A Framework and Testbed for Studying Manipulation Techniques for Immersive VR

Ivan Poupyrev [1,2], *Suzanne Weghorst* [2], *Mark Billinghurst* [2], *Tadao Ichikawa* [1]

[1] IS Lab, Faculty of Engineering,
Hiroshima University,
1-4-1 Kagamiyama
Higashi-Hiroshima, 739 Japan
{poup, ichikawa}@isl.hiroshima-u.ac.jp

[2] HIT Lab, University of Washington
Box 352142
Seattle, WA, 98195 USA
{weghorst, grof}@hitl.washington.edu

ABSTRACT

Developing virtual reality (VR) applications which enable actual work over a period of time requires optimization of the most basic interactions, such as object manipulation, so that the immersed participant can concentrate on higher-level tasks rather than on low-level motor activities. This paper presents a framework and experimental testbed for studies of VR object manipulation techniques. The framework provides a systematic task analysis of immersive manipulation and suggests a user-specific non-Euclidean system for the measurement of VR spatial relationships. The Virtual Reality Manipulation Assessment Testbed (VRMAT) is a practical implementation of the framework and is a flexible tool allowing in-depth experimental studies of immersive manipulation. Pilot studies have been conducted to evaluate this framework and testbed and to establish a baseline for further development.

Keywords: *immersive virtual reality, VR user interfaces, VR manipulation techniques, user studies, experimental testbed, theoretical frameworks*.

INTRODUCTION

Manipulation of objects in virtual environments (VEs) is often awkward and inconvenient. A lack of a tactile feedback, tracker noise, poor design of interaction techniques, and other factors can make the simple task of grabbing and moving a virtual object a frustrating experience. Numerous studies have focused on how humans manipulate objects in the real world and how tools, workplaces, etc., should be designed to achieve more effective manipulation [5, 14]. Similarly, development of effective VR applications also requires comprehensive understanding of immersive manipulation and, in particular, which virtual tools and techniques should be used and how they should be designed to be easy and effective to use [17, 27]. As Kay Stanney has stated "… if humans can not perform effectively in virtual environments, then further pursuit of this technology may be fruitless" [32].

The main challenge, however, is a methodological one. There is still insufficient understanding of the essential characteristics and parameters of VR manipulation [17]. Although immersive manipulation is similar to manipulation in the real word, there are also significant differences which have to be studied and understood in order to exploit the full potential of VR technology [32].

This paper presents a conceptual framework and experimental testbed for systematic study of interaction techniques for immersive manipulation. Our goal is to develop a formal methodology and experimental tools which can help to understand immersive manipulation and aid developers in making informed decisions when designing manipulation interfaces for VE applications. Our focus here is on the following research issues:

- *An analysis and taxonomy of VR manipulation tasks.*

There probably no optimal interaction techniques for every possible task. Characteristics of different manipulation tasks impose different requirements on manipulation techniques [2, 17, 32]. In an ideal study we would evaluate interaction techniques for every possible task. However, because there are countless manipulation tasks, we are necessarily limited to a small subset of them. This set of test tasks should, first, represent most of the relevant manipulation scenarios so as not bias our studies by focusing on tasks which a priori are better suited to certain techniques. Second, it should permit generalization of our findings beyond the particular conditions of the experiments [2, 13]. The taxonomy of manipulation tasks may also be useful as a guideline for developing immersive interfaces [32].

- *Spatial metrics and their units of measurement.*

As in the real world, the user performance when manipulating in a VE depends on its spatial configurations: positions of objects, their sizes, occlusion and so on. Formal studies of VR manipulation techniques require explicit definition of all relevant VE spatial characteristics and their units of measurement. However, using conventional metrics and Euclidean approaches for studies of human manipulation may be restricting and misleading [14, 21]. Our framework provides an alternative user-centered approach for the definition of spatial layouts and visual stimuli, which normalizes the experimental conditions across subjects.

- *Virtual Reality Manipulation Assessment Testbed (VRMAT)*

The framework has been implemented as a general purpose VR Manipulation Assessment Testbed (VRMAT). The testbed is a flexible, easy re-configurable, experimental tool which allows in-depth studies of immersive manipulation.

There are some restrictions we must put on our framework. First, we consider only immersive VR manipulation. The user is immersed using a head-mounted display or a screen projector, and six degrees of freedom (DOF) sensors track position and orientation of the head and hands. Second, we consider only free hand direct manipulation and do not consider alternative techniques, such as voice and gesture commands, gaze input, and others. Third, we do not allow the user to fly in VE and, therefore, do not consider flying techniques or combinations of flying and manipulations. We also do not consider fine aspects of grasping such as finger positions and gesture recognition. Although our framework and testbed can be applied to a wide variety of studies of immersive manipulation, in this paper we concentrate on interaction techniques only and do not address the effects of the input/output devices used.

RELATED WORK

Studies of human manipulation have a long history stretching back to the pioneering work of Woodworth at the end of the last century [16]. Since manipulation in virtual worlds is similar to manipulation in reality, we can apply the wide body of results from human factors research [16]. There are also significant differences however, which should be considered. First, in VEs users can perform actions which are impossible in the real world. For example using VR interaction techniques (such as [17, 18])

ACM Symposium on Virtual Reality Software and Technology (VRST) 1997, pp. 21-28, Lausanne, Switzerland. Copyright 1997 ACM 0-89791-953-x/97/9

users can manipulate objects located far outside normal human reach. Second, there are many VR-specific factors that affect interaction in virtual worlds: tracker noise, weightlessness of objects, lack of tactile feedback and so on [12, 16, 32].

Manipulation of computer-generated objects was initially studied for 2D interaction techniques, such as menu selection, cursor movements, icon placements and so on [1, 2, 30]. For example, Foley et al. [2] surveyed and classified interaction techniques for 2D graphical input according to the basic interaction tasks and their characteristics. Studies of 3D spatial input were mainly concerned with the evaluation of input devices, such as joysticks, 6DOF position sensors, etc., for spatial manipulation tasks [10, 20]. For example, Zhai et al. [10] compared isometric versus isotonic devices for different conditions of spatial manipulation. Other studies focused on the influence of output device characteristics on user manipulation performance. Nemire [31], for example, studied the effect of visual and aural enhancement on the user's manipulation; Watson et al. [25] studied the effects of frame time variation on tracking and placing task performance.

However, it is not only input and output devices and their characteristics that matter: different interaction techniques allow users to accomplish tasks in different ways using the same input devices [2, 27]. Hinckley et al. [29] surveyed approaches for designing interaction dialogs for spatial input, identified problems and possible solutions. Mine [17] surveyed and classified immersive interaction techniques, including those for manipulation.

Few attempts to study and categorize VR manipulation techniques within a systematic approach have been reported. The Virtual Environment Performance Assessment Battery [4] provides a set of standard procedures to investigate human performance in VR, but the scope of testbed is broad and it does not focus on the detailed aspects of VR manipulation and interaction techniques. More relevant pioneering informal usability studies which evaluated several immersive techniques for manipulation at a distance have recently been reported by Bowman and Hodges [11].

ANALYSIS OF IMMERSIVE MANIPULATION

Evaluation of manipulation techniques involves measuring user performance, using some criteria, while they accomplish test tasks [5, 16]. In the ideal study we would evaluate techniques for every task where these techniques can be used. Unfortunately it is not feasible due to the large number of possible tasks. It is important therefore to identify a basic set of test tasks that is small enough to be useful but cover most of the relevant manipulation scenarios. In particular, it should not bias our studies by focusing on tasks which a priori are better suited for some techniques [5, 13]. Furthermore, this set of tasks should permit generalization of findings beyond the specific conditions of the experiments i.e., insure external validity of the studies [4, 13].

Basic direct manipulation tasks

The general assumption of the task analyses is that the requisite of human efforts in all cases is composed of the same basic tasks, which are building blocks for more complex interaction scenarios [2, 14, 19]. Consequently, if we dissect immersive manipulation into several basic manipulation tasks we can use them as test task for our studies.

Intuitively, we can suggest that the basic VR manipulation tasks are the same tasks that we perform in the real world when we make positioning movements. We make position and orientation movements every time we reach for and/or move something to another location [5, 14]. These movements are a combination of reaching/grabbing, moving and orienting of objects.

This conclusion is consistent with the taxonomy of interaction tasks suggested by Foley et al. for computer graphics based interaction [2, 3] and supported by other researchers [8, 17]. They suggest five basic interaction tasks:

- *position* - the task of positioning an object;
- *selection* - the task of identifying an object (also referred as a target acquisition task [26]);
- *orient* - the task of orienting an object;
- *text* - the input of a string of characters;
- *quantify* - the input of a numerical value.

Mine [17] also suggested a *scale* task as a basic task for VR interaction. However, because the scale task is usually implemented in terms of the selection, positioning and orienting, these are the basic test tasks that we use for our studies.

Characteristics of the basic manipulation tasks

Identifying basic tasks, however, is not enough: even for basic tasks there are many parameters which affect user performance and, hence, must be considered [2, 6]. The user performance for an object selection task, for example, depends on the distance to and size of the object to be selected: an object located nearby could be easier to select than an object located outside the user's reach [16, 17]. Foley et al. [2] called these task characteristics *application requirements*; Norman referred to them as *task aspects* [6]; and Grissom et al. used the term *subtasks* to refer to the variations of the same task with different characteristics [8]. Generally, task parameters are all those factors which influence user performance while accomplishing tasks. They can be classified as follows [2, 6, 8, 10, 12, 17, 32]:

- *User-dependent*: experience, cognitive, perceptual and motor abilities, anthropometrical differences and so on.
- *Input/output device dependent*: attributes of the devices such as degrees of freedom, resolution, field of view, supported depth cues and others.
- *Interaction techniques dependent*: underlying metaphors of techniques, their design and implementation.
- *Application dependent*: configuration of the VE, size, shape and locations of objects, color, lightning and others.
- *Task context dependent*: required precision, initial and final conditions of the task, task constraints and others.

In the following sections we discuss parameters of the three basic manipulation tasks. We do not attempt to address every conceivable parameter; instead we discuss those which are most salient according to our experience and the related literature. We also do not address device and user dependent aspects of these tasks - we will control their influence using an appropriate experimental procedure.

Selection task parameters

- *Number of objects to be selected.*

In the simplest case we need to select only one object. The task, of selecting more than one object is often referred to as a browsing task [23].

- *Distance to the target object.*
- *Size of the target object.*

According to the Fitt's law [9], the time required to select an object depends on the ratio of the distance to the size of the object i.e., a large object located close is easier to select than small one located far away. In immersive VR, however, due to perspective effects, a large object located far away could be more difficult to select than a small one located close by [12, 16].

- *Direction to the target object.*

Different body parts and muscles are brought into the action, depending on the direction to the target relatively to the user, therefore, different performance is achieved [5].

- *Occlusion of the target object.*

When an object is occluded by another object, its selection becomes more difficult [16, 17] due to the smaller visual size and restricted access to the object.

- *Other parameters.*

There are more parameters which might be considered, such as *dynamics* of the target (dynamic target acquisition task [26]), *density* of objects around the target [5], *bounding volume* of the target object (influences 'overshoot' of objects [12, 18]), etc.

Position task parameters

Positioning is the task of moving an object from an initial to a desired final terminal location [5]. The execution of position tasks can be generally dissected into three relatively distinct phases: selection of the object, primary or gross travel, and finally, a corrective motion to position the object on the terminal with the desired accuracy [5, 16]. The parameters to consider are:

- *Initial distance to the manipulated object.*
- *Initial direction to the manipulated object.*
- *Distance to the terminal object.*
- *Direction to the terminal object.*

Studies of positional movements have shown that the time and accuracy of the movements depend on the object's initial position, the travel distance, and the direction from the user to the terminal [5].

- *Required precision of the positioning.*

Precise positioning is more difficult than imprecise for unconstrained movement [5, 16, 27]. Required accuracy affects mostly the last phase of the positioning movement: corrective motion close to the terminal.

- *Other parameters.*

There are more parameters to be considered, such as *visibility of the terminal* (blind positioning), *dynamics of the terminal* (positioning on a moving terminal*), occlusion of the terminal*, *size of the manipulated object,* etc.

Orientation task parameters

Orienting is a task of changing orientation of an object from an initial to a desired final orientation. Orientating consists of object selection, gross orienting and then fine adjustment to achieve desired preciseness of orientation. The parameters of the task to consider are:

- *Direction to the object.*
- *Distance to the object.*

Orientation of an object located close to the user and one located far away is different and may require different interaction techniques [16, 17].

- *Initial orientation.*
- *Final orientation.*

The initial and final orientation define the direction of the object rotation during task execution.

- *Required precision of orientation.*

PERFORMANCE CRITERIA

The goodness of the interaction techniques can be evaluated using the following performance criteria [2, 5, 8]:

- *Completion time:* the time taken to successfully accomplish the tasks. For a selection task this is the time from the moment when the stimulus triggers the user to select a test object until the moment when it has been successfully selected. For positioning and orienting tasks, completion time is measured from the moment when the user picks (selects) a test object until the moment it has been positioned with the required accuracy. Because position and orientation tasks allow iterative manipulation, we can also measure the time of object manipulation only, excluding the time required for each selection.

- *Accuracy:* the proximity to the desired position or orientation of the test object.

- *Error rate:* the number of failed attempts to accomplish the task. For a selection task it is the number of failed attempts to select an object; for positioning and orientation it is the number of iterative movements required for positioning/orienting an object with required accuracy.

Other criteria which can be assessed using performance criteria and questionnaires are [2, 8]:

- *Ease of use:* the cognitive load on the user while using the technique.

- *Ease of learning:* the ability of the user to improve performance with experience.

- *Sense of presence:* the user's sense of immersion and spatial awareness.

SPATIAL METRICS AND UNITS OF MEASUREMENT

To implement basic manipulation tasks and their parameters in a testbed for studying VR manipulation techniques we have to decide how to define and measure task parameters: distances, sizes, positions, occlusion, etc. Formal definitions would allow the experimenter to encode various conditions of spatial manipulation. For example, to investigate the efficiency of selection techniques for different sizes of the objects, the experimenter may present objects of various sizes and ask subjects to select them. However, the question remains as to what is the object size and what units we can use for their measurement and comparison.

For the testbed to be of practical use, these definitions should be relevant to the intended use and permit the generalization of results from the experimental conditions to any other VE. In this section we suggest definitions of the task parameters within a user-centered approach which normalizes the experimental conditions across subjects. We also discuss the limitations of conventional metrics and Euclidean approaches as a basis for studies of VR manipulation.

The user-centered reference frame

In previous studies of 3D user interaction, spatial positions of objects were usually defined as (x, y, z) triples relative to a world-centered Cartesian coordinate system [2, 8, 17]. This approach has been widely used in computer graphics as it allows for easy coding of object-to-object relationships within a 3D scene [3]. However, in studies of immersive manipulation it can be ambiguous and misleading, as it does not allow for proper expression of user-to-object relationships that occur during user-environment interactions, such as object manipulation [7, 21]. In fact, it has been found that knowledge about object location in relation to the user is encoded and processed separately from the knowledge about absolute objects-to-object relationships by different cognitive systems [7].

To overcome this problem, we use a user relative body-centered coordinate system, similar to that used in Kennedy's classic study of the three-dimensional space envelope of seated U.S. Air Force operators [22]. An object's position is defined as the length and orientation of the vector pointing from the chest of the user to the object (Fig. 1). Length d defines the distance between the user and manipulated object, and angles α, β define directions of hand movements during the reaching/positioning of the object.

Measurement of user-to-object distances, virtual cubits

Most of the current studies of human factors in VR use either real world units of measurements, such as meters, or computer graph-

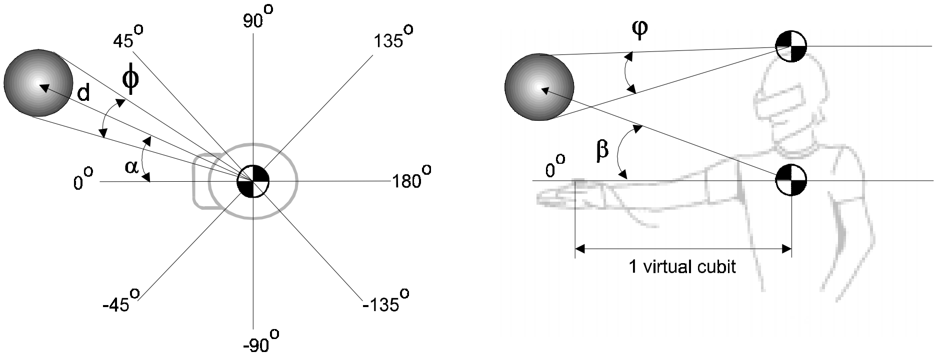

Fig. 1Position of object is defined as distance *d* and direction α, β to the object in user-centered coordinate system. Size of the object is defined in terms of vertical (φ) and horizontal (ϕ) angles of the visual field subtended by the object.

ics units, such as points (for example, see [8, 25]). However, are these the best units for studying immersive manipulation?

One argument for using real world units is the user's familiarity with them. However, because perception of distances and sizes in VR differs from the real world [33], users cannot reliably transfer their real world spatial experience into VEs. In fact, the definition of meter does not relate to the human perceptual or cognitive systems in any way, for example, one definition says that the meter is 1/1000000 of the distance between the pole and equator [14]. We can use real world units for measurements in everyday life only because their physical equivalents have been introduced (as in the case of rulers) [14]. Using meters to study VR manipulation also introduces a bias due to anthropometrical differences among subjects. Indeed, an object located one meter away will be close for one subject and far for another.

Likewise, using computer graphics units, such as points is problematic in that it is impossible to generalize results to other VEs, because immersive distances defined in points depend on implementation. Indeed, from the immersed participant's point of view, the same distance expressed in terms of points can be as small as the user's virtual hand in one VE and as big as a whole environment in the other, depending on the scale of the VE.

Both real world and computer graphics units of measurements are user independent, which makes them restrictive as a basis for studies of immersive manipulation. To overcome this problem we define a user-dependent unit of distance measurements, which is equivalent to the length of the user's maximum reach in a VE (Fig. 1). We call it a *virtual cubit* after the *cubit* - a unit of measurement used in ancient Rome, equal to the distance between the elbow and the tip of middle finger.

Although the virtual cubit is a user-dependent unit of measurement, VEs where distances are defined using virtual cubits are user-independent. Indeed, an object located at a distance of one virtual cubit will be located on the boundary of the user reach for any user and any VE. This allows us to easily generalize experimental results from our testbed to other VEs and to avoid bias due to anthropometrical differences between subjects.

Virtual cubits, however, introduce some problems. If distance to an object is defined in virtual cubits then for users with longer arms the actual position of an object would be further and therefore its visual size will be smaller then for users with shorter arms. This effect can be overcome by defining objects size in terms of *subtended visual angles.*

Sizes, visual angles

We define the size of the objects as their non-occluded *visual size*: the vertical and horizontal angles φ, ϕ which an object occupies in the user's field of view (Fig. 1). Visual angles are user-centered units: two objects with the same visual size would look the same, even if they are located at different distances and have different geometrical sizes.

The advantage of using visual angles is that it permits for the separation of the influence of distance and object size on user performance. Also, similar to virtual cubits, using visual sizes rather than geometrical sizes allows for easy generalization of experimental results, as they do not depend on the VE's implementation.

Occlusion

Occlusion influences user performance in two ways. First, when an object is occluded its visual size diminishes. In this respect selection of an occluded object is the same as selection of a non-occluded one with smaller visible size (Fig. 2). Second, occlusion partially blocks access to the target object and makes it difficult for the user to access it [5], and the close proximity of other objects increases the odds of selecting the wrong object. The definition of occlusion for studies of VR manipulation should reflect both aspects as they both influence user performance.

Because of the dual nature of occlusion, developing a general definition is difficult, so for the purpose of testbed development we elect to use a simplified operational definition of occlusion. This definition is based on the assumption that most of the occlusion cases can be decomposed into five generic cases which are presented in the first row of the Fig. 2 (numbered from 1 to 5).

Consider, for example, occlusion case number seven in Fig. 2. An

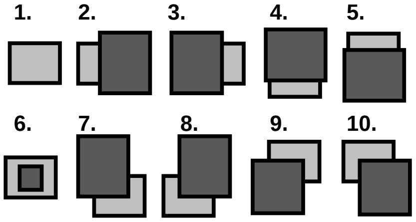

Fig. 2 Cases of occlusion from the user point of view: the dark object is an occluding object.

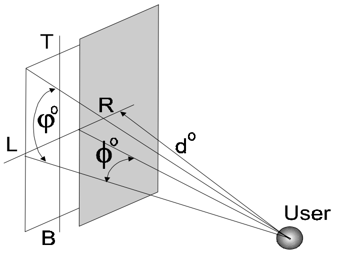

Fig. 3 Occlusion is a visual size of the object's non-occluded part (angles φ^o and ϕ^o), distance d^o and direction of occlusion (left, right, up and down).

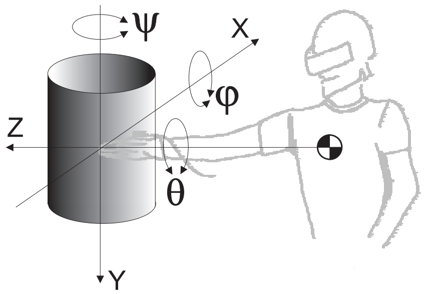

Fig. 4 Orientation of objects is defined in local coordinate system which is defined relatively to the user position.

occluded object can be selected from the right non-occluded side, from the bottom or from the corner. Because the user can choose only one of these three alternatives at a time, they can be studied separately. Furthermore, selection of the object from the right side can be represented by the basic case number three, selection from the bottom by the case number four and the corner selection by either of them.

Occlusion, therefore, is defined here as the size of the object's non-occluded part, distance to the occluding object and the direction of occlusion (Fig. 3). The size of the non-occluded part can be defined either in absolute units as visual size (visual angles φ^0 and ϕ^0 (Fig. 3)) or in relative units as the horizontal and vertical percent of occlusion. The direction of occlusion defines which side of the object is occluded: left, right, top or bottom. The distance to the occluding object is defined in terms of virtual cubits.

Orientation

As shown in Fig. 4, the orientation of an object is defined in terms of angles of rotation around the axis Z going from the user's chest to the object (angle θ) and axis X and Y going in vertical and horizontal directions perpendicular to the axis Z (angles φ and ψ respectively). Rotations about X, Y, and Z thus correspond to pitch, yaw and roll of the object in a coordinate system defined relative to the user.

THE VR MANIPULATION ASSESSMENT TESTBED

In the previous sections we discussed a framework which conceptualizes immersive manipulation, including analysis of immersive manipulation tasks and their characteristics, definition of metrics to describe spatial relationships in VEs, and criteria to evaluate user manipulation performance. In this section we describe a practical implementation of our framework as in the Virtual Reality Manipulation Assessment Testbed (VRMAT) - a flexible test and evaluation environment for systematic assessment of immersive manipulation techniques.

VRMAT design objectives

The optimal test and evaluation environment for studies of immersive manipulation techniques should meet several objectives. First, the testbed should define and implement the *test tasks* and *visual stimuli*. Although the theoretical framework defines basic manipulation tasks and their parameters, the visual representation of stimuli, their spatial configuration, the conditions of test task completion and other test procedures should be implemented and provided to the experimenter by the testbed.

Second, it should *automate* tedious aspects of studies. Studies of interaction techniques require their evaluation in a multitude of different task conditions. Suppose, for example, we are evaluating the effectiveness of several techniques for selection of objects of various sizes at various distances. The testbed should allow the experimenter to define experimental conditions by setting the ranges of object sizes and distances, while the testbed handles the actual configuration of the test environment.

Third, the testbed should help identify and minimize the effects of *nuisance variables* and *confounding factors* [4]. For example, in the first version of the VRMAT, the positions of stimuli were calculated once, before the start of each experimental session, based on the initial position and orientation of the subject in the test environment. During the experimental session the stimuli were presented to the subject one after the other in these precalculated positions. However, when we conducted pilot studies we realized that subjects were changing the position and orientation of their viewpoints *between* trials. Performance during the next task, therefore, was influenced by the position of stimuli for the previous task. To overcome this confound, we now recalculate the positions of the test objects before each trial, using the current position and orientation of the user.

VRMAT test tasks and stimuli

The VRMAT test tasks require subjects to select, position or orient virtual objects (stimuli) while their performance is measured using the performance criteria discussed earlier in the paper. All stimuli in the VRMAT are objects with simple geometrical shapes: spheres, cubes, cylinders and so on. We do not use more elaborate shapes, such as shapes of real world objects, because knowledge about their sizes and proportion in the physical world might affect the subjects' perceptions of their sizes, proportions and distances in the virtual world [21].

The exact configuration of stimuli for the experimental studies is defined by the experimenter using the VRMAT task parameters (independent variables). Table 1 presents test tasks and independent variables currently supported by the VRMAT for the three basic manipulation tasks.

Selection task

The stimuli for the selection task are solitary test objects located in the user's field of view (Fig. 5). The selection task requires participants to select stimuli using the interaction technique under investigation. After being successfully selected, the test object disappears, informing the user that the task was successfully completed and the next stimulus then appears after a fixed (four second) delay.

Positioning task

The positioning test task requires the participant to place a stimulus object on top of a terminal object indicated by a different color (Fig. 6). The positioning of the test objects can be performed using iterative movements, i.e. subjects can pick, move, and release the test object several times until the task is accomplished. The task is completed when a test object is positioned on the terminal within the required precision. After successful positioning, both objects disappear, cueing the subject that the task is finished; then the next test trial is presented to the user.

The shapes for both test and terminal objects are cylinders with

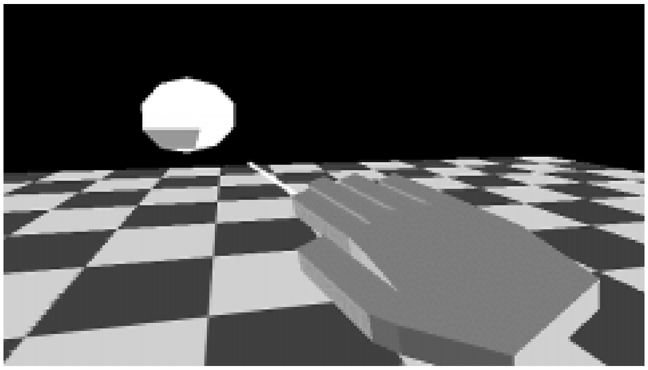

Fig. 5 Selection task: the user selects a solitary test object. The ray-casting technique is being evaluated in this example.

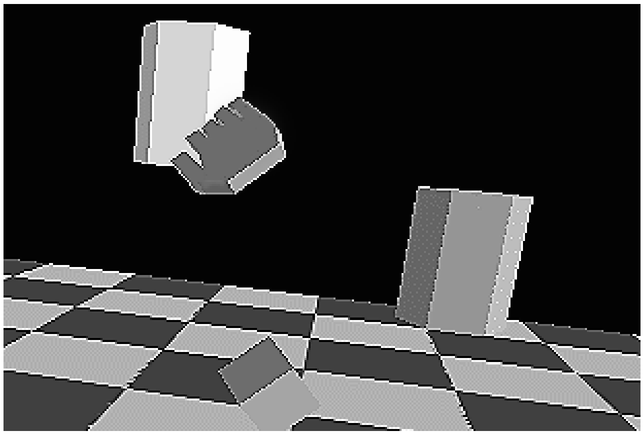

Fig. 6 Position task: the user puts a test object on top of the terminal object, indicated by a different color. The Go-Go interaction technique is being evaluated in this example (the cube in foreground represents the position of the subject's physical hand).

Task	**Independent variable**	**Metric**
Select	distance to target	virtual cubits
	horizontal and vertical directions to target	degrees of arc
	horizontal and vertical size of non-occluded portion	degrees of arc or percentage
	distance to occluding object	virtual cubits
	direction of occlusion	left/right/up/down
	horizontal and vertical visual size of target	degrees of arc
Position	initial distance	virtual cubits
	initial horizontal and vertical directions	degrees of arc
	final distance	virtual cubits
	final horizontal and vertical directions	degrees of arc
	vertical precision	percent of overlap
	horizontal precision	percent of overlap
Orient	distance	virtual cubits
	horizontal and vertical directions	degrees of arc
	initial orientation (3 angles)	degrees of arc
	final orientation (3 angles)	degrees of arc
	accuracy	degrees of arc

Table 1 VRMAT test tasks, their independent variables and units of measurement.

equal radii so as to provide subjects with visual indicator of positioning accuracy.

Orientation task

The orientation test task involves orienting the test object from an initial orientation to a straight-up orientation within a specified angular precision tolerance. The user is cued about final orientation by a reference object which is denoted by a different color. As with the other tasks, the test object disappear after successful orientation.

VRMAT design

The VRMAT environment consists of a checkered ground plane located two virtual cubits below the immersed participant. Stimuli are created dynamically during the experimental sessions according to the task parameters defined by the experimenter in the VRMAT configuration file.

An example of a VRMAT configuration file is presented in Fig. 7. The experimenter defines tasks to be tested, values of independent variables, interaction techniques to be studied for a particular condition and identification numbers of conditions (which are used to match experimental results with task conditions). To define independent variables, the researcher can either assign them a particular value or instruct the VRMAT to randomly sample them from a range of values.

During experiments the researcher uses the workstation console to choose the interaction technique and task to study. The VRMAT parses the configuration file, and builds experimental stimuli using those conditions which satisfy the task and interaction technique entered by the experimenter. After the command to start the session, the VRMAT randomizes the order of the stimuli and presents them, one after the other, with a four-second delay between them, until all task conditions have been presented or until the session is interrupted by the researcher. The experimenter can run as many experimental conditions as necessary within a single experimental session.

There are several conventions supported by the VRMAT:

- Because virtual cubits and visual angles are used to define positions and sizes of test objects, the testbed has to be calibrated to each subject individually before each experimental session. This is done by simply asking the subject to briefly extend her arm. The length of the virtual reach is then automatically derived from the head and hand tracking data. The length of a subject's reach in the VE corresponds to one virtual cubit and is used to translate distances expressed in virtual cubits to platform-dependent computer graphics units, such as points.
- Positions of test objects, their sizes, orientations and all other parameters, as well as light directions, are recalculated depending on the subject's position and viewpoint orientation before each trial. This means that all trials are presented identically for all subjects. This also eliminates the situation of subjects "losing" test objects by changing viewpoint direction between trials.
- The first trial in each session is always a "dummy" trial provided by the VRMAT. Its purpose is to trigger the attention of the user to the start of the experimental session.
- The VRMAT configures stimuli depending on the user's dominant hand: an object appearing on the right for a right-handed user would appear on the left for a left-handed user.

Apparatus

The testbed was implemented using a custom VR software toolkit developed as an extension of the Sense8 World Toolkit VR development tool. An SGI Onyx RE2 graphics workstation, equipped with a Virtual Research VR4 head-mounted display and Polhemus Fastrak 6DOF sensors is currently used. A mouse is used as a button device for selection. The frame update rate is controlled at 15Hz.

```
Selection { ID=1; IT=RayCasting; Dist=0.7;
   Size={6; 6}; Dir={ Random(-15, 15),
   Random(-15, 15) }; Occlusion={0, 0, None}
}
Position { ID=1; IT=GoGo; InitDist = 0.7;
   InitDir={-15, 0}; FinalDist = 3;
   FinalDir={15,0}; Accur={25, 25}
}
```

Fig. 7 Configuration file for the VRMAT defines all task conditions to be studied. There can be as many conditions as necessary.

Evaluation of the VRMAT

Pilot studies have been conducted to evaluate the framework and the VRMAT. Our primary focus was to "shake down" the testbed and develop a baseline for future studies of interaction techniques and testbed parameters. In this section we briefly report some results to illustrate the use of the VRMAT.

Three object manipulation techniques have been evaluated:

- *Plain (virtual) hand:* the user manipulates objects with a virtual hand which position matches position of the user's real hand.
- *Go-Go technique:* this technique uses a non-linear C-D gain to allow the user to extend their virtual reach to manipulate objects located both locally and at the distance [18] (Fig. 6).
- *Virtual ray-casting:* the user interacts with objects using an invisible infinite ray emitting from the virtual hand [17]. A short segment of a the ray is attached to the user's virtual hand as a visual reference (Fig. 5).

We focused on investigating the effect of distance and stimulus size on object selection performance for each interaction technique. For these experiments we defined task conditions in which stimuli were located close, medium and far from the user (0.7, 2.5 and 5 virtual cubits, respectively). The sizes of objects were defined as small, medium and large (4, 6 and 8 degrees of visual angle, respectively). Thus, in total we defined nine conditions for each interaction technique, except for the plain (virtual) hand technique, which supports interaction with only "close" objects. Other VRMAT independent variables were either controlled or randomized to reduce their effects on the results of these studies.

The testbed allows us to define a wide variety of experimental designs. For these studies we used a balanced within-subject (repeated measures) design. Four males and one female served as subjects, and the presentation order of the interaction techniques was counterbalanced across subjects to control for order effects.

Because the VRMAT supports independent manipulation of object size and distance, we can easily investigate how each parameter influences the efficiency of each technique. For example, Fig. 8 and Fig. 9 summarize the effects of distance on selection time performance for the Go-Go and ray-casting techniques, respectively. The notched box plots represent the distribution of the subjects' mean scores around the median for each condition. As shown in Fig. 9, selection time for ray-casting remains essentially the same at different distances (although the variance appears to increase), while for the Go-Go technique selection time appears to be non-linearly affected by distance (Fig. 8). These findings are somewhat counter-intuitive and call for further studies.

Besides evaluation of interaction techniques for various tasks we can also compare interaction techniques across conditions of immersive manipulation. Fig. 10 shows mean selection performance times (and standard deviations) side by side for the two interaction techniques for objects of different sizes (collapsed over object distance). The ray-casting technique appears to be more effective than the Go-Go technique for solitary immersive object selection (within the range of the conditions tested here).

These pilot studies have shown that the VRMAT is an efficient and flexible experimental tool. It allows us to easily accomplish in-depth studies of interaction techniques as well as comparison across conditions of immersive manipulation. The results of experimental studies can be easily transferred from the test environment to practical VE applications because task conditions are defined using user based metrics.

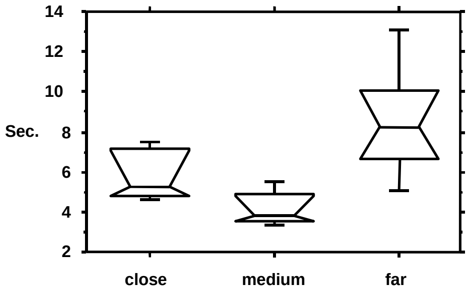

Fig. 8 Box plots for selection time of objects located at various distances using Go-Go technique (collapsed over object size).

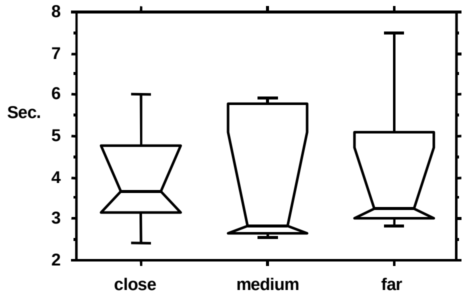

Fig. 9 Box plots for selection time of objects located at various distances using ray-casting technique (collapsed over object size).

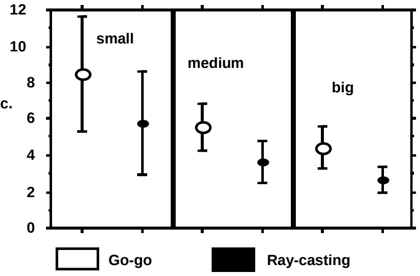

Fig. 10 Mean selection times (with one standard deviation error bars) for objects of different sizes (collapsed over object distance) using Go-Go and ray-casting techniques.

CONCLUSIONS AND FUTURE WORK

Developing VR applications which enable actual work over a period of time requires optimization of the most basic interactions, such as object manipulation. However, to achieve such optimization we need to systematically analyze and understand VR manipulation and to develop tools for experimental assessment of immersive manipulation interfaces.

In this paper we suggest a conceptual framework for immersive direct manipulation, and present a practical implementation of the framework - the VR Manipulation Assessment Testbed. A pilot study performed using the VRMAT evaluated the framework and has shown the feasibility of the suggested approach.

There are several practical implications of this work. First, the framework suggests a systematic and formal view of immersive manipulation which can guide developers in constructing immersive interaction dialogs. Second, results of experiments can be easily applied for design and evaluation of VE applications because the framework and testbed define experimental task conditions in an application-independent way, using user-centered units of measurement. Third, using the experimental testbed, developers can systematically optimize design of existing manipulation interaction techniques so as to achieve the best user performance possible. Fourth, optimization of immersive manipulation techniques is a first step toward developing more generic principles of immersive interaction which better exploit the potential of VR, and may result in new and exciting ways of interacting in VEs.

The long-term goal of this research is a set of guidelines for immersive interaction which would facilitate the development of unified cross-platform VR interface standards and development tools. We will use this newly developed framework and testbed to run a series of studies of immersive interaction techniques. However, we consider the proposed framework and testbed as the very first steps in the direction of formal and systematic assessment of immersive manipulation. Other aspects of immersive manipulation tasks will be formalized and integrated into the testbed, and the current content of the framework will be further refined and evaluated in user studies.

AKNOWLEDGEMENT

This research was partially sponsored by the Air Force Office of Scientific Research (contract #92-NL-225) and a grant from the HIT Lab Virtual Worlds Consortium. The authors want to specifically thank Edward Miller for his comments and suggestions as well as his help with the VRMAT development. We would also like to thank Sisinio Baldis, Jennifer Feyma, Prof. Masahito Hirakawa, Bryn Holmes, Jerry Prothero, Atsuo Yoshitaka, all subjects who participated in the experiments, and the anonymous reviewers for their comments.

REFERENCES

1. Shneiderman, B. Designing the user interface: strategies for effective human-computer interaction. 2 ed. Addison-Wesley, 1992, 573 pp.
2. Foley, D., Wallace, V., Chan, V. The human factors of computer graphics interaction techniques. IEEE Computer Graphics and Applications, N 4, 1984, pp. 13-48
3. Foley, J., Van Dam, A., Feiner, S., Hughes J. Computer graphics, principles and practice. 2nd edition, Addison-Wesley, 1992, pp. 1175
4. Lampton, D. R., Knerr, B. W., Goldberg S. P., Bliss J. M., Moshell J. S., Blau B. The Virtual Environment Performance Assessment Battery (VEPAB): development and evaluation. Presence, vol. 3, 1994, pp. 145-157
5. McCormick, E. Human factors engineering. 3d edition. McGraw-Hill, 1970, pp. 639
6. Norman D. Cognitive engineering. In User centered system design: new perspectives on human-computer interaction. Edited by Norman D. and Draper S., Lawrence Erlbaum Associates, Hillsdale, NJ, 1986, pp. 31-61
7. Randolph, D. E., Sholl, M. J. Object-array structure, frames of reference, and retrieval of spatial knowledge. Journal of Experimental Psychology: Learning, Memory and Cognition, 1995, Vol. 21, N 2, pp. 483-500
8. Grissom, S., Perlman, G. StEP(3D): A portable discount usability evaluation plan for 3D interaction. Ohio State University, Department of Computer Science and Information Science, Technical Report OSU-CISRC-2/93-TR7, 1993
9. Fitts, P. The information capacity of the human motor system in controlling the amplitude of movement. Journal of Experimental Psychology, 47, 1954, pp. 381-391
10. Zhai, S., Milgram, P. Human performance evaluation of manipulation schemes in virtual environments. In the Proc. IEEE VRAIS'93, 1993, pp. 155-161
11. Bowman, D., Hodges, L., An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. To appear in the Proc. of the 1997 Symposium on Interactive 3D Graphics, 1997
12. MacKenzie, I. Input devices and interaction techniques for advanced computing. Virtual Environments and Advanced Interface Design, Edited by Barfield, W. and Furness III, T. Oxford University Press, NY, 1995, pp. 437-470
13. Keppel, G. Design and analysis: a researcher handbook. Prentice-Hall, 3 ed., 1991, pp. 594
14. Mundel, M. Motion and time study. 5 ed., Prentice-Hall, Englewood Cliffs, NJ, 1978, 750 pp.
15. Grandjean, E. Fitting the task to the man. 4 ed., Taylor and Francis, 1988. pp. 363
16. Rory, S. The design of the virtual environments. McGraw Hill, 1996, pp. 274
17. Mine, M. Virtual environment interaction techniques. UNC Chapel Hill Computer Science Tech. Report TR95-018, 1995
18. Poupyrev, I., Billinghurst, M., Weghorst, S., Ichikawa, T. Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR. In Proceedings of UIST '96, 1996, pp. 79-80
19. Philips, M., Bashinski H., Ammerman, H., Fligg, C. A task analytic approach to dialogue design. In Handbook of Human-Computer Interaction, Edited by Helander, M., Elsevir Science Publisher B.V., 1988, pp. 835-857
20. Massimino, M., Sheridan, T., Roseborough, J. One hand tracking in six degrees of freedom. In Proc. IEEE International Conf. on System, Man and Cybernetics, 1989, pp. 498-503
21. Gillam, B. The perception of spatial layout from static optical information. In Perception of Space and Motion, Academic Press, 1995, pp. 23-67
22. Kennedy, K. Reach capability of the USAF population: Phase 1. The outer boundaries of grasping-reach envelopes for the short-sleeved, seated operator, USAF, AMRL, TDR 64-56, 1964
23. Plaisant, C., Carr, D., Shneiderman, B. Image-Browser taxanomy and guidelines for designers. IEEE Software, Vol. 12, N. 2, March 1995, pp. 21-32
24. Bailey, R. Human performance engineering: designing high quality, professional user interfaces for computer products, applications and systems. 3d edition, Prentice Hall PTR, 1996, pp. 636
25. Watson, B., Spaulding, V., Walker, N., Ribarsky W. Evaluation of the effects of frame time variation on VR task performance. IEEE VRAIS'96, 1996, pp. 38-52
26. Zhai, S., Buxton, W., Milgram, P. The "Silk cursor": investigating transparency for 3D target acquisition. In Proc. of CHI'94, 1994, pp. 459-464
27. Herndon, K., van Dam, A., Gleicher, M. The challenges of 3D interaction: a CHI'94 workshop. SIGCHI Bulletin, Vol. 26, N 4, 1994, pp. 36-43
28. Nielsen, J. Usability engineering. Academic Press, 1993. pp. 358
29. Hinckley, K., Pausch, R., Goble, J., Kassel, N. A survey of design issues in spatial input. Proc. ACM UIST'94, April 1994, pp. 213-222
30. Ziegler, J., Fähnrich, K. Direct manipulation. In Handbook of Human-Computer Interaction, Edited by Helander, M., Elsevir Science Publisher B. V., 1988, pp. 123-133
31. Nemire, K. Evaluating visual and auditory enhancements to a virtual object manipulation task. In Proc. Stereoscopic Displays and Virtual Reality Systems III SPIE Vol. 2653, 1996, pp. 249-260
32. Stanney K. Realizing the full potential of virtual reality: human factors issues that could stand in the way. In the Proc. IEEE VRAIS'95 Conference, 1995, pp. 28-34
33. Henry, D., Furness, T. Spatial perception in virtual environments: evaluating an architectural application. In the Proc. IEEE VRAIS'93, 1993, pp. 33-40