Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

So Predictable! Continuous 3D Hand Trajectory Prediction in Virtual Reality

Nisal Menuka Gamage, Deepana Ishtaweera, Martin Weigel, Anusha Withana

2021

Preprint:

Copyright ACM 2021. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), https://doi.org/10.1145/3472749.3474753.

So Predictable! Continuous 3D Hand Trajectory Prediction in Virtual Reality

Nisal Menuka Gamage
School of Computer Science, University of Sydney
Sydney, Australia
nisal.gamage@sydney.edu.au

Deepana Ishtaweera*
School of Computer Science, University of Sydney
Sydney, Australia
deepana.ishtaweera@sydney.edu.au

Martin Weigel
Honda Research Institute Europe
Offenbach am Main, Germany
martin.weigel@honda-ri.de

Anusha Withana
School of Computer Science, University of Sydney
Sydney, Australia
anusha.withana@sydney.edu.au

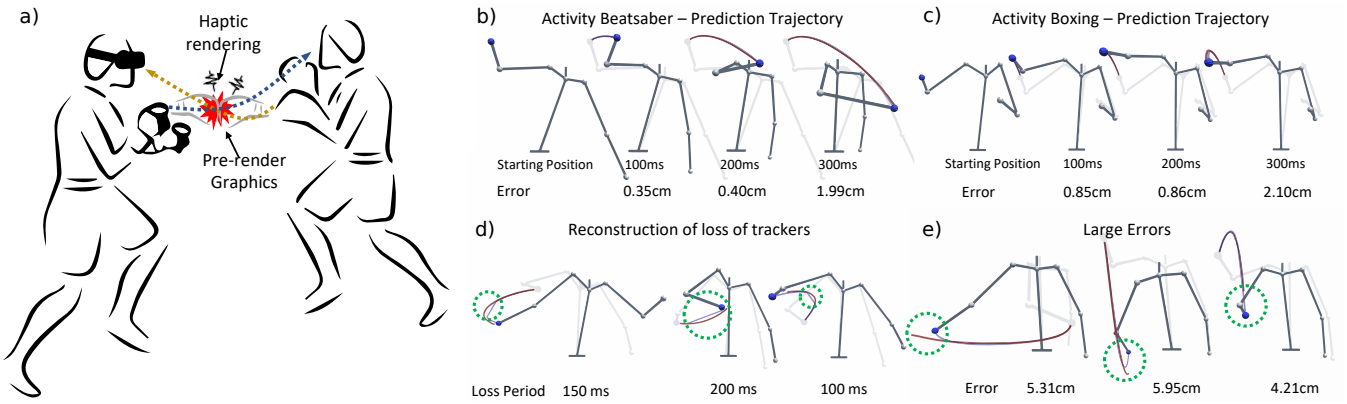


Figure 1: We present a *hybrid classical-regressive kinematics model* to predict hand motion trajectories in virtual reality. a) Future trajectory can be used to forecast events such as hand collision with other users or non-player characters, enabling pre-rendering of graphics or haptic feedback; b,c) Comparison of the predicted (red) and the real (blue) trajectories for different prediction intervals (PI) for *BeatSaber* and *FitXR-Box* games, showing average error; d) Prediction model can reconstruct the trajectory when tracking fails; e) Example cases of high errors with sudden changes to movement directions (PI = 300 ms).

Abstract

We contribute a novel user- and activity-independent kinematics-based regressive model for continuously predicting ballistic hand movements in virtual reality (VR). Compared to prior work on end-point prediction, continuous hand trajectory prediction in VR enables an early estimation of future events such as collisions between the user's hand and virtual objects such as UI widgets. We developed and validated our prediction model through a user study with 20 participants. The study collected hand motion data with a 3D pointing task and a gaming task with three popular VR games. Results show that our model can achieve a low Root Mean Square Error (RMSE) of 0.80 cm, 0.85 cm and 3.15 cm from future hand positions ahead of 100 ms, 200 ms and 300 ms respectively across all the users and activities. In pointing tasks, our predictive model achieves an average angular error of 4.0° and 1.5° from the true landing position when 50% and 70% of the way through the movement. A follow-up study showed that the model can be applied to new users and new activities without further training.

CCS Concepts

• **Human-centered computing** → **Human computer interaction (HCI)**; *Interaction paradigms*; *Virtual reality*;

Keywords

Hand motion prediction, Virtual Reality, Kinematics-based Model, User-independent, Activity-independent

ACM Reference Format:
Nisal Menuka Gamage, Deepana Ishtaweera, Martin Weigel, and Anusha Withana. 2021. So Predictable! Continuous 3D Hand Trajectory Prediction in Virtual Reality. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), October 10–14, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3472749.3474753

*Also with The Department of Electronic and Telecommunication Engineering, The University of Moratuwa.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
UIST '21, October 10–14, 2021, Virtual Event, USA
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8635-7/21/10.
https://doi.org/10.1145/3472749.3474753

1 Introduction

Accurate and timely user interaction tracking is essential in virtual reality (VR) to deliver an immersive experience with high-quality graphics and physics simulations. Despite recent hardware improvements, sensing and computations are still very time and energy consuming, particular for standalone VR headsets [16, 63]. Moreover, a significant delay in feedback such as visual, auditory or haptic would lead the user to notice the asynchrony and break the immersion of the virtual environment [48].

Anticipating future interactions can compensate for these issues. Predicting user activities has been shown to reduce delays and improve the experience in interactive applications [9, 41, 42]. In VR, predictive models using eye gaze tracking [3] and head motion prediction [1, 30] can enable pre-rendering of complex graphics scenarios to reduce latency [51]. Commercial VR headsets such as the Oculus Rift-S employ head motion prediction to estimate the head position for the next frame[1]. The most frequently used input method in VR being arm and hand movements, recent works by Henrikson et al. [27] and Clarance et al. [13] develop predictive models for pointing and reaching tasks using hands.

Existing models often predict a particular event such as landing on a target [3, 13, 27] or the collective movement [57]. However, in immersive applications, not only the end of the movement, predicting a continuous movement trajectory is important to identify intermediate events. For instance, Figure 1a shows a user playing boxing with a virtual character where the location and time of collision between the user's glove and the virtual character is not an endpoint, but a location along the trajectory of the movement. By anticipating the trajectory, such events can be predicted to pre-render rich graphics, calculate complex physics, support real-time multi-modal feedback such as haptics and sounds, and even recover short term tracking errors (Figure 1d). Despite the advantages, to the best of our knowledge, little work has applied continuous hand movement trajectory prediction in VR.

This paper contributes a novel *hybrid classical-regressive kinematics model* for continuous 3D hand trajectory prediction for ballistic movements in virtual reality. Our approach uses a coefficient interpolation method between multiple regressions to estimate a unified kinematic model independent of prediction times. We developed and validated this model using data from a user study with 20 participants. Our study collected hand motion data through a structured 3D pointing task and an unstructured gaming task in which the participants played three popular VR games. Both tasks included aimed hand movements, mainly consist of voluntary ballistic movements.

Our findings shows that, a *user- and activity-independent* model performs comparable to personalized and specialized models, and does not require additional training phases. We show that our model can achieve a low average Root Mean Square Error (RMSE) of 0.80 cm (SD=0.12 cm), 0.85 cm (SD=0.14 cm) and 3.15 cm (SD= 0.38 cm) from future hand positions ahead of 100 ms, 200 ms and 300 ms respectively across all the users and activities. In pointing tasks, our predictive model achieves an average angular accuracy of prediction 4.0° (SD = 1.6°) at 50% of the way and 1.5° (SD = 0.6°) at 70% of the way of the movement. Figure 1b and c show a reconstruction of example 3D continuous hand movements for different activities predicted by our model (red) at different prediction intervals (PI) compared to the real movements (blue).

One major challenge of using prediction for continuous hand movements is that prediction models can produce significant errors on some occasions, e.g., during abrupt movements [33]. Error distribution of our model show that such unexpected large errors are minimal in our model with 90% of the errors that occurred are less than 0.6 cm, 0.8 cm and 3.4 cm for 100 ms, 200 ms and 300 ms across all users and activities.

In summary, this paper makes three main contributions:

(1) A kinematics-based prediction approach for *structured and unstructured ballistic 3D hand movements* in VR activities.
(2) A *user- and activity-independent model* with similar performance to personalized and specialized models without the need of additional training phases.
(3) Evaluation of the model through a cross-validation and a secondary study with new participants and new activities.

2 Related Work

This section presents prior work on human motion prediction, its applications in VR and the kinematics of hand motion.

2.1 Human Motion Prediction Techniques

The primary goal of human motion prediction is to predict future positions, poses or trajectories of the human body given past motion data. This is a challenging task due to the non-linear dynamics and time-varying behaviour of the movements. Prior work explored various statistical methods [26, 32, 57] and deep-learning methods [28, 35, 39] to tackle the challenges of human motion prediction.

Template matching techniques, where the movement is compared to a library of known template movements [26, 27] is used in human motion prediction. As template matching techniques require building a motion template library first, it cannot be applied for predicting arbitrary movements, which is a major limitation. Hidden Markov Models (HMM) are also leveraged for human motion prediction [36, 57, 59] in the literature. However, similar to template matching techniques, HMMs also require to be trained onset of seed sequences, limiting its usability for predicting arbitrary movements. Regression models allow capturing the important relationships between the predicted values and the predictor variables, which is a major advantage when compared to other prediction methods. In contrast to classification models, regression outputs a continuous value making it better suited for trajectory prediction without requiring a template library or seed sequences. Prior work explores on various regression methods including end point prediction with polynomial regression [32], Electromyography (EMG) based motion prediction [11].

One class of commonly used deep-learning methods for motion prediction are Recurrent Neural Networks (RNN) due to their capability in modelling sequence-to-sequence learning problems [2,

[1] https://developer.oculus.com/documentation/native/pc/dg-render/ (Accessed on 2021-03-26).

23, 39, 45, 46, 58]. Other classes of deep learning techniques include Convolutional Neural Networks (CNN) [35], graph neural networks [38] and Generative Adversarial Networks (GAN) [28]. However, deep learning methods require large training data sets [30] and have a higher computational overhead, which is not well suited for standalone VR systems with limited computation power.

2.2 Motion Prediction for VR

Motion prediction is a key latency reduction technique used in VR, which allows pre-rendering graphics [34, 51]. However, predictive models also enable novel applications such as foveated rendering [3] and haptic retargeting [13]. Commercial VR headsets such as the Oculus Rift-S predict the head pose for the next frame [2]. Predicting the head motion is beneficial for the VR system as it allows the estimation future focus points of the eyes to pre-render future frames. Head motion prediction for VR is further explored in [6, 24, 25, 53]. Saccadic landing point prediction, which estimates the landing position of the fast eye movements [3, 22, 40] is another technique used for pre-rendering.

Hand motion prediction is important for VR as the hand is the primary method of user interaction with the VR environment. Hahn et al. [26] used template matching for long-term prediction (some tenth of a second) of hand motions in a working environment. Clarance et al. [13] proposed a deep learning model to predict the intended target for reaching activities in VR. Henrikson et al. [27] proposed a template matching technique to predict the ray landing position in a VR environment by integrating the head motion into the predictive model. However, due to the requirement of having a template library, this technique is limited when applying for arbitrary hand motions. Vu et al. [57] specifically focused on predicting hand gestures for VR applications and evaluate their performance in a table tennis game. However, they indicate that prediction heavily relies on the expertise of the user to perform the table tennis strokes accurately and show that accuracy drops with non-expert users.

Our method is related to the kinematics-based regression model for endpoint prediction for stylus targeting tasks by Lank et al. [32]. Assuming the start and end velocities to be zero for the pointing tasks, the authors develop a model of speed over the distance that permits extrapolation. However, their technique is limited to pointing tasks in 2D space. In contrast, we utilize a kinematics-based regressive model for continuous motion prediction for *arbitrary hand movements in 3D space.*

2.3 Kinematics of Hand Movements

Understanding the kinematics of hand movements is an important aspect of hand motion prediction. Prior work explored dynamic end-effector models for hand movements. *Plamondon's Kinematics Theory* and *Vector Integration to Endpoint* [8] have been proposed to explain the dynamics of hand motion. The *Minimum Jerk Model*, which was proposed by Hogan et al. [29], develops a mathematical model to describe voluntary movements of primates. It was later validated for human arm movements [19]. It states that our nervous system tries to make the smoothest movement possible when performing voluntary movements by reducing accelerative transients. Dynamic models have been proposed for specific activities such as mouse pointing [4]. Recently, Bachynskyi et al. [7] proposed a dynamic hand model integrating third-order lag model for modelling mid-air movements for pointing tasks. Compared to both models [4, 7], we focus on modelling hand motion for arbitrary activities, including structured movements such as pointing *and* unstructured movements in VR games.

[2]https://developer.oculus.com/documentation/native/pc/dg-render/ (Accessed on 2021-03-26).

3 Design Goals

This section outlines important requirements for the implementation of our prediction model. It also discusses the novelty of our system and its advantages for VR applications. The following five goals guide our design and implementation:

3.1 Continuous Prediction

Our first goal is to create a model for continuous predictions that is able to estimate the user's hand position at arbitrary time points in the future. In contrast to discreet models, continuous models can be used to predict continuous trajectories of the user's movement. This is important to predict events such as a collision between the user's hand and a virtual object or character as shown in Figure 1a.

3.2 Structured and Unstructured Motion

Most prior work on hand movement prediction studies movements with a limited and controlled set of user actions, e.g., pointing [27]. However, it remains unclear how such models transfer to movements in more generic VR applications such as VR games where user movements are unstructured and less restrictive. Our goal is to create a single model which is applicable for both structured and unstructured tasks in VR.

3.3 User-independent

While the personalization of prediction models can help to improve their accuracy, personalization requires a training phase for each new user before the prediction can be used. This is problematic for settings with many users who might want to use the system for a short time, e.g., museum exhibitions and public displays. Our goal is to create a user-independent model that works for all users. We compare our user-independent model with a personalized kinematics-based model and show it provides a similar prediction performance, without requiring new training data for each user.

3.4 Activity-independent

VR applications are used for a wide variety of activities in many domains, including entertainment [60], industry [21, 56], health care [54, 62], sketching [17], and education [12, 20]. The user interactions and activities in different applications can be drastically different. For example, they can be fast or slow, small or large, and precise or vague. A model specialized in a single activity would require new training sets for each application. Hence, application developers would need to acquire training data and validate it before they could use the model. We believe a prediction model should be generalizable to different activities to allow for simple integration into different fields. In this paper, we develop such an

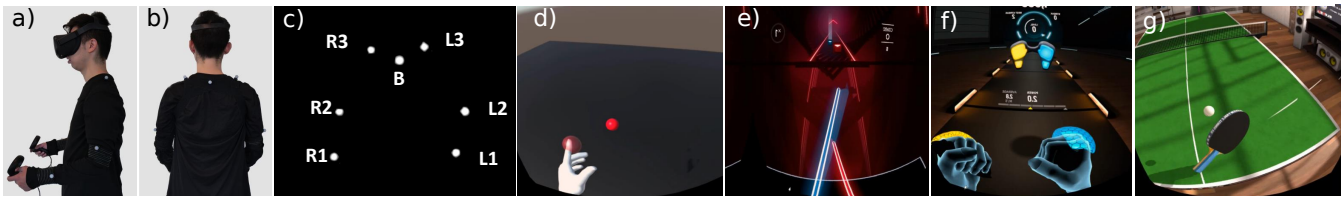


Figure 2: User study setup: a) side and b) back view of a participant with trackers attached. c) Tracker positions and labels. VR tasks in the user study: d) Custom VR application used in Task 1. VR games of Task 2: e) ©Beat Saber, f) ©FitXR and g) ©Eleven.

activity-independent model and show that it performs equally well compared to specialized kinematics-based models.

3.5 Explainable Prediction

Our final goal states that our prediction model should be explainable to researchers and practitioners. This means that the used methods for the prediction should be transparent and relate to movement parameters. This goal contrasts with typical deep learning techniques such as Long-Short Term Memory (LSTM) neural networks. Although explainable deep learning is under active research [61], most current deep learning methods lead to black-box models. Instead, we contribute a kinematics-based regression model. Such regression models have the advantage that their inner working is fully transparent and all parameters map to movement parameters, such as velocity, acceleration and jerk.

To achieve these design goals, we contribute a novel kinematics-based regression model for predicting arm movements. The model is trained and validated with structured and unstructured hand movement data, which we collected in a user study.

4 User Study

We conducted a controlled experiment in a lab environment using a series of VR applications to validate our strategy for continuous hand movement prediction. To show our model would hold for a wide range of hand movements, we gathered hand motion data on applications that focused on structured hand movements (reaching and pointing task) and unstructured hand movements (games).

4.1 Participants

We recruited 20 healthy participants (7 female, 13 male; mean age 22.4y SD=5.1y). 19 participants were right-handed and 1 participant was left-handed. Participants who require glasses were allowed to wear them during the study under the head-mounted display. Each participant was given a long sleeves t-shirt to wear, and trackers were mounted on the t-shirt during the study preparation.

The study was conducted according to COVID-19 safety guidelines and the study received ethics clearance from the Human Research Ethics Committee (HREC) of the University of Sydney (Application number: 2019/553). All apparatus was cleaned after each study adhering to the Australian Government regulations.

4.2 Apparatus

The hand motion data were recorded with the OptiTrack motion capture system (version 1.10.2) with eight cameras mounted on the ceiling. The participant wore seven trackers on the upper body, which included trackers in participant's wrist, elbow and shoulder on each arm, including a tracker on the back as shown in Figure 2a and b. The tracking data was recorded at 100 Hz. Trackers were labeled as shown in Figure 2c.

We used an Oculus Quest as the VR headset and Oculus Touch handheld controllers. To collect the structured hand movements, a custom application was developed with Unity3D (Figure 2d) where the controller position and orientation were recorded at a rate of 72 Hz in addition to the data from the OptiTrack system. For the VR games, no motion data from the Oculus was recorded as our 3rd party applications could not access the sensor data.

In addition to the sensor data, the VR screen was mirrored to a computer for screen recording. The whole user study session was video recorded with a camera.

4.3 Study Design

The study was divided into two tasks. Each task collected hand movement data for different activities. The first task (T1) samples *structured* hand movements between indicated points in a three-dimensional pointing study. The second task (T2) samples *unstructured* hand movements from three VR games to increase the external validity of our data set. Both tasks include aimed movements in VR, which contains an initial voluntary ballistic movement followed by a corrective movement [37]. For both tasks, participants were in a standing posture and were instructed not to move their legs during the tasks. However, movements such as twisting, bending the body and leaning to the sides without moving their feet were allowed.

The study was conducted in a single session taking approximately 1 hour per participant. We followed a within-subjects design for the study, where task order was counterbalanced. Participants were allowed to practice until they felt comfortable with the task and to take breaks during the study to prevent fatigue.

T1: Structured Movement via 3D Pointing

The first task followed a repeated-measures, within subject design and collected data from hand movements in all three dimensions. This study opted for a structured approach, where the participant was asked to move their hand towards virtual point targets. The targets were represented as 3D spheres with a diameter of 5 cm. For each participant, the starting hand position was initialized before the experiment with the participant placing their hand in-line with the shoulder while making an approximately 90°angle between their forearm and upper arm similar to the study conducted by Cha et al. [10]. At the start of the study, the participant places the virtual index finger on the initial position as seen from the VR headset

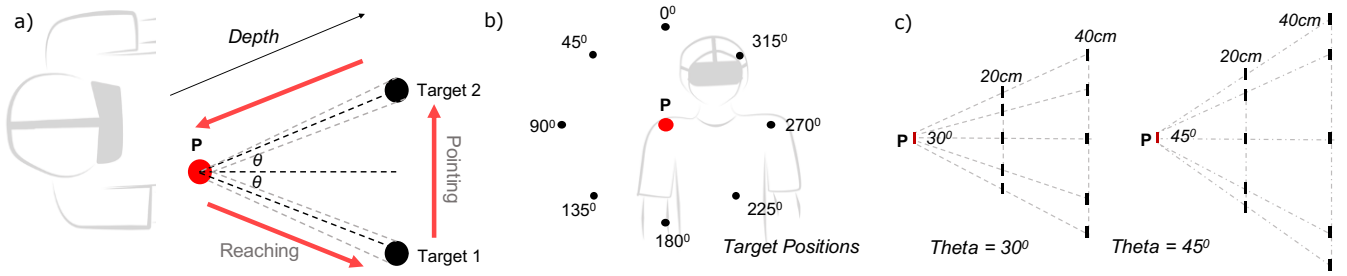


Figure 3: a) Target positions for a single iteration of the reaching and pointing task (T1). The participant started each trial at starting position *P*, reached for the first target before pointing to the second target. b) Front-view and c) side-view of the target arrangement.

(red circle in Figure 3a and b). When the first target appears, the participant moves their finger from the initial target to the first target (Figure 3a-*reachingt*). A change of colour indicated to the participant that they successfully reached the target. Afterwards, a second target on the opposite side of the same circle appears and the participant moves their hand from the first target to the second (Figure 3a-*pointing*). Finally, the participant moves their finger back to the initial target which completed one iteration of the task. The participant was asked to do all the movements as quickly and accurately as possible.

The targets were equally distributed in four circles in front of the user in 45° increments (Figure 3b). The circles differed in their distance to the start position (*Depth*) and their angular deviation (*Theta*) as shown in Figure 3c. *Depth* was measured from the starting position of the hand. Our initial experiments indicated that when an angular deviation of 60° is used, it is difficult to spot all targets due to the limitations of the viewing angles of the VR headset. To limit the duration of the study, we evaluated only two distances (20 cm and 40 cm) and two angular deviations (30° and 45°). The study contained five iterations of 32 movement blocks (2 depths × 2 angles × 8 positions). Participants could take a break after each iteration to prevent fatigue. The movement order was randomized for each iteration to avoid biases. In total, we collected 160 trials (32 movement blocks × 5 iterations) for each participant.

T2: Unstructured Movement via VR Gameplay

Arm movements in virtual reality applications can be complex. They combine various properties, such as direction, curvature, distance, and speed. In Task 2, we collected arm movement during VR gameplay to ensure our prediction is working in realistic VR scenarios. To cover a wide variety of movements, we asked participants to play three popular VR games (see Figure 2e–g).

- *BeatSaber*[3] is a rhythm game where the user needs to slash small cubes with two sabers on both hands. The game contains fast directional slashing movements from both hands.
- *FitXR-Box*[4] is a rhythm game where the user needs to hit small targets using both hands. The game contains fast and powerful forward movements of both hands which closely resembles boxing.
- *Eleven*[5] closely resembles real-world table tennis strokes with the dominant hand.

With an initial study, we observed that each game had different move dynamics. Due to the fast and wide slashing movements in *BeatSaber*, it had the highest average speed (0.72 m/s) and highest spans in horizontal and vertical directions (0.85 m, 0.95 m). *FitXR-Box* had a much lower horizontal span (0.57 m) and a highest span in frontal direction (0.75 m) as expected from the boxing movements. Meanwhile, move dynamics of *Eleven* varied greatly among users, as individuals have different styles for playing table tennis.

Each game was played for approximately three minutes. Before recording the data, participants could get familiar with the game by following the in-game tutorials and playing the game for 1 minute. Participants could take breaks between games.

4.4 Data Preparation and Presentation

We used data from the Optitrack motion capture system for training and testing of our prediction models. To reduce the noise introduced from the trackers, we apply a Gaussian Filter to smooth the trajectory similar to prior work [27]. For the hand trajectory we used the position of the marker R1 or L1, 3.5 cm above the wrist (Figure 2e) relative to the body frame with reference to centre of the shoulder plane, i.e., marker B shown in Figure 2e.

4.4.1 Training – Testing Data Split: The data collected was split to training and testing portions upon collection. We used 30 s (<10% of data from T1 and 15% of data per game from T2) as the training set. The rest was allocated for testing. Therefore, all the testing we conducted throughout this paper was conducted on data independent from the training data.

5 Hybrid Kinematic Regressive Model

Our predictive model uses classical kinematic equations as the base of a multi-layer regressive model. This section explains the process used to develop the model with an overview of metrics used to evaluate prediction accuracy, classical kinematics, prediction-time dependent kinematics regression and inferred prediction-time independent kinematic modelling.

[3]https://beatsaber.com (Accessed on 2021-03-26).
[4]https://fitxr.com (Accessed on 2021-03-26).
[5]https://linktr.ee/elevenvr (Accessed on 2021-03-26).

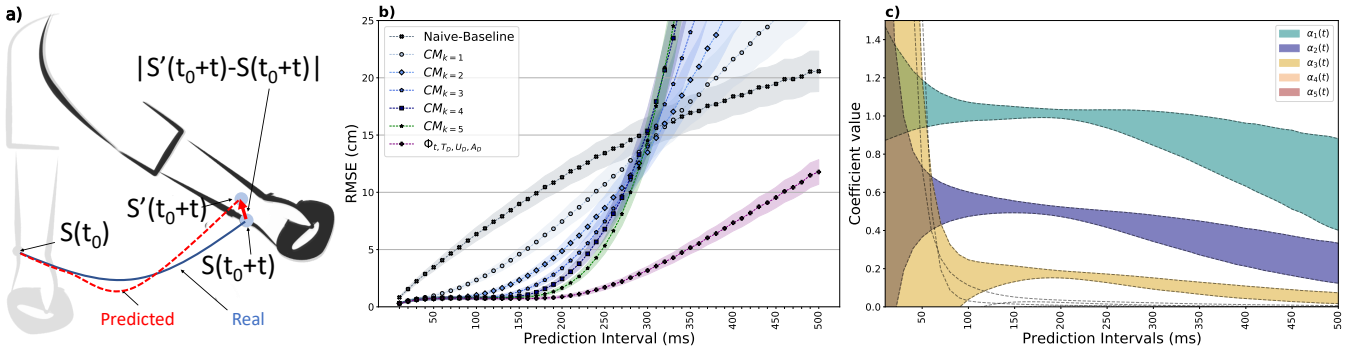


Figure 4: (a) Hand trajectory prediction at a given time t_0, from the position $S(t_0)$ to position $S(t_0 + t)$ at time t; (b) Average RMSE (in cm) from classical kinematic models showing improvement with each added derivative of motion compared to a series of prediction-time dependent regressive kinematic models Φ_{t,T_D,U_D,A_D} developed and tested for each task, user and axis; (c) Distribution of 95% confidence interval of five α_ns from Φ_{t,T_D,U_D,A_D} models.

5.1 Metrics for Prediction Accuracy

Before developing the model, it is important to identify metrics to compare the performance of the prediction accuracy. In this section, we explain our definition of the prediction error per each instance and the aggregated accuracy measures we used to compare the developed models.

Figure 4a depicts the prediction error $|S'(t_0 + t) - S(t_0 + t)|$, the distance between the predicted $(S'(t_0 + t))$ and the actual $(S(t_0 + t))$ hand locations where t_0 and t are initial time and the prediction time interval respectively. To add a relative comparison, it is common in predictive models to use the initial point as a *naive prediction* (i.e. $S'(t_0 + t) = S(t_0)$) [27]. In other words, *naive prediction* assumes the hand does not move during the predicted time period. As $S(t_0 + t)$ is the ground truth at time $t_0 + t$, *naive prediction* error can be expressed as $|S(t_0 + t) - S(t_0)|$, which is the actual displacement of the hand during the predicted time period. Therefore, we take *naive prediction* as a *naive baseline* to compare our prediction models with the actual movement.

The prediction errors defined above apply to a single point in the trajectory. Since our aim is to predict continuous hand motion trajectory, we need to aggregate the per point prediction errors to a single metric. Root Mean Square Error (RMSE), or RMSE of $|S'(t_0 + t) - S(t_0 + t)|$, is a commonly used aggregated metric to evaluate predictive trajectories [14]. Another commonly used metric is the Mean Absolute Error (MAE) [31]. We chose RMSE as the primary metric since it gives a higher weight for larger errors in prediction due to the squared terms. Therefore, RMSE is better suited when higher errors are particularly undesirable, which is important in trajectory prediction.

Furthermore, Nancel et al. [42] showed that while RMSE provides an overall measure for the accuracy, it does not capture side effects of latency compensation methods from a user's perspective. They identified seven spatial accuracy metrics to capture the side effects for touch location prediction in 2D touchscreen. We extended their metrics Lateness (slow to react to the actual movement), Over-anticipation (over-react to the actual movement) and Wrong Orientation (not going in the same direction as the motion) to 3D space. We also use these additional metrics to compare our final model with the baselines.

5.2 Classical Kinematics of Motion

Classical kinematics can be used to model behavior of moving bodies with respect to a frame of reference. As shown in Figure 4a, given the three-dimensional hand position vectors $S(t_0)$ and $S(t_0+t)$ at times t_0 and t respectively, $S(t_0 + t)$ can be expressed as:

$$S(t_0 + t) = \sum_{n=0}^{k} \frac{1}{n!} \frac{\mathrm{d}^n S(t_0)}{\mathrm{d}t^n} t^n \tag{1}$$

This equation assumes that $\frac{\mathrm{d}^k S(t_0)}{\mathrm{d}t^k}$ to be constant. For instance, movements with constant acceleration, where the second derivative is constant. We can set $k = 2$ to get the equation for displacement $D(t)$ (i.e., $D(t) = S(t_0 + t) - S(t_0)$), in the familiar form $D(t) = ut + \frac{1}{2}at^2$, where $u = \frac{\mathrm{d}S(t_0)}{\mathrm{d}t}$ and $a = \frac{\mathrm{d}^2 S(t_0)}{\mathrm{d}t^2}$ are velocity and acceleration. However, acceleration of the hand movements are not constant and changes with time with the forces exerted by muscles. Therefore, it is important to identify a k, which is low enough to make the model plausible and high enough to accommodate real hand movements. Hogan et. al. show that voluntary hand movements in mammals follow a *minimum jerk law*, where *jerk* (j) is the 3rd derivative of the motion. The law states that the nervous system tries to make the smoothest movement possible by reducing accelerative transients. The law further states that pointing movements would have a constant *crackle* (c), which is the 5th derivative of the motion [32, 52]. Therefore, we tested the classical models (CM_k) for $k \in [1, 5]$, assuming a constant *crackle*. For instance, $CM_{k=5}$ results in $S'(t_0 + t) = S(t_0) + vt + \frac{1}{2}at^2 + \frac{1}{6}jt^3 + \frac{1}{24}st^4 + \frac{1}{120}ct^5$, where s is the 4^{th} derivative named *snap*.

We used the data collected in our experiment to calculate future positions of the trajectory using these classical models (CM_k) at 50 prediction intervals at $10ms$ steps from $10ms$ to $500ms$. Figure 4b shows the average RMSE across users and activities under each k values, where RMSE values are calculated for the prediction error $|S'(t_0 + t) - S(t_0 + t)|$. Figure 4b also shows a comparison with the

naive baseline where $S'(t_0+t) = S(t_0)$, which is the classical model with $k = 0$ ($CM_{k=0}$).

The classical model with $k = 5$, $CM_{k=5}$, generated low average RMSEs until $t = 0.16s$ ($mean = 0.5cm$, $SD = 0.07cm$). However, after $t = 0.18s$, data shows that the time varying nature of real hand movements are difficult to capture in these equations and it exponentially overestimate the movements. In Figure 4b, it is important to note each added derivative contributes to better predictions at smaller prediction intervals, but at larger prediction intervals increasingly contributes to the error. Since classical models with $k = 5$, $CM_{k=5}$, is the best performing classical model, we use it as a *non-naive baseline* baseline for comparison with our models.

5.3 Prediction-Time Dependent Kinematics Regression

In Figure 4b, it is evident that the motion characteristics such as u, a, j, s and c are essential to estimate future locations; with increasing prediction intervals, their contributions with constant weights in the classical kinematics (e.g., $\frac{1}{2}$, $\frac{1}{6}$, $\frac{1}{24}$, ...) leads to an exponential error. This can be explained by accumulation of errors in the integrative nature of the equation (i.e. $u = \int_{t_0}^{t} a$). Therefore, to counter future changes to higher order derivatives, even beyond *crackle* (c), prediction-time (t) dependent weights for each derivative in Equation 1 are needed. Essentially, this can be expressed as:

$$D(t) = \begin{bmatrix} vt & at^2 & jt^3 & st^4 & ct^5 \end{bmatrix} \times \begin{bmatrix} \alpha_1(t) \\ \alpha_2(t) \\ \alpha_3(t) \\ \alpha_4(t) \\ \alpha_5(t) \end{bmatrix} \tag{2}$$

Where each $\alpha_{n(t)}$ represents a three dimensional variable (for three axis of movement), specific to a given prediction interval t, and $D(t) = S(t_0+t) - S(t_0)$. We considered identifying each $\alpha_{n(t)}$ as a regressive problem. To develop regressive models for each t, we used the training portion of the movement data collected in our user study. Our attempt to fit a model that takes prediction time as an independent variable failed with $R^2 < 0.3$ even for prediction times $t < 250ms$. Therefore, we considered creating *prediction time dependent* models where inputs to the regression was $\{v_{(t_0)}t, a_{(t_0)}t^2, j_{(t_0)}t^3, s_{(t_0)}t^4, c_{(t_0)}t^5\}$ where t_0 is current sample time and t is the prediction time interval.

For increased granularity across prediction time (t), we used 50 time steps in $[10ms, 500ms]$ range in par with our sampling interval $10ms$. To accommodate the user, task and prediction time interval dependent nature of the movements, we regressed 4000 ($50 - timesteps \times 20 - users \times 4 - task$) independent models resulting 5 α_ns per model (i.e., 4000 - α_1, 4000 - α_2, etc.), each 1×3 vector representing three axes. We represent these three models as Φ_{t,T_D,U_D,A_D} representing, prediction **t**ime, **T**ask, **A**ctivity and **U**ser **D**ependent nature of the model.

To validate each model, we used the test data portion from the user study to calculate RMSE for each model with respect to the conditions the models created against, i.e. relevant to prediction interval (t), users (U) and tasks (T). Figure 4b shows the average RMSE across each model in comparison to the *Naive Baseline* and Classical models (CM_k). A Mann-Whitney test indicated that for higher prediction intervals of $t > 160ms$, the 5th order classical model has a prediction error ($median = 0.44$) significantly greater than for Φ_{t,T_D,U_D,A_D} ($median = 0.35$), $U = 2622$, $p = 0.024$, and shows a proportional increment of error with t. Therefore, Φ_{t,T_D,U_D,A_D} can be used as the best possible model, however, it is too specific (user, task) and the prediction time, and will be implausible to apply in a realistic scenario.

5.4 Inferred Prediction-Time Independent Kinematic Modeling

The major challenge of the Φ_{t,T_D,U_D,A_D} is that any practical system needs to maintain a series of models (4000 in our case) for each specific scenario. And each new factor will exponentially increase the number of models. Furthermore, any change in the coordinate system will need either coordinate translation or re-calibrations. For real-life applications, a general model is desired, where minimum or no additional training needed for each new scenario. However, Φ_{t,T_D,U_D,A_D} showed great promise in the regressive nature of the hand movements. A generalizable regressive model would be ideal for the fast prediction of future movements.

The fitted α_ns in the Φ_{t,T_D,U_D,A_D} models can be used as a basis to develop an *inferred regression model*, which can be generalizable and independent of the scenario. Figure 4c examines the distribution of α_n in each model in Φ_{t,T_D,U_D,A_D}, with the regions indicating the 95% confidence interval at each prediction time. The model shows a converging pattern towards higher prediction intervals (t), but at lower ts, it shows high variations. This is due to the regressive model trying to accommodate large variability of higher derivatives of movements (α_3, α_4 and α_5). This intern affects the regressed coefficient of lower derivatives. A straightforward approach is to create a general model of each α_n as a regression of the distribution in Figure 4c. However, this resulted in poor fitting with $R^2_{\alpha_1} = 0.77$, $R^2_{\alpha_2} = 0.38$, $R^2_{\alpha_3} = 0.07$, $R^2_{\alpha_4} = 0.07$, $R^2_{\alpha_5} = 0.08$ for each α_n. Also, we observed that, classical models perform equally well until t reaches $0.16s$ (first statistically difference as compared in Figure 4b). Therefore, a hybrid approach of classical and regressive models is necessary to capture the high performing parts of each method. We considered two piecewise approaches to create two hybrid models.

5.4.1 Direct Classical + Regressed Piecewise Models (Φ'): In this approach, we directly replaced the first portion of the regressive model until the time interval (t), where RMSE reached a statistically significant advantage with the model Φ_{t,T_D,U_D,A_D}. Specifically, we created a piecewise split of the coefficient function at $t = 0.16s$.

$$\alpha'_n(t) = \begin{cases} \alpha^c_n & t < 0.16s \\ \alpha^r_{n(t)} & t \geq 0.16s \end{cases} \tag{3}$$

Where, $\alpha^c_n = \frac{1}{n!}$ are constant classical α values and $\alpha^r_{n(t)}$ represents the regressed and time dependent values, which can be expressed as a second order polynomial $\alpha^r_{n(t)} = \beta_0 + \beta_1 t + \beta_2 t^2$. In order to compare the effect of task, user and axes dependency, we regressed $\alpha^r_{n(t)}$ with each factor dependent and a final model which is completely independent of all the factors. Each of these models are denoted by Φ'_{T_D,U_I,A_I}, Φ'_{T_I,U_D,A_I}, Φ'_{T_I,U_I,A_D} and Φ'_{T_I,U_I,A_I} where T, U, and A indicates task, user and axes and D or I indicates

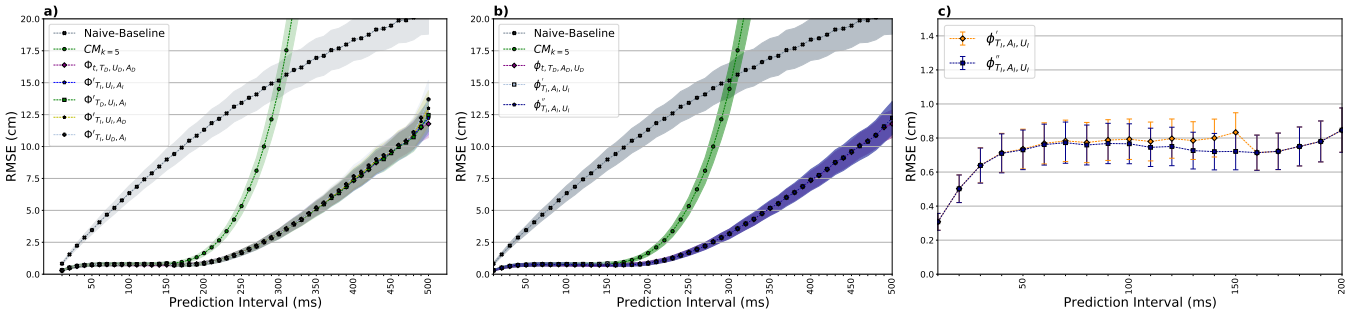


Figure 5: a) Average RMSE for models Φ'_{T_D,U_I,A_I}, Φ'_{T_I,U_D,A_I}, Φ'_{T_I,U_I,A_D} and Φ'_{T_I,U_I,A_I} compared to the baselines; b) Average RMSE of model Φ''_{T_I,U_I,A_I} compared to Φ_{t,T_D,U_D,A_D}, Φ'_{T_I,U_I,A_I} and the baselines; c) Comparison of the two models Φ'_{T_I,U_I,A_I} and Φ''_{T_I,U_I,A_I} in $t = [10, 200]$ region;

dependency or independency. The series of regressive models had an average R^2 of 0.85 with a SD=0.06.

Figure 5a shows the average RMSE against prediction interval for four models compared to the Φ_{t,T_D,U_D,A_D} model and the baselines. Surprisingly, we found no significant difference (Mann-Whitney test) of the task, user or axes dependency in the comparison. This is indicative of a task, user and axes independent model that can be developed for hand movement prediction without any personalized training. However, in the results, we noticed a sudden drop of RMSE at the piecewise junction ($t = 0.16s$). This is indicative of a gradual transition from classical to the regressive model may further increase the performance.

5.4.2 Interpolated Classical + Regressed Piecewise Models (Φ''): In the second hybrid model, our goal was to implement a gradual transition from the classical model to a regressive model. Rather than setting fixed classical values when $t < 0.16s$, we included an interpolation between the classical model and the regression model, resulting in new piecewise definition of the function:

$$\alpha''_n(t) = \begin{cases} \alpha^c_n + \frac{\alpha^r_{n(0.16s)} - \alpha^c_n}{0.16} t & t < 0.16s \\ \alpha^r_{n(t)} & t \geq 0.16s \end{cases} \quad (4)$$

This model holds the same behaviour for Φ' for prediction intervals greater than 0.16 s. Since all independent approaches should hold for the most challenging prediction intervals, we tested RMSE for this model only for all independent regressive configuration, Φ''_{T_I,U_I,A_I}. Figure 5b shows the average RMSE against prediction interval for Φ''_{T_I,U_I,A_I} compared to the Φ_{t,T_D,U_D,A_D}, Φ'_{T_I,U_I,A_I} and the baselines. We did not observe any significant difference between the models with the Mann-Whitney test. Results show RMSEs of 0.80 cm (SD=0.12 cm), 0.85 cm (SD=0.14 cm) and 3.15 cm (SD=0.38 cm) from future hand positions ahead of 100 ms, 200 ms and 300 ms respectively across all the users and activities.

Figure 5c shows a comparison of the two models Φ'_{T_D,U_I,A_I} and Φ''_{T_D,U_I,A_I} in $t = [10, 200]$ region where the two models differ. The figure shows that Φ''_{T_D,U_I,A_I} outperform Φ'_{T_D,U_I,A_I} in the region $t = [70, 150]$ and that average error of Φ''_{T_D,U_I,A_I} has improved in the prediction interval $t = [70, 150]$. However, we did not find a statistical significance with the Mann-Whitney test. The transition of the error from classical to regressive model has become smoother in the Φ''_{T_D,U_I,A_I}. Therefore, we conclude that the combination of the interpolated classical model and the task, user and axes independent regressed model, Φ''_{T_D,U_I,A_I}, resulted in the best outcomes for prediction.

5.5 Results

Our final predictive model (Φ''_{T_D,U_I,A_I}) achieves RMSEs of 0.80 cm (SD=0.12 cm), 0.85 cm (SD=0.14 cm) and 3.15 cm (SD=0.38 cm) from future hand positions ahead of 100 ms, 200 ms and 300 ms respectively across all the users and activities. Compared to *naive baseline* and $CM_{k=5}$, our model reduces RMSE by 79.1% and 78.1% for 300, 500 ms. Our model achieves MAEs of 0.28 cm (SD=0.19 cm), 0.33 cm (SD=0.23 cm) and 1.97 cm (SD=1.10 cm) for 100 ms, 200 ms and 300 ms PIs, which is in the same order as the tracking errors of the commercial VR headsets (Oculus Quest 0.69cm, Samsung Galaxy S9 1.69cm) [44].

To compare our predictive model for pointing tasks, we predicted the landing position of each pointing task at 10% increments along the way from start to end similar to Henrikson et al. [27]. Figure 6a shows the average angular accuracy of models Φ''_{T_D,U_I,A_I} compared to the *naive baseline* and $CM_{k=5}$. New Φ''_{T_I,U_I,A_I} creates average angular accuracy of prediction 4.0° ($SD = 1.6°$) at 50% of the way and 1.5° ($SD = 0.6°$) at 70% of the way of the movement. At 50% of the way, Φ''_{T_I,U_I,A_I} model shows a reduction of angular error by 74.5% and 74.4% compared to the *naive baseline* and $CM_{k=5}$ respectively. Similarly, for 70% of the way, the error reduction of our model is 82.4% and 66.9%.

We further studied how our models perform for reaching tasks, where the activity involves radial outward movements shown in Figure 3a. In this task, we report prediction accuracy as distance to target error at 10% increments along the way from the start. Figure 6b shows the average distance error of the model Φ''_{T_I,U_I,A_I} compared to distance from the target as the baseline. . Φ''_{T_I,U_I,A_I} model achieves an average accuracy of prediction 0.51cm ($SD = 0.24cm$) at 50% of the way and 0.41cm ($SD = 0.20cm$) at 70% of the way of the movement. The average accuracy of the model reaches the pointing target's diameter 5cm at 47% of the way. Compared to baseline and the $CM_{k=5}$, Φ''_{T_I,U_I,A_I} model achieves a reduction of

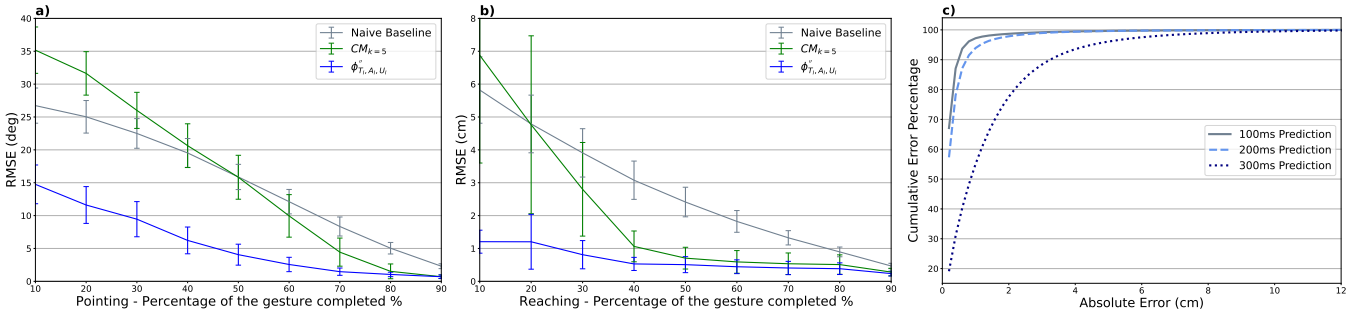


Figure 6: a) Average angular error for the pointing task at 10% increments along the way from the start for Φ''_{T_I,U_I,A_I}; b) Average distance error of the model Φ''_{T_I,U_I,A_I} for reaching task compared to the distance from the target as the baseline; c) Cumulative error distribution for the Φ''_{T_I,U_I,A_I} model for different PIs.

error by 78.9% and 28.0% at 50% of the way. Similarly, for 70% of the way, the error reduction of our model is 69.3% and 24.1%.

We also calculated the side effects introduced in [42] for our final model Φ''_{T_I,U_I,A_I}. For Lateness, Over-Anticipation and Wrong Orientation we report 1.21 cm, 1.41 cm, 27.63° for our model and 3.03 cm, 2.77 cm, 39.75° for 5th order classical model ($CM_{k=5}$). In comparison, our model reduces these side effects by 60%, 49% and 30% respectively compared to the 5th order classical model.

Furthermore, Figure 1b–c shows a comparison between the predicted trajectory by our model (red) and the real motion trajectory (blue) for activities *BeatSaber* and *FitXR-Box* for PI $[10, 300]$ ms, showing how the predicted trajectory closely follow the real one.

Since this model is axes independent, it is resilient for changes in the coordinate system given that it is with respect to the body. Also, $\alpha''_n(t)$ is a single axis function since all axes share the same coefficients. Therefore, $\alpha''_n(t)$ for displacement vector calculation using the Equation 2 can be expressed as:

$$\begin{bmatrix}\alpha_1(t)\\ \alpha_2(t)\\ \alpha_3(t)\\ \alpha_4(t)\\ \alpha_5(t)\end{bmatrix} = \begin{bmatrix}\begin{cases}1.0000 + 0.1693t & t < 0.16s\\ 1.0174 + 0.4547t - 2.4655t^2 & t \geq 0.16s\end{cases}\\ \begin{cases}0.5000 + 0.1837t & t < 0.16s\\ 0.6550 - 0.7458t - 0.2458t^2 & t \geq 0.16s\end{cases}\\ \begin{cases}0.1667 + 0.1151t & t < 0.16s\\ 0.2637 - 0.5122t + 0.1308t^2 & t \geq 0.16s\end{cases}\\ \begin{cases}0.0417 - 0.0343t & t < 0.16s\\ 0.0739 - 0.2809t + 0.2836t^2 & t \geq 0.16s\end{cases}\\ \begin{cases}0.0083 - 0.0064t & t < 0.16s\\ 0.0150 - 0.0569t + 0.0555t^2 & t \geq 0.16s\end{cases}\end{bmatrix} \quad (5)$$

5.6 Error Analysis

Figure 6c shows that our model often make smaller errors and when large errors occur, they are less frequent. For instance, 90% of the errors that occurred are less than 0.6 cm, 0.8 cm and 3.4 cm for 100 ms, 200 ms and 300 ms across all users and activities. We considered the dominant hand of the participants for our analysis and the left-handed user was not given special treatment. Surprisingly, for the left-handed user, the model performed better than for the average of all users with RMSE of 2.34 cm of 300 ms prediction. We did not observe any impact on accuracy with the VR expertise of the participants. We also did not find a direct correlation between the movement kinematics and the prediction error. Anecdotally, we observed that large errors occur with large directional changes as shown in Figure 1e which need further investigation.

6 Verification of the Model

To assure the generalizability of our model, all the accuracy measures presented in section 5 is conducted on a test data set, which was not used to derive the model. For instance, of each task (pointing and games), less than 15% of the data is used for training, and the rest is used for testing. Especially with gradually progressing tasks such as VR games, it is fair to assume a large portion of the movements in the training data would differ from that of testing. However, our model uses a portion of data from each participant and each activity. Therefore, to investigate overfitting or selection biases, we conducted two further explorations: (1) *cross validation* of the methodology and (2) a new user study with *two new activities* to apply the model to a completely independent scenario.

6.1 Cross Validation

We conducted 4 folds of cross validation by separating 25% of the users as test data and calculated the average RMSE across all folds for all the activities. Figure 7a shows the results of the cross-fold evaluation compared to the collective model's RMSE. The average RMSE of cross validation was lower than that of the collective model at the higher end of the prediction, but we did not observe any significant differences. In addition, we built 20 models adding users 1 by 1 for training and tested them across the rest of the users progressively. After 13 users the model stabilizes with the change of error 0.99 mm for consecutive models. However, this may change with other user factors (e.g., age, injuries, etc.).

6.2 New Users and Activities

To verify the applicability of the model to a new user group and a new set of activities, we recruited 3 participants (age 22 to 30, one female), and asked them to perform two new tasks. *First task was performing free form sweeping movements* including flexion, extension, abduction and tracing a horizontal figure of 8 parallel to

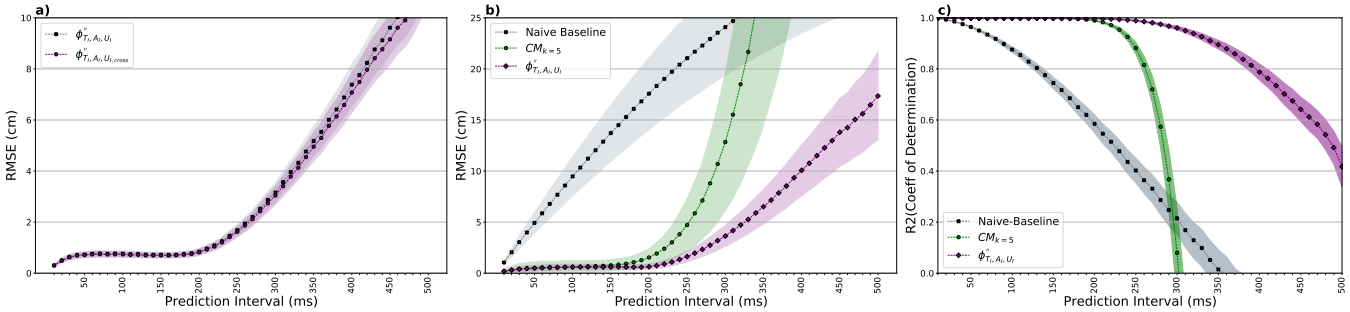


Figure 7: a) Average RMSE (in cm) of four-fold cross validation of the approach; b) Average RMSE for three new users with two new tasks compared to the baseline. c) Average R^2 score for models Φ''_{T_I,U_I,A_I} compared to the baselines.

the body plane. We selected these movements since it covers a larger area of the space and they are dissimilar to the movements used in the first study. For the second task, we selected a dancing game *FitXR Dance Mode*, which also consist of movements dissimilar to that of the previous study. Participants performed each task for approximately 6 minutes and 3 minutes consecutively. Data collection and recording followed the same procedure as the first study. Figure 7b shows the average RMSE for the new tasks against the prediction time in comparison to the overall RMSE calculated for the original users. Results show a low average RMSEs of $0.55cm$ ($SD = 0.61cm$), $0.61cm$ ($SD = 0.59cm$) and $3.36cm$ ($SD = 1.37cm$) at prediction intervals $t = 100, 200, 300$ respectively. This shows even with new users and significantly different tasks, the model performs fairly well.

7 Discussion

In this paper, we presented a user- and activity-independent parametric kinematic model for 3D hand trajectory prediction in VR environments. Our results show that the model produces better performance compared to a non-naive baseline and needs little additional training for new users and activities. Furthermore, the simplicity of the model creates low computational overhead, which is an important factor for predictive systems. This section discusses the implications of the presented system and future perspectives.

Timely and high-quality multi-modal feedback is critical to implement realistic VR systems. Despite rapid progress in hardware developments, high-quality graphics and realistic physics simulations (e.g., interactions with fluids) are still very time-consuming in stand-alone VR systems. A predictive model can help the VR systems to forecast future events (e.g., Figure 1a collision) and pre-renders complex graphics in advance [51]. Offloading heavy computational tasks to remote clouds is another solution. However, offloading introduce communication delays (40 ms) and online computing delays (100 ms) [16]. A simple predictive model like ours can significantly contribute to counter these delays.

Another important area where a predictive model can be instrumental is to overcome asynchrony in multimodal feedback. For instance, delays as small as 50 ms in haptic feedback are noticeable to users in VR [15]. However, in commercial VR systems, delays in tracking (22 ms) [43], actuation of haptic system (33 ms)[6] and other communication delays can easily exceed the required latency. Researchers also experiment with other types of sensory feedback such as thermal [47], wind [49, 55], smell and taste [50] where the onset delay in actuation is significant. These numerous delays lead to noticeable latency that breaks the immersion of the virtual environment [48]. A prediction model like ours can compensate for these delays by forecasting the future interactions of the users (Figure 1a) with minimum overhead.

Predictive models have many other potential applications in VR beyond latency reduction. One such example is an error correction mechanism. Intermittent loss of tracking data is common in motion tracking systems due to occlusion or lighting issues. We observed such losses in the data we collected. Predictive data can be used to reconstruct missing hand trajectories as shown in Figure 1d, where circled areas demonstrate how missing path is redrawn using our model. Other areas where prediction can be used include haptic retargeting [5, 13] which enables the reuse of a single physical object to provide passive haptics for multiple virtual entities.

Our model performed surprising well without any further training for new users. We believe the success of the approach is due to commonalities of movement kinematics at short time intervals (e.g., 300 ms). However, we further explored other possible factors that could degrade the performance of the model. The *generalizability* was an important concern and we evaluated our model against the dancing move set of CMU Motion Capture Dataset [18] to further explore the external validity. Our model achieved RMSE of 3.86 cm for 300 ms prediction, which is an improvement of 83.6% compared to $CM_{k=5}$. It is important to notice that, unlike our dataset, this data includes hand motion data when the user is moving their feet. Another concern was the grounded and third-person perspective of the OptiTrack system we used and whether the model will apply to first-person wearable tracking systems used in most commercial VR headsets. We tested our prediction model for the Oculus Controller data we collected from the structured task (T1). This also gave us the opportunity to test if the model is completely independent of the tracking coordinate system and the sampling rate, where Oculus records data at 72 Hz in a coordinate system with respect to the participant's head. Furthermore, the marker we used for the hand

[6] https://developer.oculus.com/documentation/native/pc/dg-input-touch-haptic/ (Accessed on 2021-03-26)

was 3.5 cm above the wrist while Oculus tracks the controller held in hand, making the data truly of the hand location. With Oculus data, our model achieved a RMSE of 2.39 cm for 306 ms prediction time, which is a smaller error compared to OptiTrack prediction data at 300 ms.

8 Limitations and Future Work

Our model was primarily trained and tested on aimed movements, which contained a majority of voluntary ballistic movements [37]. Further investigations are required on how the model perform for other types of movements (i.e., steering movements). However, the dancing task in the follow-up study (*FitXR-Dance*) was partly a steering task, where the participant copied the movements of a VR character simultaneously. For this task, error reduction is 76.6% at 300 ms with respect to $CM_{k=5}$ which is comparable to 78.1% error reduction in other tasks. We recommend our model is best used for predicting ballistic movements up to 340 ms, as the model's R^2 score decreases below 0.9 (Figure 7c) beyond this prediction interval, indicating that the confidence of the model deteriorates beyond this interval.

All participants in our study were between 18 and 39 years old, which is currently the core demographics for VR applications[7]. The findings and movement models we discussed in this paper might differ for other age groups and people with injuries or disabilities.

While only the wrist trajectory is used in this study, we expect that the proposed approach can be adapted to motion trajectories for other body locations. For example, it would be possible to consider the motion trajectories for the elbow and shoulder to build a kinematics model of the whole arm. Moreover, our model does not take *hand orientation* and *wrist flexions* into account. Their possible effects on the prediction need to be explored in future work.

Although developed and evaluated within a 3D VR environment, our model is not fundamentally limited to predictions in VR applications. Since we use 3D motion trajectory, this work can be expanded to non-VR motion prediction such as hand movements in the real world or daily activities. It would be important to investigate if the generalized model changes for non-VR activities.

9 Conclusion

This paper contributed a novel user- and activity-independent *hybrid classical-regressive kinematics model* for continuous 3D hand trajectory prediction for ballistic movements in VR. Through a user study with 20 participants, we show our model performs comparably to personalized and specialized models for both structured and unstructured ballistic hand motions. Across all the users and activities, our model achieves a low Root Mean Square Error (RMSE) of 0.80 cm, 0.85 cm and 3.15 cm for future hand positions of 100 ms, 200 ms and 300 ms. Finally, we evaluate our model through cross-validation and a follow-up study with new participants and activities. To the best of our knowledge, this is the first attempt to develop a generalized hand motion prediction model across different users and activities for ballistic movements. Our prediction model can be used in VR to pre-render graphics, calculate complex physics, support real-time multi-modal feedback, and even recover short-term tracking errors. While this paper focuses on VR, we believe there are benefits in extending our work to other domains in the future.

[7] See https://www.nielsen.com/us/en/insights/report/2017/us-games-360-report-2017/ (Accessed on 2021-03-26).

Acknowledgments

This project was funded by the Australian Research Council Discovery Early Career Award (DECRA) - DE200100479. Dr. Withana is the recipient of a DECRA funded by the Australian Government. We thank Ziyu Amy Liu, Joji Tenges for their initial explorations and Prof. Fabio Ramos, Dr. Rafael Oliveira for assistance with the Optitrack system. Finally, we are grateful to all our user study participants and anonymous reviewers for their valuable feedback.

References

[1] A Deniz Aladagli, Erhan Ekmekcioglu, Dmitri Jarnikov, and Ahmet Kondoz. 2017. Predicting head trajectories in 360 virtual reality videos. In *2017 International Conference on 3D Immersion (IC3D)*. IEEE, 1–6.

[2] Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Lars Petersson, and Stephen Gould. 2020. A stochastic conditioning scheme for diverse human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5223–5232.

[3] Elena Arabadzhiyska, Okan Tarhan Tursun, Karol Myszkowski, Hans Peter Seidel, and Piotr Didyk. 2017. Saccade landing position prediction for gaze-contingent rendering. *ACM Transactions on Graphics* 36, 4 (2017). https://doi.org/10.1145/3072959.3073642

[4] Stanislav Aranovskiy, Rosane Ushirobira, Denis Efimov, and Géry Casiez. 2016. Modeling pointing tasks in mouse-based human-computer interactions. In *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 6595–6600.

[5] 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*. 1968–1979.

[6] Ronald Azuma and Gary Bishop. 1995. A frequency-domain analysis of head-motion prediction. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 401–408.

[7] Myroslav Bachynskyi and Jörg Müller. 2020. Dynamics of Aimed Mid-air Movements. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–12.

[8] Daniel Bullock and Stephen Grossberg. 1988. Neural dynamics of planned arm movements: emergent invariants and speed-accuracy properties during trajectory formation. *Psychological review* 95, 1 (1988), 49.

[9] Elie Cattan, Amélie Rochet-Capellan, Pascal Perrier, and François Bérard. 2015. Reducing latency with a continuous prediction: Effects on users' performance in direct-touch target acquisitions. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces*. 205–214.

[10] Yeonjoo Cha and Rohae Myung. 2013. Extended Fitts' law for 3D pointing tasks using 3D target arrangements. *International Journal of Industrial Ergonomics* 43, 4 (2013), 350–355.

[11] Yang Chen, Xingang Zhao, and Jianda Han. 2013. Hierarchical projection regression for online estimation of elbow joint angle using EMG signals. *Neural Computing and Applications* 23, 3 (2013), 1129–1138.

[12] Chris Christou. 2010. Virtual reality in education. In *Affective, interactive and cognitive methods for e-learning design: creating an optimal education experience*. IGI Global, 228–243.

[13] Aldrich Clarence, Jarrod Knibbe, Maxime Cordeil, and Michael Wybrow. 2021. Unscripted Retargeting: Reach Prediction for Haptic Retargeting in Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. IEEE, 150–159.

[14] Nachiket Deo and Mohan M Trivedi. 2018. Convolutional social pooling for vehicle trajectory prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 1468–1476.

[15] Massimiliano Di Luca and Arash Mahnan. 2019. Perceptual limits of visual-haptic simultaneity in virtual reality interactions. In *2019 IEEE World Haptics Conference (WHC)*. IEEE, 67–72.

[16] Mohammed S Elbamby, Cristina Perfecto, Mehdi Bennis, and Klaus Doppler. 2018. Toward low-latency and ultra-reliable virtual reality. *IEEE Network* 32, 2 (2018), 78–84.

[17] Hesham Elsayed, Mayra Donaji Barrera Machuca, Christian Schaarschmidt, Karola Marky, Florian Müller, Jan Riemann, Andrii Matviienko, Martin Schmitz, Martin Weigel, and Max Mühlhäuser. 2020. VRSketchPen: Unconstrained Haptic Assistance for Sketching in Virtual 3D Environments. In *26th ACM Symposium on Virtual Reality Software and Technology* (Virtual Event, Canada) *(VRST '20)*. Association for Computing Machinery, New York, NY, USA, Article 3, 11 pages. https://doi.org/10.1145/3385956.3418953

[18] Behnam Esfahbod. 2011. *SFU Motion Capture Database.* https://mocap.cs.sfu.ca/

[19] Tamar Flash and Neville Hogan. 1985. The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of neuroscience* 5, 7 (1985), 1688–1703.

[20] Laura Freina and Michela Ott. 2015. A literature review on immersive virtual reality in education: state of the art and perspectives. In *The international scientific conference elearning and software for education*, Vol. 1. 10–1007.

[21] Nirit Gavish, Teresa Gutiérrez, Sabine Webel, Jorge Rodríguez, Matteo Peveri, Uli Bockholt, and Franco Tecchia. 2015. Evaluating virtual reality and augmented reality training for industrial maintenance and assembly tasks. *Interactive Learning Environments* 23, 6 (2015), 778–798.

[22] Henry Griffith, Subir Biswas, and Oleg Komogortsev. 2018. Towards Reduced Latency in Saccade Landing Position Prediction Using Velocity Profile Methods. In *Proceedings of the Future Technologies Conference*. Springer, 79–91.

[23] Liang-Yan Gui, Yu-Xiong Wang, Xiaodan Liang, and José MF Moura. 2018. Adversarial geometry-aware human motion prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 786–803.

[24] Serhan Gül, Sebastian Bosse, Dimitri Podborski, Thomas Schierl, and Cornelius Hellge. 2020. Kalman Filter-based Head Motion Prediction for Cloud-based Mixed Reality. In *Proceedings of the 28th ACM International Conference on Multimedia*. 3632–3641.

[25] Serhan Gül, Dimitri Podborski, Thomas Buchholz, Thomas Schierl, and Cornelius Hellge. 2020. Low-latency cloud-based volumetric video streaming using head motion prediction. In *Proceedings of the 30th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*. 27–33.

[26] Markus Hahn, Lars Krüger, and Christian Wöhler. 2008. 3d action recognition and long-term prediction of human motion. In *International Conference on Computer Vision Systems*. Springer, 23–32.

[27] Rorik Henrikson, Tovi Grossman, Sean Trowbridge, Daniel Wigdor, and Hrvoje Benko. 2020. Head-Coupled Kinematic Template Matching: A Prediction Model for Ray Pointing in VR. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–14.

[28] Alejandro Hernandez, Jurgen Gall, and Francesc Moreno-Noguer. 2019. Human motion prediction via spatio-temporal inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7134–7143.

[29] Neville Hogan. 1984. An organizing principle for a class of voluntary movements. *Journal of Neuroscience* 4, 11 (1984), 2745–2754.

[30] Xueshi Hou, Jianzhong Zhang, Madhukar Budagavi, and Sujit Dey. 2019. Head and body motion prediction to enable mobile VR experiences with low latency. In *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 1–7.

[31] ByeoungDo Kim, Chang Mook Kang, Jaekyum Kim, Seung Hi Lee, Chung Choo Chung, and Jun Won Choi. 2017. Probabilistic vehicle trajectory prediction over occupancy grid map via recurrent neural network. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 399–404.

[32] Edward Lank, Yi-Chun Nikko Cheng, and Jaime Ruiz. 2007. Endpoint prediction using motion kinematics. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 637–646.

[33] Rynson WH Lau and Addison Chan. 2008. Motion prediction for online gaming. In *International Workshop on Motion in Games*. Springer, 104–114.

[34] Steve LaValle. 2013. The Latent Power of Prediction. https://developer.oculus.com/blog/the-latent-power-of-prediction/. Accessed: 2021-04-01.

[35] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. 2018. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5226–5234.

[36] Hongyi Liu and Lihui Wang. 2017. Human motion prediction for human-robot collaboration. *Journal of Manufacturing Systems* 44 (2017), 287–294.

[37] Lei Liu, Robert van Liere, Catharina Nieuwenhuizen, and Jean-Bernard Martens. 2009. Comparing aimed movements in the real world and in virtual reality. In *2009 IEEE Virtual Reality Conference*. IEEE, 219–222.

[38] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. 2019. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9489–9497.

[39] Julieta Martinez, Michael J Black, and Javier Romero. 2017. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2891–2900.

[40] Aythami Morales, Francisco M Costela, Ruben Tolosana, and Russell L Woods. 2018. Saccade Landing Point Prediction: A Novel Approach based on Recurrent Neural Networks. In *Proceedings of the 2018 International Conference on Machine Learning Technologies*. 1–5.

[41] Mathieu Nancel, Stanislav Aranovskiy, Rosane Ushirobira, Denis Efimov, Sebastien Poulmane, Nicolas Roussel, and Géry Casiez. 2018. Next-point prediction for direct touch using finite-time derivative estimation. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 793–807.

[42] Mathieu Nancel, Daniel Vogel, Bruno De Araujo, Ricardo Jota, and Géry Casiez. 2016. Next-point prediction metrics for perceived spatial errors. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. 271–285.

[43] Diederick C Niehorster, Li Li, and Markus Lappe. 2017. The accuracy and precision of position and orientation tracking in the HTC vive virtual reality system for scientific research. *i-Perception* 8, 3 (2017), 2041669517708205.

[44] Daniel Eger Passos and Bernhard Jung. 2020. Measuring the accuracy of inside-out tracking in XR devices using a high-precision robotic arm. In *International Conference on Human-Computer Interaction*. Springer, 19–26.

[45] Dario Pavllo, Christoph Feichtenhofer, Michael Auli, and David Grangier. 2019. Modeling human motion with quaternion-based neural networks. *International Journal of Computer Vision* (2019), 1–18.

[46] Dario Pavllo, David Grangier, and Michael Auli. 2018. Quaternet: A quaternion-based recurrent model for human motion. *arXiv preprint arXiv:1805.06485* (2018).

[47] Roshan Lalintha Peiris, Wei Peng, Zikun Chen, Liwei Chan, and Kouta Minamizawa. 2017. Thermovr: Exploring integrated thermal haptic feedback with head mounted displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 5452–5456.

[48] Kjetil Raaen, Ragnhild Eg, and Ivar Kjellmo. 2019. Playing with delay: An interactive VR demonstration. In *Proceedings of the 11th ACM Workshop on Immersive Mixed and Virtual Environment Systems*. 19–21.

[49] Nimesha Ranasinghe, Pravar Jain, Nguyen Thi Ngoc Tram, Koon Chuan Raymond Koh, David Tolley, Shienny Karwita, Lin Lien-Ya, Yan Liangkun, Kala Shamaiah, Chow Eason Wai Tung, et al. 2018. Season traveller: Multisensory narration for enhancing the virtual reality experience. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[50] Nimesha Ranasinghe, Thi Ngoc Tram Nguyen, Yan Liangkun, Lien-Ya Lin, David Tolley, and Ellen Yi-Luen Do. 2017. Vocktail: A virtual cocktail for pairing digital taste, smell, and color sensations. In *Proceedings of the 25th ACM international conference on Multimedia*. 1139–1147.

[51] Joël Randrianandrasana, Arnaud Chanonier, Hervé Deleau, Thomas Muller, Philippe Porral, Michaël Krajecki, and Laurent Lucas. 2018. Multi-user predictive rendering on remote multi-GPU clusters. In *2018 IEEE Fourth VR International Workshop on Collaborative Virtual Environments (3DCVE)*. IEEE, 1–4.

[52] Magnus JE Richardson and Tamar Flash. 2002. Comparing smooth arm movements with the two-thirds power law and the related segmented-control hypothesis. *Journal of neuroscience* 22, 18 (2002), 8201–8211.

[53] Miguel Fabián Romero Rondón, Lucile Sassatelli, Ramón Aparicio Pardo, and Frédéric Precioso. 2020. Track: a Multi-Modal Deep Architecture for Head Motion Prediction in 360° Videos. In *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2586–2590.

[54] Maria T Schultheis and Albert A Rizzo. 2001. The application of virtual reality technology in rehabilitation. *Rehabilitation psychology* 46, 3 (2001), 296.

[55] David Tolley, Thi Ngoc Tram Nguyen, Anthony Tang, Nimesha Ranasinghe, Kensaku Kawauchi, and Ching Chiuan Yen. 2019. Windywall: exploring creative wind simulations. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction*. 635–644.

[56] Etienne Van Wyk and Ruth De Villiers. 2009. Virtual reality training applications for the mining industry. In *Proceedings of the 6th international conference on computer graphics, virtual reality, visualisation and interaction in Africa*. 53–63.

[57] Tran Huy Vu, Archan Misra, Quentin Roy, Kenny Choo Tsu Wei, and Youngki Lee. 2018. Smartwatch-based Early Gesture Detection 8 Trajectory Tracking for Interactive Gesture-Driven Applications. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 1–27. https://doi.org/10.1145/3191771

[58] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. 2017. The pose knows: Video forecasting by generating pose futures. In *Proceedings of the IEEE international conference on computer vision*. 3332–3341.

[59] Min Wang, Xinyu Wu, Duxin Liu, Can Wang, Ting Zhang, and Pingan Wang. 2015. A human motion prediction algorithm for non-binding lower extremity exoskeleton. In *2015 IEEE International Conference on Information and Automation*. IEEE, 369–374.

[60] Zhen Wang, Hengsuo Xu, and Hongliang Yuan. 2020. Research on Design and Experience of Immersive Virtual Reality Psychological Relaxation Game Based on Image. In *IOP Conference Series: Materials Science and Engineering*, Vol. 740. IOP Publishing, 012118.

[61] Ning Xie, Gabrielle Ras, Marcel van Gerven, and Derek Doran. 2020. Explainable deep learning: A field guide for the uninitiated. *arXiv preprint arXiv:2004.14545* (2020).

[62] Michael Yates, Arpad Kelemen, and Cecilia Sik Lanyi. 2016. Virtual reality gaming in the rehabilitation of the upper extremities post-stroke. *Brain injury* 30, 7 (2016), 855–863.

[63] Shulin Zhao, Haibo Zhang, Sandeepa Bhuyan, Cyan Subhra Mishra, Ziyu Ying, Mahmut T Kandemir, Anand Sivasubramaniam, and Chita R Das. 2020. Déjà View: Spatio-Temporal Compute Reuse for 'Energy-Efficient 360° VR Video Streaming. In *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 241–253.