FORCE PLUS GRAPHICS IS NOT EQUAL TO VISION PLUS HAPTICS:

TOWARDS *USABLE* HAPTIC ENVIRONMENTS

by

ARTHUR EDWARD KIRKPATRICK

A DISSERTATION

Presented to the Department of Computer and Information Science
and the Graduate School of the University of Oregon
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

August 2000

“Force Plus Graphics Is not Equal to Vision Plus Haptics: Towards *Usable* Haptic Environments,” a dissertation prepared by Arthur Edward Kirkpatrick in partial fulfillment of the requirements for the Doctor of Philosophy degree in the Department of Computer and Information Science. This dissertation has been approved and accepted by:

Dr. Sarah A. Douglas, Chair of the Examining Committee

Date

Committee in charge:
Dr. Sarah A. Douglas, Chair
Dr. Gary Meyer
Dr. Michal Young
Dr. Paul van Donkelaar
Dr. Susan J. Lederman

Accepted by:

Dean of the Graduate School

An Abstract of the Dissertation of

Arthur Edward Kirkpatrick for the degree of Doctor of Philosophy
in the Department of Computer and Information Science
to be taken August 2000

Title: FORCE PLUS GRAPHICS IS NOT EQUAL TO VISION PLUS
HAPTICS: TOWARDS *USABLE* HAPTIC ENVIRONMENTS

Approved: ______________________________
Dr. Sarah A. Douglas

Haptic environments are user interfaces incorporating a haptic display device, most commonly a point force device such as the PHANToM. Lederman and Klatzky's Exploratory Procedures work casts doubt on the usability of such devices, as the human haptic system can only perform rapidly and accurately when full hand contact is used rather than a single finger. However, this work has not been extended to virtual objects displayed by point force devices. Usability of multisensory interfaces is even more complex. How does the addition of a force display change performance in a graphical system?

This dissertation presents two benchmark tasks for human performance with point force displays. Stimuli were generated using Koenderink's shape and curvedness scales for smooth quadric surfaces. The first task, psychophysical magnitude estimation of curvature of paraboloid stimuli, was used to analyze the contribution of haptics to a predominantly visual task. Estimates using vision alone made slightly better discriminations than estimates using both senses, although the effect only approached significance. The second task extended Lederman and Klatzky’s shape recognition work to point force environments. Participants learned to haptically recognize 5 shapes from the shape scale and then identified random instances of those shapes. Despite the simplicity of the shapes, the median time was 23 s and median accuracy 87%. Adding a visual cursor did not appreciably change performance.

These results are comparable to physical shape recognition with a single finger, so the benchmarks are useful metrics of haptic environment performance. They indicate that point force haptic performance is considerably worse than full-hand haptics. The results also indicate that point force haptic perception and vision are not simply additive. The addition of haptics to the first task interfered with performance, while the addition of graphics to the second task had little effect. The poor performance of point force haptics may be due to its unfamiliarity. The experience of point force haptics appears to be sufficiently different from physical haptics that we cannot presume a simple equivalence between display technology and human senses. Users require practice to interpret both the force display and the visual display of the force location.

CURRICULUM VITA

NAME OF AUTHOR: Arthur Edward Kirkpatrick

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

University of Oregon
Maharishi International University
Pennsylvania State University

DEGREES AWARDED:

Doctor of Philosophy in Computer and Information Science,
2000, University of Oregon
Master of Science in Computer Science,
1985, Maharishi International University
Bachelor of Science in Mathematics, 1977, Pennsylvania State University

AWARDS AND HONORS:

University of Oregon Doctoral Fellowship, 1998-1999

GRANTS:

Intel Corporation, 1997, PHANToM Haptic Interface and Pentium Computer

ACKNOWLEDGEMENTS

I am deeply grateful to the organizations that provided financial support for this work. Intel Corporation donated a PHANToM and Pentium computer to our lab. Their generosity and no-strings-attached enthusiasm for a contribution to basic science is an exemplary case of industrial-academic cooperation. The Zebrafish Information Network (ZFIN) provided me a research assistantship for four years. A University of Oregon Doctoral Fellowship afforded me the luxury of working exclusively on my research for a year. This focus was a blessing at a difficult time in the work.

Many individuals contributed to this project, in a myriad of ways. Odile Wolf asked the question that got me to return to graduate school. Betty Lockwood, Jan Saunders, Cheri Smith, and Star Holberg took able care of me once I was there. Steven Keele provided a patient introduction to the psychology of human motor control. Paul van Donkelaar suggested safe paths through the swamp of experimental design. My long association with Gary Meyer taught me about rendering algorithms and human color perception. Michal Young advised me about both software engineering and the general research process of publication and community service. My thinking about haptics was considerably refined through many long conversations with Susan Lederman. The comments of members of the U of O HCI group improved my expression of my ideas once I had them.

Most of all, Sarah Douglas pushed, pulled, prodded, and cajoled me into going further than I had ever imagined myself capable. The best thanks I can offer her is to demonstrate the same concerns for value-based science and attention to my students in my work that she has done in hers.

DEDICATION

Dedicated to my parents, Lawrence and Joyce Kirkpatrick.

TABLE OF CONTENTS

Chapter Page

APPENDIX

LIST OF FIGURES

Figure Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

Computer users spend their lives in two different worlds. The first is the familiar world of *physical environments*: the places we walk through, the objects we pick up, move around, and even alter by bending or tearing. This is the environment that we have spent a lifetime learning to function within, and also the environment in which all animals on earth have evolved. This huge experience base, the physical and cognitive capabilities developed in the human biology by millions of years of evolution, and the decades of personal experience that every adult human has, makes us extremely well-suited to these environments. We lift cups to our mouths, grab pencils on our desks, throw balls to first base, and inspect an emerald in the sunlight, all with little thought or effort.

Although easy in some ways, physical environments can also be annoying. The needle may be too small to thread or the lid may be too tight to get off the jar. It takes time and energy to move things or persons from one place to another, and the further the distance the greater the time and effort. We drop objects, they break, and no amount of glue can ever make them quite as good as new. The pencil line on the paper can never be entirely erased.

The second world, the *virtual environments* presented by computer technology, is nearly the opposite of the world of physical environments. Things are strangely easier here. Erasure leaves a pristine page, there is no gravity inexorably pulling valued possessions down to their doom, and the peg always fits in its slot. Time, distance, size, and weight are nearly effaced. We can communicate with people anywhere in the world, design jumbo jets, and manipulate individual atoms on a grid (Taylor et al., 1993). It seems almost magical.

The virtual magic has its dark side, though. While the above facilities may be available in principle, the commands for making them happen are far from obvious. There will often be no apparent procedure to invoke simple operations. Once a command has been issued, its consequences can be quite different from what the user expected—a common difficulty with magic. It can be difficult to even determine what the consequences were. There may be magic here, but accessing it requires incantations beyond the ken of most mortals.

These two worlds show a striking complementarity. On the one hand, simple operations require little thought in physical environments but the physicality of objects can make the execution of those operations cumbersome. On the other hand, the insubstantiality of "objects" in a virtual environment places them outside the limitations of size and space, but the *selection* of operations is often cumbersome. Why not blend the mundane with the magical and get the best of both worlds? Mix the operational simplicity of the physical with the insubstantiality of the virtual and get a hybrid environment where it is both easy to know what to do *and* easy to execute once you have decided to do it.

This dissertation explores one approach to such a hybrid, *haptic environments*. Haptic environments are virtual environments featuring technologies for general purpose, computer-controlled force display. Just as screens are general-purpose displays for images and speakers are general-purpose displays for sounds, so force display devices can create a wide range of tactual and kinesthetic stimuli: texture, shape, friction, viscosity, and other effects. While many different forms of haptic display have been designed, the dominant commercially produced design today is the *point force display* (PFD). Point force displays provide a single point of resistance to human movement. While restricting the resistance to a single point results in a desirable design outcome of high resolution forces with low device inertia, the restricted haptic experience provided by such displays has significant impacts on human perception. A large portion of this dissertation is concerned with exploring and quantifying the consequences of restricting haptic perception to a point force.

While the hardware technology for haptic environments is now well-established and generally available, the environments built upon that technology are much less well understood. There is a paucity of production applications of these devices. Research to date has found few examples of measurable performance improvements from incorporation of point force displays. This dissertation specifically focuses on the relationship between point force and graphics displays in a haptic environment. How does the combination of these display technologies change human performance? How does it change the software architectures for virtual environments? Specifically, it considers the following questions:

1. What are good protocols for assessing the performance of haptic environments?

2. How effective is a point force device at haptic only shape recognition?

3. Is haptic performance with a point force device comparable to performance with the whole hand?

4. How wide a range of individual differences in performance do we find with point force devices? Are there cognitive factors that account for those differences in performance?

5. Does the addition of point force haptics to vision improve performance on a geometric task in display configurations featuring reduced visual depth cues?

6. Does the addition of vision to point force haptics improve performance on recognition of simple shapes?

7. More generally, does the addition of one display modality to another improve performance?

8. Is perception of virtual environments simple and effortless or are there circumstances under which it requires learning and places attentional demands?

9. Do attentional or motor control demands create negative sensory interactions, where the addition of a second display modality reduces performance of the first?

10. What are the mechanisms of point force curvature perception?

11. What are the mechanisms of point force shape recognition?

12. Does the addition of point force display to a graphical system require changes to existing user interface software architectures?

I answer these questions using a combination of exploratory empirical studies and a comparative analysis. The exploratory studies consist of controlled experiments to evaluate the overall performance of several interaction techniques in point force haptic environments and the influence of individual factors upon that performance. The results of these studies provide initial estimates of the performance of point force haptic haptics for the assessment of geometric properties of objects in virtual environments. These studies also provide first estimates of the effects of various parameters on that performance.

I emphasize the initial nature of these results. There is very little prior data on performance of point force haptics. While there is a large body of existing data on human haptic performance, it describes performance of haptics with the entire hand. The use of the entire hand permits simultaneous apprehension of multiple properties of an object at multiple points on the object. I call this sensory procedure a *broad exploratory procedure* and provide a detailed definition in Chapter II. Point force haptic perception does not permit broad procedures. Indeed, even contact using a single fingerpad provides more information than contact using a point force device. The reduction of sensory experience from multiple points to a single point has dramatic effects on human haptic performance, and the studies of haptic perception with multiple contact points can only provide partial guidance for human performance with point force haptics.

There is one study (Klatzky, Loomis, Lederman, Wake & Fujita, 1993) that has looked in detail at the relative performance of the haptic system under varying levels of properties sensed and number of simultaneous points of contact. While the data in this study provide a crucial starting point for point force haptic performance, that starting point is only rough. The tasks, the exploratory procedures, the properties accessible to the sensory system, and the amount of information simultaneously accessible are all different in point force haptics than haptics using direct contact by any part of the hand surface.

Given such limited prior data, and the dearth of widely accepted tasks for evaluating these environments, I believe that an exploratory approach is most appropriate. While the experiments in this dissertation feature some controlled independent variables and a classical analysis of their effects, the bulk of the analyses is more correlational and post hoc. The intent of these analyses is not to prove a particular model, but to suggest directions for the next round of data gathering, ultimately leading to the construction of theories of point force haptic perception.

The second methodology used in this study is a comparative analysis. In addition to a lack of data on point force haptic performance, there is a corresponding lack of data on software architectures for constructing haptic environments. Software design is not amenable to controlled experimentation—what, exactly, is being controlled?—and so I adopt a strategy of comparing the different extant approaches to constructing such architectures.

Virtual environments are likely to include acoustic displays in addition to graphics and force displays. There is an active research community concerned with auditory displays (International Community for Auditory Display, 2000) and interactions between acoustic and force displays have already been demonstrated (DiFranco, Beauregard & Srinivasan, 1997). Nonetheless, for the purposes of this dissertation I focus exclusively on systems with only force and graphical displays. Extending this framework to include acoustic displays is surely nontrivial, and so I do not consider acoustic displays in this work.

The Designer, Design Resources, the User, and Perceptual Resources

A virtual environment is a communication from the designer to the user. In this dissertation, I use the term “the designer” to refer collectively to the collaborative team of graphic designers, user interface designers, and software architects that create a software application. A central tenet of this dissertation is that perception of a virtual environment is an act of interpretation. Users do not merely passively receive a predetermined and fixed meaning from a program, but actively construct meaning using the resources available in the program displays. Thus the designer can never presume that the user will interpret a graphical or force stimulus in the way the designer intended.

Because of this, I use a terminology that carefully separates the display elements constructed by the designer from the percepts constructed by the user.

I define *perception* as a process where an external *stimulus* gives rise to a *sensation* in the user which the user in turn interprets based upon past experience and the aggregate of all current sensations to form a *percept*. From the Oxford English Dictionary, 2d ed.:

stimulus (def. 3c) Any specific change in physical energy or an event (whether internal or external) which creates a nerve impulse and gives rise to a reaction.

sensation (def. 1a) Now commonly in more precise use, restricted to the subjective element in any operation of one of the senses, a physical "feeling" considered apart from the resulting "perception" of an object.

perception (def. 6) The action of the mind by which it refers its sensations to an external object as their cause.

percept (def. 2) The general mental product or result of perceiving as distinguished from the action.

The above definitions of sensation and percept are functional, not physiological. The neurophysiology underlying the experiences of sensation and percept may be peripheral or cortical and the boundaries between sensation and percept are deliberately vague. The important element of the distinction is that "primitive" sensations from different senses can be combined to form a single percept.

The designer controls only the first stage of this process, the displays, the source of the stimuli. More precisely, the designer controls the numerical values loaded into the control registers of the display device. This is only partially determinant of the energy fields that reach the user's sensory receptors. For example, the designer may specify that a certain value is put into the frame buffer of a cathode ray tube display, which in turn determines the intensity of the electron beams radiating upon a certain point of the display, but the actual light reaching the user's retina is the product of those beams, the age of the monitor, the calibration of the monitor, the type of the monitor phosphor, the ambient room light, the degree of glare on the screen, the radius of the user's pupil, and other factors. The transduction of the resulting light energy into sensation by the early stages of the users's visual system is influenced by the adaptive state of the system, any color deficiency in the system, whether the energy falls within the foveal or peripheral eye region, and aftereffects from the images viewed immediately before this one. In summary, the register values specified by the designer contribute to the stimulus received by the user but do not completely determine it and the transduction of that stimulus into sensation is idiosyncratic to the user.

The process of constructing a percept from sensations is even more complex than the process of transduction. Does the user interpret the sensation as a flat or curved region or perhaps a sharp edge? Which direction is the curvature? Given simultaneous stimulation of the visual photoreceptors and the haptic mechanoreceptors, does the user fuse these sensations into a single percept or distinct percepts? This process too is largely out of the influence of the designer.

The user of a virtual environment is performing a task and so perception is directed towards completion of that task. Prior experience has given the user skills in interpreting sensations to form specific percepts. The user has developed procedures for moving the hands in haptic perception and for using tactile and kinesthetic feedback to guide manipulations. These skills will be described in detail in the next chapter. For now the important point is that these skills will vary from user to user and that there are many different ways of accomplishing the same task. An environment, whether physical or virtual, is a collection of *resources* that the user will draw on to accomplish the task. Thus the designer cannot presume that any particular approach will be used by any user.

The *design resources* available to the designer are the graphical and force stimuli created by the displays. The *perceptual resources* of the user are the user's interpretations of the sensations produced by these displays, based upon the needs of the task and the user's previous experience. The process of interpretation is comprised of the processes of sensory experience, vision and haptics. The relationship between design resources and perceptual resources is indirect. The designer creates graphical and force displays which the user interprets and uses as perceptual resources. If the displays are well-matched to a user's prior experience and aptitudes, there will be sufficient perceptual resources to support good task performance. If the displays do not match a user's prior experience, task performance will be poor. The title of this dissertation is the fundamental principle that must guide designers of haptic environments. Force plus graphics is not equal to vision plus haptics, because the latter are interpretive acts over which the designer has only indirect influence.

This indirect relationship has extensive implications for both the process of design and research aimed at producing a base of scientific results to inform design. Designers must provide interaction techniques—mappings from transduced human motion to visual, auditory, and haptic display—offering a multiplicity of methods for accomplishing the same goal. Researchers must identify the perceptual resources used by most of the population for key tasks, so that designers can at least provide those. Both designers and researchers must pay close attention not only to the interpretation of individual displays as sensations, but to the combination of visual and haptic sensations into a single percept. Experiments 1 and 2 of this dissertation (described in Chapter III and Chapter IV) exemplify the difficulty of this process. Even though these experiments were designed to explicitly account for differing use of perceptual resources, the results were unexpected because the participants in fact interpreted the stimuli in an unanticipated way.

The difficulty of creating good interaction techniques for these environments has implications for software architectures as well. Designers will need programming environments in which they can readily construct different interaction techniques as part of the iterative design process. Once good combinations of perceptual resources are identified, they should be packaged as reusable interaction techniques, just as successful two-dimensional interaction techniques are currently disseminated as widget toolkits.

The Structure of This Dissertation

The remainder of the dissertation advances the above argument in detail. Chapter II presents a detailed analysis of point force environments in the context of current virtual environments. It begins with a description of “situated skills”, the everyday skills humans use interacting with physical environments. To take advantage of these skills in virtual environments, we need more sophisticated input and output technologies. The next section describes the technologies for facilitating spatial interactions with a virtual environment. I argue that all these technologies can only produce a partial simulacrum of interactions with physical objects. Thus virtual environment designers will have to carefully choose an approach that provides the most appropriate perceptual resources for their particular task. I then describe two such approaches. Graspable Environments (GrEs) use specialized input devices for specific tasks. The devices afford a high level of situated skills for their tasks, but are limited in the range of tasks they support. By contrast, Haptic Environments (HEs) use general purpose programmable force displays. The general purpose nature of the devices gives them lower performance than their graspable counterparts but they offer a much wider range of possible applications. The rest of this dissertation examines the implications of haptic environments, specifically focusing on the use of Point Force Environments (PFEs), haptic environments with point force displays.

The next two chapters present experiments that explore the implications of this ecological approach. The experiments examine the detailed relationship between vision and haptics in two tasks related to the perception of geometric properties of simulated objects. Experiment 1 (Chapter III) compared the perception of curvature using vision alone and vision plus haptics. Participants appear to have found it more difficult to estimate curvature using the combination of vision and point force haptics than using vision alone. Experiment 2 (Chapter IV) measured the performance of participants using point force haptics for a simple shape recognition task. The performance was extremely slow, although only about two to three times worse than the performance predicted based upon somewhat comparable experiments using physical objects. The results of Experiment 2 also show that the presence of a visual cursor provided no improvement in performance over purely haptic perception. I suggest that this latter result is due to the unfamiliarity (and hence uninterpretability) of the interaction technique, and present some data on extended practice that suggests some participants experience performance improvements with the cursor once they have sufficient practice.

Chapter V considers software architectures for haptic environments. Given the difficulty of designing good haptic interaction techniques, designers must have tools that allow them to try new techniques and reuse proven techniques. I argue that a haptic environment contains several different program loops, such as the graphics and force rendering update loops, each of which has different structural and performance requirements. Current approaches to virtual environment architectures are organized around one of these loops, an approach that succeeds only so long as that particular loop dominates both human performance and the organization of the software. However, production haptic environments will probably not be amenable to organization around a single loop. After analyzing the two current approaches to such designs, I conclude that we do not yet know how to structure a general architecture for haptic environments. Chapter V ends with a consideration of constraint-based notations for programming haptic interaction techniques.

Chapter VI summarizes and integrates the results of the preceding chapters. The dissertation concludes with a research agenda for the haptic environments, based upon results of Experiments 1 and 2.

A Note on the Diversity of Physical Abilities

In this thesis I routinely make assumptions about the physical capabilities of “the users”. However, there are important classes of user populations that lack one or more of the capabilities I presume. Children, the elderly, and others lack precise motor skills. Some individuals lack the use of one or both hands. Many individuals have some form of visual impairment. My intent in this work is not to propose designs that will only work for “fully abled” individuals. I phrase my arguments in terms of an idealized body type simply because I must discuss the physicality of human experience in terms of some specific physical form. I believe the design approach I advocate in this work, taking into account the cues used by various individuals in the intended user population, can also be applied to produce specialized designs that are better suited to the needs of differently abled populations than those design approaches we have seen to date.

CHAPTER II

A FRAMEWORK FOR DESCRIPTION OF HAPTIC ENVIRONMENTS

Introduction

This chapter lays out a framework for thinking about haptic environments—what "haptic skills" might consist of, what properties might make haptic environments less usable than physical environments, and how we might design more usable haptic environments. The novel part of this description is its focus on virtual environments featuring continuous-valued, spatially-situated cues for perception and motor control—the cues we use every day in our interactions with our physical environment. Haptic environments offer the possibility of incorporating more of the perceptual and motor skills we use in our daily life into our interactions with computer systems. Spatial manipulations include moving objects to a new location, reorienting them for a better view, squeezing, stroking, and so on. While there is an established literature on non-haptic virtual environments incorporating some of these features, the presence of haptic displays changes both the kinds of things we can do and the kinds of things we might want to do in such environments.

In addition to spatial manipulations, haptic environments will also sometimes require the choice of one item from a list of discrete items—a command from a menu, a font from a family, a tool from a palette. The discrete choices offered in a haptic environment can be implemented using current techniques, extensions to them such as 3-D menus (e.g., Deering, 1995), or creative alternatives such as body-relative locations in space (Mine, Brooks & Sequin, 1997). The issues in these interaction techniques are very different from those of spatial manipulation and so I do not consider them in this thesis. I focus on the issues raised by spatial manipulation.

These latter issues have not been important in user interface styles to date. Although the ubiquitous Windows, Icons, Menus, and Pointing (WIMP) interface style makes extensive use of pointing, a continuous-valued perceptual / motor skill, the perceptual and motor control issues for this skill are specialized and an impressive and capable body of scientific and engineering data is available for designers (e.g., Douglas, Kirkpatrick & MacKenzie, 1999; Douglas & Mithal, 1997). Haptic environments introduce a much wider range of perceptual and motor control issues. This is not surprising. Haptic environments are *defined* by the introduction of a certain kind of perceptual experience, and so the relationship between haptic perception and action, each enabling and influencing the other, as well as their relationship to the goals of the human using the environment, must inevitably be central to any theory of haptic environment design.

For adults, spatial manipulations in physical environments are highly practiced and performed with little to no conscious thought. Haptic environments hold the tantalizing prospect of improving the naturalness and decreasing the cognitive burden of our interactions with computers by incorporating more of these practiced spatial interactions. These benefits can only be realized to the extent that the skills users have acquired interacting with physical environments transfer to interactions with haptic environments.

To understand the possibility of that transfer better, I begin with an analysis of what spatial manipulation skills are and how they are crucially related to the property that human beings are physically situated in an environment. I then revisit two established theories (Norman, 1988; Shneiderman, 1983) of usability for computer interfaces. These theories recommend incorporating principles of spatial manipulation into computer interfaces. I then analyze different hardware configurations for virtual environments, determining how much they each support these manipulation skills. I conclude that every configuration is more limited than physical environments. We cannot construct a haptic environment that permits complete and effortless transfer of spatial manipulation skills from physical environments.

I then consider two approaches to incorporating some fraction of spatial manipulations into virtual environments. The first approach, Graspable Environments (GrEs), affords physical manipulations through the use of specialized input devices. These interface styles succeed in incorporating a rich set of skills directly into human-computer interaction, but the movements they support are limited by the specialized input devices used. Haptic environments take a different approach, using general-purpose display devices and supporting nearly arbitrary motions, but at the expense of reduced affordances and ease of orientation of objects of interest. In this thesis I focus on the haptic environment approach, emphasizing those using the point force style of haptic display .

A note on terminology: In this chapter, I will be drawing together results from the mathematics of perspective projection, psychophysical experiments, and theories of virtual environment design. Each of these fields uses a different term for the human participant. While the differences partly stem from different historical backgrounds, these terms also reflect different assumed levels of human involvement. In perspective projection, mathematicians speak of a “viewer”, who consists of a view vector and nothing more. Perceptual psychologists speak of an “observer”, a more complete model that has the complex nonlinearities and inconsistencies of human perception but who is unusually earnest compared to humans outside the laboratory, willing to perform complex perceptual tasks under difficult conditions and actively seeking to complete a rather abstract task. Researchers in human-computer interaction speak of a “user”, a fidgety, inattentive, impatient creature who is looking for the most efficient means to accomplish a goal and who will quickly shift strategies from the computer to pencil and paper if that will get the job done more quickly.

To reflect the different assumptions underlying these fields, when reporting results I have used the term appropriate to the field. While this gives some paragraphs an unusual sound, switching from one term to another, I think it is important to maintain an awareness of the different assumptions upon which the result is based.

Environments—Physical, Virtual, and Haptic

The experience of interacting with a computer has often been compared to moving about and interacting with a world. The term “virtual world” dates at least back to Ivan Sutherland’s famous 1965 paper (quoted in Brooks, 1988, p. 1). The terms “virtual world”, “virtual environment”, and “virtual reality” have become so widely used that their underlying phenomenological assumptions (how humans become consciously aware of the properties of objects and the environment) and ontological assumptions (the fundamental objects and categories assumed to comprise the world) are unquestioned. A simple equivalence is presumed between the physical world and “virtual worlds”. Since a core question in this chapter is the extent to which this equivalence exists, I must adopt a vocabulary that allows me to clearly express the distinctions between the two. Only given those distinctions can I begin assessing the similarities.

I begin with a definition of “environment” that can accommodate both physical and virtual environments without presuming unwarranted similarities between the two.

As a consequence, it is rather abstract. I define an *environment* phenomenologically, as a set of external stimuli that change in response to human movement. Some movements result in global changes, as for example the change in visual and acoustic stimuli when we walk about a room. Other movements change localized stimulus regions, as for example the specific change in the visual field when we lift a cup. An environment also has global state, local state, and observer state. The global state includes things like the level of illumination in the room. Local state is associated with the localized stimulus regions. The observer state includes the current location and point of view of the observer. It is the observer state that places the observer *within* the environment. I define *manipulation* as the actual physical movement required to effect a change in an environment.

A *physical environment* is a space, inside a structure or outdoors, with the furniture, tools, plants, and so forth contained in that space. For physical environments, the above abstract concepts map readily to common experience. The global stimuli and state correspond to the space in which the observer is situated, the localized regions of stimulus and state are the objects in the space, and the observer state is the physical location and direction of gaze of the observer. Manipulations consist of moving or altering objects or moving the observer. Human manipulation produces consistent visual, auditory, and haptic responses from physical objects because the physical laws governing the structure of the object determine its response in each modality.

A *virtual environment* is a combination of hardware and software with which a human interacts. The mapping of the above definition of environment to virtual environments is more problematic. First of all, there is no longer a direct link between the observer's movement and the change in stimuli. Instead, the observer's movement is transduced by some device, a change in the program state is computed by an algorithm, and the displays are updated. This is the implementation of the interaction technique. Secondly, while the objects in the physical environment have physical existence, the “objects” in a virtual environment are illusions. Separate algorithms compute the graphical and force responses to manipulation. Excess response time, coarse discretization, discrepant graphical and force displays, or instability of control algorithms for haptic displays can all ruin the illusion. Given the ephemeral nature of these responses, I hesitate to call them “objects” in the same sense as the objects of the physical world. They only become objects when the observer interprets the several sensations as a percept of “object”.

Many different kinds of virtual environments exist, each with different definitions of “object” and space. The most common virtual environments today are WIMP interfaces. In these environments, the objects are widgets such as scroll bars or buttons and the movements are transduced by a keyboard and a two-dimensional pointing device such as a mouse. The observer state is not clearly defined. The most useful metaphor is to think of the user as having a completely stationary point of view and using direct dragging and scrollbars (an indirect form of dragging) to move objects within that point of view.

I define a *haptic environment* as a virtual environment that includes a programmable display for force or tactile cues. The range of such displays is extremely broad. While many of my arguments in this thesis apply to all haptic environments, some are based upon the specific structure of point force haptic displays (PFDs). As its name implies, a PFD can display forces at a single mathematical point in its working volume. Virtually all current commercial haptic displays are two or three dimensional PFDs, including the Feelit Mouse (Immersion Corporation, 1999), the PenCAT and MouseCAT (Haptic Technologies, 1999), the PHANToM (SensAble Technologies, 1999), and force feedback joysticks. This is likely to be the dominant type of haptic display for the near future. Its restriction to a single point of force creates significant limits on human performance and so I focus on it to provide a careful analysis of these limits. I call any haptic environment whose haptic display device is a point force display a *point force environment*.

Virtual environments with haptic devices will obviously be different from WIMP interfaces. In the next section, I will describe the kind of virtual environment with which this thesis is primarily concerned.

Workspace Environments

Humans act in many different kinds of environments, and the properties of those environments and the activities we perform in them vary widely. For the purposes of this thesis, I focus on a specific kind of environment, a *workspace*. This kind of environment both typifies the form haptic environments are likely to take and is amenable to study by controlled experiment.

I define a workspace as a volume of space up to roughly two meters in height and width and one meter deep, extending in front of the user, containing the region of space the user can reach without leaving their chair or turning around. The space contains objects of a size and weight that can be comfortably held in the hand or hands. Some of the objects are tools, others are materials. The space will often contain surfaces for organizing the objects and perhaps for supporting the materials as they are being worked upon. The basic operations that the user of a work space performs are (1) to pick up and analyze the materials and (2) to modify them. The user may use the tools to perform the analysis and modification. Physical examples of workspaces include a woodworker's bench and tool rack, an artist's easel and palette, and a dentist's chair and tools. Examples of virtual implementations of workspaces are the WebBook (Card, Robertson & York, 1996) and the Virtual Table (Schmalstieg, Encarnacao & Szalavari, 1999).

The restriction to a volume reachable from the user's chair (or place where they are standing) is a crucial simplification. While the user may move their head and upper torso, they do not change their location. Thus they do not have to perform locomotion and wayfinding. Wayfinding is a complex behavior and requires careful design to be properly supported in virtual environments (see, for example, Darken & Sibert, 1996). As we shall see, simply supporting the basic operations of picking up and feeling objects is difficult in virtual environments. For now, it seems prudent to avoid adding the extra complications engendered by wayfinding.

A second consequence of my focus on workspaces is that they are task-oriented. The user performs some operation on the material. The utility of workspaces is best evaluated by how quickly and comfortably the intended user population can perform the task. For this thesis, I am not considering environments used for entertainment or communication. While there have been intriguing haptic environments designed for social interaction (Dodge, 1997; Fogg, Cutler, Arnold & Eisbach, 1998) and games (Ishii, Wisneski, Orbanes, Chun & Paradiso, 1999; Johnson, Wilson, Blumberg, Kline & Bobick, 1999), these are outside the scope of this thesis because the criteria of success for such environments are not efficiency, accuracy, and comfort.

Human Situated Perceptual and Motor Skills

With the above definitions of various environments in hand, I now consider the nature of human physical skills. Human perceptual and motor faculties in a physical environment are diverse, flexible, and highly skilled. The sources of perceptual information are rich, redundant, complementary, and synergistic. Human perceptual systems combine a multiplicity of these environmental cues to form a far more precise percept than could be inferred from any single cue. If one cue is unavailable due to injury or unusual environmental conditions, the perceptual systems can often combine other cues to produce a functionally equivalent percept. Human motor systems are also highly flexible, permitting a far broader range of movements than the minimal number required for a three-dimensional environment (the "degrees of freedom problem", Rosenbaum, 1991, pp. 5–7).

A lifetime of experience with physical environments has provided adult[1] humans with important skills in those environments. They have learned a rich repertoire of manipulations, they are very good at determining what manipulations are possible in a given situation, and they can mutually orient objects of interest and their point of view. Furthermore, practice is enjoyable and rewarding because learning is smooth and continues to produce performance enhancements over extended periods of time. Finally, exercise of these physical skills is enjoyable. I call this extensive basis of skills *situated skills*.

This means there is a base of sophisticated skills that designers of physical objects can rely upon in nearly the entire human population. These skills also represent a po-

---

[1] These skills develop over time in childhood and adolescence. This developmental process is beyond the scope of this thesis, and so I restrict myself to adults. The basic principles of this work can be applied to environments for children and adolescents by designing for the appropriate skill level for a given group.

tentially important resource for designers of virtual environments. However, accessing these skills is more difficult in virtual environments. Designers of physical environments receive the situated skills “for free”—they are a consequence of physics and the embedding of objects in the same environment as the user. Designers of virtual environments, on the other hand are not only designing an object, they are designing the systems by which the user will perceive that object and the motor systems by which the user will manipulate that object. This requires a far more comprehensive design viewpoint.

As a basis for the design of virtual environments (particularly point force environments), in this section I briefly describe the situated skills. Drawing on the definition of environment in the previous section, I separate these skills into skills of movement and skills of perceptual processing of environmental stimuli. This separation is only for expository purposes. Movement is a fundamental part of perception, and perception in turn guides movement. However, separating the two is useful because it corresponds to the hardware of input transduction and output display I will consider in a later section.

Situated Skills of Movement

I defined an “environment” as something that changes in response to human movement. There are four broad categories of movements, each producing a change in a corresponding part of the environment’s state. Locomotion changes the state of observer location. Point of view movement changes the observer’s point of view state. Manipulatory procedures change the position or form of an object, altering the localized state associated with that object. Exploratory procedures, the final category of movement, provide the human with information about localized regions of the environment. I will consider each of these categories in turn.

I will not be considering *locomotion* in this thesis. While it is of great importance in physical environments and also of importance in the kind of virtual environment typically termed “virtual reality”, it also introduces great complexities into the design of the environment. In this thesis I focus on virtual implementations of workspace environments, where no locomotion is performed.

Point of view movement requires the integrated use of several complex sets of muscles. While keeping the location constant, the point of view can be modified by moving the upper torso, the neck and head, and the eyes. Smaller muscles within the eyes adjust the focal point of the lenses and the pupil size. Adults are highly practiced at the coordinated use of these muscle systems to adjust their point of view to meet their perceptual needs.

Point of view movement requires that the object of observation be located in the same space as the observer. The observer moves their point of view from side to side to see different sides of the object, forward to view parts of the object in greater detail, and back to shrink the object to a smaller portion of the visual field and thereby see the whole. All of these movements are done with little or no conscious effort. Eye movements have been extensively categorized (Rosenbaum, 1991, Chap. 5). I do not describe them further here, because these movements do not need to be transduced or modeled to produce an effective virtual environment[2].

In addition to providing changing the point of view for a more efficacious view of the object, these movements provide information about the depth of the object and its relative depth within the environment (Goldstein, 1996, Chap. 6). Movement of the head produces *head-motion parallax*, visible difference in the relative rates of movement

---

[2] While eye-tracking is not required for virtual environments, it can be used to optimize generation of visual displays (e.g., Watson, Walker, Hodges & Worden, 1997).

of near and far surfaces. *Vergence*, the motion of the two eyes onto a common point of focus, and *accommodation*, the focusing of the lenses, provide further depth cues. The direct contribution of these two cues to depth perception may be small, but there is evidence that they contribute to the interpretation of other cues (Buckley & Frisby, 1993; Reinhardt-Rutland, 1996).

Manipulatory procedures (MPs) are an extremely broad class of hand movements, many of which are task-specific. They can be performed either to alter the shape of an object (bending, stretching, and so on) or to reorient an object for better perception by the visual or haptic systems. I am unaware of any taxonomy of the manipulatory movements of the hands, but there are several taxonomies of static grasp shapes. Klatzky, Lederman, Pellegrino, Doherty, and McClosky (1990) define four classes of shape the hands assume when manipulating an object, while Burdea (1996, p. 23) provides a more elaborate taxonomy of grasp shapes.

Bimanual manipulation adds another layer of complexity. A key aspect of human manipulation is the asymmetric use of both hands. Guiard (1987) has proposed the Kinematic Chain model of asymmetric bimanual action. This model emphasizes that for many tasks, such as writing, the two hands have distinctly different roles. The nondominant hand is used to hold and orient the object of interest and establishes a base coordinate system. The dominant hand works upon the object of interest with reference to the coordinate system provided by the other hand. Hinckley, Pausch, Proffitt, Patten, and Kassell (1997) experimentally demonstrated this fundamental asymmetry in an insertion task: Participants were 23% faster when they held the target in their nondominant hand and the tool in their dominant hand than when the hand positions were reversed. Guiard's theory has also informed the design of bimanual computer interfaces (e.g., Hinckley, Pausch, Proffitt & Kassell, 1998; Kurtenbach, Fitzmaurice, Baudel & Buxton, 1997; Leganchuk, Zhai & Buxton, 1998).

The specific details of manipulatory procedures are of little concern in this thesis. It is sufficient to know that human manipulatory capabilities are extensive, well-practiced, and rely on the enormous flexibility built in to the structure of the human hand and the even greater flexibility provided by simultaneous use of two hands.

Exploratory procedures (EPs) are the most well-described category of hand movements[3]. Contemporary quantitative study of the haptic system began with an object-recognition task studied by Klatzky, Lederman, and Metzger (1985). They found that blindfolded participants could identify common objects in just a few seconds. In a later paper, Lederman and Klatzky (1987) identified six stereotypical hand movements, forming the basic set of exploratory procedures. Each EP is visibly distinct from the others and is associated with the apprehension of a specific object attribute. The *lateral motion* EP is used to apprehend texture, the *pressure* EP is used to apprehend hardness, the *static contact* EP is used to apprehend temperature, the *unsupported holding* EP is used to apprehend weight, the *enclosure* EP is used to apprehend global shape and volume, and the *contour following* EP is used to apprehend exact shape.

An EP is chosen based upon the specific intent of the perceiver. Lederman and Klatzky (Klatzky & Lederman, 1993; 1996) have carefully identified the criteria by which people choose EPs, and the sequence in which EPs are typically used. Several interrelated concepts determine this: sufficiency, optimality, and compatibility. Sufficiency and optimality rate the accuracy with which a given EP can apprehend a given attribute. Each EP is *optimal* for the attribute associated with it—no other EP can appre-

---

[3] These movement styles can be performed by other body parts as well, but virtually all research has been done on hand movements.

hend that attribute as accurately. However, every EP can apprehend one or more other attributes at a reduced level of accuracy. If an EP can apprehend an attribute to some approximate degree, it is called *sufficient* for that attribute. The number of different attributes for which an EP is sufficient is its *breadth of sufficiency*. An EP that is sufficient for many attributes is *broadly sufficient*.

Some movements combine multiple EPs simultaneously. For example, holding an object provides a rough determination of both its weight and temperature. Two EPs that can be performed concurrently are said to be *compatible*. When compatible EPs are combined in a single movement, each EP may perform at reduced capacity. Participants have found combining EPs useful in object categorization tasks (Lederman & Klatzky, 1996). Several compatible EPs are combined, performing a quick "scan" of multiple attributes of an object. The rough attributes apprehended by the scan may sometimes be enough to categorize the object. If they are insufficient, participants next perform the optimal EP for the attribute that is diagnostic for the categorization. A significant result of this work (summarized in Lederman & Klatzky, 1996) is that participants nearly always chose to perform this broad "scan" first, rather than immediately performing the optimal EP for the targeted attribute. Participants apparently find broad but coarse information on a variety of haptic attributes to be invaluable.

I extend the terminology of Lederman and Klatzky (1996) with the notion of *broadly capable* EPs. Where a broadly sufficient EP apprehends multiple attributes simultaneously (albeit at reduced accuracy), a broadly capable EP apprehends a single attribute at multiple points on an object. On pp. 29–32, I will describe an experimental demonstration (Klatzky et al., 1993) of the importance of broad capability. Their experiment showed that apprehending geometric attributes at multiple points on the object improves object recognition time and accuracy. The notion of broad capability has not become salient in the other work on EPs because in those studies the whole hand could be used unrestrictedly. When considering point force displays, however, the distinction between broad and narrow capability will become significant.

Situated Skills of Perception

The rich skills of movement are one factor contributing to high rates of human performance in situated environments. Another source of high performance is the rich sources of stimuli in physical environments, which contain tremendous lawful structures that serve as diverse and redundant sources of information. These stimuli are invaluable resources for the control of movement, navigation, and selection of action (e.g., Gibson, 1966; Gibson, 1979).

There are a potentially unlimited number of such stimuli. In this section, I will consider only those that prove problematic to display in virtual environments. I will discuss display hardware in a later section. For now, I will simply make the case that certain cues are important to task performance. Bear in mind that there are many cues that are important to human performance that I do not describe here because they can be readily displayed using common technology.

The only visual cue whose display is problematic is *binocular disparity*, the presentation of a different image to each eye. This has been demonstrated to be a powerful depth cue for various tasks in virtual environments when the display technology is available (Hubona, Wheeler, Shirah & Brandt, 1999; Sollenberger & Milgram, 1993; Zhai, Buxton & Milgram, 1996). In a later section I will consider the hardware required to produce this effect. It is enough to say here that this effect is always present in physical environments and is generally consistent with the other depth cues present. Neither of these properties necessarily holds in virtual environments.

Humans are also skilled at extraction of multiple kinds of haptic information from the environment. Klatzky, Loomis, Lederman, Wake, and Fujita (1993) distinguish between the broad categories of material and geometric haptic cues. *Material cues* include thermal cues, compliance, mass, and texture, while *geometric cues* are concerned with the global and local shape of an object. The combination of all these sources of perceptual information is essential for rapid identification of objects. Klatzky et al. (1993) found that object recognition times were increased between 44% and 95% by gloves that substantially reduced access to material cues.

A second type of haptic information is the various counter forces experienced in response to application of force to an object. These include inertia, friction, and environmental forces such as gravity and magnetism. While humans may not be familiar with the mechanical theory of statics and dynamics, they are familiar with the *feel* of these forces and will in at least some cases expect them. These forces may take the form of displacements or torques. Displacement forces are vectors while torques are rotations about a point.

Synergies of Movement and Perception

The richness of situated perception and action combine to produce even higher levels of performance. Many examples are possible. I will consider just two here. First, I will look at the synergy between haptic perception and number of points of contact. This will prove crucial later in the discussion of haptic rendering. Second, I will briefly consider the combination of multiple depth cues into a single percept of depth.

Klatzky, Loomis, Lederman, Wake, and Fujita (1993) performed an important study separating the effects of material and geometric cues from the effect of multiple contact points in an object recognition task. Since their results have important implications later for the potential usability of point force devices, I will describe their experiment in some detail.

Klatzky et al. (1993) asked participants to haptically identify common, hand-sized objects. The experiment had a total of ten different experimental conditions. Of these, only seven are relevant to PFDs[4]. In the baseline condition, participants explored freely with one hand. The remaining conditions were divided amongst shape recognition and object recognition tasks. In the shape recognition tasks, participants wore gloves covering the entire hand surface. Since material properties such as thermal cues were inaccessible, participants identified the objects entirely from their shape. In the object recognition tasks, the fingertips of the gloves were cut away. Since material cues were accessible to the fingertips, these conditions were object recognition (i.e., shape plus material cues). The gloves enforced three different kinds of hand configuration. In one configuration, all fingers could be used and could normally flex. In the second configuration, all fingers could be used but splints restrained the fingers from flexing. In the third condition, only one finger could be used and it was restrained from flexing. The fingertip / no fingertip factor was crossed with the three hand configuration for a total of 6 different glove types. Response times and accuracies are given in Table 1.

In the most restrictive condition, shape-1, participants took approximately seven times longer to haptically identify common objects than in the ungloved condition. Adding fingertip material cues enhances identification speed: The object-1 condition

---

[4] I use different terms from Klatzky et al. (1993) because I intend to use this data as an indicator of usability of PFDs.

TABLE 1. Mean Times and Accuracies for Object Recognition

	Shape recognition			Object recognition		
Fingers	Time (s)	SEM (s)	Acc. (%)	Time (s)	SEM (s)	Acc. (%)
5 (ungloved)	—			6[a]	—	95
5 (flexible)	16	3	93	10	3	93
5 (inflexible)	25	2	90	18	5	90
1 (inflexible)	45	7	74	23	7	85

Note. Estimated from "Haptic identification of objects and their depictions", by R. L. Klatzky, J. L. Loomis, S. J. Lederman, H. Wake, and N. Fujita, 1993, *Perception and Psychophysics*, 54, Fig. 2. Copyright 1993 by the Psychonomic Society. Adapted with permission.

[a] Material properties available to the entire hand surface.

was only four times worse than the ungloved condition.

Increasing the number of fingers from one to five, going from the shape-1 to the shape-5-inflexible condition, improved performance just slightly less than adding material cues (the object-1 condition). Permitting the participants to flex their fingers (the shape-5-flexible condition) resulted in even better performance than the object-1 condition. The geometric information presented by all five fingers, particularly when they could flex around the object, was as valuable for object identification as material cues. Significantly better performance resulted from combining both geometry and material cues: The object-5-flexible condition had a response time that was only 60% of the shape-5-flexible condition.

It is clear from the above results of Klatzky et al. (1993) that humans identify objects using sophisticated combinations of cues extractable by the haptic system, and that participants adjust their identification strategies based upon the availability of various haptic dimensions. The ability to move multiple fingers over an object and the perception of multiple object cues at each point of contact combine in a synergy that is crucial for rapid object identification.

The second type of cue combination I would like to consider is the combination of depth cues into a single depth percept. There are several contending models of this computation (Hubona et al., 1999, sect. 2.1; Landy, Maloney, Johnston & Young, 1995). Some models specify a weighted linear combination of the cues, others use multiplicative combinations. For the purposes of this thesis, it matters less which specific formula is used than that the weights and combinations vary with the task, display conditions, and experience of the observer. Two examples of this process have important consequences for haptic environments. Buckley and Frisby (1993) demonstrated that the relative weights applied to stereo and texture cues were different for physical objects and stereoscopic graphic displays. More recently, Ernst, Banks, and Bülthoff (2000) demonstrated an interaction between haptic experience and the weights assigned to visual depth cues. After observers spent 30-40 minutes manipulating a virtual cube (displayed using both graphics and forces) in a haptic environment, the weights they assigned to purely visual cues changed—even though the forces were no longer displayed.

These two examples demonstrate an important attribute of human situated perception. The multiple stimuli presented by the environment are selectively interpreted to form a percept. In the case of Buckley and Frisby (1993), the difference (most likely accommodation, awareness of the focal plane of the lens of the eye) between physical objects and stereograms generated on a video screen caused different priority to be given to disparity cues. In the case of Ernst et al. (2000), cues perceived by the haptic system changed the interpretation of cues by the visual system.

In summary, human perception of physical environments is highly successful because the multiplicity of stimuli are generally consistent (although the human sensory mechanisms may not veridically register those stimuli). By contrast, the stimuli present in virtual environments are not inherently consistent, will sometimes be highly discrepant, and at times will lack some stimuli altogether. The degree to which this is problematic is hard to predict. On the one hand, human observers are demonstrably good at downweighting cues that are apparently non veridical and compensating for cues that are clearly absent. On the other hand, observers are at least occasionally prone to making incorrect assumptions about which cue is veridical or even whether a cue is present at all. The process evidently becomes still more complex when we consider interactions between cues from both haptics and vision. I believe that it is safe to say, however, that the sophisticated perceptual skills humans use in physical environments will not *inherently* transfer to virtual environments.

Affordances: Perception for Action

The situated skills described in the previous sections were concerned with perceiving cues of objects and manipulating those objects. *Affordances* (Gibson, 1979) are a third situated skill functioning as the crucial link between perception and action. The visibly discernible shape and texture of objects allows us to estimate where to grasp them, how much force we are likely to need to lift them, and how slippery they might be, all before we even touch the object.

An affordance is a percept used in the direct guidance of action. Affordances can work on different levels. At the cognitive level, the affordance may simply indicate that an action is possible. For example, consider a button in a WIMP interface. This is a rounded rectangle on a computer screen, shaded so as to appear raised from its background, indicating that a command will be performed if the pointing device button is pressed while the cursor is contained within that rectangle. The percept of “button” simply indicates “clickable”. More complex affordances also guide the motor operations required to carry out an action. Consider the physical on/off button on a monitor displaying a WIMP interface. In addition to the richer visual cues suggesting its three-dimensional shape (note that the shading of the physical button changes with the ambient room light, while the shading of the virtual WIMP button is fixed), the physical button provides continuous haptic feedback as the user feels it: how hard to press and a click when the power has actually been disconnected. The haptic experience of the physical button guides its operation whereas the WIMP button has no intrinsic haptic experience at all[5].

Affordances can be characterized in terms of the control theory concepts of feed-forward and feedback. Affordances that indicate the availability of actions are *feed-forward*. Affordances that guide the progress of an ongoing movement are *feedback*. The distinction between these two is not hard and fast, however. When our limb movement bumps against an impenetrable obstacle, we receive both feedback (we have not changed position, despite the application of force) and feedforward (further movement in this direction is not possible, at least with the current level of force).

Visual affordances tend to indicate possible actions, as for example the sight of

---

[5] The button on the pointing device, which the user presses to activate the screen button, does have a haptic experience and haptically guides the pressing action. The visual feedback, however, is binary: The screen button is pressed or not. In this case, the mapping from screen widget to physical control is strong for haptics but not vision. For actions such as dragging, however, the mapping is conventional, not derived from either aspect of familiar physical experience.

a doorway affording the possibility of entering a room. A few visual affordances also guide motor skills, as for example the sight of a golf ball on the tee guiding the golfer's swing. By contrast, haptic affordances always guide action. As the example of the physical button shows above, even a rocker switch will offer motor-control guidance.

Haptic affordances occur when an object is contacted and may detect affordances that were not visually apparent. Klatzky and Lederman (1999) found that blindfolded participants who had only 200 msec of haptic contact could nonetheless recognize objects with reasonable accuracy (72%), provided they were given a cue naming the basic category of the object. Klatzky and Lederman point out that not only was the object recognized, but the "haptic glance" apprehended enough information that participants were able to orient their hand for further manipulation of the object. Gaver (1991) has described a related phenomenon, the "sequential affordance". He uses the example of a door handle. The visual appearance of a door handle is an affordance for grasping but may not indicate the direction for turning the handle. Grasping the handle reveals a haptic affordance indicating the correct direction to turn it.

Because they function as a link from perception to action, affordances are an important acquired component of situated skills. If the same affordances can be provided in virtual environments, the environments will require significantly less learning to operate.

Learning Situated Skills

Performance of situated skills improves with practice. The power law of practice, which states that performance time on a task decreases in a negative exponential relationship with the trial number, is an extremely robust and well-established relationship for physical performance (Newell & Rosenblum, 1981). Newell and Rosenblum (1981, pp. 6–7) cite a study by Crossman that found that performance of operators of cigar-making machines continued to improve up to the *three millionth* cigar (and only began to abate when it reached the lower limit of the machine's cycle time). Due to the exponential nature of the curve, the rate of improvement per cigar was considerably less than at the beginning, but it was still measurable. Practice pays off in physical manipulations. Humans appear to be well-adapted to learning such skills, even without conscious focus on the process of learning.

Individual Differences

I have described situated skills as a single body of knowledge, but there may well be wide differences across individuals. Individuals will use different perceptual cues to accomplish the same situated task equally well. For example, approximately 10% of the sighted population lacks stereopsis, the ability to infer depth from binocular disparity, yet these individuals have equally good driving records as those possessing stereopsis (Reinhardt-Rutland, 1996). The perceptual cues required to accomplish a given task may vary strongly between individuals.

The Role of Situated Skills in Virtual Environments

The previous section describe the rich body of situated skills available to adults. These skills have the merits of being well-practiced, having low cognitive overhead, readily learned, and being enjoyable. Might virtual environments incorporating these skills have reduced learning times, lower cognitive load, and be more enjoyable to use?

Note that physical objects are not always easy to use. Norman (1988) argues that objects must have certain properties to be truly usable. He emphasizes the importance of providing good *affordances*[6], that the physical appearance and feel of objects should suggest the manipulations one can perform on those objects. He recommends providing *conceptual models* for how complex systems operate. Good *mappings* place controls in the same spatial arrangement as the items being controlled and make the direction of movement of the control match to the direction of movement of the item being controlled (such as a motorized car window). Finally, proper *feedback* indicates the results of an action to the user and allows the user to readily determine the current state of the system.

The theory of direct manipulation interfaces includes some similar design principles. Baecker and Buxton (1987, p. 432) quote Shneiderman's (1983) definition of direct manipulation:

1. Continuous representation of the object of interest.

2. Physical actions (movement and selection by mouse, joystick, touch screen, etc.) or labeled button pushes instead of complex syntax.

3. Rapid, incremental, reversible operations whose impact on the object of interest is immediately visible.

These are clearly similar to some of Norman's (1988) recommendations.

These recommendations are a potential approach to incorporating the features of usable physical environments into virtual environments. Norman (1988) claims that his design principles apply to the virtual environments of computer systems and his work is widely cited in the human-computer interaction community. However, all the examples of good design he provides in his book are in fact physical systems: the light switches in

---

[6] Norman defines affordance in terms of an information-processing model rather than the direct perception model of Gibson (1979). This apparently subtle shift has significant implications, especially for two-dimensional interfaces. Since my focus is on spatial interfaces, I favor Gibson's definition.

his laboratory, the controls on his Mercedes, stovetops, power plant controls enhanced by beer-keg handles, and faucets. Are physical environments in some way inherently easier to design for good affordances, conceptual models, mappings, and feedback than virtual environments, at least the extant style of virtual environments featuring two-dimensional Windows, Icons, Menus, and Pointing (WIMP) interfaces?

Physical environments are amenable to Norman's (1988) recommendations because the structure of physical interactions and the embedded nature of the user within an environment underlie those recommendations. Physical manipulations have strong and clear affordances and the user's extensive world knowledge provides a clear predictive conceptual model for their behavior. No mappings are necessary, because the user is operating directly rather than indirectly on the object of interest. Feedback is extremely clear—visual and haptic inspection provides continuous information about the consequences of manipulations and the current state of the system. Norman (1988) can be characterized as an argument that virtual environments should be designed to be as much like physical environments as possible.

The benefits of physical interaction will only occur in virtual environments if the perceptual and reasoning skills we have learned in physical environments transfer to virtual environments. This notion can be formalized in two criteria:

1. Do the affordances from physical environments carry over into a given virtual environment?

2. How many of the rich repertoire of manipulatory procedures and exploratory procedures we have learned in physical environments can we use in virtual environments?

The ability of a virtual environment to satisfy these criteria is crucially dependent upon the display and transducer technologies of its hardware. I must describe these technologies before I can consider these criteria.

The Hardware of Transduction and Display for Virtual “Situated Spaces”

For the situated skills of physical environments to be of use in virtual environments, the properties of physical environments upon which those skills depend must be replicated in virtual environments. At first encounter, these properties seem simple enough. A screen can display three-dimensional scenes in perspective (after all, movies seem to be convincing illusions), haptic devices can display forces, and various devices are available to transduce three-dimensional motion. Upon closer inspection, however, that simple formula does not guarantee that situated skills will apply in the resulting environment. A monoscopic graphic display presents only rudimentary depth cues and a single point of input motion and force display is not the same as ten fingers on two hands.

In this section, I consider the degree to which different transduction, graphical display, and point force display technologies recreate the circumstances in which situated skills can be used. Each transducer technology senses different types of human movement. Each display technology can display different stimuli. To the extent that the technology can sense and display the results of more kinds of motion, it can create an illusion closer to physical experience and afford the use of more situated skills. A price-performance tradeoff exists: We can provide more cues with more elaborate technology. In order to know how much hardware to buy, we need to know what cues will be necessary for the desired level of human performance. This section prepares us for the central argument of this thesis, that the required level of cues can only be computed in terms of the task and the intended user population.

The phenomenological definition of environment provides a useful structure for thinking about virtual environments. In this definition, the environment is something that changes in response to human movement. In a virtual environment, the user's movement is transduced, a change is computed in the program state, and the various displays are updated. Each step constrains the range of stimuli that the virtual environment can present. The transduction hardware restricts the range of movements to which the environment responds. It cannot respond to aspects of movement that have not been sensed. The processing further constrains the stimuli, as there are sometimes good reasons to ignore some of the degrees of freedom of the sensed data. For example only one dimension of movement is displayed when dragging a scrollbar[7]. Finally, the display device can impose further constraints, such as the lack of binocular disparity.

This section focuses on the most widely available technologies for transduction and display (plus one research device, the rotating mirror display). The limitations of these devices will constrain the haptic environments we can construct for the next several years, so it is important to understand their capabilities. There are some exotic devices for both transduction and display that offer the possibility of relaxing some of the constraints described below. These devices are too new to assess their feasibility, so I do not consider them here.

---

[7] The situation is slightly more complex. When dragging a scrollbar, the mouse cursor remains displayed, moving in its full two dimensions. However, the user is primarily attending to the feedback provided by the scrollbar, which has only one-dimensional motion.

Limitations of Transducer Technologies

On pp. 23–28, I described point of view movement, manipulatory procedures, and exploratory procedures, the three kinds of movement I will be considering in this thesis. Each of these must be transduced before it can have any effect in a virtual environment.

Capturing full point of view movement requires tracking the head location, the gaze of each eye, and the focal point of each lens. Head tracking technology is available and has been used in virtual environments (e.g., Deering, 1995; Hix, Templeman & Jacob, 1995; Ware, Arthur & Booth, 1993; Ware & Lowther, 1997). A range of eye-trackers are available, trading off accuracy for unrestricted head movement. The styles that leave the user's head sufficiently unfettered to comfortably move are currently only accurate to within one degree of visual field (e.g., Jacob, 1995; Zhai, Morimoto & Ihde, 1999). Tracking gaze more accurately (or tracking lens focal point at all) while still allowing the user's head free movement will probably not be possible for several years.

These transduction limitations restrict the kinds of feedback we can provide to user movements, inhibiting the use of situated skills in virtual environments. An environment that lacks head tracking cannot display perspective projection accurately for the user's current head position. Instead, the environment must compute projections for a standard viewing position, typically some fixed distance from dead center of the display. As the user moves further away from this position and the projection does not change, the projection will be less accurate. Studies have shown that observers can partially compensate for this distortion for both monoscopic (Goldstein, 1987) and stereoscopic (Bereby-Meyer, Leiser & Meyer, 1999) display technologies, but this compensation is incomplete. Perspective will be only an approximate representation of depth for virtually all viewing positions the user might assume.

The absence of transducers for lens focal length prevents the display of the accommodation depth cue. While this cue is not very accurate in its own right, it may have greater influence through its interaction with other cues (Reinhardt-Rutland, 1996). The further absence of transducers for gaze prevents the display of convergence depth cues. Once again, the implications of this are difficult to assess. Some authors have suggested that the combined absence of convergence and accommodation in standard stereoscopic displays is in part responsible for the fatigue experienced by teleoperators (Reinhardt-Rutland, 1996).

Manipulatory and exploratory procedures are transduced by sensors on the hands and fingers. Recall from pp. 29–32 that the multifinger and bimanual nature of these movements was a crucial requirement for high levels of performance. Unfortunately, point force displays can only transduce one point of movement. This suggests that human performance for complex geometric tasks in point force environments will be considerably lower than for performance of comparable tasks in physical environments. I will return to this point at the end of this section.

In some environments motions of the non-dominant hand are also transduced by a two-dimensional pointing device (e.g., FreeForm, SensAble Technologies, 2000), permitting limited bimanual interaction techniques. These techniques are probably far from optimal, however. Bimanual spatial interaction techniques for purely visual virtual environments are incompletely understood (Hinckley et al., 1998; Leganchuk et al., 1998), and the more sophisticated MPs and EPs required for haptic environments will probably require development of new interaction techniques.

Limitations of Display Technologies

The available haptic and graphic display hardware also limit the number of stimuli from physical environments that can be rendered in haptic environments.

Table 2 lists the environmental cues available with different graphical display technologies. The main point of the table is that increasing the numbers of displayed visual cues requires extra graphic hardware, and that even the most elaborate graphical display hardware currently available is still incapable of providing a visual environment as rich as the physical one in which we exercise our situated skills. A few of the column names require elaboration. “Pictorial” refers to the visual cues of perspective, object motion, and texture. These can be rendered by any graphical display. “Full range of angles” refers to the ability to look at a displayed object from any side without experiencing clipping of the viewed object. “Coincident display” refers to whether the point of oculomotor focus is coincident with the point of hand movement. Displays with this cue allow virtual environments to be constructed featuring direct pointing rather than indirect pointing. “Large volume” describes the maximum size of the viewing volume that can be displayed. Typical “small volume” displays have a viewing volume on the order of 0.1 cubic meter, whereas “large volume” displays have a volume two orders of magnitude larger.

The rows of the table list the different graphical display technologies plus physical environments for comparison. “Head mounted stereo” refers to the typical display used in immersive virtual environments. “Mirror” refers to a novel display constructed from a rotating mirror (Plesniak & Pappu, 1998). The resulting image is truly three-dimensional. While this display is still only suited for the research laboratory, I include it here because it is the only current technology that provides accurate convergence and

TABLE 2. Environmental Properties of Display Technologies

Graphical display	Pictorial	Binocular disparity	Full range of angles	Vergence, accomm.	Coinc. display	Large volume
Monocular	X				opt.[a]	
Stereo	X	X			opt.[a]	
HMS[b]	X	X	X [c]		X	X
Mirror[d]	X	X	X	X	X	
PEs[e]	X	X	X	X	X	X

[a] Available as an option, but distorted if no head tracking provided.

[b] Head Mounted Stereo.

[c] Available assuming head tracking is provided, which is the usual case for this kind of display.

[d] Also provides correct perspective for any point of view, even without head tracking technology.

[e] Physical Environments.

accommodation cues and because it also provides accurate perspective from any point of view even when the user's head position is not sensed.

Table 2 demonstrates that selecting a graphical display technology requires judicious tradeoffs. The least expensive and intrusive technology is also the least effective at reproducing physical environments. Is it worth going to a more expensive technology that displays more cues, for example? In a later section I will describe a design method that provides some guidance for answering such questions.

The range of haptic cues that can be displayed by point force devices is more restricted than the range of visual cues. Thermal cues cannot currently be displayed. These are an important component of material cues, which in turn are an important factor in human performance at object recognition (see pp. 29–32). Other material cues such as compliance and at least some forms of textures (Minsky, 1995) can be displayed by these devices. While algorithms for rendering these cues have been developed, they have been little used because material properties convey less useful information in virtual environments than in physical environments. I will discuss the possible uses of material properties in more detail on pp. 113–114.

Display of torque forces is also limited. Current production models of point force displays can display displacements but not torques. A research version of the PHANToM has been developed that displays both displacements and torques, but due to the mechanical complexity of such systems they will always be more expensive (and perhaps less reliable) than their displacement-only counterparts. Under some circumstances, torque forces can be approximated by displacements, but the effectiveness of this varies with the application. Perceptual illusions that roughly approximate torques using displacements have been used in the commercial FreeForm product (SensAble Technologies, 2000) and a “virtual lathe” (Plesniak & Pappu, 1998). In these applications, the disparity between the illusion and actual torques does not significantly hinder performance, but surgical training devices must render actual torques (Mor, 1998) rather than illusions.

The most significant haptic display limitation is the restriction to displaying only a single point of force. This will be considered in detail in the next subsection.

(Anti-) Synergies of Movement and Perception in Point Force Environments

In physical environments, the rich diversity and multiple sources of cues are used in combination by the human perceptual systems to achieve a high level of performance. We have seen that by contrast haptic environments are deficient in many visual and haptic cues, and that different transducing and display technologies will make different sets of cues available. What might be the effect on human performance?

The most striking limitation occurs in the haptic sense. Ten fingers on two hands are replaced by a single point, and important material cues are missing. Recall that this is almost exactly the situation in the shape-1 condition in Table 1, where participants had no access to material cues and could only use a single unflexed finger. Performance was seven times slower than unrestricted exploration. One of the key factors is the lack of broadly capable and broadly sufficient exploratory procedures. In physical environments, these are used to get an initial quick scan that is used to select the more specific procedure that follows. This strategy was unavailable in the shape-1 condition of Table 1 and is unavailable in point force environments.

Further performance limitations may result from the use of unimanual or limited bimanual techniques. While I am not aware of any study directly comparing the performance of unimanual and bimanual object recognition[8], Hinckley et al. (1997, Experiment 2) demonstrated that when participants were asked to place two objects relative to each other in space without visual feedback, bimanual interaction allowed twice as accurate placement as using a single hand to sequentially place the objects. If this bimanual relative frame of reference is also used in object recognition tasks, similar performance differences may be noticed in that task.

The magnitudes of the effect sizes in Klatzky et al. (1993)) and Hinckley et al. (1997, Experiment 2) are large (seven and two times, respectively). The single most important research question in human performance with point force environments is the extent to which performance suffers from the restriction to a single point of contact and ways in which that reduction can be ameliorated. This amelioration may take the form of

---

[8] Comparing Klatzky et al. (1985) (where both hands were used) and the “baseline” condition of Klatzky et al. (1993) (where only one hand was used), the one handed method appears to be slower. However, the data reported in the two papers is insufficient to make a quantitative comparison.

bimanual interaction techniques explicitly designed to facilitate exploratory procedures or point force interaction techniques designed to facilitate broadly capable and broadly sufficient exploratory procedures.

A lesser limitation occurs in visual depth perception. All the graphical display technologies are limited and do not render all the cues available in physical environments, so there is the possibility of performance reductions here. However, studies have shown the human visual system is capable of compensating at least partially for missing cues, potentially mitigating the degree of performance reduction.

To summarize this section, the seemingly simple concept of space and objects within that space is surprisingly subtle and difficult to simulate with a computer system. Physical and virtual environments are not equivalent. Different combinations of display and transduction technologies offer different kinds of cues, feedback for aspects of human movement. The possible combinations of graphical and force displays produce a huge design space. The research results on haptic perception discussed above imply that the performance characteristics will vary widely across this design space. Designers cannot simply assume that an arbitrary combination of display technologies will be “close enough to physical experience”. Current technology does not allow the complete simulation of all the properties of physical environments that afford human situated skills. The only way to define “close enough” becomes “close enough to effectively perform a certain task”. Designers of haptic environments will have to carefully match the cues provided by their environments to the needs of their users and the task.

Graspable Environments: Specialized Devices for Situated Skills

There are two approaches to using the above technologies to incorporate situated skills into two- and three-dimensional virtual environments. In this section, I describe the *graspable environment* (GrE) approach, which constructs input devices specialized to a single task. The devices in GrEs have specific physical forms and transduce aspects of movement significant to their task. These devices often also deliberately constrain the physical range of motion of the user to the optimal paths for a given task. This approach succeeds well in affording situated skills. Since the devices are physical objects, they can be designed to have full physical affordances suggesting the movements they transduce. However, the specialized devices restrict the environments to single applications. Point force environments, described in the next section, have a broader range of application because they use general force displays.

The original graspable environment (named a “graspable user interface”, GrUI, by its creators, Fitzmaurice & Buxton, 1997; Fitzmaurice, Ishii & Buxton, 1995) used specialized input devices that transduced multiple degrees of freedom and in some cases had physical constraints on the movement of their parts. Each input device was bound to a specific interaction technique built around the input data. For example, a “brick” was an input device that both transduced two-dimensional location and rotation and was used to specify an orientation at a given point in space. A “stretchable square” was a device that specified the location of the two diagonally-opposite corners of a rectangle. Manipulation of the square was constrained—it could be moved as a whole and diagonally opposite corners could be slid towards or away from each other—but the shape of the device was always constrained to be rectangular. The square was an efficient way to specify a rectangular region using two hands.

Graspable environments take advantage of situated skills. The shape and feel of the objects provide clear affordances for their operation, and the devices live up to the promise of those affordances by transducing all the manipulations their affordances suggest. For example, the constraints on the shape of the stretchable square are readily apprehended by the haptic sense and interpreted as affordances suggesting precisely the limited kind of manipulation which the associated interaction technique requires as input—the specification of a rectangular region.

Graspable environments also take advantage of human proficiency at situated two-handed interaction by providing *spatial multiplexing* of input devices (Fitzmaurice & Buxton, 1997). Rather than the *temporal multiplexing* of current WIMP interfaces, where a single pointing device is connected to different virtual objects serially over time, GrEs have multiple input devices each connected to a single virtual object for an extended time. This permits the manipulation of a separate object by each hand and also the use of two-handed manipulation of the same object, such as the use of two hands to stretch the stretchable square.

Similar work[9] has been done on Tangible User Interfaces (TUIs) (Ishii & Ullmer, 1997; Ullmer, Ishii & Glas, 1998), Manipulable User Interfaces (MUIs) (Harrison, Fishkin, Gujar, Mochon & Want, 1998), and Props (or Proxy) User Interfaces (PrUIs) (Hinckley, Pausch, Goble & Kassell, 1994; Schmalstieg et al., 1999) In each of these cases, since the input device is a physical device embedded in the same physical space as the user, Norman's (1988) principles of affordances, constraints, and visibility can be directly applied. The user's tremendous skills at situated manipulations are directly

---

[9] I group these interface styles together based upon the styles of movement each supports. They were each designed with different aims, and hence they are quite distinct, but the differences appear on the level of semantics, not their movements.

accessed.

These interface styles have an inherent limitation: The input devices used are all specialized. The great strength of these devices, their physicality, also restricts them to being used with a single interaction technique. Fitzmaurice (1997) argues that this is in fact desirable because more specialized input devices afford higher performance for their single task. He proposes that users acquire input devices specialized to each of their different computer applications just as they currently acquire specialized tools for different physical tasks.

I think this will only cover a fraction of the applications for which we might wish to use computers. While the work on GrEs has produced some compelling fits of interaction technique to task, each of these techniques has been limited in its range of applications. Much of the power of computers comes from their general purpose nature as abstract symbol systems. Specialized devices have their specialized place—and that place includes applications of great social and commercial value—but they can never keep up with the tremendous range of a general-purpose computer. This generality is not only across applications but also influences the structure of a single application. Software can stretch in ways that physical objects cannot. For example, the bibliographic software I use began with 100 references and currently handles 1000 with ease and with some reduction in performance has handled a database of 30,000 references. It is hard to imagine a physical analog that could go from 100 to 500 without a major redesign, let alone to 30,000.

Graspable environments meet the first but not the second criteria proposed on pp. 38–39: The affordances learned in physical environments apply in GrEs because the input devices are themselves physical objects. However, each device only supports a single style of movement and so only a small fraction of situated skills can be applied in these environments.

Point Force Environments: General-Purpose Devices for Situated Skills

Graspable environments are designed around task-specific devices. There is another class of input devices, point force devices (PFDs). Point force devices represent a different trade-off from graspable devices, weaker in visual affordances but featuring programmable force display and the capability to transduce general movements. Point force devices enable a different style of virtual environment, which I call *point force environments* (PFEs)[10]. The greater generality of these devices permits a wider range of interaction techniques to be used with a given hardware configuration as well as the development of more flexible interaction techniques.

For a graspable device, the visual affordances result from a combination of the physical structure of the device and the graphical feedback provided by the interaction technique. However, the haptic affordances result strictly from the physical structure of the device—lacking programmable force display, the interaction technique cannot dynamically change the haptic feedback.

Consider the specification of control points on a spline curve using two “bricks” (Figure 1). The visual appearance of the physical structure of the bricks is an affordance suggesting grasping. The interaction technique provides feedback about the current state of the spline curve based upon the locations and rotations of the two bricks, a visual affordance. However, there is no haptic resistance to the rotation of the bricks, so the haptic affordance of bending forces is not provided. This form of motor control,

---

[10] In contradistinction to graspable environments, these might be called “pokable environments”.

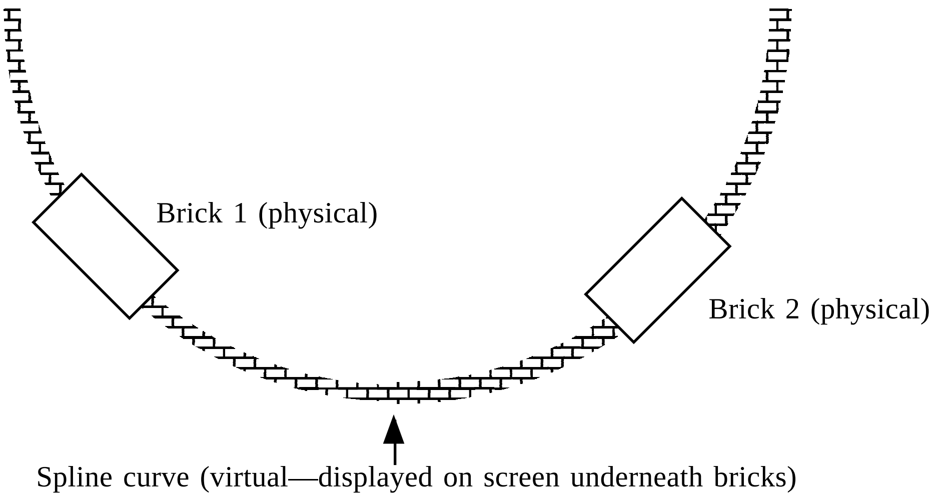


FIGURE 1. Using Two “Bricks” to Bend a Virtual Spline Curve. Adapted from Fitzmaurice et al., Fig. 6

where a visual representation at one location guides the use of the hands at another, is more highly mediated than in physical environments, where the haptic experience (in this case, resistance to bending) guides hand use.

By contrast with graspable devices, the force display of a point force device is under program control and the interaction technique can affect the haptic affordances of the device as well as the visual affordances. Consider once again the task of specifying two control points on a spline curve, only this time using two point force devices with torque display. With this display hardware, the interaction technique can dynamically render the resistance to bending and twisting of a physical drafter’s spline, providing haptic feedback for the motor task.

This difference has significant implications for the two kinds of environments. Graspable devices have strong visual affordances that suggest the kind of movements they transduce. Furthermore, the devices permit spatial multiplexing (the interaction technique is initiated by reaching and grasping the device) whereas the point force devices are temporally multiplexed (the interaction technique is initiated by moving the device to a region of space and then binding the device to the virtual widget). For these reasons, graspable environments will probably support higher rates of human performance for their specialized tasks than are possible with point force environments applied to the same tasks.

However, these advantages may be small. Since the transduced movements and force displays of PFEs are general rather than specific, they can be used with many interaction techniques. Once a given interaction technique has begun, the force display can constrain the user's movement. Thus while the appearance of the point force device, its visual affordances, does not suggest its possible movements as clearly as does the appearance of a graspable device, during the actual execution of an interaction technique point force and graspable devices are roughly equivalent.

This suggests that while point force environments may have lower performance than graspable environments for a few specialized tasks, point force environments may have overall satisfactory rates of performance for a much broader range of tasks than the specialized graspable environments. By the criteria on pp. 38–39, PFEs have the opposite strengths and weakness of GrEs. The affordances from physical environments do not carry over so well into PFEs, but a far larger proportion of the rich repertoire of movement skills is available (although still not the full repertoire). The generality of point force devices may prove a better match to the computational generality of computer systems than the specialized devices used in graspable interfaces. For the remainder of this thesis I focus on the possibilities offered by point force environments.

Point force environments are inevitably compromises between the complexity of the hardware and software and the perceptual cues presented to the user. To be usable they must be designed with careful attention to the task and perceptual cues. The research results on human performance in these environments is limited.

Prior Research on Point Force Haptic Environments

The following sections summarize the research to date on point force haptic interfaces.

Sandpaper

The Sandpaper system (Minsky, 1995) explored a novel application of an active force-feedback joystick. The joystick's bandwidth range from 100–1000Hz and was controlled by an algorithm creating variations of lateral force on the joystick. One of the major research results of this project was that it is possible to generate a tactile sensation using a kinesthetic stimulus. This blurs the boundaries between tactile and force displays. Kaczmarek and Bach-y-Rita (1995) point out that the high bandwidth of the joystick was essential to the success of the simulation. The system was most effective at displaying grating textures—surfaces with small, periodic ridges—and least effective at displaying smooth and randomly-varying textures. These latter textures will probably require genuine tactile displays for effective simulation.

Nanomanipulator

The Nanomanipulator (Taylor et al., 1993) integrated force output with stereoscopic visual output for the control of a Scanning Tunneling Microscope (STM). The STM measures the height of a microsurface at about the level of accuracy of an atomic radius. The Nanomanipulator allows the operator to both see the microsurface and feel it.

The description of the system in (Taylor et al., 1993) emphasizes the visual display and only tersely describes the force output system. In fact, there isn't a single example given of the haptic display contributing to a scientific discovery. One of the difficulties may be that the perceptual cues given by this system differ significantly from those we receive in daily life. The force output technology used in the Nanomanipulator does not have the high bandwidth of the Sandpaper system and so cannot represent textures with any degree of verisimilitude. Furthermore, the system is operated by a handgrip, creating the kinesthetic stimuli in the arm and shoulder rather than the fingers and wrist stimuli presented by Sandpaper's joystick. Finally, there is no visual representation of the user's arm, simply a disembodied cursor at the current scan point.

While the use of force feedback in this context is interesting, the technology does not seem to have contributed much to this application. A newer version of the Nanomanipulator (Taylor, 1997) uses the PHANToM force display whose higher bandwidth permits more accurate rendition of textures. However, no performance results have been reported with this newer version.

GROPE-III

GROPE-III (Brooks, Ming, Batter & Kilpatrick, 1990) is the latest in a thirty-year series of projects exploring the use of force output in analyzing molecular docking problems. The system attempts to provide a simulation environment for chemists to discover low-energy docking configurations more quickly and to develop a “feeling” for the dynamics of molecular docking. The system displays three dimensional models of two molecules, typically a protein and a drug. The user maneuvers the smaller molecule, usually the drug, into a low-energy configuration with the larger molecule.

In addition to the visual display of the molecules, GROPE-III provides both visual output and active force output of the energy configuration of the molecular system. The visual output takes the form of a thermometer displaying the current energy level, while the force output provides resistance or attraction proportional to the atomic forces acting on the docking molecule.

In a simplified version of the docking task, biochemists using the force output version without any visual display were 2.2 times as fast as when they used a visual-output only version. This simplified task was unrealistic, as there was only a single minimum-energy solution. In the force output version, if the handgrip were simply allowed to run free it would find the minimum-energy point automatically. The main purpose of this initial experiment was to demonstrate that force output was an adequate modality for conveying the energy level of a molecular system. It is encouraging that the biochemists were able to locate the configuration more quickly using only the kinesthetic sense than when using only the visual sense.

A second experiment measured performance with actual drugs and proteins, systems which have multiple locally minimal configurations. For this task, the biochemists used both visual-only and visual plus force versions. The total docking times were not significantly different between the two versions. When the docking was broken down into suboperations, differences were found. The time to do the six-dimensional phase of the docking required 25% less time in the version with haptic output.

Both GROPE and the Nanomanipulator are in production use by biochemists and physicists. They are the only examples of production haptic interfaces which I am aware of. Brooks et al. (1990) report some interesting observations:

1. There appears to be a twofold maximum performance improvement from supplementing visual output with haptic output.

2. The participants had no problem accommodating the haptic output. In fact, they often didn't even notice it was there until it was turned off.

3. The authors claim that the most significant outcome of the technology may not be reduced task time, but enhanced understanding, leading to new approaches to drug design.

Multi-Modal Mouse

Akamatsu (Akamatsu, MacKenzie & Hasbroucq, 1995; Akamatsu & Sato, 1994) developed a mouse enhanced to provide limited tactile and force feedback. They conducted two performance experiments with the device on a target-selection task. The first (Akamatsu & Sato, 1994) used both tactile and force feedback to distinguish the target. In this experiment the haptic feedback provided a significant improvement to pointing, increasing the Index of Performance (IP), the measure of pointing task throughput of the device, from 2.96 to 3.23. In the second experiment (Akamatsu et al., 1995), tactile (only), sound, color, and combined feedback were compared with the normal condition of no feedback. No significant difference was found overall, but there was a significant difference between the feedback modes for the time spent after positioning the cursor over the target. No multiple comparison results are given, but their Figure 3 suggests that the means clustered into three groups: tactile and combined, sound and color, and normal (no feedback). Tactile feedback increased the effective width of the target, decreasing the difficulty of the task.

Akamatsu et al. (1995) claim that positional feedback could only reduce the time of movement over the target since the feedback was only engaged when the cursor crossed the boundary of the target. Hill and Salisbury (1978) reported a similar result in a teleoperation docking task, finding that force output did not decrease time to initially position the manipulator but did speed up the actual docking operation. On the other hand, these results seem to be contradicted by some microstructure models of rapid aimed movement. Several experiments (MacKenzie, Marteniuk, Dugas, Liske & Eckmeier, 1987; Meyer, Abrams, Kornblum, Wright & Smith, 1988) have demonstrated that the entire course of movement, and not merely the portion of movement over the actual target, is affected by the width of the target.

While the results of Akamatsu et al. (1995) do not indicate performance gains of any size, they note that the benefits of extra sensory modalities of feedback will be most apparent in conditions where the visual system is already overloaded. The tasks used in both studies were too simple to demonstrate the benefits of haptic feedback. Perhaps the differences predicted by the microstructure models will produce more discernible differences in performance under more demanding task conditions.

Palm-Sized Display with Haptics

There is only one study exploring the use of continuous-valued haptic output in circumstances where the user is primarily attending to another task. Noma, Miyasato and Kishino (1996) describe a palm-sized display attached to an active force feedback arm. The display has two modes of use. In the observing mode, the display acts as a virtual camera, showing a view of the virtual world as seen by an observer whose line of sight is perpendicular to the display.

Pressing a button shifts the display into handling mode. In this mode, the virtual object currently in the center of the display is “grasped”. Moving the display causes the object to move. In this mode, the kinesthetic output provides feedback for grasping objects and collisions between objects. This feedback is designed for the specialized purpose of making it easier to place objects relative to one another.

The authors report a pilot study which suggests that the kinesthetic feedback was useful. Four engineers from their lab each performed a series of 240 trials, picking up a virtual box and placing it next to a virtual wall. Three wall distances were used, ranging from a short reach of 25 cm (10 inches) to an extended reach of 65 cm (26 inches). Using the force feedback and visual display, the engineers were able to place the objects in 70% of the time and with far less error than using only the visual display.

There is a subtle difficulty with this device. In handling mode, it does not correspond to any physical part of our body—it is a combination of a hand and an eye. As such, the authors devoted considerable time to considering various interaction techniques: Should it be more like an eye or a hand? Furthermore, the display is located at some distance from the object being manipulated. It is as though the user is moving an object glued to the end of a pole rather than grabbing the object itself and manipulating it.

The differences between working with the palm-sized display and our hands and eyes are subtle yet significant. It is impossible to completely capture the relationship between the background and the object being moved. Rotations of the object are especially problematic. When the display is rotated, is the center of the rotation the object or the display? Does the object rotate within the display while the background remains fixed or does the object stay fixed while the background moves? The authors compared several methods and determined that the best method may vary with the task. Their results were not conclusive enough to come up with recommendations of which relationship works best with which task.

Tremor-Resistant WIMP

Rosenberg and Brave (1996) investigated the use of passive and active force output to make WIMP interface widgets resistant to hand tremor. The system was intended to make it easier for users with high degrees of tremor (due to neuromotor disabilities) to use scrollbars, buttons, and menus. The authors programmed a force output joystick to “snap” to the center of the target. In a pilot study, the authors found that users with tremor could activate the controls in less time using the force output system than using a conventional graphics-only system. Active force output increased performance more than passive force.

Other Pointing Studies

Several recent studies have measured the effect of haptically enhanced buttons on pointing tasks (Arsenault & Ware, 2000; Eberhardt, Neverov, West & Sanders, 1997;

Oakley, McGee, Brewster & Gray, 2000) and steering tasks (Dennerlein, Martin & Hasser, 2000). As with the previous work, pointing performance was found to improve within the range of 10-15%. The benefits for the steering task were larger: 52% for a pure steering task and 25% for a combined steering and targeting task. While these latter results are encouraging, they are applicable only to a small set of practical tasks where the direction of the user's motion is known in advance and can be assisted by haptic constraints.

Summary of Previous Work

The diverse work to date on point force haptic environments has not produced many examples of improved human performance for the overall task. The respective authors have typically argued that the benefits of these technologies lie in the realms of improved understanding of scientific data or greater affective value. While these latter benefits are surely of value (albeit hard to experimentally verify due to the tremendous difficulty of operationalizing such concepts), the lack of response time benefits is nonetheless puzzling. The haptic system is crucial for the effective performance of a variety of tasks in daily life. Why doesn't it produce similar benefits in virtual environments?

These results suggest that our approach to introducing haptics into a computer interface does not take all the requisite factors into account. The next two chapters explore the mechanics of haptic interaction techniques in detail. In the final chapter, I will reconsider the issue of implementing haptic environments in the light of the results of these experiments.

CHAPTER III

EXPERIMENT 1—CURVATURE ESTIMATION

Introduction

The descriptive framework presented in Chapter II emphasizes that haptic environments function as perceptual resources for accomplishing a task. Haptic environment designers must strive to provide perceptual resources that will be of use to the broadest possible range of users for the tasks for which the environment is designed. The "intuition" of the designer, often merely a euphemism for the results of the designer's introspective analysis of how she or he performs the task, will probably not be a reliable guide to the perceptual resources required by actual users. These users will typically have far less experience with both the specific environment and with haptic environments in general. Indirect three-dimensional pointing and the unnatural combination of depth cues will make these environments unfamiliar to the users.

Empirical data is a more sturdy foundation for design than introspection, and gathering such data is the purpose of this chapter and the next. In these chapters, I describe experiments exploring the causal factors influencing human performance of two tasks in haptic environments. These experiments are primarily exploratory. That is, they are intended more to generate hypotheses than to prove them. While there are some testable hypotheses of causal effects involving experimentally manipulated variables, a broader goal of the experiments is to define the factors that may be significant determinants of human performance in haptic environments. In particular, the most important thing is to develop some experimental tasks, see what they measure, what factors might impinge upon them, and even how long typical trials take. Developing initial tasks and protocols is itself an essential contribution in this early stage of understanding of haptic environments, for it is the task that will define the space of possible factors we can explore. The task also defines the operationalization of the causal factors. Finally, the experiment provides an initial estimate of the effect sizes of these factors, how much each one affects human performance. As preliminary studies, the effect sizes reported will of necessity be rough, perhaps only reliable within a half order of magnitude.

The discussion of haptic environments in Chapter II influences the design of this experiment in several ways. First, the experimental task is derived from a perceptual task that is likely to occur in applications of haptic environments. Second, the experimental task examines the mechanics of perception, and in particular the interaction between vision and haptics and the changes in their respective contributions when working in an environment where the perceptual resources are significantly reduced from physical environments. Third, the experiment compares the performance of alternative interaction techniques, vision alone versus vision and haptics, on the same perceptual task. Finally, I examine the influence of individual cognitive differences as measured by a mental rotations test, on performance of the task.

I think it is equally important to emphasize that these experiments will not address the crucial issues of reliability and validity of the measures. At this early stage, I believe the first task is to propose tasks and causal factors and get initial estimates of their importance. Only after this groundwork has been laid will it be appropriate to rigorously test the constructs.

Evaluating the Interaction of Perceptual Systems

One task that may well arise in haptic environments is the estimation of curvature of an object. This task occurs in solid modeling applications, where the user models a three-dimensional object using mathematical surfaces such as Non-Uniform Rational B-Splines (NURBS). Users of solid modeling programs are often directly concerned with the curvature of the objects they create. For example, in a model, the user may have specified control points for a curved surface and want to see just how curved the resulting surface is. This is an application where precise determination of curvature is fundamental. There are currently a plethora of commercial non-haptic solid modeling applications as well as the FreeForm (SensAble Technologies, 2000) haptic modeling program. All of these programs are predominantly used with monoscopic displays, although some can accommodate stereoscopic display hardware if available. While most of these applications do not currently support force display, they would readily incorporate such displays if the displays can be shown to improve performance.

Klatzky, Lederman, and Matula (1993) found that when vision and haptics were both available, vision was used almost exclusively for geometric judgments (size and shape), while haptics were important for material judgments (roughness, compliance, temperature, and weight). However, haptic perception of geometry is also reasonably good in physical environments. Observers using haptics alone can discriminate between three types of simple unfamiliar shapes in one to two seconds (Klatzky, Lederman & Reed, 1987; Lederman, Klatzky & Reed, 1993). In the study described on pp. 29–32 (Klatzky et al., 1993) where observers identified common physical objects (which are more complex) using only shape, recognition times were approximately 17 seconds with 94% accuracy (see “shape-5-flexible” row of Table 1).

In Chapter II, I proposed that human perceptual systems might perform differently in a point force environment than in physical environments. In physical environments, multiple redundant cues combine to give each individual perceptual system great power. However, in haptic environments there are fewer cues for both the visual and haptic systems. Depth information is typically far less completely specified for the visual system and the haptic system has been reduced to a single point of force. In such circumstances, the individual systems might be reduced to much lower performance than in physical environments.

The most widely available desktop hardware configuration features a monoscopic display. It would be advantageous if haptic environments could achieve satisfactory performance with such a display, as the only remaining hardware expense would be a haptic display. Unfortunately, this configuration also gives the least number of depth cues. I term such a configuration a Low Cue Visual Environment (LCVE). How might this lack of cues affect human performance of curvature estimation? Vision may not be as effective at apprehending shape in an LCVE as in physical environments. However, two perceptual systems combined might apprehend the kind of rich, redundant set of cues that give the individual systems such power in physical environments. The supplementary cues provided by haptics might enhance performance.

The haptic system clearly apprehends shape cues. Are these cues available to be integrated with corresponding visual cues? When blindfolded observers are asked to sort objects that “would look like one another” they almost exclusively use object shape as the sorting criterion (Klatzky et al., 1987), ignoring material properties. This suggests that some common representation exists between haptic and visual perception and therefore the haptic system could supplement the visual cues.

On the other hand, performance of the haptic system is also likely to be reduced in point force environments because of the reduction from multiple to single points of contact with the object. In particular, shape cues are far less readily available in point force environments than full-hand environments (see Table 1). Thus we have two contending influences. Haptic perception provides an additional source of cues, supplementing the impoverished cues received from an LCVE by the visual system, but the cues apprehended by the haptic system are themselves impoverished. It is not clear, then, how much the addition of haptic cues will improve the perception of shape in LCVEs. The experiment described in this chapter examines the relative contributions of vision and haptics to curvature estimation in a LCVE with a point force haptic display.

Operationalizing Estimation of Curvature

To operationalize the above research question, I need a precise definition of curvature. Several different definitions of curvature have been used, and several different experimental protocols have been used for curvature estimation. Most previous research on perception of the curvature of physical objects (Goodwin & Wheat, 1992; Gordon & Morison, 1982) has used spherical stimuli, as has the only study to date of haptic curvature perception in point force environments (Tan, 1997). For spherical stimuli, the curvature is measured in units of m^{-1} and is defined as the inverse of the radius.

A disadvantage of spherical stimuli is that they only have a single parameter, their curvature. It is impossible to consider families of related shapes. Koenderink and van Doorn (Koenderink & van Doorn, 1992) addressed this problem by defining a class of curves whose shape and curvature can both be varied. They consider surfaces described by the quadric equation

$$z(x,y) = \frac{(k_1 x^2 + k_2 y^2)}{2} \tag{3.1}$$

where k_1 and k_2 define the two *principal curvatures*. For such a surface, they define the haptic dimensions of *shape* and *curvedness* as follows:

$$S = -\frac{2}{\pi} \arctan(\frac{k_1 + k_2}{k_1 - k_2}), \qquad k_1 \geq k_2 \tag{3.2}$$

$$C = \sqrt{\frac{k_1^2 + k_2^2}{2}} \tag{3.3}$$

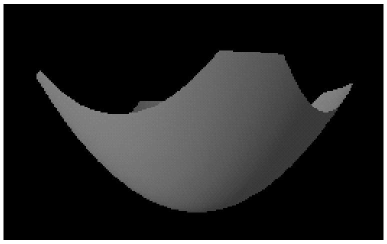
The shape index, S, varies from -1, a concave spherical paraboloid, to 1, a convex spherical paraboloid (see Figure 2). This value represents the overall contours of the form and is scale-independent. The curvedness index, C, represents the degree of curvature. It is measured in units of m^{-1} and varies from zero, denoting a flat surface, to infinity, denoting an infinitely curved surface. Note that curvedness is scale-dependent. Although curvedness is measured in the same units as curvature for spheres, it represents a different quantity and the two cannot be directly compared. For the paraboloids used in this chapter, I use a rule of thumb that the curvedness value is twice the curvature value of a comparable sphere[1].

For this experiment, I used only the surfaces with $S = 1$ ($k_1 = k_2 < 0$). These are convex paraboloids of revolution. Each stimulus consisted of two such paraboloids back to back. For these surfaces, the curvedness is simply equal to $-k_1$.

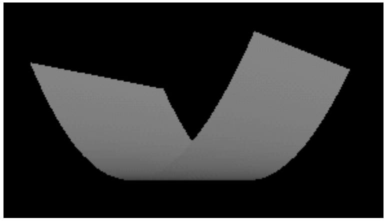
This experiment compares the performance of vision with vision and haptics by

---

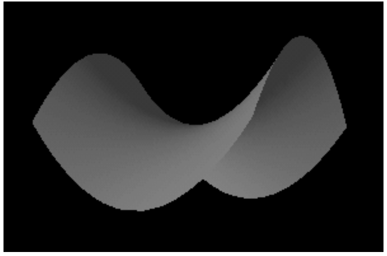
[1] The rule is derived by fitting a parabola through the 0, 90, and -90 degree points of a unit circle. The curvedness of the resulting parabola is twice the curvature of the circle.



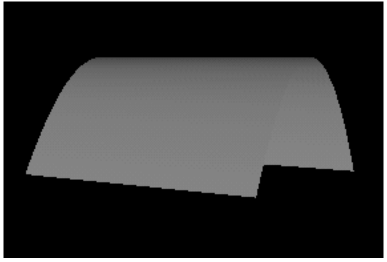
(a) Cup: $S = -1$



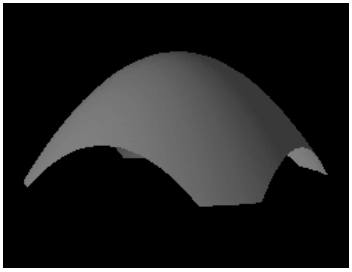
(b) Groove: $S = -0.5$



(c) Saddle: $S = 0$



(d) Ridge: $S = 0.5$



(e) Top: $S = 1$

FIGURE 2. The Five Shapes at the Critical Points of the Shape Scale.

comparing the psychophysical functions for the estimated curvedness of these paraboloids. For prothetic (magnitude) continua, the intensity of a stimulus, ϕ, and the perceived value according to a subjective scale of the observer, ψ, are related by Stevens' power law, (Gescheider, 1997, p. 298):

$$\psi = k\phi^a \tag{3.4}$$

The exponent of the function, a, varies with the stimulus dimension. For tactual dimensions, the exponents range from .42 for viscosity of silicon fluids to 1.7 for discomfort due to cold, with an outlier at 3.5 for electric shock through the fingers (Gescheider, 1997, p. 303). The exponent is typically computed as the slope of the regression line performed on the log of the intensity and log of the estimated magnitude. A larger slope indicates that a given increment in stimulus intensity produces a larger perceived difference. Lederman and Klatzky (1999) argue that steeper slopes therefore indicate greater discriminability. This experiment compares the discriminability of vision and vision together with haptics.

Spatial Abilities and Mental Rotations

I made the case in Chapter II that adults have tremendous skill at manipulating objects in physical environments. While some of these are motor skills, it is also possible that some significant portion of the skill is knowledge about the world, independent of dexterity of performing physical manipulation. In this section I consider what such a skill might be and how we might measure it. I derive a version of a mental rotations test that I will use in Experiments 1 and 2 to assess how well skill at mental rotations predicts individual performance in these experiments.

The construct of *spatial abilities* is an operationalization of the notion of abstract knowledge of orienting objects and one's own point of view in three-dimensional space. Spatial abilities is a broad construct and some have argued that it is too loosely defined to have any explanatory value (Caplan, MacPherson & Tobin, 1985). Furthermore, it has become so closely associated with the arguments for and against the existence of gender-related cognitive differences that most articles on spatial ability focus on gender to the exclusion of other considerations. The issue of gender-related cognitive differences is well outside the scope of this work. My concern is whether some subset of what is known as "spatial abilities" can predict performance of users on tasks in haptic environments. Nonetheless, I will discuss some of the results on gender differences simply because so much of the spatial abilities work is couched in those terms.

The most extensive meta-analysis of research on spatial abilities focused on differences related to gender. Voyer, Voyer, and Bryden (1995) analyzed 286 studies of spatial abilities and divided spatial abilities tests into the categories of mental rotations, spatial perception, and spatial visualization. They found that mental rotations tests produced the most consistent and strongest measure of such differences. However, the mental rotations category was itself not sufficiently homogenous to be considered a measure of a single effect. Voyer et al. (1995) divided it further into studies using the Cards Rotation Test, studies using the test of Vandenberg and Kuse (1978), and an aggregate category titled Generic Mental Rotations. The Cards Rotation Test uses rotations of two-dimensional stimuli, while the other two tests use rotations of three-dimensional stimuli. The Vandenberg and Kuse (1978) is a paper and pencil test yielding only accuracy data, while tests in the Generic Mental Rotations category are computerized and produce both response time and accuracy. Voyer et al. (1995) found that studies using the specific Vandenberg and Kuse (1978) test were themselves heterogeneous, with different effect sizes depending upon the scoring method used.

My interest in this data is not in putative cognitive differences in gender but in what this meta-analysis indicates about various spatial ability tests. The Voyer et al. (1995) meta-analysis indicates that mental rotation tests are reliable overall, although individual tests and scoring methods differ in the strength of their outcomes. Because of this result I chose to use a mental rotations test as a measure of spatial abilities in this experiment. In particular, I chose the PsychExperiments (1999) mental rotations test, a computerized test using three-dimensional stimuli, which would fall in the Voyer et al. (1995) Generic Mental Rotations category. The test uses stimuli similar to those of Vandenberg and Kuse (1978).

A mental rotations test has been used in at least one prior study of three-dimensional interaction techniques. Grissom and Perlman (1995) used results of the Vandenberg and Kuse (1978) test as a covariate in their standardized evaluation plan, StEP(3D). The test results did not have a sufficient correlation with performance to provide a useful covariate. However, this lack of predictive power of the test may have been due to the limited data provided by that particular test. As a paper and pencil test, it only measures accuracy and provides no response time data. The computerized test used in this experiment provides both accuracy and times, and hence its results may have more predictive power.

Development of Mental Rotations Scale

By itself, the PsychExperiments (1999) mental rotations test only provides a pair of results, mean trial time and total accuracy. There is currently no standardized scale for relating performance of an individual to the broader population. The University of Mississippi database was used to construct a preliminary scale of mental rotations ability. This database contains tens of thousands test results, collected via the World Wide Web over a span of several years.

All test results from the period Nov. 30, 1999 to April 20, 2000 were downloaded. The resulting file had responses for 18,663 trials performed by 515 participants. Screening the data revealed six participants who appeared to be answering randomly: Their performance was at chance level and their mean response times were well under a second. These six were deleted, leaving a reference data set of 509 individuals.

The reference set consisted of 328 females and 181 males, 453 right-hand dominant individuals, 37 left hand dominant, and 19 “mixed”, with ages ranging from 5 to 63, and mean (median) age of 21 (20) years. Seven-number summaries[2]of the distributions of age, mean log response time, and percent correct are given in Table 3, Table 4, and Table 5, respectively. The age and gender proportions are obviously different from the overall population—the seven-number summary indicates that 75% of the participants are between 18 and 23—and the group is self-selected, but it is not clear at this time whether the distributions of response time and accuracy for this group differ from those of the overall population. For the purposes of this study, I assumed that this reference set was representative of the overall population of adult, college-educated, computer-literate individuals, a population that clearly includes the participants in my study.

Response times for the reference set were distributed lognormally. For each individual in the set, the mean of the log response time was calculated. Figure 3 shows a normal-quartile plot of the resulting distribution. There is a single outlier on the positive

---

[2] See Appendix A for a description of seven-number summaries.

TABLE 3. Distribution of Participant Ages in Mental Rotations Reference Set

n = 509, mean = 21.2		
	20.0	
19.0	(20.0)	21.0
18.0	(20.5)	23.0
5.0	(34.0)	63.0

TABLE 4. Distribution of Response Times in Mental Rotations Reference Set

n = 509, mean = 1.027		
	1.033	
0.813	(1.032)	1.252
0.647	(1.039)	1.431
0.022	(1.231)	2.441

TABLE 5. Distribution of Accuracy Rates in Mental Rotations Reference Set

n = 509, mean = 0.821		
	0.844	
0.750	(0.844)	0.938
0.635	(0.802)	0.969
0.344	(0.672)	1.000

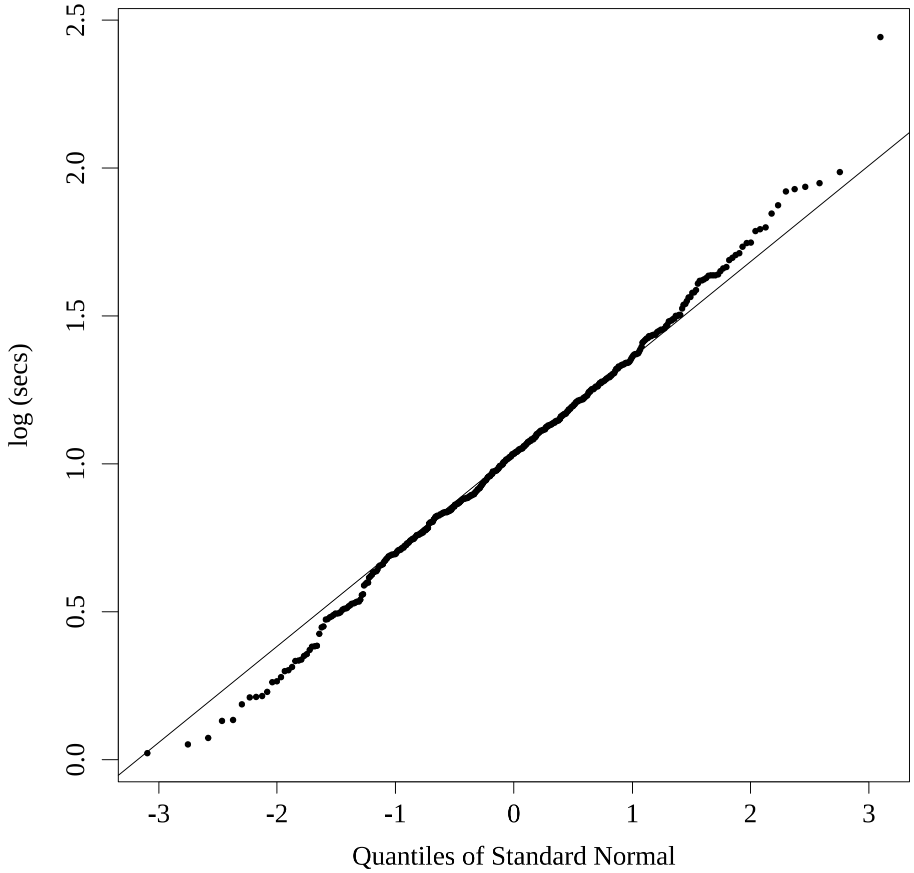


FIGURE 3. Mean Log Response Time for 509 Participants from Reference Set.

end, and the distribution is slightly heavy-tailed, but overall it is very close to normal. Figure 4 shows a normal-quantile plot of the distribution of percentage correct for the reference set. Within the constraints of the discrete nature of this measure, and a ceiling effect (seven per cent of the participants had perfect accuracy), the resulting distribution is reasonably close to normal.

The time and accuracy measures are effectively uncorrelated (Pearson product-moment correlation $r = .072$, see Figure 5), so they potentially measure two different

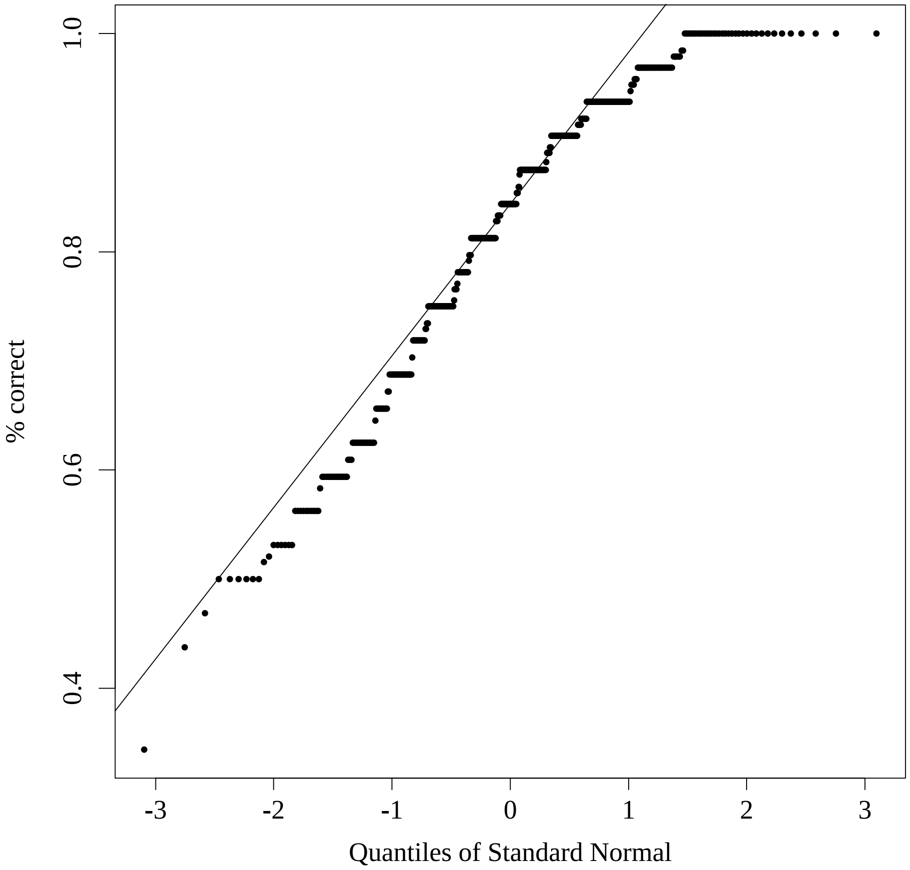


FIGURE 4. Percent Correct for 509 Participants from Reference Set.

aspects of spatial skills. Scales were constructed for both measures in the following way. The percentile for each participant for the time score in this study was computed as the percentage of individuals in the reference set with scores greater than or equal to that of the participant. Note that the resulting time percentile must be interpreted with care: A higher percentile means a *lower* (faster) time. An accuracy percentile scale was constructed similarly, except the percentile represents the percent of scores less than or equal to that of the participant. I will use these two scales as covariates in Experiments 1 and 2, examining how well mental rotations skill predicts individual performance.

Experimental Design

The experiment was a 2 x 2 within-subjects design with display modality and block as the factors. Display modality had two levels, vision alone and vision plus haptics. There were two blocks. Participants performed an absolute magnitude estimation protocol (Gescheider, 1997, pp. 248–255) to estimate curvature. Magnitude estimation has been successfully applied to haptic curvature perception in previous studies with physical stimuli (Gordon & Morison, 1982, Exp.5). Within each block, participants performed 28 trials using one display modality, followed by an equal number of trials in the other modality. Order of first modality within the block was counterbalanced across participants. Each stimulus was presented twice in each block, in random order, and randomly assigned one of 14 possible orientations according to an algorithm described below.

The dependent variables were response time and estimated magnitude of curvature for each trial.

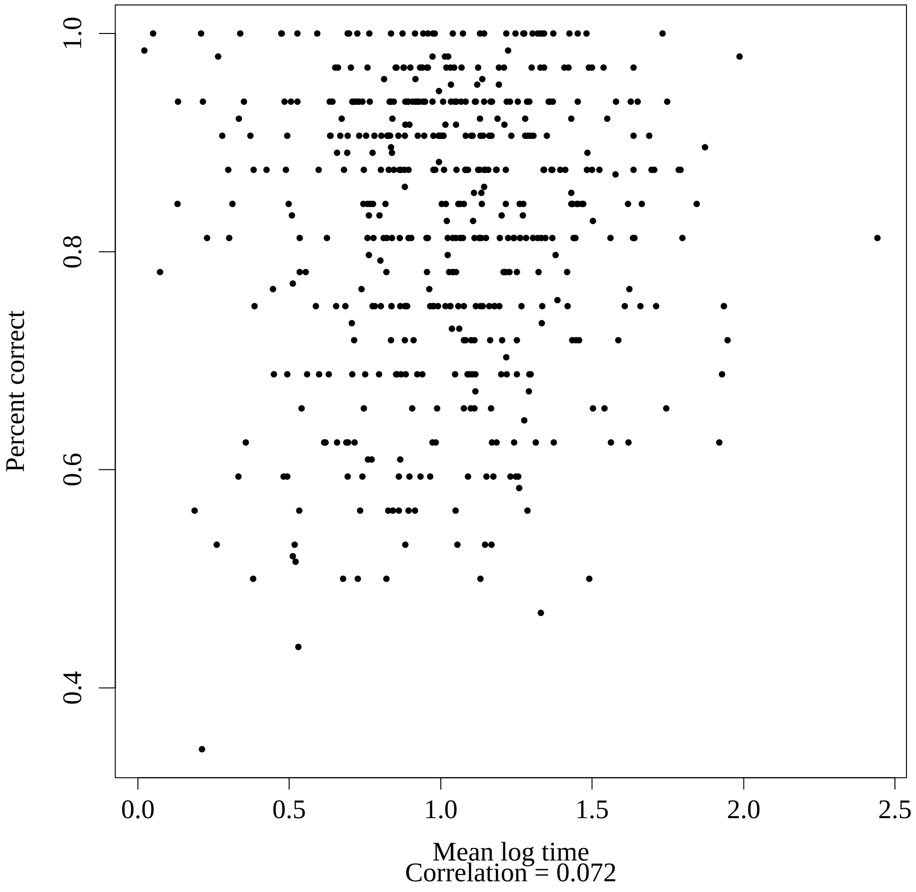


FIGURE 5. Response Time vs Accuracy for Participants in Reference Set.

Experimental Hypotheses

This experiment tests the claim that the extra cues provided by haptics will increase discriminability over vision alone. Using the argument of Lederman and Klatzky (1999), this claim becomes operationalized as

> ### Hypothesis 3.1
>
> The mean slope of the psychophysical functions for observers in the haptics plus vision category will be greater than the mean slope for observers in the vision alone category.

In addition to greater discriminability, the increased cues provided by haptics may help observers make more consistent estimates. This would result in the estimates lying more closely together, decreasing the variance. This becomes operationalized in terms of regression as

> ### Hypothesis 3.2
>
> The r^2 for the regressions of the psychophysical functions for observers in the haptics plus vision category will be lower than those for observers in the vision alone category.

While in general, it seems reasonable to expect that better skill in mental rotations will lead to better performance, this is impossible to precisely operationalize given the limited nature of the current data. The scale derived above has never been used, there are two values (time and accuracy) for each participant, and two dependent variables (discriminability and r^2). Since there is no strong theory on which to base a priori hypotheses, I only do a post hoc analysis on the mental rotations data.

Method

Participants

Participants were 12 unpaid volunteer computer science graduate students from the University of Oregon. Ages ranged from 24 to 48 with a median of 30.5. All reported themselves as right-handed, and 9 out of 12 scored in or above the 6th decile of right-handedness in the Edinburgh Handedness Inventory (Oldfield, 1971), with a median decile value of R-7 (i.e., as right-handed as 70% of the right-handed population). Eight participants were male and four female.

Stimuli

Stimuli were displayed without any environmental cues such as a ground and horizon. While environmental cues are invaluable for determining the distance of an object from the observer and the relative distance between objects (Gibson, 1979), they are not useful for determining the shape of a single unknown object. Only one light source was used. Hubona, Wheeler, Shirah, and Brandt (1999) found that two light sources decreased performance in several spatial tasks, and results of my pilot studies also indicated that multiple light sources made the shapes more difficult to interpret. Table 6 lists the OpenGL parameters used to graphically display the stimuli.

Texture is an important visual shape cue. Wanger, Ferwerda, and Greenberg (1992) found that texture increased accuracy a small (3.3%) but significant amount in a task matching the size of spheres. Texture is also used in estimation of degree of planar slant (e.g., Ernst et al., 2000). Its influence on perception of curved shape will probably be somewhat larger. Nonetheless, I chose to display these stimuli without graphic (or

TABLE 6. OpenGL Parameters for Display of Stimuli

Parameter	Value
Ambient light	
GL_LIGHT_MODEL_AMBIENT	.4, .4, .4, 1
Directional light	
GL_POSITION	−35, 0, 35
GL_SPECULAR	.6, .6, .6, 1
GL_DIFFUSE	.6, .6, .6, 1
Surface reflectance	
GL_AMBIENT_AND_DIFFUSE	0, 1, 0, 1
GL_SPECULAR	1, 1, 1, 1
GL_SHININESS	100
Projection matrix	
gluPerspective parameters	45, 1.46, 5, 65
Model view	
Viewer distance	-35

force) texture. Adding texture might have created a ceiling effect, where observers could visually estimate the curvature so accurately that no further cues were useful. Given the early nature of this work and the lack of data on human performance in point force environments, for this first work I chose to use simple stimuli without texture cues. Future work can examine interactions between more complex shapes and different shape cues.

The stimuli for the magnitude estimation task ranged from 60 m^{-1} to 255 m^{-1} in curvedness in steps of 15 m^{-1}, for a total of 14 values. While these curvedness values are useful for relative comparisons, they may not have validity as physical measurements. The value of curvedness is scale-dependent. Given an object, changing the scale of the coordinate system by which the object is measured will change the value of its curvedness. For the haptic display of a stimulus, the user's kinesthetic experience of the stimulus anchors it to a specific coordinate system and hence a measurable curvedness. This is not the case, however, for the visual display. As described above, the stimuli were

TABLE 7. Parameters for Surface Polygonizer

Parameter	Value
function	`z - 0.5 * (k*x*x + k*y*y)/300`
size	1
zLimit[a]	-5
starting x, y, x	0, 0, 0
mode	TET

[a] See text for description of this parameter.

not displayed with any indication of a specific depth and the display hardware provides no vergence and accommodation cues. Consequently, the user was free to define an idiosyncratic visual coordinate system, and hence an idiosyncratic visual curvedness. Given that haptic curvedness and visual curvedness are most likely discrepant, it is hard to say what the curvedness of these stimuli is in absolute terms.

Different representations were used for the graphical and force displays of the stimuli. The graphical representation was a polygonal mesh computed using the implicit surface polygonization routine of Bloomenthal (1994). Table 7 lists the parameters for the routine. The routine was slightly modified to use a different termination criterion for the polygonization. Instead of computing a fixed number of cubes from the starting cube, the algorithm created polygons up to the z value specified by `zLimit`. While this criterion will not terminate the algorithm for an arbitrary shape, it is guaranteed to terminate for the stimuli in Experiment 1. The variable `k` was set to the stimulus curvedness, while the constant 300 was used to produce an appropriate stimulus size for the display. The haptic stimuli were rendered using a custom-written subclass of `gstShape` in the GHOST 2.1 toolkit (SensAble Technologies, 1998).

Each stimulus was presented in one of 14 possible orientations. An orientation was specified as a rotation around the x and y axes. Orientations were randomly assigned to stimuli. All of the orientations provided some side view of the object. None of the orientations showed a stimulus head-on, which would have made curvature difficult to visually estimate. Every orientation was used exactly four times in a block. Thus a stimulus would nearly always be shown at four different orientations in the same block (although there was some small possibility of the same orientation getting assigned two or more times for the same stimulus) and most of the eight total presentations of a stimulus in the session were with different orientations (although the exact proportion varied between participants). Figure 6 shows a sample stimuli at two different orientations.

Stimuli were displayed with a white curve along one of the parabolas passing through the origin. The rotation of the curve varied, but was always clearly visible (i.e., the curve was not placed on the silhouette of the object). This curve provided a reference for the curvature estimation task (described below). The rotation of the curve was specified as part of one of the 14 stimulus orientations.

Equipment

The experimental program ran on a 300-Mhz Pentium II with Windows NT 4.0. The graphic display device was a monoscopic color screen and the force display device was a PHANToM model 1.5. The haptic rendering loop ran at 1000 Hz and consumed approximately 22% of the processor time. The graphics loop ran with a mean cycle time between 55 and 95 msec, depending upon the curvedness (and hence the number of polygons) of the stimulus. Note that the only part of the graphic display that moved was the PHANToM cursor—the stimulus never moved. For each trial, the program recorded the time from initial display of the stimulus until the participant ended the trial

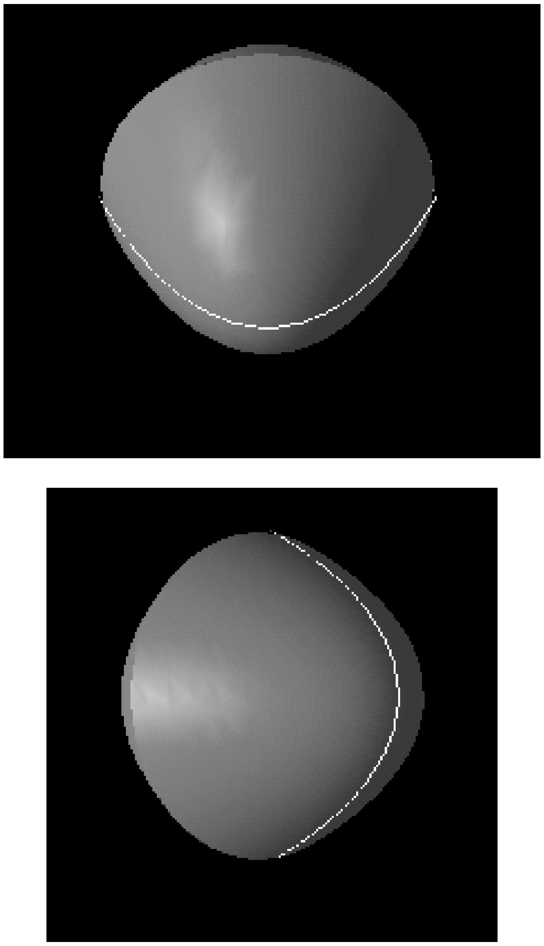


FIGURE 6. A Stimulus, Curvedness = 5, at Two Different Orientations.

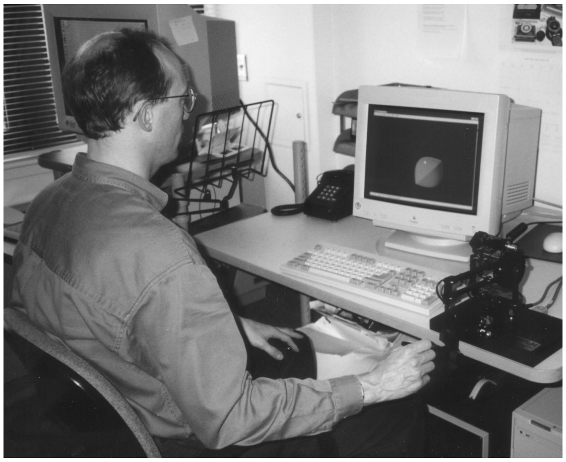


FIGURE 7. Setup for Experiment 1.

by pressing the PHANToM stylus.

Participants sat at a computer table holding the PHANToM stylus in their dominant hand and entered numeric magnitude estimates with their non dominant hand. Figure 7 is a photograph of the experimental setup. At the beginning of the experiment, participants adjusted the chair height to a level where the keyboard and PHANToM could be comfortably held. The elbow of the arm holding the PHANToM was supported by the arm of the chair.

Task

For the experiment, participants were asked to “estimate the curvature of the object along this white line”. The task was organized as an absolute magnitude estimation protocol (Gescheider, 1997, pp. 248–255) and used a slightly modified wording from the sample on p. 254 of Gescheider (1997). In this protocol, participants are asked to make a separate judgment of each stimulus magnitude, without regard to any set scale

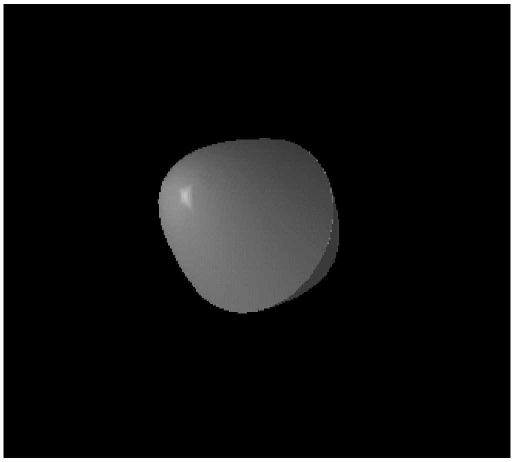


FIGURE 8. Example Stimulus of “High” Curvature.

and independently of any previous estimations they have made. Participants are free to choose any scale they deem appropriate. Studies have shown that different participants will make relatively similar estimates for the same stimulus, within about an order of magnitude of each other (Zwislocki & Goodman, 1980). The absolute magnitude protocol has the paradoxical effect of reducing the variability of estimates between participants by allowing them to choose their own scale.

To anchor the scales of all participants in a common direction, during instruction the participants were shown printouts of Figure 8 and Figure 9 and told that a higher number should be assigned to Figure 8[3]. To avoid biasing their judgments, participants were never told any example magnitude values.

---

[3] Overall, this instruction worked well, as eleven out of the twelve participants used the requested direction for their scales. One participant, P10, used the opposite scale (where Figure 9 received a higher value). I transformed P10’s data so that the slopes of the psychophysical lines (in log-log space) for each block were negated and then used the transformed data for P10 in all subsequent analyses. The transformation appears benign, as the slopes of this participant’s lines are close to the medians for the entire participant group.

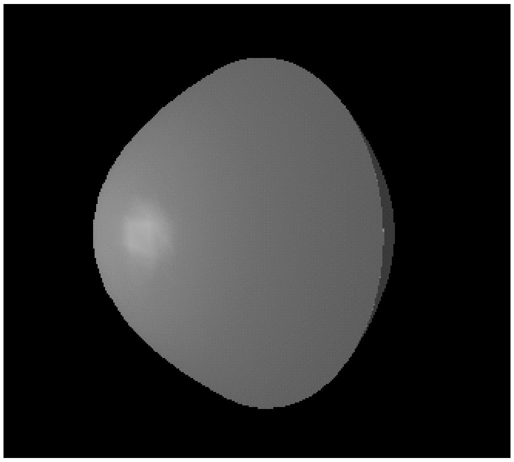


FIGURE 9. Example Stimulus of “Low” Curvature.

Depending upon the current experimental condition, the participant could explore the stimulus using either vision alone or both vision and haptics. In the vision-only trials, participants simply estimated curvature by looking at the object. In the vision plus haptics trials, they were asked to run the PHANToM cursor over the white line at least once, moving at whatever speed felt comfortable. Participants were told they could move over the trail several times and also over other regions of the surface. They were reminded, however, not to spend a long time on any given estimate. Appendix A has the exact experimental instructions.

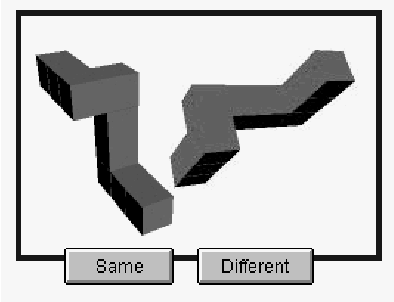


FIGURE 10. Sample “Same” Trial in Mental Rotations Test.

Procedure

Participants began the session with a slightly modified[4] version of the PsychExperiments (1999) mental rotations test. The test consisted of pairs of three-dimensional stimuli. Participants were asked whether the two stimuli were the same (although perhaps at different rotations) or different. Figure 10 and Figure 11 show trials with “same” and “different” stimuli pairs, respectively. Participants responded by clicking on the appropriate screen button. The software recorded the correctness of the answer and the response time. Each block consisted of “same” and “different” pairs for each of the eight rotations from 0 to 315 degrees in increments of 45 degrees, for a total of 16 trials per block. Presentation order was randomized for all participants. This test consisted of two blocks.

Participants were then instructed in the absolute magnitude estimation task. Participants performed several practice trials in the vision and vision plus haptics conditions. After two or three practice trials in each condition, the testing phase of the exper-

---

[4] The original was designed to be a standalone test and so had a sequence of preliminary demographic questions and a consent form. Since the version used in this experiment was simply a pretest, these were removed. Second, to protect the confidentiality of my participants, their responses were never sent to the central database at the University of Mississippi.

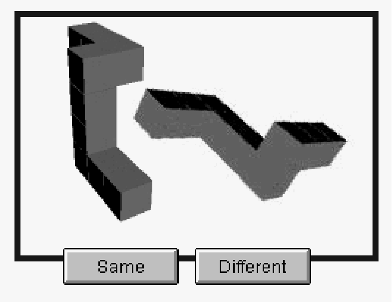


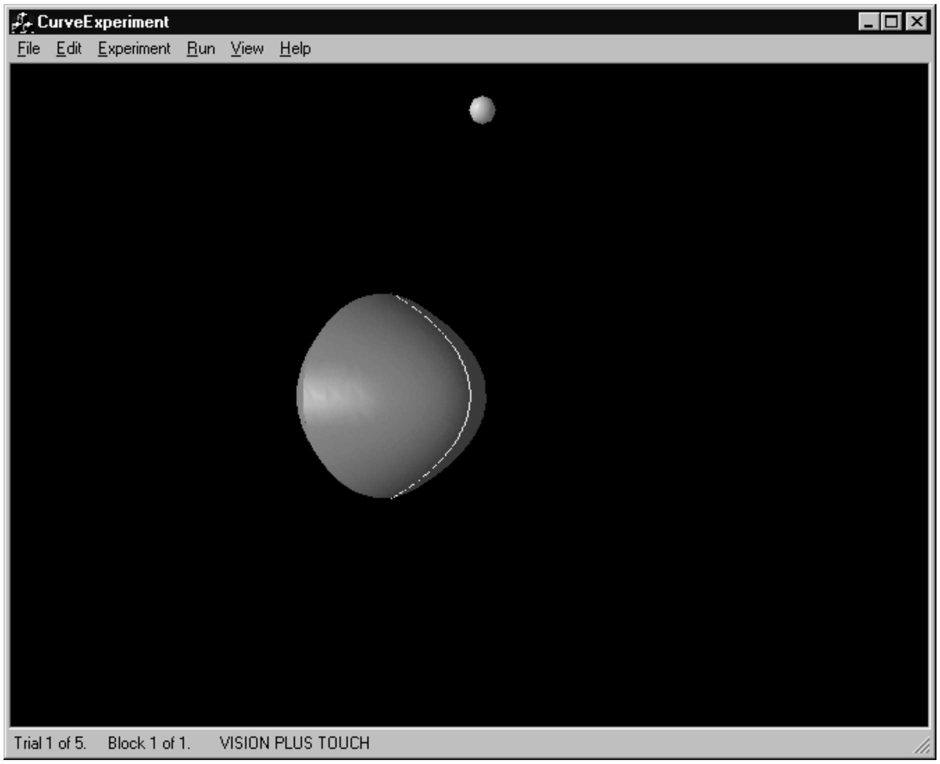
FIGURE 11. Sample “Different” Trial in Mental Rotations Test.

iment began.

The testing session consisted of two blocks of 56 trials each, alternating between 28 vision only trials and 28 vision plus haptics trials. The starting mode (vision only or vision plus haptics) was counterbalanced across participants.

Participants began a trial by clicking the stylus of the PHANToM, causing a stimulus to be displayed both visually and (in the haptics trials only) haptically. Once they had estimated the curvature magnitude, they clicked the stylus button again. The stimulus was turned off and a dialog box appeared requesting an estimate of curvature. Any positive number was accepted as an estimate. After the dialog box closed, the screen remained blank until the stylus was clicked again, starting the next trial. The task environment included a display at the bottom of the window indicating the trial number, block number, and current mode (“VISION” or “VISION PLUS TOUCH”). See Figure 12. The sphere above the stimulus is the cursor indicating the current PHANToM position.

After all trials were completed, participants were interviewed about their experiences during the experiment.



Trial 1 of 5. Block 1 of 1. VISION PLUS TOUCH

FIGURE 12. Screen Layout for Experiment 1.

(unit=.1)

Vision only		Vision plus haptics
9	2	9
5	3	
4	4	36
8	5	3379
168	6	05
0	7	
5	8	4
	9	
13	10	8
	11	
1	12	0

FIGURE 13. Psychophysical Slopes for All Participants for the Two Conditions.

Results

Psychophysical Functions of Curvedness

Magnitude estimates were analyzed using the method described in Lederman and Klatzky (1999, Sect. 4.2). The arithmetic means of the raw curvature estimates were computed for each participant in each block. The grand mean for all participants and blocks was then calculated. The raw score for each estimate was normalized by dividing by the specific mean for that observer and condition and multiplying by the grand mean. Linear regressions of the natural log of the normalized scores against the natural log of the stimulus curvature were computed for each participant for each block. The slopes for each participant are displayed in a back-to-back stem plot[5] in Figure 13.

A repeated-measures analysis of variance was conducted on the slopes of the regression lines, with block and mode as within-subjects factors. In all analyses done on

---

[5] See Appendix A for a description of stem plots.

this data, significance was set at 5%. Block was highly significant ($F_{1,11} = 14.804, p = .003$), with mean slopes of .725 for the first block and .625 for the second block. Mode approached significance ($F_{1,11} = 3.428, p = .091$), with mean slopes of .700 and .649 for vision and vision plus haptics, respectively. Note that this is the opposite from the direction predicted by Hypothesis 3.1—performance was better with vision alone. The block by mode interaction effect was non-significant and its size insubstantial ($F_{1,11} = .266, p = .616$). This indicates that rates of learning for the two modes were essentially identical.

The r^2 values for the regressions indicate the degree to which the log of the intensity predicted the log of the participants' estimates. The r^2 values for the each modality represents the consistency of the modality. For this case, each data point in the regression was the mean of four estimates made by the participant for the same intensity (two estimates in each block for each modality). The resulting r^2 values for each participant in each mode are displayed in a back to back stem plot in Figure 14. The mean r^2 values were .78 and .75 for the vision and vision plus haptics modes, respectively. No significant difference was found between the distributions of the r^2 values for the two modes (Wilcoxon signed-rank test, $n = 12, V = 52, p = .339$). Thus Hypothesis 3.2 was not supported.

As a robust check on the regression analysis, Figure 15 is a plot of the normalized responses for all participants together with a nonparametric loess fit (Cleveland, 1993, Sect. 3.2) to the points. The plot supports the conclusion that the slope for vision is slightly higher than vision plus haptics. Both the spread of the data points and the slopes of the fits for the two conditions are very similar. The loess curves are also nearly linear, indicating that a linear fit is appropriate for this data, although there is a slight

(unit=.1)

Vision only		Vision plus haptics
	3	7
	4	
	5	8
355	6	
04	7	1346
224679	8	01458
0	9	1

FIGURE 14. Regression r^2 values for All Participants for the Two Conditions.

tapering of the slopes at curvedness values above 150 m^{-1}.

Trial Times

The times required to perform each estimate were distributed log-normally. A repeated-measures analysis of variance was performed on the log of the times, with block and mode as within-subjects factors. Again, block was significant ($F_{1,11} = 6.918, p = .023$), with geometric means of 5.71 and 4.90 seconds for the first and second block, respectively. Mode was highly significant ($F_{1,11} = 65.969, p < .0001$), with geometric means of 2.805 and 9.977 seconds for the vision only and vision plus haptics modes, respectively. The block by mode interaction effect was non-significant ($F_{1,11} = 1.83, p = .203$), although larger than the block by mode interaction for psychophysical slope.

In interviews after the experiment, some participants reported that they found haptics of greater use when the stimulus was rotated in such a way that it was seen more head-on than from the side. Statistically, this would be expressed as an interaction effect between the two factors of rotation and mode. This reported effect would be small at best, because the rotations for the stimuli had been selected in such a way that they were

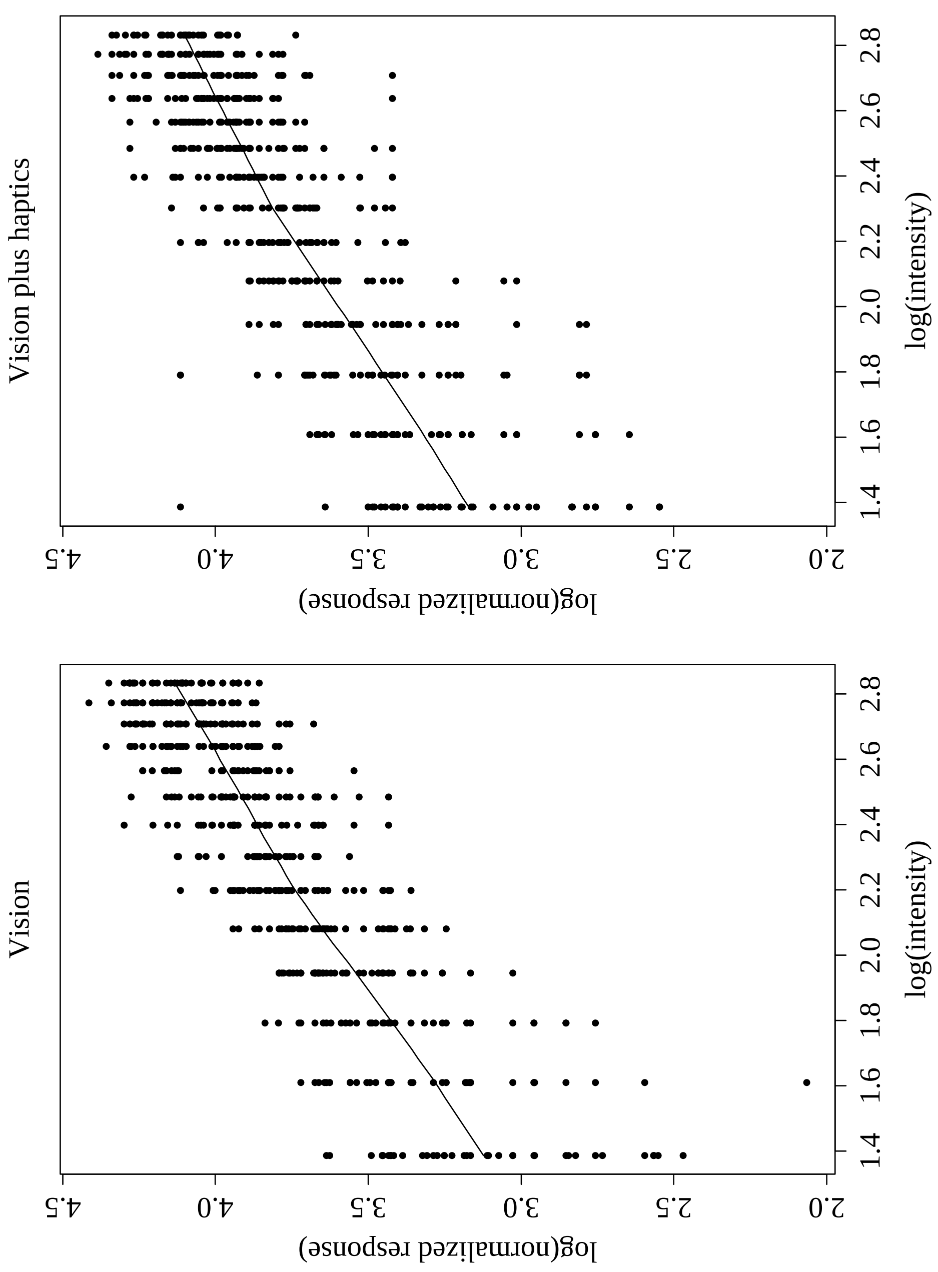


FIGURE 15. Normalized Estimates and Loess Fit.

always seen at least somewhat from the side. Computing separate psychophysical functions for each rotation for each mode did not reveal any effects for rotation. However, slopes for two of the fourteen rotations were consistently apart from the others. The rotation x=45, y=-45 had a slope consistently higher than the others, while the rotation x=-45, y=90 had a consistently lower slope. These results suggest that view of the object may have a small effect on participants' ability to estimate curvature, but they must be taken with caution, since they are both post-hoc and currently lacking any theoretical foundation.

Mental Rotations Covariate

The participants in this study represent a clear subpopulation of the mental rotations reference set. Figure 16 shows boxplots of the scores for all 26 participants for Experiment 1 and Experiment 2 (described in Chapter IV). Both scores are clustered far from the 50th percentile, with the participants in these experiments being slower and more accurate than the individuals in the reference set. As with the reference set, data for the participants in Experiments 1 and 2 showed no correlation between correctness and time (Pearson's product-moment correlation $r = -.035$).

The influence of mental rotation skill on magnitude estimation of surface curvature was investigated using linear regression. Fitting mean response time to mental rotation percentile scores did not produce a useful model; only 3.6% of the variance was explained. Fitting mental rotations percentiles directly to the psychophysical slopes produced a nonlinear fit, but taking the log of the slopes normalized the residuals and produced a better-fitting result:

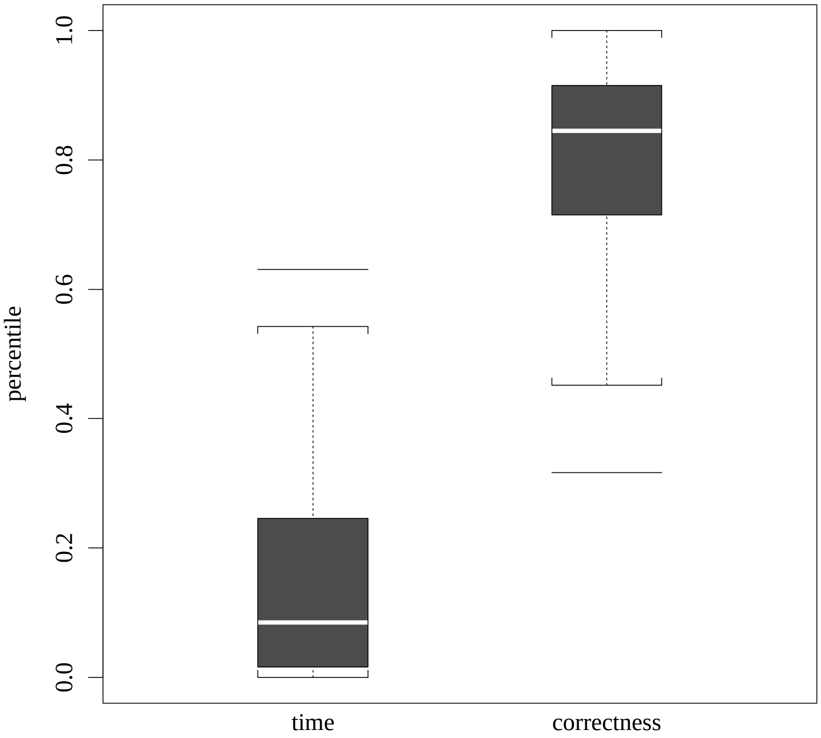


FIGURE 16. Percentiles on Mental Rotations Test for all 26 Participants in Experiment 1 and Experiment 2.

$$\log(slope) = -.514 + .958mrt - .227mrc \tag{3.5}$$

where mrt and mrc are the mental rotation response time percentile and accuracy percentile, respectively. The resulting formula has an R^2 of .195. Of the two coefficients, only the response time is significant ($t(45) = 3.156, p = .003$). The implication of this result is that participants who performed faster on the mental rotations test (recall that higher percentiles indicate faster speeds) discriminated curvature values more than those whose mental rotation times were slower.

Discussion

I will discuss the results of this experiment in terms of the apparent lack of synergy between haptics and visual cues for this task, the possible role of individual differences in strategies and spatial skills, and the usefulness of this task for evaluating haptic environments.

Lack of Synergy Between Perceptual Systems

The addition of force display did not improve the ability to discriminate curvature and may even have reduced that ability. Even though the reduced depth cues of the monoscopic display make the stimuli somewhat ambiguous, the visual display apparently provided enough cues that participants could perform curvature estimation. In the experimental debriefings many participants commented that they made an initial estimate from the first visual glance. While four said that their initial estimates were later revised based upon the haptic experience, the other eight said haptics provided no new information and two even referred to haptics as a “nuisance” or “distraction”. The strong effect of haptics on time—a threefold increase—indicates that participants were in fact using the PHANToM to explore the stimulus for a considerable time, but this increased time did not improve their ability to discriminate curvatures. Evidently, curvature estimation can be performed rapidly and accurately enough from visual cues that further time observing the stimulus does not improve discriminability.

Participants may even have experienced a slight decrease in their ability to discriminate curvature when using the PHANToM. While the result only approached significance, it is based upon enough data points to warrant some analysis. Converting the respective slopes back to the original scales with an exponential transformation, the ranges of the estimates are 4.63 and 3.83 for the vision-only and vision plus haptic conditions, respectively. The range of estimated curvatures is therefore 17% higher for the vision-only condition versus the vision plus haptics condition.

What factors might underlie this? Perceived curvature magnitude is a fundamentally subjective quantity. There is no “correct” value against which to measure participants’ responses. Furthermore, the r^2 values indicate there was a fair amount (roughly 22%) of remaining within-subject variability to the magnitude estimates of the vision-only condition. In principle, at least, participants could have visually discriminated curvatures more clearly than they did. The results of this experiment suggest that haptic experience did not provide enough extra information for that improvement to occur.

The analysis of perception of virtual environments in Chapter II suggests discussing this outcome in terms of the perceptual cues available as resources to support the task. From this point of view, we need to understand what cues might have been available from visual experience and what cues might have been available from haptic experience.

I consider the visual cues first. Comparing the sample stimuli in Figure 9 and Figure 8 more closely, we see that there are multiple redundant pictorial cues suggesting the curvedness. The curvedness affects all aspects of the stimulus' appearance: its size, its silhouette, even the size and sharpness of the specular reflection on its tip. The visual display provides precisely the kind of rich, redundant cues that have been demonstrated to support accurate perception. It is possible that participants could accurately estimate the "curvedness" of these stimuli simply by considering them as two-dimensional shapes, without accounting for depth cues at all. Indeed, the vergence, accommodation, and binocular disparity cues from the monoscopic display would all support such an interpretation. Given such a rich set of visual cues, haptic cues would have to provide a radically enriched experience before there would be a significant improvement in discriminability. However, point force devices instead provide a very small number of new cues.

Philips and Todd (1996) found a similar effect for a visual shape discrimination task using five shapes from the Koenderink (1990) family (including the $S = 1$ paraboloid used in this experiment). Observers had much greater difficulty distinguishing the shapes when their field of view was restricted to 2 ° than when it was 3 ° or 4 °. Philips and Todd concluded that observers in the 2 ° condition had insufficient access to the global structure of the shape and had to discriminate based upon estimates of the principal curvatures. Observers with the wider fields of view could use estimates of the overall shape and performed far better. This is analogous to the results in my experiment, where observers appear to have found global structural cues far more useful than estimates of local curvature.

The results of this experiment indicate that the haptic experience was not that rich and may even have detracted from performance. I suggest that performance was in fact reduced and this reduction was caused by the temporal integration imposed by the point force display. In haptic trials, participants reported that they typically made initial estimates from visual assessment, then traced the white line with the PHANToM. As shown in the Klatzky, Loomis, Lederman, Wake, and Fujita (1993) study, identifying a shape with a point force requires far longer than identifying the same shape with the whole hand, and presumably requires far more cognitive resources because the representation of the shape must be constructed by temporal integration. Consequently, participants in this experiment may have gone through a three-step process. First, they quickly formulated a visual estimate, then they performed the cognitively demanding task of temporally integrating the shape, potentially disrupting the experience of their initial estimate, and finally had to reformulate their estimate, the original having been lost in the disruption. As one participant put it, "after fiddling with all that [the PHANToM], *then* I would say, 'OK, now look at it and estimate.' "

There are currently no studies examining the consequences of protracted practice on temporal integration, so it is currently not possible to say whether the cognitive load reduces or whether the speed increases with practice. Within the limited number of trials in this experiment, some learning did occur (Figure 17). The geometric mean haptic trial time in the second block was 82% of that for the first block. This greater speed was accompanied by a reduction in discriminability, however, as the arithmetic mean slope went from .725 in the first block to .625 in the second. Thus it is impossible to say how much of the time improvement was due to simply increased motor skill with the PHANToM and how much due to increased skill of temporal integration.

Overall, the difference in mean slopes of the psychophysical functions for the

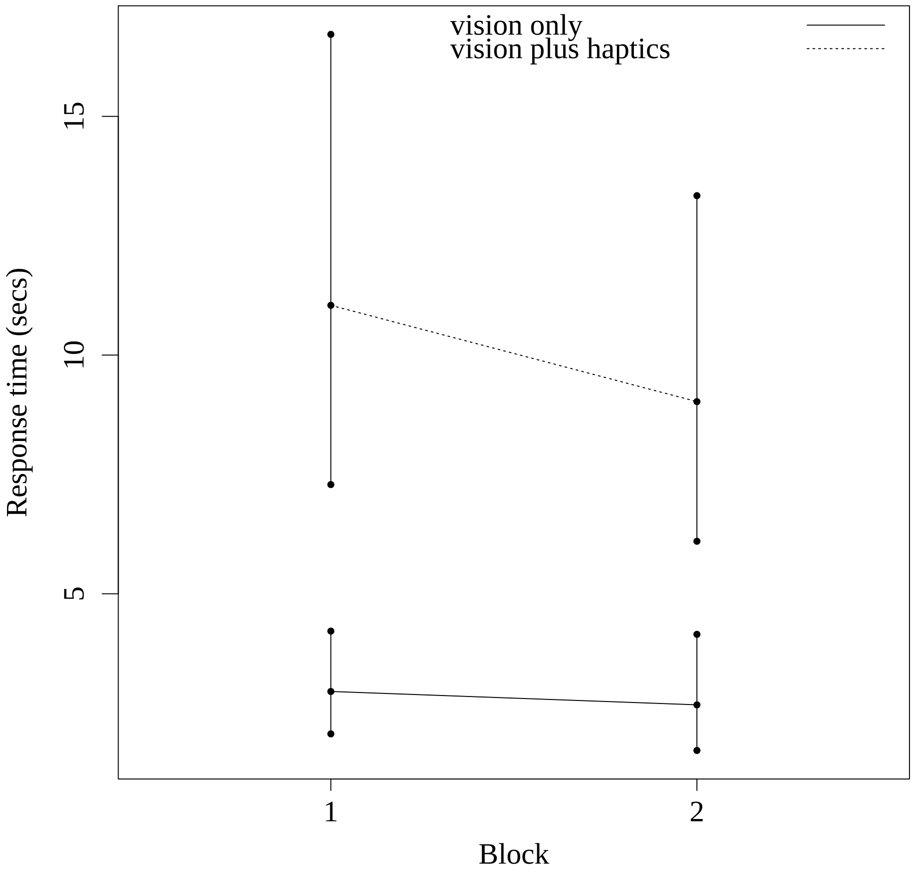


FIGURE 17. Change in Response Time Between Blocks (With 95% Confidence Intervals).

vision and vision plus haptics conditions is large enough to warrant further study to see if the effect is robust and, if robust, what factors might be causing it. In particular, the effects of temporal integration should be examined more closely.

Individual Strategies and Individual Differences

Individuals might use different perceptual cues for a given task. Despite the overall poor showing of haptics in this experiment, perhaps certain classes of individuals found the cues useful. Performance of individual participants might have varied depending upon the conscious strategy they used or different cognitive abilities.

To assess the role of individual strategy, participants were asked after the experiment whether they thought the haptic experience was useful. Five stated they thought it had helped to some degree and had some strategy for using it, while four thought it had reduced their performance and had paid little attention. Note that while the participants were naive to the specific hypotheses of this experiment, as members of the Department they most likely assumed that the experimenter was hoping to find positive results for the haptic condition. Thus, their responses are likely to be biased. Because of this, and because of the potentially biased coding of these responses (the questions were free form and the experimenter coded the responses himself), I evaluated the accuracy of these responses using an informal rule of thumb rather than a formal inference technique.

I defined an absolute difference of greater than .05 between the visual and visual plus haptic slope to be “substantial”. Using this definition, only three of those who thought they had done better with haptics in fact had psychophysical slopes that were substantially greater in the vision plus haptic case, while none of those who thought they had done better with vision actually had psychophysical slopes that were substantially greater in the vision case. In fact, three of those who claimed to have done better with vision in fact had substantially greater slopes in the vision plus haptics case.

In short, participant reports were not reliable predictors of the condition in which they were able to more clearly discriminate curvatures. This suggests that whatever differences existed between participants, it was not due to consciously selected strategies for curvature estimation. At this time, I am not sure if there is any value in collecting participants' reports of their personal strategies, since the responses are so unreliable.

Spatial abilities, as measured by a simple computerized test of mental rotations of three-dimensional objects, are an objective means of characterizing individual differences. For this participant group, the mental rotations test predicted a useful fraction of individual performance. While explaining 20% of the variance is not huge, it is still substantial, given the many factors underlying human performance on a task as complex as this. However, given the multivariate nature of the relationship, and the somewhat unusual nature of the discovered relationship, there is a real possibility it is only a chance effect. The results in this experiment must be replicated in different contexts before any firm conclusions be drawn. Nonetheless, the test is simple and quick, and I recommend it be included in future studies of haptic environments.

Comparison with Comparable Physical Tasks

Comparing the results of this experiment with the literature of quantitative results on haptic curvature perception suggests some reasons why the haptic cues did not provide much supplementary information in this task. All the haptic studies before this one used haptics without vision.

The slope of the psychophysical function in this experiment can be compared with one derived for physical stimuli. Gordon and Morison (1982) computed the psychophysical function for estimating spherical curvature over a 6-cm distance using the middle finger. For curvatures ranging from 1.44 to 5.73 m^{-1}, the slope was 1.2, while for curvatures ranging from 1.91 to 19.12 m^{-1} the slope was 0.9. The stimuli used in the experiment described in this chapter were paraboloids rather than spherical, so their curvedness values cannot be immediately compared to the curvature values of the older experiment. Using the rough curvedness to curvature conversion function described earlier, the range of “curvatures” in this experiment was from 30 to 127.5. The slopes of the psychophysical functions for these curvatures were 0.70 (vision alone) and 0.65 (vision plus haptics). These results are consistent with the earlier work, given that Gordon and Morison (1982) found that larger ranges of curvature produced a lower slope and extrapolating their results to the curvature range used in this experiment. The above comparison is of course extremely rough, as Gordon and Morison used a haptics only protocol while the estimates in this experiment appear to have been predominantly visual.

The prior curvature estimation studies provide further suggestions to why point force haptics was not a useful addition to vision for the task of curvature estimation. Table 8 lists selected discrimination thresholds from five studies on haptic perception of spherical curvature. The first four used physical stimuli while the fifth used a point force environment. The results from the physical stimuli demonstrate the tremendous role that the structure of the human hand plays in curvature perception. The discrimination thresholds cover two orders of magnitude, depending upon which part of the hand contacts the stimulus and how far it is moved. The first two studies (Goodwin & Wheat, 1992; Gordon & Morison, 1982) show a striking shift—the threshold is an order of magnitude lower when the finger is moved than when it is passive. The third study (Pont, Kappers & Koenderink, 1999, Exp. 3) shows that moving the finger even longer distances results in better discrimination. The fourth study (Pont, Kappers & Koenderink, 1997) shows discrimination thresholds are comparable to those of the third when larger areas of the hand are placed in static contact with a curved region.

At first, two mechanisms might appear to be involved. More refined discriminations can be achieved either by placing a larger hand region in contact with the surface or by moving the fingertip further. However, Pont, Kappers, and Koenderink (1999) argue that these results are due to a single underlying mechanism. Pont et al. found that the most accurate model for the data assumed that the observers used an extremely simple algorithm for computing curvature: Observers compared the *attitudes* (i.e., the angle of the surface against the finger) at the two most extreme available points on the object. The benefit of larger hand surface and longer distance was that both provided more extreme points at which to compare the attitudes, making the attitudes more different and hence increasing discriminability (see Figure 18). While the attitude comparison model was not a complete predictor of performance on these tasks, it was by far the largest factor.

This elegant explanation has important consequences for curvature estimation with point force devices. If the Pont et al. (1999) model is correct, then little to no temporal integration was observed in the four studies in Table 8. Indeed, in the studies where participants actively moved their index finger (a situation similar to that with point force devices) and where temporal integration was a possible approach, observers found it far more effective to exploit the distributed nature of cutaneous finger tip per-

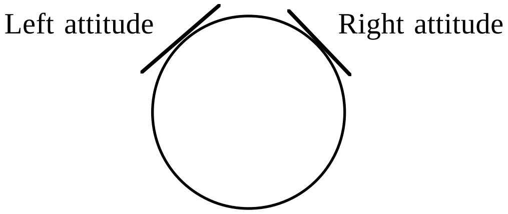
TABLE 8. Reported Curvature Discrimination Thresholds

Study	Contact area	Distance (cm)	Limb moved	Curvature range (m^{-1})	Disc. thresh. (m^{-1})	% for disc. threshold[a]
1	Index finger pad	0	None (static)	286–397 154–211	37 28	75 75
2	Middle finger pad	2	Middle finger	0–5.7	1.8	75
3	Index finger tip	5	Finger plus possibly forearm	(-4)–4	1.25–1.75	84
	Index finger tip	15	Finger plus forearm	(-4)–4	0.5	84
4	Palmar hand, along fingers	0	None (static)	(-1.8)–1.8	0.5	84
	Palmar hand, across fingers	0	None (static)	(-1.8)–1.8	0.9	84
	Dorsal hand	0	None (static)	(-1.8)–1.8	> 2	84
5	PFD	2–16	Hand plus forearm	12.5–100	30[b]	See text

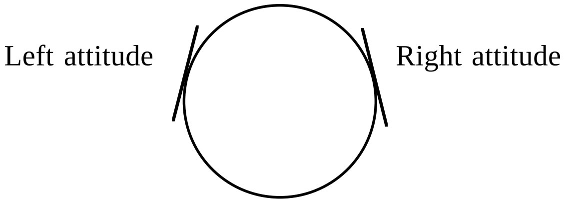
Note. Results from: 1. (Goodwin & Wheat, 1992) 2. (Gordon & Morison, 1982) 3. (Pont et al., 1999, Exp. 3) 4. (Pont et al., 1997) 5. (Tan, 1997)

[a] Different studies reported the thresholds for slightly different accuracy values.

[b] This is not a discrimination threshold. See text for details.



(a) Close attitudes



(b) Distant attitudes

FIGURE 18. Comparison of Local Attitudes at Different Spacings. For a given sphere, attitudes that are further apart on the sphere differ more than closely-spaced attitudes. The more separated attitudes permit more accurate identification of the sphere's curvature.

ception and do a single comparison of sensation at the two most widely-separated points available. In other words, when observers had any alternative, they avoided temporal integration.

Unfortunately, they have no alternatives with a point force device, at least with current interaction techniques. Due to its point nature, there is no way to assess attitude of the stimulus with respect to the contact point. The only way to assess curvature is to compute position changes over time (or a derivative indicator such as velocity or acceleration)—temporal integration. The fifth study (Tan, 1997) in Table 8 suggests that the cost of this will be quite high. Using an absolute identification paradigm, Tan (1997) found that observers could reliably identify 3 to 4 different sphere sizes over the 12.5 to 100 m^{-1} range. Since this is an identification paradigm, it can only provide a rough comparison with the other studies, which used discrimination paradigms. Nonetheless, a coarse comparative figure can be computed by linearly interpolating across the range of curvatures, giving a “threshold” of 30 m^{-1}. This is two orders of magnitude worse than the best thresholds in the other studies, suggesting once again that temporal integration dramatically reduces performance. Note in particular that the range of stimuli curvatures used by Tan (1997) was 1 to 1.5 orders of magnitude larger than those used in the three other studies using hand-sized stimuli. That the performance should be so low despite using much more highly curved (and thus more discriminable) stimuli is a further indicator of the cost of temporal integration.

Some authors have argued that many tasks are most effectively supported (or can only be supported) by haptic devices that display spatially distributed forces (Lederman & Klatzky, 1999). The results of Experiment 1 provide further support for this argument.

Conclusions

This experiment did not find support for either of its explicit hypotheses. Haptics did not significantly improve either discriminability or consistency of curvature estimation for these stimuli, and may even have detracted from performance. While the experiment failed to demonstrate that adding haptics to a virtual environment improves performance, it nonetheless has useful implications for future research in haptic environments.

I believe that curvature is an essential percept in haptic environments. While edges are an important indicator of shape, many objects have large regions that consist of smoothly curving surfaces. Many of the objects of interest in haptic environments, such as the computed isosurfaces of a volumetric data set, will consist entirely of such curved surfaces. If haptics is to provide any useful contribution to the perception of these objects, haptic environments will have to provide sufficient perceptual resources for rapid and accurate curvature perception. Curvature estimation is consequently a useful task for evaluating human performance in haptic environments.

I conceived this experiment as the simplest possible instantiation of such an evaluation task. The stimuli were concisely specified, the task readily described, and the protocol well established. However, the experimental results demonstrate that this simplicity was only apparent, that in fact performance on this task is influenced by an extremely rich set of underlying factors. While the paraboloid stimuli are mathematically simple, they are graphically rich, and my participants used this richness as a resource for extracting multiple visual cues for curvature estimation. In contrast, the point force display provided exactly one haptic cue, curvature. Thus the experiment effectively compared visual perception of many cues, including curvature, with that same multicue visual perception supplemented by the single cue of haptic curvature perception. Given the rich cues already existing in the visual environment, it is not surprising that adding one more cue did not significantly improve performance. In fact, the temporal integration required by haptics appeared to reduce performance.

From the perspective of ecological validity, the above comparison is fair. Ultimately, force displays will only be justified if they provide measurable performance benefits in graphically rich environments. However, I suggest that applying this criterion is premature, given our limited knowledge about human performance in haptic environments. This experiment looked at aggregate metrics of performance—discriminability, consistency, and response time—without an underlying model of visual and haptic curvature perception. I implicitly assumed that the effects of haptics would be so strong that they would be readily demonstrated using such aggregate metrics. In retrospect, this was naive.

I propose that near-term research on curvature perception in haptic environments should set aside such aggregate metrics and focus instead on constructing more detailed models of point force perception of curvature. The two orders of magnitude difference between the results of Pont et al. (Pont et al., 1999, Exp. 3) and Tan (1997) demonstrate that experiments using a single index finger on physical stimuli do not predict results using a point force device for a somewhat comparable task. While prior work on physical stimuli can serve as a guide, an effective theory of point force perception will require a body of basic research results directly obtained from observers using point force devices. The studies on physical stimuli summarized in Table 8 can be a guide to the data required for point force devices.

I specifically suggest the following:

1. Estimation of absolute threshold and difference threshold for curvature perception using a point force device. The threshold may well differ depending upon the range of movement of the hand, so different stimuli sizes should be used and their thresholds compared.

2. Repeat this experiment using a narrower visual field of view. Philips and Todd (1996) found that restricting the visual field to 2° prevented observers from using global shape cues and forced them to rely on visual perception of curvature. In addition to the vision and vision plus haptics conditions used in this experiment, I recommend adding a haptics-only condition. The revised experiment is an attempt to compare the perception of curvature in the three display modalities in the absence of any supplementary visual cues.

3. Explore alternative interaction techniques for curvature estimation. Pont, Kappers, and Koenderink (1999) argue that observers use stimulus attitude as the primary cue for curvature perception with their hands, circumventing the limitations of temporal integration. Is there an interaction technique that might allow point force users to similarly circumvent temporal integration? For example, high frequency force variations might be used to provide a "texture" for the displayed surface (Minsky, 1995) that indicates how much the angle of the tangent plane at the current point differs from that of some reference point. Point force devices that can display torque forces might also permit richer displays of curvature information.

4. The recently developed constrained scaling technique (West, Ward & Khosla, 2000) might be a useful technique for reducing intersubject scale variability in experiments on psychophysical scaling. By providing a standardized scale for the observer's magnitude estimates (the very opposite approach to absolute magnitude estimation), it

is possible to directly compare the *accuracy* of observer's judgements under two conditions, a comparison that is not possible in absolute magnitude estimation.

5. Explore whether performance of curvature estimation with a point force device improves with learning. Can the effects of temporal integration be mitigated by practice?

The above experiments represent a good start at constructing a basic theory of point force curvature perception. With that in hand, we can return to the more complex question of how haptic displays might improve human performance at curvature estimation in more ecologically valid tasks.

This chapter has explored the interaction of vision and haptics in curvature estimation, a subcomponent of shape recognition. The next chapter explores the mechanisms of haptic shape recognition in the absence of vision.

CHAPTER IV

EXPERIMENT 2—SHAPE RECOGNITION

Introduction

The task used in Experiment 1 measured aggregate performance for two different interaction techniques for curvature estimation. The emphasis was upon the performance effects of intersensory integration. In this chapter, I describe an experiment that looks at the detailed mechanics of a single perceptual system, point force haptic perception. As I described in Chapter II, geometric properties must be induced through point force haptics by performing temporal integration. These results from experiments with physical objects suggest that temporal integration is a slow, effortful, and error-prone process. For complex shapes, such as common household objects, the tip of the finger provides insufficient cues to infer shape from one or two points of contact, and temporal integration results in considerably reduced performance (see Klatzky et al., 1993, and the shape-1 condition in Table 1). Unfortunately, temporal integration is the only means by which users of point force displays can perceive shape using currently known interaction techniques. Since temporal integration is so central to shape perception in point force environments, it is the focus of this chapter. The chapter begins with a model of the mechanisms of temporal integration. The bulk of the chapter describes an experiment evaluating the human performance of haptic (only) shape recognition in point force environments. Even though this experiment focuses on haptics, however, possible interactions with vision must still be taken into account.

This chapter contributes to the following elements of the ecological theory of design presented in Chapter II:

1. Demonstration of a second evaluation task for haptic interfaces.

2. Isolation of determining factors of human performance. This chapter is concerned in particular with the influence of visual proprioception and object size on human performance of temporal integration with point force devices.

3. Psychometric assessments of performance. This experiment provides a second example of using a mental rotations test to predict performance on a haptic task.

4. Evaluation of a proposed interaction technique for shape recognition in point force environments.

5. Empirical data on changes in performance over an extended period of practice with a point force environment. Where Experiment 1 explored the use of point force haptics for magnitude estimation of curvature, one aspect of object shape, Experiment 2 explores the overall task of haptic shape recognition.

Haptic Perception of Shape

What is the likely role of haptic perception in point force environments? When haptics and vision are both present in physical environments, there is a strong division of labor: vision is the preferred Exploratory Procedure (EP) for geometric properties, while haptics is used for material properties (Klatzky et al., 1993). As shown in Chapter II, the effectiveness of both perceptual systems is reduced in such environments. However, while vision is likely to continue to be extremely useful for apprehending geometric properties (and Experiment 1 provides an initial demonstration of that), the role of the haptic system is likely to be disrupted, as several factors will reduce the need for material judgments:

1. Some important material properties cannot be displayed at all using current technologies, and other material properties are only partially displayed.

2. Material properties are arbitrary in haptic environments. Whereas for physical objects the material properties are a valid guide to its mechanical structure, objects in haptic environments have no inherent structure. Material properties provide less useful information about their associated objects.

3. Material properties are unnecessary for accurate manipulation. In physical environments, material properties are essential guides to how much force is required to lift an object, how tightly to grip it, and whether it is slipping out of the hand. These are all important for grasping an object, which is the prerequisite for manipulation in physical environments. In point force environments, grasping is not possible and so this information is unimportant.

Given these limitations, if the haptic system is to be useful at all in point force environments, it will be used to apprehend object geometry. There are several kinds of haptic tasks that involve perception of geometric properties. At the highest level, I distinguish between *object recognition* and *shape recognition* tasks. An object recognition task requires that the participant identify an object from some set of well-known objects. The participant is free to use any combination of material and geometric properties. Material properties afford substantial performance enhancements because they can often restrict the range of possible items to a very small range, facilitating rapid identification. For example, the distinctive material properties of styrofoam—its texture, its unusual combination of rigid local structure and compliant global structure, its lack of thermal flow—make it rapidly identifiable. Once the material of an object is identified as styrofoam, the range of possible identifications becomes quite small. Unrestricted bimanual haptic perception can achieve extremely rapid and accurate object identification (Klatzky et al., 1985). However, since material cues will not be useful in point force environments, object recognition tasks are unlikely to be used in them.

Shape recognition tasks require that the participant identify the geometric structure of an object without reference to its material cues. One application of haptic shape recognition is in the analysis of scientific data sets such as seismic data (McLaughlin & Orenstein, 1997). In these applications, the user would be presented with a large, complex, unfamiliar shape and asked to produce a list of features characterizing the shape, such as a list of synclines and anticlines. The likely approach to making such a characterization would be to explore the entire surface, identifying local features, and then assemble these local features into a coherent spatial arrangement describing the overall shape. Biederman (1987) has proposed a similar model for visual shape recognition, *recognition-by-components*. Lakatos and Marks (1999) present evidence that this process occurs in haptic shape recognition as well. They found that observers exploring unfamiliar shapes gave local features more salience in the early stages, while global shape became more salient later. In this kind of shape recognition, the local features would themselves be familiar shapes, identified by fairly simple procedures. Thus the approach would consist of a combination of local EPs specialized for discriminating amongst a small set of local features, interspersed with larger-scale EPs intended to discern the overall arrangement of the local features.

Local feature recognition is a fundamental component of shape recognition, and any performance limits on this task are limits on shape recognition as a whole. Thus local feature recognition is an excellent task for evaluation of performance of point force environments. The task in Experiment 2 (this chapter) is an abstraction of the process of local feature identification. In this experiment, participants are first trained to identify five basic shapes. The shapes can be discriminated on the basis of the *signs* of their two principal curvatures. Unlike Experiment 1, in Experiment 2 participants did not have to estimate the magnitude of curvature, only determine whether a curve was concave, flat, or convex. If the curvature is large enough, this can be done with the simple EP of moving the point of contact along the curve. Two such EPs, performed orthogonally along the surface, are therefore sufficient to discriminate the stimulus shapes. In the experimental task, participants haptically identified those shapes when they are presented at various orientations in space. The shapes represent the familiar local features in the above approach to shape recognition. The task represents the process of recognizing a local feature at some point on the larger shape. Once this process is understood, the integration of these local experiences into a single global arrangement can be studied with different tasks.

Underlying Mechanisms of Reduced Performance in Point Force Shape Recognition

Knowledge of the properties of a single point in space is rarely useful. That knowledge only becomes useful when its spatial relationship to other points of interest is known. For normal haptic experience of physical environments, many of those relationships are sensed at any given moment through kinesthetic awareness of the location of multiple contact points with the object using broadly capable EPs. However, point force haptics only provides kinesthetic awareness of the location of a single contact point. This process has been shown to be slow and unreliable in physical environments (Table 1) and is likely to be so in haptic environments as well.

Both processes are based upon kinesthetic experience. Why is the process using both hands rapid and effortless, while the one using a single finger is slow and effortful? I propose two possible sources of this difference. First, the construction of a representation of shape is based upon simultaneous sensations for broadly capable EPs, a process of spatial integration, while for point force haptics the spatial relationships between points must be computed by comparing sequential kinesthetic sensations over time, a process of temporal integration. Second, the actual form of the kinesthetic experience is in fact different, since different limb systems are used. Each of these is likely responsible for some of the reduction in in performance of point force haptics, and their relative contributions may depend upon the task.

I first consider temporal integration. At the gross level the computation is the same whether the observer is using spatial or temporal integration: A representation of space is constructed from kinesthetic experience. The difference is that one process consists of direct comparisons of two immediate sensations while the other consists of the comparison of an immediate sensation with the memory of a prior sensation. The poor performance of temporal integration probably results from limitations of kinesthetic memory. Current cognitive architectures (Card, Moran & Newell, 1983, pp. 28–31; Kieras & Meyer, 1997) suggest two possible points where these limits might arise. They model memory of perceptual experience as a two-stage process. Sensations are first stored in perceptual memories, which record exact physical parameters of experience for brief periods of time. These experiences are then encoded in a higher, symbolic representation. These symbolic representations are often, although not always (see Kieras & Meyer, 1997, pp. 404–405), presumed to be stored in a different type of memory with longer decay times than the perceptual memories. If the perceptual memory for kinesthetic experience has a very short time decay, a temporally extended kinesthetic experience might be lost before it can be completely encoded in symbolic form. Alternatively, the encoding of kinesthetic experience into symbolic form may be highly approximate. The form of such encoding (if any) is not currently known, but at the very least, the encoding of kinesthetic experience accessible to *conscious* experience is demonstrably rough, as people find it very difficult to provide precise verbal descriptions of kinesthetic experiences. Either or both of these possible limitations of kinesthetic memory, rapid decay or coarse encoding, may account for the limitations in temporal integration of kinesthetics.

The other possible mechanism underlying poor performance in point force haptics is not the memory of kinesthetic experiences, but the experiences themselves. The physical arrangement of the limbs in point force haptics may result in less accurate kinesthetic perception than the arrangement of the limbs in the broadly capable EPs used in haptic perception of physical objects. When multiple fingers from the same hand contact an object, their relative locations can be computed by comparing the positions of each finger. The distances between finger positions are small and constrained by the structure of the hand. This permits the recovery of highly accurate relative locations. By contrast, the current position of the tip of a point force device is only known in terms of the position of the entire arm holding it. The difference can be shown by considering the task of determining the relative locations of two points on opposite sides of a one cm sphere. The crucial factor in accurate kinesthetic measurement is not the Euclidean distance between the points (one cm), but the distance along the lengths of the limbs that must be compared to infer the Euclidean distance. For the case of the index finger and thumb of a typical hand, that distance is the combined length of the two fingers, on the order of 15–20 cm. For the case of comparing those same two points using two successive positions of a point force device, the distance is twice the length of the arm, on the order of 120 cm. As a consequence, even if the memory of kinesthetic experience is completely veridical, the original kinesthetic experience may be so approximate for point force devices that the relative positions of points in space cannot ever be known to any accuracy using kinesthesia alone.

While conceptually distinct, the effects of coarse kinesthetic sensation and coarse encoding of kinesthetic experience in perceptual memory are difficult to separate in practice. For most tasks, they will be indistinguishable in terms of dependent variables. However, it might be possible to tease apart their effects by asking participants to perform broadly capable EPs over time rather than simultaneously. In the case of point force displays, inaccuracies due to the length of the limb would be reduced by only moving the wrist while operating the PFD, while inaccuracies due to memory encoding would be unaffected. This distinction is left for future work. The experiment described in this chapter is intended only to provide initial estimates of the overall effect sizes.

The Effect of Visual Proprioception on Haptic Shape Perception of Shape

When a point force haptic system is apprehending geometric properties, the visual system still has a potentially valuable role, providing proprioceptive feedback for movements. Visual proprioception could potentially ameliorate the effects of either non-veridical kinesthetic memory or inaccurate kinesthetic experience. First, the visual system seems better adapted to recording and comparing the motion of a point through space. The oculomotor system is known to have smooth pursuit mechanisms for visual tracking of moving objects (Rosenbaum, 1991, pp. 178–180) and it is likely there are cognitive mechanisms supporting such a task as well. Therefore, there are probably less limitations in the memory of visual locations than kinesthetic locations. Second, the visual system provides considerably more accurate perception of spatial locations in haptic environments because the graphical display more nearly matches the visual field of the physical environment, although as noted in Chapter II the depth cues are often considerably reduced compared to their physical counterparts. The visual and haptic systems could thus provide complementary data that is combined into a single percept. The haptic system would provide sensation of local forces, while the visual system would provide the sensation of relative location that would be used to place the haptic sensations in context with each other.

Experiment 2 compares the performance of the shape recognition task with and without a graphical cursor. If the above synergy exists, displaying a graphical cursor should improve performance of haptic shape recognition.

The Effect of Shape Size on Haptic Shape Perception

Shape size is another factor that might affect performance of haptic point force shape recognition. The discussion on pp. 116–119 of the mechanisms of shape recognition suggests several possible outcomes. If the performance of point force haptics is limited by the short term of kinesthetic memory, the effect of shape size will depend upon the speed which observers move along smaller shapes. Presuming a constant sampling rate, if they move the contact point at the same or greater speed for smaller shapes, shape recognition should improve because a larger fraction of the shape would be available in perceptual memory for computing the change in curvature. On the other hand, if they move the contact point more slowly, the short term of perceptual memory will be an equal problem for all shape sizes.

If point force haptics is limited by the reduced accuracy of kinesthetic awareness resulting from the use of the whole arm rather than only the fingers, it is likely that this inaccuracy will have a more pronounced effect at the smaller distances used for smaller shapes. Performance with small shapes will consequently be poorer than for larger shapes.

There is also a potential interaction effect between shape size and the presence or absence of visual proprioception. If visual proprioception and point force shape recognition have a synergistic effect producing higher performance, then any deleterious effects of small shape size will be mitigated by the availability of visual proprioception. In point force environments, the interaction technique can be specifically designed to enhance this effect. If the cursor movement is displayed with a larger control-display gain, say one mm of haptic movement moves the cursor three mm on the screen, the effect of visual proprioception should be even more powerful.

Predictions of Effect Direction

On pp. 116–119 I described three possible underlying mechanisms of the poor performance of point force haptics. The mechanisms are not mutually exclusive and it is likely they all contribute to some degree. On pp. 119–121 I discussed the performance effects of the presence of a graphical cursor and the size of the shape. Table 9 lists the predicted direction of effect for each combination of factor and the three possible limitations of point force shape recognition, depending upon the underlying mechanisms. Experiment 2 is not designed to determine the size of each these effects. Rather, the experiment is an exploratory study aimed at determining the direction of these effects.

TABLE 9. Possible Factors Affecting Performance of Point Force Shape Recognition

Kinesthetic mechanism	Factor				
		Vision- small shape interaction	Shape size		
	Vision present		Small distance	Velocity	
				if same	if slower
Perception	↑	↑	↓	—	↓
Memory encoding	↑	↑	↓	—	—
Memory span	↑ or —	↑	—	—	↓

Note. Arrows indicate the predicted change in human performance when the factor named in the column heading is added. Horizontal bars indicate a prediction of no change.

Experimental Design and Specific Hypotheses

Experiment 2 was a 2 x 2 within-subjects design with factors of interaction technique and shape size. The levels of the interaction technique factor were (visual) cursor present or absent. The levels of the shape size were small and large. Participants performed a shape recognition task using a point force device. The shape was never displayed graphically, although a red dot was displayed at the center of the shape. Additionally, in the cursor present conditions a graphic cursor indicated the current position of the PHANToM tip. The dependent variables were response time and the name of the shape.

The presence of the cursor should allow more rapid and accurate identification of the shapes. This is operationalized as:

Hypothesis 4.1

The mean trial time for the cursor-present condition will be less than the mean trial time for the cursor-absent condition.

Hypothesis 4.2

The mean accuracy by participant for the cursor-present condition will be less than for the cursor-absent condition.

Since there are multiple models of the effect of shape size, with opposing outcomes, there are no operationalized hypotheses for the effect of that factor. However, the interaction effect between the cursor condition and shape size can be operationalized:

Hypothesis 4.3

There will be a significant interaction effect between cursor condition and shape size for trial time.

Hypothesis 4.4

There will be a significant interaction effect between cursor condition and shape size for accuracy.

As with Experiment 1, the mental rotations data were analyzed in an exploratory, post-hoc, manner due to the lack of a theory from which to formulate a priori hypotheses.

Method

The experiment was divided into two phases. In the training phase, participants learned to identify five shapes. After they had reached a criterion level of performance, the testing phase began. The stimuli and task were slightly different in the two phases.

Participants

Participants were 12 unpaid volunteer computer science graduate students from the University of Oregon. None had participated in Experiment 1. Ages ranged from 22 to 42 with a median of 30.6. Nine were male and three were female. All had normal or corrected to normal vision. Ten reported themselves as right handed. Their scores on the Edinburgh Handedness Inventory (Oldfield, 1971) were somewhat lower than those of the participants in Experiment 1, with a median decile of R4.5. Two participants reported themselves as left handed. Their deciles on the L (left-handed) scale were L4 and L8.

Stimuli

The experimental stimuli were the five shapes located at the transition points of the Koenderink and van Doorn (1992) shape scale described in Chapter II. The names of the five shapes and their indices on the shape scale were cup (-1), groove (-0.5), saddle (0), ridge (0.5), and top (1)[1]. Three sizes of shapes were used. The "small" and "large" shapes were used in the testing phase of the experiment, while the "training" shapes were used in the training phase. Table 10 lists the shapes together with their shape index, curvedness, and principal curvatures (k_1 and k_2). These shapes can be discriminated using only the directions of the two principal curvatures. No magnitude estimation was required. The principal curvatures were sufficiently large that participants could readily distinguish their signs.

In the training phase, the shapes were displayed "head-on": with their z axis point-

---

[1] Koenderink and van Doorn (1992) name them cup, rut, saddle, ridge, and cap. I changed the names of "rut" and "cap" to "groove" and "top", respectively, to produce names that were more phonetically distinct from one another and reduce the likelihood of response slips by participants.

TABLE 10. Parameters of Stimuli in Experiment 2

Size	Shape name	Shape index	Curvedness (m^{-1})	k_1 (m^{-1})	k_2 (m^{-1})
small	cup	-1.0	50.0	50.0	50.0
	groove	-0.5	50.0	70.7	0.0
	saddle	0.0	50.0	50.0	-50.0
	ridge	0.5	50.0	0.0	-70.7
	top	1.0	50.0	-50.0	-50.0
training	cup	-1.0	100.0	100.0	100.0
	groove	-0.5	100.0	141.4	0.0
	saddle	0.0	100.0	100.0	-100.0
	ridge	0.5	100.0	0.0	-141.4
	top	1.0	100.0	-100.0	-100.0
large	cup	-1.0	187.5	187.5	187.5
	groove	-0.5	187.5	265.2	0.0
	saddle	0.0	187.5	187.5	-187.5
	ridge	0.5	187.5	0.0	-265.2
	top	1.0	187.5	-187.5	-187.5

TABLE 11. Orientations of Stimuli in Experiment 2

Identifier	z	y	x
head-on	0	0	0
A	0	30	-30
B	45	0	0
C	90	-30	30

Note. The z axis extends out of the screen, the y axis extends up, and the x axis extends to the right. The z rotation was performed first, then y, then x.

ing out of the screen at the user. In the testing phase, the shapes were displayed in three orientations. These orientations are listed in Table 11 and examples shown in Figure 19, Figure 20, and Figure 21. Note that these graphic displays are only for the purposes of this document—they were never displayed to the participants. In the testing phase none of the shapes was displayed "head-on". Every shape was displayed in every orientation for each cursor condition. Only haptic displays of the shapes were provided—no graphic display of the shapes was ever provided to participants, although a red dot was displayed at the location of the exact center of the shape. The red dot and cursor (if present) where displayed using the OpenGL viewpoint and perspective parameters listed in Table 6.

The shapes were rendered floating in space, without any haptic background. When participants moved off the surface they stopped feeling any forces. This made it very clear when they had left the stimulus, whereas when a background is present participants can have difficulty distinguishing experience of the background from experience of the stimulus (Tan, 1997, p. 201).

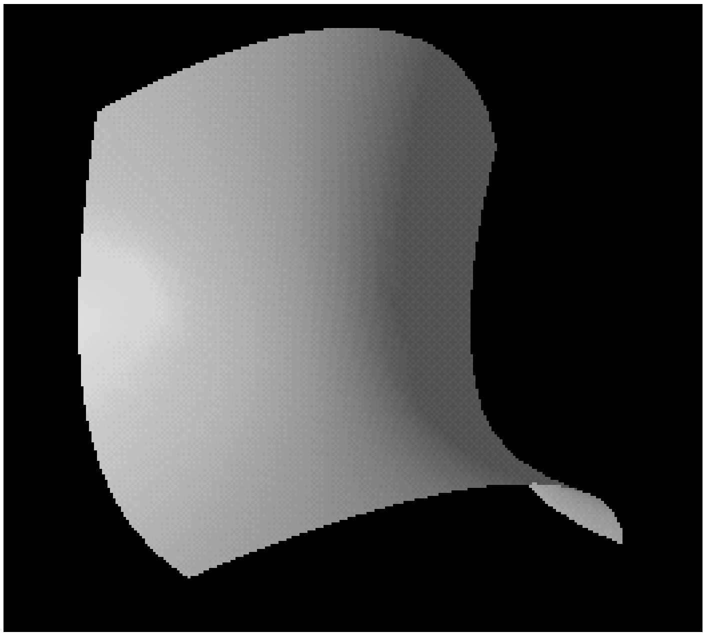


FIGURE 19. Saddle Displayed in Orientation A.

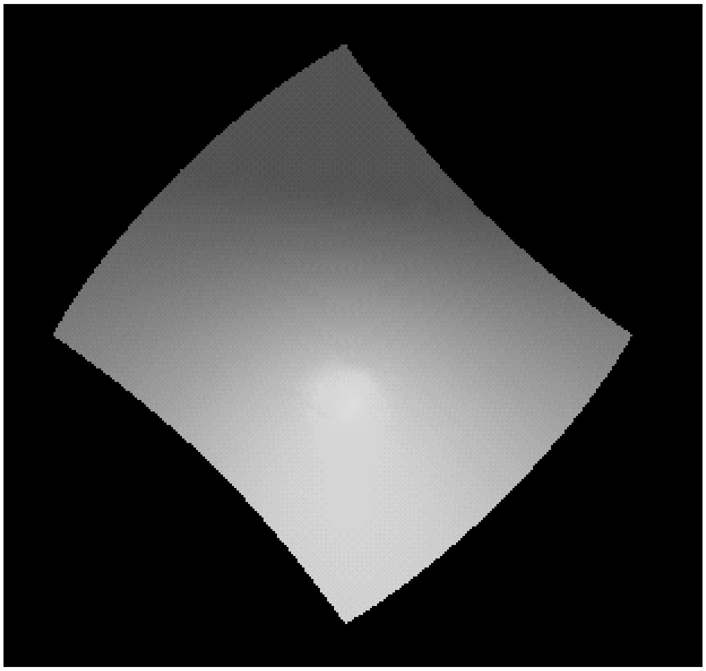


FIGURE 20. Saddle Displayed in Orientation B.

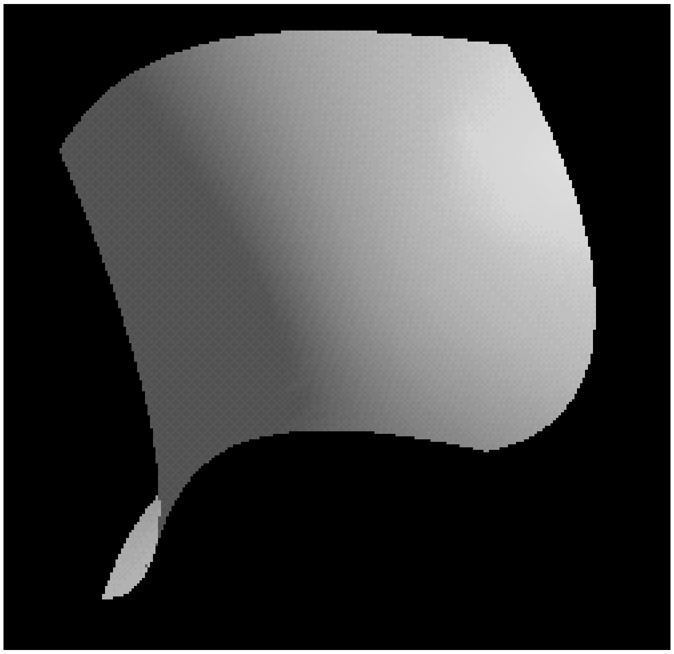


FIGURE 21. Saddle Displayed in Orientation C.

Equipment

The computer hardware and software was the same as Experiment 1. Unlike Experiment 1, the shapes were represented as a triangular mesh and rendered using the GHOST 2.1 (SensAble Technologies, 1998) `gstTriPolyMeshHaptic` class. Using this class, the haptic rendering loop consumed approximately 30% of the processor time. The graphics loop, which only displayed a red dot and (when present) the cursor, ran every 10–30 msec.

A photograph of the experimental setup is given in Figure 22. The setup was similar to that of Experiment 1, except that the PHANToM was moved back from the edge of the desk and the participant’s forearm rested on the desk supported by several softcover books. In pilot studies, I found that the shape recognition task was more

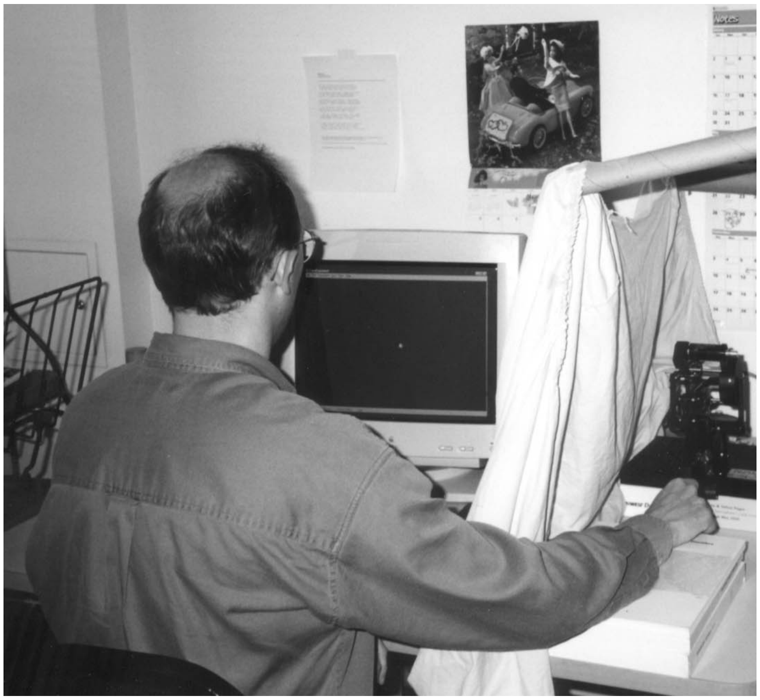


FIGURE 22. Setup for Experiment 2.

physically demanding than the curvature estimation task, and I attempted to alleviate that by providing more extensive arm support and enforcing a two to five minute break between cursor conditions. With this arrangement, no participants reported discomfort during the session.

In Experiment 1, the participant was working with full visual access. However, Experiment 2 includes a condition where the participant is deliberately denied any visual feedback as they explore the shape. To ensure that no visual cues were available, during the testing phase, a curtain was placed so that participants could not see the location of the hand holding the PHANToM. The curtain was absent during the training phase, allowing participants to see their hands.

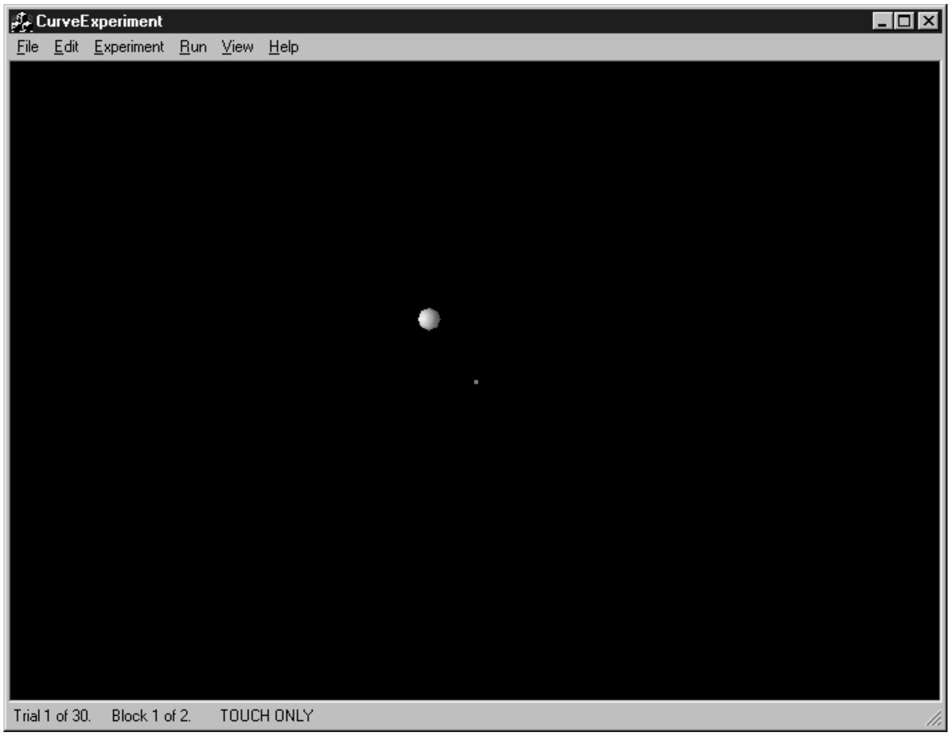
The experimental software recorded the location of the PHANToM tip for every iteration of the graphics loop. It also recorded summary data for each trial, including the total trial time and the time from first contact with the object to the end of the trial.

Task

The cursor-present trials, whatever the phase of the experiment, had the same basic structure. Initially the only object displayed on the screen was the cursor, a yellow ball. This corresponded to the location of the PHANToM tip in the graphical space. The cursor provided limited depth cues through changes in size as it moved along the z axis. The participant began a trial by holding the PHANToM in their dominant hand and pressing the Enter key with their non-dominant hand. A red dot appeared, denoting the center of the shape. The dot provided no depth information in and of itself, but would obscure view of the cursor when the cursor was behind it. The participant could determine the relative location of the PHANToM tip and the shape's center by comparing the relative locations of the cursor and the dot.

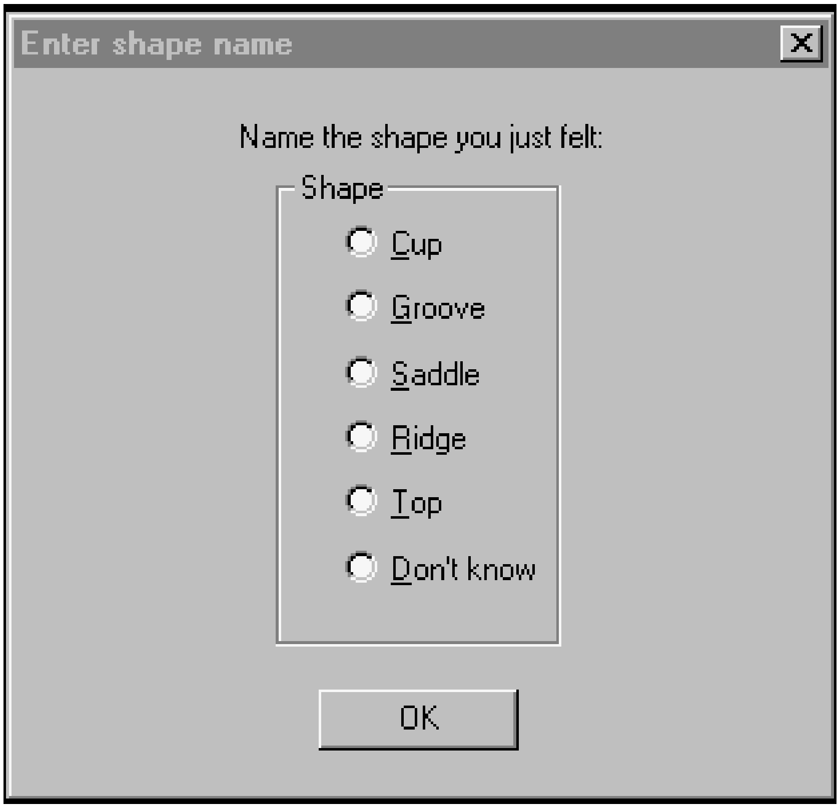
The screen layout is shown in Figure 23 . The status line at the bottom of the screen gave the trial and block numbers. The notation “TOUCH ONLY” in the status line was fixed. The only onscreen indication of the two conditions (cursor-present and cursor-absent) was given by the cursor itself. Participants had no difficulty distinguishing the two conditions after they had completed the training phase.

The participant moved the PHANToM over the surface of the shape until they believed they had identified the displayed shape, at which time they pressed the Enter key once again with their non-dominant hand. The red dot vanished and a dialog box appeared (Figure 24). The participant would select the radio button corresponding to the shape they had identified, or could press “Don't know”. Participants were told the



Trial 1 of 30. Block 1 of 2. TOUCH ONLY

FIGURE 23. Layout of Screen.



Enter shape name

Name the shape you just felt:

Shape

○ Cup

○ Groove

○ Saddle

○ Ridge

○ Top

○ Don't know

FIGURE 24. Dialog Box for Entry of Shape.

radio buttons could be selected either with the mouse or by entering the first letter of the shape name. They were also told that the timer stopped when they pressed the Enter key the second time, so there was no requirement that they enter the shape name quickly. Closing the dialog box ended the trial.

Trials in the cursor absent condition did in fact display the cursor during portions of the trial. In pilot studies, participants found it extremely difficult to even find the shape, let alone identify it, when the screen cursor was never displayed. Therefore I configured the experimental application to display the cursor whenever it was outside the minimal bounding box of the shape. During a cursor-absent trial, the presence of the cursor indicated to participants that they were far away from the shape. By comparing the relative locations of the center dot and the cursor, participants could bring the PHANToM close enough to the stimulus to restore contact, but once they were in contact they no longer had graphical representation of their location on the surface. With this modification, none of the participants had any apparent trouble making initial contact with the shape or regaining contact when they had slipped off.

Procedure

In the training phase participants learned the five shapes and the recognition protocol. Once they had passed a criterion test of performance, the testing phase began.

Training Phase: Learning the Five Shapes

The participants were first told the names of the five shapes. They were then asked to feel “cup”, the first shape. They were asked to move the cursor so that it obscured (in other words, it was in front of) the red dot and then push the PHANToM stylus away from themselves. This brought them into contact with the shape. When a participant went behind the shape (and everyone did at some point), the experimenter pointed out that the red dot was in front of the cursor and that this indicated they were behind the shape. Note that the shapes were one-sided: The participant would feel resistance pressing into the shape from the front but would encounter no resistance passing through the shape from back to front.

While learning the five shapes, the cursor location was displayed, the participant could see their hand, and the name of the current shape was displayed in the status line at the bottom of the screen. Once the participant indicated that they had sufficient experience with the cup shape, they pressed the Enter key and selected “cup” in the shape name dialog box. Once the dialog box was gone, they pressed Enter again to initiate the next shape, groove. The process repeated through the other three shapes, whereupon the participants cycled through all five shapes one more time. All shapes were displayed in the training size and in the head-on orientation.

During this process, the experimenter told them several important points about the shapes. First, he noted that all the shapes were smooth curves without any localized regions that were radically different from the main part. Second, he pointed out that the task the participants were going to perform in the actual experiment was to distinguish the five shapes. He encouraged the participants to learn the features of the five shapes that distinguished them from each other.

Training Phase: Learning the Recognition Task

After participants had experienced every shape twice, they began to learn the recognition task. Every participant was eager to move on; all were quite comfortable with the five shapes after experiencing each one twice. They were also quite comfortable with the structure of a trial. Now they began to learn the actual task they were going to perform in the experiment. In these trials, no indication was given of the current shape name.

In the recognition task, the five shapes were presented in random order. Participants felt a shape, then chose the appropriate name in the dialog box. If they correctly named the shape, they immediately went on to the next trial. If they gave an incorrect name, a second dialog box informed them that they had chosen incorrectly and gave the correct name of the shape. Once they had completed a practice block of all five shapes, a dialog box informed them of the number of correct choices they had made during that block and they went on to the next practice block.

The cursor conditions in the recognition practice task were presented in the same order that the participants would experience in the testing phase: Those whose first testing block was cursor present began the recognition practice task with cursor present trials, and vice versa. The first time participants began a cursor-absent block, they were told the cursor was only going to be displayed when it was far from the object. Once they had done the first cursor-absent trial, all participants appeared quite comfortable with the procedure.

Participants continued doing recognition practice blocks until they had fulfilled one of two criteria: Either two consecutive perfect blocks or three consecutive blocks with only a single error in total. Nine of the twelve participants performed ten correct practice recognitions in a row—a strong indication that they had learned the shapes and understood the task. The other three participants had slightly more difficulty learning the task, but generally picked it up quickly.

The training phase took about 30 minutes for both learning the shapes and the recognition task.

Testing Phase: Performing the Experimental Task

Once participants had met the criterion for performance, they began the testing phase. These trials differed in several important ways from the training trials. In these trials, no feedback was given of the correctness of their answers. Furthermore, shapes were now displayed in orientations A, B, and C, rather than head-on. Shapes were displayed in the small and large sizes rather than the training size.

Most importantly, during the actual experimental trials a curtain was set up between the participant and their dominant arm so that the participant could not see their hand. This ensured that their performance during the cursor-absent trials was purely based upon kinesthetic experience and that their performance during the cursor-present trials was purely based upon the combination of kinesthetic experience and the visual cursor.

Shape type, size, and orientation were fully crossed within a block for a total of 30 trials per block. Participants performed two blocks, one in each cursor condition. The order of starting conditions for the testing phase was counterbalanced, with half the participants performing blocks in cursor-present, cursor-absent sequence and the other half in cursor-absent, cursor-present sequence.

Participants were told they could take a break at any time between trials, and were reminded that the red dot was displayed during a trial. They were required to take at least a two to five minute break between blocks, to allow their arms time to rest. The experimenter remained in the room while they performed the complete experiment, observing performance, answering questions, and monitoring the system for any problems.

The experimental trials took between 45–60 minutes to complete. After the experiment participants answered a series of six open-ended questions. If they expressed interest, at this time they were told the experimental hypotheses of the study. Total time for a session was about 90 minutes.

Results

Five trials were dropped from the data set because the participants pressed the “Enter” key immediately after starting the trial and never had any contact with the shape. One trial of p3 was dropped because the participant never actually made contact with the shape, despite spending 44 seconds. All other trials were retained for the analysis.

Recognition Times and Accuracy

Participants took quite a while to recognize the shapes. Since the task of interest is recognizing the shape, not finding it in the first place, I use the time from first contact with the shape until the end of trial, the *contact time,* as my measure of response time (rather than the total time, including time to find the shape). Seven-number summaries[2] of the distributions of time until first contact and contact time are provided in Table 12 and Table 13, respectively. Participants generally had very little trouble finding the shape: Over 87% of all trials had time to contact of 2.14 seconds or less, although one trial took 12.5 seconds before the shape was contacted. Once contact was made, participants spent a considerable time exploring the shape, with a geometric mean, median, and arithmetic mean, time of 22.5, 23.8, and 28.6 seconds, respectively, with 50% of the values between 13.7 and 37.7 seconds.

The midsummary column of the response times, together with the large difference between the arithmetic mean and the median and geometric mean response times, clearly indicates that the distribution of times is skewed. Quantile-quantile plots showed the distribution of time from contact to be log-normal; a logarithmic transformation produced a distribution extremely close to normal, and the resulting transformed scores were used in the inference procedures described below. Consequently, when the means and confidence interval bounds of the effect sizes computed in these procedures are inversely transformed to the original scale, the effects are ratios of the geometric means rather than differences of the arithmetic means. I report all such ratios as percent changes, and give both the estimated value and the bounds of its 95% confidence interval, abbreviated CI. Thus, for example, I report the effect of block as -14%, CI=[-23%, -3%]. This means

---

[2] See Appendix A for a description of seven-number summaries.

TABLE 12. Time to First Contact (s) in Experiment 2

n = 714, mean = 1.251		
	1.011	
0.651	(1.122)	1.593
0.431	(1.286)	2.142
0.000	(6.274)	12.548

TABLE 13. Contact Time (s) in Experiment 2

n = 714, mean = 28.6		
	23.759	
13.732	(25.695)	37.657
9.612	(30.388)	51.164
2.944	(95.858)	188.772

the geometric mean of the time for the second block was 14% less than the geometric mean of the first block, with a 95% confidence interval from -23% to -3%.

There is a surprisingly large number of trials that took a very long time: 12.5% of all contact times were between 51.2 and 188.8 seconds. The longest trials were distributed fairly evenly across participants: Six participants had one or two trials longer than 90 seconds.

Breaking down the distribution by participant and block shows large individual differences. Figure 25 is a box plot of the contact times grouped by block by participant for participants whose first block was cursor-present. This group is quite consistent. Every participant had lower median times in the second, cursor-absent, block than in the cursor present block, although for p13 and p7 the reductions were small. Three of the participants, p2, p3, and p9, had substantial decreases in the variance of their trials in the second block, although p10 shows a modest increase. Overall, this group did better in the second block, although it is impossible to say whether this was because they found the cursor- absent condition easier or they had improved with practice.

The cursor-absent-first group was less consistent (Figure 26). Four of these participants had smaller median times in their cursor-present (second) blocks, although only p14 and p4 had substantial decreases. Changes in variance were mixed for these four participants, with one having greater variance and three having less in the second block. For these four, as with the six who began with the cursor present condition, it is impossible to separate improvements due to the change in interaction technique from improvements due to learning. The remaining two participants had worse performance in their second block. P1 had a substantial increase in median time and variance, while P5 also had a slight increase in time while variance remained constant. In the free-form questions after the experiment, p1 reported that the cursor was distracting and p5 reported that the cursor was “not very helpful”. For these two, the evidence is much stronger that the second interaction technique, the cursor-present condition, was harder to use than the cursor-absent condition.

The mean score for participant accuracy was 84.5% (s.d. 12%). Figure 27 is a stem plot[3] of the mean accuracies for all participants. Performance was generally good, although three participants only performed about two-thirds of their trials correctly.

A two-way within-subjects ANOVA was computed for natural log of time. Hypothesis 4.1 was not supported: the effect of cursor condition was both small (CI=[-14%, 16%]) and unreliable in direction ($F_{1,11} = .098, p = .760$). Note that while the data does not permit inference of a direction of the cursor effect, it does permit inference that the effect size was small. The effect of shape size was significant ($F_{1,11} = 6.986, p = .023$)

---

[3] See Appendix A for a description of stem plots.

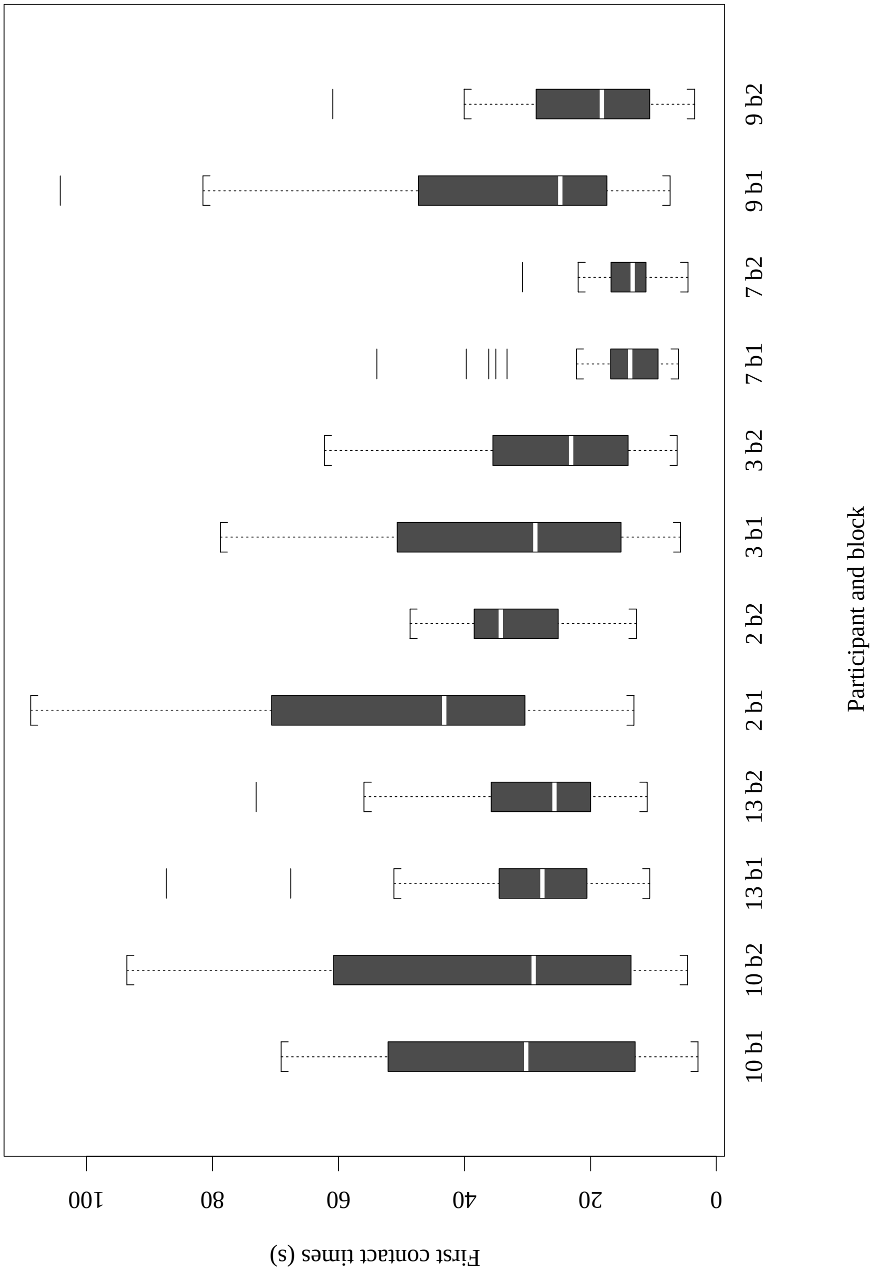


FIGURE 25. Contact Times (s) for Participants Whose First Block Was Cursor-Present. One trial of 189 seconds was not plotted.

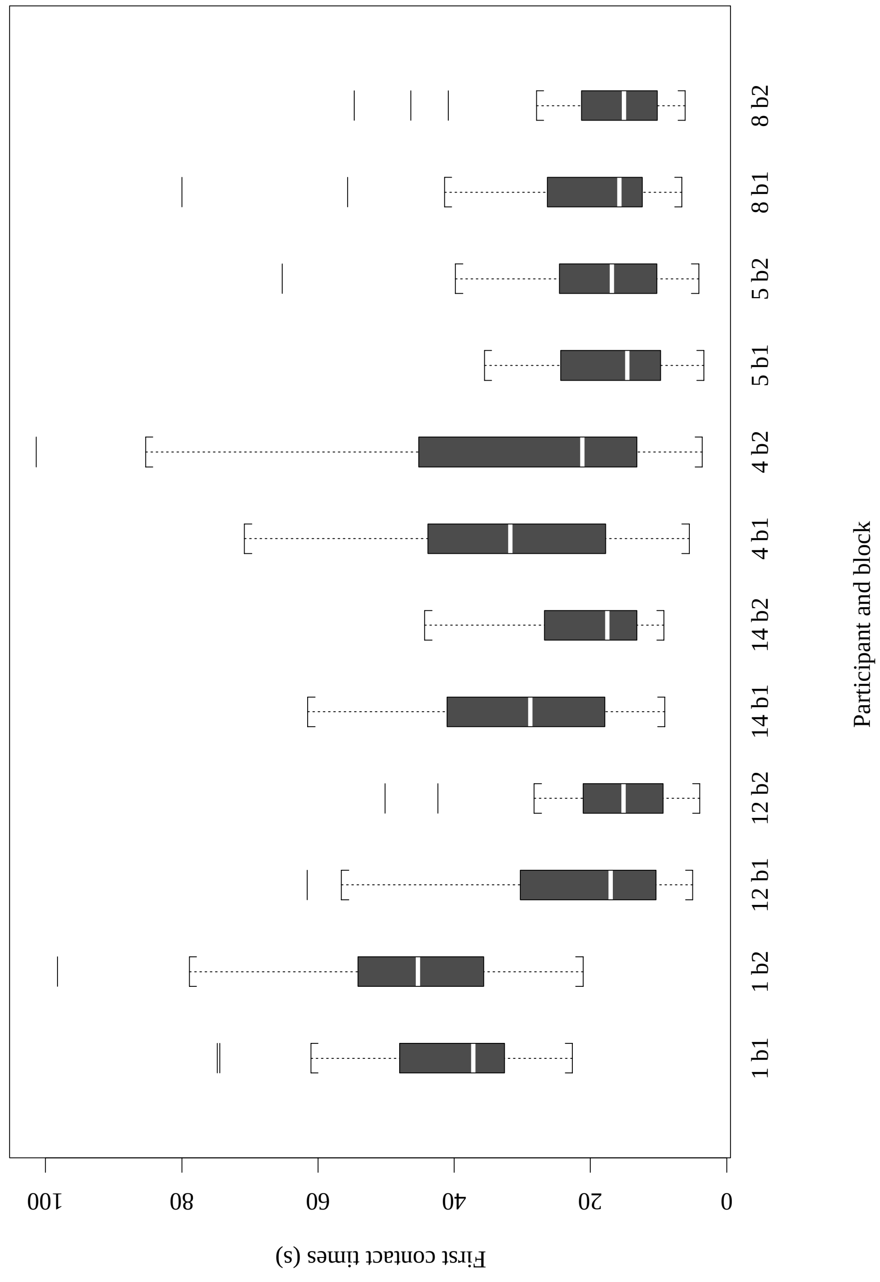


FIGURE 26. Contact Times (s) for Participants Whose First Block was Cursor-Absent. Two trials of 123 and 130 seconds were not plotted.

Depths	(unit=10%)	
2	6	77
4	7	08
6	8	13
6	9	015588

FIGURE 27. Mean Accuracy for Participants ($n = 12$) in Experiment 2.

Depths	(unit=10%)	
2	0	23
(5)	0	67789
4	1	1244

High: 63%

FIGURE 28. Mental Rotation Time Percentiles for Participants ($n = 12$) in Experiment 2.

but small (12%, CI=[2%, 22%]). The cursor had no reliably different effect on large objects than small objects (Hypothesis 4.3): the cursor condition x shape size interaction effect was nonsignificant ($F_{1,11} = 2.790, p = .123$)

The two-way within-subjects ANOVA of accuracy produced no significant effects. The provision of the cursor did not reliably improve the accuracy (Hypothesis 4.2) overall nor for small shapes in particular (Hypothesis 4.4).

Unlike the participants in Experiment 1, the mental rotations percentiles for both time and accuracy for the participants in this experiment were tightly clustered. Figure 28 and Figure 29 give stem plots for time and accuracy percentiles, respectively. Due to this close clustering, linear regressions of the percentiles produced no useful models.

Depths	(unit=10%)	
2	7	22
5	8	444
(5)	9	22222
2	10	00

FIGURE 29. Mental Rotation Accuracy Percentiles for Participants ($n = 12$) in Experiment 2.

Learning

Participants showed evidence of learning throughout the course of the 60 testing trials. The second block was was reliably faster (paired $t(11) = -2.760, p = .002$) although the effect was moderate (-14%,CI=[-23%, -3%]). There is no evidence of asymmetric transfer of skill: The effect of order of cursor mode presentation was small (10%) and unreliable ($F_{1,10} = .211, p = .656$). Figure 30 shows all trial times less than 60 seconds[4], together with loess and loglinear regression fits of the points. The regression produced the line $time = 3.45 \ln(\text{sec}) - .107 \ln(\text{sec}) / \ln(\text{trial})$ with $r^2 = .018$. For such nonhomogenous data, the r^2 value is not a good diagnostic of the appropriateness of a linear model. Comparing the linear fit with a robust loess nonparametric fit is a better test. The loess fit has very similar shape and slope, supporting the conclusion that the trend was loglinear, although the loess is about seven seconds higher because it was taken over the raw times rather than their logarithms.

The slope of the curve indicates the rate of learning. Over the course of the 60 trials in an experimental session, execution time on the regression line falls from 32 to 20 seconds. Overall, participants appeared to have been nearing a practiced level of

---

[4] Eight per cent (58) of the trials were longer than 60 seconds. They were evenly distributed throughout the session. Note that the two curves were fitted to all the points, not just those less than 60 seconds.

performance, as this aggregate learning curve only drops from 22 to 20 seconds between trials 30 and 60, and extrapolating to trial 120 gives a time of 19 seconds. Unfortunately, this implies that learning clearly occurred throughout the first block, and probably over the second as well, so comparisons of individual performance between the two modes are confounded with learning. The counterbalancing (and lack of order effects) allows the overall performance of all 12 participants to be compared, however.

Effects of Extended Practice

To test the effects of extended practice I invited two participants, p8 and p9, to each return for two further experimental sessions. The structure of each session was the same as the original, except the participants did not go through the training phase. At the start of the extra sessions, they performed a few practice trials to refamiliarize themselves with the procedure, then did two blocks of testing trials. The counterbalancing order of the original session was maintained. P8 performed blocks in the order cursor-absent, cursor-present, while p9 performed blocks in the order cursor-present, cursor-absent. The extra sessions lasted 30–45 minutes each. The blocks from the second session were numbered three and four, while those of the third session were numbered five and six.

Two trials from p8 were removed from the raw data set. Trial 14 of block 3 was deleted because the experimenter observed the participant had confused the back wall of the haptic bounding box for the stimulus. Trial 4 of block 1 was deleted because it was not representative of any other trial by that participant (50% higher than the next highest time and 10 times the interquartile range from the 75th percentile) and regression diagnostics indicated it was an overwhelmingly influential point.

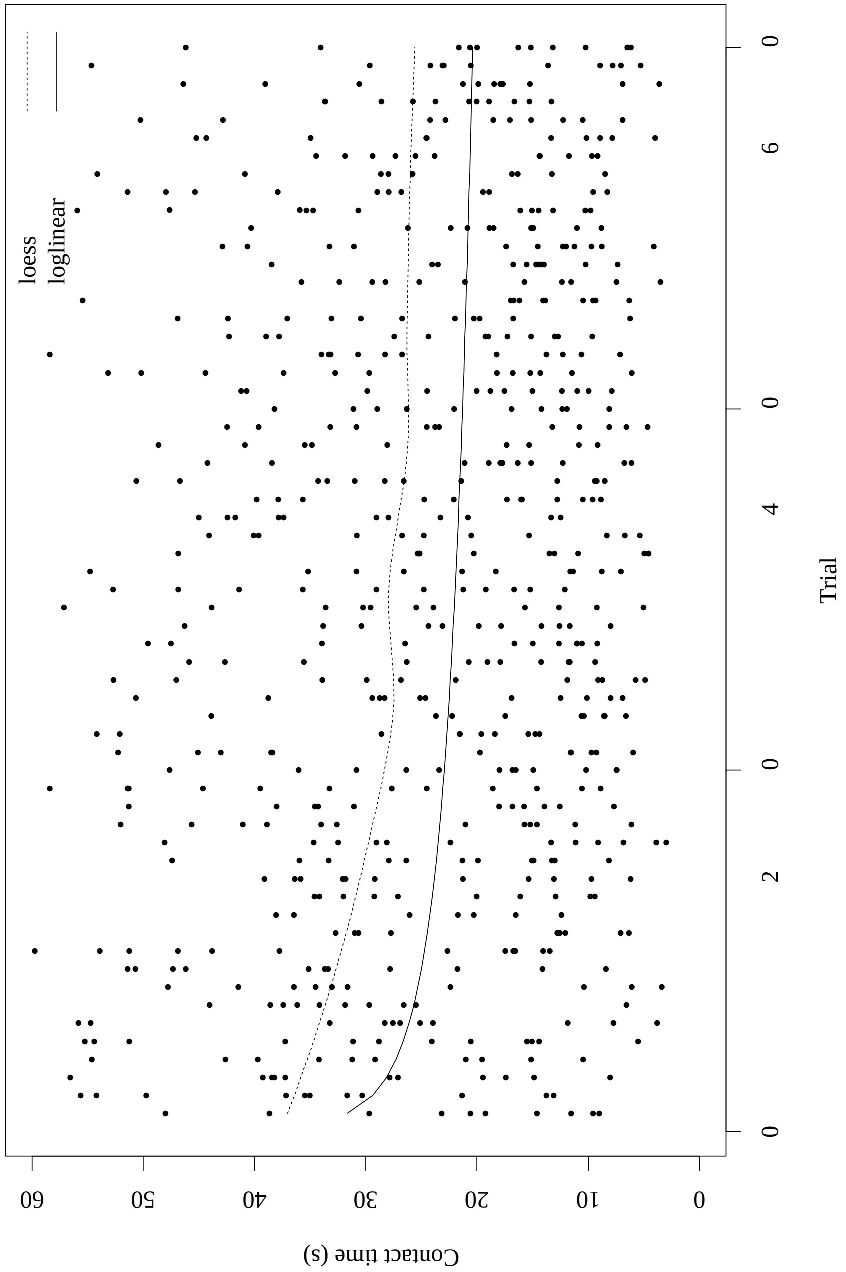


FIGURE 30. Trial Times for All Participants in Experiment 2. Fifty-eight trials longer than 60 seconds are not displayed, although they were used in the regressions.

Figure 31 is a plot of all trials less than 60 seconds[5] for the 358 trials performed by these participants. The loglinear regression lines for their combined[6] points was $time = 3.71 \ln(\text{sec}) - .242 \ln(\text{sec}) / \ln(\text{trial})$, with $r^2 = 0.14$. Transformed back to seconds and trials, the resulting curve goes from 40.9 seconds in trial 1, to 15.2 seconds for trial 60, ending with 11.6 seconds for trial 180. The accuracy of these two participants was consistently good. Over all 180 trials they achieved a combined accuracy of 96% for both cursor conditions and for the last block they achieved 97% and 100% for cursor-absent and cursor-present, respectively. Compared to the other ten participants, these two were more accurate, faster, learned more with each trial, and continued learning well into the third session. The loess plot shows a similar curve, although it suggests that learning may have abated in the second session and resumed during the third.

Figure 32 shows boxplots of the trial times for these two participants for all six blocks. The most striking feature is the difference between the cursor-absent block (5) and cursor-present block (6) in the final session of p8. The effect was substantial (46%, CI=[22%, 75%]) and significant (paired $t(29) = 4.28, p = .0002$). However, the effect for the two conditions in the final session of p9 could not be reliably determined.

---

[5] Only 6 trials (3%) were longer than 60 seconds. Four occurred in the first block, one in the second, and one in the third.

[6] Separate loglinear regressions on the two participants produced similar results, with p8 having a lower intercept and slope than p9. By trial 180 the two lines were identical. In the interest of clarity, I display and discuss only the regression of their combined points.

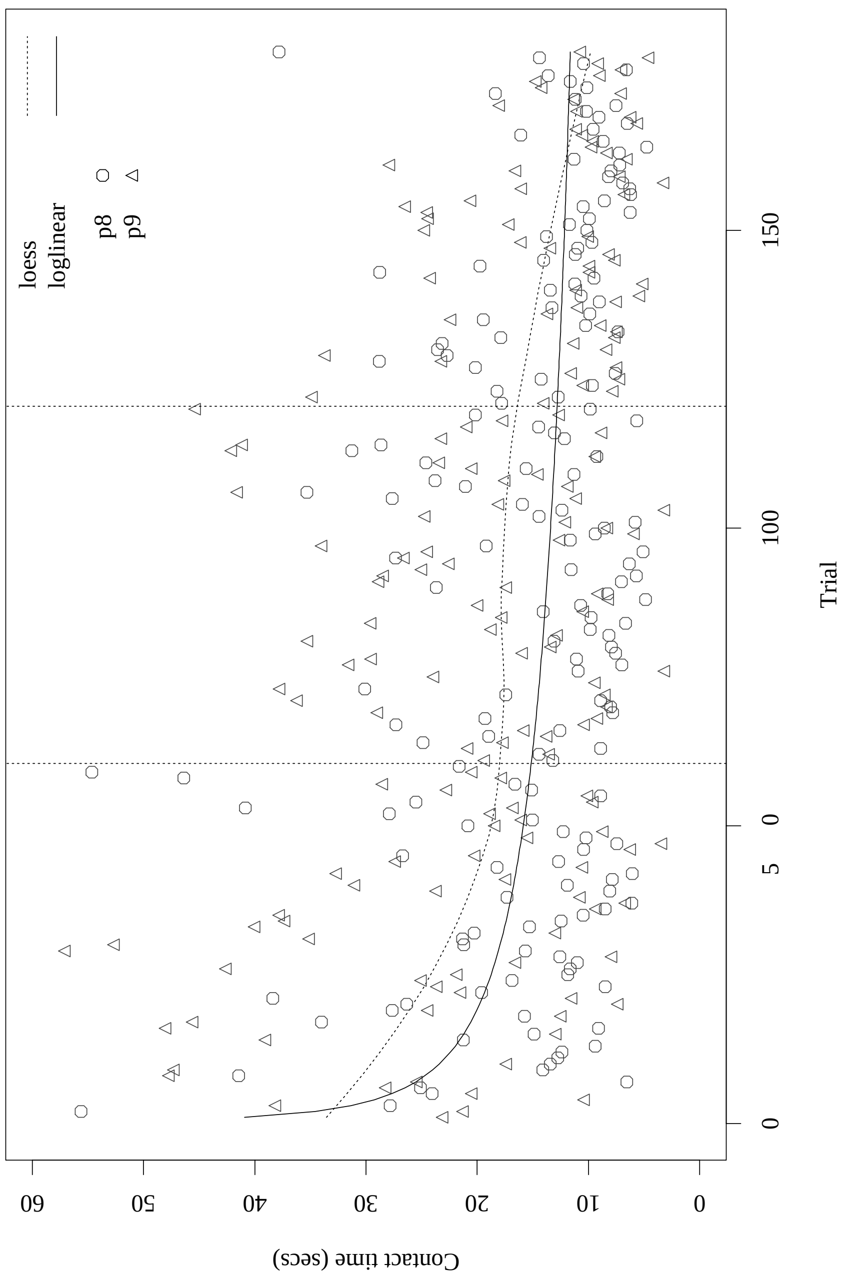


FIGURE 31. Trial Times for P8 and P9, Blocks 1–6. Six trials longer than 60 seconds are omitted.

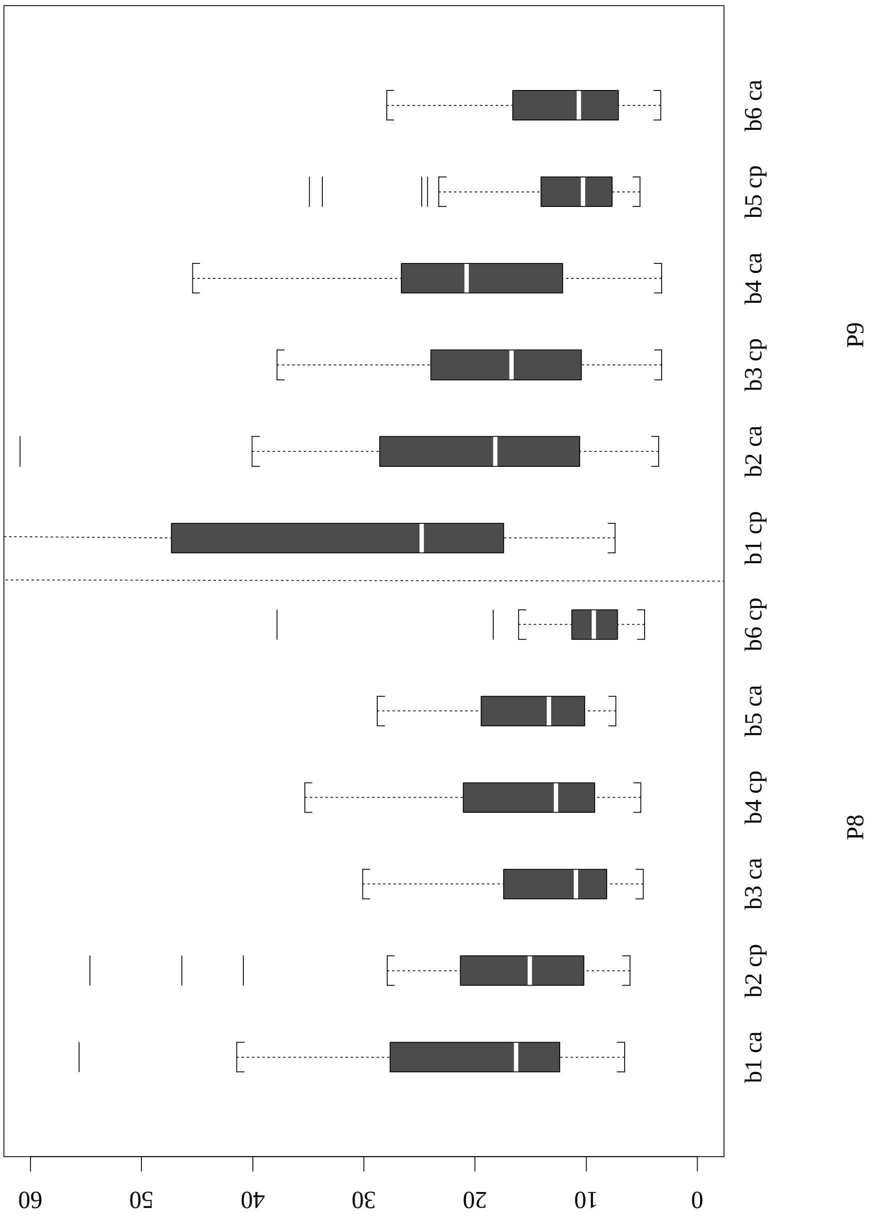


FIGURE 32. Trial Times (s) for P8 and P9, Blocks 1–6. Six trials longer than 60 seconds are omitted. ca=“cursor absent”, cp=“cursor present”.

TABLE 14. Response Times for Recognition of Shapes of Various Complexities

	Vision		
	None		
Haptics	Shape rec.	Object rec.	Full
None	—	—	
PFD	23 (L)	—	
1 Finger	47 (H)	24 (H)	
5 Unflexed	26 (H)	18 (H)	
Full hand	17 (H)	10 (H)	
Two hands	2 (L)	5–6 (H)	1 (L), 1 (L)

Discussion

Temporal Integration

The most striking result of this study is the difficulty of the task. Despite the simplicity of the stimuli and task, participants still had a mean time of 22.5 seconds with a 15% error rate. How does performance in this environment compare with human performance at recognizing physical objects? While no exact counterpart exists using physical instances of our shapes, several previous studies have used similar tasks (Table 14). Lederman, Klatzky, and Reed (1993) devised stimuli simpler than ours, three ellipsoids of revolution that differed only in their height to width ratio, not their material. Using both hands, observers could distinguish these objects in 1.0 seconds. In another study using common household objects, whose shapes are more complex than those we used, observers were able to haptically recognize the objects in under 5 seconds with a 4% error rate (Klatzky et al., 1985). With a slightly different set of objects and under the restriction that they could not lift objects, the mean time was 6.2 seconds with a 5% error rate (Klatzky et al., 1993).

These times are far faster than the performance in the haptic environment of this experiment, but comparisons must be made cautiously. The two latter studies used object recognition tasks, where participants had access to material properties, so identification was based upon more cues than simply shape. When Klatzky et al. (1993) changed the task to shape recognition by requiring their participants to wear gloves, the mean response time rose to about 16 seconds. When they further restricted their participants to using a single gloved index finger, mean response time leapt 2.7 times to 42 seconds with an error rate of 25%. Restricting the haptic flow to a single point, requiring the observer to induce object shape over time, dramatically limits performance in physical environments.

The same limitation, both in kind and degree, appears to apply in point force environments. The response times of this experiment fall comfortably in the ranges of the other shape recognition tasks. The experimental stimuli in Experiment 2 fall at the midpoint of complexity, requiring more complex discriminations than the width to height ratio required by Lederman et al. (1993) and less complex discriminations than the household objects used by Klatzky et al. (1993), and the times in Experiment 2 fall between the times of those studies. Temporal integration is at least as hard to do in point force environments as in physical environments, and the lack of spatially-distributed cues from contact of the fingertip with a physical object may well make the haptic environment worse.

These comparisons also provide a useful validation of the evaluation task itself. The response times in this task are well within the range that would be predicted from data on a task with physical objects of comparable complexity. The shape recognition protocol of this experiment appears to measure the determining factors in performance of shape recognition.

Mechanisms of Temporal Integration

On pp. 116–121 I proposed a partial model of temporal integration, summarized in Table 9. Table 15 is the same table emphasizing (with double lines) the entries that were supported by the results. As Experiment 2 is an exploratory study, this table is intended to suggest future research directions, not definitively establish a model.

Visual proprioception produced a rather small effect and no significant interaction with shape size. While there may well have been factors confounding this result (further discussed below), to the extent that this generalizes to practiced use of the PHANToM it suggests that the visual memory of recent cursor positions is as short term as the kinesthetic memory. This is similar to a result of Loomis, Klatzky, and Lederman (1991), who found that the visual system performs shape recognition poorly when restricted to a limited field of view. If this result is generally true, it implies that humans have as much difficulty temporally integrating visual data as they do with haptic data. This has unfortunate implications for point force environments since it indicates that whatever greater spatial acuity is possessed by the visual system will be of little use in mitigating the effects of haptic temporal integration. This is a potentially crucial implication and should clearly be studied in more detail.

The predicted effects of small shapes were found. However, at the present level of data analysis I cannot separate the effects of distance from the effects of velocity. Either smaller distance or lower velocity could have produced the shape size effect. The trace data recorded from this experiment is sufficiently detailed to compute the velocity, but for now I cannot say to what extent the users moved more slowly on small objects. The shape size effect indicates that such a trace analysis is worth performing and will provide valuable data on the role of distance and velocity in haptic temporal integration.

TABLE 15. Temporal Integration Model and the Results of Experiment 2

Kinesthetic mechanism	Factor				
	Vision present	Vision-small shape interaction	Shape size		
			Small distance	Velocity	
				same (?)	slower (?)
Perception	↑	↑	⇓	—	⇓ (?)
Memory encoding	↑	↑	⇓	—	—
Memory span	↑ or =	↑	—	—	⇓ (?)

Note. Double-lined entries in the model were supported by the data.

Visual Proprioception

The effect of the cursor condition was small, between -14% and 16%. As noted above, this may be an accurate reflection of fundamental limits on human temporal integration of spatial locations. However, it may also have been due to a possible confounding factor, the unfamiliarity of the participants with the PHANToM and the interaction technique. If participants spent most of the session learning such prerequisite skills as moving the PHANToM through space and learning to interpret the visual cursor position, with its incomplete representation of the depth dimension, then they would not have been able to take advantage of the cursor when it was present. Participants do not appear to have reached skilled performance during the course of the experiment. Some participants reported that they found the visual cursor condition distracting. Many had their hands full merely attending to the haptic sensations. I speculate that the sensory overload may have reduced with practice. The visual cursor might have proved significant when participants had achieved practiced performance.

The results of the two participants who returned for extended sessions are mixed: In the last session, p8 performed significantly faster with the cursor than without, while p9 had essentially equivalent performance. However, this probably still does not represent practiced performance, as they appeared to continue learning right through the sixth block. It is not clear how many more trials would have been required before their learning became negligible. This result differs from the aggregate practice curve for all twelve participants, which appeared to be approaching practiced performance during the second block. The current data cannot resolve this question. Longitudinal studies are required to determine changes in performance with practice and how many trials are required to reach a practiced level of performance. This is also a prerequisite to determining the influence of cursor condition. Only when we have reliable experimental protocols for testing practiced performance will we be able to definitively determine the contribution of visual proprioception to point force shape recognition.

Individual Differences

There were large individual differences in performance. Figure 25 and Figure 26 show great diversity of both median times and variances between individuals. Figure 27 shows a 1.46 to 1 range in accuracy for the participants. The learning data on p8 and p9 indicates that they learned more quickly than the overall group.

There is also evidence that a potentially substantial proportion of the population find this task extremely difficult. Fourteen participants were originally recruited for this study, but despite extensive practice two could not perform the training test sufficiently accurately and did not proceed to the testing phase of the experiment. These two participants understood the task completely—they could recognize some shapes. In informal followups, both could reliably recognize plaster models of the shapes when they used their five fingertips. However, when performing the task with the PHANToM, they were both extremely slow and unreliable. That one seventh of participants, recruited from a technically sophisticated group, should have such difficulty with a simple task using a point force device suggests that a large subgroup of the general population will find point force environments extremely difficult to use. I speculate that the proportion of such individuals may be even higher in non-technical populations.

The mental rotations test was not a useful predictor of individual performance in this experiment, despite the large individual differences in various measures of performance. The individual factors accounting for this wide range of performance do not appear to be measured by the test.

Conclusions

Shape recognition is an important task in point force environments, used in a wide variety of potential applications. Yet in this experiment, twelve participants required approximately 23 seconds to recognize a class of simple shapes. For this task, a point force environment proved anything but natural. The temporal integration imposed by a point force display appears to have significantly limited performance on this task, just as it has on similar tasks with physical objects. Furthermore, it is unlikely the participants could have attended to any other tasks while performing this one. In fact, they reported difficulty even attending to the visual cursor indicating their current position. This would seem to preclude the use of haptics to display shape information when the visual channel is overloaded.

If this result generalizes, it suggests that crucial improvements must be made to the usability of point force environments before they can be applied to a wide range of tasks. Given the mechanical difficulty of improving the hardware displays, the main areas of potential improvement are the interaction techniques and user training. The interaction technique explored in this experiment, a visual cursor display providing visual proprioception, did not reliably improve performance. Further study is required to see if this result is due to the inexperience of the participants or some inherent aspect of human temporal integration. Whatever the outcome of this interaction technique, development of further techniques that substantially improve performance over the level of participants in this study is essential to the success of point force environments.

Training is a complementary approach to improving performance. It matters little that we can improve performance to, say 5 seconds, if this this can only be achieved after users engage in days of practice. Most practical applications of point force environments will require that users can achieve adequate performance with only little practice. It may be possible to increase the rate of learning with specialized practice environments that assist the user in learning to use the novel perceptual resources presented by these environments.

All of this requires further empirical work. Perhaps the most important contribution of this chapter is the demonstration of an evaluation protocol for shape tasks in haptic environments. Given the existence of this protocol, the next step is to explore these initial results in greater detail. This experiment clearly should be replicated with a longitudinal design, investigating the degree to which these limitations persist as users become skilled in the use of point force haptics. While visual proprioception did not enhance performance in this experiment, perhaps greater practice will permit users to benefit. Finally, a more diverse sample of participants should be recruited, to determine the range of individual differences in this task. We cannot begin to apply point force environments to the solution of real-world shape recognition tasks until they are capable of supporting far higher rates of performance than were found in this experiment.

CHAPTER V

SOFTWARE ARCHITECTURES FOR HAPTIC ENVIRONMENTS

The previous three chapters have presented a description of human interaction with haptic environments and two experiments on human performance in these environments. These chapters have emphasized the integrated nature of human perception in these environments and advocated a view of these environments as perceptual resources for accomplishing the intended task.

This chapter surveys the current state of the art for haptic environment user interface software architectures (abbreviated “architecture” for the rest of this chapter) and how they might evolve in the near future. It concludes with speculations on how the perceptual theories of Chapter II might influence the future development of these architectures.

There are several fundamental problems confronting the application programmer for a haptic environment. First, as I have argued throughout this dissertation, usable interaction techniques for haptic environments must be designed with careful attention to perceptual issues. Application programmers typically have neither the time nor the training to do this well. Second, haptic environments place extreme demands upon the available processing power. Third, haptic environments have several crucial loops that each must be executed sufficiently rapidly to maintain effective response time. Organizing the software so that all these loops are serviced at the correct rate is a challenge, especially under conditions of limited processing power, conditions that are likely to prevail for the forseeable future. The haptics rendering loop must run at least a half order of magnitude higher than current graphics loops require and the complexities of the virtual environments supported by current hardware are much lower than the ones we would like to create. Finally, as with any complex software, the system code must be organized in a manner that places related functionality together. This chapter considers possible solutions to the above problems, and most importantly methods to package the solutions so that application programmers can readily use them. None of the currently available architectures provides a satisfactory solution to all these problems.

Myers (1995, pp. 81–82) characterizes user interface tools in terms of whether they assist in the design phase, the runtime-phase, or the evaluation phase. This chapter is concerned exclusively with the runtime phase, the architecture of the running haptic application. The software architecture packages two kinds of solutions for the application programmer. It includes libraries of routines that can be invoked by the application and the overall structure imposed on the application by those libraries. A good architecture benefits the programmer in several related ways (Myers, 1995, pp. 66–68; Olsen, 1992, pp. 8–11):

1. Common bookkeeping functions, which typically require mundane but tedious programming, are already written for the application programmer.

2. Complex algorithms can be written and optimized by experts and then used by the application programmer. These algorithms may be complex due to their mathematical nature or due to the complex performance requirements imposed by human perception.

3. A predefined architecture provides a framework grouping related code. The code could be related by a common data structure, a common execution thread, or a common performance stratum (high rate vs low rate).

4. An architecture can specify interfaces corresponding to the division of labor between members of the development team, as for example an interface between the graphical representation of a world and its procedural implementation.

5. Development is faster because the programmer is modifying a running system rather than writing a system from scratch, with the attendant wait to have even an initial working prototype.

Current Architectures for WIMP Applications

The initial starting point for an architecture for haptic environments is the common architecture for 2-D WIMP interfaces. While these can be (and have been) implemented in a myriad of ways, the toolkit and application framework architecture is one of the most common and so I select that for my example in this chapter. In this architecture, a window is divided into two regions, the control region and the content region. The control region consists of standard widgets such as scrollbars, the window title bar, and the menu[1]. A widget is simply a subroutine that implements a standard interaction technique for the system. When users execute an interaction technique with a widget, the widget manages all interaction and only passes on a single event to the application at the conclusion of the interaction. Toolkits are libraries of predefined widgets, and hence predefined interaction techniques.

Toolkits not only improve productivity by providing pre-written code, they improve the quality of applications by providing a library of pre-written interaction techniques whose perceptual properties have been carefully designed. Programmers using

---

[1] In the MacOS implementation, the menu bar is detached from the window and placed at the top of the screen, but logically it is linked to the window.

a toolkit are not only reusing code, they are reusing designs that successfully support human perceptual needs. The need for the specialized skills of designing perceptually adequate WIMP interfaces is mitigated through encapsulation of pre-coded interaction techniques into widget libraries. Repeated exposure to the same widget set also gives users familiarity with the widgets, allowing them to develop accommodation skills for any deficiencies in the widgets.

The collection of interaction techniques provided by widget libraries is incomplete: Virtually all interaction techniques in the content region of the window must be programmed from scratch by the programmer using low-level calls to the input and display devices. Where toolkits provide an implementation of the control region of the window, application frameworks provide an implementation of the content region. The framework implements the basic event loop and provides limited functionality for the core features of an application, such as opening and saving files and an “About” dialog. The programmer overrides specific behaviors of the framework by creating subclasses that override methods called at specific points in the event loop, such as when a dialog box is closed. More global application-specific behavior can be coded by modifying the event loop itself. In the application framework approach, the framework code is a starting point. While application programmers are not encouraged to modify deep internals such as the event loop, they are perfectly free to do so if their application requires it.

Extant WIMP software architectures are in fact rather thin—they are focused on interpreting a single stream of input events, handling them entirely within self-contained widgets, and displaying changes on the screen. The deeper aspects of the application semantics, the content region of the window, are unsupported. While there are many reasons for this, one important reason is that the semantics differ so much between applications that no standard architecture or library of components can be constructed for them. A lesser cause is that creating the algorithms and data structures, while potentially time-consuming, is well within the abilities of a typical programmer.

Applying these Architectures to Haptic Environments

These reasons for not supporting the content region do not apply in haptic environments, and so the architectures in these environments will differ from those for WIMP environments. Since haptic environments are inherently spatial, the perceptual issues involved in haptic interaction techniques are extremely challenging. It is unlikely that a typical application programmer will have sufficient knowledge to implement interaction techniques providing adequate perceptual resources for users to efficiently perform their tasks, so we will need some means of packaging and distributing solutions. On the more positive side, the interaction techniques of interest in haptic environments are concerned with the content region, so there is a real possibility of coming up with packages that are useful for the content regions of many different programs. As discussed in Chapter II, one of the goals of haptic environments is to make interaction more direct by exploiting the existing situated skills of the users. This implies shifting the interaction techniques out of the control region, where a virtual object is manipulated indirectly by manipulating a control, and into the content region, where virtual objects are manipulated by directly locating the graphic and haptic cursors within the spatial region of the object. As discussed in Chapter II and empirically explored in Experiment 1 and Experiment 2, the perceptual issues involved in such interaction techniques are profound and challenging. Haptic environments therefore pose a challenge to existing styles of runtime user interface tools, since the interaction techniques of greatest importance in these environments are the most difficult to create, and are located in a portion of the application software that has proven difficult to serve with conventional runtime tool designs.

While the spatial nature of haptic environments poses problems, it also offers a potential solution. All applications in these environments have a common requirement for code to create, display, and manipulate a simulated world. Furthermore, the spatial interaction techniques will tend to be homogenous across applications, because these techniques are designed to support the common set of human situated skills described in Chapter II. Thus unlike WIMP environments, there is potentially a uniform set of interaction techniques for the *content* region of the window. Runtime tools that support simulation of three-dimensional worlds and interaction techniques for manipulating objects in those worlds may be the path to both improving the productivity with which such applications can be created, as well as providing a high quality of perceptual resources for accomplishment of spatial tasks.

Common Algorithms and Data Structures in Haptic Environments

There is a variety of key algorithms in point force environments. These algorithms are challenging to write because they are essential to high performance of both the machine and (through the provision of adequate perceptual resources) the user. As such, they represent potentially important functionality for the architecture to provide. Three of the algorithms are loops that update some portion of the simulation on the graphical or force display, two other loops update the central world model, and one loop updates the global application data. The characteristics of these six loops are summarized in Table 16. Human perceptual requirements differ for the various objects maintained by these loops, and so the preferred execution rate of these loops varies. The loops can

TABLE 16. Update Loops in a Point Force Environment

Item updated	Typical rate (Hz)	Type	QoS dimension
Haptics loop	1000+	Synch.	Global geometry
Interaction technique	80-100	Synch.	Local geometry
Graphical scene	24–40	Synch.	Local geometry
World model	1–1000	Synch.	Physical fidelity
Collision detection	1–1000	Synch.	Physical fidelity
Input event	1–2	Asynch.	User attention

also be characterized by whether they are synchronous, executing regularly at fixed time intervals, or are asynchronous, scheduled by irregularly occurring external events. The fourth column details the standard techniques by which Quality of Service (QoS) is best maintained during periods of processor overload. The column lists the dimension that can be degraded, freeing up processor cycles with least perceptual impact.

The rate of the haptics loop[2] depends upon the haptic display technology. The PHANToM point force display is designed to be updated at least a thousand times a second. The motivation for this requirement is partly the temporal resolution of the human tactile system, as the P and NPII channels can sense changes up to 500 Hz (Bolanowski, 1996), giving a Nyquist frequency of 1000 Hz. The requirement is also motivated by the stability of control algorithms, as the PHANToM algorithms can become unstable at lower update rates. Because instability poses a physical threat to the user, this forces a minimum update rate for the PHANToM—the device driver will shut down the application if the update rate falls below 1000 Hz. Given a fixed display rate, the application programmer can only respond to insufficient processing resources by reducing fidelity

---

[2] To be in accord with my nomenclature of Chapter II, I should call this the “force display loop”. However, the term “haptics loop” is so deeply established in discussions of architectures for haptic environments that I retain the more widely-used terminology in this chapter.

of displaying global geometry while keeping fidelity of rendering local geometry high to ensure stability.

The current interaction technique—the cursor, anything dragged by the cursor, and any background elements that change beneath the cursor—should all be updated 60–100 times a second. Lower rates produce perceptible flicker and jumping of the cursor for many observers (Foley, van Dam, Feiner & Hughes, 1990, p. 157). The exact performance deficits, if any, caused by this have not to my knowledge been measured, but users do report annoyance with flicker. Given limited processing resources, the typical approach is to display the local geometry of the object of interest in reduced fidelity, as for example the display of the outline of a window being dragged in a windowing system.

The overall graphical scene can be rendered at a lower rate without producing noticeable jerkiness. Cinematic displays operate at 24 Hz, while videotapes display at 30 Hz (Foley et al., 1990, p. 1058). The 24–40 Hz range is commonly cited in the graphics literature. Card, Robertson, and Mackinlay (1991) suggest a lower minimum of 10 Hz. While the required rate is lower than for the current object of interest, the complexity of the rendering task is far higher for an entire scene, so performance limitations can easily occur. Graphical rendering is degraded in the opposite manner from haptic rendering: Fidelity is preserved to the global geometry while local geometry is represented inaccurately.[3]

The world model is conceptually separate from both the graphical and force rendering loops and can be updated at a different rate, although many current systems

---

[3] Some real-time graphic systems reduce resolution the reverse way, presenting degraded images outside the foveal region (e.g., Watson, Walker, Hodges & Worden, 1997). However, if an adequate resolution cannot be represented within the foveal region, global geometry will take precedence over local features.

update the world model as part of one of the rendering loops (see next section). Slower update rates reduce the model's fidelity to an actual physical system; the consequences of this reduced fidelity are highly application-specific. A special case of the world model loop is the collision detection loop. This can be an extremely computationally intensive process, as it potentially requires comparing the relative positions of every pair of objects in the world. Due to its intensity, it may be done at a lower rate than the rest of the world update loop, or may be done by a separate thread.

The final loop processes completed discrete input events from the user, such as menu choices or picking an object in the simulated world. These should produce some acknowledgment within about a second (Card et al., 1991) to match the expectations of standard human dialogs. If processing resources do not permit this level of responsiveness, there is no algorithmic solution. Users will simply lose attention and interest in working with the application.

The key issues in the update loops vary differ for synchronous and asynchronous loops. For synchronous loops, the primary challenge is to maintain an update rate within the perceptually preferred range. For the asynchronous input event loop, the challenge is to dispatch events to the appropriate software component for handling them. This often requires identifying a specific scene object that is associated with the event.

The last algorithm is not an update loop and is in fact not a single algorithm. This is the family of algorithms that implement Quality of Service (QoS) policies. When processing resources are insufficient to produce a simulation completely at the highest level of fidelity, the QoS algorithms determine which properties will be displayed at lower fidelity. The goal is to reduce display fidelity of those properties that will have the least perceivable impact on the quality of the service. These algorithms will be discussed at length in a later section.

The central data structure of these applications is the *world model*, the representation of the simulated world: what objects are present, where they are located, and any hierarchical relationships between them. Routines for constructing and traversing this data structure are amongst the most frequently executed in the whole system, so they must be efficient. The code for traversing the structure is also nontrivial to write. Thus an architecture that provides a good set of world model routines truly enhances the application programmer productivity. The world model is often implemented as a hierarchical *scene graph*. In this chapter, I will use world model to describe the general data structure and scene graph only for those architectures that explicitly use a scene graph.

Algorithms Implemented in Current Architectures

While no tool exists that provides implementations of all the above algorithms, every one has been provided in one or more systems. Table 17 lists several systems and the algorithms implemented by each. There are two broad categories of such systems. The first six on the list provide implementations of the algorithms, often with methods for overriding specific functionality. The last system (Jacob, Deligiannidis & Morrison, 1999) provides a high-level declarative language for writing interaction techniques and managing the input event loop.

TABLE 17. Algorithms Implemented by Various Runtime Tools

System	Haptics loop	Interaction technique loop[a]	Scene loop	World model loop	Collision detection	Input event loop
OpenGL	—	—	—	—	—	—
MFC/ OpenGL	—	—	—	—	—	override
Cognitive Coprocessor	—	override	override	—	— (?)	override
GHOST/ OpenGL	override	—	—	override	PHANToM only	—
GHOST/MFC/ OpenGL	override	—	—	override	PHANToM only	override
Java3D	—	override 3-D pick	fixed	static[b]	override	override
Jacob et al.	(NA)	constraint declarations	(NA)	(NA)	(NA)	state transitions

[a] All systems can take advantage of the multiplicity of WIMP 2-D widgets available.

[b] The package maintains a world model data structure, but all updates to it must be programmed by the application programmer using overrides.

Architectures Providing Implementations of the Algorithms

The simplest implementation of a haptic application uses only a three-dimensional graphics package such as OpenGL[4]. In this configuration, the programmer is responsible for writing all the central algorithms of the application. This trades off efficient use of the machine for inefficient use of the programmer. While many current research systems are built this way, it does not scale up well to the demands of writing commercial applications.

Haptic applications invariably require more than just spatial interactions. The typical bookkeeping operations, such as creating windows, opening or saving files, or setting options, can be handled well enough by traditional WIMP methods. These technologies can be readily incorporated by an application framework such as Microsoft Foundation Classes (MFC) or MacApp, which provide a standard event loop implementation whose behavior can be overridden at various points. This improves programmer productivity for the bookkeeping code, but provides no assistance for the heart of the haptic application.

One of the central algorithms of a haptic application is animated graphical rendering of the simulated world. Robertson, Card, and Mackinlay (1989) developed the Cognitive Coprocessor architecture to support graphical animation of abstract three-dimensional worlds. The architecture implements a central scheduling loop for both the input and the output rendering tasks. The Cognitive Coprocessor presumes that each application is running in one or more threads and itself serves as a moderator for access

---

[4] There are many packages implementing various forms of three-dimensional rendering. Haptic environments require real-time rendering, which currently can only be satisfactorily performed by polygonal renderers. OpenGL is by far the most widely used polygonal rendering package and so I use it as an exemplar of the whole class.

to the shared resources of input devices and display screen. The Cognitive Coprocessor is unique amongst all the systems described in that it was explicitly designed to provide performance that matches human perceptual rates (Card et al., 1991). The system features an innovative *Governor* mechanism that monitors the response rate of the visual display and ensures an acceptable QoS. The rendering tasks query the Governor and reduce their rendering quality when the processor becomes overloaded. Note that the Governor simply acts as a central repository of performance information. Unlike the governor of a mechanical engine, which actually restricts the rate of rotation, the Cognitive Coprocessor Governor relies upon the cooperation of the rendering tasks to reduce their performance demands.

None of the systems described so far provides any explicit support for a haptics loop. The GHOST Software Development Kit (SensAble Technologies, 1998) for the PHANToM organizes the application around a haptics loop. Graphics are typically rendered using OpenGL, although GHOST does not presume any particular graphics package. The application programmer specifies objects in the simulated world as a scene graph. The GHOST haptic loop provides default point force rendering for simple geometric objects and polygonal meshes plus the ability to render custom shapes through overriding. The haptics loop not only implements haptic rendering, it also updates the world model, computing the trajectory of every moving object in each time interval. The haptics loop also detects collisions between the PHANToM and any objects. This is a simpler case than the general collision detection algorithm. If the application requires the more general algorithm, the programmer must explicitly code it. GHOST uses a two-threaded implementation, with one thread devoted to the haptic loop and the second thread used for all other application code. GHOST provides routines for synchronizing access by each thread to the world model. It is relatively straightforward to implement 2-D WIMP interaction techniques in a GHOST application by embedding GHOST into the MFC framework, providing an overridable event loop.

Java3D

The most ambitious commercial-grade 3-D graphics package to date is Java3D[5]. Java3D implements a superset of all the features in the packages described so far, except a haptics loop. Java3D is organized around a scene graph. Unlike any other system described here, Java3D provides a single, fixed graphical rendering algorithm for that scene graph, without overrides. While the application programmer can provide hints to the algorithm by specifying *capabilities* for each object, such as whether the object will move or not, the programmer cannot insert code into any point of the rendering process. Even the polygonal shading algorithms are fixed (and unspecified—presumably an implementation can use either Gouraud or Phong shading (Foley et al., 1990, Sect. 16.2)). Java3D also implements a collision detection algorithm, a significant productivity gain for the application programmer. In the default behavior objects simply pass through each other—the programmer must provide the routines that define object behavior when collisions occur. Java3D supports multiple graphical Levels of Detail (LoD), a form of QoS. However the choice amongst LoD is based entirely on distance from the user; there is no mechanism such as the Cognitive Coprocessor's Governor by which the application can determine that the LoD must be adjusted to meet processor demands.

The input architecture of Java3D is also more sophisticated than earlier systems. It is organized around an Event and Behavior architecture. The application specifies

---

[5] Some, but not all, of the features of Java3D are also provided by OpenInventor.

behavior for an input Event, such as a mouse click, by registering one or more Behaviors for that event. The Behavior code is called by the framework whenever that Event occurs. While most interaction techniques must be coded from scratch by providing Behaviors for basic events, routines are provided to select items from the scene graph according to geometric criteria. These routines can be used to implement Pick input tasks, where the user selects an item by pointing to it. The Java3D input model also provides explicit support for head-tracked displays, automatically coordinating changes in the displayed view with changes in the user's head position. It also supports input from 6-degree of freedom (DOF) devices such as the Polhemus or PHANToM.

The Java3D Application Programming Interface (API) is entirely designed to permit multithreaded implementation. Application-defined subclasses of Behavior, which implement the world loop and input event loops, are executed in nondeterministic order, perhaps even in parallel. While increasing the possibility of improved performance in multiprocessor systems, this also increases the complexity for the application programmer, who now may have to perform synchronization between the Behaviors in the application.

The Java3D architecture derives considerable benefit from maintaining a central scene graph. The scene traversal and object rendering algorithms can be combined for greater efficiency. Nodes that do not have the capability to move can have their transformation matrices partially precomputed. Collisions with these nodes can also be detected more efficiently. The Pick routines also use the scene graph (and some of the same data structures used for collision detection) to select objects by spatial location. Storage efficiencies ensue from only having a single copy of these data structures that is used by multiple algorithms. While Java3D supports a world model (in the form of a scene graph), it does not implement a world update loop. Unlike the GHOST haptic rendering loop, the Java3D graphic rendering algorithms do not compute object motion. This must be programmed for the application using Behavior nodes.

At this time, Java3D provides the most complete support of any commercially-available system for the construction and graphical rendering of simulated 3-D worlds.

Architectures for Improving Programmer Productivity

Where the six previous systems all provided implementations of common algorithms, Jacob, Deligiannidis, and Morrison (1999) instead proposed a system that supports more rapid implementation of algorithms. Jacob et al. defined a state transition network notation to program the input event loop and a constraint notation to program the continuous-valued relationships of the interaction techniques. The constraint language used for interaction techniques is declarative and nonprocedural. The constraints can be solved by one of several constraint solving algorithms, either backward- or forward-chaining. In this way, the nonprocedural program specified by the application programmer can be implemented by one of several pluggable algorithms. Jacob et al. focused on the specification of graphical interaction techniques and their links to the underlying model of the program. In a later section I will discuss how this nonprocedural approach may become even more important in haptic applications.

Architectures Providing All the Common Algorithms

The GHOST architecture is the current state of the art for construction of haptic environments. The Java3D architecture shows promise as a basis for graphically rendering 3-D worlds and might be extended to include haptics. In this section I compare

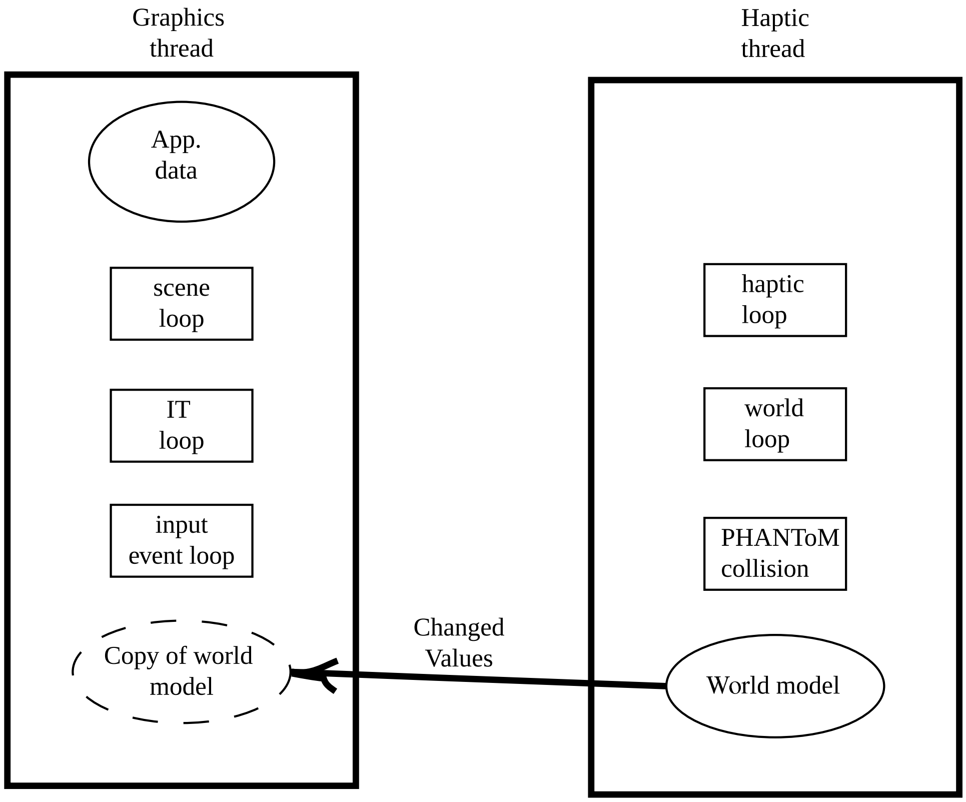


FIGURE 33. The GHOST Architecture

these two approaches, showing the strengths and weaknesses of each.

I will first consider the architecture of the GHOST system in detail. Systems built under this architecture are the only current systems that implement all the algorithms described in the previous sections. Figure 33 represents the major components of the GHOST architecture when used without MFC. The architecture features two threads, one called the “haptics thread” and one called the “graphics thread”, although each thread implements several of the loops described in Table 16.

The haptic thread is extremely high performance, executing at 1000 Hz. This thread maintains the world model, in the form of a scene graph. The haptics rendering loop, the world update loop, collision detection for the PHANToM, and any user-provided generalized collision detection execute in this thread.

The graphics thread runs at lower priority and executes at a far lower rate. The primary function of the graphics thread is graphical rendering of the scene, and it is this that consumes most of its time. The graphics thread generally implements interaction techniques[6] and the input event loop. A central aspect of this architecture is the way the two threads manage access to the world model. The master world model is maintained by the haptics thread. The graphics thread never directly accesses this data structure but instead keeps a shadow copy of the world model. The GHOST API provides a function called by the graphics thread to transfer any changed values from the haptic thread world model to the graphics thread copy. This design minimizes memory contention between the two threads for the shared data structure, allowing the haptics thread to run at a high rate, at the expense of maintaining two copies of the world model.

This design fulfills many of the goals for a user interface architecture outlined at the beginning of this chapter. It provides the bookkeeping code associated with constructing a scene graph and accessing it from two threads. It provides a high-performance implementation of the essential (and complex) haptic rendering and world simulation loops and part of collision detection.

However, it does not fulfill some other important goals. The important interaction technique loop is updated at whatever rate the graphical loop runs, typically well under 80 Hz. While GHOST provides the bookkeeping code for the world model, it provides no support for maintaining the copy in the graphics thread. Most importantly, the group-

---

[6] However, the gstDynamic class can be used to implement interaction techniques in the haptics thread.

ing is rough: Far too much unrelated functionality is thrown together in the graphics thread. This code belongs to different performance strata (the higher performance of the interaction technique loop, the asynchronous performance of the input event loop) and performs logically distinct functions. The duplication of data structures also suggests a problematic organization. As the size of the application grows, the grab-bag nature of the graphics thread is likely to become more troublesome.

Adding Haptics to Java3D

Figure 34 illustrates the architecture of Java3D without haptics. The entire architecture is organized around the world model. The central box, which I term the *rendering box* features the key algorithms for traversing and maintaining the world model and graphically rendering it. All the rest of the application is linked to the central box through instances of Behavior. Note that the rendering box will typically be implemented as several distinct threads. The box notation does not indicate a single thread in this case, but a collection of threads programmed to appear as consistent as if they were a single thread.

Figure 34 suggests that the duplication problems of the GHOST architecture have been resolved, but this is misleading. The GHOST system required explicit synchronization between the haptics and graphics thread because the haptics thread implemented the world update loop and actively changed the world model. Java3D does not implement the world update loop in the rendering box, but instead requires the application to implement the loop through instances of class Behavior. These instances are likely to contain localized state information apart from the scene graph, constituting extensions to or duplicate contents of the world model.

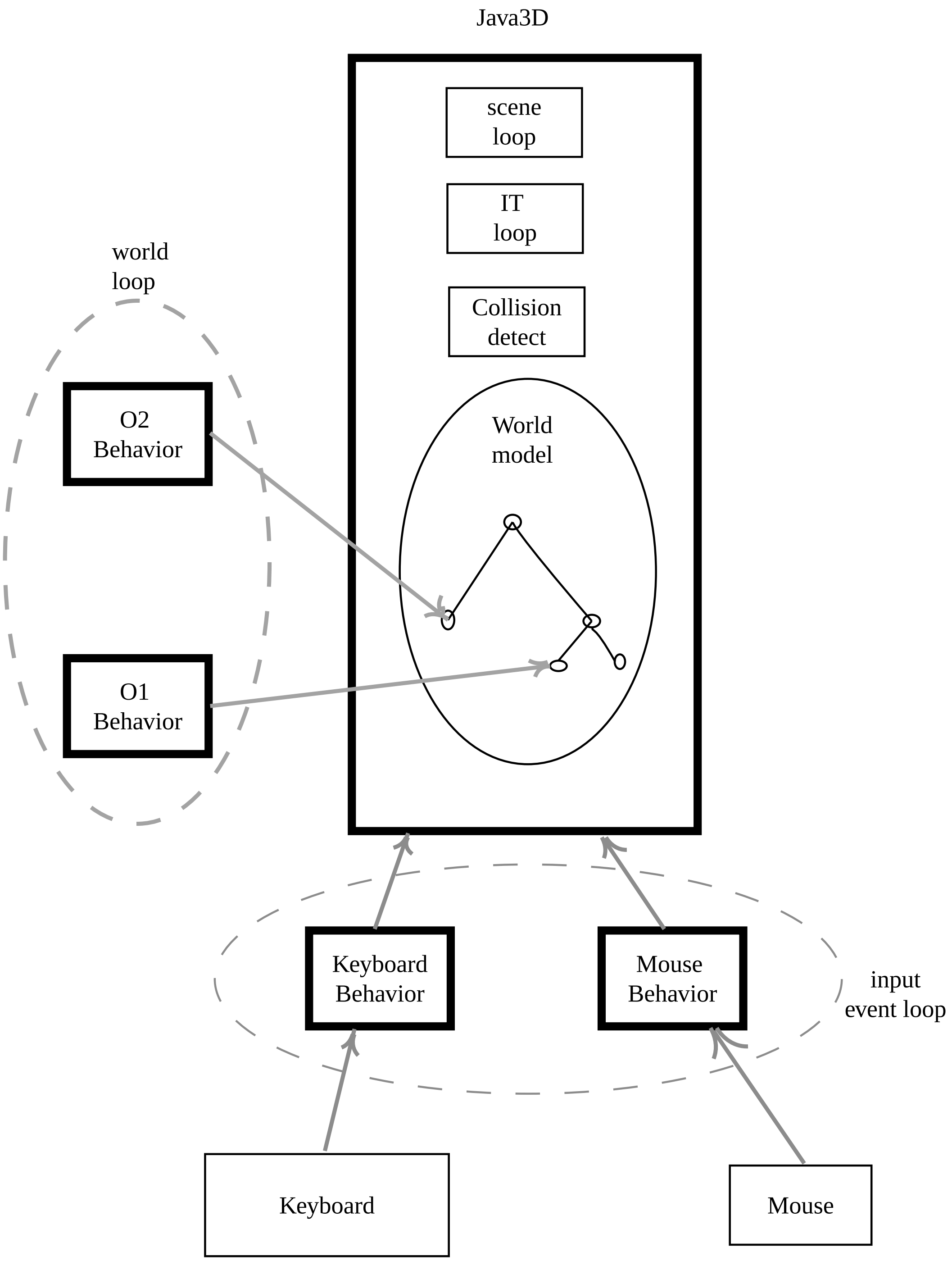


FIGURE 34. The Java3D Architecture

Figure 34 also illustrates the reverse phenomenon of Figure 33: Where the architecture of GHOST groups many unrelated functions together, the Java3D architecture spreads each function thinly across many threads. The world update loop, the interaction technique loop, and the input event loop are all spread across a multiplicity of Behavior instances. Each instance is executed nondeterministically from the others, with no guarantees of their relative order. Java3D provides a limited synchronization facility through *postId events*, and of course the considerable synchronization facilities of the Java language are also available, but these merely mitigate the problem rather than solve it.

Adding a haptic rendering loop to Java3D introduces some new problems. The most likely place to put another rendering loop is in the rendering box. This provides full access to the world model and all associated data structures, allowing the haptic loop to take advantage of optimizations as the spatial tree of bounding volumes that was already constructed for the graphics rendering. Adding the haptics loop to the rendering box may not be straightforward, for several reasons. First, the high performance demands of haptics rendering may cause contention for the scene graph with the other threads accessing it. While Java3D derives great benefit from being built around the scene graph, it also becomes vulnerable to access contention. Second, the haptics rendering algorithms may require different data structures from the graphics rendering. For example, the required density of the polygonal mesh may be different for the two algorithms. Third, implementing the haptics loop within the rendering box probably requires inextricably mingling the functions for both graphics and haptics rendering. As a result, haptics could not be added to an existing Java3D implementation simply by installing a device driver, but would require a major revision of the code.

An alternative approach would place the haptic loop outside the rendering box. This would eliminate the above problems but now a second copy of the world model would be required, which would have to be kept synchronized with the world model in the same way as the GHOST architecture. With this implementation, the distinctions between the Java3D and GHOST architectures become small indeed.

Ultimately, we do not currently have a satisfactory architecture for haptic environments. Architects are confronted with two opposing needs. The central role of the world model argues for a single copy shared between the two rendering loops, but this introduces potentially unacceptable levels of contention. Making separate copies for the two rendering loops minimizes contention at the cost of extra memory and synchronization between the copies. Second, current architectures either group unrelated functionality together or spread related functionality too far. Research is needed to see to what degree these several opposing factors can be resolved.

Quality of Service Issues

The previous sections have considered architectures for integrating all the key algorithms for virtual environments. The proposals so far have gathered the algorithms unchanged into a single architecture. A more sophisticated approach might integrate the graphics and haptics rendering algorithms into a whole greater than the sum of the individual parts. In particular, such an integration might open up new possibilities for Quality of Service (QoS) algorithms.

While QoS has not been considered to date in haptic environments, it has been an active area of research in multimedia. For a digitized stream of images or sound to be perceived as high quality it must be displayed at a steady rate. To achieve this steady display rate, researchers have suggested rate-controlled disk I/O scheduling (Reddy & Wyllie, 1993), process scheduling (Yau & Lam, 1996), and network transport layers (Campbell & Coulson, 1996). Many of these proposals focus on maintaining a minimal transfer rate, with no provisions for fluctuations in the amount of processor time the media loader will need. This is appropriate for digitized media, where the amount of time to load or display a frame shows little variance. One exception to this is network transfer, where changing network load can produce significant variation in the transmitted frame rate. When the network slows down, the media transfer algorithms maintain a constant transfer rate first by fine-grain adjustments, degrading the image resolution, and then by coarse-grain adjustments, dropping entire frames (Campbell & Coulson, 1996).

Unlike media streams, the processing demands of interactive applications vary widely depending upon the data they are processing. Guaranteed-rate algorithms will therefore not be of use in haptic environments. Mechanisms for degrading quality in the least perceptually objectionable way will be required. A system such as the Cognitive Coprocessor (Robertson et al., 1989), which monitors the total level of processor utilization and coordinates the QoS adjustments by the various tasks would be useful. Nakajima and Tezuka (1994) constructed a similar QoS coordinator that degraded image quality for less important video displays while maintaining a high quality for the most important display.

Haptic environments are novel in that they feature several very different rendering algorithms whose quality is least degraded by very different types of changes (Table 16). As discussed in Chapter II, the relative importance of the display modalities will vary with the task. A central QoS coordinating mechanism in a haptic environment would therefore have both more flexibility than previous systems (such as the Cognitive Coprocessor) and have a more demanding job. The application might well have to provide it with hints, akin to the capability mechanism of Java3D, indicating which display modality should have its quality maintained and which one should be degraded.

A central QoS governor is more effective when it is less constrained by details. If the rendering algorithms for point force devices are coded in procedural form where every rendering detail is specified, there is little room for adjustment. On the other hand, if the algorithms are coded in nonprocedural form, the leavning more rendering details to the architecture, the QoS governor would have more leeway to adjust the quality. For this reason, the constraint notations of Jacob et al. (1999) might prove useful not only for providing a more concise and productive means of expressing interaction techniques, but also to provide finer adjustments of rendering quality as the processor load varies.

Conclusions

Software architectures for haptic environments are still in their infancy. Robust HEs require the integrated use of several complex and resource-intensive algorithms, making the architectures complex. The central role of the world model imposes contention restrictions on multithreading, while the multiplicity of update loops makes it difficult to organize the code in a way that cleanly groups related routines.

Overall, the strongly centralized model of Java3D offers the best prospects, but there are potential limitations and the impact of those limitations will not be known until a haptic version of Java3D has been implemented. In the longer term, integrating the graphic and haptic rendering routines under a combined quality of service governor may produce architectures better suited to human perceptual needs. The use of nonprocedural specifications may be essential to such QoS mechanisms.

CHAPTER VI

CONCLUSIONS AND FUTURE WORK

The preceding chapters individually the context of point force environments (and haptic environments), some initial empirical data on human performance in those environments, and the architectural implications of those environments. In this chapter I consider the implications of the empirical data and combine these results into a long-term research agenda for haptic environments.

The Standard Model of HE Design and the Results of Experiments 1 and 2

Current designs for point force environments have produced only one success story. After over five years of production of the PHANToM hardware, the only commercial point force environment is FreeForm (SensAble Technologies, 2000). I believe that our current design theory hampers the development of usable PFEs because it does not account for the fundamental differences between them and physical environments. The assumptions of our design theory affect our designs, our science base, and our evaluation methods. This section makes the case that the current design perspective relies upon a “world simulation” model. In summary, the world simulation approach assumes that perception is a transparent process and that the user interprets the graphical and force displays exactly as the designer intended. In contrast, I believe we must explicitly focus on the interpretive nature of perception. The approach emphasizes that perception is itself a task, which the observer performs using the available resources. In this section I consider the world simulation model in light of Experiments 1 and 2 in this dissertation.

Nearly all of the work to date on “sensory redundancy” and “sensorial transposition” in virtual environments (e.g., Srinavasan, Beauregard & Brock, 1996; see also the review in Stanney, Mourant & Kennedy, 1998, Sect. 2.4) presumes that percepts are constructed as the sum of independent unitary sensations directly derived from the stimuli provided by the display modalities. These are all assumptions of the world simulation model. This model describes perception at a rather high level, and studies based upon it can provide useful design guidance for display modalities which are familiar to the user and for interaction techniques producing discrete input values. However, these assumptions do not apply in situations where the sensation is unfamiliar to the user and the user must learn to interpret it, perhaps even engaging in problem-solving behavior. They do not apply in situations where the sensations are not independent, where the presence of a second sensation changes the first, as for example when one sense makes attentional demands that inhibit the interpretation of another. And they do not apply in situations where one sensation guides the performance of another, as for example the use of vision to guide the motor skills upon which a haptic sensation depends.

The results of Experiments 1 and 2 are evidence that many interactions in point force environments do not satisfy those assumptions. The point force display modality is unfamiliar. Experiment 2 found evidence of substantial learning over the course of 60 (and in two cases, 180) trials. Typical point force interaction techniques involve extended exploration of or interaction with a stimulus rather than a brief interaction producing a single input event. Interactions in Experiment 2 took upwards of 23 seconds. Point force haptics can conflict with the experience of other stimuli. Experiment 1 suggests that adding haptics reduced the effectiveness of curvature discrimination. Point force haptics can make large attentional demands. While no formal measure of mental workload was taken in Experiment 2, it is unlikely that participants could have attended to any other stimuli while performing the shape recognition task—many found it difficult to even attend to the visual cursor. Subjective measures of mental workload have been used in previous studies of haptic environments (e.g., Oakley et al., 2000) and could well be applied in future versions of these protocols. Objective measures such as response times on a probe task could also be used.

However, Experiments 1 and 2 did not provide evidence that vision and haptics experience can combine to produce a more complete percept of either curvature or shape. In each case, I have suggested that confounding factors overwhelmed the predicted effect. The task in Experiment 1 probably offered sufficient visual cues that the single cue provided by haptics could not improve performance. Participants in Experiment 2 may have been learning how to interpret the PHANToM and the visual cursor. Whether the predicted effects will be found when these factors are controlled can only be determined through further experiments. At the very least, the small effect sizes in these experiments indicate that the interactions of vision and haptics are not so powerful that they appear regardless of other conditions. The world simulation model assumption that sensations are unitary appears to be correct to a first order approximation.

The overall results of Experiments 1 and 2 certainly indicate that the intersensory interactions in these environments are rich and differ in important ways from physical environments. While the evidence to date does not provide direct guidance on which perceptual resources will be most important for an arbitrary task, it does support the claim that designers of haptic environments clearly must attend to perceptual resources in their designs. Experiments focusing on the detailed processes of perceptual interactions can provide important data.

Individual Differences of Performance

If the stimuli in a haptic environment serve as perceptual resources for use at the discretion of the user, then we should see evidence of individual differences in performance in Experiments 1 and 2. It is difficult to define individual differences in Experiment 1, due to the inherently subjective scales used in magnitude estimation. The range of individual raw scores, about an order of magnitude, corresponds with the range usually obtained from an absolute magnitude estimation protocol (Zwislocki & Goodman, 1980). The range of slopes of the psychophysical functions (computed from normalized scores) is half an order of magnitude (.29–1.21). This is difficult to interpret, as the psychophysical slopes reflect not only perceptual differences but also differences in how participants assign numeric values to sensory experience. In the original design for Experiment 1, I proposed using sensory response function estimation (Gescheider, 1997, pp. 274–285) to correct for the latter form of individual differences. However, in pilot studies, the sensory response correction increased rather than reduced variance of the scores, so I did not use the method in the actual experiment. The constrained scaling method (West et al., 2000) is a recently developed alternative that may permit analyzing individual differences within a magnitude estimation protocol.

Definite individual differences are observable in the data of Experiment 2. Figure 25, Figure 26, and Figure 31 show large differences in the location and spread of the response times, while Figure 27 shows accuracies ranging from 67% to 98%.

The data of Experiments 1 and 2 provide only hints as to what perceptual and cognitive factors might cause these differences. Participant reports of estimation strategies in Experiment 1 did not correlate with actual performance, so conscious strategies did not appear to substantially determine performance. The mental rotations test proved to be of limited explanatory value. It provided a model that explained a useful proportion of the variance for the slope of psychophysical functions in Experiment 1. However, it provided no useful model for Experiment 2 because the range of mental rotation scores in Experiment 2 was more narrowly clustered than for Experiment 1. Apparently, the individual differences observed in Experiment 2 were a consequence of factors that are not measured by the mental rotations test. This does not necessarily contradict the model derived in Experiment 1—perhaps the mental rotations test might have produced a useful model in Experiment 2 for a group of participants with a larger range of mental rotations abilities.

Nonetheless, the lack of utility of the mental rotations test in Experiment 2 dampens my enthusiasm for it. The relationship between experiments and models is paradoxical: Experiment 1, which was predominantly perceptual, produced a useful model, while Experiment 2, which had a far higher cognitive component (specifically including rotations in three dimensions), did not. The mental rotation result of Experiment 1 may be due to chance. For now, I recommend continuing to use the test, if only because it is so easy to run, gathering enough replications to determine the conditions under which it produces reliable predictions. The range of differences of Experiment 2 also suggests we should search for other psychometrics that explain variation unaccounted by mental rotations.

Transfer of Situated Skills and Naturalness

Chapter II defined the concept of situated skills and made the case that haptic environments permit a general transfer of skills learned from tasks in physical environments to tasks in virtual environments. The results of Experiments 1 and 2 show mixed evidence for such transfer in the case of point force environments. While Experiment 1 did not expressly address the issue of transfer (due to the subjective nature of magnitude estimation), the discussion of the experiment demonstrated that haptic curvature estimation using a single finger probably uses attitude cues, which are unavailable in point force environments. Whatever the relative performance of curvature estimation in physical and point force environments, it seems clear that the mechanics will be different and hence some large portion of situated skills will not transfer for this task.

However, Experiment 2 showed some evidence of transfer. The geometric mean response time of 23 seconds in that experiment was within the range of recognition times found for haptic shape recognition of physical objects, which themselves tend to be rather slow (see Table 14). It appears that participants were able to perform the task—there was transfer of some situated skills of shape perception—but they suffered a 2–3 fold reduction in performance, presumably due to temporal integration.

The goal of transferring situated skills was to provide a more natural form of interaction with computers. The results of Experiment 2 demonstrate how far away that goal remains. A response time of 23 seconds is unconscionably long for practical application of haptic shape recognition in point force environments. Given the central role of shape recognition and curvature estimation are likely to play in haptic environments, this performance must be improved. Research into improving performance in these tasks, either through better interaction techniques for point force environments or else a shift to different force display technologies that permit application of broad EPs, must be a central priority of the haptic environment community.

On pp. 38–39, I listed two criteria for assessing whether the benefits of physical interaction will be realized in haptic environments. First, do the affordances from physical environments carry over, and second, how many of the rich repertoire of movement skills learned in physical environments can be directly applied in haptic environments. On pp. 51–54, I argued that graspable environments will do well by the first criterion but not the second, while for point force environments the reverse will hold. For this argument as well, the results of Experiments 1 and 2 provide mixed support. Participants in Experiment 1 appeared to apply visual affordances learned in physical environments directly to the experimental stimuli, despite the reduced nature of the depth cues in those stimuli. As a result, they were able to visually estimate curvature with little trouble. As noted above, Experiment 2 showed some evidence of movement skill transfer, but at substantially reduced effectiveness.

No two experiments could provide definitive resolution of these questions. I believe that point force environments do offer a potential increase in naturalness of interaction, but the results of this research suggest these gains will be neither easy nor automatic. We clearly need a much better understanding of the mechanisms of perception in these environments before we will be able to construct environments of widespread utility.

A Research Agenda for Point Force Environments

This dissertation has presented some preliminary research into factors determining human performance in point force environments. Because it is a preliminary effort, it is unavoidably vague and tentative—every point requires further experimental testing and refinement. The development of a theory, experimental methods, and a base of empirical results will be an iterative process, with the theory guiding the choice of experimental protocols and the resulting data requiring revisions to the theory. The following points represent the outlines of this process.

Development of Evaluation Tasks

Good evaluation tasks catalyze research. For example, theories of rapid aimed movement (e.g., Fitts, 1954) and theories of pointing device design (e.g., Douglas et al., 1999; Douglas & Mithal, 1994) have both benefited tremendously from use of the Fitts task as a reliable protocol for evaluation. The task is valuable because it focuses attention on the two stimulus factors most important in evaluating performance, target distance and width, and how they combine to determine the difficulty of the task. The associated analysis method allows human or device performance to be evaluated in terms of a single value, the index of performance. The value of the Fitts task is not that it describes all the factors influencing performance, but that it focuses attention on the *important* factors.

The design of point force environments would benefit tremendously from analogous tasks focusing attention on the variables of interest. Given that these environments have much more general movements than two-dimensional pointing, we will probably need several evaluation tasks. The experiments in the next two chapters present two candidate tasks.

Isolating the Determining Factors of Human Performance

An evaluation task is an operationalization of a theoretical claim that certain factors are the important determinants of performance. The development of the task and the theory go hand in hand. This chapter has suggested some broad groups of factors that may prove significant in point force environments:

1. Perceptual cues: The visual and haptic affordances of skilled motor control for a task. The reduction in performance due to the absence of some of those cues in point force environments.

2. Temporal integration: The increased cognitive load ensuing from reducing multiple simultaneous points of contact to a single point.

3. Intersensory integration: The reduction in performance due to attentional and motor control conflicts between perceptual systems. The enhancement of performance due to the replacement of missing cues for one perceptual system by cues from another system.

4. Learning: The initial level of user performance. The improvement of skill with practice.

5. Individual cognitive differences: The cognitive factors, varying across individuals, that determine performance.

For each of the above factors, empirical data is required to assess how large a performance effect it has, how its importance varies from task to task, and how it might interact with other factors.

Attention and Workload

Are there limited central cognitive resources governing the use of perceptual systems in point force environments? For example, does the use of one perceptual system, such as haptics with a point force device, limit the effectiveness of vision? How do these attentional demands change with practice?

Work on this issue can be a contribution to general models of human cognition as well. Kieras and Meyer (1997, p. 397) argue that performance limitations demonstrated to date can all be explained as structural limitations in human motor control, perception, or memory rather than as consequences of a central processing bottleneck in human cognition. However, the tasks considered in point force environments are more complex combinations of motor skills and perception than have been used in previous experiments and might exercise novel combinations of these faculties. Empirical data on human motor and perceptual performance in point force environments can refine general theoretical models such as EPIC (Kieras & Meyer, 1997).

Psychometric Assessments of Cognitive Factors

If individual cognitive factors are found to influence performance, then reliable and valid psychometrics must be developed to measure these factors.

Technologies for Training

If absent perceptual cues limit human performance in point force environments, what training aids might allow rapid adjustment to other cues provided by the environments? For example, for a user hampered by the absence of binocular disparity in a desktop haptic environment, what training aids might facilitate the user learning to rely on the available cues, such as object size, perspective, texture, and haptic cues?

Path Analysis for Hand Movements

Broad factors that determine group differences can be experimentally assessed by comparing group means using analytic tools such as analysis of variance. However, differences between individuals and between trials of the same individual cannot be assessed using aggregate measures. Instead, the actual path taken by the hand over time must be modeled. This kind of analysis has been done in other domains of psychology and human computer interaction to determine the cognitive implications of haptics (Lederman & Klatzky, 1987), the microstructure of pointing device movement (Mithal & Douglas, 1996), the usability of color model interfaces (Douglas & Kirkpatrick, 1999; Wells & Tassinary, 1998), and the compatibility of device movement constraints and visual feedback (Jacob, Sibert, McFarlane & Mullen, 1994).

Similar analysis methods need to be developed for point force environments. The process is somewhat easier in these environments because a digitization of human movement is inherently available. However, the statistical models for analyzing the digital trace remain an open and challenging area for future research. Given such models, the exploratory procedures performed by users of PFEs can be categorized and compared with the established research base of movements of observers in physical environments.

Conclusion

Given a task and a user population, the usability of a virtual environment results from tradeoffs along three dimensions: hardware sophistication, software sophistication, and user learning. The designer can sometimes get higher usability by adopting more

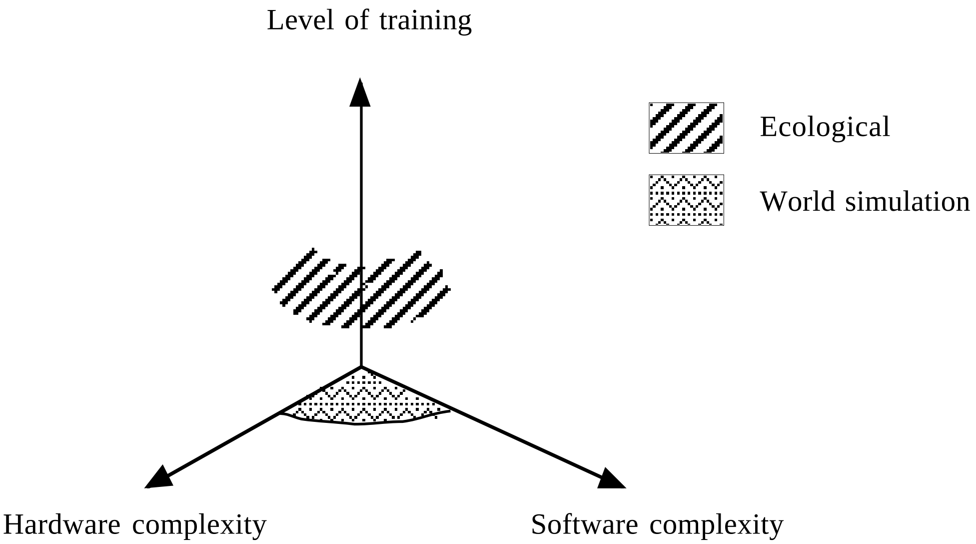


FIGURE 35. The Design Space for Haptic Environments

sophisticated display and transduction hardware. The designer may instead opt to spend more time and money developing interaction techniques that better support the task for given hardware. Or the designer may decide to require that the users learn new skills to interact with this environment. These three dimensions are graphically represented in Figure 35.

The world simulation approach is based upon the hope that haptic environments can be designed in a small region close to the origin of Figure 35, where relatively simple hardware and interaction techniques provide a level of functionality that users can access with little learning, relying on their existing base of situated skills. The evidence assembled in this dissertation suggests that this hope is naive. Haptic environments are so different from physical environments that users will have to learn new skills of interaction. Furthermore, the simplest combinations of hardware and interaction techniques are so different from physical environments that the learning burden may be quite high.

A more realistic point in the design space may be further away from the origin. We may need more complicated hardware, such as binocular displays or sophisticated input devices for the nondominant hand. We will surely need more carefully-designed interaction techniques. And in the end, despite our best efforts, some learning will still be required. An important research direction is exploring the kinds of training tools that might reduce that learning burden.

While the degree of effort implied by this proposed research is high, so is the possible payoff. I hope that this dissertation has demonstrated the central role of situated skills in human experience. I find it hard to imagine a “natural” form of human-computer interaction that does not engage the human user’s situated skills in some significant degree. However slow the progress might be, the ultimate increase in usability of computer systems will make the effort worthwhile.

APPENDIX

DATA DISPLAYS

Stem plots and five-number summaries are numeric displays introduced by Tukey (1977) and later extended by various authors (Hoaglin, Mosteller & Tukey, 1983). These displays are designed to provide clear summaries of the data along with as much of the actual data as can be reasonably represented on the page. Current statistical style guides (e.g., Wilkinson & APA Task Force on Statistical Inference, 1999) recommend using displays such as these that incorporate summary statistics and individual data points. These displays present supplement estimates of location (the mean or median) with such data characteristics as spread, symmetry, heaviness of tails, and extreme data points.

Stem plots (sometimes called stem-and-leaf plots) are used for relatively small data sets. The plot shows both the original data points (to two significant figures) and their distribution. A stem plot is essentially a histogram drawn with the trailing digits of the actual data values rather than bars. The leading digits of all values are listed in ascending order in the left column. Then the second digit of every data value that has the same first digit is listed to their right. For example, the values 1.2, 1.5, 1.9, and 2.3 would be displayed as:

1	259
2	3

This representation provides a visual indicator of the grouping of data together with the actual data values. The stem display is often annotated with some extra data (Emerson & Hoaglin, 1983). The units of the values are listed above the display. Thus the values .12, .23, .25 would be displayed as:

(unit=.1)

1	2
2	35

The depths of the values are written in a separate column on the left-hand side. The depth, the number of values from the nearest extremum, can be used to assess features such as symmetry of the dataset. For example, corresponding order statistics in each tail can be readily determined by looking for the two matching depths. For the row (if any) containing the median, the “nearest extremum” is ambiguous, so the count of row values is used rather than the depth, enclosed in parentheses to indicate that it is not a depth. Finally, if a value is significantly separated from the body of the data, it is written separately. Taking all of these principles together, the values .023, .034, .035, .036, .044, .051, and .100 would be displayed as:

Depths	(unit=.1)	
1	2	3
(3)	3	456
3	4	4
2	5	1

High: .100

Back to back stem plots can be used to contrast the raw data values of two data sets. The only difference from basic stem plots is a second data set is displayed on the left, rather than the depth count.

Seven-Number Summaries

Seven-number summaries, also called “letter values”, are a concise display of important characteristics of the distribution of values of a data set. The displays are useful for data sets that are too large to list in a stem plot. Instead, a structured summary of percentile statistics is displayed.

Table 18 is a legend for seven-number summaries. The top line lists the number of values and arithmetic mean of the data set. The next two lines summarize the location and spread of the central portion of the distribution, giving the 25%ile, the median, and the 75%ile values. The next two lines indicate the heaviness of the two tails of the distribution, giving the 0%ile (minimum value), 12.5%ile, 87.5%ile, and 100%ile (maximum value).

The middle column, termed the *midsummary* values (Hoaglin, 1985), is the arithmetic mean of its neighbors. Comparing the midsummaries to the median indicates the skewness of the distribution for the central portion, the middle tails, and the extrema. Thus if the average of the 25%ile and 75%ile is larger than the median, the central portion of the distribution is positively skewed.

See Hoaglin (1983) for a more detailed description of computational methods and tradeoffs in the use of these displays.

TABLE 18. Legend for Seven-Number Summaries

		n = (number of values), mean = (arithmetic mean)		
		50%ile (median)		
	25.0%ile	$\left(\frac{25.0\%\text{ile}+75.0\%\text{ile}}{2}\right)$	75.0%ile	
	12.5%ile	$\left(\frac{12.5\%\text{ile}+87.5\%\text{ile}}{2}\right)$	87.5%ile	
(min)	0.0%ile	$\left(\frac{0\%\text{ile}+100\%\text{ile}}{2}\right)$	100.0%ile	(max)

BIBLIOGRAPHY

Akamatsu, M., MacKenzie, I. S., & Hasbroucq, T. (1995). A comparison of tactile, auditory, and visual feedback in a pointing task using a mouse-type device. *Ergonomics, 38*, 816-827.

Akamatsu, M., & Sato, S. (1994). A multi-modal mouse with tactile and force feedback. *International Journal of Human-Computer Studies*, *40*, 443-453.

Arsenault, R., & Ware, C. (2000). Eye-hand co-ordination with force feedback. In *Proceedings of CHI 2000* (pp. 408-414). New York: ACM Press.

Baecker, R. M., & Buxton, W. A. S. (Eds.). (1987). *Readings in human-computer interaction: A multidisciplinary approach*. San Mateo, CA: Morgan Kaufman.

Bereby-Meyer, Y., Leiser, D., & Meyer, J. (1999). Perception of artificial stereoscopic stimuli from an incorrect viewing point. *Perception & Psychophysics*, *61*, 1555-1563.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, *94*, 115-147.

Bloomenthal, J. (1994). An implicit surface polygonizer. In P. S. Heckbert (Ed.), *Graphics Gems IV* (pp. 324-349). New York: AP Professional.

Bolanowski, S. J. (1996). Information processing channels in the sense of touch. In O. Franzen, R. Johansson, & L. Terenius (Eds.), *Somesthesis and the Neurobiology of the Somatosensory Cortex* (pp. 49-57). Basel, Switzerland: BirkhŁuser Verlag.

Brooks, F. P., Jr. (1988). Grasping reality through illusion—Interactive graphics serving science. In *Proceedings of CHI 88* (pp. 1-11). New York: ACM.

Brooks, F. P., Jr., Ming, O.-Y., Batter, J. J., & Kilpatrick, P. J. (1990). Project GROPE-haptic displays for scientific visualization. *Computer Graphics*, *24*(4), 177-185.

Buckley, D., & Frisby, J. P. (1993). Interaction of stereo, texture and outline cues in the shape perception of three-dimensional ridges. *Vision Research*, *33*, 919-933.

Burdea, G. C. (1996). *Force and touch feedback for virtual reality*. New York: Wiley.

Campbell, A., & Coulson, G. (1996). A QoS adaptive transport system: Design, implementation, and experience. In *Proceedings of ACM Multimedia 96* (pp. 117-127). New York: ACM Press.

Caplan, P. J., MacPherson, G. M., & Tobin, P. (1985). Do sex-related differences in spatial abilities exist? A multilevel critique with new data. *American Psychologist*, *40*, 786-799.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The Information Visualizer, An Information Workspace. In *Proceedings of CHI '91* (pp. 181-188). New York: ACM.

Card, S. K., Robertson, G. G., & York, W. (1996). The WebBook and Web Forager: An information workspace for the World Wide Web. In *Proceedings of CHI '96* (pp. 111-117). New York: ACM Press.

Cleveland, W. S. (1993). *Visualizing data*. Summit, NJ, USA: Hobart Press.

Darken, R. P., & Sibert, J. L. (1996). Wayfinding strategies and behaviors in large virtual worlds. In *Proceedings of CHI '96* (pp. 142-149). New York: ACM Press.

Deering, M. (1995). HoloSketch: A virtual reality sketching/animation tool. *Transactions on Computer-Human Interaction*, *2,* 220-238.

Dennerlein, J. T., Martin, D. B., & Hasser, C. (2000). Force-feedback improves performance for steering and combined steering-targeting tasks. In *Proceedings of CHI 2000* (pp. 423-429). New York: ACM Press.

DiFranco, D. E., Beauregard, G. L., & Srinivasan, M. A. (1997). The effect of auditory cues on the haptic perception of stiffness in virtual environments. In *Proceedings of ASME Dynamic Systems and Control Division, DSC-61* (pp. 17-22). New York: ASME. The Bed: A medium for intimate communication

Douglas, S. A., & Kirkpatrick, A. E. (1999). Model and representation: The effect of visual feedback on human performance in a color picker interface. *ACM Transactions on Graphics*, *18,* 96-127.

Douglas, S. A., Kirkpatrick, A. E., & MacKenzie, I. S. (1999). Testing Pointing Device Performance and User Assessment with the ISO 9241, Part 9 Standard. In *Proceedings of CHI 99* (pp. 215-222). New York: ACM Press.

Douglas, S. A., & Mithal, A. K. (1994). The effect of reducing homing time on the speed of a finger-controlled isometric pointing device. In *Proceedings of CHI 94* (pp. 411-416). New York: ACM.

Douglas, S. A., & Mithal, A. K. (1997). *The ergonomics of computer pointing devices*. London: Springer Verlag.

Eberhardt, S., Neverov, M., West, T., & Sanders, C. (1997). Force reflection for WIMPS: A button acquisition experiment. In *Proceedings of ASME Dynamic Systems and Control Division, DSC-61* (pp. 23-28). New York: ASME.

Emerson, J. D., & Hoaglin, D. C. (1983). Stem-and-leaf displays. In D. C. Hoaglin, F. Mosteller, & J. W. Tukey (Eds.), *Understanding robust and exploratory data analysis* (pp. 7-32). New York: Wiley.

Ernst, M. O., Banks, M. S., & Bulthoff, H. H. (2000). Touch can change visual slant perception. *Nature Neuroscience, 3*, 69-73.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 47*, 381-391.

Fitzmaurice, G. W., & Buxton, W. (1997). An empirical evaluation of graspable user interfaces: Towards specialized, space-multiplexed input. In *Proceedings of CHI 97* (pp. 43-50). New York: ACM Press.

Fitzmaurice, G. W., Ishii, H., & Buxton, W. (1995). Bricks: Laying the foundations for graspable user interfaces. In *Proceedings of CHI 95* (pp. 442-449). New York: ACM.

Fogg, B. J., Cutler, L. D., Arnold, P., & Eisbach, C. (1998). HandJive: A device for interpersonal haptic entertainment. In *Proceedings of CHI 98* (pp. 57-64). New York: ACM press.

Foley, J., van Dam, A., Feiner, S., & Hughes, J. (1990). *Computer Graphics: Principles and Practice*. (2d. ed.). Reading, MA: Addison Wesley.

Gaver, W. W. (1991). Technology affordances. In *Proceedings of CHI 91* (pp. 79-84). New York: ACM.

Gescheider, G. A. (1997). *Psychophysics: The fundamentals*. (3d. ed.). Mahwah, NJ: Lawrence Erlbaum.

Gibson, J. J. (1966). *The senses considered as perceptual systems*. Boston: Houghton-Mifflin.

Gibson, J. J. (1979). *The ecological approach to visual perception*. Boston: Houghton Mifflin.

Goldstein, E. B. (1987). Spatial layout, orientation relative to the observer, and perceived projection in pictures viewed at an angle. *Journal of Experimental Psychology: Human Perception and Performance*, *13,* 256-266.

Goldstein, E. B. (1996). *Sensation and perception*. (4th ed.). New York: Brooks/Cole.

Goodwin, A. W., & Wheat, H. E. (1992). Human tactile discrimination of curvature when contact with the skin remains constant. *Experimental Brain Research*, *88,* 447-450.

Gordon, I. E., & Morison, V. (1982). The haptic perception of curvature. *Perception and Psychophysics, 31*, 446-450.

Grissom, S. B., & Perlman, G. (1995). StEP(3D): A standardized evaluation plan for three-dimensional interaction techniques. *International Journal of Human-Computer Studies*, *43,* 15-41.

Guiard, Y. (1987). Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *The Journal of Motor Behavior*, *19,* 486-517.

Haptic Technologies. (1999). PenCAT and MouseCAT. Montreal. World Wide Web page visited April 28, 1999. http://www.haptech.com/text/prod/cat.htm.

Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., & Want, R. (1998). Squeeze me, hold me, tilt me! An exploration of manipulative user interfaces. In *Proceedings of CHI 98* (pp. 17-24). New York: ACM.

Hill, J. W., & Salisbury, J. K. J. (1978). *Study to design and develop remote manipulator system* . Menlo Park, CA: Stanford Research Institute.

Hinckley, K., Pausch, R., Goble, J. C., & Kassell, N. F. (1994). Passive real-world props for neurosurgical visualization. In *Proceedings of CHI 94* (pp. 452-458). New York: ACM.

Hinckley, K., Pausch, R., Proffitt, D., & Kassell, N. F. (1998). Two-handed virtual manipulation. *Transactions on Computer-Human Interaction*, *5,* 260-302.

Hinckley, K., Pausch, R., Proffitt, D., Patten, J., & Kassell, N. (1997). Cooperative bimanual action. In *Proceedings of CHI 97* (pp. 27-34). New York: ACM.

Hix, D., Templeman, J. N., & Jacob, R. J. K. (1995). Pre-screen projection: From concept to testing of a new interaction technique. In *Proceedings of CHI 95* (pp. 226-233). New York: ACM Press.

Hoaglin, D. C. (1983). Letter values: A set of selected order statistics. In D. C. Hoaglin, F. Mosteller, & J. W. Tukey (Eds.), *Understanding robust and exploratory data analysis* (pp. 33-57). New York: Wiley.

Hoaglin, D. C. (1985). Using quantiles to study shape. In D. C. Hoaglin, F. Mosteller, & J. W. Tukey (Eds.), *Exploring data tables, trends, and shapes* (pp. 417-460). New York: Wiley.

Hoaglin, D. C., Mosteller, F., & Tukey, J. W. (Eds.). (1983). *Understanding robust and exploratory data analysis*. New York: Wiley.

Hoaglin, D. C., Mosteller, F., & Tukey, J. W. (Eds.). (1985). *Exploring data tables, trends, and shapes*. New York: Wiley.

Hubona, G. S., Wheeler, P. N., Shirah, G. W., & Brandt, M. (1999). The relative contributions of stereo, lighting, and background scenes in promoting 3D depth visualization. *Transactions on Computer-Human Interaction, 6*, 214-242.

Immersion Corporation. (1999). FEELit Mouse. San Jose, CA. World Wide Web page visited April 7, 1999. http://www.force-feedback.com/feelit/feelit.html.

International Community for Auditory Display. (2000). ICAD Home Page. . World Wide Web page visited July 16, 2000. http://www.icad.org/.

Ishii, H., & Ullmer, B. (1997). Tangible bits: Towards seamless interfaces between people, bits, and atoms. In *Proceedings of CHI 97* (pp. 234-241). New York: ACM.

Ishii, H., Wisneski, C., Orbanes, J., Chun, B., & Paradiso, J. (1999). PingPongPlus: Design of an athletic-tangible interface for computer-supported cooperative play. In *Proceedings of CHI 99* (pp. 394-401). New York: ACM Press.

Jacob, R. J. K. (1995). Eye tracking in advanced interface design. In W. Barfield & T. A. Furness (Eds.), *Virtual environments in advanced interface design* (pp. 258-288). Oxford: Oxford University Press.

Jacob, R. J. K., Deligiannidis, L., & Morrison, S. (1999). A software model and specification language for non-WIMP user interfaces. *Transactions on Computer-Human Interaction, 6*, 1-46.

Jacob, R. J. K., Sibert, L. E., McFarlane, D. C., & Mullen, M. P., Jr. (1994). Integrality and separability of input devices. *Transactions on Human Computer Interaction, 1*, 3-26.

Johnson, M. P., Wilson, A., Blumberg, B., Kline, C., & Bobick, A. (1999). Sympathetic Interfaces: Using a Plush Toy to Direct Synthetic Characters. In *Proceedings of CHI 99* (pp. 152-158). .

Kaczmarek, K. A., & Bach-y-Rita, P. (1995). Tactile displays. In W. Barfield & T. A. Furness (Eds.), *Virtual environments and advanced interface design* (pp. 349-414). New York: Oxford University Press.

Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, *12*, 391-438.

Klatzky, R. L., Lederman, S., & Reed, C. (1987). There's more to touch than meets the eye: The salience of object attributes for haptics with and without vision. *Journal of Experimental Psychology: General*, *116*, 356-369.

Klatzky, R. L., & Lederman, S. J. (1993). Toward a computational model of constraint-driven exploration and haptic object identification. *Perception*, *22*, 597-621.

Klatzky, R. L., & Lederman, S. J. (1999). The haptic glance: A route to rapid object identification and manipulation. In D. Gopher & A. Koriats (Eds.), *Attention and Performance XVII. Cognitive regulations of performance: Interaction of theory and application* (pp. 165- 196). Mahwah, NJ: Lawrence Erlbaum Associates.

Klatzky, R. L., Lederman, S. J., & Matula, D. E. (1993). Haptic exploration in the presence of vision. *Journal of Experimental Psychology: Human Perception and Performance*, *19*, 726-743.

Klatzky, R. L., Lederman, S. J., & Metzger, V. A. (1985). Identifying objects by touch: An "expert system". *Perception and Psychophysics*, *37*, 299-302.

Klatzky, R. L., Lederman, S. J., Pellegrino, J. W., Doherty, S., & McClosky, B. P. (1990). Procedures for haptic object exploration vs. manipulation. In M. A. Goodale (Ed.), *Vision and action: The control of grasping* (pp. 110-127). Norwood, NJ: Ablex.

Klatzky, R. L., Loomis, J. M., Lederman, S. J., Wake, H., & Fujita, N. (1993). Haptic identification of objects and their depictions. *Perception and Psychophysics*, *54*, 170-178.

Koenderink, J. J. (1990). *Solid Shape*. Cambridge, MA: MIT Press.

Koenderink, J. J., & van Doorn, A. J. (1992). Surface shape and curvature scales. *Image and Vision Computing*, *10*, 557-565.

Kurtenbach, G., Fitzmaurice, G., Baudel, T., & Buxton, B. (1997). The design of a GUI paradigm based on tablets, two-hands, and transparency. In *Proceedings of CHI 97* (pp. 35-42). New York: ACM.

Lakatos, S., & Marks, L. E. (1999). Haptic form perception: Relative salience of local and global features. *Perception and Psychophysics*, *61*, 895-908.

Landy, M. S., Maloney, L. T., Johnston, E. B., & Young, M. (1995). Measurement and modeling of depth cue combination: In defense of weak fusion. *Vision Research*, *35*, 389-412.

Lederman, S. J., & Klatzky, R. L. (1987). Hand movements: A window into haptic object perception. *Cognitive Psychology*, *19*, 342-368.

Lederman, S. J., & Klatzky, R. L. (1996). Action for perception: Manual exploratory movements for haptically processing objects and their features. In A. M. Wing, P. Haggard, & J. R. Flanagan (Eds.), *Hand and brain: The neurophysiology and psychology of hand movements* (pp. 431-446). New York: Academic Press.

Lederman, S. J., & Klatzky, R. L. (1999). Sensing and displaying spatially distributed forces in haptic interfaces for teleoperator and virtual environment systems. *Presence*, *8,* 86-103.

Lederman, S. J., Klatzky, R. L., & Reed, C. L. (1993). Constraints on haptic integration of spatially shared object dimensions. *Perception*, *22*, 723-743.

Leganchuk, A., Zhai, S., & Buxton, W. (1998). Manual and cognitive benefits of two-handed input: An experimental study. *Transactions on Computer-Human Interaction*, *5,* 326-359.

Loomis, J., Klatzky, R. L., & Lederman, S. J. (1991). Similarity of tactual and visual picture recognition with limited field of view. *Perception*, *20*, 167-177.

MacKenzie, C. L., Marteniuk, R. G., Dugas, C., Liske, D., & Eckmeier, B. (1987). Three-dimensional movement trajectories in Fitts' task: Implications for control. *The Quarterly Journal of Experimental Psychology*, *39A*, 629-647.

McLaughlin, J. P., & Orenstein, B. J. (1997). Haptic rendering of 3D seismic data. In *Proceedings of Second PHANToM User's Group Workshop*. Cambridge, MA: MIT AI Lab (http://www.ai.mit.edu/conferences/pug98).

Meyer, D. E., Abrams, R. A., Kornblum, S., Wright, C. E., & Smith, J. E. K. S. (1988). Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychological Review*, *95*, 340-370.

Mine, M. R., Brooks, F. P., & Sequin, C. H. (1997). Moving objects in space: Exploiting proprioception in virtual-environment interaction. In *Proceedings of SIGGRAPH 97* (pp. 19-26). New York: ACM Press.

Minsky, M. D. R. (1995). *Computational haptics: The Sandpaper system for synthesizing texture for a force-feedback display*. Unpublished Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Mithal, A. K., & Douglas, S. A. (1996). Differences in movement microstructure of the mouse and the finger-controlled joystick. In *Proceedings of CHI 96* (pp. 300-307). New York: ACM Press.

Mor, A. B. (1998). 5 DOF force feedback using the 3 DOF Phantom and a 2 DOF device. In *Proceedings of Third PHANToM User's Group Workshop*. Cambridge, MA: MIT AI Lab (http://www.ai.mit.edu/conferences/pug98).

Myers, B. A. (1995). User interface software tools. *Transactions on Computer-Human Interaction*, *2*, 64-103.

Nakajima, T., & Tezuka, H. (1994). A continuous media application supporting dynamic QOS control on real-time Mach. In *Proceedings of Multimedia 94* (pp. 289-297). New York: ACM Press.

Newell, A., & Rosenblum, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1-55). Hillsdale, NJ: Lawrence Erlbaum.

Noma, H., Miyasato, T., & Kishino, F. (1996). A palmtop display for dextrous manipulation with haptic sensation. In *Proceedings of CHI 96* (pp. 126-133). New York: ACM.

Norman, D. A. (1988). *The Psychology of Everyday Things*. New York: Basic Books.

Oakley, I., McGee, M. R., Brewster, S., & Gray, P. (2000). Putting the feel in 'look and feel'. In *Proceedings of CHI 2000* (pp. 415-422). New York: ACM Press.

Oldfield, R. C. (1971). The assessment and analysis of handedness: The Edinburgh inventory. *Neuropsychologia*, *9*, 97-113.

Olsen, D. R. J. (1992). *User interface management systems: Models and algorithms*. San Mateo, CA: Morgan Kaufmann.

Phillips, F., & Todd, J. T. (1996). Perception of local three-dimensional shape. *Journal of Experimental Psychology: Human Perception and Performance*, *22*, 930-944.

Plesniak, W., & Pappu, R. (1998). Coincident display using haptics and holographic video. In *Proceedings of CHI 98* (pp. 304-311). New York: ACM Press.

Pont, S. C., Kappers, A. M. L., & Koenderink, J. J. (1997). Haptic curvature discrimination at several regions of the hand. *Perception & Psychophysics, 59,* 1225-1240.

Pont, S. C., Kappers, A. M. L., & Koenderink, J. J. (1999). Similar mechanisms underlie curvature comparison by static and dynamic touch. *Perception & Psychophysics*, *61*, 874-894.

PsychExperiments. (1999). PsychExperiments Mental Rotation Experiment. Oxford, MS: University of Mississippi Psychology Department, http://psychexps.olemiss.edu/Exps/MentRot/startmr.htm.

Reddy, A. L. N., & Wyllie, J. (1993). Disk scheduling in a multimedia I/O system. In *Proceedings of the first ACM international conference on multimedia* (pp. 225-233). New York: ACM Press.

Reinhardt-Rutland, A. H. (1996). Remote operation: A selective review of research into visual depth perception. *Journal of General Psychology*, *123,* 237-248.

Robertson, G. G., Card, S. K., & Mackinlay, J. D. (1989). The Cognitive Coprocessor architecture for interactive user interfaces. In *Proceedings of UIST 89* (pp. 10-18). New York: ACM Press.

Rosenbaum, D. A. (1991). *Human motor control*. New York: Academic Press.

Rosenberg, L., & Brave, S. (1996). Using force feedback to enhance human performance in graphical user interfaces. In *Proceedings of CHI 96 Conference Companion* (pp. 291-292). New York: ACM.

Schmalstieg, D., Encarnacao, L. M., & Szalavari, Z. (1999). Using transparent props for interaction with the Virtual Table. In *Proceedings of the 1999 symposium on Interactive 3D graphics* (pp. 147-153). New York: ACM Press.

SensAble Technologies. (1998). *GHOST SDK programmer’s guide, Version 2* . Cambridge, MA.

SensAble Technologies. (1999). PHANToM. Cambridge, MA. World Wide Web page visited April 28, 1999. http://www.sensable.com/products.htm.

SensAble Technologies. (2000). FreeForm. Woburn, MA, http://www.sensable.com/FreeForm/.

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *IEEE Computer, 16* (Aug.), 57-69.

Sollenberger, R. L., & Milgram, P. (1993). Effects of stereoscopic and rotational displays in a three-dimensional path-tracing task. *Human Factors, 35*, 483-499.

Srinavasan, M. A., Beauregard, G. L., & Brock, D. L. (1996). The impact of visual information on the haptic perception of stiffness in virtual environments. In *Proceedings of ASME Dynamic Systems and Control Division, DSC-58* (pp. 555-559). New York: ASME.

Stanney, K. M., Mourant, R. R., & Kennedy, R. S. (1998). Human factors issues in virtual environments: A review of the literature. *Presence: Teleoperators and VIrtual Environments, 7*, 327-351.

Tan, H. Z. (1997). Identification of sphere size using the PHANToM: Towards a set of building blocks for rendering haptic environment. In *Proceedings of ASME Dynamic Systems and Control Division, DSC-Vol. 61* (pp. 197-203). New York: ASME.

Taylor, R. (1997). *The Nanomanipulator* (World Wide Web URL http://www.cs.unc.edu/research/nano/nanopage.html): University of North Carolina-Chapel Hill.

Taylor, R. M., II, Robinett, W., Chi, V. L., Brooks, F. P., Jr., Wright, W. V., Williams, R. S., & Snyder, E. J. (1993). The Nanomanipulator: A virtual-reality interface for a scanning tunneling microscope. In *Proceedings of SIGGRAPH 93* (pp. 127-134). New York: ACM.

Tukey, J. W. (1977). *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Ullmer, B., Ishii, H., & Glas, D. (1998). MediaBlocks: Physical containers, transports, and controls for online media. In *Proceedings of SIGGRAPH 98* (pp. 379-386). New York: ACM Press.

Vandenberg, S. G., & Kuse, A. R. (1978). Mental Rotations, a group test of three-dimensional spatial visualization. *Perceptual and Motor Skills, 47*, 599-604.

Voyer, D., Voyer, S., & Bryden, M. P. (1995). Magnitude of sex differences in spatial abilities: A meta-analysis and consideration of critical variables. *Psychological Bulletin*, *117*, 250-270.

Wanger, L. R., Fewerda, J. A., & Greenberg, D. P. (1992). Perceiving spatial relationships in computer-generated images. *IEEE Computer Graphics and Applications*, *12*(3), 44-58.

Ware, C., Arthur, K., & Booth, K. S. (1993). Fish tank virtual reality. In *Proceedings of CHI 93* (pp. 37-42). New York: ACM Press.

Ware, C., & Lowther, K. (1997). Selection using a one-eyed cursor in a fish tank VR environment. *Transactions on Human-Computer Interaction, 4,* 309-322.

Watson, B., Walker, N., Hodges, L. F., & Worden, A. (1997). Managing level of detail through peripheral degradation: Effects on search performance with a head-mounted display. *Transactions on Computer-Human Interaction, 4,* 323-346.

Wells, E., & Tassinary, L. G. (1998). Standardized performance trajectory as a measure of usability. In *Proceedings of Symposium on Human Interaction with Complex Systems (HICS 98)* (pp. 226-233). Los Alamitos, CA: IEEE Computer Society.

West, R. L., Ward, L. M., & Khosla, R. (2000). Constrained scaling: The effect of learned psychophysical scales on idiosyncratic response bias. *Perception & Psychophysics*, *62*, 137-151.

Wilkinson, L., & APA Task Force on Statistical Inference. (1999). Statistical methods in psychology. *American Psychologist*, *54*, 594-603.

Yau, D. K. Y., & Lam, S. S. (1996). Adaptive rate-controlled scheduling for multimedia applications. In *Proceedings of ACM Multimedia 96* (pp. 129-140). New York: ACM Press.

Zhai, S., Buxton, W., & Milgram, P. (1996). The partial-occlusion effect: Utilizing semitransparency in 3D human-computer interaction. *Transactions on Human-Computer Interaction*, *3*, 254-284.

Zhai, S., Morimoto, C., & Ihde, S. (1999). Manual And Gaze Input Cascaded (MAGIC) pointing. In *Proceedings of CHI 99* (pp. 246-253). New York: ACM Press.

Zwislocki, J. J., & Goodman, D. A. (1980). Absolute scaling of sensory magnitudes: A validation. *Perception and Psychophysics*, *28*, 28-38.