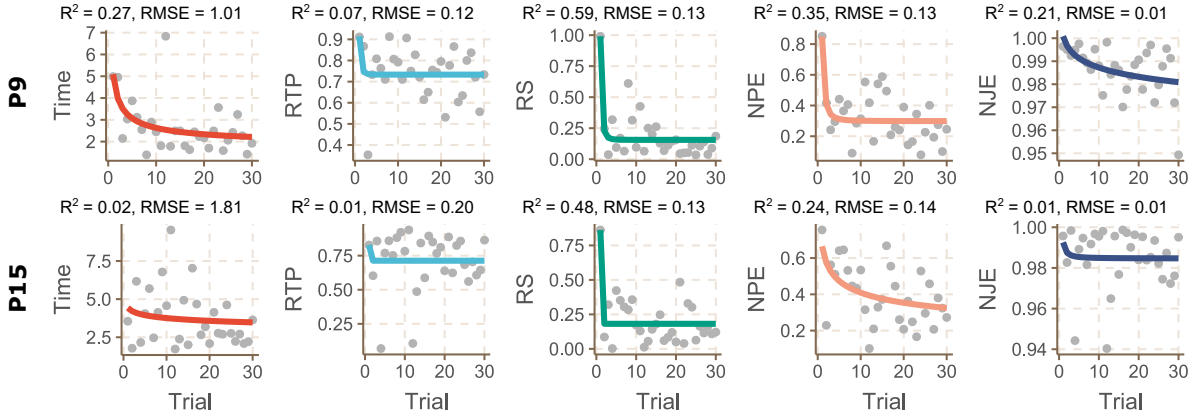

Figure 5: Fitting results of the power function in LT Task regarding two individuals (P9 and P15) in adaptation.

selection and action execution are both essential but distinct components in the motor learning pathway [44]. While *RS* can be more relevant to action selection (i.e., determining how to move to achieve a selected goal), *RTP*, *NPE*, and *NJE* could quantify better the quality of action execution (i.e., motor acuity). In contrast, *Time* could be a metric encompassing all possible learning stages—it provides a simple outcome, but without offering enough granularity to analyze learning in-depth.

Additional insights can be obtained by referring to individual results in Figure 5—P9 and P15 might have different learning speeds for the same learning component. P9 rapidly acquired the ability to align with the optimal path and achieved convergence, whereas P15 gradually developed this capability throughout the entire learning phase. P9 appeared to be fine-tuning the movement smoothness while P15 did not (i.e., *NJE* did not demonstrate learning in P15). These individual differences may necessitate different methods to facilitate learning a new movement mapping. Overall, the new metrics help determine the source of motor learning more precisely, thus guiding us to improve the techniques—we will discuss specific improvement strategies in Section 5.3.

5.2 Patterns in Learning Movement Mappings

From the metrics results, we infer how users learned the movement mappings over different learning blocks.

5.2.1 Adaptation. When introducing a new movement mapping in VR, there is a need for motor learning in addition to cognitive learning (e.g., facilitated by the video introduction and oral explanation in our study). Users need to get accustomed to the new mapping by forming a more accurate initial movement impulse (i.e., the ballistic movement), finding a more optimal path for achieving the task, and familiarizing themselves with the new movement (i.e., smoothing their movement trajectory). Some learning processes could happen within a few repetitions (e.g., aligning the ballistic endpoint with the target), while others may require more trials and errors (e.g., optimizing the path in LT and NLR Task). Eventually, the learning may result in a decreased task completion time. These requirements of initial learning were consistent from the metrics data and aligned with other experimental results documented in the literature (e.g., [45, 61]).

5.2.2 De-adaptation. When transitioning back to a 1:1 mapping, users might need to undergo a "re-learning" process to perform optimally—during the de-adaptation phase, initial performance is often inferior to the last few trials in the training phase, where users became accustomed to the 1:1 mapping. This is called "the washout effect" in motor learning [44, 84]. Notably, from the *RTP* and *RS* results in LT Task (Figure 4), users' initial movement prediction was even worse in the de-adaptation block than in the training block (i.e., when they first put on the VR headset to learn the task)[4]. One comment from a participant also resonated with this learning effect: "*Wait, this was slower than normal. You did something to my hand.*" Despite being explicitly told that the de-adaptation block was returning to 1:1 mapping, the participant inaccurately predicted that a slower-than-normal mapping needed to be learned. Therefore, it is crucial to consider the washout effect when rapidly switching between VR applications with/without movement mappings.

5.2.3 Re-adaptation. When re-learning a movement mapping, the user has to restart the learning processes (i.e., improving ballistic movement, path, and movement smoothness). However, our results showed that the users were almost always better in the first few attempts[5]. This effect is partially attributed to the "savings" in the motor learning literature—reacquiring learned skills is typically faster than learning it the first time [20, 43]. The results triggered some interesting questions for future work: How much can movement mapping techniques be "memorized" with long-term practice? Can we lower this friction when re-adapting to a known (learned)

[4]Student's t-test revealed that the first trial of de-adaptation had significantly higher *RTP* ($p = 0.007$) and *RS* ($p = 0.002$) than training. Assumptions of t-test, including normality and homogeneity of variance, were supported by the Shapiro–Wilk test and Levene test.

[5]Mann-Whitney tests suggested that the first trial of re-adaptation had significantly lower *RS*, *NPE*, and *NJE* (all $p < .02$) than adaptation but not *RTP* ($p = .32$) in LT Task. We chose a non-parametric test because the normality assumption was violated, as indicated by the Shapiro-Wilk test.

movement mapping in VR? The proposed metrics are essential steps to understand the motor learning behavior for such re-adaptations.

One notable point is that we instructed participants to complete the task as quickly as possible, and they could only proceed to the subsequent trial if they completed the task correctly. These instructions and study settings may impact the convergence speed of metrics, especially when a future study prioritizes different speed and accuracy trade-offs.

5.3 Advancing Movement Mapping Techniques

The metrics also inspired us to reflect on how to improve movement mapping techniques or provide appropriate feedforward/feedback methods to help learn these new mappings.

5.3.1 Optimizing learning curves in metrics. Three general strategies can be inferred from the learning curves ($y = a \cdot x^{-k} + b$). First, we could lower the bar of the initial adaptation (i.e., the first attempt where the trial number $x = 1$). For example, instead of providing a "ding" sound and switching to the new mapping abruptly, additional feedback, such as visual guidance [17], could be presented to users to better prepare them for the new movement mapping. Motor learning literature refers to a connected phenomena as parallel adaptation to multiple motor programs [61, 84]. An every-day example of it is using a different mouse (and a cursor gain) at home and at work. The home and work environments give a cue between two distinct but familiar, learnt mappings, and thereby adapting to those is immediate. Second, we could improve the learning rate k by, for instance, informing users about their movement results (e.g., endpoints) each trial to provide them with a more comprehensive understanding of how to adapt subsequently [8, 92]. Third, we could also try to break through their performance limits (b) by showing a concurrent animation of an expert's movement to reinforce them to execute more optimized movements [47].

5.3.2 Optimizing motor learning processes. Specific strategies can be designed based on the metrics. For example, by knowing that a user is constantly under/overshooting a target with the ballistic movement (from *RS*), we could dynamically adjust the target position or the cursor position (i.e., the movement mapping function) to make the target easier to achieve with the first movement impulse. We could also visualize the optimal movement path (e.g., showing a straight line between the virtual hand and the target position), knowing users want to decrease *NPE* so that users could follow the optimal path along the way. Such feedback could help, in particular, the refined movement phases. In contrast, the ballistic movement can benefit more from initial cues (i.e., feedforward [57]) or adaptive changes in the mapping, like discussed above.

However, we also want to note that such augmented feedback may make users too dependent on them and decrease retention [52, 66]. Therefore, these should be applied cautiously and possibly considering fading them away during practice. Altogether, we believe the metrics can inspire the community to devise better VR interaction techniques requiring a lower movement learning cost.

5.4 Limitations and Future Work

We want to emphasize several limitations of the current study and identify additional avenues for future research.

First, the presented metrics should not be perceived as an exhaustive list for quantifying motor learning in HCI. These metrics are the ones we deemed most reasonable and straightforward from the literature. Other options still exist, and specific components of the metrics are switchable. For example, we could replace the optimal path or speed/jerk profiles with assumptions from other optimal control models [23].

Second, while we took precautions in controlling user fatigue in the experiment (e.g., arms on the table, force breaks between tasks, and optional breaks between trials), the influence of fatigue was inevitable during the study. The fatigue effect could interplay with the learning effect. One possible future solution is to use models that simulate fatigue [11, 39, 85] to tease out the effect of learning.

Third, visual tracking errors could play a role in the data. To mitigate their impact, we instructed participants to perform the pinch in front of the camera and also smoothed the collected data before processing. In future applications, employing similar strategies when utilizing these metrics might be beneficial.

Finally, our initial motivation was to propose metrics for movement mapping techniques in VR on object selection and manipulation. However, we speculate that the metrics may also work for other interaction techniques, such as redirection techniques (which may apply more subtle movement mappings in VR) [29, 30] or cursor movement-based techniques like Bubble Cursor [31]. The metrics can also provide empirical ground for reinforcement learning-based simulation models [37, 58]. We thus release an open-source toolkit to help future research test the metrics' applicability and evaluate new techniques in various contexts.

6 CONCLUSION

We have presented metrics of motor learning for analyzing and improving VR techniques that introduce movement mappings. The metrics were proposed based on the motor learning literature while considering specific HCI task requirements. The metrics also aimed to bring insights into how users learn a new movement mapping regarding their initial movement prediction, the closeness to an optimal path, and the smoothness of the movement trajectory.

Our findings have shown that the new metrics captured learning behaviors that task completion time, a common metric for assessing learning in HCI, cannot. The metrics also illustrated behavior changes when users first learned a new movement mapping, returned to the normal (1:1) mapping, and re-learned the new movement mapping. The suggested metrics can provide comprehensive insights into users' learning process of movement mappings and accompany existing metrics that assess user performance at a summarization level (such as task completion time). Moreover, the metrics can inspire strategies aimed at assisting users in more effectively acquiring a new movement mapping and optimizing their performance when utilizing such mappings.

REFERENCES

[1] Parastoo Abtahi, Sidney Q. Hough, James A. Landay, and Sean Follmer. 2022. Beyond Being Real: A Sensorimotor Control Perspective on Interactions in Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 358, 17 pages. https://doi.org/10.1145/3491102.3517706

[2] M Haith Adrian and W Krakauer John. 2013. Theoretical models of motor control and motor learning. In *Routledge handbook of motor control and motor learning*. Routledge, 7–28.

[3] Ferran Argelaguet and Carlos Andujar. 2013. A survey of 3D object selection techniques for virtual environments. *Computers & Graphics* 37, 3 (2013), 121–136. https://doi.org/10.1016/j.cag.2012.12.003

[4] Mahdi Azmandian, Mark Hancock, Hrvoje Benko, Eyal Ofek, and Andrew D. Wilson. 2016. Haptic Retargeting: Dynamic Repurposing of Passive Haptics for Enhanced Virtual Reality Experiences. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1968–1979. https://doi.org/10.1145/2858036.2858226

[5] Belén Rubio Ballester, Laura Serra Oliva, Armin Duff, and Paul Verschure. 2015. Accelerating motor adaptation by virtual reality based modulation of error memories. In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 623–629. https://doi.org/10.1109/ICORR.2015.7281270

[6] Nikola Banovic, Tovi Grossman, and George Fitzmaurice. 2013. The Effect of Time-Based Cost of Error in Target-Directed Pointing Tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1373–1382. https://doi.org/10.1145/2470654.2466181

[7] Joanna Bergström, Tor-Salve Dalsgaard, Jason Alexander, and Kasper Hornbæk. 2021. How to Evaluate Object Selection and Manipulation in VR? Guidelines from 20 Years of Studies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 533, 20 pages. https://doi.org/10.1145/3411764.3445193

[8] Teige C Bourke, Angela M Coderre, Stephen D Bagg, Sean P Dukelow, Kathleen E Norman, and Stephen H Scott. 2015. Impaired corrective responses to postural perturbations of the arm in individuals with subacute stroke. *Journal of Neuro-Engineering and Rehabilitation* 12, 1 (2015), 1–15. https://doi.org/10.1186/1743-0003-12-7

[9] J.P.G van Brakel. 2014. Robust peak detection algorithm using z-scores. https://stackoverflow.com/questions/22583391/peak-signal-detection-in-realtime-timeseries-data/22640362#22640362

[10] Marco Caimmi, Stefano Carda, Chiara Giovanzana, Eliseo Stefano Maini, Angelo Maria Sabatini, Nicola Smania, and Franco Molteni. 2008. Using kinematic analysis to evaluate constraint-induced movement therapy in chronic stroke patients. *Neurorehabilitation and neural repair* 22, 1 (2008), 31–39. https://doi.org/10.1177/1545968307302923

[11] Noshaba Cheema, Laura A. Frey-Law, Kourosh Naderi, Jaakko Lehtinen, Philipp Slusallek, and Perttu Hämäläinen. 2020. Predicting Mid-Air Interaction Movements and Fatigue Using Deep Reinforcement Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376701

[12] Jiawen Chen, Shahram Izadi, and Andrew Fitzgibbon. 2012. KinÊtre: Animating the World with the Human Body. In *ACM SIGGRAPH 2012 Talks* (Los Angeles, California) *(SIGGRAPH '12)*. Association for Computing Machinery, New York, NY, USA, Article 39, 1 pages. https://doi.org/10.1145/2343045.2343098

[13] Yi Fei Cheng, Tiffany Luong, Andreas Rene Fender, Paul Streli, and Christian Holz. 2022. ComforTable User Interfaces: Surfaces Reduce Input Error, Time, and Exertion for Tabletop and Mid-air User Interfaces. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 150–159.

[14] Angela M Coderre, Amr Abou Zeid, Sean P Dukelow, Melanie J Demmer, Kimberly D Moore, Mary Jo Demers, Helen Bretzke, Troy M Herter, Janice I Glasgow, Kathleen E Norman, et al. 2010. Assessment of upper-limb sensorimotor function of subacute stroke patients using visually guided reaching. *Neurorehabilitation and neural repair* 24, 6 (2010), 528–541. https://doi.org/10.1177/1545968309356091

[15] Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences*. Academic press.

[16] Cheryl Coker. 2017. *Motor learning and control for practitioners*. Routledge.

[17] Kody R. Dillman, Terrance Tin Hoi Mok, Anthony Tang, Lora Oehlberg, and Alex Mitchell. 2018. A Visual Interaction Cue Framework from Video Game Environments for Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3173714

[18] Lionel Dominjon, Simon Richir, Anatole Lecuyer, and J-M Burkhardt. 2006. Haptic hybrid rotations: Overcoming hardware angular limitations of force-feedback devices. In *IEEE Virtual Reality Conference (VR 2006)*. IEEE, 167–174. https://doi.org/10.1109/VR.2006.68

[19] Julien Doyon, Allen W Song, Avi Karni, François Lalonde, Michelle M Adams, and Leslie G Ungerleider. 2002. Experience-dependent changes in cerebellar contributions to motor sequence learning. *Proceedings of the National Academy of Sciences* 99, 2 (2002), 1017–1022. https://doi.org/10.1073/pnas.022615199

[20] Hermann Ebbinghaus. 2013. Memory: A contribution to experimental psychology. *Annals of neurosciences* 20, 4 (2013), 155.

[21] João Marcelo Evangelista Belo, Anna Maria Feit, Tiare Feuchtner, and Kaj Grønbæk. 2021. XRgonomics: Facilitating the Creation of Ergonomic 3D Interfaces. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 290, 11 pages. https://doi.org/10.1145/3411764.3445349

[22] Tiare Feuchtner and Jörg Müller. 2017. Extending the Body for Interaction with Reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 5145–5157. https://doi.org/10.1145/3025453.3025689

[23] Florian Fischer, Arthur Fleig, Markus Klar, and Jörg Müller. 2022. Optimal Feedback Control for Modeling Human–Computer Interaction. *ACM Trans. Comput.-Hum. Interact.* 29, 6, Article 51 (nov 2022), 70 pages. https://doi.org/10.1145/3524122

[24] Paul M Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology* 47, 6 (1954), 381. https://doi.org/10.1037/h0055392

[25] Kenneth Flowers. 1975. Ballistic and corrective movements on an aiming task: intention tremor and parkinsonian movement disorders compared. *Neurology* 25, 5 (1975), 413–413. https://doi.org/10.1212/WNL.25.5.413

[26] Scott Frees, G. Drew Kessler, and Edwin Kay. 2007. PRISM Interaction for Enhancing Control in Immersive Virtual Environments. *ACM Trans. Comput.-Hum. Interact.* 14, 1 (may 2007), 2–es. https://doi.org/10.1145/1229855.1229857

[27] Zihan Gao, Huiqiang Wang, Hongwu Lv, Moshu Wang, and Yifan Qi. 2020. Evaluating the effects of non-isomorphic rotation on 3D manipulation tasks in mixed reality simulation. *IEEE Transactions on Visualization and Computer Graphics* 28, 2 (2020), 1261–1273. https://doi.org/10.1109/TVCG.2020.3010247

[28] Eric J. Gonzalez, Parastoo Abtahi, and Sean Follmer. 2019. Evaluating the Minimum Jerk Motion Model for Redirected Reach in Virtual Reality. In *Adjunct Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 4–6. https://doi.org/10.1145/3332167.3357096

[29] Eric J Gonzalez, Elyse D. Z. Chase, Pramod Kotipalli, and Sean Follmer. 2022. A Model Predictive Control Approach for Reach Redirection in Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>New Orleans</city>, <state>LA</state>, <country>USA</country>, </conf-loc>) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 39, 15 pages. https://doi.org/10.1145/3491102.3501907

[30] Eric J Gonzalez and Sean Follmer. 2023. Sensorimotor Simulation of Redirected Reaching using Stochastic Optimal Feedback Control. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Hamburg</city>, <country>Germany</country>, </conf-loc>) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 776, 17 pages. https://doi.org/10.1145/3544548.3580767

[31] Tovi Grossman and Ravin Balakrishnan. 2005. The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Portland, Oregon, USA) *(CHI '05)*. Association for Computing Machinery, New York, NY, USA, 281–290. https://doi.org/10.1145/1054972.1055012

[32] Adrian M Haith and John W Krakauer. 2013. Theoretical models of motor control and motor learning. *Routledge handbook of motor control and motor learning* (2013), 1–28.

[33] Andrew Heathcote, Scott Brown, and Douglas JK Mewhort. 2000. The power law repealed: The case for an exponential law of practice. *Psychonomic bulletin & review* 7, 2 (2000), 185–207. https://doi.org/10.3758/BF03212979

[34] Neville Hogan and Dagmar Sternad. 2009. Sensitivity of smoothness measures to movement duration, amplitude, and arrests. *Journal of motor behavior* 41, 6 (2009), 529–534. https://doi.org/10.3200/35-09-004-RC

[35] Kasper Hornbæk. 2006. Current practice in measuring usability: Challenges to usability studies and research. *International journal of human-computer studies* 64, 2 (2006), 79–102. https://doi.org/10.1016/j.ijhcs.2005.06.002

[36] Ludovic Hoyet, Ferran Argelaguet, Corentin Nicole, and Anatole Lécuyer. 2016. "Wow! i have six Fingers!": Would You accept structural changes of Your hand in Vr? *Frontiers in Robotics and AI* 3 (2016), 27. https://doi.org/10.3389/frobt.2016.00027

[37] Aleksi Ikkala, Florian Fischer, Markus Klar, Miroslav Bachinski, Arthur Fleig, Andrew Howes, Perttu Hämäläinen, Jörg Müller, Roderick Murray-Smith, and Antti Oulasvirta. 2022. Breathing Life Into Biomechanical User Models. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 90, 14 pages. https://doi.org/10.1145/3526113.3545689

[38] Jun Izawa, Tushar Rane, Opher Donchin, and Reza Shadmehr. 2008. Motor adaptation as a process of reoptimization. *Journal of Neuroscience* 28, 11 (2008), 2883–2891. https://doi.org/10.1523/JNEUROSCI.5359-07.2008

[39] Sujin Jang, Wolfgang Stuerzlinger, Satyajit Ambike, and Karthik Ramani. 2017. Modeling Cumulative Arm Fatigue in Mid-Air Interaction Based on Perceived

Exertion and Kinetics of Arm Motion. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3328–3339. https://doi.org/10.1145/3025453.3025523

[40] Jussi P. P. Jokinen, Sayan Sarcar, Antti Oulasvirta, Chaklam Silpasuwanchai, Zhenxin Wang, and Xiangshi Ren. 2017. Modelling Learning of New Keyboard Layouts. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4203–4215. https://doi.org/10.1145/3025453.3025580

[41] Avi Kami, Gundela Meyer, Peter Jezzard, Michelle M Adams, Robert Turner, and Leslie G Ungerleider. 1995. Functional MRI evidence for adult motor cortex plasticity during motor skill learning. *Nature* 377, 6545 (1995), 155–158. https://doi.org/10.1038/377155a0

[42] Alexander T Korenberg and Zoubin Ghahramani. 2002. A Bayesian view of motor adaptation. *Current Psychology of Cognition* 21, 4/5 (2002), 537–564.

[43] John W Krakauer, Claude Ghez, and M Felice Ghilardi. 2005. Adaptation to visuomotor transformations: consolidation, interference, and forgetting. *Journal of Neuroscience* 25, 2 (2005), 473–478. https://doi.org/10.1523/JNEUROSCI.4218-04.2005

[44] John W Krakauer, Alkis M Hadjiosif, Jing Xu, Aaron L Wong, and Adrian M Haith. 2019. Motor learning. *Compr Physiol* 9, 2 (2019), 613–663. https://doi.org/10.1002/cphy.c170043

[45] John W Krakauer, Zachary M Pine, Maria-Felice Ghilardi, and Claude Ghez. 2000. Learning of visuomotor transformations for vectorial planning of reaching trajectories. *Journal of neuroscience* 20, 23 (2000), 8916–8924. https://doi.org/10.1523/JNEUROSCI.20-23-08916.2000

[46] Byungjoo Lee, Mathieu Nancel, Sunjun Kim, and Antti Oulasvirta. 2020. Auto-Gain: Gain Function Adaptation with Submovement Efficiency Optimization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376244

[47] Christian Leukel and Jesper Lundbye-Jensen. 2013. The role of augmented feedback in human motor learning. In *Routledge handbook of motor control and motor learning*. Routledge, 143–162.

[48] Jialei Li, Isaac Cho, and Zachary Wartell. 2018. Evaluation of Cursor Offset on 3D Selection in VR. In *Proceedings of the 2018 ACM Symposium on Spatial User Interaction* (Berlin, Germany) *(SUI '18)*. Association for Computing Machinery, New York, NY, USA, 120–129. https://doi.org/10.1145/3267782.3267797

[49] Lei Liu and Robert van Liere. 2009. Designing 3D selection techniques using ballistic and corrective movements. In *Virtual Environments 2009: Joint Virtual Reality Conference of EGVE-ICAT-EuroVR (Lyon, France, December 7-9, 2009)*. Eurographics Association, 1–8. https://doi.org/10.2312/EGVE/JVRC09/001-008

[50] Xiaolong Liu, Lili Wang, Shuai Luan, Xuehuai Shi, and Xinda Liu. 2022. Distant Object Manipulation with Adaptive Gains in Virtual Reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 739–747.

[51] Gordon D Logan. 1992. Shapes of reaction-time distributions and shapes of learning curves: a test of the instance theory of automaticity. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 18, 5 (1992), 883. https://doi.org/10.1037/0278-7393.18.5.883

[52] Richard Magill and David I Anderson. 2010. *Motor learning and control*. McGraw-Hill Publishing New York.

[53] Lorenzo Masia, Flaminia Frascarelli, Pietro Morasso, Giuseppe Di Rosa, Maurizio Petrarca, Enrico Castelli, and Paolo Cappa. 2011. Reduced short term adaptation to robot generated dynamic environment in children affected by Cerebral Palsy. *Journal of neuroengineering and rehabilitation* 8 (2011), 1–13. https://doi.org/10.1186/1743-0003-8-28

[54] Rene M Maura, Sebastian Rueda Parra, Richard E Stevens, Douglas L Weeks, Eric T Wolbrecht, and Joel C Perry. 2023. Literature review of stroke assessment for upper-extremity physical function via EEG, EMG, kinematic, and kinetic measurements and their reliability. *Journal of NeuroEngineering and Rehabilitation* 20, 1 (2023), 1–32. https://doi.org/10.1186/s12984-023-01142-7

[55] Jess McIntosh, Hubert Dariusz Zajac, Andreea Nicoleta Stefan, Joanna Bergström, and Kasper Hornbæk. 2020. Iteratively Adapting Avatars Using Task-Integrated Optimisation. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 709–721. https://doi.org/10.1145/3379337.3415832

[56] Daniel Mendes, Fabio Marco Caputo, Andrea Giachetti, Alfredo Ferreira, and Joaquim Jorge. 2019. A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. In *Computer graphics forum*, Vol. 38. Wiley Online Library, 21–45. https://doi.org/10.1111/cgf.13390

[57] Andreea Muresan, Jess Mcintosh, and Kasper Hornbæk. 2023. Using Feedforward to Reveal Interaction Possibilities in Virtual Reality. *ACM Trans. Comput.-Hum. Interact.* 30, 6, Article 82 (sep 2023), 47 pages. https://doi.org/10.1145/3603623

[58] Roderick Murray-Smith, Antti Oulasvirta, Andrew Howes, Jörg Müller, Aleksi Ikkala, Miroslav Bachinski, Arthur Fleig, Florian Fischer, and Markus Klar. 2022. What Simulation Can Do for HCI Research. *Interactions* 29, 6 (nov 2022), 48–53. https://doi.org/10.1145/3564038

[59] Allen Newell and Paul S Rosenbloom. 1981. Mechanisms of skill acquisition and the law of practice. In *Cognitive skills and their acquisition*. Psychology Press, 1–55.

[60] James L Patton, Mary Ellen Stoykov, Mark Kovic, and Ferdinando A Mussa-Ivaldi. 2006. Evaluation of robotic training forces that either enhance or reduce error in chronic hemiparetic stroke survivors. *Experimental brain research* 168 (2006), 368–383. https://doi.org/10.1007/s00221-005-0097-8

[61] Toni S Pearson, John W Krakauer, and Pietro Mazzoni. 2010. Learning not to generalize: modular adaptation of visuomotor gain. *Journal of neurophysiology* 103, 6 (2010), 2938–2952. https://doi.org/10.1152/jn.01089.2009

[62] Zachary M Pine, John W Krakauer, J Gordon, and C Ghez. 1996. Learning of scaling factors and reference axes for reaching movements. *Neuroreport* 7 (1996), 2357–2361. https://doi.org/10.1097/00001756-199610020-00016

[63] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. The go-go interaction technique: non-linear mapping for direct manipulation in VR. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*. 79–80. https://doi.org/10.1145/237091.237102

[64] Mahfuz Rahman, Sean Gustafson, Pourang Irani, and Sriram Subramanian. 2009. Tilt Techniques: Investigating the Dexterity of Wrist-Based Input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1943–1952. https://doi.org/10.1145/1518701.1518997

[65] Brandon Rohrer, Susan Fasoli, Hermano Igo Krebs, Bruce Volpe, Walter R Frontera, Joel Stein, and Neville Hogan. 2004. Submovements grow larger, fewer, and more blended during stroke recovery. *Motor control* 8, 4 (2004), 472–483. https://doi.org/10.1123/mcj.8.4.472

[66] Alan W Salmoni, Richard A Schmidt, and Charles B Walter. 1984. Knowledge of results and motor learning: a review and critical reappraisal. *Psychological bulletin* 95, 3 (1984), 355. https://doi.org/10.1037/0033-2909.95.3.355

[67] Robert A Scheidt and Tina Stoeckmann. 2007. Reach adaptation and final position control amid environmental uncertainty after stroke. *Journal of neurophysiology* 97, 4 (2007), 2824–2836. https://doi.org/10.1152/jn.00870.2006

[68] Jonas Schjerlund, Kasper Hornbæk, and Joanna Bergström. 2021. Ninja Hands: Using Many Hands to Improve Target Selection in VR. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 130, 14 pages. https://doi.org/10.1145/3411764.3445759

[69] Dominik Schön, Thomas Kosch, Florian Müller, Martin Schmitz, Sebastian Günther, Lukas Bommhardt, and Max Mühlhäuser. 2023. Tailor Twist: Assessing Rotational Mid-Air Interactions for Augmented Reality. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 400, 14 pages. https://doi.org/10.1145/3544548.3581461

[70] Anne Schwarz, Christoph M Kanzler, Olivier Lambercy, Andreas R Luft, and Janne M Veerbeek. 2019. Systematic review on kinematic assessments of upper limb movements after stroke. *Stroke* 50, 3 (2019), 718–727. https://doi.org/10.1161/STROKEAHA.118.023531

[71] Yeongho Seol, Carol O'Sullivan, and Jehee Lee. 2013. Creature Features: Online Motion Puppetry for Non-Human Characters. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Anaheim, California) *(SCA '13)*. Association for Computing Machinery, New York, NY, USA, 213–221. https://doi.org/10.1145/2485895.2485903

[72] Nataliya Shishov, Itshak Melzer, and Simona Bar-Haim. 2017. Parameters and measures in assessment of motor learning in neurorehabilitation; a systematic review of the literature. *Frontiers in human neuroscience* 11 (2017), 82.

[73] Maurice A Smith, Ali Ghazizadeh, and Reza Shadmehr. 2006. Interacting adaptive processes with different timescales underlie short-term motor learning. *PLoS biology* 4, 6 (2006), e179. https://doi.org/10.1371/journal.pbio.0040179

[74] R William Soukoreff and I Scott MacKenzie. 2004. 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 (2004), 751–789. https://doi.org/10.1016/j.ijhcs.2004.09.001

[75] William Steptoe, Anthony Steed, and Mel Slater. 2013. Human tails: ownership and control of extended humanoid avatars. *IEEE transactions on visualization and computer graphics* 19, 4 (2013), 583–590. https://doi.org/10.1109/TVCG.2013.32

[76] Paul Strohmeier, Aske Mottelson, Henning Pohl, Jess McIntosh, Jarrod Knibbe, Joanna Bergström, Yvonne Jansen, and Kasper Hornbæk. 2022. Body-based user interfaces. In *The Routledge Handbook of Bodily Awareness*. Routledge, 478–502.

[77] Craig D Takahashi and David J Reinkensmeyer. 2003. Hemiparetic stroke impairs anticipatory control of arm movement. *Experimental brain research* 149 (2003), 131–140. https://doi.org/10.1007/s00221-002-1340-1

[78] Jordan A Taylor, John W Krakauer, and Richard B Ivry. 2014. Explicit and implicit contributions to learning in a sensorimotor adaptation task. *Journal of Neuroscience* 34, 8 (2014), 3023–3032. https://doi.org/10.1523/JNEUROSCI.3619-13.2014

[79] Hans-Leo Teulings, José L Contreras-Vidal, George E Stelmach, and Charles H Adler. 1997. Parkinsonism reduces coordination of fingers, wrist, and arm in fine motor control. *Experimental neurology* 146, 1 (1997), 159–170. https://doi.org/10.

1006/exnr.1997.6507
[80] Wen-Jie Tseng, Samuel Huron, Eric Lecolinet, and Jan Gugenheimer. 2023. FingerMapper: Mapping Finger Motions onto Virtual Arms to Enable Safe Virtual Reality Interaction in Confined Spaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 874, 14 pages. https://doi.org/10.1145/3544548.3580736
[81] Andrea Turolla, Omar A Daud Albasini, Roberto Oboe, Michela Agostini, Paolo Tonin, Stefano Paolucci, Giorgio Sandrini, Annalena Venneri, Lamberto Piron, et al. 2013. Haptic-based neurorehabilitation in poststroke patients: a feasibility prospective multicentre trial for robotics hand rehabilitation. *Computational and mathematical methods in medicine* 2013 (2013). https://doi.org/10.1155/2013/895492
[82] Kohei Umezawa, Yuta Suzuki, Gowrishankar Ganesh, and Yoichi Miyawaki. 2022. Bodily ownership of an independent supernumerary limb: an exploratory study. *Scientific reports* 12, 1 (2022), 2339. https://doi.org/10.1038/s41598-022-06040-x
[83] Rafael Veras, Gaganpreet Singh, Farzin Farhadi-Niaki, Ritesh Udhani, Parth Pradeep Patekar, Wei Zhou, Pourang Irani, and Wei Li. 2021. Elbow-Anchored Interaction: Designing Restful Mid-Air Input. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 737, 15 pages. https://doi.org/10.1145/3411764.3445546
[84] Adrien Verhulst, Yasuko Namikawa, and Shunlchl Kasahara. 2022. Parallel Adaptation: Switching between Two Virtual Bodies with Different Perspectives Enables Dual Motor Adaptation. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 169–177. https://doi.org/10.1109/ISMAR55827.2022.00031
[85] Ana Villanueva, Sujin Jang, Wolfgang Stuerzlinger, Satyajit Ambike, and Karthik Ramani. 2023. Advanced modeling method for quantifying cumulative subjective fatigue in mid-air interaction. *International Journal of Human-Computer Studies* 169 (2023), 102931. https://doi.org/10.1016/j.ijhcs.2022.102931
[86] Johann Wentzel, Greg d'Eon, and Daniel Vogel. 2020. Improving Virtual Reality Ergonomics Through Reach-Bounded Non-Linear Input Amplification. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376687
[87] Daniel M Wolpert, Jörn Diedrichsen, and J Randall Flanagan. 2011. Principles of sensorimotor learning. *Nature reviews neuroscience* 12, 12 (2011), 739–751. https://doi.org/10.1038/nrn3112
[88] Andrea Stevenson Won, Jeremy Bailenson, Jimmy Lee, and Jaron Lanier. 2015. Homuncular flexibility in virtual reality. *Journal of Computer-Mediated Communication* 20, 3 (2015), 241–259. https://doi.org/10.1111/jcc4.12107
[89] Ching-yi Wu, Chia-ling Chen, Simon F Tang, Keh-chung Lin, and Ya-ying Huang. 2007. Kinematic and clinical analyses of upper-extremity movements after constraint-induced movement therapy in patients with stroke: a randomized controlled trial. *Archives of physical medicine and rehabilitation* 88, 8 (2007), 964–970. https://doi.org/10.1016/j.apmr.2007.05.012
[90] Difeng Yu. 2023. *Enhancing Mid-Air Selection and Manipulation for Complex Virtual Reality Interaction*. The University of Melbourne.
[91] Difeng Yu, Ruta Desai, Ting Zhang, Hrvoje Benko, Tanya R. Jonker, and Aakar Gupta. 2022. Optimizing the Timing of Intelligent Suggestion in Virtual Reality. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 6, 20 pages. https://doi.org/10.1145/3526113.3545632
[92] Difeng Yu, Hai-Ning Liang, Xueshi Lu, Kaixuan Fan, and Barrett Ens. 2019. Modeling endpoint distribution of pointing selection tasks in virtual reality environments. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–13. https://doi.org/10.1145/3355089.3356544
[93] Difeng Yu, Xueshi Lu, Rongkai Shi, Hai-Ning Liang, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2021. Gaze-Supported 3D Object Manipulation in Virtual Reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 734, 13 pages. https://doi.org/10.1145/3411764.3445343
[94] Difeng Yu, Qiushi Zhou, Joshua Newn, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2020. Fully-occluded target selection in virtual reality. *IEEE transactions on visualization and computer graphics* 26, 12 (2020), 3402–3413. https://doi.org/10.1109/TVCG.2020.3023606

A METRICS IMPLEMENTATION

RTP, *RS*, *NPE*, and *NJE* require the following information to be recorded in the system: trial/movement starting time t_{start}, trial/movement ending time t_{end}, and movement trajectory data. For 3D interaction, the movement trajectory data can be a list of (p_x^t, p_y^t, p_z^t), representing the 3D position/rotation information from t_{start} to t_{end} in the frequency of f. We recommend $f \geq 50Hz$ (i.e., recording at least every 0.02 seconds).

With the recorded data, we can approximate the velocity profile (v_x^t, v_y^t, v_z^t), where $v^t = (p^t - p^{t-1}) \cdot f$, assuming $v^{t_{start}} = 0m/s$. We can apply a similar strategy to compute the acceleration profile (a_x^t, a_y^t, a_z^t) and jerk profile (j_x^t, j_y^t, j_z^t). Additionally, we can calculate the movement distance L_a by accumulating the Euclidean distances between the data points in the movement trajectory.

While *task completion time* can be simply calculated as $t_{end} - t_{start}$, both *RTP* and *RS* require an algorithm to find when the ballistic phase ends (t_b) based on the trajectory data. We applied a z-score-based peak detection algorithm [9] (pseudocode in Appendix A.1), which based on our pilot tests detected the initiation of a peak. At a high level, it identifies whether an incoming data point deviates away from a moving mean by, for example, three standard deviations. The moving mean is an average of the last, say 5, observations. Because the algorithm detects the initiation of a peak while we need the ending timestamp of the first ballistic movement peak, we feed the movement data in reverse order (i.e., from the end to the beginning) to calculate t_b. With t_b, we can calculate *RTP* with Equation 2. We can also retrieve position information $(p_x^{t_b}, p_y^{t_b}, p_z^{t_b})$ and $(p_x^{t_{end}}, p_y^{t_{end}}, p_z^{t_{end}})$ to calculate *RS*.

We can also calculate *NPE* once we compute L_a from the movement trajectory data and fill L_i with the relevant task settings. Furthermore, *NJE* can be determined with the derived speed and jerk profiles as well as L_a and D (=$t_{end} - t_{start}$). Our programming implementation can be found in the open-source toolkit.

A.1 Peak Detection Algorithm

We applied a peak detection algorithm based on Z-scores [9] to identify the ballistic movements in speed profiles. We attach the pseudocode of the algorithm for reference, while more detailed explanations can be found in the original post [9]. The parameters `lag=3`, `threshold=3`, and `influence=0.1` are empirically determined through trial and error.

```
"Set parameters"
lag = 3  # dynamic list that stores mean and std.
threshold = 3  # signals if the new data point > mean +/- 3std.
influence = 0.1  # influence of the new data point on mean and std.

"Initialize variables"
signals = [0, ..., 0]
filteredY = [y(1), ..., y(lag)]
avgFilter[lag] = mean([y(1), ..., y(lag)])
stdFilter[lag] = std([y(1), ..., y(lag)])

"Peak detection"
for i in (lag + 1, t)
  if abs(y[i] - avgFilter[i-1]) > threshold * stdFilter[i-1]
    if y[i] > avgFilter[i-1]
      signals[i] = 1  # positive signal
    else
      signals[i] = -1  # negative signal
    filteredY[i] = influence * y[i] + (1 - influence) * filteredY[i-1]
  else
    signals[i] = 0  # no signal
    filteredY[i] = y[i]
  avgFilter[i] = mean([filteredY[i-lag+1], ..., filteredY[i]])
  stdFilter[i] = std([filteredY[i-lag+1], ..., filteredY[i]])
```

B TASKS

B.1 Task Descriptions

B.1.1 LT Task: Linear Translational Gain. The task requires the participants to move a 3D cube from a starting position (a blue square) to a red target circle on the table (see Figure 3 left). The cube is influenced by gravity, and the target circle turns green (from red) as long as the cube is inside the horizontal range of the circle. This task mimics many VR applications (e.g., *Hand Physics Lab* and *Job Simulator*) where a user needs to move an object on the table.

The target circle can appear in different depths, representing the task assumption of *difficulty levels*, and in different directions with respect to the starting position, representing the *variance levels*. In our task setting, the targets only appear at the center or right sides of the table, which are supposed to be more comfortable movement positions [21]. This is one of the steps to prevent fatigue with repetitive trials, which could interplay with the learning effect.

The participants complete the tasks in two mappings: with a virtual hand whose position is mapped 1:1 from the position of their physical hand, and with a virtual hand whose position is mapped with a linear translational gain. The linear gain scales the virtual hand movement in the depth dimension (the forward z-axis) according to a constant factor k once it passes the starting position [48]. This mapping allows users to reach further distances outside their real reach while maintaining the integrity of the lateral movement.

B.1.2 NLR Task: Non-linear Rotational Gain. The task requires the participants to rotate a 3D orange bunny to a target orientation as the bunny in white (see Figure 3 middle). The outline of the orange bunny turns green (from red) once its orientation is within a tolerant threshold. The three displayed axes for the bunny are meant to help ease the mental rotation load for users. The task mimics VR application scenarios such as 3D modeling, where users need to re-orientate an object into a desired direction [56, 93]. The position of the bunny is fixed to eliminate the need for object translation which is covered in the LT task.

The target orientations create different *difficulty levels* and *variance levels*, depending on the rotation degrees and directions. Based on previous works on wrist-based input [64, 69], our task needs only wrist flexion and pronation if grabbing the bunny from the side. The required rotations are within the comfortable range [64].

The participants complete the tasks again in two mappings: with a 1:1 mapping and with a non-linear rotational gain mapping. This mapping is used to overcome the physical limitations of wrist rotation and may assist in fine-tuning the accuracy [18, 27, 50]. We replicate the *dynamic non-isomorphic rotation* technique by Gao et al. [27], where the virtual hand rotation can be amplified or attenuated based on the angular movement speed of the physical hand during manipulation. The scaling factor k can be described by the following equation:

$$k = \begin{cases} 0 & v < v_{min} \\ k_{max} \cdot \frac{v - v_{min}}{v_{max} - v_{min}} & v_{min} < v < v_{max} \\ k_{max} & v > v_{max} \end{cases} \quad (7)$$

We used the same parameters as Gao et al. [27]: $v_{min} = 5°/s, v_{max} = 90°/s$, and $k_{max} = 3$. The virtual hand is aligned with the physical hand once finishing a continuous manipulation (i.e., releasing the pinch).

In our pilot tests, we found this task can be challenging for some users, likely due to the need for mental rotation. Therefore, we use this task primarily to assess the GENERALIZABILITY of the metrics by representing a scenario where the task is difficult.

B.1.3 FA Task: Finger-to-Arm Mapping. The task requires users to select a white sphere (see Figure 3 right). Before selection, the user must place the pink cursor on the index fingertip in a cube (i.e., the home). The cube disappears randomly within a predefined time range to indicate the initiation of a selection trial. To select the sphere, the cursor must stay within the target sphere for a specific duration. The sphere's boundary turns pink once the cursor enters the sphere boundary, and the countdown clock next to the target starts counting until task completion. This task represents a simple, common target selection scenario in VR [3, 7].

The target distances represent different *difficulty levels*, and the target directions (along the sphere formed by the home position and the target position) describe different *variance levels*. In our task setting, the target only appears above the left side of the table, allowing more precise right index finger tracking—the finger is used in the movement remapping technique described below.

The participants complete the tasks in two mappings: with a 1:1 finger-to-finger mapping, and with a multi-joint finger-to-arm technique called *FingerMapper* [80]. *FingerMapper* controls the virtual arm movement with the movement of an index finger, which can help reduce arm fatigue and enables full-arm motions with subtle finger movements in a confined physical space. We applied the *Attach* mapping function—a user controls the proximal phalanx for the arm reach direction and bends their index finger to retract the virtual wrist to their shoulder. Essentially, the movements of different finger joints are used to determine the actual movement of the virtual arm.

B.2 Task Parameters

During the development and pilot study phase, we established a set of parameters within the task environment to balance task difficulties. We provide these parameters here for replication purposes.

```
"LT Task"
difficultyLevelFar = [0.6, 0.7, 0.8]  # distance for gain amplification
difficultyLevelClose = [0.24, 0.28, 0.32]  # distance for 1:1 mapping
varianceLevel = [0, 10, 20, 30, 40]  # angles to z-axis
targetSize = 0.06  # target diameter
scaleFactor = 8  # scale factor of gain amplification

"NLR Task"
difficultyLevel = [-30, 50, 70]  # pronation (distance in degrees)
varianceLevel = [65, 60, 55, 50, 45]  # flexion (direction in degrees)
targetSize = 30  # tolerance angle in degrees

"FA Task"
difficultyLevel = [0.15, 0.2, 0.25]  # distance to the start position
varianceLevel = [-40, -30, -20, -10, 0]  # horizontal angles to z-axis
heightVariance = [0, 5, 10]  # randomized vertical angles to z-axis
targetSize = 0.05  # sphereical target diameter
startSize = 0.03  # start cube size
startPosition = Vector3(0, 1.1, 0.3)  # 3D position of the cube
dwellDuration = 1  # required dwell time for selection
randomStart = Random.Range(0.5, 0.8)  # starting time after dwelling
```

C DATA PROCESSING

We recorded the following data during the study to enable the application of the metrics and the analysis of the results: trial completion time (which is equivalent to $t_{end} - t_{start}$), movement trajectory data in the frequency of $50Hz$ (with the `FixedUpdate()` function in Unity), and other task-relevant information such as trial number and condition number. The movement trajectory data were 3D cube positions (in meters) in LT Task, 3D bunny rotations (in radians) in NLR Task, and 3D cursor positions (in meters) in FA Task. The speed profile comes from a finite difference approximation from position data, and we used a kernel regression smoother `ksmooth()` in R language with $bandwidth = 4$ to stabilize the calculated speed profile. The number of points in the kernel smoother to evaluate the fit (the $n.points$ parameter) equals the number of logged timestamps in a trial.

We then treated a trial as an outlier if its task completion time was three standard deviations away from the average value of all participants in a learning trial. Note there were 120 learning trials across the four learning blocks for each participant in a task. We removed those outliers because they were far from typical performance, possibly due to participants' confusion in that trial [7, 92]. Through this process, we discarded 36 trials (1.9%) in LT Task, 47 trials (2.4%) in NLR Task, and 20 trials (1.0%) in FA Task.

To analyze the VALIDITY of the metrics, we fit power functions (Equation 1) to the LT Task data. We used a general-purpose optimization function in R language that employs the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (i.e., `optim()` function with `L-BFGS-B`) to minimize the sum of squared errors. The lower and upper boundaries of a, k, and b were set as $(-\infty, 0, 0)$ and (∞, ∞, ∞). Furthermore, we calculated the coefficient of determination (R^2) and the root mean square error ($RMSE$) to estimate the goodness of fit. While we referred to the rule of thumb thresholds for R^2 in behavioral science ('small', 'medium', and 'large' values are 0.01, 0.09, and 0.25) [15], we were also aware R^2 are context dependable and took precautions when interpreting them. We also used $RMSE$ to gauge how far predictions fall from the ground truth. We extended this method to NLR Task, FA Task, and individual data for evaluating GENERALIZABILITY.

The RESILIENCY of a metric was determined through linear mixed models (`lmer()` in the `lmerTest` package) that test whether there was a significant effect of *difficulty levels* on the metric data. We developed the following formula: $MetricData \sim DifficultyLevel + VarianceLevel + Phase * RepetitionsWithinPhase + (1 \mid Participant)$. We treated the *difficulty levels*, *variance levels*, *phases*, *repetitions within a phase*, and a known interaction between *phases* and *repetitions within a phase* as continuous fixed effects. We considered *participants* as a random effect. All data were normalized with the `bestNormalize` function, which estimated the best normalizing function from a suite of transformations (e.g., Box-Cox, log, and Yeo-Johnson). We visually inspected the Q-Q plots to ensure the residuals reasonably aligned with the normality assumption.

D ADDITIONAL STUDY RESULTS

Figure 6 concatenates training and de-adaptation of *NPE* in LT Task. Figure 7 and 8 show the fitting results of the power function regarding NLR Task (non-linear rotation) and FA Task (finger-to-arm mapping). Figure 9 illustrates how different mappings and *difficulty levels* influenced metrics.

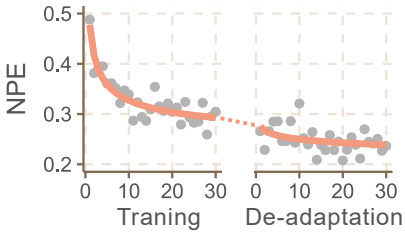

Figure 6: Concatenating training and de-adaptation data of *NPE* in LT Task. Data points are averaged across participants.

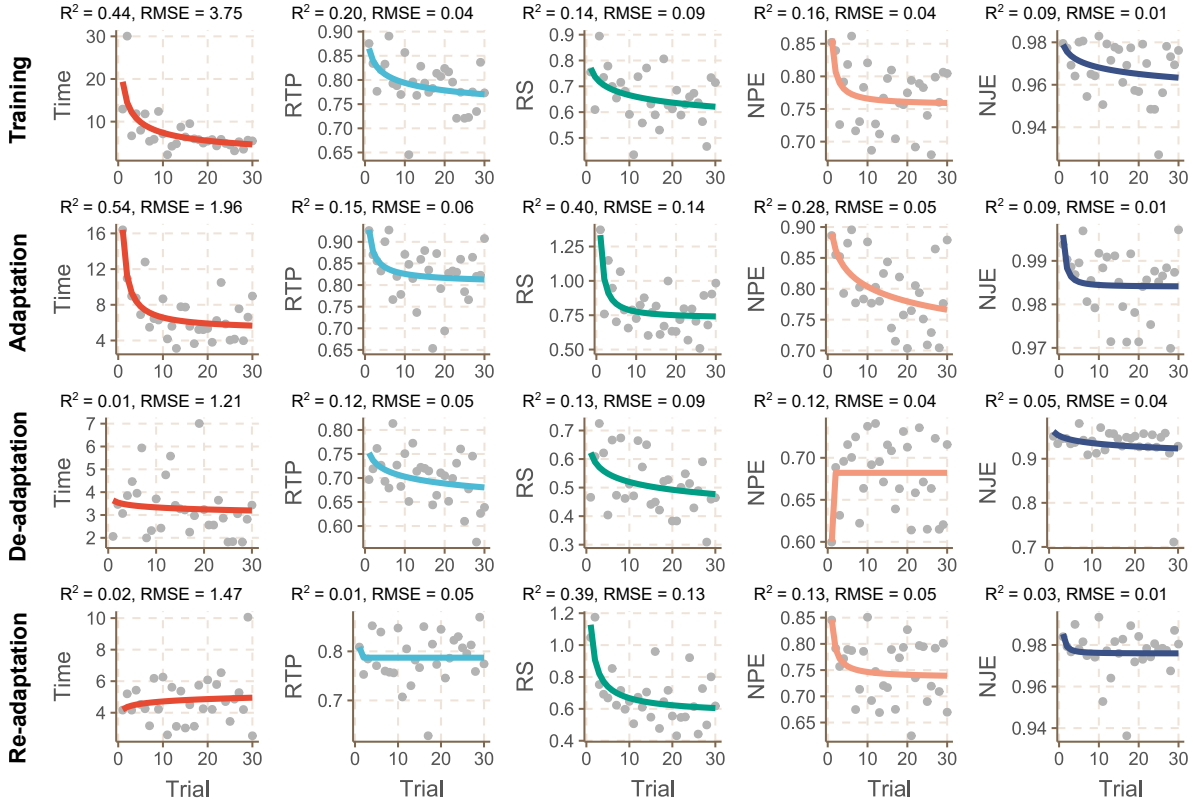

Figure 7: Fitting results of the power function in NLR Task (non-linear rotation). Data points are averaged across participants.

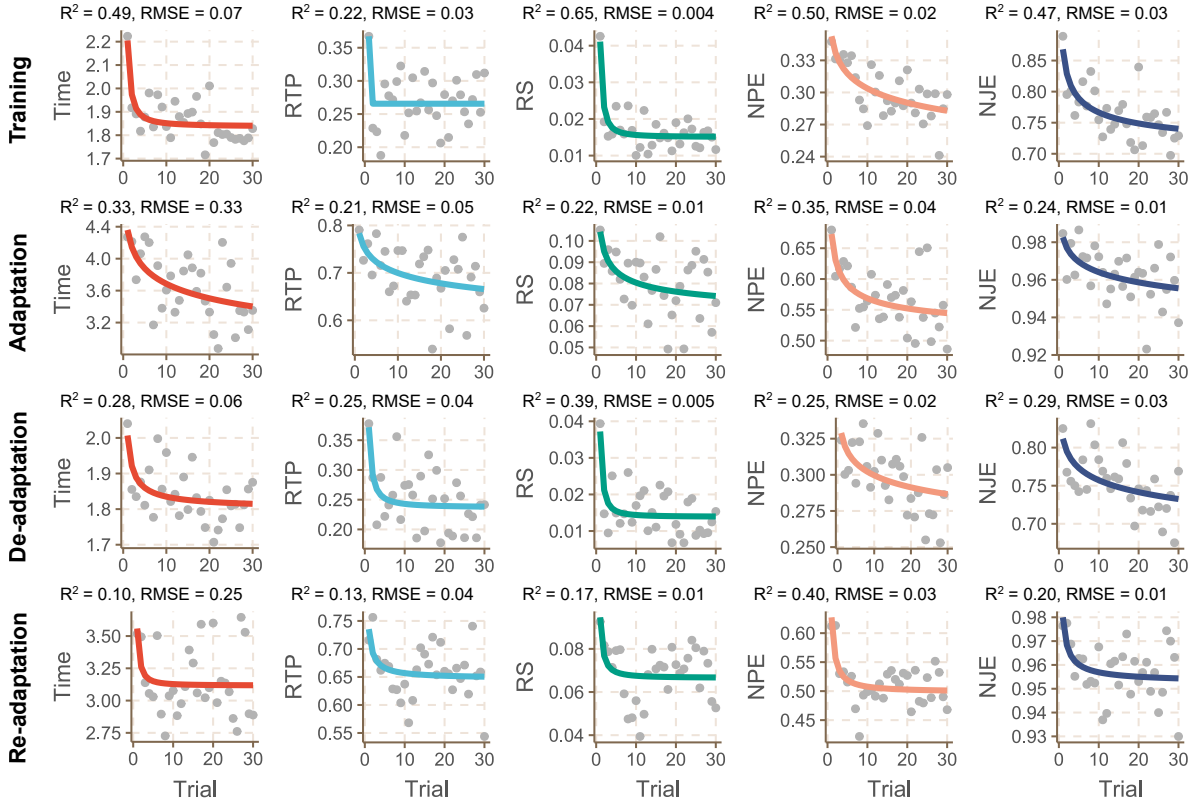

Figure 8: Fitting results of the power function in FA Task (finger-to-arm mapping). Data points are averaged across participants.

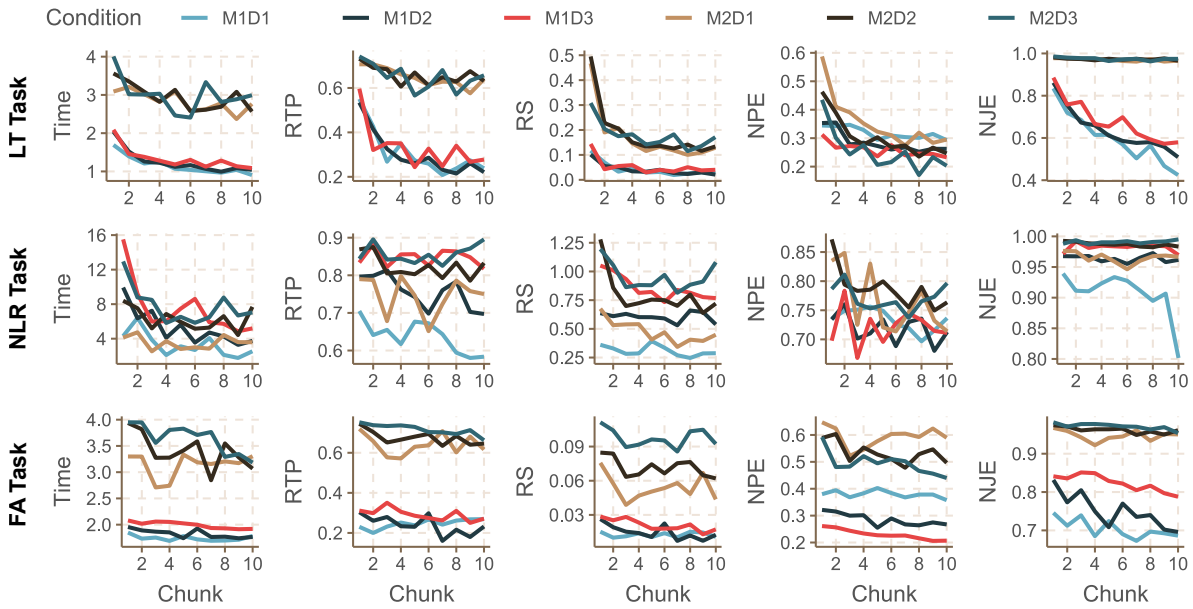

Figure 9: Line plots on how mappings M and difficulty levels D, browsed every three trials (a chunk), influenced metrics.