

Interaction Techniques for Dynamic Virtual Environments

Roland Schröder-Kroll Kristopher Blom Steffi Beckhaus

University of Hamburg
interactive media / virtual environments
Vogt-Kölln-Str. 30, D-22527 Hamburg
+49 (0)40 428 83-2425, imve@informatik.uni-hamburg.de

Abstract: There is an increasing interest in creating environments containing components that change over time and are yet interactive. As interaction research has focused only on static environments, it is now necessary to start gaining an understanding of how to interact in such environments. In this paper, we present initial work on interaction with Dynamic Virtual Environments (DVEs). As it is typically the precursor to all other interactions, this work focuses on the selection of moving objects. Three techniques to specifically support dynamic object selection have been developed and tested in an immersive projection system. Two of these methods are shown to be very well suited for the selection of dynamic content while remaining usable for static selection.

Keywords: Interaction, Selection, Dynamic Virtual Environments

1 Introduction

Virtual Environments (VEs) have been the subject of scientific research for a long time. An important aspect of that research has been on techniques to interact with the VEs. However, most VEs, especially the ones created for research purposes, are static in nature. In many, the only moving entities are the users themselves. Presenting dynamic content, that is content that changes over time, adds an important dimension to convey information. It can make environments more interesting and realistic. It can provide insight into information that is not perceivable with lower dimensional display. Interactive exploration of content is generally more effective than the consumption of predefined presentations, as the participant becomes part of the process and feels a part of the environment. For these reasons and many more, the presentation of Dynamic Virtual Environments (DVEs) and the ability to interact with them are important.

However, when looking at the Virtual Reality (VR) community: interaction research in VR has been developed and tested almost exclusively with static environments and the areas of VR that rely on dynamic content, such as those in scientific simulations like physics or weather simulations, often end up having little if any interactivity. In contrast to the state of VR, video games are both highly dynamic and interactive. However, there is little formal research on video game interaction techniques. Also, the vast majority of the games use traditional input and output devices, i.e. mouse, keyboard, and monoscopic displays, that do not support 3D interaction very well. Game

designers also spend a lot of time perfecting specific interaction for their context. We feel that one of the major reasons that interaction with dynamics is not more widespread is that interacting with dynamic content is simply more difficult than with static content. In addition to the spatial precision required for interaction with static objects, temporal precision is required for interaction with dynamic objects.

This paper introduces initial work we are performing in an area we are certain will grow in importance in the coming years. We present an initial formal investigation into the area of interaction with dynamic VE content. As selection is the precursor to most other interactions, this work will focus on the selection of moving objects. We highlight the issues involved in selection of dynamic content and how those issues affect the usability of classic techniques. To support the user in selection of dynamic content, three new techniques are developed and presented in this work. Initial results of an evaluation of these techniques are also presented.

In the section that follows, related work is presented. Section 3 discusses the issues surrounding selection in the space of Dynamic Virtual Environments. A discussion of the applicability of existing selection methods fills out the section. The development of three new methods is presented in Section 4. Section 5 presents results of a preliminary evaluation, comparing the usability of four methods for selection of a moving object. Conclusions from the results of this work are drawn in Section 6.

2 Related Work

Interaction with moving objects is not a new activity, but not something well researched, particularly in a Virtual Reality context. This section presents the most relevant research on the topic, spanning a number of areas. Different external research areas that deal with dynamics are briefly mentioned first. Then, the most relevant research areas related to selection are presented. The first of the areas discussed are studies that deal with interaction with dynamic objects in VEs. The second area of research presented is focused on using VEs as experimental platforms for studies in other areas. Finally, related Virtual Reality interaction work is introduced.

Several research areas investigate movement and interaction in the real world. Motion analysis has been done in Medicine, particularly Sports Medicine, as well as in Mechanics. Motor skills and their control, such as hand-eye-coordination, have been researched in Neurology, Cognitive Psychology, and Robotics. Some related experiments from Neurology and Psychology are described below. Extensive research on target acquisition (selection) has purportedly been conducted on behalf of the military, but is not publicly available. Real-world games, especially sports and field games, have a highly dynamic character. The challenge of many of the games lies precisely in interaction with dynamic objects. The specific interactions of those games have been studied. Many video games are similarly characterized by interaction with dynamics. Even given this spectrum of areas interested in interaction with dynamics, the authors are not aware of any directly relevant general treatments of interaction with dynamic content.

The most relevant VR work is the “Silk Cursor” technique from Zhai, et al. [ZBM94]. The Silk Cursor technique enabled the “grabbing” of a moving fish by placing a semi-transparent box around the fish. The technique was performed in 3D using 6 degree of freedom (6DOF) tracking. The space of interaction was within arms reach, and the experiment was performed on a monitor. Such a form of “selection” is naturally limited in usability, due to having to encompass the object. Several related works from the area of telemanipulation are of interest. The focus of these studies were pursuit tracking of moving objects, i.e. keeping the object selected [ZM93, MSR89]. This was performed by overlaying a controlled virtual object onto the moving object, both with 6DOFs. This task placed the emphasis on continually pursuing the object as it moved rather than the acquisition of the target.

The other relevant set of research focuses on using VEs as a platform for research in other areas, like Psychology and Neuroscience. Several experiments on “hitting moving targets” have been performed [BMSB03, BSB05]. A simulated spider running across a table was to be hit with a handheld device or finger. The influences of target properties (e.g. speed, velocity, and dimension) on the task performance were examined. [RW99] conducted catching experiments in a virtual environment to develop a computational model that describe how humans time catching a ball. The advantages and disadvantages of virtual environments as a tool for studying interceptive action are discussed by [ZM03]. Their experiments required participants to intercept flying virtual balls with their foreheads or to judge where the balls would pass them by. [DPB04] compared catching movements in real and virtual environments. The movement of suspended balls swinging past the participant were compared to similar virtual ball trajectories that were displayed stereoscopically in a Virtual Reality system.

In the immersive Virtual Reality area, research has been limited to interaction with static objects. The suitability of the classic selection techniques of VR for the selection of dynamic objects is discussed in Section 3. The sole work to explicitly deal with time in selection is [Ste06]. He discusses the temporal aspects of the selection process. In his treatment, the target objects are static, while the user’s hand moves, for example to refine a selection.

3 Selection Techniques for Dynamic Virtual Environments

Selection (also called target acquisition) is a precursor task for most other interactions with VEs, required to inform the computer of the object of interaction. It is also a task with its own set of challenges and the focus of a relatively large set of research. The biggest difficulty of selection comes from achieving the spatial precision required to select objects in a 3D landscape. Small, distant, and partially occluded objects are challenges for selection. Moving from selection of static objects to dynamically changing objects, additional difficulties are introduced to the process. Section 3.1 briefly introduces the characteristics of environments that contain dynamic content, Dynamic Virtual Environments. The challenges of selecting dynamics and a discussion of the effectiveness of standard techniques in this domain are discussed in Section 3.2.

3.1 Dynamic Virtual Environments

Dynamic Virtual Environments (DVEs) are those environments that contain content that changes over time. This could be concrete, visible, or tangible objects or even content like fog, heat, waves, or forces that are only indirectly perceivable. Change can influence a variety of properties of an entity. These properties include the movement, i.e. translation and rotation of an object, as well as, surface properties, like color, visibility, and friction, or physical properties, like weight, inertia, and temperature. Furthermore, dynamics can also encompass other modalities, like sound or smell, abstract states, or the user's view. Combining, breaking apart, creating, and deleting objects also constitute dynamic changes to the environment. The characteristic of the changes over time could be predetermined, be random, be reactive to user input, or follow more complex functions such as the behavior of an autonomous entity. All of these things demonstrate the breadth of the design space of dynamics in VEs.

3.2 Selection of Dynamic Content

This work investigates interaction with such DVEs. For this initial investigation, the selection of an appropriate interaction and an object of interaction is critical. Perhaps the most likely of the possible dynamics to interact with is a moving object. It is representative of the predominant issues of interaction in dynamic environments, yet is more straight forward than other options. It is also the most related case to the bulk of the existing research, which focuses on the selection and spatial manipulation of objects in the VE. In this work, we will explicitly restrict our investigation to the selection of moving objects. Selection is chosen for the interaction focus, as it is prerequisite for the manipulation tasks common to other research.

This sub-section discusses the issues surrounding selection of dynamic content. The factors that influence the user's ability to select dynamics in an environment are discussed first. Existing selection methods and their presumed effectiveness in this area are then discussed.

3.2.1 Issues of Dynamic Selection

The task of targeting an object is a fundamental part of selection. The addition of dynamics to objects introduces the difficulty of having to achieve temporal accuracy in this task. In other words, the user has to not only indicate the object in space, but also has to manage to do this with a target that is changing its spatial position over time. For example, kicking a ball lying still on the ground is simple when compared to kicking a ball flying through the air. Luckily, the advantages of interacting in a mediated space can be leveraged to design an interaction method that supports the user in such a task.

Moreover, selection in DVEs suffers more strongly from issues that already affect static environments. Within limits, the precision issue of object selection can be compensated by users, e.g. by

adapting their position or taking more time to complete a task. There is always a speed-accuracy-tradeoff [Fit54]. With the added temporal precision requirement, these adaptations are less applicable. Object occlusion issues are compounded, since the objects movement may cause occlusion to change often. Additionally, if the object changes trajectories while occluded, it may be difficult for the user to even find the object again. The occlusion factor is further compounded by the need to follow the object constantly in order to detect changes in the trajectory; users tend to exclusively focus on the object to the exclusion of the rest of the environment. This means the user may not be aware of occlusion before it happens. Another factor of moving objects is that the size of the object may change significantly over time due to either its motion (in depth) or its orientation. In extreme cases, a rotating object that is wide, but thin, may virtually disappear as the thin side is oriented to the user. This requires the user to time the selection in an entirely different manner and is particularly susceptible to input lag issues.

Hand jitter and input data noise may potentially impact performance. This is even more true for dynamic selection tasks than static selection. A selection specific issue is the sudden change of selected targets in ambiguous situations. When two targets are located close together, the noise may cause the selection to jump across objects. We will refer to this problem as “selection flicker.” In the case of dynamics such problems are more difficult to cope with, largely due to the occlusion problems discussed above. Additionally, specialized supporting techniques may be more susceptible to such selection flicker.

3.2.2 Applicability of established Techniques

A number of existing selections techniques can be considered for use. A possible approach to handle dynamic situations is to pause the flow of time. This approach is fairly common in the simulation community. It allows the use of established interaction techniques. However, this approach is not always feasible, e.g. in multi-user environments. Some established techniques are at least partly suitable for the selection of dynamic content. For example, the Virtual Hand technique uses a direct mapping of the user’s hand to its virtual representation. This is intuitive and suitable for selection of dynamic objects within arm’s reach. Users can directly transfer a lot of their real-world experience in grabbing a moving object (e.g. catching a ball) to virtual environments. However, for mid- and long-range interaction, extensions like the Go-Go technique [PBWI96] are needed. They introduce additional learning effort that we wanted to avoid in initial testing and suffer from similar issues as Raypicking in respect to our problem space.

Raypicking or Raycasting uses a ray (more often a line segment of a shorter length) extending out from a hand held device, the hand itself, or the direction the user is looking. To reduce ambiguity, typically a pre-selection is indicated by some form of feedback. This pre-selection is performed by touching an object (i.e. intersecting its geometry) with the ray. The actual selection typically starts a manipulation and is indicated by pressing a button on an input device. Raypicking is the standard selection technique for wand-like devices. It is an effective technique, because it is straightforward and intuitive. However, it requires a lot of precision to select objects that appear small. It is also

very susceptible to positional and especially orientational jitter; Both of which are normal in the human hand.

The Cone Selection technique (also called Flashlight or Spotlight technique, [LG94]) improves on the raypicking technique, using a cone extending from the user's wand as a selection volume. Selecting an object by overlaying it with the cone requires less precision than a single ray. However, it also provides a more blunt tool, making fine-grained tasks and tasks involving distant objects more difficult. Some extensions attempt to improve this by adding methods to refine the selection [dHKP05]. A somewhat common approach is to refine the initial selection by moving the cone to exclude and de-select unwanted objects (see [Ste06]). Such an approach implicitly uses the element of time. However, keeping an object in focus, while trying to exclude others, becomes increasingly difficult for moving objects.

4 Proposed Techniques

We propose four techniques for selection of dynamics for further exploration and testing. Raypicking is proposed as a reference technique, both for comparison purposes and because it is so widely used. Three techniques are proposed and developed here. These techniques attempt to improve the user's ability, by addressing different issues related to the selection of dynamics, while remaining usable in general contexts.

We developed new methods based on three approaches we felt were most promising to explore: introducing constraints or providing assistance to alleviate the issues of spatial and temporal precision, incorporating the element of time in the method, and incorporating the dynamic nature into the technique. These approaches have been explored with the developed techniques. The Snapping Pointer uses constraints. The Time Cone actively incorporates the element of time. The Trajectory Based selection uses the movement of an object to support selection.

4.1 Design Requirements and Considerations

The requirements for selection techniques in general used in our development were: a) Errors should be minimal, the result of a selection should be the object desired by the user. b) The method should be easy to learn for novice users. c) Feedback to the currently selected object should be provided. d) The method should be transparent, users should be able to concentrate on the task instead of thinking about the interface.

In addition, a number of requirements that are specific to selection of dynamic objects were developed: a) Methods need to enable the selection of a single moving object and b) the selection of a single object out of a group of moving objects. More specifically, the method should provide the ability to c) discern objects moving very differently, as well as, d) to discern objects moving in a very similar manner (for instance objects in a swarm or on a conveyor belt). The methods should

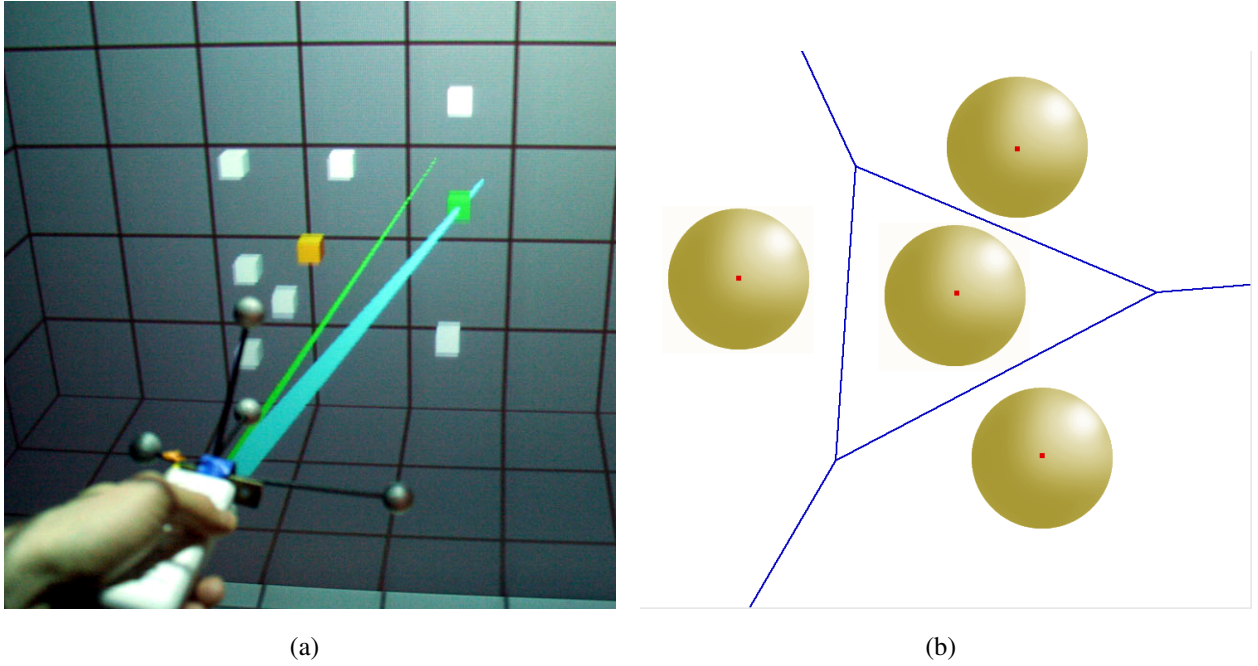


Figure 1: (a) The Snapping Pointer selection technique in use. The thinner (green) ray represents the true direction; the thicker (blue) ray snaps to the nearest object, which is highlighted (green). (b) The enlarged selection space of objects, represented by a Voronoi tessellation.

furthermore e) remain usable for standard static selection purposes. These requirements, together with the identified issues of selection in DVEs have motivated the technique development.

4.2 Snapping Pointer

Based on the classic Raypicking technique, the Snapping Pointer assists the user by pre-selecting the object that is closest to the selection ray. The process is visualized by the ray snapping to the selected object, which is additionally highlighted. This reduces the needed spatial precision of intersecting an object to select it. The user then presses a button to select the object. Figure 1 (a) shows the method in use. The enlarged selection space of objects is illustrated in Figure 1 (b).

The approach is comparable to the "snap to grid" features offered by some 2D graphics tools. There, graphical objects snap to predefined positions on a two-dimensional grid in order to precisely align them to other components. Generalized, the methods work by introducing constraints that restrict the possible choices to some predefined special cases. For the task of selecting one object out of a set of objects in a 3D environment, reducing the freedom of possible selections to this set of objects might make sense. The Snapping Pointer approach does this by making only the objects selectable. In our implementation, the nearest object is then the selected object.

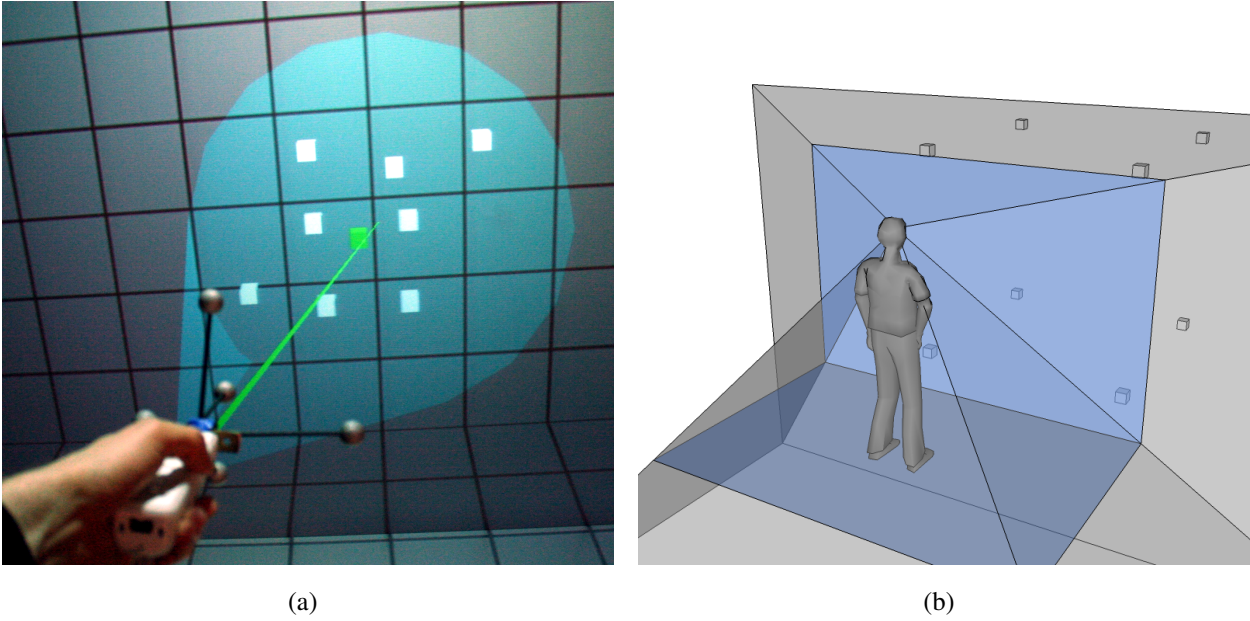


Figure 2: (a) Time Cone selection technique, from the user's perspective. The (green) cube, closest to the cone's center axis is highlighted for selection. (b) Viewing volume and test setup illustration of the L-shape IPT. The cubes represent the size and starting positions of target objects.

4.3 Time Cone

The second technique is a time-based extension of the standard cone selection method described previously. This method is partially inspired by the IntenSelect Method by [dHKP05], which takes a similar approach for selection in complex static situations. The Time Cone extends the original concept by actively including the time dimension. The reciprocal value of an object's distance from the cone's center axis is used as a weight for selection. This value is integrated over the time an object spends inside the selection cone. The object with the highest weight is pre-selected and indicated by highlight feedback. This allows the user to only roughly keep the object in the cone and is especially robust against "selection flicker," i.e. losing the focus to another object crossing the target's path. However, this is achieved at the cost of a delay to switch to subsequently indicated objects. Figure 2 (a) illustrates the use of the method.

4.4 Trajectory Based Selection

The third method attempts to help the user in selecting dynamics by making the selection criteria also be based on dynamics. It is also partly based on the observation of people pointing at moving objects, like birds or cars, following their movement with the hand. Compared to the "snap to position" approach of the Snapping Pointer technique, the trajectory based selection is a "snap to movement" approach. It transfers the tracking of an object from position to direction and velocity. The goal is to assist the user in selecting an object by following or imitating its movement. Using a pointing device, the user matches an object's trajectory by performing a similar motion. The most direct way to do this is to follow the object by continuously pointing at it, but there is no

need to directly point at the target. Technically, an object's trajectory is compared to that of an invisible object, attached to the ray. Feedback is provided by pre-selection highlighting and a second snapping ray like in the Snapping Pointer described above.

5 Evaluation

In this section we report on a preliminary study of the usability of the proposed techniques for selection of dynamics. This preliminary study was performed with a group of 7 users. The testing environment was an immersive projection technique display, called the L-shape, composed of two 2 * 3 m projection screens, illustrated in Figure 2 (b). The area was tracked using an optical tracking system and 6 DOF pointing device was used, as shown in Figure 2 (a). The study was a within-subjects test design, where all users tested all methods and task sets.

Since this study was the first to study the selection of dynamic objects, Raypicking was used as a reference method. Tests were performed in four blocks, one for each technique. The testing started with a practice phase, where the user got acquainted with the devices, selecting static objects with standard Raypicking. Techniques were then presented to users in random order. Between each task (technique) set, qualitative feedback was obtained with questionnaires. This also served to give the user a break from the immersive environment.

Two basic categories of tasks were considered: selecting a single target object and selecting a target out of a set of objects, the "cluster test". The order of these tasks was randomized for each technique. Test objects were cubes with 10 cm side lengths, moving at speeds of either 0.5 m/s or 1 m/s. The objects moved parallel to one of the orthogonal axes. Different starting positions and travel directions were included to test for effect of distance or direction preferences. Starting positions and movement were chosen, such that interaction happened in a medium range working volume, within a 2 - 4 m distance from the user. The objects were always visible in the display. For the cluster tasks, the target object was surrounded by 8 identical "obstructing" objects. The obstructing objects varied from the target in speed and direction by 10%, based on reproducible pseudo-random values; In this way each participant saw the same "random" movements for each task. The tests were carried out in a virtual room of 10 m side length. Figure 2 (b) illustrates starting (and stopping) positions of test objects as they appeared with respect to the physical space. Between each task, the user had to select an object in the center of the space; this guaranteed a standard starting point and allowed the user to control the start of the next task. A total of 92 selections were performed per technique.

An initial analysis of the recorded data provides indications of the performance of the techniques. The accuracy of the methods is measured by the percentage of successful selections from all trials combined. The arithmetic mean time to complete a task serves as the speed metric. The results of user testing are shown in Figure 3.

Users achieved only a 73% accuracy and took roughly 1.3 seconds mean time with the reference method, Raypicking. This is a better performance than expected; However, such a low percentage

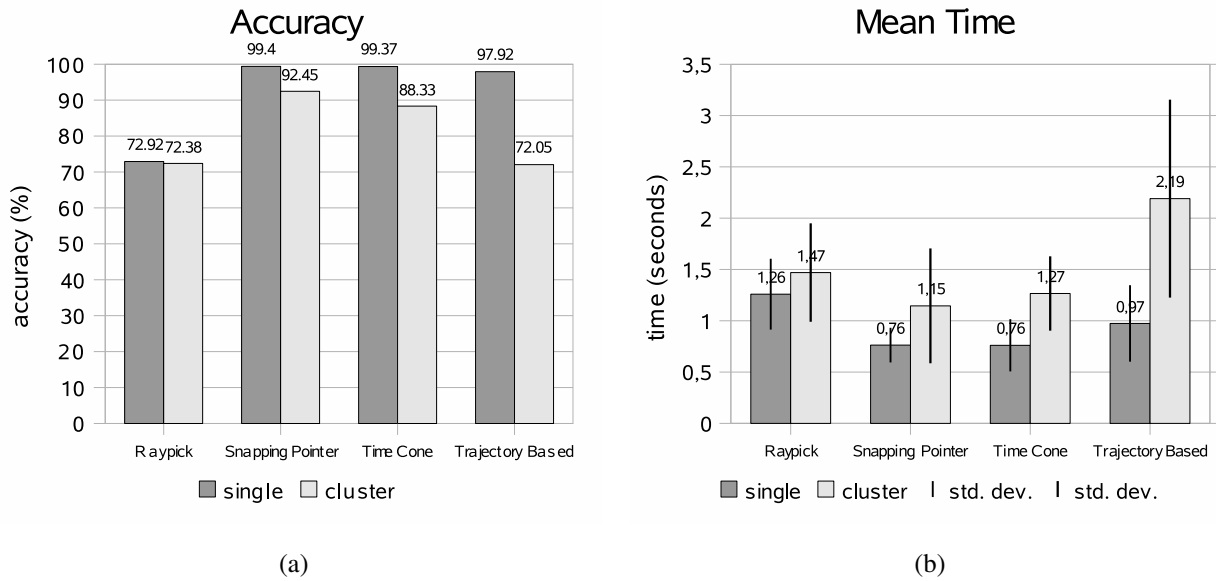


Figure 3: Test results of single and cluster task sets. (a) Accuracy: percentage of correctly selected targets. (b) Mean time to select target with standard deviation.

is typically not considered to be acceptable. It is interesting to note that there are no significant differences between the single selection and cluster selection with this technique.

As expected for the single task, the three assistive techniques have roughly the same large advantage over Raypicking. They all come close to 100% accuracy at less than 1 second mean time. This is to be expected as the assistive techniques essentially make the selection for the user in the case of only a single selectable target. As can be expected, accuracy is lower and mean time is higher for clusters tasks than single tasks using the assistive techniques.

The Snapping Pointer technique performs well in both task sets with high accuracy and reduced time. Observations indicated that the susceptibility to selection flicker is the main cause for the slightly reduced performance in the cluster tasks. Our testing showed it to be the most suitable method for dynamic selection; User ratings of the techniques supported this conclusion.

The Time Cone comes second in test performance and is almost on par in user rating. The robustness of the method to selection flicker is achieved at the price of increased selection time. Time Cone requires time to switch the focus to subsequently selected objects, causing a noticeable lag.

The Trajectory Based selection technique shows a dramatic increase in the time needed to select an object for the cluster task, while maintaining an accuracy approximate to Raypicking. Informal evaluations, that accompanied the development, showed that the method works well for movements that differ significantly in direction or speed. However, it is evident from the user tests, that it reaches its limits as these differences become small. The moving objects in the cluster task could not be sufficiently discerned with this method; selection flicker to similarly moving objects occurred. Since the method is effectively based on a derivative value, it is highly susceptible to noise. Even with smoothing, hand jitter and input data noise were major issues. In its current implementation, Trajectory Based selection clearly has limited applicability.

6 Conclusions

In this paper we have presented initial work on the development of techniques for interaction with dynamic content. The design space of Dynamic Virtual Environments was briefly introduced and the issues of selection in such environments were discussed. Three new techniques for the selection of objects were developed and introduced. An initial evaluation of selection of dynamic objects using these techniques was presented. A basic analysis of the data accumulated in user testing has been performed. From that analysis and the qualitative feedback from the test users, the following conclusions can be drawn:

Although performing better than expected in tests and user rating, the traditional Raypicking method does not perform suitably for the selection of dynamic content. The study indicated that the developed Trajectory Based selection technique is suited only for objects with a significant difference in speed or direction. For very similarly moving objects, its performance drops significantly. It was to be strongly susceptible to jitter and input data noise. It was also the least liked of the methods tested. Trajectory Based selection, in its prototype implementation, clearly has a restricted range of application.

The Snapping Pointer technique is well suited to support interaction with moving objects. The Snapping Pointer was found to be susceptible to selection flicker, i.e. briefly switching to an obstructing object that crosses the target's path. This could, for example, be remedied by introducing a hysteresis function. Our implementation also requires a definition of which objects are selectable a priori. The Time Cone technique proved to be well suited for dynamic content and robust against obstructing objects. Its robustness is achieved at the price of a delay to switch to subsequent targets.

After this initial survey, a more thorough analysis of the existing data as well as further development and testing of the techniques is planned. Correlations to target velocity, direction, distance and relative size need to be analyzed. We believe this initial investigation of Dynamic Virtual Environments can form a basis for further research and development into this interesting area.

References

- [BMSB03] A. Brouwer, T. Middelburg, J.B.J. Smeets, and E. Brenner. Hitting moving targets: A dissociation between the use of the target's speed and direction of motion. *Experimental Brain Research*, (152):368–375, 2003.
- [BSB05] A.-M. Brouwer, J. B.J. Smeets, and E. Brenner. Hitting moving targets: Effects of target speed and dimensions on movement time. *Experimental Brain Research*, 165(1):28–36, 05 2005.
- [dHKP05] Gerwin de Haan, Michal Koutek, and Frits H. Post. IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection. In *Proceedings of the 9th IPT and 11th Eurographics VE Workshop (EGVE)*, pages 201–209, 2005.

- [DPB04] Joost C. Dessing, C. E. Peper, and Peter J. Beek. A Comparison of Real Catching With Catching Using Stereoscopic Visual Displays. In *ECOLOGICAL PSYCHOLOGY*, volume 16, pages 1–21. Lawrence Erlbaum Associates, Inc., 2004.
- [Fit54] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47:381–391, 1954.
- [LG94] Jiandong Liang and Mark Green. JDCAD: A highly interactive 3D modeling system. *Computers & Graphics*, 18(4):499–506, 1994.
- [MSR89] Michael J. Massimino, Thomas B. Sheridan, and James B. Roseborough. One handed tracking in six degrees of freedom. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 498–503, 1989.
- [PBWI96] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. The Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR. In *UIST '96: Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 79–80, 1996.
- [RW99] Simon K. Rushton and John P. Wann. Weighted combination of size and disparity: a computational model for timing a ball catch. In *Nature Neuroscience*, volume 2, pages 186 – 190, 1999.
- [Ste06] Anthony Steed. Towards a General Model for Selection in Virtual Environments. In *IEEE Symposium on 3D User Interfaces*, pages 103–110. IEEE Computer Society, 2006.
- [ZBM94] Shumin Zhai, William Buxton, and Paul Milgram. The Silk Cursor: investigating transparency for 3D target acquisition. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 459–464. ACM Press, 1994.
- [ZM93] Shumin Zhai and Paul Milgram. Human Performance Evaluation of Isometric and Elastic Rate Controllers in a 6oF Tracking Task. In *VRAIS '93: Proceedings of the Virtual Reality Annual International Symposium*, pages 155–161. IEEE, 1993.
- [ZM03] Frank T. J. M. Zaal and Claire F. Michaels. The information for catching fly balls: judging and intercepting virtual balls in a CAVE. In *Journal of Experimental Psychology. Human perception and performance*, volume 29, pages 537–555, 2003.