Towards Preferences in Virtual Environment Interfaces

Chadwick A. Wingrave, Doug A. Bowman[1], Naren Ramakrishnan[1]

College of Computing, Georgia Tech, 801 Atlantic Drive, Atlanta, Georgia, USA
[1]Department of Computer Science, Virginia Tech, 660 McBryde Hall, Blacksburg, Virginia USA

ABSTRACT
Virtual Environment interfaces are designed by implementing an interaction metaphor and comparing it to existing implementations. This technique has proven effective in desktop interfaces but the difficulty of working inside a VE remains because VE interfaces do not understand what the user is trying to do, only what the metaphor wants to do. To handle this problem, we investigated a lower-level approach in interface design of letting the user work as they wish and the interface adapting to the user's method of interaction. Two exploratory experiments were performed on the task of selection to learn how users want to work, with the results being that users do not know until guided by affordances and feedback. Discussed is the intelligent capturing and dealing with VE interface data in terms of Nuances that can represent the details of the interface.

Categories and Subjects: H.5.2 [User Interfaces]: Graphical User Interfaces, Theory and Methods, User-Centered Design; I.3.6 [Methodology and Techniques]: Interaction Techniques

1 Introduction

Virtual Environment (VE) interfaces can be designed by either choosing a metaphor and using its associated techniques or identifying techniques that cover the requirements of the interface. Such design methods are top-down and give the user an interface as opposed to understanding what it is the user is perceiving and trying to accomplish. We believe there needs to be a lower-level understanding of the user and the interface with the design built on affordances and feedback.

In VEs, the environments can appear so real and the means of interacting so similar to natural human interaction, users are sorely disappointed when they realize how poor the interaction is. Therefore, they adapt to the imposed interface and yet for some reason, do not seem to mind. Our goal as designers should be to understand how users want to work innately and give them that interface with any optimizations for speed and comfort we can provide.

In this work, we wanted to create interfaces that were designed by the user. To do this, we focused on the task of selection. This interface, since created by the user, would account for their method of thinking and physical form. To do this, we let them guide and change the base-line techniques for selection based upon their preference, being careful not to guide but yet make sure they explored many possible design configurations. This type of work, when combined with intelligent interfaces, could lead to a major increase in usability.

Not only is this a different way of looking at interface design but this is a different way of looking at intelligent interfaces. Most intelligent interfaces focus on agents gathering information to reduce user cognitive loads or agents performing simple tasks for the user. The approach of our research is to first identify how users wish to work and second use intelligence to recognize the indicators of their wishes. The second part has already been shown to be feasible and very effective in a limited case[17]. The first part, the understanding of how users wish to work, is covered in this paper. The two together would create interfaces that change and adapt based upon the user and task at hand.

1.1 Selection Task

This work focuses on only one of the four tasks in VEs [2], the task of selection. Selection tasks are what the user does when singling out a specific object or point in a VE. Most metaphors for this can be broken down into the following categories; ray casting, occlusion and arm extension. Ray casting is where a ray, going to infinity, is projected from the user's finger and objects that intersect can be selected. Usually a button is pressed at that point to verify that the user truly did intend to select that object. Its feedback is the ray coming out from the fingertip and can be implemented such that objects are highlighted when the ray falls on top of them. Occlusion [12] is similar to ray casting in that a ray is drawn and falls on an object but that ray originates from the user's eye and continues through a point, usually the fingertip, to infinity. Arm extension has several implementations that vary only slightly in each case. The concept is that of the hand being the point of selection and it moving in the environment according to some function of the distance it is away from the user. A simple case would be a one-to-one linear scaling so if the user moves their tracked hand one foot forward, the virtual hand moves one foot forward. The linear function can also be scaled such that one real inch forward is ten virtual inches forward, one foot is ten virtual feet, etc. This helps in selecting objects at a distance. Another implementation, the Go-Go [13] technique uses a linear scaled hand up close but as the hand extends, it scales exponentially. This increases the range of the hand. Each of these techniques has their advantages but no single selection metaphor is optimal in all cases as it has been shown that pointing is more optimal for objects that are distant but arm extension is more optimal for near [14].

1.2 A Neural Network Attempt

In one of our previous experiments [17], we were able to create a selection technique using Neural Networks that was very robust. The environment had three balls positioned in front of the user, just out of reach. The user trained the NN by doing several selection trials where they selected the ball colored red by pinching their fingers together when they felt they were indicating the ball. The NN was then trained and was able to identify the ball they were indicating with little error and little user cognitive load. The reason for this was that the NN learned the correlation between hand and ball location. Additionally, it took advantage of the user unconsciously orienting their hand because the natural action of selecting a ball from others in a line had the wrist rotating in a consistent manner. The failing of the NN was that the representation only allowed for three balls to exist. This led us to realize that in order to build better interfaces, we have to recognize the atomic parts and build a system that operated on the representation.

2 Experiments

To understand how to build the interface, we chose to take a hands-off approach and observe the user in the constrained task of selecting a single sphere. All the data was recorded and we mined the data afterwards to look for trends and clues as to the type of atomic representation that could be used.

The equipment used for these phases was an SGI Indigo 2 with Max Impact graphics with the user inside a Virtual Reality V8 Head Mounted Display (HMD). They had their hands and head tracked using a Polhemus 3 Space Fastrak™ magnetic tracker and finger pinches were recorded using Fakespace PinchGloves™. A selection was considered to have taken place when the user pinched either their index and thumb or middle and thumb fingers together. Since the purpose of this research was an exploration into a direction for larger statistical studies, only eight users were used until the experimental design could be finalized. All eight of the users in this exploratory study were taken from a graduate level course on virtual environments though not all had experience with VEs at the time of the study. There were five males and three females between the ages of 24 and 54. Their compensation was receiving extra credit.

1.3 Phase 1

Phase 1 was to discover how users preferred to use three VE selection techniques: arm extension, ray casting and occlusion. Most interface designers stop at these high-level techniques and do not try to tune them for the user beyond the obvious. For example, ray casting is almost always implemented with a ray extending directly from the hand without trying to discover if this is the optimal configuration. For each technique, users were told how the technique worked in wording vague enough to not guide their actions but informative enough to let them know how it works and is implemented (Table 1). The concept was that users have an existing model of how they wish to interact with the environment and if we isolate that underlying model, then we can use that knowledge to recognize their actions.

Ray Casting

The Ray Casting technique involves pointing a hand at an object and pinching when you believe that a line extends from your hand to the object.

Arm Extension

The Arm Extension technique involves reaching your hand out to the object to be selected. When you feel that your extension has indicated the object that you wish to select, you will pinch your finger and thumb. This is rather vague so discuss this technique with the researcher to make sure that you have a good understanding of it.

Occlusion Selection

The Occlusion Selection technique involves placing the tip of your finger between your eye and the object so that the fingertip occludes or covers the object you wish to select.

Table 1. The wording of the selection techniques given to the users was left intentionally vague so as not to guide the users actions.

The difficulty we encountered was to make an interface where the user would act naturally and not adapt. To this end, we attempted to remove all forms of feedback from the interface relevant to each selection technique. For this reason we did not implement ray casting with a ray extending from the user's finger or even a hand for arm extension. We then assumed that since the user was operating according to their own definition of optimality, and we knew their goal to be the selection of the orange sphere, then each time they conclude a selection with a pinch, they were correct in their selection.

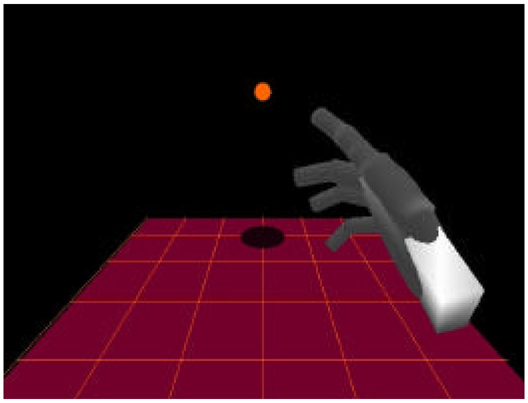

Figure 1 Phase 1 with the ball at a distant position. Notice how the shadow of the sphere and the gridded floor helps to add depth to the sparse scene.

1.3.1 Environment

The environment (Figure 1) in all three trials, one for each selection technique, had the user standing on a platform overlooking a floor with one orange sphere that they were told to select using the selection technique that was currently being tested. Their head was tracked and a virtual hand, at the same position and orientation as the user's physical hand, was shown except in the arm extension technique. To account for a lack of depth cues, the users were told the sphere was the same size throughout the experiment and that the floor was a grid of one-meter squares. There was also a shadow, properly scaled for depth and approximately scaled for height, placed below the sphere on the ground. Each trial had 38 episodes where the sphere was moved through different locations with the first three episodes being the sphere at its furthest distance, middle distance and closest distance to help the user understand the environment's depth. Twenty-seven episodes had the sphere randomly located at a position of near, mid and far; low, level and high; left, center and right. The final eight had the sphere positioned very near the user at a position of close left, center and close right; close low, level, and close right. One side-effect noticed in the pilot study was that users entered a cycle of quickly performing selections without thought. To counter, we added a three second pause between each selection episode and added an audible sound to tell them when the orange sphere reappeared. This had the effect of slowing them down and making them think about their task.

1.3.2 Data Collected

The data from this phase was the position of the user's head and hands as well as the position of the sphere at the time of pinch. This data was then used to discover if users made consistent errors based upon the location of the intended object of selection and the selection technique. Clustering techniques were then used.

1.3.3 Results

With users free from the feedback of the interface, we expected them to revert to their most natural model of interaction built off of innate and proprioceptive intuition. What occurred was an amazing display of adaptation on the part of the user, completely unnatural and inefficient but incredibly adept at making use of the scarce feedback that was left in the system. As an example, one subject spent the entire occlusion selection trial making selections with their palm facing out. This is a very uncomfortable position, even for short periods of time, and especially for objects elevated in the environment. Most importantly, the palm occlusion *completely* occludes the environment reducing accuracy.

There were some strategies observed. One was that users tended to roll their wrists clock-wise when selecting objects that were high or close to them. Another was that since the objects being selected were spherical and since the palm of the hand was circular, some users attempted to position their hand such that the sphere was perfectly occluded by the palm of the hand achieving an eclipse, much like aperture based selection. Another strategy was to switch to pinching with the middle finger and thumb instead of the index and thumb to make a selection, to avoid the "Heisenberg effect" [4] which adds errors to pinches (Figure 2).

Figure 2 The "Heisenberg" effect of making a selection induces errors in the orientation of the tracker device.

Users of arm extension were found to not have a concept of depth. We expected users to scale the extension of their arm to the objects being selected but found that users only divided space into "far" and "near" with far being a fully extended arm and near being a half extension. The lack of other objects in the scene at the time of pinch could have made indicating depth unnecessary so the result may be inconclusive. Because of this and the fact that users do not seem to have a good grasp of depth except through proprioception [8], the usefulness of arm extension, without a following manipulation task, was questioned.

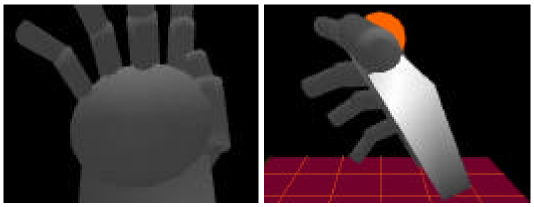

Figure 3. Two occlusion selections used most commonly in phase 1. Left is the palm occlude (with the sphere behind the palm) and right is the thumb knuckle occlude. Both are inaccurate and highly occlude the scene but for some reason users converged to them.

Occlusion selection contained the most interesting results. Since we did not remove the hand from the scene, users had almost all the feedback of the full implementation. Because of this, we expected nearly optimal usage. The users however choose unusual points on the hand as the occluding points. The two most common were actually the palm of the hand and the knuckle where the thumb meets the hand (Figure 3). The palm of the hand occlusion technique occluded most of the scene making the accuracy very low. The thumb knuckle technique is inaccurate and again occluding. It does however leave the hand in a natural and thus non-fatiguing state. A few users did choose to use the more accurate and less occluding fingertips.

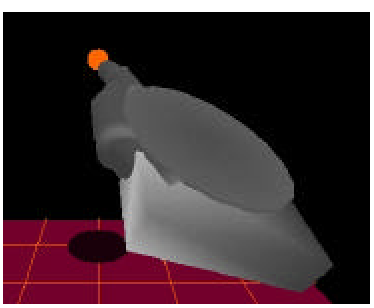

Figure 4 All but one user considered ray casting to be like a fingertip occlusion technique in Phase 1.

For ray casting selection, only one user did true ray casting. All the other users occluded the object with the tip of their finger and considered that pointing at the object (Figure 4). This completely voided the concept of "shooting-from-the-hip" to reduce fatigue but with the lack of a ray extending from the fingertip, this provided the most feedback to the user.

1.3.4 What We Learned

The results lead to the following conclusion:

> Users largely do not have a model of interaction with the environment but a model of how to respond to feedback the environment provides.

Stated another way, when freed from an interface, users attempt to align their actions with feedback and affordances in the environment and not necessarily some innate model based upon real-world experience. The effect of user experience may play an important role.

1.3.5 Aftermath

Our original intent was to build personalized selection techniques for the users. After reviewing the results, it was not considered possible to use the data since the users were so inefficient with their interaction in VEs without feedback to guide them. Clustering was performed to see if trends existed in user data but the trends just mimicked the observations. Because of the results, we needed to change our assumptions and retry.

1.4 Phase 2

If users do not have a model of an interface but of how to respond to the feedback and affordances it produces, then the atomic representation we should be isolating are those of the feedback and affordances. So, how do we go about dealing with the problem of the adaptability of the user since even a poorly tuned interface they will adapt to? To do this, we need some sort of pressure to improve as simply asking the users to do so will create slight tuning and much adaptation. We also require some notion of getting users to search for alternate possibilities of selection within a technique. Looked at from another perspective, there are several local maxima of the tuned techniques the user can create and we need to make sure they search through many of them to find their global maxima. Another problem is making the system friendly and usable such that users will not mind staying in the environment long enough to tune the system to their needs. Lastly, what types of feedback and affordances should the selection techniques have and what parameters of the interaction techniques should be tunable?

To provide pressure to improve, we framed the selections into trials of selection episodes where the user was told to select as quickly and accurately as possible. At the end of each trial, a qualitative ranking of their performance was returned to the user in the hope that the competitiveness of the user would make them want to achieve better and better rankings. To provide pressure to search, we asked the users to try several predefined configurations of the selection technique (Tables 2 and 3). Each configuration was either created by an expert or was a typical method of selecting from phase 1. This was done to see how phase 1's results compared against the expert configurations now that feedback and affordances were included.

Ray Casting Selection

Config 1: The ray extends straight from the fingertip with a 10-degree snap-to angle.

Config 2: The ray has negative pitch with a 10-degree snap-to angle.

Config 3: The ray has positive pitch with a 10-degree snap-to angle.

Config 4: The ray is straight but with a 40-degree snap-to angle.

Config 5: The ray is straight but with a 3-degree snap-to angle.

Config 6: The ray has positive pitch and heading with a 10-degree snap-to angle.

Table 2. The six configurations of ray casting used in Phase 2. The original hand orientation is considered to be out flat, palm down.

Occlusion Selection

Config 1: The bull's-eye is on the index finger and has a 10-degree snap-to angle.

Config 2: The bull's-eye is on the middle finger and has a 10-degree snap-to angle.

Config 3: The bull's-eye is on the thumb's knuckle with a 10-degree snap-to angle. This was a configuration that was used by almost every user in the first implementation.

Config 4: The bull's-eye is on the palm of the hand with a 10-degree snap-to angle. This was a configuration that was used by almost every user in the first implementation.

Config 5: The bull's-eye is placed a few centimeters off of the palm with a 10-degree snap-to angle.

Config 6: The bull's-eye is placed on the index finger and has a 45-degree snap-to angle.

Config 7: The bull's-eye is placed on the index finger and has a 3-degree snap-to angle.

Table 3. The seven configurations of occlusion selection used in Phase 2.

To make the interface user-friendly and non-threatening, we did several things. The first was to have the researcher direct the user through the experiment, answering questions along the way. We also allowed the user to work at their own pace and change the predefined paths of the experiment by being able to redo a configuration. Because of the need to have the user control so much of the experiment, we implemented the TULIP menuing system [3] to handle the experiment control. It was chosen because it has low fatigue, allows the occluding hand to be removed from the scene easily and is fairly fast. These properties of TULIP outweigh its steeper learning curve but with the researcher able to answer questions, users quickly understood TULIP's use. Lastly, personalization of a selection technique was done by selecting a parameter of that technique to tune using TULIP and rotating the left hand to change the value.

1.4.1 Environment

The environment was a 3x3x3 array of 27 blue cubes placed in front of the user (Figure 5). The head and hands were tracked and the user was wearing PinchGloves™. In the occlusion selection interface, there was a bull's-eye on the hand and in the ray casting interface there was a ray extending from the fingertip. The left hand was labeled with the TULIP menuing system at all times and its submenus were displayed on the right hand when the choices need to be displayed. The three menu types were: "Configure", "Trials" and "Personalize". The "Configure" menu had seven configurations for occlusion selection and six for ray casting. The "Trials" menu allowed the user to select a trial of 2, 4, 10 and 27 selections as well as to stop the current trial. The options of the "Personalize" menu will be explained. The environment displayed text to the user such as the ranking they receive for each trial, if a selection was correct or not and when each trial started.

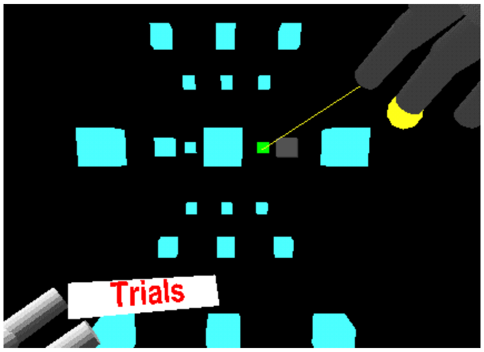

Figure 5. Phase 2 had users selecting among many cubes with either a ray or a bulls-eye shown. Here, both are pictured. To the lower left is part of the TULIP menuing system.

There was one environment for each selection technique. In each environment, the user was asked to do at least one trial of 10 episodes for each predefined configuration. They were then asked to rate that configuration on a scale of 1 to 5. After going through all the configurations, they were introduced to the methods of personalizing the interface. The researcher encouraged them to do several small trials with each personalized technique and then at least two full trials (27 selections) with their configuration.

1.4.2 Selection Techniques: Their Feedback and Tunable Attributes

There were two selection techniques in Phase 2; ray casting and occlusion selection. Arm extension was dropped because it was deemed to be not well suited for selecting at a distance. Again, all selection techniques were triggered by either a middle and thumb or index and thumb pinch.

The passive feedback for ray casting was the ray extending from the fingertip. This guided users in their selection, such as when the ray passed over objects, so it was easy to see where the ray was in relation to the object. Active feedback was also added such that when the user's ray passed close to an object, the ray snaps to that object, changing the color of the ray and object. The properties that were tunable with this implementation were the yaw and pitch values of how the ray extends off of the fingertip with yaw being the angle across the fingers and pitch being the direction the fingers move when the hand closes. When the user tuned the values by rotating their left hand, the angles change immediately so they had an immediate feedback as to the newly tuned position of the ray. The user also had control over the snap-to angle and when they were tuning it, a cone representing the snap-to angle was drawn.

In occlusion selection, a user-facing, passive feedback, bull's-eye was attached to the hand representing the point where the ray from the eye passes through and extended into the environment. Additionally, there was an active feedback snap-to angle but instead of a ray snapping-to, the bull's-eye snapped-to and changed color. The tunable properties were the x (across fingertips), y (along fingers) and z (out from palm) positions of the bull's-eye in relation to the hand as well as the snap-to angle.

1.4.3 Data Collected

Data was collected from several sources in this experiment. The system logged data on user trials, accuracy and preference. The user used the speak-aloud protocol for qualitative data and gave rating of each configuration which was recorded along with the researcher's own observations. There was also a comfort rating and post-experiment questionnaire.

1.4.4 Results

After this implementation, we obtained acceptable results for tunable properties of the selection techniques. Users were able to modify the selection techniques easily. They also seemed eager enough to try extra trials with the average number of trials being 15 when the minimum required amount was 8 for occlusion selection and 7 for ray casting. This was encouraging and made us believe that users did search the interaction space well.

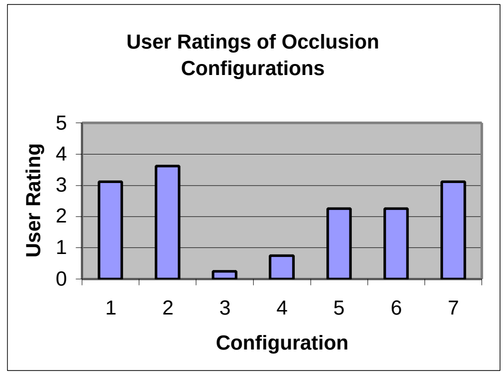

Figure 6. User ratings of the predefined configurations showed that they did not like their configurations from the previous phase (3 and 4).

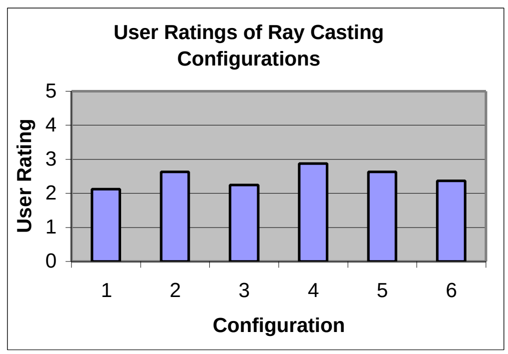

Figure 7. Users tended to not have a major preference among the ray casting configurations.

1.4.4.1 General User Ratings

The configurations for occlusion selection that were ranked highest were generally those where the bull's-eye was on one of the fingertips, configurations 1, 2, 6 and 7 (Figure 6), as was expected. The only difference between those configurations was the snap-to angle and the finger that users chose to pinch. Configuration 5 had the bulls-eye a few centimeters off the palm. Configurations 3 and 4 of occlusion selection, which were preferred in Phase 1, were ranked very low as compared to the other configurations as was expected. These results add support to the notion that users need feedback to determine the value of a configuration. In the ray casting configurations (Figure 7), there was little variance among the different configurations. Also, various snap-to angles made little difference. In general, snap-to angles were set low in occlusion configurations, possibly because there was no need for feedback to try and refine a selection since the proprioceptive sense helps guide the user. In ray casting, snap-to angles were larger though users complained about the ray flickering between objects. This snap-to could be more useful in sparser environments where the flickering would not be such a problem.

Quantitative ratings were taken in user surveys on factors of the selection techniques such as speed and accuracy and the tunability and comfort of the technique. Figure 8 gives the user's average rating of all the configurations for these four criteria in each selection technique. Occlusion selection was shown to be preferred to ray casting in all but the comfort category. This was expected because ray casting can be performed with the hand in a resting position.

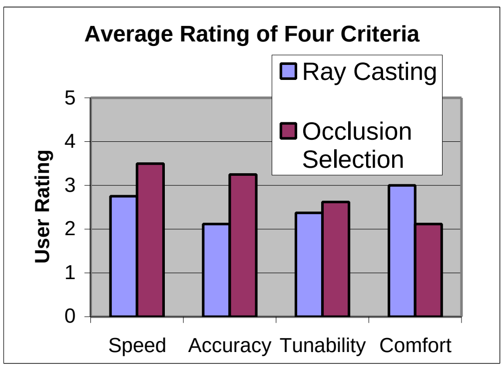

Figure 8. The average user rating of four criteria taken from user surveys shows users preferring Occlusion selection in all but the comfort criteria.

There were a few different configurations that users preferred. For the most part, there was very little tinkering with factors other than the snap-to angles as most users knew what they wanted in their configuration from the start. Users of occlusion selection had no preference between either their index or middle finger for selection. The major difference between the occlusion techniques was the snap-to angle which was set to (in degrees) 20.60, 16.02, 0.0, 10.0, 4.47, 1.83 and 3.0 twice. Ray casting had four general configurations (Figure 9) with the first being the ray extending straight from the hand with normal to larger snap-to angle. The second personalized ray-casting configuration had the ray extending up and to the right when the hand was parallel with the ground. This lets the user hold the hand in front in a natural 45-degree angle and make selections with a relatively small snap-to angle of 4.54 degrees. This configuration was not anticipated. Another configuration not anticipated had the ray extending down and to the left when the hand was placed flat. This allowed them to keep the hand in a natural position of close to a 45-degree angle and low, which reduced stress. Users of these two unanticipated tunings also liked large snap-to angles which might be explained by the difficulty in aiming the ray when the hand is so far from the head. In the final unanticipated configuration, the hand was held pitched up and rotated outward. This placed the hand close to the face and yet not held too far in front of the user so as to be accurate (close to the eyes) and less fatiguing (not extended). There was a relatively small snap-to angle which made sense because with the hand close to the eyes, it was already easy to see where the ray was pointing.

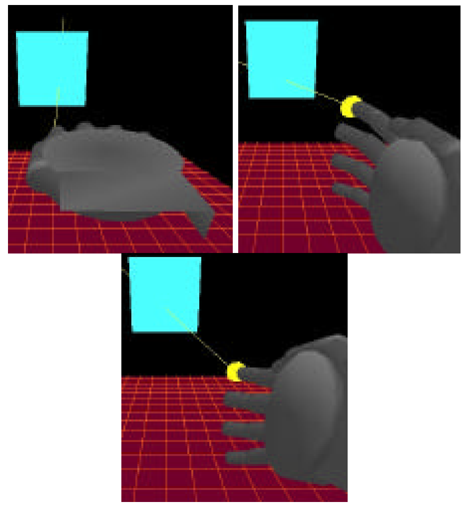

Figure 9 Unusual ray casting configurations created by the users: (left) the ray extends up and to the right (center) the ray extends down and to the left (right) the ray extends up

1.4.4.2 Occluders and Ray Casters

Users were also asked which technique they preferred overall. This was used to divide the users into ray casters and occluders; we had four of each. These two groups had different trends of preference for the selection configurations (Figures 10 and 11). Because of the low number of users in each group, statistical evaluation was not performed but there were clear monotonic divides in the preference data.

Of interest in the ray casting configurations are configuration 3 and 4. In configuration 4, occluders liked the wide snap-to angles as opposed to other ray casting techniques. This is probably because with a large snap-to angle, the task becomes less of a ray casting technique and more of a feedback alignment task. Configuration 3 is probably the least like pointing because when the ray was aligned with the object, the user was pointing at the ground. Occluders overall only liked configuration 4 more than this technique whereas ray casters liked it the least. The other configurations were liked more by ray casters than occluders as would be expected.

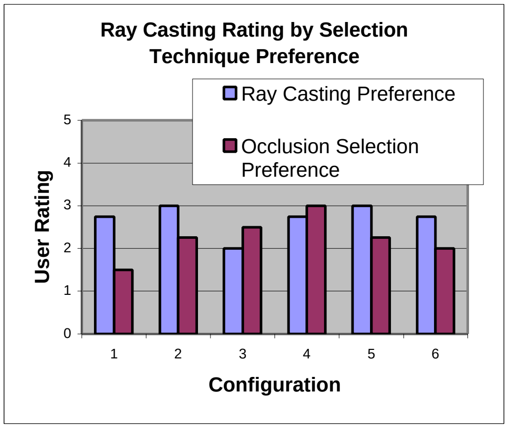

Figure 10. Dividing users into ray casters and occluders showed that trends exist for preferences between the ray casting configurations.

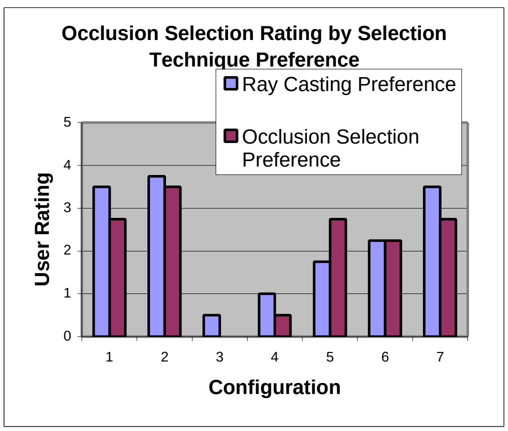

Figure 11. Dividing users into ray casters and occluders showed that trends exist for preferences between the occlusion selection configurations.

Of interest in the occlusion configurations is the discrepancy of preference of configurations 1 and 7. These two configurations have the bulls-eye on the index finger with low to mid snap-to angles. This made its use very much like the mistaken Phase 1 Ray Casting technique where users actually performed Occlusion Selection when asked to use Ray Casting. Additionally, occluders like configuration 5. With its medium snap-to angle and off-the-palm location of the bulls-eye, it allowed the user to place the bulls-eye close to the object and the snap-to angle handled the details. Ray casters were confused by the location of the bulls-eye.

1.4.5 What We Learned

Many users tended to select through the interface rather quickly until they encountered a selection that was difficult to make, due to the object being at a distance, near other objects or the user being sloppy. This difficult selection caused them to slow down and select with more accuracy. Other times this happened were when the feedback was not matching what they were expecting, such as when the bull's-eye was placed a few inches off the palm or when the snap-to angle flickered between objects. This "bad vibes" behavior tended to last for a few selections until the user ramped back up to speed or discovered the source of their confusion.

There were six general parameters identified that users demonstrated which could be used to explain their actions. The first was *spatial awareness*. Some users had the ability to understand where they existed in the space and this greatly improved their ability to use ray casting and most likely extends to other proprioceptive tasks. This is most likely linked to Bodily-kinesthetic and Spatial intelligence, two of Howard Gardner's seven types of intelligence[6]. The next parameter was *feedback alignment*. Users responded to feedback with varying degrees of acceptance. Those with high feedback alignment considered the feedback to be infallible such that any indication of correctness, such as a snap-to event occurring, immediately caused them to pinch their fingers. Users with these tendencies preferred smaller snap-to angles so they could wave their hand at the object and rely on their reflexes to pinch when they were pointing at the object. A third parameter was *exploration*. Those with high exploration were able to create strategies to adapt to bad configurations quickly. They would inspect the current configuration and make changes to how they operate such as orienting the hand differently to reduce occlusion or fatigue. Those with little or no exploration basically never changed the way they selected throughout a configuration no matter the fatigue or difficulty. The forth parameter of a user was *resilience*. A resilient user would not mind fatiguing selection techniques. This can be an advantage because they can use techniques that are accurate or fast but highly fatiguing. This could be a problem for users if they spend a long time in VEs. The fifth parameter was user *precision*. This affected how much users test the boundary conditions of a selection technique. So, if a technique accepts up to fifteen degrees of error, a non-precise user will expand their sloppiness to fit the fifteen-degree boundary condition. Those with a high precision rating will always provide a certain level of precision even though the interface may require less. This extra effort to give that precision can cost time and fatigue and those with low precision will test the boundary conditions more and keep their precision errs within those bounds. The last parameter and near antithesis of precision is user *speed*. This affects how fast the user moves and how long they must receive positive feedback before they go forward with an action. For example, some users in ray-casting would move their hands quickly and pinch quickly whereas others would move slowly, letting the ray dwell on an object for a moment or two, and then pinch. All of these parameters, taken merely from observation, are most likely part of a set of parameters that can be used to explain the user's model of how to perform a selection technique.

The implications are this:

> There exists an innate set of parameters of the user and the interface that can be used to explain the actions of users.

In machine learning, a reward function is a reward given based upon the state of the system and the action of the agent. So, a reward function can be created where the state is based on the interface and these parameters and the actions are those the environment allows. Future work would be to use Inverse Reinforcement Learning [9] to discover the user reward function so that we can recognize and predict user actions.

3 Related Work

Gesture recognition is the most obvious similar domain and has been used for such things as handwriting recognition [5], hand recognition [16] and even walking in place [15]. They have applied hidden markoff models and neural networks to the problem with great success in classifying actions.

Multimodal interfaces combine many modes of human interaction into one interaction metaphor and are currently being applied with success [11]. These modalities can be text, speech, position trackers, gestures, eye tracking, force-feedback devices, haptics, etc. The mixing of modalities allows the user the freedom to choose the correct modality for the situation.

Perceptual user interfaces and machine vision are both very similar in the domains they deal with. Both are attempting to wean any amount of data they can from whatever medium they are sensing such as pictures, video, sound, pressure plates, etc. They then attempt to build robust applications from the small amount of data they can identify. Because of this dearth of data, they have effectively experimented with methods such as HMM, NN and Bayesian Statistics to deal with the problem.

These techniques are limited in that they do not consider the interface as a whole. For example, gestures can be recognized but ignored is any type of correlation between a string of gestures or the speed the gestures were given at. Perceptual user interfaces have acknowledged this fact and use mutual disambiguation [10] to deal with data correlations. Vision systems have acknowledged this fact too, probably because of the field's roots in human perception [7].

4 Nuances

Nuances can frame user intentions by representing the atomic parts, the smallest undividable unit of information that affects user actions, of a VE. Methods placed on top of nuances can map them to interface actions. A system like this has not fully been attempted in the past because of the work required in recognizing all the atomic parts to the interface. A VE interface that uses nuances and intelligently acts on them would not require the interface designer to work out all the details because the interface would have some ability to reason about itself. Given enough observation, it could be possible to compute an interface directly from the nuances since they are an atomic representation.

Nuances operate on nuance *properties*, nuance *concepts* and four categories of nuances, all of which we shall refer to collectively as nuances. For ray-based selection tasks, some of the properties would be the size, shape and location of the objects. A concept could be the angle of error an object lies off of a specified ray. The four categories of nuances are *object*, *environment*, *refinable* and *supplemental*. Object nuances arise from some affordance of the object whereas environmental nuances represent the dependencies between objects due to the user's perceptual belief or environment design. Refinable nuances alter behavior of existing nuances. There are also supplementary nuances that are grouped into *strategies* that users develop and *discoverables* which arise from interactions between behaviors which were not intended. A better explanation with examples can be found in [18].

There are two parts to interface design that nuances can help with. The first part is the recognition of when the user wants to perform an action. In the past, this has been a trivial task since there is little ambiguity in 2D and command-line interfaces. For VEs, this is not so trivial. The second part is the optimization of how the user performs tasks. For instance, when a user wants to perform an action, there are several steps they must take to complete their goal with each of these steps being identified by the interface. Nuances help because they can represent the identifiable actions a user takes (the actions induced from observational data) and support the optimization of the paths to complete a task (the deduction of how to perform their actions faster).

The design of nuances will be built upon the atomic interface data. This data can then be recorded and fed into learning algorithms to train nuance libraries. Over time, as atomic interface data are discovered and training data is logged in nuance libraries, adding interfaces to a VE would be performed by simply describing user tasks and placing objects in the environment.

1.5 Patterns

The usage of nuances has similarities to architectural patterns. Christopher Alexander created patterns from his observations of successful designs, the purpose being that once listed as a pattern, the information can be reused. This was useful in the design of a large Peruvian housing complex [1] where he created a simple step-wise process to design each house according to the wants and needs of the occupants. Similarly, with nuances we will be able to store VE information and reason with the information. That manner of reasoning is the purpose of our future work and will most likely take the form of functionally modeling the user.

5 Conclusion

The implications of this work are subtle but important. If we are going to build interfaces for people in VEs, then we must pay close attention to how they think and perceive. This is because people form their model of how to behave in a large part from the mixes of feedback and affordances in the environment and not from some internal model. This will require splitting these affordances and feedback into atomic chunks and studying interfaces not just as a sum of these atomic parts but also as a sum of the interaction of the parts. This will require different methods of storing and working with interface information, undoubtedly requiring computational intelligence. We propose nuances in a Nuance-Oriented interface as a means of representing and reasoning with the atomic parts as a whole.

References

1. Alexander, C. et. al. *Houses Generated by Patterns.* Center for Environmental Structures, 1969.
2. Bowman, D. “Interaction Techniques for Common Tasks in Immersive Virtual Environments: Design, Evaluation, and Application” Georgia Tech Diss., 1999.
3. Bowman, D. Wingrave, C. “Design and Evaluation of Menu Systems for Immersive Virtual Environments”, *Proceedings of IEEE Virtual Reality 2001*, pg 149-156.
4. Bowman, D. Wingrave, C. Campbell, J. Ly, V. “Using PinchGloves for both Natural and Abstract Interaction Techniques in Virtual Environments”, *HCI International* 2001.
5. Garris, M. D. “Intelligent System for Reading Handwriting on Forms”, 1998 *International Conference on System Science*, v3, p233-242.
6. Gardner, H. (1983). *Frames of Mind.* New York: Basic Books.
7. Marr, D. *Vision: A Computational investigation into the Human Representation and Processing of Visual Information*, W.H. Freeman, San Francisco, 1982.
8. Mine, M. Brooks, F. Sequin, C. “Moving Objects in Space: Exploiting Proprioception in Virtual Environment Interaction”, *Proceedings of SIGGRAPH*, pg 19-26, 1997.
9. Ng, A. Y. Russell, S. “Algorithms for inverse reinforcement learning.” *International Conference on Machine Learning*, 2000, Stanford, CA: Morgan Kaufmann.
10. Oviatt, S. “Mutual disambiguation of recognition errors in a multimodel architecture”. *Proceeding of the CHI 99 conf. on Human factors in computing systems: the CHI is the limit*, 1999, pg 576 – 583.
11. Oviatt, S. Cohen, P. "Multimodal Interfaces That Process What Comes Naturally" *Communications of the ACM*, Vol. 43, No. 3, March, 2000, pp. 45-53.
12. Pierce, J. Forsberg, A. Conway, M. Hong, S. Zeleznik, R. Mine, M. “Image plane interaction techniques in 3D immersive environments” *Proc. of the 1997 symposium on Interactive 3D graphics*, pg 39.
13. Poupyrev, I. Billinghurst, M. Weghorst, S. Ichikawa, T. “The Go-Go Interaction Technique: Non-Linear Mapping for Fdirect Manipulation in VR”, *Proceedings of the ACM Symposium on User Interface Software and Technology*, pg 78-80, 1996.
14. Poupyrev, I. Weghorst, S. Billinghurst, M. Ichikawa, T. “A Framework and Testbed for Studying Manipulation Techniques for Immersive VR”, 1997 *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pg 21-28.
15. Slater, M. Usoh, M. Steed, A. “Taking Steps: The Influence of a Walking Technique on Presence in Virtual Reality.” *ACM Transactions on Computer-Human Interaction*, vol 2, no 3, pg 201-219, 1995.
16. Starner, T. Weaver, J. Pentland, A. “Real-time american sign language recognition using desk and wearable computer-based video”. *IEEE Trans. Pattern Anal. Machine Intell.*, 20(12), December 1998.
17. Wingrave, C. Bowman, D. Ramakrishnan, N. “A First Step Towards Nuance-Oriented Interfaces for Virtual Environments”. *Proceedings of the Virtual Reality International Conference ’01*, pg 181-188.
18. Wingrave, C. “Nuance-Oriented Interfaces in Virtual Environments”. Virginia Tech thesis, 2001 www.cc.gatech.edu/~cwingrav/papers/thesis.pdf