Using Multi-Level Precueing to Improve Performance in Path-Following Tasks in Virtual Reality

Jen-Shuo Liu, Carmine Elvezio, Barbara Tversky, and Steven Feiner

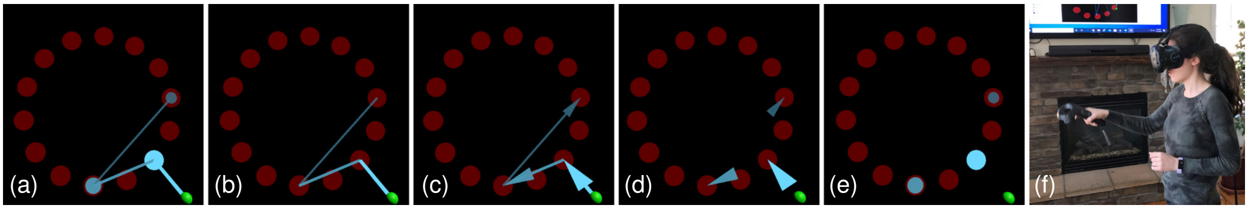

Fig. 1. Two-level precueing visualizations for a VR path-following task in ring of circular targets, depicted using combinations of circles (indicating places), arrows (indicating places and directions), and lines (indicating paths). The small green sphere shows the current location in the path. Each visualization includes a large bright cyan cue, followed by two successively smaller, dimmer, and more transparent precues. Visualizations, ordered by decreasing effectiveness in our study, use: (a) circles and lines, (b) lines, (c) arrows and lines, (d) arrows, and (e) circles. Note: Precues appear dimmer in Figures 1–4 viewed on monitors or on paper than in the study. The authors confirmed that the precues are clearly visible on VR headsets. (f) A remote study participant (used with permission). Each participant in the remote pilot study and formal study performed the experiment within their own home with their own SteamVR-compatible equipment.

Abstract— Work on VR and AR task interaction and visualization paradigms has typically focused on providing information about the current step (a cue) immediately before or during its performance. Some research has also shown benefits to simultaneously providing information about the next step (a precue). We explore whether it would be possible to improve efficiency by precueing information about multiple upcoming steps before completing the current step. To accomplish this, we developed a remote VR user study comparing task completion time and subjective metrics for different levels and styles of precueing in a path-following task. Our visualizations vary the precueing level (number of steps precued in advance) and style (whether the path to a target is communicated through a line to the target, and whether the place of a target is communicated through graphics at the target). Participants in our study performed best when given two to three precues for visualizations using lines to show the path to targets. However, performance degraded when four precues were used. On the other hand, participants performed best with only one precue for visualizations without lines, showing only the places of targets, and performance degraded when a second precue was given. In addition, participants performed better using visualizations with lines than ones without lines.

Index Terms—Virtual reality, path following, visual cues, task precueing, remote VR user study

1 Introduction

Many tasks that people perform alone or together entail completing a sequence of steps. Examples include navigating in an environment, assembling or repairing something, cooking, and playing games. For many tasks, the sequence of steps may not be known, or may be dependent on unexpected factors that emerge during action. Users could consult instructions for each step, though that slows the process considerably, or they could memorize step-by-step instructions, though that adds considerable load. If the instructions could instead be dynamically integrated with the task itself, and synchronized with their progress through the task, performance might be greatly improved.

In an attempt to realize this possibility, virtual reality (VR) and augmented reality (AR) have been applied in many situations to provide users with cues, often visual ones, intended to improve their performance. For example, these cues can be used to call attention to important items in the environment while driving [21, 31], or to help the user analyze information about tasks they must accomplish [26].

We use the term *cueing* to refer to providing information (*cues*) about the current step immediately prior to or during its performance. In contrast, we use the term *precueing* to refer to providing information (*precues*) in advance about future steps. Precueing has been used in a number of domains; for example, in video games, *Gran Turismo V* renders a long textured arrow on the road ahead, graphically precueing upcoming turns and ideal acceleration/deceleration opportunities [28], and *Beat Saber* precues multiple tasks beyond the current one with a series of labeled boxes approaching the player [2]. In *Eagle Flight*, one cue and one precue are used to indicate the user's upcoming path [38]. In everyday tasks such as cooking, multiple precues are commonly used; for example, as the user previews future steps in a recipe while performing the current step. If the user can parse and process the information about current and future tasks in parallel, it is possible to improve task performance (e.g., by decreasing completion time or improving accuracy).

There has been some work investigating the quantitative benefits of precueing in 2D and 3D user interfaces. Hertzum and Hornbæk [15] analyzed the gains provided by precueing the next action when performing tasks using a mouse or touchpad device. In a similar vein, Volmer et al. [41] investigated the benefits of providing a precue for the following action when performing a series of button-pressing tasks in a projector-based AR environment. Both of these studies used only a single precue for the immediately next action after the current cued

- *Jen-Shuo Liu is with Department of Computer Science, Columbia University. E-mail: jl5004@columbia.edu.*
- *Carmine Elvezio is with Department of Computer Science, Columbia University. E-mail: ce2236@columbia.edu.*
- *Barbara Tversky is with Department of Human Development, Teachers College, Columbia University. E-mail: bt2158@tc.columbia.edu.*
- *Steven Feiner is with Department of Computer Science, Columbia University. E-mail: feiner@cs.columbia.edu.*

Manuscript received 15 Mar. 2021; revised 11 June 2021; accepted 2 July 2021. Date of publication 27 Aug. 2021; date of current version 1 Oct. 2021. Digital Object Identifier no. 10.1109/TVCG.2021.3106476

action.

In this paper, we investigate the potential advantages and limitations of multiple levels of precueing, providing information about more than one step in advance. We designed a VR testbed and user study to explore two questions concerning the effectiveness of precueing sequences of actions: (a) How many precues for upcoming steps beyond the current step can people use effectively? (b) What visualization styles are most effective for communicating the cue and precues?

In our testbed and study, a graphical cue and precues guide the user in a manual path-following task in which a hand-held VR controller must be moved through a specified series of locations. The locations are circular targets arranged in a ring (Figure 1). Comprehending and using the cue and precues depend on several cognitive and perceptual processes: working memory [6, 7, 19, 25], perceptual discrimination [6, 7, 25], and visual tracking [12, 29]. Each of these is limited and their interactions are unknown. Based on this research and an earlier pilot study, our experiment studies 0–4 precues.

Building on previous work [37], we compare visualization styles that emphasize the *path* the hand follows to an object to be acted on and/or the *place* of that object. We examined five visualization styles for the cue and precues: a line to the target (indicating the path) that terminates at the target with an arrowhead (indicating the direction and place) or a circle (indicating the place); a line to the target by itself; and an arrowhead or circle alone at the target.

Thus, we contribute a VR user study that examines the effectiveness of different numbers of precues and different precue visualization styles on performance in a path-following task. Our analysis shows that:

- For visualizations with lines, participants performed best when given two to three precues, but performance degraded when a fourth precue was added.
- For visualizations without lines, participants performed best when given only one precue, and performance degraded when a second precue was added.
- Participants performed better with visualizations using lines and ranked them higher than visualizations without lines.

2 Related Work

Cueing visualizations have been used in a variety of AR and VR systems. One application emphasizes information users might have missed; for example, indicating potential roadside hazards to drivers [31]. A similar approach can be used to draw a user's attention to a specific part of an image [21]. In another application, cueing is used to guide the user to an object with which they need to interact [5]. Cueing is also commonly used in maintenance tasks [26, 33].

Cues and precues can be presented in many ways. Biocca et al. proposed the *attention funnel* [4], which guides the user to an object or place of interest using spatial cues. They found that the attention funnel could catch the user's attention more efficiently than highlighting and audio cueing, and could also decrease mental load. Bau and Mackay developed OctoPocus [1], a system that guides users to draw and memorize the members of a 2D gesture set. OctoPocus has feedback and feedforward mechanisms: the path that a user has already drawn is shown as a thin line to provide feedback, while the available continuations of a partially drawn gesture are presented with thick lines as feedforward. This feedforward mechanism can be considered as a precueing approach. Ellis et al. [9] explored the effects of latency on a path-following task. Although they did not explicitly address precueing, the visualization of the remaining path to trace all the way to the end could be considered an example. While precueing shows information about future tasks, predictive control techniques aim to aid users to preview future system or task status based on current operation. Predictive control techniques have been extensively discussed for flight/orbit maneuvering tasks and telerobotics [3, 13, 16, 27, 32].

Precueing is also used in a variety of rhythm games, either VR-based or mouse/keyboard-based. In *Beat Saber* [2], labeled cubes rapidly approaching a player in VR precue the upcoming actions the player needs to perform in time to a song. Similarly, in *osu!* [8], a player

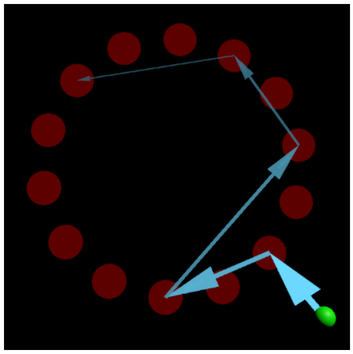

Fig. 2. Study testbed environment. A ring of 14 circular targets is presented to the participant, aligned with the *xy* (vertical) plane. A small green sphere represents the VR controller, instead of a full-size controller model, to avoid obscuring the targets, cue, and precues.

uses a mouse and keyboard to follow designated paths and hit targets, guided by precues, also in time to a song. In these games, cues and precues are not only visual. The player can use their familiarity with the song and its rhythm to predict upcoming actions. Further, players are required to act on each target at the right time, rather than as fast as possible.

According to Fitts's law [11], the time to move to a target is a function of the distance to the target and its area. MacKenzie and Buxton investigated Fitts's law for 2D tasks [22]. ISO9241-411 [18] provides requirements and an evaluation method based on Fitts's law, for the ergonomics of physical input devices. Kohli et al. proposed redirected touching [20], which overlaid a virtual training pattern in a virtual environment with a monitor in the real world that the trainee needed to touch. (The training pattern is similar to the one for the multi-directional pointing task in ISO9241-411.) At first, trainees performed worse than in the real environment, but improved with practice in the virtual environment. Their work focused on comparing performance in real and virtual environments and used a path with a regular pattern.

Hertzum and Hornbæk investigated the effect of precueing on 2D displays with mouse/touchpad input [15]. The scene consisted of a single precued target centered in a surrounding circle of targets, which were randomly cued. The user was asked to alternately move between a surrounding cued target and the center precued target. The center target was considered precued because users could predict that they always needed to travel to it every other step. The study found that users spent less time moving to the center precued target than to the randomly selected surrounding targets. However, the single precued target is always the center one. In contrast, our work evaluates randomly precued targets in VR for one or more future steps (i.e., multiple precues) for which the user must rely on the visualizations.

Volmer et al. investigated the effect of using one precue in button-pressing tasks in a projector-based AR environment through two experiments [41]: one with only the current target and next target labeled, and one with all buttons labeled. The scene used for both studies consisted of a physical hemisphere with 16 buttons located on it. The cue and precue were projected on the hemisphere to guide the participants to follow the paths. The paths were randomized so that the participants would not be able to predict the next target, unlike the central target in the work by Hertzum and Hornbæk [15]. It was shown that in both cases, connecting the current and the first upcoming destinations was better for improving the task performance than changing the color of the precue or making the precue blink. While their work investigated different precue designs when one precue for the next task was provided, we explore the benefit of using multiple precues for multiple upcoming tasks, the styles of multi-level precuing, and the interaction between these two factors and how it affects people's ability to understand precues.

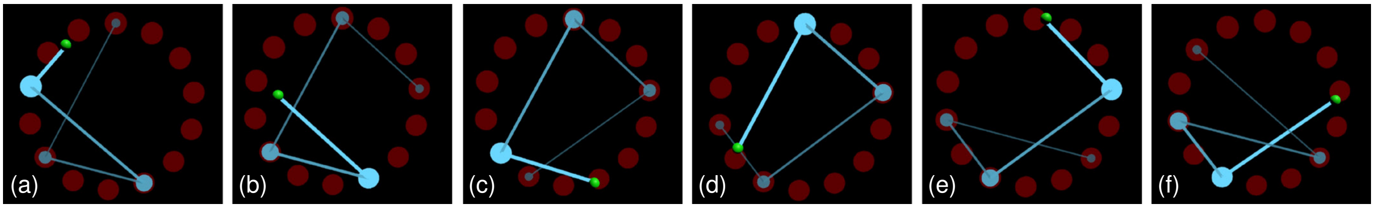

Fig. 3. Visualization (using CircleLine style) with one cue and three precues. (a–f) Six successive steps in the task.

3 Cueing and Precueing

In our path-following task, the user is asked to perform a series of movements (subtasks) between 14 circular targets. An example of the test scene is illustrated in Figure 2 and its design will be further discussed in Section 4.

3.1 Cue

The cue visualization shows the current subtask (segment). It always guides the user to move the VR controller to the current destination. As the user finishes the kth subtask, the cue visualization updates to guide the user to the next destination in the task sequence (the $k+1$st destination).

3.2 Precues

The precue visualizations provide information about succeeding steps while the current step is being performed. The precue visualizations can support showing many upcoming steps. For a condition with m precues, when the user is working on subtask k, the first precue visualization shows the destination of subtask $k+1$, whose origin is also the destination of subtask k. The 2nd, 3rd, ..., mth precue visualizations show subtasks $k+2$, $k+3$, ..., $k+m$, respectively.

Once the user completes the current subtask k, the first precue becomes the cue for destination $k+1$ and its visualization becomes brighter and completely opaque. The 1st, 2nd, 3rd, ..., mth precue visualizations now show subtasks $k+2$, $k+3$, ..., $k+m+1$. This continues until the user finishes the entire task, which includes 42 subtasks.

Figure 3 shows an example that uses a cue and three precues. Subfigures (a–f) show six successive steps in the task. The cue guides the user to the current destination, while the three precues show the information for the next three steps. Different sizes and colors are assigned to the visualizations. The cue visualization is the largest, brightest, and fully opaque, while each precue is successively smaller, dimmer, and more transparent. To eliminate one potential confound when using these visualizations, we removed lighting effects from the material used for the visualizations. When the user completes the current step, all precues enlarge, brighten, and become less transparent proportionally, with the first becoming the cue. Additionally, a new smaller, dimmer, and more transparent precue is added at the end of the chain. This continues until the user finishes the complete task.

3.3 Visualization Styles for Cues and Precues

To understand how different variations of precueing impact user task performance, we explored five different visualization styles, summarized in Figure 1, which shows a cue and two precues in each style. To go beyond simple arrows [14], we chose styles based on their varying ability to communicate place, path, and direction. Our styles are intentionally quite standard—they represent three of the four basic PowerPoint "arrowhead" styles: none, arrow, circle, diamond [24].

ArrowLine: The ArrowLine visualization (Figure 1c) connects the origin and destination of a specific subtask using a line. In addition to the line, an arrowhead is attached to the destination to emphasize it. The arrowhead for the cue visualization is designed to have the same size as a target, while the ones for the first to the fourth precues are 64%, 36%, 16%, and 4% of the target size, respectively. The lines used for the first to the fourth precues have a width that is equal to 80%, 60%, 40%, 20% of the width of the line used for the cue. The ArrowLine visualization provides *place*, *direction*, and *path* information for the subtask.

Arrow: The Arrow visualization (Figure 1d) shows only the arrowhead of the ArrowLine visualization. We designed this visualization to investigate the importance of the path information provided by the line in ArrowLine. The Arrow visualization provides *place* and *direction* information.

CircleLine: The CircleLine visualization (Figures 1a and 3) is similar to ArrowLine, except a circle is used to highlight the destination. Each circle has the same area as the corresponding arrowhead in ArrowLine and Arrow. The circle for the cue fully covers a target, while the ones for the first to fourth precues are 64%, 36%, 16%, and 4% of the target size, respectively. Unlike the arrowhead, the circle does not provide *direction* information. We designed this visualization style to see whether the *direction* information of ArrowLine offers additional performance benefits. The CircleLine visualization provides *place* and *path* information, and highlights the destination to the user.

Circle: The Circle visualization (Figure 1e) shows only the circle of the CircleLine visualization. It provides only the *place* information of the destination and does not have the *path* information that the line provides. Compared with Arrow, Circle does not communicate *direction*.

Line: The Line visualization (Figure 1b) shows only the line of ArrowLine or CircleLine. The Line visualization connects the origin and destination and provides *path* information. We designed this visualization to explore the benefit that might be offered by the circle in CircleLine (communicating *place*) and the arrowhead in ArrowLine (communicating *place* and *direction*).

4 Test Environment

We have implemented a VR testbed that supports a variety of cueing and precueing visualizations for a path-following task to help elucidate their potential benefits and limitations.

The test scene, shown in Figure 2, consists of 14 circular targets (each of which can serve as the origin or destination of a translation action), visualizations that cue or precue the origin and destination of an action, and a small green sphere that indicates the position of the handheld controller. The targets, each of radius 3 cm, are evenly distributed on a circular ring of 23 cm radius. The ring lies in a plane that is parallel to the xy-plane, and thus is parallel to the user's view plane. The ring radius was set to 23 cm during a early pilot study to ensure that the user's arm moves sufficiently far that it is unlikely that the user touches a target by accident, while also preventing the user from feeling fatigued after a small number of trials. The targets were sized to ensure they were visible and could accommodate imperfect tracking (as we cannot ensure sub-2 cm tracking accuracy in participants' home environments). A small 3D sphere, rather than a controller model, is used to decrease occlusion of the targets and cueing/precueing visualizations.

There are 42 subtasks in the sequence, where in each subtask the user moves their hand to the kth destination, which in our test scene is always one of 14 targets. The 42 movements are randomly generated, with the constraint that the entire path is connected. This means that the destination of step k will be the origin of step $k+1$, where k is an integer and $1 \leq k \leq 41$. The 42 movement actions form a closed loop, so we can pick a random start point in the path for a task, while making sure all tasks using that path traverse the same set of segments.

To avoid confounding effects related to a participant's anticipation of the completion of a trial, we present an entire set of additional actions,

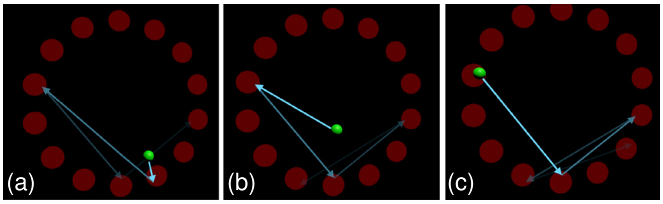

Fig. 4. Visualization used in the pilot study. (a–c) Three successive steps.

past the point of the last actual movement action to be completed, such that our precueing system will render additional visualizations past the end. Thus, unless a participant is counting the steps, which in piloting we found to not be the case, the participant will not anticipate the end of a trial.

Depending on the experimental condition, there will be one cue and zero or more precues in the scene that are used to show the information of the next *n* steps. Each visualization shows the destination of the step it represents. Each precueing visualization is aligned with the *xy*-plane.

The user interface and associated application interface were developed using Unity 2019.3.5f1 [39] and the open-source Mercury Messaging framework for Unity [10]. They were initially tested with HTC Vive Pro, Oculus Rift CV1, and Acer AH101-D8EY Windows Mixed Reality headsets and controllers, but were designed to work well with all SteamVR-compatible headsets and controllers [40].

5 User Study

We conducted a pilot study and a formal study. Due to the COVID-19 pandemic, both of our studies were conducted remotely (Figure 1f) after approval by our institutional review board. Each participant was paid $15 for their participation. The recruiting information was sent through our department email lists and posted on several VR forums. The consent forms were sent to, signed by, and returned by the participants electronically prior to beginning the study.

We coordinated with participants to ensure they had the right equipment and software installed before the start of the study. Since the COVID-19 pandemic prevented in-person studies, we made our design support any SteamVR-compatible headset and controller. Each participant was asked to use their own equipment, and to ensure that their computer met the minimum requirements necessary to run SteamVR [40]. In addition, participants were asked to ensure that their software was completely up to date. The studies were conducted over Zoom [44], allowing the study coordinators to communicate with the participants. Before each individual session, a study executable was sent to the participant. At the beginning of each session, the participant was first given the Ishihara Pseudo-Isochromatic Plate (PIP) test for color vision deficiencies.

5.1 Pilot Study

We used a pilot study to help us better understand people's ability to utilize precues and to help refine the design of the visualizations. We used only the ArrowLine visualization style in the pilot study, because we thought it was the most intuitive choice. We investigated the impact of the number of precues on task performance, using only the following numbers of precues: 0 (cue only), 1, 3, 5, 7, 9, and 11.

Note that, as shown in Figure 4 (a–c), the size of the arrowheads and the width of the lines in the pilot study are different from those discussed in Section 3.3 and used for the formal study. Arrowhead size and line width in the pilot study are identical for the cue and all precues and the arrowheads are smaller than the targets.

We recruited 14 participants (7 female), 19–49 years old (average 30), that had no knowledge of our project for the pilot study. We later found that data for two of the participants were corrupted. These two participants are excluded in the following discussion. A total of five HTC VIVEs, three Oculus Quests with Quest Link cables, three HP Windows Mixed Reality headsets, and one Valve Index were used by the participants. At the end of the pilot study, the participants filled out a questionnaire in which they ranked the visualizations in order of preference.

We then examined the performance (completion time) to see how many precues would lead to the best result. The participants performed better with the help of one precue than no precue, but we found no significant improvement when increasing the number of precues beyond the first precue. On the other hand, two of the authors performed a separate self-pilot study in which the improvement stopped at three precues. We suspected that difference was caused by the authors being more familiar with the visualizations. Therefore, in the formal user study, discussed below, we tested up to four precues for all visualization styles. Three regular pilot-study participants reported in the questionnaire that they had difficulty distinguishing the precues during the study. To address this, we performed several self-pilot studies to explore modified visualizations and decided to add size differences and increase brightness differences to help participants differentiate precue order for the number of precues used in the formal study.

5.2 Hypotheses

In the formal user study, we focused on investigating the impact of the *style* of the visualizations on the performance and the best number of precues for each visualization style. We considered the five visualization styles shown in Figure 1 and discussed in Section 3.3. For each style, we tested a cue and 0–4 precues. Based on our initial design goals, and observations and results from the pilot study, we formulated the following hypotheses:

H1. *For each visualization style, users will perform better (lower completion time for a task overall and per subtask) when a precue is shown versus when only a cue is shown.* Previous work [15, 41] has shown that one precue helps improve task performance. The pilot study confirmed this for ArrowLine, but we wanted to check this for more visualization styles, especially when some of them do not provide path (or connectivity) information.

H2. *When the number of precues is fixed, users will perform better (lower completion time per subtask) when lines are used in the visualization style than when lines are not used.* The lines provide both path and place information, while the arrows and circles indicate only the places of the destinations.

H3. *Users will be able to make use of more precues with visualizations using lines than with visualizations that do not use lines (i.e., the best performance will happen with more precues for styles that use lines than for those that do not use lines).* We formulated this hypothesis since we suspected that users would perform better with styles that use lines because they can make use of more precues in those cases.

5.3 Methods

5.3.1 Participants and Equipment

Seventeen participants (9 female), 19–62 years old (average 27.1) joined the study from their homes and used one HTC VIVE Pro, two Oculus Quests with Quest Link cables, one Oculus rift S, one Oculus CV1, and eight Windows Mixed Reality headsets (one HP, two Samsung Odyssey, five Acer), one Valve index, and three Vive Cosmos Elites with Valve Index controllers. None of these participants were in the pilot study and all passed the PIP test. The participants were recruited in two phases: ten in the first phase and the other seven in the second phase. We later found that data for one participant recruited in the first phase were corrupted due to low system frame rate, which caused this participant to spend more time to finish the tasks. The participant was therefore excluded in the following analysis and discussion. We compared the results from the two groups of participants recruited in different phases and found the trends in the data were similar. Therefore we combined the data from both groups.

5.3.2 Design

We expected the difficulty of a subtask to depend on the absolute positions of its origin and destination targets, not just the segment length. The difficulty of a subtask also depends on its subsequent subtasks, since they provide visual context. Therefore, rather than designing different sequences that have the same total length, we created a set of

Table 1. Percentage of segments labeled as outliers in each condition.

	Cue	Precue1	Precue2	Precue3	Precue4
CircleLine	3.79	4.76	4.84	5.65	6.18
Line	3.65	4.54	5.06	4.91	6.18
ArrowLine	2.68	4.17	4.02	4.69	5.36
Arrow	3.65	5.73	4.32	4.09	5.80
Circle	4.54	4.76	2.16	2.60	4.32

three 42-subtask sequences (tasks) that are guaranteed to visit every target on the ring. We then randomly picked a different starting point for each combination of sequence, block, and participant. We made sure that each sequence did not contain any subsequence in which the $k-1$st destination is the same as the $k+1$st destination (i.e., no backtracking movements). One of the three sequences was used in the untimed practice trials. The other two were used in the timed trials. Of the two sequences used in the timed trials, one was designed to be easier (having fewer crossovers) than the other.

Trials were blocked by condition (visualization type and precue count). Each trial contains a 42-subtask sequence (task). Each block consisted of one untimed practice trial and two timed trials. There were 25 unique conditions (ArrowLine, Arrow, CircleLine, Circle, and Line, crossed with 0–4 precues for each style), each used in one block. The order in which the blocks appeared was shuffled to ensure each participant experienced a different order of conditions. The numbers of blocks and of trials per block were decided based on the feedback from the pilot study, in which the participants reported their frustration on the length of the study.

5.3.3 Procedure

In the session, the participant was first given the Ishihara Pseudo-Isochromatic Plate (PIP) test for color vision deficiencies. The participant was then briefly introduced to the flow of the experiment. They were asked to run the Unity executable and share their screen, so the study coordinator could monitor the status of the study. The shared screen was recorded through Zoom, so the experimenters could check after the study for potential issues. After starting the executable, the participant was first asked to adjust the height of the ring of targets to reduce fatigue during the experiment. After adjusting the height, the participant was asked to proceed to the system "free mode," which functions as a sandbox in which random conditions are picked and the study coordinator can instruct the participant on the format of the study, the nature of the visualizations, and answer any questions the participant may have. After finishing the free mode, the participant proceeded to the formal trial blocks.

During the study, we recorded the trial, task, and subtask completion time, 3D tracking data, and a video of the interaction. The 3D tracking data were later used to help analyze the outlier segments (Section 5.5). After finishing all blocks, the participant was asked to upload their test data to an online directory on our institution's specialized version of Google Drive (with adjustments to enhance security), and to fill out a questionnaire, which included an unweighted NASA TLX survey, a ranking of user preference for the visualizations, and demographic information. The whole process took about one and half hours for a typical participant to complete.

5.4 Results

We sought to understand if there was a relationship between the visualization style and user performance, measured in completion time. Note that each task could not be completed without completing each subtask (verified by detecting collision of the controller sphere with the relatively small 3 cm-radius target), so we focus on time rather than accuracy in terms of assessing user performance. We determined target size based on the authors' self-piloting.

On study completion, we processed completion time results generated automatically by our system before analyzing them. We identified outliers in the data before analysis. We examined the data and the videos and found that outliers appear because participants moved their arms to an incorrect target, or went to the correct target but failed to collide with it successfully on the first try. We used Tukey's outlier filter [36] to label outliers. The "outside fence" for each condition was computed separately, since we expected the conditions would have significant effect on completion time. Segments that took more than $Q3+1.5\text{IQR}$ or less than $Q1-1.5\text{IQR}$ were labeled as outliers, but in fact no segment was finished within time that was less than $Q1-1.5\text{IQR}$. The percentage of segments labeled as outliers under each condition is shown in Table 1.

The average subtask completion time after outlier removal under each condition is shown in Figure 5. Figure 5(a) reveals that the standard deviations are relatively large. This is because segments with different length are used in the experiment and individual participant performance differs. To better show between-subject differences, we plotted individual participant performance by visualization type and by number of precues in Figure 6. It can be seen that for 14 out of 16 participants, CircleLine yields the best result among the five visualization styles, while there is no consistent best number of precues when the five visualization styles are combined. Close examination of individual performance over number of precues reveals that performance seems to drop with four precues for five participants and even for three for six of them. The performance of two is best with four precues. Not surprisingly, different participants seem to have different asymptotes.

We evaluated the hypotheses for significance with $\alpha = .05$. We fit a linear mixed-effects model to our data using the MATLAB Statistics and Machine Learning Toolbox [23]. In the model we use, subtask completion time is the observation, visualization type and number of precues are the fixed-effect variables that have an interaction term between them, and segment (defined as the vector from the origin to the destination in a step in the path) and user ID are the random-effect variables. Unlike the fixed-effect variables, there is no interaction term between random-effect variables. These terms are picked based on a comparison between the current model and other alternative models using a likelihood ratio test (also with Akaike and Bayesian information criteria). The p-values for the fixed-effect terms and the interaction terms between them are $< .0001$, showing that the effects of visualization type and number of precues are significant, as is the interaction between these terms.

We also found that adding device type as another fixed-effect variable would result in a model with a better log-likelihood result, but would cause high p-values for the device terms (three of the six terms are $> .2$). We suspect this is because the effects caused by the device type and user are coupled, as each type of device was used by a specific subset of users. Therefore, we decided not to use device type as a fixed-effect variable. For the full details of the linear mixed-effects model, please see the supplementary materials.

We refer to the number of precues in a condition as Cue (cue only) or Precue$n$, where n is the number of precues. To determine, for each visualization type, if there is a significant difference between Cue and Precue1, we conducted five post-hoc Wilcoxon signed-rank tests. The p-values of these tests are all $< .0001$, showing that the difference is significant and therefore **H1** is supported. This is also consistent with Hertzum and Hornbæk [15] and Volmer et al. [41], though the task domain is different. We computed the effect sizes of the Wilcoxon signed-rank tests using the method proposed by Rosenthal [30]. For the pairs for CircleLine, Line, and ArrowLine, $r > 0.77$, showing the effect sizes are large. For Arrow and Circle, $r = 0.63$ and 0.48, respectively, showing the effect sizes are medium. It is also worth noting that except for Circle, users performed better with the help of 1–4 precues than with no precue.

For **H2**, we compared two pairs: ArrowLine vs. Arrow and CircleLine vs. Circle. For each pair, we conducted Wilcoxon signed-rank tests separately for each number of precues. All yielded $p < .0001$, noting that the participants performed better in Circle-Cue than in CircleLine-Cue. This can be seen in Figure 5(a). For other pairs, visualizations with lines yielded better result. Therefore, **H2** is mostly supported. We then computed the effect sizes for these pairs. For the comparisons between ArrowLine and Arrow, $0.30 > r > 0.11$ for Cue and Precue1, and $0.63 > r > 0.53$ for Precue2–4. For the comparisons

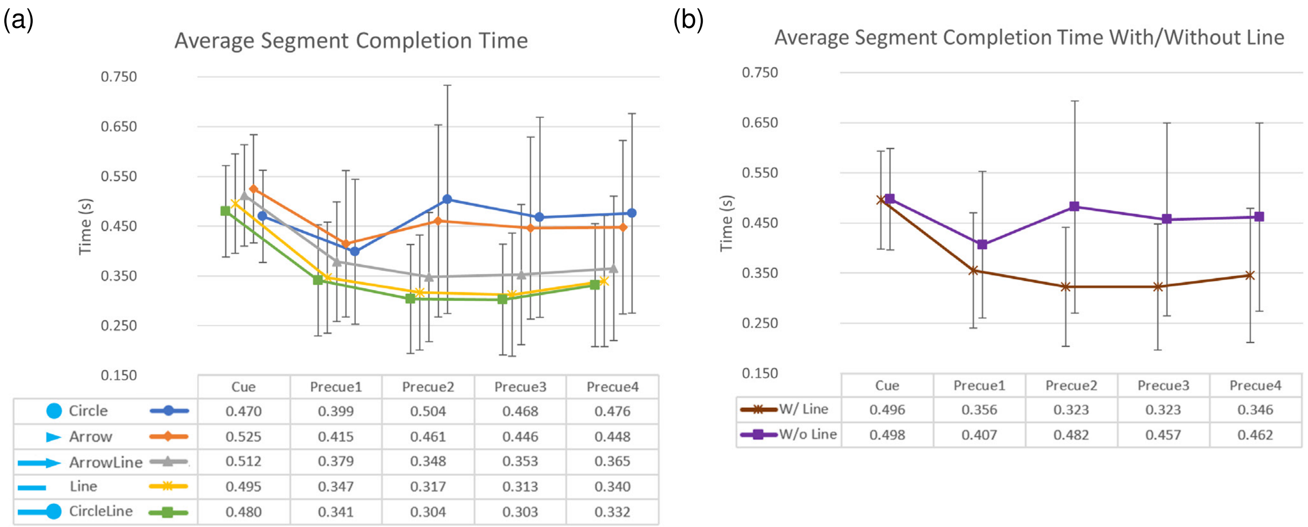

Fig. 5. Average segment completion time in seconds. When only the cue visualization was shown, the performance for different visualization styles did not differ much. When one or more precue visualizations were shown, ArrowLine, CircleLine, and Line performed significantly better than Arrow and Circle.

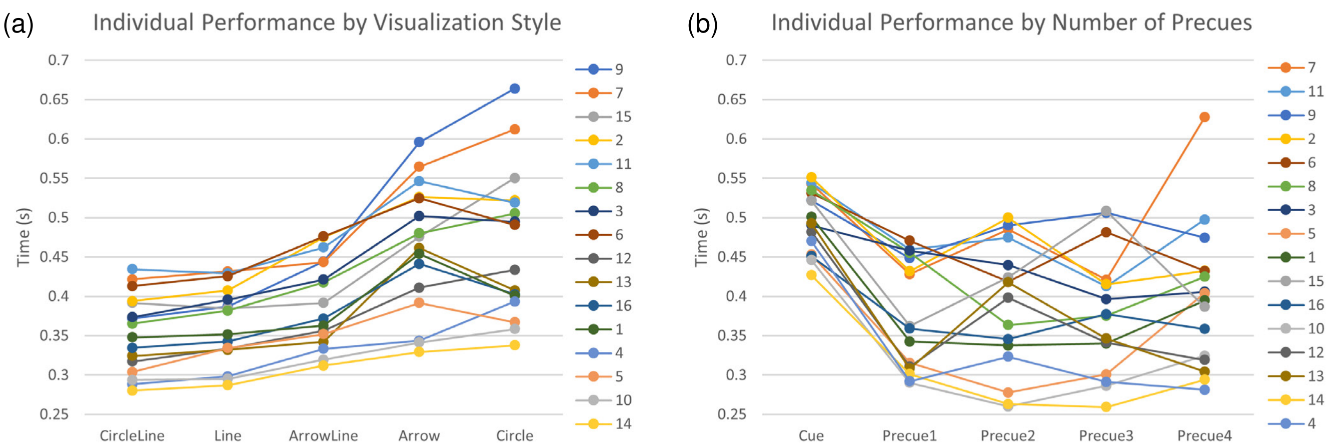

Fig. 6. Individual participant performance (seconds per segment) plotted by (a) visualization style and (b) number of precues.

between CircleLine and Circle, $0.41 > r > 0.13$ for Cue and Precue1, and $0.81 > r > 0.70$ for Precue2–4. To help visualize the performance difference between styles with and without lines, we plot Figure 5(b). It can be seen that while in Cue the difference is quite small, the participants performed much better when lines were used in the visualizations and when at least one precue was used.

For **H3**, we first observed in Figure 5(a) that participants performed best with two to three precues in CircleLine, Line, and ArrowLine, but performed best with only one precue in Arrow and Circle. When a fourth precue was given in CircleLine, Line, and ArrowLine, or when a second precue was given in Arrow and Circle, the performance degraded. To determine if there is support for **H3**, we ran several Wilcoxon signed-rank tests: Precue2 and Precue3 versus Cue, Precue1, and Precue4 for CircleLine, Line, and ArrowLine, and Precue1 versus Cue, and Precue2–4 for Arrow and Circle. All Wilcoxon signed-rank tests yielded $p < .0009$. Therefore, **H3** is supported. Relative to Cue, $0.86 > r > 0.79$ (strong) for styles with lines, and $0.63 > r > 0.47$ (medium) for styles without lines. Comparing Precue2 and Precue3 to Precue1 and Precue4, $0.43 > r > 0.09$ for all visualization styles, showing the effect sizes are relatively small. That the best number of precues varies depending on visualization style reflects that there are strong interactions between the fixed-effect terms in the linear mixed-effects model we ran to fit the data.

To validate the hypotheses, we ran a total of 39 pairwise Wilcoxon signed-rank tests (5 for **H1**, 10 for **H2**, and 24 for **H3**) and reused two of the five used for **H1** for **H3**. Since the p-values that showed significance were all $< .0009$, all of the tests would still survive a Bonferroni-corrected α of $.0012(.05/39)$. For the detailed p-values and effect sizes for the Wilcoxon signed-rank tests, please see the supplementary materials.

5.4.1 User Feedback

At the end of the session, each participant was given a questionnaire that included demographic information (including which equipment they used), a set of custom unweighted NASA TLX surveys, and questions on user preference.

We wanted to avoid stopping the study after every block to get feedback. Thus we chose to give only a single questionnaire at the end of the study. Images of the conditions were provided in the questionnaire to help the participants answer the questions.

5.4.2 NASA TLX

In the questionnaire, we first let the participants answer a separate unweighted NASA TLX for each visualization style. The results are shown in Figure 7. Generally speaking, visualization styles with lines got better results for each metric than visualization styles without lines.

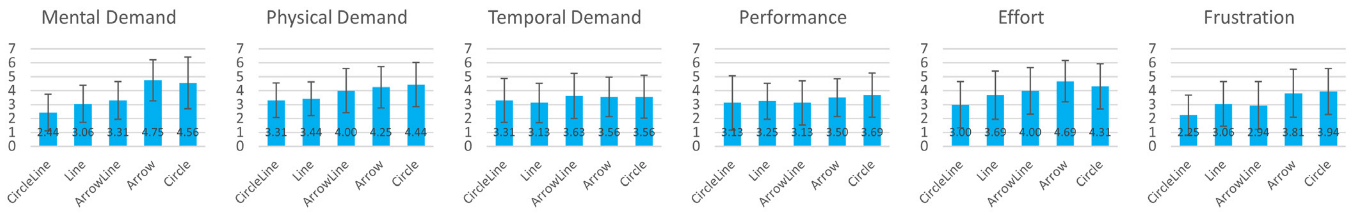

Fig. 7. NASA TLX results. Generally speaking, CircleLine was the least mentally and physically demanding visualization style. It also required less effort and caused less frustration. (Note: Like the other scales, the right-hand side of Performance is the worst.)

We conducted Friedman tests on the NASA TLX results, yielding $p_{MentalDemand} < .0001$, $p_{PhysicalDemand} = .0016$, $p_{TemporalDemand} = .2190$, $p_{Performance} = .6036$, $p_{Effort} = .0013$, and $p_{Frustration} = .0014$. This shows that the differences in mental demand, physical demand, effort, frustration are significant. However, temporal demand and performance are not significant. While the participants' rating on these two metrics are not significant, the objective results in Figure 5 show that they performed better in terms of completion time for visualizations with lines.

5.4.3 User Preference

The participants then ranked the visualization styles by their preferences. The results are shown in Figure 8. Most participants preferred CircleLine over the other styles and, generally speaking, Circle was the least preferred visualization. We also ran a Friedman test for the preference result. showing the result is significant ($p_{Preference} < .0001$).

The participants were also asked how many precues they thought were best for their performance. Among the 16 participants, five answered one precue, five chose two precues, and the remaining six participants picked three precues. We ran a linear regression on the preferred number of precues the users picked against their age, gender, and average subtask completion time. The *p*-values for each coefficient in the resulting model revealed no significant differences.

5.5 Error Analysis

In Section 5.4, we labeled outlier segments using Tukey's method [36]. To have a better understanding of which kind of errors occurred in these segments, we plotted each participant's controller movements and manually labeled the error types. We opted to manually label because errors could be a mixture of multiple error types. We found that there were several common types of errors made by participants (Figure 9):

Tracking Issues: In rare cases, tracking would fail and cause the users to spend additional time on some segments. This is typically indicated by a discontinuity in the path plot, as shown in Figure 9(a).

Wrong Target: In some instances, a participant moved toward the incorrect target, and then, with or without hitting it, realized their mistake and turned to the correct one, as shown in Figure 9(b). This often happened in visualization styles without lines and when multiple precues were presented.

Missed Target: In these cases, the participant moved toward the correct destination, but failed to collide with it. In some cases, the participant corrected this immediately after the failed attempt, but sometimes they first incorrectly moved toward the next destination and then turned back after realizing their error, as shown in Figure 9(c). This occurred for a number of reasons, including: missing the target by failing to hit it on the *xy*-plane, or reaching the correct *x*- and *y*-positions, but missing the capsule collider of the target due to an error in the *z*-axis.

Other: All other errors that could not be categorized into the previous three types were identified as "Other." In a few segments, participants spent more time than the Tukey "outside fence" in finishing the segment, but no oddity in the path geometry was observed, as shown in Figure 9(d). In other cases, the error appeared to involve a combination of the other three types.

Table 2. Error rates as percentage of each type of error out of all segments in each condition.

	CircleLine	Line	ArrowLine	Arrow	Circle
Tracking Issue	0.03	0.01	0.01	0.03	0.06
Wrong Target	0.80	0.82	0.80	1.59	1.62
Missed Target	2.81	2.93	2.65	2.10	1.24
Other	1.40	1.09	0.71	1.01	0.76

Table 2 lists the percentages of each type of error that occurred. It can be seen that the study participants made more errors by going to wrong targets under Arrow and Circle than under CircleLine, Line, and ArrowLine. To determine whether the differences were significant, we ran several chi-square tests on wrong target errors for: Circle and Arrow versus CircleLine, Line, and ArrowLine. All chi-square tests yielded $p < .001$, showing that people did make more Wrong Target errors under Circle and Arrow. This can be explained by the observation that the line connecting the controller and current destination easily prevented the participants from going to a wrong target.

For Missed Target errors, we first observed that CircleLine, Line, and ArrowLine have roughly the same percentage of these types of errors, while Arrow was slightly lower and Circle had the lowest. We wanted to know if different terminators lead to different percentages of Missed Target errors; therefore, we performed two sets of comparisons. In the first set, we ran three chi-square tests for the following pairs: CircleLine–Line, CircleLine–ArrowLine, and ArrowLine–Line. The p values for these three pairs were .680, .560, and .320, respectively, meaning the error percentage differences between these three pairs were all insignificant. For the second set, we ran one chi-square test between Arrow and Circle. The test yielded $p < .001$, meaning study participants made more Missed Target errors in Arrow than in Circle.

5.6 Discussion

5.6.1 Comparison between Visualization Styles

As mentioned in the results for **H2**, participants performed better with precue visualizations that used lines to indicate path than with ones that did not show path. Participants reported that the lines allowed them to more readily follow the path than by mentally computing it from the relative brightness, size, and transparency of the cue and precues. That computation relies on users discriminating the cue target from the successive precue targets under time pressure, as well as keeping the sequence in mind. Human capacities for discrimination, working memory, [6,7,25] and attention required to track moving objects [12,29] interact with one another and are limited. Since the cue and precues are presented simultaneously, participants need to discriminate them, calculate their order, and track the path to the upcoming target. Because the lines extend in the target space and contain some place information, they are easier to discriminate, to determine order and to track to the target, than the placeholders of Arrow and Circle. Therefore, the participants were able to use more precues in visualizations with lines than in visualizations without lines. When too many additional precues were provided, the participants were overloaded and their performance degraded.

As for the comparison between CircleLine, Line, and ArrowLine,

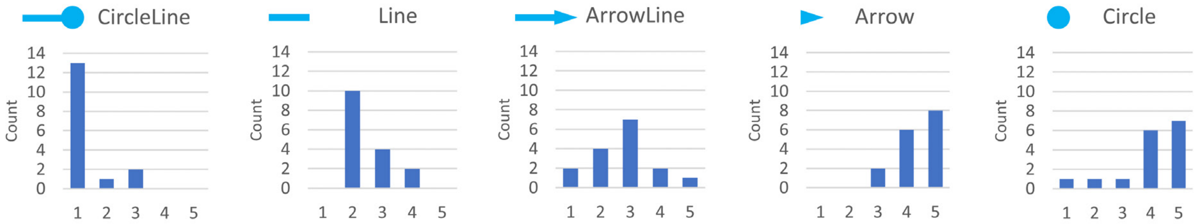

Fig. 8. Histograms of user preferences for visualization styles, from 1 (highest = best) to 5 (lowest). Most users preferred CircleLine over the other styles.

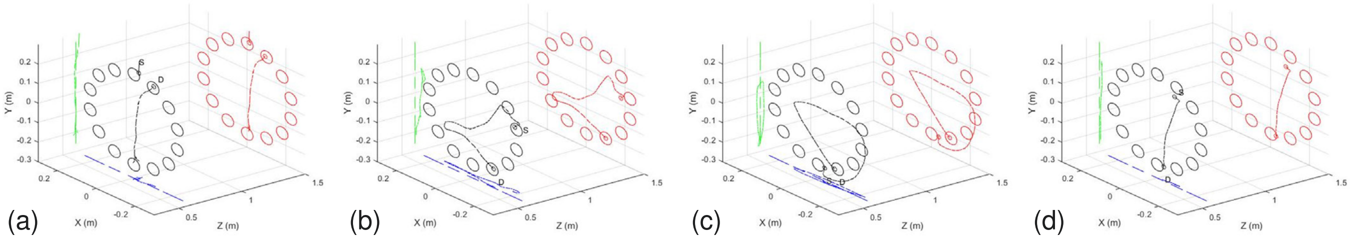

Fig. 9. Examples of common error types in outlier segments. Each subfigure shows a representative outlier segment from a single user (a) Tracking issue. (b) Wrong target. (c) Missed target. (d) Other. Black circles are targets. Lines indicate user movement. Dots labeled with S and D are the source and destination of the segment, respectively. Red, green, and blue plots are projections of the original plot onto the *xy*-, *yz*-, and *xz*- planes.

we believe CircleLine is the most helpful, as it highlights the target destination for the user. ArrowLine, however, was confusing to some of the participants. Participant 8 pointed out that "*Arrowheads were too long, especially when they overlapped two dots*." Other participants complained that when the controller approached the destination, the arrowhead partially obscured the destination target and the sphere for the controller (even though the tip of the arrow always visibly ended at the center of the target). We believe this was the reason that ArrowLine led to worse performance than CircleLine. Similarly, we believe that Line was between CircleLine and ArrowLine in performance because it gave partial direction information and did not obscure the target.

For the visualization styles without lines, Arrow was superior to Circle in most cases, most likely because it gave some direction information relevant to path as well as pointed to the target destination. That is, Arrow showed the place as well as suggesting the path. The visualization that gave the best performance in most cases was CircleLine, as it unambiguously showed both path and place. Interestingly, as we discussed in Section 5.5, arrowheads did not cause more Missed Target errors in ArrowLine than in CircleLine and in Line, though the participants did make more Missed Target errors in Arrow than in Circle.

Preattentive processing [34, 35, 42, 43] might account for a single precue performing best for Circle and Arrow. However, discriminating among multiple precues is likely to require attention. In contrast, while the same considerations hold for precues that include a line, their performance improved with additional precues. Together, this suggests that discriminating the next target is likely a larger factor in performance than preattentive processing, and that the lines help in this process.

A similar comparison between precues with and without lines can be found in Volmer et al. [41], though unlike our exploration of multi-level precueing, they investigated only a single precue, projected (using spatial AR) near one of a set of physical buttons mounted on part of a hemisphere. Their precue visualization styles included a line from cue to target, comparable to our Line visualization style, and a colored target, comparable to our Circle visualization style. (Their other visualization styles used blinking icons and small icons local to the cue pointing in the exact or general direction to the target.) In terms of both completion time and Wrong Target error rate, line was better than color for Volmer et al., much as Line was better than Circle for us, supporting the robustness of this result for single precues.

5.6.2 Individual Participant Performance

In Figure 6, we showed each participant's performance under each visualization type and number of precues. In Figure 6(a), we first average each participant's performance by visualization type. It can be seen that all participants performed the best in CircleLine, followed by Line and ArrowLine. While Arrow and Circle had the worst results for all participants, for some people Circle was worse than Arrow, and for others, Arrow was worse than Circle. Figure 6(b) shows the average of each participant's performance by number of precues. This time, there is no common trend for all participants.

5.6.3 Visualization Styles and Error Rates

In Section 5.5, we investigated the rate of each type of error for each visualization style. The fact that visualizations without lines have more Wrong Target errors suggests that the lines do help specify the target. On the other hand, the Missed Target rate suggests a speed–accuracy tradeoff. When given visualizations with lines, the participants performed faster, but have more Missed Target errors. They may target the correct destination, but their aim was off.

5.6.4 Comparison between Performance of Experimenters and Participants

As described in Section 5.1, we found that two of the experimenters, highly practiced in this sort of task, performed better (faster) and were able to make use of more precues than less-practiced participants in our pilot study. After running the formal study, we again compared the participants' performance with that of the experimenters. The performance of the expert experimenters, though faster, exhibited the same pattern as that of the study participants, who were all novices. Both the expert experimenters and the novice participants could make best use of the same number of precues: two to three for visualizations with lines, and one for visualizations without lines. This adds support to the claims. This also suggests that adding size differences after the pilot study helped the participants distinguish the precues and make use of more of them. The experimenters were able to use more than one precue in the pilot study because they were more familiar with the task and visualizations and were able to use the more subtle brightness differences to distinguish the precues.

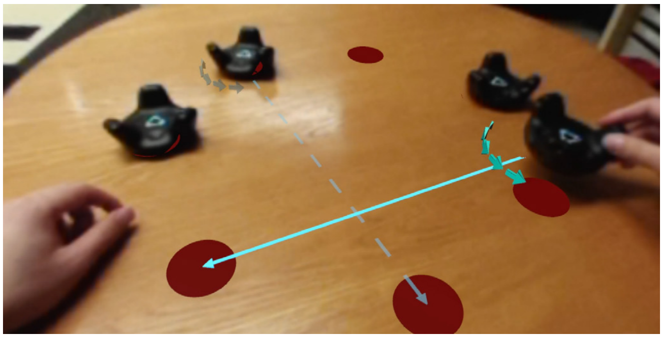

Fig. 10. Precueing in augmented reality. In this version of our testbed, not used in this paper, we used different arrow visualization as cues, as well as rotation-guiding rings to aid the user in performing rotations.

6 Future Work

Building on the results of the study and our discussion above, we have identified several research questions to explore in future work.

6.1 Precueing in Augmented Reality for Position- and Orientation-Tracking

Although our testbed system was studied in a VR environment for position-tracked tasks, it was first implemented for AR for tasks also requiring rotation (Figure 10). We decided to switch to VR for the studies reported on in this paper, since we were unable to run experiments in person because of the COVID-19 pandemic, and the difficulty of getting a large number of Vive Trackers to remote study participants, who would also need to have compatible AR headsets and tracking base stations. The AR mode was implemented using the video–see-through AR mode of the Vive Pro, using Vive SRWorks SDK [17]. Moving forward, we want to take the lessons learned from the pilot and formal studies, and apply them to AR, potentially addressing occlusion and more complex 6DoF tasks, performing a user study with that system. The results might differ as the task becomes more complex, and thus a redesign of the visualizations might be needed. However, we believe that our current study is an important first step and is applicable to real task domains, such as equipment control panels.

6.2 Parallel Tasks

The work reported on in this paper focuses on a serial task. There will be no more than one "current" step shown at the same time, a simplification of real world tasks. In fact, parallel task performance is required in many real world tasks, including simultaneous actions while playing musical instruments or video games. However, performing and understanding parallel tasks can increase mental load for users. Thus, improper precueing may hurt task performance by increasing time. It would be valuable to explore methods for designing precueing visualizations that can improve efficiency and accuracy in more complicated tasks in which selection and manipulation might occur simultaneously.

6.3 Tasks with Different Time and Accuracy Constraints

In our study, participants were asked to finish the tasks as fast as they could, decreasing the time available to process cues and precues. Depending on the application, this assumption might not hold. For example, in many maintenance tasks, workers need to value accuracy over speed, increasing the time available for processing cues and precues. In contrast, in the rhythm games mentioned above [2, 8], players need to finish tasks on beat at specific times. It is worth investigating whether cues and precues will be processed differently under these different time constraints.

7 Conclusions

We explored how multi-level precueing could be used to provide information about multiple steps after the current step on which a user is working. After consulting previous work and performing a pilot study, we selected, implemented, and compared five visualization styles for path-following tasks with a cue and 0–4 precues. We verified that for all five visualization styles tested, task completion time improved through the use of precues. Further, users performed best when given two to three precues for visualization styles using lines, as opposed to only one precue for visualization styles without lines. When using visualizations with lines, performance begins to degrade when a fourth precue is provided. Similarly, when using visualizations without lines, performance degrades when a second precue is provided. In addition, when the same number of precues are shown, users perform far better for visualization styles with lines indicating the path, than for visualization styles without lines. We believe that our results could also be applicable to 2D desktop environments, though a study using appropriate 2D technology would be needed to validate that.

Acknowledgments

This research was funded in part by National Science Foundation Grants CMMI-2037101, IIS-1514429, and IIS-1513841.

We thank Casey Bradshaw, Rishabh Dudeja, and Ian Kinsella for their assistance developing the linear mixed-effects model, and the participants of our studies.

References

[1] O. Bau and W. E. Mackay. OctoPocus: A dynamic guide for learning gesture-based command sets. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pp. 37–46. ACM, New York, NY, USA, 2008. doi: 10.1145/1449715.1449724

[2] Beat Games. *Beat Saber*. https://beatsaber.com/.

[3] A. K. Bejczy, S. Venema, and W. S. Kim. Role of computer graphics in space telerobotics: preview and predictive displays. In R. J. P. de Figueiredo and W. E. Stoney, eds., *Cooperative Intelligent Robotics in Space*, vol. 1387, pp. 365 – 377. International Society for Optics and Photonics, SPIE, 1991. doi: 10.1117/12.25440

[4] F. Biocca, A. Tang, C. Owen, and F. Xiao. Attention funnel: Omnidirectional 3D cursor for mobile augmented reality platforms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pp. 1115–1122. ACM, New York, NY, USA, 2006. doi: 10.1145/1124772.1124939

[5] L. Bonanni, C.-H. Lee, and T. Selker. Attention-based design of augmented reality interfaces. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pp. 1228–1231. ACM, New York, NY, USA, 2005. doi: 10.1145/1056808.1056883

[6] N. Cowan. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and brain sciences*, 24(1):87–114, 2001.

[7] N. Cowan, C. Morey, and Z. Chen. The legend of the magical number seven. *Tall tales about the brain: Things we think we know about the mind, but ain't so, ed. S. Della Sala*, pp. 45–59, 2007.

[8] Dean Herbert. *osu!* https://osu.ppy.sh/home.

[9] S. R. Ellis, F. Breant, B. Manges, R. Jacoby, and B. D. Adelstein. Factors influencing operator interaction with virtual objects viewed via head-mounted see-through displays: Viewing conditions and rendering latency. In *Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality*, pp. 138–145. IEEE, 1997.

[10] C. Elvezio, M. Sukan, and S. Feiner. Mercury: A messaging framework for modular UI components. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pp. 588:1–588:12. ACM, New York, NY, USA, 2018. doi: 10.1145/3173574.3174162

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

[12] S. Franconeri, S. Jonathan, and J. Scimeca. Tracking multiple objects is limited only by object spacing, not by speed, time, or capacity. *Psychological science*, 21(7):920–925, 2010.

[13] A. J. Grunwald and S. R. Ellis. Visual display aid for orbital maneuvering—design considerations. *Journal of Guidance, Control, and Dynamics*, 16(1):139–144, 1993. doi: 10.2514/3.11438

[14] J. Heiser and B. Tversky. Arrows in comprehending and producing mechanical diagrams. *Cognitive science*, 30(3):581–592, 2006.

[15] M. Hertzum and K. Hornbæk. The effect of target precuing on pointing with mouse and touchpad. *International Journal of Human–Computer Interaction*, 29(5):338–350, 2013. doi: 10.1080/10447318.2012.711704
[16] R. A. Hess and Y. C. Jung. An application of generalized predictive control to rotorcraft terrain-following flight. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):955–962, 1989.
[17] HTC, Inc. *Vive SRWorks SDK*, May 2019. https://developer.vive.com/resources/knowledgebase/intro-vive-srworks-sdk/.
[18] ISO/TS 9241 Ergonomics of human-system interaction—Part 411: Evaluation methods for the design of physical input devices. Standard, International Organization for Standardization, Geneva, CH, May 2012.
[19] T. Kleinsorge and N. Apitzsch. Task preparation based on precues versus memory: Precues lead to superior performance with two tasks but not with four tasks. *Journal of Cognitive Psychology*, 24(2):140–156, 2012.
[20] L. Kohli, M. C. Whitton, and F. P. Brooks. Redirected touching: Training and adaptation in warped virtual spaces. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 79–86. IEEE, 2013.
[21] W. Lu, B. H. Duh, and S. Feiner. Subtle cueing for visual search in augmented reality. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 161–166, Nov 2012. doi: 10.1109/ISMAR.2012.6402553
[22] I. S. MacKenzie and W. Buxton. Extending Fitts' law to two-dimensional tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '92, pp. 219–226. ACM, New York, NY, USA, 1992. doi: 10.1145/142750.142794
[23] The MathWorks, Inc. *Matlab Statistics and Machine Learning Toolbox*. https://www.mathworks.com/help/stats/index.html.
[24] Microsoft Corporation. *Draw or delete a line or connector*. https://support.microsoft.com/en-us/topic/draw-or-delete-a-line-or-connector-f304ef73-9514-450b-9bb9-28c6057020f2.
[25] G. A. Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.
[26] O. Oda, C. Elvezio, M. Sukan, S. Feiner, and B. Tversky. Virtual replicas for remote assistance in virtual and augmented reality. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software 38; Technology*, UIST '15, pp. 405–415. ACM, New York, NY, USA, 2015. doi: 10.1145/2807442.2807497
[27] E. A. Palmer, S. J. Jago, D. L. Baty, and S. L. O'Connor. Perception of horizontal aircraft separation on a cockpit display of traffic information. *Human Factors*, 22(5):605–620, 1980. doi: 10.1177/001872088002200509
[28] Polyphony Digital Inc. *Gran Turismo V*, November 2010. https://www.gran-turismo.com/us/products/gtsport/.
[29] Z. W. Pylyshyn and R. W. Storm. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial vision*, 3(3):179–197, 1988.
[30] R. Rosenthal. Parametric measures of effect size. *The handbook of research synthesis*, 621(2):231–244, 1994.
[31] M. L. Rusch, M. C. Schall, P. Gavin, J. D. Lee, J. D. Dawson, S. Vecera, and M. Rizzo. Directing driver attention with augmented reality cues. *Transportation Research Part F: Traffic Psychology and Behaviour*, 16:127–137, 2013. doi: 10.1016/j.trf.2012.08.007
[32] T. B. Sheridan. Three models of preview control. *IEEE Transactions on Human Factors in Electronics*, HFE-7(2):91–102, 1966.
[33] A. Tang, C. Owen, F. Biocca, and W. Mou. *Performance Evaluation of Augmented Reality for Directed Assembly*, pp. 311–331. Springer London, London, 2004. doi: 10.1007/978-1-4471-3873-0_16
[34] A. Treisman. Preattentive processing in vision. *Computer Vision, Graphics, and Image Processing*, 31(2):156–177, 1985. doi: 10.1016/S0734-189X(85)80004-9
[35] A. Treisman and J. Souther. Search asymmetry: a diagnostic for preattentive processing of separable features. *Journal of Experimental Psychology: General*, 114(3):285, 1985.
[36] J. W. Tukey. *Exploratory data analysis*, vol. 2. Reading, Mass., 1977.
[37] B. Tversky, M. Agrawala, J. Heiser, P. Lee, P. Hanrahan, D. Phan, C. Stolte, and M.-P. Daniel. Cognitive design principles for automated generation of visualizations. *Applied spatial cognition: From research to cognitive technology*, pp. 53–75, 2006.
[38] Ubisoft Entertainment SA. *Eagle Flight*, October 2016. https://www.ubisoft.com/en-us/game/eagle-flight/.
[39] Unity. *Unity*. https://unity.com/.
[40] Valve Corporation. *SteamVR*. https://store.steampowered.com/app/250820/SteamVR/.
[41] B. Volmer, J. Baumeister, S. Von Itzstein, I. Bornkessel-Schlesewsky, M. Schlesewsky, M. Billinghurst, and B. H. Thomas. A comparison of predictive spatial augmented reality cues for procedural tasks. *IEEE Transactions on Visualization and Computer Graphics*, 24(11):2846–2856, 2018. doi: 10.1109/TVCG.2018.2868587
[42] J. M. Wolfe. Forty years after feature integration theory: An introduction to the special issue in honor of the contributions of anne treisman. *Attention, Perception, & Psychophysics*, 82(1):1–6, 2020.
[43] J. M. Wolfe. Visual search: How do we find what we are looking for? *Annual review of vision science*, 6:539–562, 2020.
[44] Zoom Video Communications Inc. *Zoom*. https://zoom.us/.