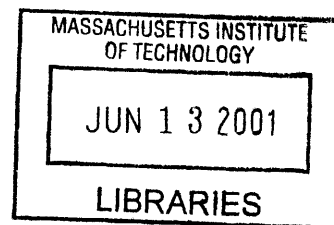


Haptic holography: an early computational plastic

Wendy J. Plesniak

S.M., MIT, 1988
B.S.E.E., CMU, 1986
B.F.A., CMU, 1984

SUBMITTED TO THE PROGRAM IN MEDIA ARTS AND SCIENCES,
SCHOOL OF ARCHITECTURE AND PLANNING,
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY, MASSACHUSETTS INSTITUTE OF TECHNOLOGY



June 2001

(C) MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 2001. ALL RIGHTS RESERVED

ROTCH

(author) Wendy J. Plesniak Program in Media Arts & Sciences

(accepted by) Stephen A. Benton E. Rudge ('48) and Nancy Allen Professor of
Media Arts & Sciences, MIT Media Laboratory

(certified by) Stephen A. Benton Graduate Officer,
Program In Media Arts & Sciences

Haptic holography: an early computational plastic

Wendy J. Plesniak

SUBMITTED TO THE PROGRAM IN MEDIA ARTS AND SCIENCES,
SCHOOL OF ARCHITECTURE AND PLANNING,
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY, MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Abstract

This dissertation introduces *haptic holography*, a combination of computational modeling and multimodal spatial display, as an early *computational plastic*. In this work, we combine various holographic displays with a force feedback device to image free-standing material surfaces with programmatically prescribed behavior.

We present three implementations, *Touch*, *Lathe*, and *Poke*, each named for the primitive functional affordance it offers. In *Touch*, we present static holographic images of simple geometry, reconstructed in front of the hologram plane (in the viewer's space), and precisely co-located with a force model of the same geometry. These images can be visually inspected and haptically explored using a hand-held interface. In *Lathe*, we again display holo-haptic images of simple geometry, this time allowing those images to be reshaped by haptic interaction in a dynamic but constrained manner. Finally in *Poke*, we present a holo-haptic image that permits arbitrary reshaping of its reconstructed surface.

As supporting technology, we offer a new technique for incrementally computing and locally updating interference-modeled holographic fringe patterns. This technique permits electronic holograms to be updated arbitrarily and interactively, marking a long-held goal in display holography. As a broader contribution, we offer a new behavior-based spatial framework, based on both perception and action, for informing the design of spatial interactive systems.


Thesis advisor: Stephen A. Benton, E. Rudge ('48) and Nancy Allen Professor of
Media Arts & Sciences, MIT Media Laboratory


Doctoral Dissertation Committee



Stephen A. Benton

E. Rudge ('48) and Nancy Allen Professor of
Media Arts & Sciences, MIT Media Laboratory


Hiroshi Ishii


Fukutake Career Development Professor of
Research in Education, MIT Media Laboratory


Bruce Blumberg


Asahi Broadcasting Corporation Career
Development Professor of Media Arts & Sciences,
MIT Media Laboratory

Contents

Acknowledgements	11
Introduction	13
Motivation	13
Inspiration	13
Criteria	14
Challenge & merits	14
Approach	14
Three demonstrations	14
Organization	15
Designing for spatial seeing, feeling, and acting	17
Design for spatial behavior	17
New and known modes of input and output	17
What about designing with emerging technologies for interactive tasks?	18
Behavior-based spatial classification to structure and inform design choices	18
Visual space perception	19
Overview	19
Classic cues to depth and layout perception: artistic and scientific inquiry	20
Rating, ranking and ranges of cue effectiveness for perception	22
Classifying perceived visual space	23
Vision for action	24
Dominant view: perception as the ultimate goal of vision	24
Changing trends: visually-guided action	24
Haptic/vestibular space perception	25
Other cues that inform perception and movement of our bodies in space	25
Visually-directed navigation of space	27
Exploring space and building cognitive maps	27
Prehension and the cues that guide it	29
The choreography of manual prehension	29
Pre-contact: sensing and perception for reach and grasp	29
Contact: sensing and expectations	30
Post-contact: Hand movements for sensing, perception and use	31
Post-contact: coupling to and controlling dynamic systems	32
Prehension prefers certain visual cues	32
Eye-hand coordination and the pleasure of seeing and feeling materials	34
From piecework to framework	34
Behavior-based classification of space	35
Overview: manipulatory, ambulatory and reconnaissance spaces	35
Manipulatory space	35
Ambulatory space	37
Reconnaissance space	39
Temporal issues matter too	40
Summary	41
Putting visual display in the manual workspace	43
Visual display in the manual workspace	43
Instrumented physical interface	43
Magic windows	45
Magic glasses and optically-combined spaces	45
Magic mirrors	47
Spatial projectors	48
Exploring holography: an ideal holo-haptic system	48

Holography: from passive to interactive display	51
Basics of display holography	51
Holography in context	51
Traditional holography	52
Holographic stereograms	53
A basis for computing holograms	55
Computational holography	55
Early days: Fourier Transform methods	55
Moving forward: Interference simulation	58
Moving faster: Computational shortcuts	62
Faster, with trade-offs: Computational stereograms	63
Realism revisited: Incrementally-computed holograms	67
Motivation	67
Algorithm basics	68
Computing elemental fringes	70
Populating a model with holo-points: randomizing initial phase or position	73
Incremental modeling	74
Indexing: determining elemental fringe contributions to the hologram	75
Incremental updating	75
Updating the holographic display	76
Discussion of algorithm performance	77
Visual appearance of images	84
Overall timing comparison to other methods	85
Toward interactive holographic display with better image quality	89
Touch: Two static haptic holograms	91
Motivation and goals	91
System design	91
Holograms	91
Haptics	92
Results	94
Two haptic holograms	94
Feeling holograms	96
Discussion	97
Spatial misregistration	97
Occlusion violations	97
Volume violations	98
Optical-haptic surface property mismatch	99
In context of the ideal holo-haptic system	100
From static to dynamic display	101
Lathe: A haptic hologram with limited update	103
Motivation and goals	103
Overview	103
Haptic simulation	104
Haptics modeling	104
Workspace Resource Manager	105
Translating model changes into hologram changes	106
Hologram simulation	107
Precomputed holograms and holovideo indexing	108
Results and discussion	110
System Lag	110
Differences in Visual and Haptic Renderings	111
Differences Between Simulation and Real Task	112

Sensory affordances and conflicts	112
General discussion	112
Poke: A haptic hologram with arbitrary update	115
Motivation and goals	115
Overview	115
Haptic simulation	116
Surface simulation	116
Surface deformation	118
Communicating changes to the holovideo module	121
Hologram updates	121
Using precomputed elemental fringes to assemble a hologram	122
Updating the hologram as holo-points change	123
Results and discussion	124
Update	124
Look and feel	125
Discussion and longer view	127
Material or just simulated material?	127
Mimesis and behavioral believability	127
Illusion and physical believability	128
Spatio-temporal calibration and user metric errors	128
Display limitations	128
Reality modeling limitations	128
Where we are	130
Our residence within the broad class of computational plastic technologies	130
What else will it take to assemble our ideal holo-haptic system?	130
Summary and contributions	133
Onward	134
References	135

Acknowledgements

I can think of no way to adequately thank Steve Benton; as a gifted scientist, he has been a true inspiration; he has provided a place for me to work and learn for many years; his vision has drawn together some of the smartest people I've ever had the benefit and pleasure of working with; and he has been notably patient and supportive when my desire to work for awhile as a non-student, my need to go off on an adventure or to spend time with family became more important than chasing advanced degrees.

Since my first awareness of Hiroshi Ishii's work, its sustained innovation and aesthetic beauty have inspired me. He seriously considers and guides the intellectual and professional development of all his students (and the occasional stray), and fosters a thriving and energetic research community. For his steady sustenance of ideas, respect, philosophical connection, and friendship, I thank him deeply.

Bruce Blumberg is a master of measuring out both interesting challenges and unwavering support in an easy and unassuming manner; this talent silently strengthens his students and friends, improves their judgement, their ideas and approaches. However comfortably his influence may proceed, it goes well-noticed and truly appreciated.

My family is the quintessence of family. Two Edwards bracket us — my nephew, Eddie, with artistic genius, empathy and humanity startlingly deep for his age, and my father, Ed, with mechanical genius, personal tenacity and resilience, always a strong anchor for our band of Plesniaks. My sisters, Jill and Jodi, are two of the strongest, smartest, and funniest women I know, the Apollo and Dionysis who keep life balanced and interesting. These four have buoyed me and made me proud more often than they know. To my mother Eileen, whose oil paintings still hang, whose piano I can still hear, and who would have been the world's most successful internet daytrader; to my grandma Elizabeth with her infinite heart and patience, whose apple pie and powdery scent were pure comfort; and to my grandpap Earl who could plant and grow anything, who always let young me taste his whiskey and puff his pipe, who had a million stories to tell (and remains the subject of a million more); their memory has encouraged me every day.

So many friends have inspired and helped me, strengthened my work, made me laugh and supported me over these years at MIT, especially Jane F. White, Melissa Yoon, Linda Peterson, Margaret Minsky, Mike Bove, Janet Cahn, Barry Aarons, Lisa Stifelman, Dina Freedman, Mike Hawley, Carlton Sparrell, Pierre St-Hilaire, Mike Klug, Bill Butera, Brygg Ullmer, Steve Smith, Nyssim Lefford, Ryder Nesbitt, Tom Nwodoh, Aaron Weber, Aditya Prabhakar, Elroy Pearson.

Rarely is someone lucky enough to have a supernatural friend like John Underkoffler; many years and shared experiences have crafted our connection and it runs vast and deep. He brings magic, art, inspiration and humor into my life; his talent is so huge and multidimensional it's scary. Tomoko Koda is a rare and precious creature whose blend of intellect and adventurousness, passion, and professional excellence make her awesome; she has been so generous with her spirit, heart, and emotional strength. Mike Halle has been a treasured mainstay for lively discussion and experimenting with new ideas and implementation strategies, photography and easily a thousand other things. Kent Diehl is my beloved ancient friend and a brilliant engineer; he has been my brother through great and terrible times since I was basically a kid. Veronica DeGuzman was my first buddy on many fantastic adventures into wildernesses, oceans, and seas; she introduced me to an entire world of wonder and thrill. Joan Spada is an adored and hugely-admired friend; razor-sharp, warm and loving, and full of passion. I voted for her for President. Lynne Berkowitz-DeAngelo, compassionate, fierce, and beautiful, has been my friend forever; she encourages me to be fearless, supports me to make it possible, and I love her with all my heart.

Finally I come to Ravikanth Pappu. It has been pure pleasure and privilege to watch many of his limitless talents strengthen and combine over the years. His sheer delight and enthusiasm to try and share new things have made him my cherished partner in research, adventures, and life. He is a river of fantastic and creative ideas, humanistic dreams and acts, witty jokes (which amuse him greatly), tremendous intelligence and competence, and cravings for delicious food. He paints and grows my life every day, and helps me to feel like anything is possible.

1. Introduction

1.1 Motivation

1.1.1 Inspiration

After growing up with frisbees, Barbie dolls, and Tupperware, it's especially hard to imagine that before Bakelite in the early 1900s, we had only natural materials to fashion into the stuff of our everyday lives. The new synthetics changed our material life fundamentally — not only in a practical way, through the near ubiquity of their use; but also in an ideological way, with the revelation that we could actually *create new materials* with qualities unlike any natural substance.

Since Bakelite, countless new synthetic materials have been invented, but most remain popularly undistinguished. Polyethylenes, acrylics, silicones, and so many other appreciably different substances are commonly piled into the same big bin — *plastic*. As such, plastic seems to be infinitely mutable. It can be silky-smooth, pliable or squishy, hard, durable, soft and fuzzy, shiny or matte, colored, patterned, textured, or clear. But once it's fashioned into a particular shape, appearance, and material consistency, those qualities largely remain as prescribed, static and fixed.

What if a plastic substrate could be instructed to change? Materials research is beginning to envision miniature, embedded means of sensing, of switching a material's visual, tactual, and bulk mechanical properties and of actuating its movement. With the eventual fusion of computational algorithms and this fabricated, malleable, programmable material, we would expect another huge practical and philosophical transformation in our lives. The advent of such a *computational plastic* would certainly offer an intriguing newness to any *thing* comprised of it; imagine clay that could be sculpted then instructed to look and feel like either powder coat or patinaed bronze — the implications of such things are still unknowable.

We can think of these computational plastics as a broad class of technologies, each a mixture of programmable behavior and both active and inert encapsulating matter. At the boundaries of their definition are two extremes: one is a completely autonomous computational substrate; the other is a non-material composite of simulation and perfectly mimetic display. In anticipation of the manufacture of the matter-part (and the means of its address and instruction), we can begin to experiment at the more accessible extreme of this alloy's spectrum. From an amalgam of computation and innovative display technologies, both visual and haptic, we might prototype a computational plastic, and begin to contemplate the properties, usefulness, and broader implication of its eventual physical instantiation.

Initially, deploying such a material in interactive computational systems would bring simulation out into the material world, and begin allowing us to apprehend and manipulate purely simulated objects almost as comfortably, skillfully, and artfully as we do real ones. It would also provide designers of interactive systems with the freedom to *embody* purely poetic meaning and idiosyncratic physics in the domain of the physically-based, thus opening up rich new territory for interaction design and artistic experimentation.

1.1.2 Criteria

A good prototype of computational plastic will have several basic properties: it must allow us to shape a surface freely, must admit programmable appearance and behavior, must be available to the senses just like a physical material, and when fashioned into an object, must offer some unique physical affordances for use. The first nontrivial goal of any prototype would be to nominally demonstrate these properties.

The next broad goal would likely be to represent something familiar, simple, and comparable to a physical instantiation. Many past inventions travelled this route to our conceptual apprehension and acceptance; the first celluloid objects imitated the appearance of tortoise-shell, amber, linens and wood, while still being marketed as something that *could* assume innumerable and unimagined forms and colors. In this context, the operative measures of a prototype's success are, generally, whether the eyes are convinced, and whether touch confirms the reality-check.

1.2 Challenge & merits

How might we fulfill these basic prescriptions using computation and display?

This question defines the broad engineering challenge of our work. We have approached it using haptic and holographic displays with co-located output, and computational modeling that prescribes its multimodal behavior. Still infant technologies themselves, haptic and holographic displays pose many engineering challenges of their own. Yet, since its earliest demonstration, holography has maintained the quiet promise of striking visual realism. And, less popularly known, haptic displays that stimulate our skin, muscle, and joint senses promise realistic and direct bodily interaction with computational models. Both of these technologies are still developing, and with them comes the potential for more vividly mimetic display.

Why is a purely non-material prototype important to build? There are several answers to this question. First and quite simply, there will always remain something new to imagine and evoke, beyond the boundaries of what can be physically assembled. In that certainty, art and simulation will long have plenty to do. Second, and practically speaking, time and physical resources are spent in fabrication; much of design and experimentation will likely continue to proceed freely and economically in a purely computational environment. Given the first two, the third answer has to do with the need to manifest simulation in our own physical space. During the last two decades, an attempt to qualify our subjective impression of being part of a "virtual" simulated environment grew into the general notion of *presence*. While it's generally agreed that this impression is desirable, it remains hard to qualify — and for *not only* this reason, it is difficult to achieve. Though no evidence squarely shows that the impression of presence affects performance, it may be useful when the understanding of our own body's relationship or interaction with a simulation is important. So while it remains hard for us to wholly tele-occupy a remote space, we may have better success making the simulation present in ours.

1.3 Approach

1.3.1 Three demonstrations

In a suite of three demonstrations described in this dissertation, we combine various holographic displays with a force feedback device, drive them with computational models that together represent the multimodal look and feel of their spatially co-

located and metrically-registered output. Through these experiments, we work progressively toward demonstrating all of the previously stated properties that a computational plastic should exhibit.

Each demonstration, *Touch*, *Lathe*, and *Poke*, is named for the primitive functional affordance it offers. In *Touch*, we present static holographic images of simple geometry, reconstructed in front of the hologram plane (in the viewer's space), and precisely co-located with a force model of the same geometry. A participant can visually inspect the holographic image through a wide angle of view while manually inspecting the force model using a hand-held interface. In *Lathe*, we again display holo-haptic images of simple geometry, this time allowing those images to be reshaped by haptic interaction in a dynamic but constrained manner. Finally in *Poke*, we present a holo-haptic image that permits arbitrary reshaping of its reconstructed surface.

Given our haptic and holographic technological underpinnings, we will describe these various implementations from a design and engineering point of view — in fact, the details and rigors of making such systems work comprise the bulk of the effort. In the strictest sense, by relying on such fledgling technologies, *Touch*, *Lathe*, and *Poke* all teeter on the edge of being convincing prototypes of computational plastic. They suffer many visual and haptic display limitations and also exhibit lag due to the enormous computational burden of hologram computation. But, by rendering three-dimensional computational models into physical space that we can literally see, touch, lathe, and poke, this work gives us a solid early glimpse of this new idea and marks a path for future and better approaches.

1.4 Organization

This dissertation begins in chapter 2 by discussing fundamental design issues for spatial interactive systems using novel means of input and display. We first review current thought about how vision (independently and in concert with other sensory modalities) informs both perception and action. We conclude the chapter by proposing a new behavior-based classification of interactive space, useful for designing systems that employ spatial display and bodily interaction. Chapter 3 offers a sampling of many and diverse research efforts related by the desire to conjoin eyes and hands at the simulation interface, and outlines the requirements for an idealized holo-haptic approach.

We continue in chapter 4 by describing important developments in the field of display holography and providing an overview of its fundamentals. We identify basic challenges to interactive display using electro-holography, and describe some existing techniques to address them. We introduce an important new technique that allows us to arbitrarily modify the image by making only local changes in the computed holographic fringe pattern (rather than completely regenerating it). Next, we motivate and describe the implementation and performance of the systems *Touch*, *Lathe*, and *Poke* in chapters 5, 6, and 7, respectively. These systems combine computational force models with the holographic techniques described in chapter 4, and address many of the visual-haptic workspace design considerations identified in chapters 2 and 3.

In chapter 8, we identify and discuss some easy and hard problems involved in producing the "perfectly mimetic" display needed for a simulation to qualify as a bona fide computational plastic. We conclude by summarizing the research contributions represented in this work, describing the overall strengths and weaknesses of our current demonstrations, and indicate directions of future research.

2.1 Design for spatial behavior

2. Designing for spatial seeing, feeling, and acting

2.1.1 New and known modes of input and output

When designing the human interface to a novel technology, we work with the design of information and interaction, physical form and affordance. We fashion them for and anchor them to the task, the cognitive and sensorimotor abilities, emotional and aesthetic sensibilities, and the knowledge base of an audience. It's a job requiring two distinct sensibilities; the first is broadly creative and innovative, the second involves research and meeting constraints. Of course, the nature of creativity, technological invention, and the evolving knowledge of human factors ensure that there are no canonical precepts for design — there is always something new to imagine, and something new to design with and for. As a result, we combine creative vision with an assembly of design principles, guidelines and methodologies¹ based on collective wisdom and experience. These principles, guidelines, and methods hope to comfortably admit each new design challenge and to efficiently guide designers through inventories of task concerns and usability considerations toward solutions that inspire, entertain us, or streamline our work.

Most of these methodologies apply to generally *known* modes of system input and output, and thus concentrate on usable design for the cognitive consumption and knowledge base of a particular audience. In practice, we filter and refract these methodologies through our own creativity and experience, through past examples, and each other, and our work is refined through evaluation and iteration. But to design with *new* kinds of sensing, display, and actuation in interactive systems, we also need to carefully revisit the fundamentals — basic human sensing, perception and performance. All too often our approach is to consult a few experts in related domains, and then just experiment and iterate until a workable system emerges. In this chapter, we describe a more comprehensive way to initially frame and weigh human factors issues when designing with new spatial interface technologies; we begin our discussion by first taking a broader view of design.

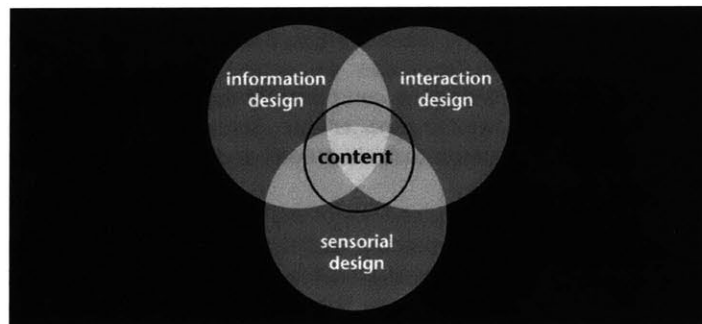


Figure 2.1 Interface design as a combination of disciplines (adapted from Schedroff [2]).

A general design space is diagrammed in Figure 2.1, borrowed from one illustrated by Nathan Shedroff². His outline shows three broad challenges in interface design: *information design*, *interaction design* and *sensorial design*. All categories require aesthetic address, but each has slightly different underpinnings. *Sensorial design* considers the modalities most appropriate for delivery of information, taking into account the kind of information to be communicated, our human sensory capabilities, and the information-bearing and emotion-inspiring capacity of different sensory channels. *Interaction design* considers commerce between human and system, often focusing on our mode and style of querying and responding to it. Importantly, interaction design also considers the ergonomic comfort and general motor capabilities of an interacting person. Finally, *information design* expresses carefully constructed and rendered system content through available display and interaction channels. It considers the knowledge and experience of its audience, their cognitive capabilities, emotional and aesthetic sensibilities.

2.1.2 What about designing with emerging technologies for interactive tasks?

These three categories are straightforward encapsulations of concerns; but such simple diagrams and definitions belie a clear and organized way to address them. For instance, no matter which set of guidelines a designer employs to streamline and focus her process, it remains a burden to consider the mountain of information housed in handbooks for human sensation, perception and performance. And to reiterate, for designers of new kinds of technological systems used by *mobile people with reaching arms, grasping hands, moving eyes and head*, it is clearly important to consider this information — especially when more comfortable, natural engagement with systems in various interactive spaces is the goal.

So, when the answers to very basic design decisions are hard to answer and require consulting domain experts and/or the literature, we need useful strategies to organize, consider, and apply what we learn. For instance, to research the benefits of incorporating stereoscopic display into a novel product, one might consult several compendiums and seemingly contradictory research publications. How should such information be weighed?

2.1.3 Behavior-based spatial classification to structure and inform design choices

In this chapter, we suggest that to weigh the usefulness of new types of visual displays, innovative controllers, or the intervening computation for spatial interactive systems, we can consider their usefulness within the context of spatial behavior, and what our perceptual, visuomotor and our haptic/motor systems have evolved to let us do.

To this end, we offer a general behavior-based classification of interactive space (*manipulatory, ambulatory, and reconnaissance space*) which provides an organization of generalized space-specific behaviors and the space-specific sensory cues that best inform them. This approach requires drawing loose analogies between the tasks we design for and basic ones we know in the physical world (which is not always reasonable, strictly speaking). However, when the analogy is good, this strategy collapses a seemingly unbounded list of human factors considerations into one holistic and (arguably) ecologically-sound package that can more clearly inform sensorial and

interaction design, and maybe influence information design too.

We arrived at this spatial classification scheme by first noticing that data from the study of visual space perception and of visually-guided action appear to fit within a common spatial framework; that what we perceive and how we act may be best guided by cues specific to a particular *region* of egocentric space (Figure 2.2). Subsequently this seems a rather obvious and common-sense arrangement of all this information anyway — and one with which our haptic senses appear to agree as well. To provide a brief (and far from comprehensive) review of supporting research, we will first discuss traditional cue-based visual space perception, haptic/vestibular space perception, and an evolving trend in vision science that relates to *action*. Next, we will describe visual and haptic sensing for action, focusing specifically on prehension within manipulatory space. Finally, we discuss our encapsulating behavior-based classification of interactive space, which fits the described data on action and perception. Notably, our discussion omits meaningful consideration of *auditory* sensing and perception of space, but future work should also include this important spatial modality.

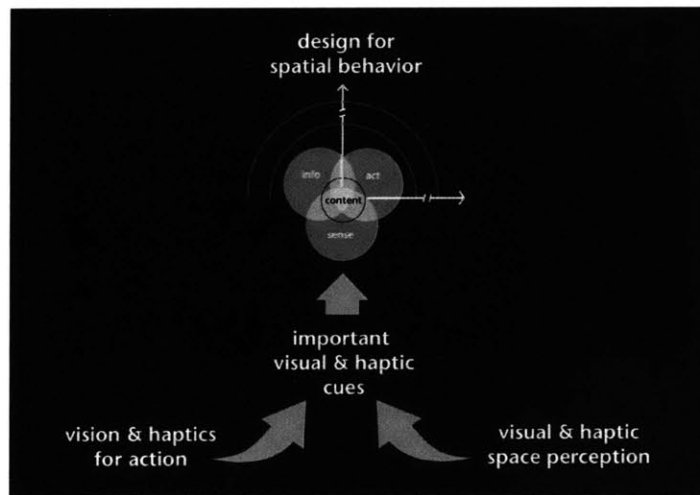


Figure 2.2 Behavior-based classification of interactive space, and its underpinnings

2.2 Visual space perception

2.2.1 Overview

Most studies of visual space perception start with the assumption that the *physical* world exists apart from the observer; and that the observer experiences a *perceptual* impression or model or representation of that world. Most of us perceive three-dimensional (3D) space, objects, entities, and phenomena populating it, and these changing over time. We perceive the *distances* and *directions* from ourselves to these recognizable things in the physical world.

The question of how we understand depth and layout in the world around us — and in pictures too — has engaged philosophers, artists, and psychologists for a long time. To assemble that understanding, consider that fundamentally (and remarkably), we must direct our various physical sensors in 3D space to optimally transduce physical energy into neural impulses, which are then processed into a meaningful and *useful* internal representation of ourselves in the environment. The fundamental issue in the study of visual space perception, as concisely stated by Maurice Hershenson in his book³ on the topic, is "...to relate perceived qualities of visual space to specific aspects of stimulation and to processes occurring in the visual system". In this section, we describe this topic in terms of the cues to depth and layout available in the spatially and temporally extended visual array.

2.2.2 Classic cues to depth and layout perception: artistic and scientific inquiry

There are many visual cues available for space perception; when in agreement (as they usually are in the physical world), they contribute to a powerful, unambiguous perception of depth and layout in a scene. However, when conflicts between them arise (as they often can in spatial displays), parsimony of perception may lead to errors in judgements of size, shape and distances.

Well before academic and scientific study would produce numerous lists of these important cues, *artists*, who deeply contemplated the representation of objects and figures in space, formulated the pictorial ones. In the fourteenth century, Giotto, whose painting was enormously influential in Western figurative arts both during and after his time, is credited with "rediscovering" the third dimension in painting. Giotto re-introduced the illusory reconstruction of physical space on a flat painterly surface and a more natural style of rendering the spatial qualities of objects, architecture, and the figure — all of which had been largely lost to Western artistic practice during the Middle Ages.

Mainly as a result of his influence, artists of the trecento began organizing form in space using various schemes that loosely imitated linear perspective. These were prescriptive, derived from careful observation, but still lacked the formal understanding of perspectival principles. (For instance, planes above eye-level were to be inclined downwards, those below eye-level were to incline upwards, and so on.) Architectural and natural spaces produced in this fashion are depicted in Giotto's beautiful *Ognissanti Madonna*, and *Il Miracolo del assetato*. Linear perspective, invented by Brunelleschi in about 1413 and published by Alberti in 1425, provided the consistent framework for these prescriptions and became a subject of great academic and artistic study.

During the Renaissance, methods of organizing and representing convincing pictorial space were formally studied in Western painting. Leonardo da Vinci's analysis of the pictorial representation of space resulted in a definition of many pictorial cues to depth and layout. Since then, of course, this list has been formalized, appended, and refined according to many diverse fields of study, and its cues have been sorted into numerous taxonomies.

A reasonably current list of these cues follows, here grouped according to whether they involve *static or moving monocular*, or *binocular* vision:

static monocular: occlusion, texture (or density) gradient, perspective, illumination and reflectance edges, shading and shadow, relative size, height in the visual field, color and transparency, aerial perspective, accommodation, and gradient of focus;

moving monocular: parallactic motion (motion parallax, motion perspective, dynamic occlusions, and perspective transformation), motion in depth (size change, expansion/contraction);

binocular: vertical edge disparity, differential perspective, differential occlusion, convergence.

Definitions and thorough descriptions of these cues are well-documented in the literature^{3,4,5,6,7}, along with various other interesting ways of associating them into groups. Probably no individual source of information about depth and layout has been studied more than vertical edge disparity, an aspect of binocular vision described in vast detail in textbooks^{7,8} and handbooks⁹.

Briefly, vertical edge disparity (along with differential perspective and differential occlusion) arises from the fact that our two eyes observe the world from slightly different locations, and often along different lines of sight too. As a result of this binocular parallax, first described by Leonardo da Vinci (Figure 2.3), the projections of an observed object have different relative positions on our retinas. Binocular disparity is the difference between corresponding points in these two projections. These retinal disparities pose an interesting problem to visual science — despite the differences in the two eyes' images, the environment and objects populating it appear *unified*.

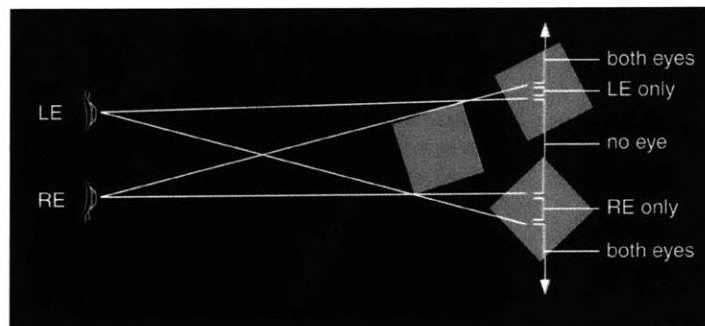


Figure 2.3 Leonardo's paradox and the problem of binocular vision

When disparities are small, they yield *stereopsis*, or the appearance of a fused, single and dimensional object. Large disparities produce *diplopia*, or the perception of

double images. Both conditions are usually present in natural viewing of the world around us, though we are usually unaware of diplopic images (like those created by the nose, for example, which always occupy a portion of the binocular visual field).

2.2.3 Rating, ranking and ranges of cue effectiveness for perception

How accurately can we perceive the distance from ourselves to a target using various of these sources of information? In fact, a profusion of studies have sought to qualify how effectively the many cues can (individually or in various combinations) inform our perception of relative and absolute depth, and how the efficacy of one compares to another^{10,11,12,13,14,15,16,17,18}. One would assume, given the metric qualities of stereopsis and vergence movements, that these would offer the most veridical information about egocentric distance. However, in the absence of other supporting cues, convergence is thought to be a relatively weak source of information; and reports of egocentric distance from stereopsis can exhibit large differences among individuals, large adaptational differences^{19,20}, and are not metrically precise even in the presence of other cues^{21,22}. Moreover, when put into conflict with occlusion, for instance, binocular cues can fail to carry the percept²³.

In light of the way their effectiveness can vary across distance and the way they interact with one another, trying to rank cues in order of their importance has proved to be a daunting task. In general, given the numerousness of possible mixed-cue comparisons, the combinatorics rapidly render this an unfeasible approach to building an understanding of layout perception. Moreover, from an ecological standpoint, a paradigm which presents only one cue, or two, with one pitted one against another in a contest for dominance, is not guaranteed to tell us how vision works in an environment rich with visual information.

However, two general empirical trends do emerge from experiments in which observers judge depth while the number of cues presented is varied. First, as more sources of information are added to a visual stimulus, an observer sees more depth; and second, an observer judges depth more accurately and consistently²⁴ — a particularly important issue for the design of interactive spatial systems.

How accurate are overall depth judgements? In general, perceived distance judgements can be described by a power function with exponent close to unity²⁵. This result includes judgements of both frontal and egocentric distance, but estimates of lateral distances within the frontal plane and orthogonal to the viewer's line of sight are more accurate, while judgements of egocentric distances are systematically *underestimated*. Thus, and curiously, cognitive reports of egocentric distance exhibit inaccuracy, with errors increasing to 50% or more for distances beyond 20 meters^{27,28}. However, visually-guided action demonstrates veridical estimates of distance (as will be discussed later in this chapter), and through the combination of many sources of information found in typical visually rich environments, we can make "reasonably accurate" (within about 10-15%) cognitive judgements about layout too, for distances within about 20 meters.

2.2.4 Classifying perceived visual space

An alternate and interesting approach to relating the catalog of cues to our perception of depth examines their effectiveness, on some common scale, throughout space. Along these lines, Nagata²⁹ determined a ratio scale for relating these fundamentally dissimilar cues and presented plots comparing their effectiveness over distance. His work was motivated by the desire to increase the perception of depth in 2D images, and his data was used to determine which cues operate most powerfully within picture-viewing distances. Nevertheless, his graphical data hints that different sets of cues may operate effectively within different ranges of distance from the observer.

Cutting and Vishton²⁴ provided a similar but much expanded common-scale analysis of nine sources of information for scene layout. Like Nagata's, their graphs of cue effectiveness over distance suggested subspaces in which different groups of cues optimally inform perception. However, unlike Nagata's study, theirs formalized this observation and folded it into a framework for considering how cues contribute to the perception of layout. They identified and described three circular egocentric regions: *personal space*, *action space*, and *vista space*, in which different sets of the various cues are more useful for communicating information about layout.

Their framework (Figure 2.4) defines *personal space* as a zone immediately surrounding an observer's head, which extends slightly beyond their outstretched arm. Within this space, their data indicates that the six most effective sources of information (ranked in diminishing order) are: occlusion, retinal disparity, motion, relative size, convergence, and accommodation. *Action space* extends from personal space to roughly 30 meters from the observer; within this space, an observer can talk, walk, throw and catch, etc. The six sources of information found most relevant in this space are: occlusion, height in the visual field, relative size, motion, binocular disparity, and density gradient. *Vista space* stretches beyond action space, and the five sources of information that operate most effectively at these great distances are given as: occlusion, relative size, height in the visual field, density gradient, and aerial perspective. The boundaries between these spatial regions are not hard or metrically specific; rather, they grade into one another.

This taxonomy of space provides an accommodating framework for the great number of cues and their diverse encapsulation of depth information. It relates visual sensing and perception directly to distinct subspaces in which humans function differently. The work implies but does not explicitly formulate a relationship between visuomotor behavior within a space and the collection of cues that inform a person's action within.

Our behavior-based spatial taxonomy is particularly inspired by this work, but in contrast, it also includes relationships between specific sensory cues and *spatial behaviors* within each egocentric subspace. To render a sense of the relationship between spatial cues and behavior, we begin a discussion on the commerce between sensing and action — sometimes without perceptual mediation — in the following sections.

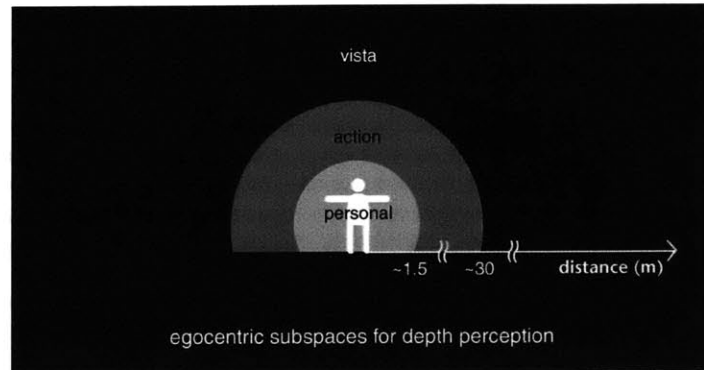


Figure 2.4 Visual space: regions defined by the use of various cues

2.3 Vision for action

2.3.1 Dominant view: perception as the ultimate goal of vision

As implied in the previous section, it has long been generally considered that the function of vision is to help assemble an internal representation of the world that, in turn, informs both our recognition of objects and entities, and any action we take upon them. In this common view, visual processing first results in a unified *perception* that then serves both the *behavioral* and *cognitive* life of an individual. Its underlying assumptions, implicit in much (but not all) formal thought about vision and visual system organization, consist of the primacy of "the percept", and the functional dichotomy of the processes it informs.

Various hypotheses have emerged over the last decades that embed this view of the visual system into functional descriptions of the neural projections from the retina and through the central nervous system. In particular, several "two visual systems" hypotheses^{30,31,32} have been proposed that distinguish brain mechanisms of vision in support of this notion of perception and its dual functions. These hypotheses reinforced the views that vision is a modular process rather than a monolithic one, and that *multiple* visual systems exist in the primate brain.

Probably the most well-known and influential of these conceptions is the "what" *versus* "where" distinction between two visual systems, in which one (what) pathway processes the identity of objects, and another (where) pathway localizes them in space^{31,32}. Generally in the "what" vs. "where" paradigm, two pathways are considered to address different aspects of the stimulus array in order to ascertain answers to their respective questions; the outcome of the various processing is a unified perception of space and the things that populate it.

2.3.2 Changing trends: visually-guided action

However, there have also been implications that perception-independent systems for visuomotor processing are present among brain mechanisms of vision. For example, the classic paper "What the Frog's Eye Tells the Frog's Brain"³³ describes cells in the frog retina involved in processing motion and helping to direct a frog's tongue fly-

ward. And, from the study of eye movements, it has long been understood that the superior colliculus performs its own motion processing and is involved in controlling the direction of gaze³⁴. Of course, J.J. Gibson's^{5,35,36} enduring ecological view maintains that the expanded spatial and temporal stimulus array contains enough information about the observer and the environment to directly inform her action.

Along these lines, influential recent work such as that by Goodale and Milner^{37,38}, argues strongly against the preeminence of perception, and asserts the biological importance of adaptive visuomotor control as a direct outcome of visual processing. In their "two visual systems" hypothesis, visual perception and visual control of action depend on functionally and neurally independent systems. Here, perception is more narrowly defined, and excludes the processing required for both reflexive phenomena and the online control of skilled actions. Thus, motor output and perception are considered two separate functions of vision — the former being embraced as the likely reason vision evolved in the first place.

This argument is strengthened by their compelling findings on the Titchner illusion³⁹ which certainly fools perception but has no effect on motor behavior. But the strict dichotomy of their model has recently been challenged by both neuroimaging and pathological studies. These data indicate that the dorsal pathway, attributed exclusively to visuomotor transformation by Goodale and Milner, can also be involved in purely conscious perceptual activities^{40,41,42}. Thus, such a rigid partitioning of brain resources between action and perception may be an oversimplification; in practice, these resources probably often collaborate.

Nevertheless, fresh emphasis on the biological demands of the behaving organism is turning more attention toward research on vision for action. Favoring its investigation is the fact that technology now makes it easier to measure the kinematics of complicated movement (like reaching and grasping). Thus, experimental methodologies that examine these kinds of motor output in direct relation to visual stimuli are simpler to incorporate into a study. Such methodologies are qualitatively distinct from those requiring verbal/cognitive report or *ad hoc* motor responses to visual discrimination tasks, both of which are better suited to the investigation of perceptual phenomena.

Research in the field of motor control has often employed such careful measurements to study eye, head, and arm movements during reach to grasp movements made toward visual targets. Though much of this work is aimed at understanding the agencies involved in planning, regulating, and controlling such movements, it also acknowledges the intimate relationship between sensing and behavior. In the following sections, we touch upon some ways we use sensing to both perceive and to control our body's position, orientation and behavior in space — from our most basic ability to maintain stable posture, to moving around, and to performing complicated manipulatory actions.

2.4 Haptic/vestibular space perception

2.4.1 Other cues that inform perception and movement of our bodies in space

Of course, our senses do not usually act alone or independently. Gibson⁵ emphasized

that spatial behavior, retinal motion stimulation, and *muscular/tactile stimulation* were intimately linked with one another. In this brief section, we use the term *haptic* broadly, to denote skin, muscle and joint senses; these senses report body position and orientation, movement and contact with the environment and, along with vestibular sensing, help to register and stabilize changes in the visual array due to environmental or self-motion.

The adjustment of our own posture, something we are mostly unaware of, is necessary for stabilizing many body movements we make. These adjustments serve three behavioral functions⁴³. They support the head and body against gravity, they balance our center of mass over a base of support, and they stabilize the supporting parts of the body while others move around. Changes in our body posture are made through reflex actions, and both anticipatory mechanisms and feedback responses. The former align the head with gravity and body position. The latter two are recruited by sensory mechanisms, shaped in part and refined by experience, practice, and learning. Anticipatory mechanisms predict disturbances in the environment and invoke pre-programmed postural changes (both before and during voluntary movement) that ensure stability. Feedback responses to adjust posture are extremely rapid, and have a relatively stereotyped choreography.

Haptic, visual, and vestibular senses recruit these postural adjustments; shearing forces on the skin of our feet interfacing with the ground, proprioceptive feedback from the ankles, static and changing position of our limbs, head orientation relative to our bodies and the gravity vector, and visual sensing of movement in extra personal space are all important to trigger reflexes and anticipatory and compensatory responses. Spatial processing of haptic, visual, and vestibular information together also informs *voluntary* movement through space, like reaching to grasp and rhythmic motor patterns like walking and running. Such sensory information informs both feed-forward and feedback control of complicated motor actions; the former being essential for rapid movements and the latter for stable regulation of slower movements⁴⁴.

When sensory information concerning our body's movement and orientation is surprising or atypical in association with our expectations or intentions, *motion sickness* (a collective term for symptoms including nausea, headache, drowsiness, and others) can result. Usually, prolonged exposure to either real or apparent motion that causes sensory conflict⁴⁵ is required to induce feelings of motion sickness. Flight simulators, entertainment rides, and wide-screen movie theaters often induce motion sickness by affording compelling visual cues to motion without the patterns of vestibular or haptic sensations that typically accompany body acceleration and changes in orientation. Likewise troubling are slow, rolling boat rides or back-seat car travel on winding roads, providing vestibular and haptic cues to motion without corresponding visual ones. Motion sickness symptoms may be eased or avoided if we are able to plan on or anticipate motion cues; for instance, aircraft pilots are exposed to but usually unaffected by the same sensory conflicts that inspire air-sickness among passengers. Of course, in the absence of sensory conflict, haptic, visual, and vestibular cues together afford our efficient and accurate movement through space, around obstacles,

over uneven terrain; and they also guide complicated spatial interactions with static and dynamic objects, systems, and entities.

2.4.2 Visually-directed navigation of space

To prepare for movement, visual spatial information about objects in the external world must be correlated with haptic and vestibular information about the position of our head, trunk, and limbs. In the absence of feedback from one particular modality, other available modalities can often compensate well enough to accomplish many actions. For example, in a familiar environment, or after initial visual inspection in a stable and unchanging one, we can apprehend a previously-glimpsed object even without the aid of vision during movement. In fact, a number of studies show that the extensity of intersensory calibration even allows an observer to walk to a spatial location fairly accurately without vision, after just an initial glimpse.

The investigation of visually-directed walking (walking to an initially-sited target without visual feedback) has repeatedly shown reliable responses^{27,46} which do not exhibit the same depth foreshortening evident in perceptual judgements mentioned in the previous section. In this general experimental paradigm, an observer views a spatial extent, closes her eyes and walks toward one endpoint, then proceeds to the next, stopping briefly to indicate each as accurately as possible. Observers reach the first endpoint accurately, though subsequent segments of the path are typically overwalked — however, there is no systematic difference between judgements of traversed frontal and egocentric distances. There is active and interesting debate about why cognitive responses differ from motor responses: does an observer "compute" an undistorted extent based on the perceived egocentric distances to its endpoints?; do we calibrate motor output to correct for underlying perceptual distortions?; does motor output proceed independently of perception entirely?

In any event, this is not to say that haptic perception of space cannot be biased. Okabe *et al.*⁴⁷ found that estimated traversed distance dilated with increased muscular effort; making a route more arduous by increasing travel speed or physical resistance makes it feel longer. Lederman *et al.*⁴⁸ investigated length distortion in pathways explored (without vision) both by hand and by walking, and found that movement duration biased length judgements in the former but not the latter. In practice, of course, limiting vision, movement or haptic exploration in the experimental methodology does not reflect the most typical way we interact with the environment. People estimate layout and distances more precisely when using *both* visuomotor and haptic/motor means to explore, than when exploring without vision or by passive looking and reporting. It appears that spatial behavior has access to fairly accurate information regarding the location and orientation of self, and about distances and directions to other landmarks in the environment, even under impoverished viewing conditions.

2.4.3 Exploring space and building cognitive maps

Space has been partitioned according to criteria^{48,49,50} other than the operational range of visual cues, including the generalized *exploratory* strategies that particular subspaces admit and which perceptual and sensorimotor systems are involved. For

instance, a *large-scale space* is explored from multiple vantage points using eye and/or head movements and whole-body locomotion, involves integration of information over time, and encloses an observer. A *small-scale space* may also be explored from multiple vantage points using eye and/or head movements, possibly over time; however it does *not* enclose an observer and therefore does not admit whole-body locomotion.

Given that two challenges in a large-scale space are *navigation* and *situational awareness* within the environment, the particular information that supports our cognitive mapping process is of special interest. Downs and Stea⁵¹ introduced the term "cognitive map" to indicate the internal representation we construct of a particular the space. According to the Landmark-Route-Survey (LRS)⁵² model of spatial knowledge acquisition, processes essential to constructing spatial knowledge can be described as having three stages: the identification of landmarks; forming a procedural (sequential and alternate) route knowledge by actually traveling between landmarks; and encapsulating those within an overall spatial survey which collectively situates and scales landmarks and routes. The importance of landmarks (like architectural structures in urban areas) for both wayfinding and organizing large-scale layout has been underscored in recent work⁵³. It would appear that to form a robust cognitive map of a particular real or virtual space, we benefit from visual survey, *perhaps* a wide visual field-of-view and active locomotion⁵⁴, and making map-type inferences or references.

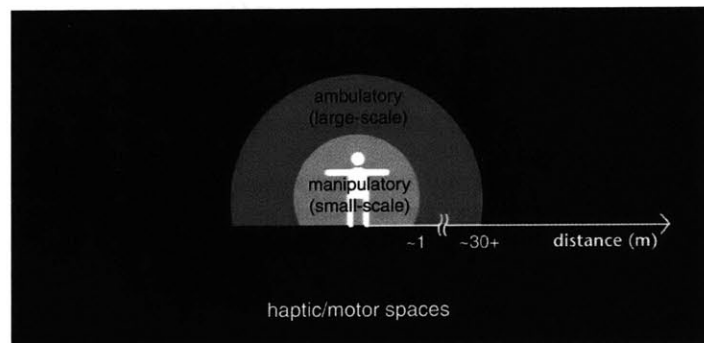


Figure 2.5 Movement space: classification based on systems used for sensing and exploring.

In their work, Lederman *et al.*⁴⁸ emphasize active locomotion and the haptic sensing involved in exploring spatial environments, referring to large-scale space as *ambulatory* (explored with toes/feet/legs), small-scale space as *manipulatory* (explored with fingers/hands/arms), and the entire spatial theater as *movement space* (Figure 2.5). These classifications of space are especially attractive because they reflect a more active relationship with the environment around us (and in that spirit we borrow from it for our behavior-based space taxonomy). Yet, because these frameworks are constructed to research *perceptual phenomena* and *cognition*, they primarily view action as the deployment of sensing, and both as the means to spatial perception.

Of course, it is true that action is almost always required to bring about sensing of our environment. But sensing informs action too, and as has been previously discussed, the importance of spatial action (sometimes perhaps proceeding without perceptual influence) is important to consider as well. In the next section we take a brief but closer look at the mechanics of a specific important behavior — manual prehension — as a perfect example of this bi-directional link between sensing and action, and as affected by cognitive factors too.

2.5 Prehension and the cues that guide it

2.5.1 The choreography of manual prehension

We use the term *prehension* to include the transport of the hand to an object through reaching, and the application of functionally effective forces by the hand to an object through grasp and manipulation. In the immediate space around our body, vision and haptic senses guide our reaching, grasping, and manipulation of objects in the environment. These prehensile movements involve specific motor systems, have a well-known choreography, and we make them countless times every day.

We begin a reach by first directing eyes toward the target, then rotating the head toward it, and lastly launching the arm. The act of *transporting* the hand/arm to a location can be broken into two distinct stages. The first is a ballistic stage, characterized by high velocity, low-visual-feedback movement, and the second is a more highly controlled deceleration phase in which visual sensing is usually employed⁵⁵. During transport, the hand is opened slowly to a peak aperture (reached at about 70 percent of travel through the movement distance) which then closes as the hand nears the object. By the end of the reach, the wrist is oriented comfortably with respect to the object and a grip aperture is shaped to sufficiently enclose it^{56,57}. The velocity profile of the whole movement is stereotypically smooth and bell-shaped.

2.5.2 Pre-contact: sensing and perception for reach and grasp

Many factors influence manual prehension (Figure 2.6). Before and during a reach to grasp, vision informs about target location, initial hand location, and the state of both target and hand during the deceleration phase of movement. Though vision is used throughout movement, it appears to be most crucial prior to the reach, and during the deceleration phase of transport⁵⁸. During open-loop reaching, (when visual feedback is absent) longer movement times during deceleration is a common finding; there are conflicting reports as to whether movement time increases during the acceleration phase of reach. Whether and/or how grasp is affected during open-loop reaching is also a matter in question^{58,60,61}. In any case, reach is most efficient and accurate when the eyes are trained on its target; the kinematics of movement are affected when our view is impoverished, though of course, we can still accomplish the task⁶².

We can also use memory to apprehend *remembered* objects — those which have been seen and/or apprehended in the past, but are now absent. Remembered objects, of course, can offer no visual or haptic feedback. Goodale *et al.*⁶³ showed that both reach to and grasp of remembered objects (even for objects removed as little as 2 seconds prior to movement) were "pantomimed", meaning that the overall movements were slower, and the path followed by the hand was more highly curved above the surface

on which the object sat. Additionally, the maximum grip aperture was smaller than normal and fixed, rather than reaching a wider opening and narrowing before contact. Pantomimed grasping has also been observed in virtual reality systems when haptic feedback fails to indicate contact between the hand and object⁶⁴. Thus, impaired visual and/or haptic sensing of the target object may lead to atypical reach and grasp, but again, the task can still be accomplished.

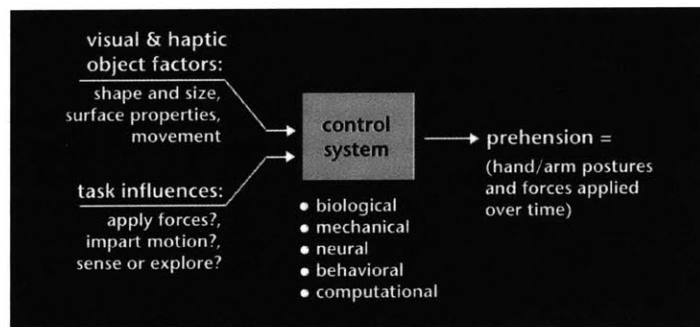


Figure 2.6 Manual prehension (adapted from MacKenzie and Iberall [59]).

2.5.3 Contact: sensing and expectations

Once contact is made with an object, subsequent *grasp and manipulation* depend on sensory information supplied by both vision and haptics; cutaneous and kinesthetic sensing provide information about contact, joint positions, net forces and object slip^{62,65}. Visual information is important when precise spatial positioning, of object part or between object and hand, is required. Haptic feedback is essential to controlling contact conditions, and stable control upon and during contact is crucial for effectively grasping and manipulating an object. Being able to sense force, slip, and torques at a telemanipulator, for instance, helps an operator to avoid damaging or mishandling objects with the manipulator and to complete a task much more quickly than when no force display is available⁶⁶. Even with force but no tactile feedback, the applied grasping forces are still likely to be greater than needed to prevent object slippage⁶⁷.

Our expectations, or what we know about an object, can affect both reach and grasp. When we encounter an unrecognized object, visual information is used (perhaps to access knowledge of how object appearance is associated with weight, slipperiness, brittleness, fragility, etc.) to determine appropriate manipulative forces to be applied upon first contact^{68,69}. In particular, the deceleration phase of reach appears to be affected by the force precision and spatial accuracy required at contact^{70,71}. An expectation that larger objects will require greater grip and load forces seems to strongly influence initially applied grip and lift. Thus, ascribing unusual visual properties to known objects, or assigning familiar visual properties to unknown objects may influence the way they are initially apprehended.

2.5.4 Post-contact: Hand movements for sensing, perception and use

Haptic cues might also invite particular manual actions. Once an object is in-hand, the material or bulk properties of the object may influence the way that it is manipulated. Lederman and Klatzky⁷² have classified many haptic object properties into three categories:

substance-related: texture, hardness, temperature, weight

structure-related: weight, volume, global shape, exact shape

functional: part motion, specific function

They have further identified eight distinct patterns of whole-hand choreography employed to apprehend these properties; they call these stereotypical patterns of movement *exploratory procedures (EP's)*:

Lateral motion is used to sense texture by making sideways movement between skin and interior (non-edge) parts of object surface;

Pressure, produced by applying torque or pressure against one part of an object while stabilizing another is used to sense hardness;

Static contact, in which the skin passively and minimally rests against the object surface, is used to sense temperature;

Unsupported holding, in which the object is hefted away from a supporting surface, is used to determine weight;

Enclosure, in which the hand molds itself to the envelope of the object to contact as much of its surface as possible, is used to sense global shape and object volume;

Contour following, a smooth and nonrepetitive movement along object features, is used to apprehend the exact shape of an object;

Part motion testing is applied when a moving part is discovered, in order to understand its movement;

And **function testing** is applied when testing the specific mechanical functionality of an object.

Interestingly, each EP appears to be optimal for sensing its associated object property or properties, and their use is so reliable that experimenters were able to accurately guess the object property being sensed by observing a subject's movement patterns. Whether or not apprehension of material or bulk object properties is part of a specific task, ascribing particular *haptic cues* to an object or an instrument certainly could influence haptic/motor behavior. For instance, if considerable contact between object and skin surface is desired for good electrical grounding, it may be wise to build an

object with comfortable heft, or one whose global shape invites enclosure by the hand. Or if intermittent, high-force contact is desired, a surface with interesting compliance might be useful, and so on.

2.5.5 Post-contact: coupling to and controlling dynamic systems

Sometimes an interface is not as mechanically "simple" as an object that we sense, transport, or manipulate; often we physically couple ourselves to systems having quite complicated physical dynamics. In this case, it becomes important to consider how we sense system dynamics and adjust our own motor output to achieve stable control. One approach to these issues is to apply a control theoretic framework to the modeling of control and coordination in humans. For instance, much of the theory and analytical modeling of the manual control of vehicles was developed in the 1960's, and one important result is the Crossover Model⁷³ proposed by McRuer *et al.* This model describes the human in the system loop as extremely adaptive; for many systems, a person actually acts to *combine* with the controlled element to produce a simple and invariant closed-loop transfer function for a variety of input. For operation near the crossover frequency ω_c for a given system, the combined human and system transfer function has the following basic form

$$H(s)G(s) = \frac{\omega_c}{s} \exp(-\tau_e s)$$

where $-\tau_e$ is the delay due to human reaction time, $H(s)$ is the transfer function of the interacting human and $G(s)$ is that of the controlled system. McRuer's Crossover Model underscores how plastic our sensorimotor systems are. That we adapt to the dynamics of a physical system is not altogether surprising; that, upon closing a feedback loop, we adjust our own control dynamics to those of various systems in a predictable way is both surprising and useful. A basic diagram depicting both human and controlled element in a closed loop system is shown in Figure 2.7.

Designing the characteristics of performance feedback a person receives becomes very important in this scenario: in short, the bandwidth of display and control information needs to match our capabilities for sensing and movement. In prehensile tasks, we should consider both visual and haptic feedback. Visual feedback is perhaps the easiest to provide in computational interactive systems, though questions of how image quality, lag and low frame rate affect performance are still under investigation. Interface requirements for manual input and tactile and haptic feedback are also important to consider for the stable control of dynamic systems.

2.5.6 Prehension prefers certain visual cues

Which visual cues does the visuomotor system prefer? A recent stream of studies has investigated the usefulness of visual cues for the planning and online control of prehension. In general, this work indicates that the same set of visual cues to depth and layout found to operate most efficiently within Cutting and Vishton's²⁴ *personal space* are used to plan and control prehension^{74,75,76,77,79,81,78,61} in a region corresponding to the *manipulatory space* of Lederman *et al.*⁴⁸. These experiments have

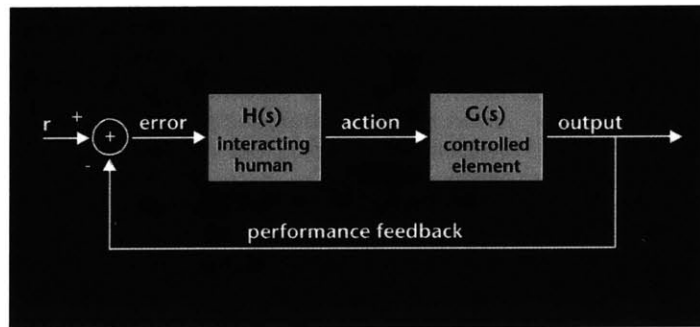


Figure 2.7 Combined person-machine system diagram

empirically assigned primacy to binocular cues for visually-guided prehension.

For example, in an early set of such studies, Servos *et al.*^{74,75,76} examined the kinematics of reaching and grasping movements under monocular and binocular visual feedback, while varying the size and distance of target objects. Their experimental setup was also replete with monocular depth cues; thus any kinematic difference between the monocular and binocular conditions could be attributed only to the presence or absence of binocular depth information. They found that movements made under monocular viewing conditions showed longer movement times, lower peak velocities, longer deceleration phases and smaller grip apertures than the movements made under binocular conditions. Their data implies that people underestimate a near object's size and distance under monocular viewing, and that binocular viewing contributes to better planning and on-line control of prehensile movements.

Other recent studies confirm the importance of binocular vision during reaching and grasping^{77,78}; a notable difference between binocular and monocular reaches is manifest in the number of on-line *corrections* in trajectory and grip size, notably at the closing phases of the movement. The greater number of on-line corrections (observed as additional peaks and plateaus in the movement velocity and grip aperture profiles) subjects made under monocular viewing conditions revealed a striking difference from binocular reaches, probably due to errors in estimating the target's distance and size.

Obviously, people without binocular vision can still reliably locate and apprehend objects in space using monocular information, so other cues must provide sufficient information for the visuomotor system. In a recent experiment by Marotta *et al.*⁷⁹, the role of head movements in the control of manual prehension was investigated. In this study, subjects viewed stimuli under either binocular or monocular conditions, with their heads restrained or free to move, and reached to grasp a target sphere positioned some distance away. The results indicate that in the absence of binocular vision, monocular information derived from motion parallax can suffice to calibrate manual prehension. A study of subjects with only one eye found that larger and faster head movements were employed during their reach as compared to binocular

subjects wearing an eye patch, suggesting behavioral adaptation to monocular vision in favor of optimal performance⁸⁰. In another study investigating monocular cues, Marotta and Goodale⁸¹ found that angular elevation can also inform prehension in the absence of binocular and motion parallax cues. These studies imply that binocular vision, motion parallax, and monocular cues contribute to the performance of reaching and grasping, but that people clearly perform best when using binocular vision.

2.5.7 Eye-hand coordination and the pleasure of seeing and feeling materials

Internal motivation or external sensory stimulation directs our eyes, and eyes direct hands, whose sensing, movement and manipulations appear to *reciprocally* direct and stabilize our eye movements. When both hand and eye are employed in tracking for instance, the improvements in dynamic characteristics of smooth pursuit eye movements are notable; the smooth pursuit maximum velocity may increase by 100%, and the delay in eye-to-target movement may decrease from 150ms to 30ms⁸². This means we might visually "lock on" to a target more quickly, and maintain it in foveal vision by involving hands in the task.

This bi-directional coordination is fundamental to our skilled use of instruments and tools, to image-making and the working of materials. The strength of eye-hand coordination is embodied or implied in the proliferation of stuff that populates our man-made world: its artifacts (our crafted, constructed, and manufactured objects) and its instruments and appliances (our tools). As Malcom McCullough comments in his book on visuo-manual skill in craft⁸³, "Because this kind of coordination satisfies, we pursue it in play, too: sports, musical performance, building projects and traditional handicrafts give pleasure through coordination. Reflection finds harmony in the steady flow of hand, eye, tool and material."

Artistic approaches to shaping raw materials into form can also reflect a kind of profound mental *empathy* with an object. The sculptor Henri Moore describes⁸⁴ the deep visual, proprioceptive and kinesthetic involvement with the sculpted form as something an artist must experience: "...He gets the solid shape, as it were, inside his head — he thinks of it, whatever its size, as if he were holding it completely enclosed in the hollow of his hand. He mentally visualizes a complex form *from all round itself*; he knows while he looks at one side what the other side is like; he identifies himself with its center of gravity, its mass, its weight; he realizes its volume, as the space that the shape displaces in the air." Thus, to one trained or naturally predisposed to develop them, *seeing* and *feeling* can embody much more than just the sensory consumption of physical cues for perception and action; they can be channels for contentment and satisfaction, intellectual appreciation and even emotional exhilaration.

2.5.8 From piecework to framework

From this brief accounting of three fundamental topics — visual and haptic perception of space, visual information for action, and the stereotypical mechanics of prehension fed by both visual and haptic cues and cognitive factors — we can begin to sketch how we look, see, move to, apprehend and manipulate objects. Drawing from this

information and previously mentioned classifications of visual and exploratory space, we assemble a behavior-based classification of space to guide the design of interactive systems. In the following sections, we describe the relevant subspaces and the generalized behaviors occurring within them. Based on the foregoing survey of studies, we list the visual and haptic cues shown either to operate most potently in each subspace or to inform particular behaviors there. Other important senses (audition, olfaction, *etc.*) operating within these subspaces are conspicuously absent, but should be included in future discussions.

2.6 Behavior-based classification of space

2.6.1 Overview: manipulatory, ambulatory and reconnaissance spaces

Borrowing from the work of Cutting and Vishton²⁴ described earlier, we also coarsely divide our egocentric environment into three subspaces — here called *manipulatory*, *ambulatory*, and *reconnaissance space* — characterized by the general kinds of spatial behavior likely to occur within them and their sensorimotor underpinnings. And borrowing from the work of Lederman, Klatzy and others^{48,49}, we link the approximate physical dimensions of each subspace to our sensory-motor apparatus and their constituent capabilities:

In *manipulatory* space, we might inspect, grasp, or manipulate objects, tools, ourselves or other entities. We most often use our eyes, hands and possibly feet to sense; we act with the latter two; we move our head/neck, arm/hand, leg/feet, and trunk systems to direct sensing and action. This space extends from our body surface to an egocentric distance of roughly 1.5 meters. The character of manipulatory space might be considered as intimate, apprehensible and productive.

In *ambulatory* space we might throw something at a target, track, approach, or run away from a location or entity, or be social. Here, we use whole-body locomotion to move, our visual, cutaneous, and kinesthetic systems to sense. This space extends out from manipulatory space to an egocentric distance of roughly 30 meters. The character of ambulatory space is accessible, remote but relevant to the body.

In *reconnaissance* space, we might gather information to inform future plans and ensure readiness, or, in the complete absence of distractions or anything to prepare for, we might peacefully enjoy the sights and sounds. In this space which stretches outward from ambulatory space, we use vision as both a focal and ambient sense, and our head/neck and general body posture to orient ourselves to the view (Figure 2.8). The character of reconnaissance space is vast and can invite inspiration and contemplation.

We can begin to associate (and in some cases provide relative rankings of) visual and haptic cues with each subspace, based on their relative effectiveness for spatial perception and the spatial behavior they afford or influence.

2.6.2 Manipulatory space

Based on the work of Nagata²⁹, Cutting and Vishton²⁴, Servos and Goodale⁷⁵ and

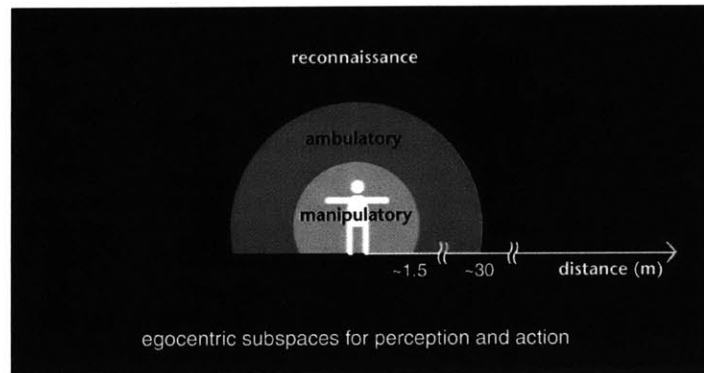


Figure 2.8 Behavior-based classification of interactive space

others, the most important visual cues to depth and layout in manipulatory space are occlusion, binocular disparity, motion parallax, relative size, the yoked mechanisms of convergence, accommodation and pupil diameter, and perhaps differential and dynamic occlusions too. These cues appear to *best* inform visually-guided prehension and our perception of depth ordering and egocentric distance. Here, attention is focused and vision is mostly foveal. Visual size, the appearance of object motion, and its surface and bulk properties will also influence our action within this space.

In addition, kinesthetic, proprioceptive and cutaneous cues inform and stabilize the diverse, complex, and precise movements we make within manipulatory space, and our interactions with near objects and entities. Perhaps the most crucial haptic cue we sense for interaction is contact force; other important interaction forces are friction and cutaneous shear, and the forces generated by dynamic physical systems we couple our bodies to. Important haptic cues about object bulk and surface properties include texture, hardness, temperature, weight, volume, global and exact shape, part motion, and specific object and part function. Together, these visual and haptic cues permit us to efficiently and precisely inspect, apprehend, and manipulate objects, and also to interact with or control many kinds of physical systems in manipulatory space. These qualifications of manipulatory space are summarized in Table 1.

In our own work, the kinds of spatial interaction we are most interested in are those operating within this domain of visual resolution and manual dexterity, including examples like innovative musical controllers, surgical simulation, teleoperation, and the tangible manipulation of symbolic information. The most challenging aspects of designing in manipulatory space are to provide: display and control with enough resolution and bandwidth; information with enough multisensory richness; comfortable controllers that can admit our motor dexterity; and imperceptible temporal lag and latencies.

For this space we have a variety of high-quality 2D visual displays, and input devices

which are of a scale appropriate to the space. 3D displays are available, yet most are still cumbersome to deal with in some way, and they offer variable image quality compared to their 2D cousins. Though we know that binocular cues are important in visuomotor tasks, we must still weigh their practical benefit against diminished image quality and a viewer's discomfort. Providing programmable tactual feedback is still a substantial technological challenge, though a variety of haptic displays can provide force or vibrational cueing to the fingers and hands. By far, the best haptic display fidelity we can currently provide is to use instrumented physical objects as interface to the hand. Many examples of such work have been demonstrated^{85,86,87,88,89,90} including the beautiful "Luminous Room"⁹¹ project by Underkoffler *et al.* In this work, a camera optically tracks coded objects while they are manipulated by an interacting person, and a projector, nearly co-located with the camera, projects responsive visual information onto the objects and their supporting surface. Here, comfortable, natural and direct manipulation of objects is coupled to display of information related to them (either very literally, or in a more symbolic way). Other examples of manipulatory interactive systems, inspiring and related to our own work will be the topic of discussion in the next chapter.

egocentric subspace	egocentric distance	primary ranked visual layout cues	important haptic senses
<i>manipulatory space:</i>	0-1.5m around head/shoulders	occlusion, binocular disparity, motion parallax, relative size, convergence and accommodation	cutaneous proprioception and kinesthetic

Table 1: Manipulatory space: minimal provisions

2.6.3 Ambulatory space

In ambulatory space, like that of a village, ship, or a baseball field for instance, we rely upon the following visual cues to static and changing depth and layout: occlusion, angular elevation in the scene, motion perspective and perspective transformation, relative size, binocular disparity, texture density, and motion in depth. These cues embody the character of the space, its spatial separation from our bodies and its accessibility. In ambulatory space, peripheral vision also becomes very important in the detection of motion.

It is also possible that the more gradually changing state of physiological visual cues in this space may also reflect its character — inappropriate near convergence and accommodation have been suspected^{92,93} of directly or indirectly causing egocentric distance misperception and of misinforming visually-guided locomotion. In particular, when head-up (HUDs) or head-mounted displays (HMDs) are used to combine virtual images with a true background, convergence and accommodation may respond to the projected images or the optical combiner itself, and thereby affect judgements or actions in physical space. That this might impact visually-guided locomotion has been

suggested by Stanley Roscoe⁹². As one of several anecdotal examples, he offers: "...When flying by reference to panel-mounted or head-mounted imaging displays, helicopter pilots approach objects slowly and tentatively, and still they are frequently surprised when an apparently distant tree or rock suddenly fills the wide-angle sensor's entire field of view."

In addition, visual, haptic, and vestibular senses give us a sense of our own position and motion within ambulatory space; shifting pressure or shearing forces on the skin, the senses of muscular effort, changing body position and orientation, and visual motion cues are all essential informers of spatial perception and behavior. Ambulatory space is primarily a display space in which we strategically and cognitively situate ourselves, and where we monitor and predict the behavior of other moving objects and entities. We operate in and on this space by navigating and moving ourselves through it, and by interacting with other entities, things, and events exhibiting ambulatory-scale movement. All these qualifications of ambulatory space are summarized in Table 2.

Ambulatory space is a bit harder to design for than manipulatory space. Though doable, it remains challenging to outfit a larger space for sensing and display, and it's also cumbersome for a participant to carry these with her as she ambulates. Still, many technologies exist, most of them still experimental, to accomplish both of these approaches. As mentioned before, flight simulators, entertainment-rides, expanded-screen theaters and ambulatory VR environments are examples of ambulatory-space technology. Here, motion platforms, (sometimes stereoscopic) displays that accommodate head motion and have a large field of view, are employed to display information appropriate to the scale and character of the space. There are still technological limitations that cause simulation lag and latency, compromising the important tight link between viewer-sensed self-motion and sensed changes in the environment. And of course, motion platforms cannot always display the forces from acceleration that should accompany visual motion through the environment. Finally, as mentioned previously, the use of HMDs or other wearable visual displays may compress visual space by some amount, which could in turn affect spatial behavior.

An interesting example of an interactive system in ambulatory space is the "City of News" by Sparacino *et al.*^{94,95}, which has its roots (in part) in the fascinating and ancient spatial scheme for organizing memory called the *memory palace*⁹⁶. The system is a 3D web browser that assembles a 3D urban landscape of information. In this system, a participant wears a jacket that houses tracking transmitters, computer, minimal keyboard embossed on the jacket sleeve for input, and power. For display, commercial lightweight SVGA head mounted glasses are used with one transparent aperture, and one display aperture. A participant is free to navigate a modest-scale constructed environment while mapping browsed web-based information onto the facades of selected landmarked buildings in a set of familiar cities. Here, the structure and navigation of "cognitively mapped" city spaces can be associated with information for presumably easy later retrieval.

egocentric subspace	egocentric distance	primary ranked visual layout cues	important haptic senses
<i>ambulatory space:</i>	1.5-30m and 1.5-20x eye-heights	occlusion, angular elevation, motion perspective, relative size, binocular disparity, relative density	cutaneous, kinesthetic, and proprioceptive cues

Table 2: Ambulatory space: basic visuo-haptic provisions

2.6.4 Reconnaissance space

In reconnaissance space, we rely upon occlusion, relative size, angular elevation in the scene, relative density, aerial perspective and motion perspective as cues to visually reveal static and changing spatial relationships. A wide visual field of view is essential. Here again, the stable and unchanging state of accommodation and convergence may be a cue to the vast egocentric distances characterized by this space. Sensing in this space can often be more ambient (visual and auditory survey, haptic sensing of temperature, moisture or wind changes), since greater distances generally imply the availability of more time to respond to whatever's out there.

In the absence of anything in manipulatory, ambulatory or reconnaissance space requiring immediate attention or monitoring, the vast character of this space can be contemplative and inspiring. Architects design for this space to evoke feelings of sublimity and spirituality, and impressions of omnipotence. For instance, the upward rush of form and structure in a Gothic cathedral invites us to turn our eyes and head skyward; bending the neck far backward and replacing the stabilizing visual reference to near environment with the appearance of brilliant light dematerializing distant stone can literally and emotionally induce unstable and dizzy feelings. Reconnaissance space is primarily a display space that we monitor and in which we strategically and contemplatively situate ourselves. These qualifications of reconnaissance space are presented in Table 3.

The Hubble Space Telescope, traffic helicopters, and weather satellites are good examples of technological sensing systems designed for reconnaissance space. Of course, *sensing* phenomena on the scale of reconnaissance space has become technologically feasible, while *displaying* that information at a similar scale is a much harder problem. Examples of currently feasible reconnaissance-scale displays are basically architecture and large-scale environmental art, or *Land Art*. Typically, the various information collected from technological reconnaissance sensing is relayed to an observer through scaled-down local displays which allow us to monitor distant, possibly changing information, and construct general strategic readiness.

The "Ambient Room"⁹⁷ and some applications of "Music Bottles"⁹⁸ of Ishii *et al.* are excellent examples of reconnaissance space tasks mapped into manipulatory and ambulatory spaces. The style and mode of display in these systems speak to ambient

sensing, evoking wind, background sounds *etc.*, while modes of interaction reflect those found in nearer spaces. The allure and quiet magic of the information and interaction in these systems is (at least) threefold: first, in the poetics of symbolically containing or invoking "uncontainable" information in metaphorical ways; second, in the convenience and relative spatial coziness of manipulatory and ambulatory access to remote information; and third, in the pleasure of monitoring or manipulating beautiful objects to retrieve that information.

A contemplative and magical example in the extreme is Étienne-Louis Boullée's "Centograph for Newton" c. 1784, a consummate (though unrealizable) technological proposal for this vast space. On paper, this design was intended as an architectural memorial to Isaac Newton, in the form of a vast empty sphere/machine whose manifestation would be the essence of Edmund Burke's definition of the *sublime*⁹⁹. Boullée intends his design to have enormous and theatrical scale, in which he can only "paint with nature". His sketches describe a vast orb whose exterior reflects the light and arc of the sky, and whose interior shell renders the heavens at night, with nothing put between them and a visitor to the quiet, expansive space. In his book reviewing European architecture between 1750 and 1890, Barry Bergdoll¹⁰⁰ comments "...In this great hollow sphere, a space in which the hand cannot confirm what the eye sees, one is overcome with the palpable sense of infinity; by bodily sensation one understands the magnitude of Newton's discoveries."

egocentric subspace	egocentric distance	primary ranked visual layout cues	important haptic cues
<i>reconnaissance space:</i>	>30m and >20x eye-heights	occlusion, relative size, angular elevation, relative density, aerial perspective motion perspective	temperature, moisture, wind, orientation to ground plane

Table 3: Reconnaissance space: basic visuo-haptic provisions

2.6.5 Temporal issues matter too

In interaction with the physical world on *any* spatial scale, latency, lag, and low frame rate are not things we normally encounter; these are very real problems in technological and computational systems we design. In our more conventional interactive systems, subjects are not explicitly viewing their *hand* or a *hand-held tool* manipulating objects in the workspace, but are instead visually monitoring a *hand-controlled cursor* moving toward and affecting displayed information. The state of the cursor is usually determined by polling a tracking device, like a mouse, held by the hand. The effective sampling rate of these tracking devices (which may include time for data filtering), software overhead, and the time it takes to update output devices can all contribute to some degree of temporal delay in a system.

We can perform very well in this kind of workspace, as long as lag is negligible and visual frame rate is high; both can affect one's feeling of *presence* in a virtual

environment¹⁰¹ and, both can significantly impact task performance. Lag in head-tracked stereoscopic displays has been found not to impair performance to a crucial degree (though it can promote motion sickness¹⁰²), but lag in the hand-tracking system is critical to performance. Ware *et al.*¹⁰³ modeled the effect of hand-tracked lag as a multiplicative variable to the Fitts' Law index of difficulty; using this model they show that even a small lag can have a significant effect on performance of sensory-motor tasks (measured as increased movement times and error rates) in interactive systems — with lag measured at 75ms, performance differences are notable, and at 225ms, performance suffers considerably.

Many studies have reported that eye-hand coordination becomes very difficult when frame rates drop below 10 fps (7fps¹⁰⁴ or 4fps¹⁰⁵); movement of the hand becomes saccadic for both monoscopic and stereoscopic visual feedback. Even operating with a visual update of 10 fps on dynamic information can be quite challenging. The presence of lag in a system, whatever its source, will obviously increase the amount of time it takes a user to complete any action which sets the scene, and compromise the comfort with which that action is accomplished. Thus, not only do we need to concern ourselves with supplying correct cues to depth and layout in a scene, displaying correct temporal relationships is crucial too.

2.6.6 Summary

As presented, this spatial framework for considering sensorial and interaction design has both strengths and shortcomings. We have already listed the need to include audition as an important spatial sensory modality. And obviously, not every task maps so obviously into this basic suite of behaviors and calls for the associated sensorimotor affordances — however, sensorimotor plasticity may permit, and poetics might suggest certain refreshing and imaginative mappings to be made. This spatial classification scheme considers our basic physical operation within characteristically different subspaces around us, and offers an encapsulation of diverse human factors information. It does *not* address the ways we might use space to think, or how we solve spatial cognitive problems.

Further, as it stands, this classification of space serves to frame task-based system and interaction design, but offers no specific provision for recreational social or collaborative work systems. Social considerations within different egocentric spaces are complicated and vary according to culture, custom, context, and individual differences in comfort. These issues are considered in other spatial classifications — like ones that frame the design of architectural and urban spaces¹⁰⁶. These may be compatible with the one we suggest and are certainly important to investigate, especially as more technological means for social interaction becomes available and digital communities grow.

However, as it stands, this spatial framework can help us to

- investigate and weigh the usefulness of different sensing and displays,
- of controllers and the repertoire of actions they admit, and
- to provide sensorial and interaction design that speaks to our sensorimotor abilities for specific spatial tasks.

Expanding this framework to include important missing sensory modalities, social considerations and the sensorimotor bandwidth and resolution requirements at ambulatory and reconnaissance scales are left for future work.

We might also consider that phenomenological experience and imagination also influence behavior and perception. For instance, our assumptions about object form, function, and behavior are largely built upon our experience; but as adaptive and imaginative individuals, we can be persuaded by unusual but convincing demonstrations of something new. Thus, in the interest of designing the spatial and temporal characteristics of computational objects, agents, and phenomena for *imagination*, future work might also consider what makes them *believable*.

3. Putting visual display in the manual workspace

Many other research efforts are collectively advancing technology and interaction styles that provide some of the basic properties of computational plastics summarized previously. In particular, our work is inspired and motivated by those efforts which truly co-locate eyes and hands in the workspace. Of course, most display and control channels for interaction — including a typical computer monitor and mouse — offset the visual and manual workspaces and offer an experience that might be described as "looking *here*, acting or feeling *there*". While this approach is certainly valid for many applications, we focus on a different approach in this chapter by describing several styles of spatially coincident visuo-manual interaction and some important research that demonstrates them.

3.1 Visual display in the manual workspace

3.1.1 Instrumented physical interface

One approach to joining the eyes and hands in a coincident workspace is to use manipulable, instrumented, physical objects as controllers for computational objects or processes. Various research efforts are investigating the use of physical handles to virtual objects by attaching sensors, tiny displays, and other electronics to real objects. These *tangible* objects then act as *physical controllers* for virtual processes or *representations* of the processes themselves, providing whole-hand interaction and rich visuo-haptic feedback that seems both natural and obvious. In these applications, a participant perceives her own body interacting with physical interface objects, and monitors the action-outcome on either a coincident or a spatially disparate display.

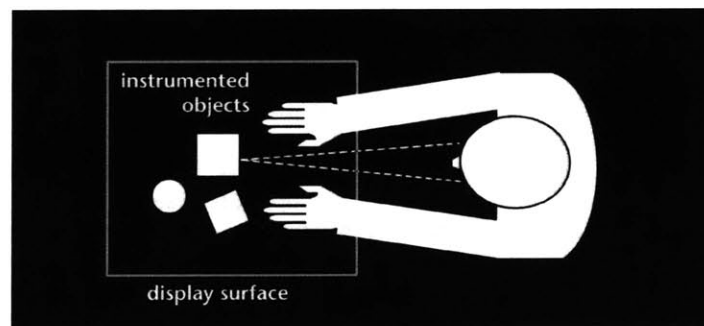


Figure 3.1 Instrumented physical space

One such project by Fitzmaurice *et al.*¹⁰⁷ called *Graspable User Interface: Bricks*, instrumented basic physical objects called "bricks", which acted as physical instantiations of either virtual objects or functions. Once a brick was attached to a virtual object, the computational model became itself *functionally graspable*. A brick might be used, for instance, to geometrically transform a virtual object associated with it, availing direct control through physical handles. Visual, tactile and kinesthetic feedback are present and exploitable with such an interface; thus the ability to operate

quickly and efficiently, using two-handed input is possible. Depending on how the workspace is arranged, computed information can be displayed coincident with or spatially offset from physical controllers.

Two systems which present a kind of computational desktop space without "monitors and mice" have been demonstrated; one is Wellner's *Digital Desk* project¹⁰⁸ at EuroPARC and the other is Ishii's *metaDESK* project^{109,110} at the MIT Media Laboratory. Both provide desktops on which physical and digital information commingle, and accept either gestures made with hands / pencils (Digital Desk) or the manipulation of active objects (metaDESK) as input. The Digital Desk project represents an attempt to render the computer desktop onto a real desk surface, and to merge common physical desk-objects with computational desktop functionality. The system employs a video projector situated above the desk for display of information, and a nearly co-located camera to monitor a person's movements in the workspace. Hand gestures are interpreted by a computational vision algorithm to be requests for various utilities that the system offers.

The metaDESK project attempts to physically instantiate many of the familiar GUI mechanisms (menus, windows, icons, widgets, *etc.*) in the form of tangible user interfaces (TUI's). The mapping between physical icons and virtual ones can be literally or poetically assigned; for instance placing a small physical model of MIT's Great Dome on the desk surface might cause an elaborate map of MIT to be displayed. In addition to summoning the map to the display and indicating its position, the physical Great Dome icon can be moved or rotated to correspondingly transform the map. The metaDESK system design includes a flat rear-projected desk surface, physical icons and functional instruments for use on the surface. The state of these physical objects is sensed and used as application input. Not only can the state of virtual objects be changed by manual interaction with physical objects, but part of the display itself can be "hand-held" and likewise manipulated.

Underkoffler's *Luminous Room*¹¹¹, mentioned previously, shows a more spatially and thematically literal coupling between the physical controllers and the displayed quantities they manipulate — so literal in some applications, that the disjunction between controllers and displayed information nearly vanishes. Here, as in other more recent examples from Ishii's research group, the tangible interface becomes a literal or abstract representation or extension of the displayed information, rather than just a simple physical handle. For example, in an urban planning application¹¹², a person manipulates physical architectural models, which the system tracks and registers with computationally-generated and projected shadows and reflections. It is quite natural for a user to think of these shadows as arising from the model itself, and not as "system output". Perhaps this system, more than any other that uses tangible controllers, spatially and functionally blends haptic and visual I/O into a seamless coincident display.

These representative systems suggest a broad and imaginative palette of ideas for instrumenting interactive space, and appeal to the rich set of sensibilities and skills people develop from years of experience with real world objects, tools, and their

physics. The richness of visual and haptic cues resident in these systems does indeed compare with that available when operating entirely in an unwired physical space, and natural strategies for manipulating the instrumented controllers can often be used.

3.1.2 Magic windows

A very interesting system which incorporates force feedback but no stereo viewing has been demonstrated by researchers at Carnegie Mellon University called the *WYSIWYF* (What You See Is What You Feel) display^{113,114}. The visual display behaves like a moveable "magic window", interposed between the viewer's eyes and hand, and through which the hand can be seen interacting with a virtual, tangible scene. The work employs a six degree-of-freedom haptic manipulator and monographic visual rendering to combine three pieces of information in this final coincident spatial display: a video image of the operator's hand/arm, the computer graphically rendered scene, and the accompanying force model. An interacting person looks "through" the display to monitor her hand, sees its video image manipulating a model, and feels forces due to the interaction.

Visual display and capture are provided by a color LCD panel with a CCD camera attached to its backplane. This display/camera unit can be moved with respect to the physical scene, while vision-based pose estimation is employed to determine its new orientation. The displayed image is assembled by compositing a computer graphic view of the synthetic scene, generated from the newly-computed viewpoint, with a live Chroma Keyed image of the operator's hand/arm interacting with the haptic device. This display cannot currently reproduce correct occlusion relationships between the hand/arm and virtual objects and provides only monocular cues to scene depth (no stereo viewing or head-tracked motion parallax is available). Yet, through its display window, the operator can see a view of her hand presented where her real hand is spatially positioned, and can feel forces due to interaction with the virtual environment.

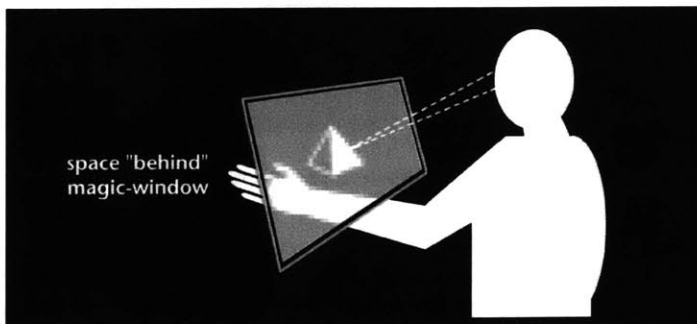


Figure 3.2 See-through "magic windows" for compositing information

3.1.3 Magic glasses and optically-combined spaces

An example thematically related to one of our holo-haptic projects is the compelling

Virtual Lathe described by Deering in 1992¹¹⁵. In this work, a head-tracked stereo display showed a virtual stock, spinning about its long axis, which a person could interactively lathe using a 3D mouse in the shape of a rod. The demonstration required a participant to wear LCD shutter glasses for stereo viewing, presented the stereoscopic space straddling the workstation screen, and used techniques to metrically correct and register stereoscopic display space with physical space. Since the shutter glasses transmit the actual physical scene while time-multiplexing stereoscopic views to the eyes, the physical tip of the 3D mouse and the hand holding it could be seen in contact with the three-dimensional virtual stock. However, in this work, no force display was available to indicate contact or carving.

A wide variety of virtual reality (VR) and augmented reality (AR) application areas such as telesurgery, entertainment, and maintenance analysis and repair, do employ computational haptics and stereo computer graphics to feel, see, and interact with data. Most existing demonstrations offset the visual and manual workspaces, but a growing number of efforts to join eyes and hands in a coincident workspace are being reported. One compelling example is UNC Chapel Hill's *nanoWorkbench*¹¹⁶, which not only allows an operator to see, feel and interact with three-dimensional molecular models, but permits her to work at an entirely different *scale* as well. The technology used in this pioneering project has evolved over time, but the system currently uses force feedback and head-tracked stereo visual display with LCD shutter glasses, and lets the hand-held device appear to operate directly on the visually-displayed data.

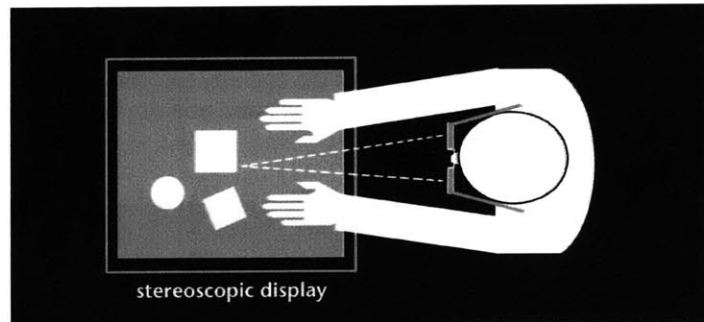


Figure 3.3 Seeing your body (or your body implied) combined with the simulation

In other systems which employ a coincident workspace, the use of a silvered or half-silvered mirror to either combine or superimpose an image of the CRT's pixel plane with the haptic workspace is a historically popular and frequently used technique (probably first reported by Schmandt¹¹⁷). One such example is the *Virtual Workbench*^{118,119}, developed at the The Virtual Environment Technologies for Training (VETT) Group at MIT's Research Lab for Electronics. This system, used to study human sensorimotor capabilities and to develop training applications, employs a Phantom haptic interface and uses the mirror technique to visually substitute stereo computer

graphics for the actual view of the haptic workspace. Additionally, audio cueing is presented in the system which is designed to present all three modalities in spatial register. Using the Virtual Workbench, an operator looks toward the space where her hand is operating the haptic device and feeling force cues, sees an optically-relayed stereoscopically-rendered view of the model and watches the tool operating on it. No view, either real or rendered, of the hand is presented.

At SIGGRAPH'99, Billinghamurst *et al.* showed a demonstration called *Shared Space*¹²⁰, which arose from the collaboration between the University of Washington and Sony ATR. In this demonstration, a lightweight HMD and video camera assembly are worn by a viewer; computer vision techniques are used to extract scene geometry, and rendered data is then carefully composited with the real-world view. Using the system, a viewer may hold a physical card in her hand and see a moving video image or a 3D animated character displayed on its surface. This demonstration combines haptic feedback from real-world objects and stereo visual display of simulated scenes.

3.1.4 Magic mirrors

Many systems present a viewer with a display that shows them composited and interacting with additional rendered or optically-captured information. The use of video for interactive capture and display has been demonstrated literally for decades by Myron Krueger; his evolving *VideoPlace* installation¹²¹ allows a viewer's "digital shadow" to interact with composited information. Performance issues aside, our experience with shadows and mirrored reflections of ourselves allows such composited displays to be visually persuasive, if they are carefully rendered and projected.

The *ALIVE* system¹²², developed at the MIT Media Laboratory used a reflected display format and permitted people to interact with autonomous agents (a dog and a hamster) exhibiting convincing and lifelike behaviors. Especially since gestures were used to interact with the animals in *ALIVE*, and no high-precision manual skills were required, the "magic-mirror" was strikingly effective — unless a viewer tried to pet the dog, her sense of occupying an ambulatory space with computational animals was quite strong.

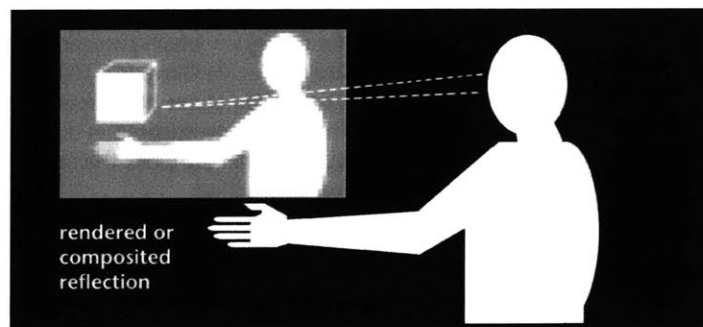


Figure 3.4 Magic-mirror display format

A simpler version of the Shared Space¹²⁰ demonstration presents a graphics window in which a video image of an interacting person holding a rectangular card is reflected, and simple synthetic objects are rendered as sitting on the card. The visual responsiveness of the demonstration and using two hands to both *feel* and *manipulate* the card make this experiment especially convincing.

3.1.5 Spatial projectors

A system which optically projects a three-dimensional view and offers programmable haptic display is Dimensional Media's *High Definition Volumetric Display*. This system incorporates force feedback and a re-imaging display, which employs optical components to relay and composite images of already-existing 3D objects and/or 2D display screens. As a result, visual images are strikingly realistic, and a force feedback device can be used to inspect with the optical output. While haptic modifications are possible, changing the geometry of the visually displayed object is not possible with such a system.

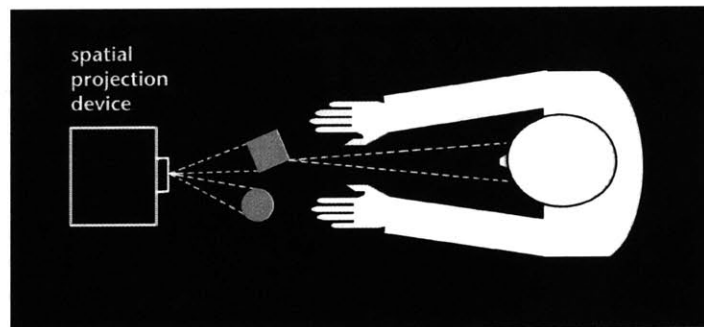


Figure 3.5 Spatial projectors reconstructing a simulation in the workspace

The combination of haptics and holography was first reported by researchers at De Montfort University for an object inspection task¹²³. Visual display was provided by a reflection transfer hologram which presented an aerial image of a control valve. A Computer Controlled Tactile Glove (CCTG) provided coincident haptic display of the same data. Similar combinations of reflection transfer holograms and force-feedback were also informally later investigated in our laboratory. But, in all of these efforts, the interacting hand could literally block the reflection hologram's illumination and prevent image reconstruction.

3.1.6 Exploring holography: an ideal holo-haptic system

However, the immediacy of seeing and feeling the interaction among image, hand, and tool — even with its many technical problems — was compelling; this early work catalyzed the research described in this dissertation.

As a useful point of reference for discussion of this work, Figure 3.6 shows an ideal holo-haptic system. The top of Figure 3.6 shows an interacting person seeing and feeling cues being displayed within the multimodal workspace. At the right, a haptic device is indicated which displays force and tactile cues to the interacting person, tracks their manual interaction and reports it back to the haptic simulation. Here, simulation dynamics are computed, and appropriate haptic cues are sent back out into the workspace. These updates must occur *at least* 1000 times per second in order to maintain simulation stability.

In this ideal system, changes in the haptic model are propagated instantly to the model underlying hologram computation; that computation is performed and the holographic display is updated at least 30 times per second. In this way, haptic changes are also reflected in the visual channel without noticeable lag, and the underlying hologram and haptic models are perceived by the interacting person as being in perfect spatial register.

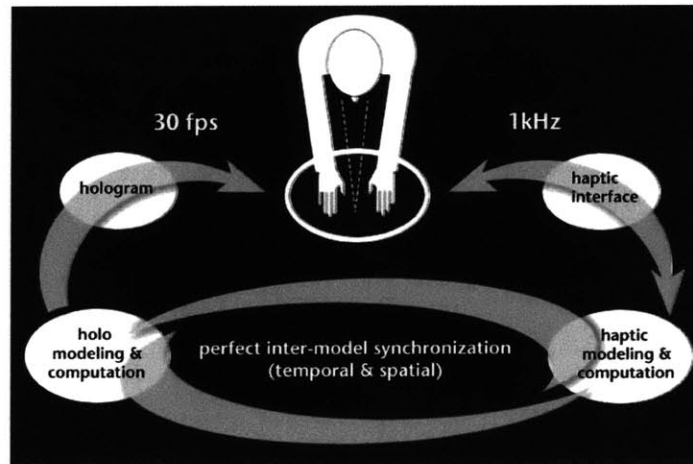


Figure 3.6 Ideal holo-haptic system

Even our early work will meet the specifications on the right side of Figure 3.6, but the left side (holographic pipeline) will present the greatest engineering challenge. In order to illuminate the issues that underlie that challenge and some methods to address it, the next chapter will describe holography in more historical and technical detail. Following that, we will describe the combination of holography and force feedback in three experiments.

4.1 Basics of display holography

4. Holography: from passive to interactive display

4.1.1 Holography in context

In 1947, while working to improve the resolution of the electron microscope, Denis Gabor invented holography¹²⁴. Thus was born a compelling three-dimensional display method whose fundamentals, oddly enough, were understood well-before any means of recording and play-back were available to give them visual form.

Not until 1960, when Theodore Maiman reported the pulsed ruby laser, and 1961 when Ali Javan developed the continuous-wave Helium-Neon (HeNe) laser, was the requisite source of coherent light finally available; a year later in the U.S.S.R., Yuri Denisjuk produced white-light-viewable reflection holograms¹²⁵. Nearly simultaneously, Emmett Leith and Juris Upatnieks introduced an improved *off-axis* recording modification to Gabor's ideas, and were able to produce (now-famous) laser transmission holograms¹²⁶.

Among other endeavors, this early work inspired the idea of *computing* holograms. And here again, the fundamentals of generating complex-valued fringe patterns were developed and understood long before any viable recording or play-back device could demonstrate them. Consequently, early techniques^{127,128,129,130,131,132} were developed to suit the limited means of display available; computing and recording holograms truer to their optically-produced counterparts would still await a device that could easily write and reconstruct them.

With the invention of a holographic video system by Benton *et al.* in the late 1980's^{133,134}, such a device finally became available. Carefully computed physical models of monochromatic light propagating from a computational object, and interfering with a reference wave could be easily displayed for viewing. For the first time in computational holography's history, need for better computational approaches to holographic image-making arose *before* the available display technology would fail to admit their testing and verification.

But, of course, the advent of electro-holography during the proliferation of *interactive systems* framed research strongly in that context; shortly after the first truly visually compelling images were computed^{135,136}, generating and updating images rapidly took priority over improving computational models for visual realism. In the interest of efficiency, computational methods have been designed to employ lookup tables¹³⁷, difference methods¹³⁸, non-uniform sampling methods¹³⁹ and compressed, encoded representations of the fringes themselves¹⁴⁰.

As a result of this push for interactivity, present electro-holographic systems can achieve frame update rates of almost ten frames per second. Certainly, this frame rate is still lower than that offered by other 3D display technologies like head-tracked stereo computer graphics, and the image quality of electro-holographic displays remains generally inferior. That said, however, algorithms for rapid hologram generation, as well as technology underlying the required computation, communication, and modulation bandwidths, are certainly improving in the favor of

holography's viability.

While the race for speed in computation and playback continues, the quiet promise of computational holography remains its capacity for remarkable realism. Consequently, current research approaches often operate around a distinct trade-off: methods to improve realism do not support interactivity, and methods for rapid image generation currently operate at the cost of image quality. Yet, for electro-holographic technology to be a truly qualified display choice in an *interactive* forum, and particularly in any which incorporates haptic feedback, responsiveness *and* superior image quality will be required of it.

In the interest of advancing computational holography's potential for *both* realism *and* interactive update, our current methods are focusing on incrementally-updating holograms computed with the physical simulation of interfering wavefronts. This incremental computing method allows scene-based changes to be rapidly incorporated into the holographic image, is designed to be suitable for hardware assistance, and is also compatible with techniques that provide realistic scene rendering. It is hoped that this approach might be pressed into reliable interactive updating without compromising the visual realism at the true heart of holography.

In this section, we first provide some background by touching upon the fundamentals of traditional optical holography and holographic stereography. We then review various computational approaches in display holography, some of historical significance, and some capable of producing striking visual images on current electro-holographic displays. We will conclude the chapter by describing our current experiments with incremental computation of hologram updates.

4.1.2 Traditional holography

The basics of traditional holography are conceptually straightforward. In a recording step, a photosensitive material (usually film) is exposed simultaneously to two coherent beams of light. One beam, called the *object-beam*, is formed by light that has been scattered from an object or scene. The other beam, call the *reference-beam*, is

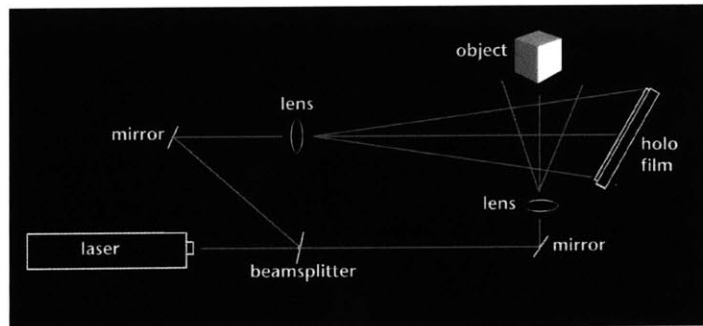


Figure 4.1 A simple off-axis holographic recording setup.

comprised of unmodulated laser light. The reference beam allows both the magnitude and phase of the incident object wave to be recorded wherever both beams overlap on the recording material, essentially recording variations in object-beam intensity and its direction. Figure 4.1 shows an example of one standard hologram recording geometry.

After processing the exposure, an image of the original object or scene can be reconstructed by illuminating the recorded pattern with an illumination beam which is related to the original reference (Figure 4.2). For a viewer looking through the hologram, each eye is presented with a slightly different image, and a continuously-varying parallax view of the scene is available during movement through the viewzone. Through these processes of encoding (interference) and decoding (diffraction) the description of the scene, once reconstructed, is nearly indistinguishable from the original recorded scene; and this is the essence and the simple magic of holography.

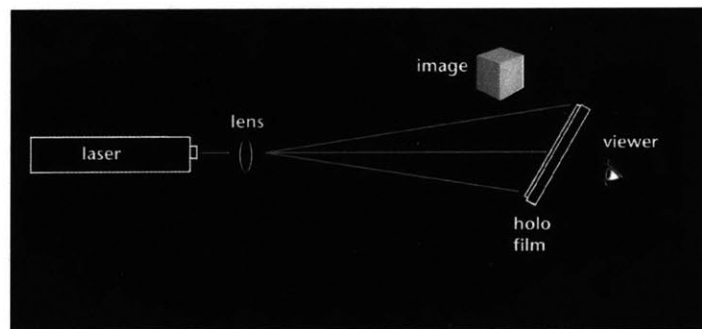


Figure 4.2 Reconstruction of off-axis hologram.

4.1.3 Holographic stereograms

However magic, holography has certain limitations which quickly challenged image-makers. Among them were the difficulty in controlling scene lighting, and restrictions on size and the material quality of recordable objects. A solution which circumvented these problems was found in a quasi-photographic, quasi-holographic process called holographic stereography¹⁴¹.

In the hologram recording step, holographic stereograms substitute a sequence of carefully captured or generated two dimensional views of a scene for the scene itself. This sequence of 2D projections describes a set of parallax views of the scene, much more coarsely sampled than a traditional hologram would regard. If many requirements are met^{142,143,144} (*i.e.* if the geometry for capture of the 2D views is carefully matched to the hologram recording and viewing geometries; if the individual 2D views are taken very close to one another; and if the scene depth is not too great for the view sample spacing used) the holographic stereogram can display convincing binocular and motion parallax in addition to all the pictorial detail in the original

captured views. For a viewer looking through the holographic stereogram, each eye is presented with a slightly different 2D image, and a smoothly sampled parallax view of the scene is available during movement through the viewzone. A simple image-capture geometry is shown in Figure 4.3, as well as basic holographic stereogram recording and simple reconstruction.

An extension of these basic methods was used to produce the edge-illuminated transfer holograms (first to render component images, and then to optically record the stereogram masters) for our system *Touch*, described in the next chapter.

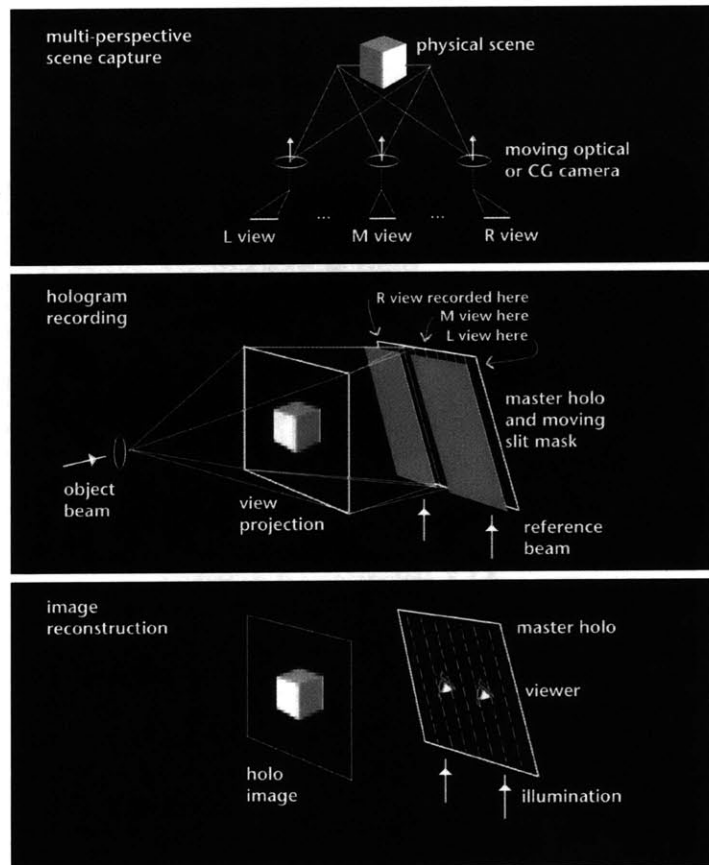


Figure 4.3 Image capture for, recording and reconstruction of holographic stereograms

4.2 Computational holography

4.1.4 A basis for computing holograms

At present, *computation* for electro-holograms usually takes one of these two general approaches also¹⁴⁵. In the "fully-computed" approach, the fringe pattern is computed at the recording plane by modeling light propagating from simulated surfaces and interfering with a reference beam. In the "stereogram approach", a number of two-dimensional perspective views of a scene are rendered and assembled into a fringe pattern that angularly multiplexes them. Both of these computational methods will be described in the following sections, after an overview of early computer-generated hologram techniques.

4.2.1 Early days: Fourier Transform methods

Historically and still, generating the complex-valued holographic fringe pattern presents two difficult problems: first, one needs to simulate and compute the pattern with enough spatial resolution to produce the intended image, and second, one must write the pattern to some material or device which can record its detail.

Most early techniques computed holograms appropriate for Fraunhofer diffraction, where the reconstructed two-dimensional image $i(\mu, \nu)$, as given by

$$i(\mu, \nu) = \iint h(x, y) e^{j2\pi(x\mu + y\nu)} dx dy$$

is essentially the 2D Fourier Transform of the hologram distribution, where μ and ν in the reconstructed image correspond to the hologram coordinates x and y , scaled by a constant related to the reconstruction geometry. The holograms themselves were computed as inverse Fourier descriptions of the desired image amplitudes. The patterns were recorded using fairly low-resolution, greyscale or binary output devices, and then reconstructed in a manner shown in Figure 4.4.

The computational *binary detour-phase hologram* developed by Brown and Lohmann¹²⁷ approximates the amplitude and phase of an object's inverse Fourier Transform with an $M \times N$ grid of binary-valued elements. Each element, corresponding to a single sample of the overall pattern, is an opaque cell of dimension $W \times H$ in which rectangular aperture of variable dimension $w_{mn} \times h_{mn}$ has been opened. The area of the aperture in each cell is proportional to the magnitude of the Fourier coefficient it represents; the displacement of the aperture's center from the cell's center encodes the corresponding phase (Figure 4.5). The resulting far-field diffraction pattern $F(x, y)$, at an observation plane z_0 approximates the hologram's discrete Fourier Transform as the sum of diffracted fields $f_{m,n}(x, y, z_0)$ from each contributing aperture:

$$F(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f_{m,n}(x, y, z_0)$$

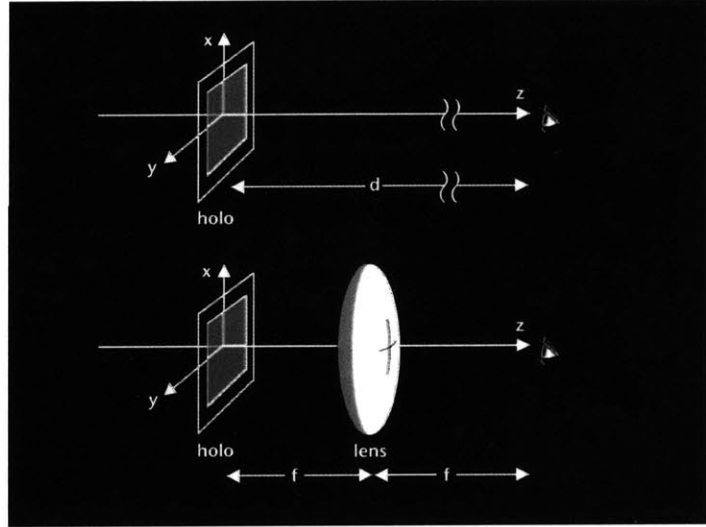


Figure 4.4 Reconstructed image is the Fourier Transform of the hologram distribution

If the hologram is reconstructed with collimated (plane) illumination, the $f_{m,n}$ can be described by

$$f_{m,n}(x, y, z_0) = E_0 w_{m,n} h_{m,n} \operatorname{sinc}\left(\frac{w_{m,n}x}{\lambda z_0}\right) \operatorname{sinc}\left(\frac{h_{m,n}y}{\lambda z_0}\right) \exp\left[j\left(k_x + \frac{2\pi x}{\lambda z_0}\right)(mW + \Delta_{m,n})\right] \exp\left[j\left(k_y + \frac{2\pi y}{\lambda z_0}\right)(nH)\right]$$

where $k_x = 2\pi/W$ and $k_y = 2\pi/H$. In the far-field, where $w_{m,n}x$ and $h_{m,n}y$ are both $\ll \lambda z_0$ this expression for the $f_{m,n}$ can be given the following form:

$$f_{m,n}(x, y, z_0) = E_0 w_{m,n} h_{m,n} \exp[jk_x(mW + \Delta_{m,n}) + jk_y nH] \exp\left[j\left(x \frac{2\pi}{\lambda z_0}(mW + \Delta_{m,n})\right) + j\left(y \frac{2\pi}{\lambda z_0}(nH)\right)\right]$$

Since the area of each aperture is related to the magnitude of a corresponding sample in the computed hologram by

$$w_{m,n} h_{m,n} = |h(mW, nH)|$$

and each aperture's offset is also related to that sample's phase by

$$\Delta_{m,n} = \left(\frac{W}{2\pi}\right) \phi(mW, nH)$$

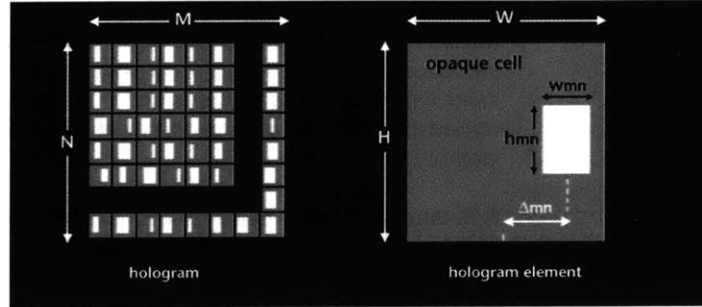


Figure 4.5 Binary detour-phase hologram (left) and a single cell (right)

where $\Delta_{m,n} \ll \lambda z_0$. Then the diffracted field from an aperture, as observed in the output plane, is given by

$$f_{m,n}(x, y, z_0) = E_0 |h(mW, nH)| \exp[j\phi(mW, nH)] \exp\left[j\frac{2\pi}{\lambda z_0}(xmW + ynH)\right]$$

Now the resulting superposition of such diffracted fields from *all* contributing apertures, which describes the reconstructed image, is expressed as

$$\begin{aligned} F(x, y) &= E_0 \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |h(mW, nH)| \exp[j\phi(mW, nH)] \exp\left[j\frac{2\pi}{\lambda z_0}(xmW + ynH)\right] \\ &= E_0 \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} h(mW, nH) \exp\left[j\frac{2\pi}{\lambda z_0}(xmW + ynH)\right] \end{aligned}$$

This expression is recognizable as the Discrete Fourier Transform of the hologram. Thus, if the hologram itself is computed as the inverse Fourier Transform of the desired final image, then this detour-phase method of encoding and subsequent reconstruction will reproduce that image.

Other similar methods followed the binary detour-phase approach. In 1970, Lee¹²⁹ developed a method for representing a Fourier hologram by dividing each cell in the output plane into four different vertical subcells at laterally displaced locations. The subcells themselves represent phase angles of 0, 90, 180 and 270 degrees; in each hologram cell, an arbitrary phasor could be encoded with the assignment of opacity or semi-transparency to these subcells. Burkhardt¹³⁰ implemented an approach with only three apertures per cell, representing three mutually noncollinear phasors, and thus simplified Lee's method of encoding.

In these types of holograms, values are computed at sampled locations, and the final patterns are written on printers or plotters (which introduce their own sampled

output) and then photographically reduced to obtain the high spatial frequencies required to diffractively produce images. The final images exhibit aliasing artifacts introduced by the printing process, which cause higher order images to reconstruct in lower diffracted orders. Other errors¹⁴⁶ also diminish the quality of the final reconstructed image; and the opacity of the overall mask reduces reconstructed image intensity.

Offering better image quality, the Kinoform¹³¹ uses a similar encoding method, but assumes a unity magnitude and modulates only the phase of the Fourier coefficients. The computed greyscale pattern is photographed and then bleached to convert intensity variations into a phase relief pattern that matches the phases of the Fourier coefficients. An extension of the kinoform approach is the referenceless on-axis complex hologram (ROACH), which encodes both the amplitude and phase of the Fourier coefficients in separate layers of color film¹³².

While detour-phase and kinoform-type holograms are useful as spatial imaging devices in some imaging and filtering applications, they produce only two-dimensional patterns far from the hologram plane. Attempts have been made to compute far-field holographic fringe patterns of *three-dimensional* objects by "slicing" them in depth, stacking their computed holograms, and, consequently, the far-field images of each slice¹⁴⁷. Computing a "slice-stacked" hologram in this fashion is cumbersome, however, and the final images are a more heroic than visually-satisfying result.

4.2.2 Moving forward: Interference simulation

In 1986, a different approach which instead models the propagation of light from an *analytically-described* three-dimensional object, and computes the interference of this wave with a reference wave was reported^{148,149,150}. This work featured the notion of representing an object as a collection of analytically-defined "primitives", and assembling a composite pattern from the interference of these primitives with the reference wave.

Shortly afterward, Underkoffler^{135,136} reported his work using analytic field contributions to generate computed patterns for the *first generation* MIT Holographic Video System (holovideo). This early work, motivated strongly by the quest for realism in computational holography, marked the first successful display of images on an electro-holographic system. Further, with its simulation of occlusion and smooth shading, this work set a standard for realistic rendering in computational holography which still holds at the time of this writing.

Underkoffler's object beam was analytically expressed as a sum of wavefronts generated by many self-luminous spherical emitting points, which were used to densely populate the surfaces of polygonally-described objects. The field radiated by one point source is given by

$$E_{sph}(\vec{r}, t) = \frac{E_{0sph}}{|\vec{r} - \vec{r}_0|} e^{-j\left(\frac{2\pi}{\lambda}|\vec{r} - \vec{r}_0| + \phi_0 - \omega t\right)} \quad 4.2.1$$

where E_0 is the source amplitude, and ϕ_0 is an arbitrarily assigned initial phase. In this work, as well as all electro-holography at the MIT Media Laboratory up to and including the time of this writing, holograms were computed for *horizontal parallax only* (HPO) systems. These holograms are diffractive in the horizontal direction, but have video-like resolution in the vertical direction. Thus, the holograms are modeled as a "stack" of line holograms, where the y-quantization matches the vertical resolution of the holovideo display. Underkoffler's collection of spherical emitters populated their underlying geometry such that each was assigned the appropriate y-value to situate it on one of the display lines.

The reference wave was modeled as a tipped plane wave, to match the physical illumination in the holovideo system. The field radiated by a plane wave travelling in direction \hat{q} is expressed as

$$E_p(\hat{q}, \hat{r}, t) = E_{0pl} e^{-j\left(\frac{2\pi}{\lambda} \hat{q} \cdot \hat{r} + \phi_0 - \omega t\right)} \quad 4.2.2$$

where ϕ_0 is an arbitrarily assigned phase offset. These scalar expressions for electromagnetic radiation assume that all point and plane wave sources share an identical linear polarization.

The elimination of t- and y-dependence in the field equations simplifies the physical modeling of wave interference, which is consequently performed only in the x-z plane for each "line" of the hologram (or *hololine*). Then, the expressions for spherical and plane wave radiation in equations 4.2.1 and 4.2.2 can be expressed in an easy-to-evaluate form:

$$E_{sph}(x, z) = \frac{E_{0sph} e^{-j\frac{2\pi}{\lambda} \sqrt{(x-x_0)^2 + (z-z_0)^2} - j\phi_0}}{\sqrt{(x-x_0)^2 + (z+z_0)^2}} \quad 4.2.3$$

$$E_{pl}(x, z) = E_{0pl} e^{-j\frac{2\pi}{\lambda} (q_x x + q_z z) - j\phi_0} \quad 4.2.4$$

where, for a plane wave tipped at angle θ_{ref} to the z-axis,

$$q_x = \hat{q} \cdot \hat{x} = \sin\theta_{ref}; \quad q_z = \hat{q} \cdot \hat{z} = \cos\theta_{ref} \quad 4.2.5$$

and where the location of the spherical emitter is given by

$$\hat{r}_0 = (x_0, z_0) \quad 4.2.6$$

Constructing an object wave as a sum of M spherical emitters, and modulating it with an inclined plane wave produces a composite field given by

$$E_{tot} = \sum_{i=1}^M E_{sph}^i(x, z) + E_{pl}(x, z) \quad 4.2.7$$

which can be evaluated at the hologram plane, and used to produce an intensity pattern as follows:

$$I(x, z) \propto |E_{tot}(x, z)|^2 = E_{tot}(x, z)E_{tot}^*(x, z) \quad 4.2.8$$

This intensity pattern is similar to that which film would (more continuously) record, if placed at the hologram plane and exposed to the same combined physical disturbance. A graphical representation of the process is shown in Figure 4.6; the hologram is computed as the magnitude squared sum of object point wavefronts and the collimated reference wave, at sampled increments of Δx in the $z=0$ plane.

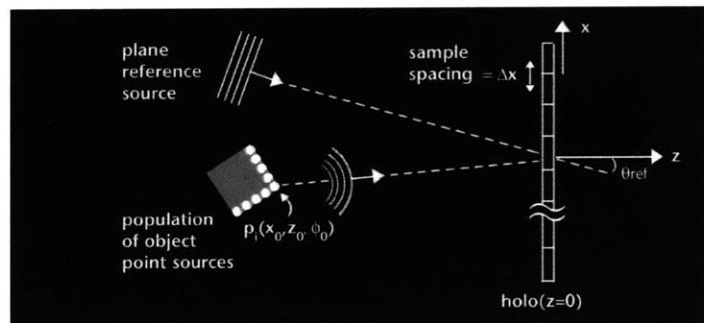


Figure 4.6 Simulated object and reference wave interfering at hologram plane

Since the computed intensity of the interference pattern is a sampled version of a continuously evaluable function of (x, z) , the maximum spatial frequency that pattern can represent must fall short of the Nyquist limit. Determining this frequency will be mentioned later in this chapter when we describe incremental computing.

Underkoffler's hologram computation, often referred to as the *fully-computed approach*, included occlusion processing, which was accomplished by calculating, for each object point, the hologram regions to which it contributes, given that it may be geometrically occluded in some parallax views by other spans of points in the scene¹⁵¹. This work displayed appropriate accretion and deletion of object parts as a viewer moved throughout the viewzone, and finally imparted a sense of solidity to even computed wireframe images.

Additionally, since object points populated underlying polyhedral data, each point could inherit its associated polygon's surface normal. Using this normal, an object point's amplitude E_{0sph} , from equation 4.2.3, could be made to express the output of a computer graphics-style shading calculation, based on assigned object surface parameters and N specified light sources, as in the simple model below:

$$E_{0sph}(x, z) = \left[I_a k_a + \sum_{i=1}^N I_i (k_d \cos \theta_i + k_s \cos^n \alpha_i) \right]^{1/2} \quad 4.2.9$$

In this simple illumination model, also illustrated in Figure 4.7, the ambient illumination reflected by the object surface is specified by both the incident ambient light intensity I_a and the surface's ambient diffuse reflection constant k_a , $0 \leq k_a \leq 1$. The diffuse light reflected from the object is modeled by I_i , which is the incident intensity from the i^{th} illumination source, and the surface's diffuse reflection constant k_d , $0 \leq k_d \leq 1$. Here, the angle θ_i separates a given object point's normal vector and the i^{th} light source direction. For specular reflections, n specifies the power that approximates the spatial distribution of specularly-reflected light, with α_i , $\pi/2 \leq \alpha \leq \pi/2$, as the angle between a reflection vector, R , and the "line of sight", between the point and a given hologram sample. In this very basic model, the constant k_s serves as a simple substitution for a more complex reflectance function. The final square root turns computed intensity values into amplitude values, suitable for interference modeling.

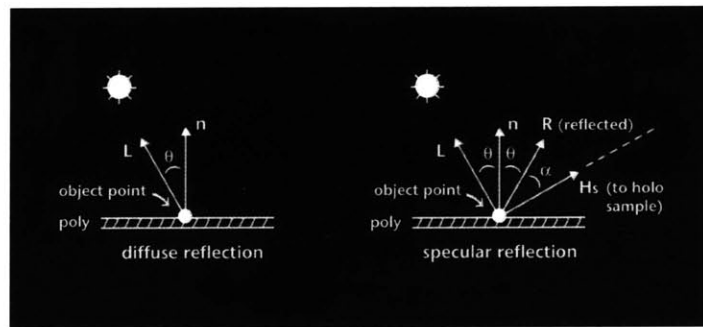


Figure 4.7 Simple modeling of diffuse and specular reflection

If all the object points populating a single polygon inherit their polygon's normal, or, if the points' normals are computed by interpolating across the underlying polygon in the style of Phong shading, this simple illumination model can be applied for every point, at every hologram sample, to vary amplitudes and give the appearance of smooth shading.

Underkoffler's work implemented Gouraud shading; Phong shading and other

computer graphics techniques such as texture and reflection mapping remain ripe for incorporating into hologram computation.

4.2.3 Moving faster: Computational shortcuts

Dissecting the intensity pattern given in equation 4.2.8 into its object and reference contributions reveals the *functional* composition of the distribution expressed in equation 4.2.8:

$$I(x, z) \propto |E_{tot}(x, z)|^2 = |E_{obj} + E_{ref}|^2 = \underbrace{|E_{obj}|^2}_A + \underbrace{|E_{ref}|^2}_B + \underbrace{2Re\{E_{obj}E_{ref}^*\}}_C \quad 4.2.10$$

Term *A* in equation 4.2.10 arises from the object points interfering with themselves (and is thus called *object self-interference*). In traditional optical holography, the unwanted artifacts generated by this phenomenon can be spatially separated from the reconstructed image by using a reference beam angle several times the size of the angle subtended by the object at the hologram center. In computational holography, where a more shallow reference and illumination angle are used, this term often introduces unwanted image artifacts — but, we have the option of not including it.

Term *B* in equation 4.2.10, called *reference bias*, represents a slowly spatially-varying bias that increases the intensity of the entire hologram. In traditional optical holography, we carefully adjust the ratio between the object and reference beam intensities to affect the modulation of the overall interference pattern, trading a low reference bias (and good fringe modulation) for weaker object self-interference fringes. In computational holography, again, we have the option of simply not including this bias term.

Finally, real-valued term *C* in equation 4.2.10 comprises the fringes that are sufficient to reconstruct the desired image. This term fully describes the interference between the object point wavefronts and the reference wave, and without terms *A* and *B*, the calculation of useful fringes can expand to fill the (still limited) dynamic range available for the computed pattern.

A technique introduced by Burch¹⁵², and revisited by Lucente¹⁵³ who called it *Bipolar Intensity*, uses only term *C* in equation 4.2.10 to model hologram fringes. If we borrow from equation 4.2.6, and specify that $z=0$ on the hologram plane, we can evaluate the distance from hologram samples to the i^{th} object point as

$$r_i(x) = |\vec{r} - \vec{r}_i| = \sqrt{(x - x_i)^2 + z_i^2} \quad 4.2.11$$

Then, using plane and spherical waves (equations 4.2.3 and 4.2.4) for the reference and object waves, respectively, only the useful fringes in the intensity pattern are

modeled by:

$$I(x, z) \equiv 2 \operatorname{Re} \{ E_{obj} E_{ref}^* \} \quad 4.2.12$$

$$\begin{aligned} &= 2 \operatorname{Re} \left\{ \left(E_{0pl} e^{j \frac{2\pi}{\lambda} x \sin \theta_{ref} + j \phi_0} \right) \sum_{i=1}^M \frac{E_{0sph}^i e^{-j \frac{2\pi}{\lambda} r_i(x) - j \phi_0^i}}{r_i(x)} \right\} \\ &= 2 E_{0pl} \sum_{i=1}^M \frac{E_{0sph}^i}{r_i(x)} \cos \left(\frac{2\pi}{\lambda} (r_i(x) - x \sin \theta_{ref}) + \psi^i \right) \end{aligned}$$

given that $x(\hat{q} \cdot \hat{x}) = x \sin \theta_{ref}$ for the plane reference wave evaluated in the $z=0$ plane.

In this form, the interference is approximated by a scaled sum of M elemental interference calculations, as M point sources each individually interfere with the reference wave. The new phase term ψ^i , $0 \leq \psi^i \leq 2\pi$, describes the initial phases of the reference and the i^{th} object point waves lumped together, and is assigned a random value for each elemental interference calculation.

Given that computed holograms are usually normalized to fit the available dynamic range for a given output device, the uniform scale by the reference beam amplitude can be eliminated so that the intensity pattern can be evaluated from the following model:

$$I(x, z) \equiv \sum_{i=1}^M \frac{E_{0sph}^i}{r_i(x)} \cos \left(\frac{2\pi}{\lambda} (r_i(x) - x \sin \theta_{ref}) + \psi^i \right) \quad 4.2.13$$

In 1992, Lucente used this approach to pre-compute a table of fringes, each of which modeled the interference of the collimated reference wave with a single point source at some z -distance from the hologram plane^{153,137}. This approach greatly speeded hologram computation, since computing fringes for a collection of points required only summation of pre-computed elements, and subsequent normalization. Moreover, since the reference wave was collimated, a single fringe used to reconstruct a point at depth d , would reconstruct a *translated* point at depth d when translated on the hologram plane. This approach forms the basis of the incremental computing method which will be described later.

4.2.4 Faster, with trade-offs: Computational stereograms

Almost as soon as pretty images appeared on electro-holographic systems, the need to address electro-holography's usability as output for an interactive "visualization-type" workstation rechannelled efforts toward faster computing. Mostly for this reason,

experiments with the computation of holographic stereograms began to dominate the field.

The stereogram technique greatly expands image content possibilities for computational as well as optical holography. Generating the component 2D views aside, the burden of fringe computation is fixed for a given display, and independent of scene composition (quite unlike fully-computed holograms). Furthermore, using this technique, images can be displayed of scenes for which there is no available analytic or underlying geometric description.

A computed stereogram is designed to behave like an optically-recorded one: when a viewer's eye is positioned in each *sub-viewzone* (just as shown in Figure 4.3), the hologram plane is observed to display one particular 2D view; moving from one sub-viewzone to the next allows the viewer to observe the next parallax view. The computed holographic stereogram is generated to diffractively deliver the correct parallax view to the correct place in the viewzone.

To computationally produce holographic stereograms, a topic examined extensively by Lucente^{154,155}, a set of N optically captured or computer-graphically rendered parallax views are required (each with resolution $W \times H$), as well as a set of diffractive patterns that angularly multiplex the views. These diffractive patterns are referred to as *basis fringes*.

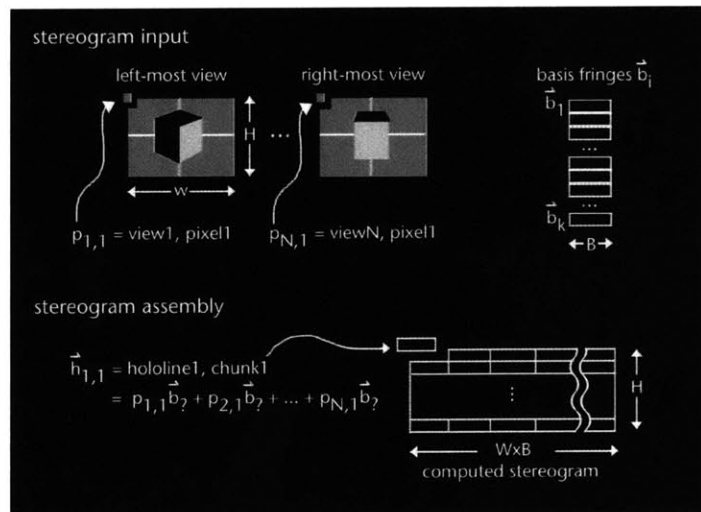


Figure 4.8 Computing holographic stereograms from 2D views and basis fringes

In HPO displays, the computed stereogram inherits the same vertical resolution H , as the 2D component views. In the horizontal dimension however, the hologram is

comprised of W chunks containing the same number of samples B , as each basis fringe b_i . Each chunk h_w , which is a linear combination of basis fringes weighted by image pixel values, is designed to display N different pixels, each in a different direction, and each over its entire spatial extent (Figure 4.8). Consequently, from each of the N sub-viewzones, the entire chunk is seen to "light up" with a different pixel value. This chunk is termed a *hogel*, or a "hologram-pixel".

To deliver the 2D view pixels well, so that there are no gaps in the viewzone and the image appears equally bright throughout, the basis fringes are designed to satisfy a set of spatial and spectral constraints. These design details, as well as the description of an iterative constraint method and a synthetic annealing approach used to generate the basis fringes are described in [156].

Determining which basis fringes are appropriate to each hogel is entirely dependent on the diffractive behavior one wants the hogel to exhibit, something that varies among display formats. In different display formats, a basis fringe can diffract the

incoming illumination in a variety of manners; for instance, from a particular pixel location on the hologram to a particular sub-viewzone as in Figure 4.9 (left). Stereograms of this type were some of the first computed — basis fringes were calculated to diffract light (of multiple wavelengths¹⁵⁷) from every pixel to every subzone of the holographic viewzone¹⁵⁵. This approach could require as many as $W \times N$ separate basis fringes for each wavelength. However, the number of fringes required could be less if the reconstruction geometry allowed some to be reused in multiple hogels within a hologram line.

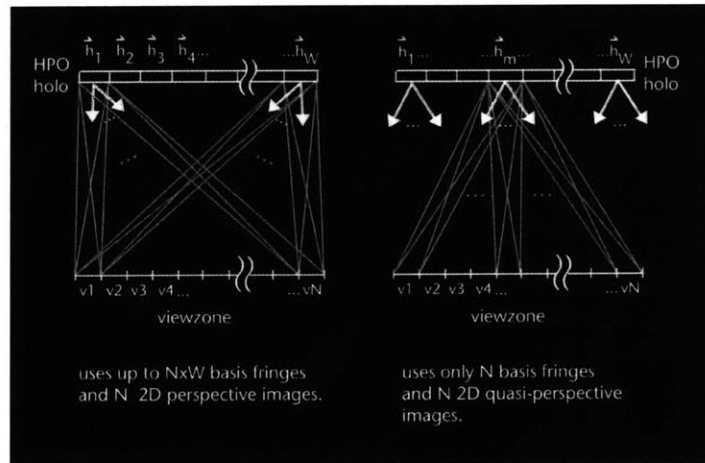


Figure 4.9 Using basis fringes to compute holographic stereograms

In later work, a smaller set of basis fringes was employed to project each 2D view from

the hologram plane in a collimated fashion (Figure 4.9, right). In this work, the number of basis fringes equalled the number of 2D parallax views. An example of this approach, which was used by Lucente¹⁵⁶ to generate images for the second-generation MIT holovideo system, employed just 32 basis fringes, each 1024 samples long, to direct the illumination in 32 different directions. 32 monochromatic rendered views were also generated as input to stereogram computation, each of dimension $W=256$ by $H=144$ pixels. To generate each hogel, corresponding pixels from each of the 32 parallax views (*i.e.* the first pixel from the first row in each image) were used to weight the set of basis fringes. The entire hologram had 144 hologram lines, each containing 256 hogels given by

$$\vec{h}_w = \sum_{i=1}^{32} p_{i,w} \vec{b}_i \quad (1 \leq w \leq 256) \quad 4.2.14$$

where the $p_{i,w}$ are the pixel weights. Given the astigmatic optical projection of the component parallax views and the HPO nature of the display, correctly-captured parallax views would employ an orthographic projection in the horizontal direction, and a perspective projection, matching the hologram viewing geometry¹⁵⁸, in the vertical direction (Figure 4.10). However, standard perspective renderings were used in this work, resulting in a geometrically distorted output.

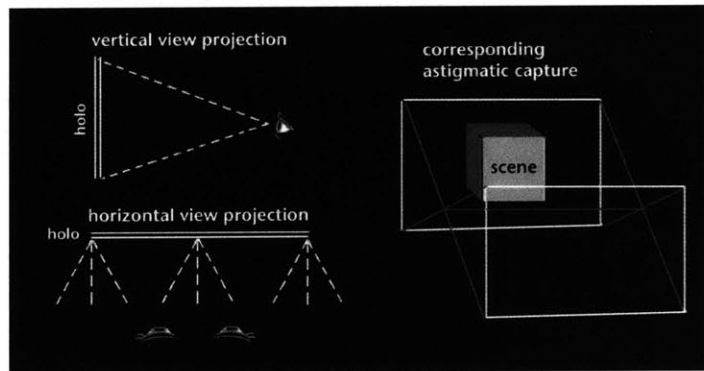


Figure 4.10 Astigmatic capture for one computed stereogram format

Lucente eventually considered this stereogram computing method a specific case of a more general approach called *diffraction-specific computing*. In essence, the diffraction-specific approach considers the hologram not purely as a sampled version of many continuous analytic field contributions, but rather as independent hologram-elements¹⁵⁶, each computed to project a specific piece of a scene in a finite number of directions (Figure 4.11). This representation supported experiments to encode the independent hologram-elements in various ways, to sample and compress them spectrally, as well as spatially.

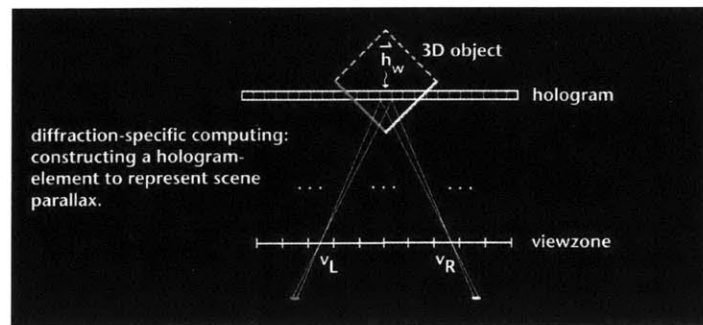


Figure 4.11 A "diffraction-specific" hologram element encodes sampled scene parallax

The computation of these images was indeed very rapid — well-less than 10 seconds per hologram frame, varying for different implementations. However, computed stereograms do not always compare favorably to their fully-computed cousins. Diffraction-specific holograms of higher-resolution light-fields may hold more promise, but thus far have only been demonstrated to represent a single point in space.

In general, images reconstructed from stereograms inherit the potentially impoverished horizontal resolution of the 2D parallax views. However, since we normally assume that providing spatial resolution below the limit of our visual acuity is unnecessary, this compromise is not unacceptable in principle. A more troublesome issue remains the need for an appropriate gamma correction for a given electro-holographic display. Currently, the tonal dynamic range within the reconstructed image does not likely match that of the input images, and smoothly-shaded objects often appear quite flatly rendered in the final reconstructed view. Various compression and encoding schemes have been presented to speed computation, but all contribute artifacts that further degrade image quality.

Finally, since phase is unconstrained in the modeling of the basis fringes, the resulting phase discontinuities between parallax views can cause the image to jitter slightly from view to view, and thus appear less spatially stable and somewhat "flattened". In an HPO system, this effect is worse if an observer is not positioned at the correct viewing distance. However, issues of image quality aside, this work clearly showed that significant decreases in computation time could be achieved, and that electro-holography could someday serve as a viable choice for interactive display.

4.3 Realism revisited: Incrementally- computed holograms

4.3.1 Motivation

Mostly as a result of our experiments in combining haptic feedback with electro-holographic images^{159,160}, the need to deliver images looking "so real that you might touch them" renewed our interest in revisiting realistic hologram modeling. In our first holo-haptic experiment (the *Lathe* system described in a following chapter), a holographic image of a hemi-cylinder was positioned to coincide with a

computational haptic model; the holo-haptic simulation could be "carved" by a user who felt the model changing (using a force feedback device), and saw the hologram being updated. In *Lathe*, a holographic stereogram was generated first, from 32 rendered views of a hemi-cylinder. The resulting image, though fast to produce and offering the requisite binocular and motion parallax, had image quality too poor, for reasons mentioned above, to address the rather unforgiving requirements of the multi-modal simulation. *Lathe* consequently employed interference-modeled wireframe images which, although not photo-realistic *per se*, rendered the cylinder's geometry with sharp lines, and provided output as metrically-correct as the *second generation* MIT holovideo system would permit¹⁶¹.

Lathe was able to achieve its update speed by assembling an image of the carved stock from a finite set of pre-computed holograms. Since the carving was constrained to be rotationally-symmetric, a variable image of a rotational solid could be constructed by using appropriate lines from each of a set varying-diameter hemi-cylinder-holograms. Carving is usually carried out slowly and in one region at a time, so only a few hologram lines required changing per each visual update.

In order to *arbitrarily* update an underlying model geometry and reflect these changes in the final hologram at interactive rates, the *incremental computing* algorithm was developed. Like [153,137], this method uses a pre-computed table of hologram elements, or *elemental fringes*, which are associated with model primitives which we call *holo-points*. The depth of each holo-point in our model is used to determine the elemental fringe appropriate to represent it. As in earlier work, the final hologram is computed by combining all elemental fringes needed to reconstruct the images of all holo-points at their prescribed spatial locations.

4.3.2 Algorithm basics

For example, the left of Figure 4.12(a) shows two orthogonal views of a set of holo-points arrayed in *x* and *y* to form a square. Since these points all share the same depth value, the same elemental fringe is used to represent each in the hologram. The associated elemental fringe is indicated in the fringe table at the right of the figure, and the hologram that results is shown in the middle. Elemental fringes in Figure 4.12(a) were computed for a very long wavelength ($\lambda=2$); thus, the resulting hologram cannot be reconstructed optically, but serves here to illustrate its assembly from elemental fringes, and their association with holo-points. A diagram of its reconstruction is shown in 4.12(b).

Using the incremental computing method, the image of any holo-point can also be *erased* from the hologram by subtracting the elemental fringe that represents it, and can be *re-imaged* by adding the fringe back again. As long as the reference wave used during computation and the illumination wave are collimated, other simple operations are possible, too. For instance, translation of an elemental fringe on the hologram results in translation of the holo-point it reconstructs. Reversing the order of samples of any elemental fringe produces an image of the holo-point at its *negative* depth value. Finally, to represent a holo-point changing in depth, the point's original elemental fringe is subtracted from the hologram, and a new elemental fringe which

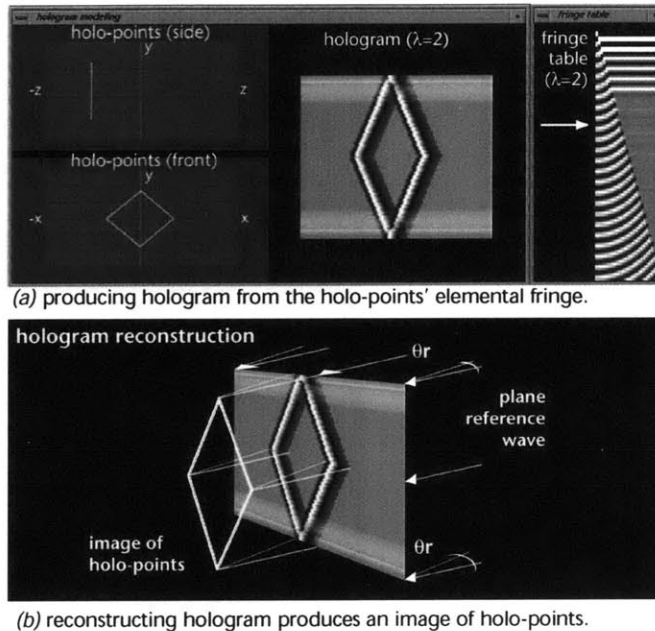
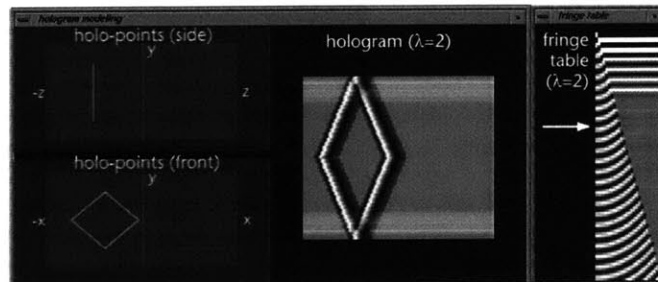


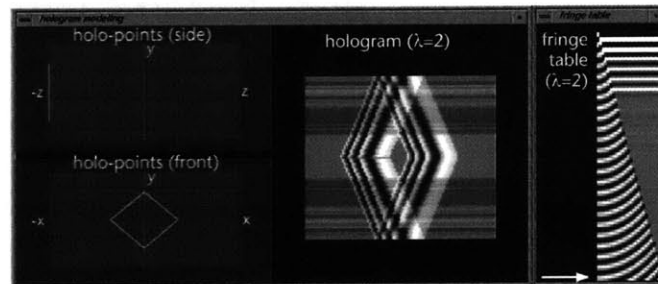
Figure 4.12 Assigning elemental fringes to holo-points and reconstructing the hologram

reconstructs the holo-point at a new depth is chosen and added to the hologram. This method allows a hologram to be *minimally updated* to reflect underlying geometry changes in the scene. Only the parts of the hologram affected by scene changes are modified, and modifications are effected by simple operations. The amount of computational effort required at each update is roughly related to the number of scene primitives that change and the area of the hologram they effect.

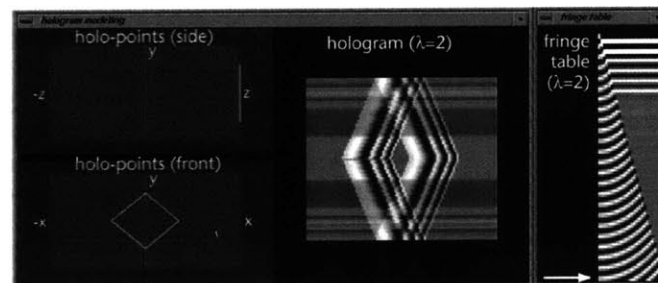
Examples of these operations are shown below in Figure 4.13. At the left of 4.13(a), note that holo-points have been translated in the -x direction. Since the depth of these points remains the same as shown above in Figure 4.12, the elemental fringe representing all points is also the same. However, these fringes have been translated in the final hologram, in order to translate the reconstructed images of the holo-points. Figure 4.13(b) shows the translation of the holo-points in depth, and a new elemental-fringe assigned to represent them in the final hologram. Figure 4.13(c) shows the same elemental fringe with the order of its samples reversed, used to represent the holo-points at their opposite depth value. Basically, if a model can be represented by holo-points, then a hologram of this model can be constructed using such elemental fringes as "building-blocks". Further, model changes can be expressed in the hologram by *independently manipulating* elemental fringes associated with changed holo-points, rather than by recomputing the entire hologram.



(a) translating holo-points translates elemental fringes in the final hologram.



(b) changing holo-point depth requires a new elemental fringe to be assigned.



(c) flipping a holo-point's depth reverses sample order of its elemental-fringe.

Figure 4.13 How changes in holo-points correspond to changes in elemental fringes

4.3.3 Computing elemental fringes

To generate the set of elemental fringes, we use the bipolar intensity method mentioned previously to compute the interference of a point source with a plane reference wave, and table the results for points located at 100 depth increments — every 0.5mm for $x = 0, y = 0, z = i$ for $0 < i \leq 50mm$. The elemental fringes f_i are described by

$$f_i(x) = \frac{a_i}{r_i(x)} \cos \frac{2\pi}{\lambda} (r_i(x) - x \sin \theta_{ref} - \phi_{random_i})$$

4.3.1

where $\lambda = 633 \times 10^{-6} \text{mm}$, a_i is holo-point $_i$'s amplitude, r_i is the radius of curvature of a its spherical wave given in equation 4.2.11, ϕ_{random} is a random initial phase assigned to a spherical wave (between 0 and 2π), and θ_{ref} describes reference wave angle.

No elemental fringe spans the entire hologram. Each is computed over some span of hologram samples bounded by *a.*) the location where interference produces a spatial frequency equal to zero, and *b.*) the location where the spatial frequency reaches a maximum permissible value. Satisfying the Nyquist Sampling Theorem requires that our sampling frequency f_s be more than twice the maximum spatial frequency f_{max} contained in a fringe. A fringe's spatial frequency is related to the diffraction angle θ_{out} by

$$f = \frac{\sin(\theta_{out}) - \sin(\theta_{ill})}{\lambda} \quad 4.3.2$$

To amply avoid aliasing, we compute f_{max} to be slightly less than half the sampling frequency of our given display as follows

$$f_{max} = \frac{1}{2.2 \times w_s} = 794.37 \frac{cy}{mm} \quad 4.3.3$$

where $w_s = 57.2 \times 10^{-5} \text{mm}$ is the sample spacing in the holovideo system. Substituting this result into equation 4.3.2 and using holovideo's illumination angle $\theta_{ill} = 15^\circ$ and a wavelength of $633 \times 10^{-6} \text{mm}$ shows that we can compute elemental fringes for a maximum diffraction angle of 14.1° .

To compute for the display, we let $\theta_{ref} = \theta_{ill} = 15^\circ$. Then the number of samples in each elemental fringe N_i , is determined by the geometrical relationship

$$N_i = \frac{z_i(\tan \theta_{out} + \tan \theta_{ref})}{w_s}, \quad 50 \text{mm} \leq i < 0 \text{mm} \quad 4.3.4$$

where z_i is the holo-point's depth. Consequently, elemental fringes that reconstruct points farther from the hologram require more samples than those which reconstruct closer points. In our implementation, we let the maximum diffracted angle θ_{out} be 12° , so that the largest tabled fringe (for $z = \pm 50 \text{mm}$) contains 42002 samples, and the smallest (for $z = \pm 0.5 \text{mm}$) contains 420 samples. The general geometry for modeling these fringes is illustrated in Figure 4.14. Each pre-computed elemental fringe, when scaled by an amplitude value, normalized, and sent to the display, is capable of imaging a holo-point at location $x = 0, y = 0, z_i$, when illuminated with holovideo's reconstructing plane wave.

Since all elemental fringes have finite extent, effectively being clipped by a rectangular aperture, they are subsequently filtered to reduce diffractive artifacts generated at

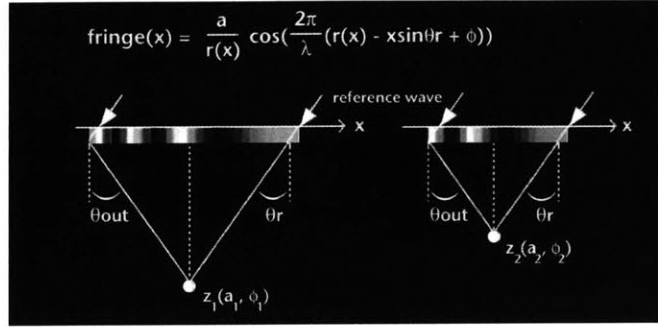


Figure 4.14 Generating elemental fringes

fringe boundaries. Especially in wireframe images, these unwanted artifacts resemble faint but distracting web-like structures woven around image features. In our implementation, the N_i samples of each elemental fringe are multiplied by a Blackman Window¹⁶⁶, given by

$$w_i(n) = 0.42 + 0.5 \cos\left(\frac{2\pi n}{N_i}\right) + 0.08 \cos\left(\frac{4\pi n}{N_i}\right) \text{ for } |n| \leq \frac{N_i}{2} \quad 4.3.5$$

This set of pre-computed, filtered fringes can image points between $\pm 50\text{mm}$ at half-mm increments, into the hologram image volume. Using these fringes in conjunction with an HPO display effectively quantizes the addressability of the image volume, most coarsely in y , less so in z , and much more finely in x . Of course, it is possible to compute a fringe table to represent a much larger depth range and a smaller depth increment; reasonable choices for these parameters would consider limitations of a particular display as well as the stereo acuity of an average viewer.

In our specific case, the depth range of 100mm for the MIT *second generation* holovideo display is appropriate. To determine a reasonable depth increment, we first consider a viewer's stereo acuity to be on the order of one arcsecond. Then using this angular separation at the eye, we determine the millimeter separation of two just-resolvable holo-points in the image volume closest to a viewer's eyes (say, 550mm away, in the MIT system which has a viewing distance $d=600\text{mm}$). Then, for an interocular distance $IOD = 60\text{mm}$, the minimum point-separation distance Δ can be approximated by the geometric relationship

$$\frac{1}{360^\circ} = \text{atan}\left(\frac{(550 + \Delta)\text{mm}}{(IOD/2)\text{mm}}\right) - \text{atan}\left(\frac{550\text{mm}}{(IOD/2)\text{mm}}\right)$$

Solving for Δ , we have

$$\Delta = \frac{IOD}{2} \tan\left(\frac{1}{360^\circ} + \text{atan}\left(\frac{550\text{mm}}{(IOD)/2}\right)\right) - 550\text{mm} = 0.491\text{mm}$$

We have chosen our 0.5mm depth resolution for a 600mm viewing distance to be close to the value computed above. A re-sampled table of 100 filtered fringes computed to these specifications is shown in Figure 4.15. In our implementation,

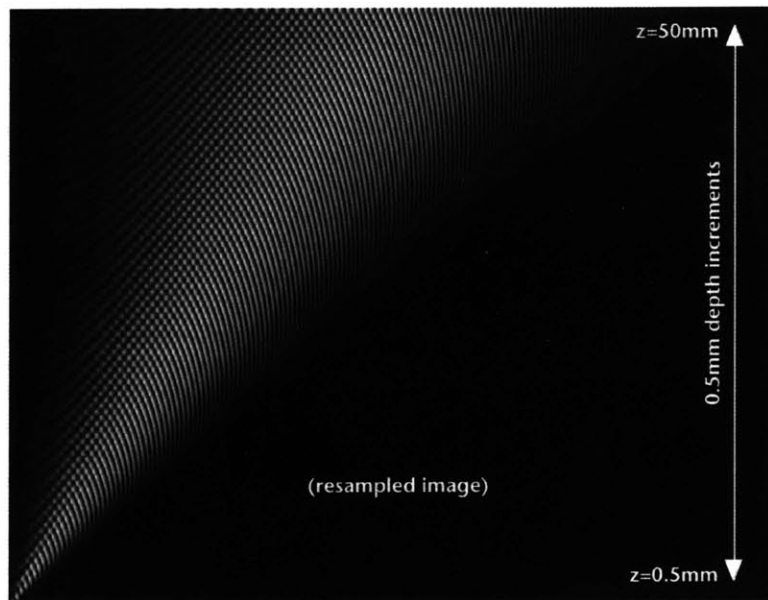


Figure 4.15 100 pre-computed fringes at 0.5mm increments

these fringes are represented as integers with 16-bit resolution. This way, as many as 65535 elemental fringes can be accurately accumulated into the same span of samples in a 32-bit "working" hologram, if necessary, to image as many points.

4.3.4 Populating a model with holo-points: randomizing initial phase or position

To create an object suitable for computation, we use the *Occfilt* system developed by Underkoffler^{136,151} which takes a polygonal description of an object, carefully populates it with point sources, and can also perform occlusion processing and shading calculations. This system incorporates a polygon-based renderer called *RenderMatic*¹⁶², developed at the MIT Media Laboratory, permitting the scene to be more conventionally "previewed" while holo-points are being generated. *Occfilt* can be used to generate wireframe models by populating polygon edges with holo-points, or

to approximate surfaces with a dense veneer of holo-points; for now, we use *Occfilt* only to generate the former.

Occfilt uses Bresenham's algorithm¹⁶³ to populate polygon edges with holo-points and thus generates a data set optimized for a *particular model orientation*. A polygon edge slanted at an angle less than 45 degrees from the vertical is represented by one holo-point in every hololine; polygon edges slanted greater than 45 degrees from the vertical are represented by multiple holo-points in each hololine, depending on the edge's slope. Thus, while the initial set of holo-points is generated with the characteristics of the holovideo display in mind, subsequent rotations of these holo-points are not guaranteed to look as good. Since the model is only populated once, some incrementally-computed transformations result in varied appearance of the final image.

In *Occfilt*, the desired density of holo-points along an edge (or over a surface) can be specified. In general, linear densities from five to ten points per millimeter simulate continuity of line and surface; with a greater inter-point spacing, individual holo-points within a hololine begin to be discernible. In any case, however, the resulting regular grid of coherent in-phase point sources will interfere with one another causing high contrast fringes to appear in the output.

Traditional fully-computed approaches randomize the phase of each holo-point to diminish this artifact of coherent illumination. In our approach however, though elemental fringes are generated with point sources having random phases, holo-points at the *same* depth (which are represented by the same elemental fringe) will share the same initial phase. To prevent high-contrast fringes in the output from these same-depth holo-points, all holo-points are randomly spatially jittered by a very small amount (a uniform deviate between 0.01 and 0.1mm) to break the regularity imposed by the populating algorithm. This spatial perturbation sufficiently diminishes the contrast of any visible inter-point interference at the output. The one-time spatially-jittered set of holo-points is used as input to all subsequent incremental computing operations.

4.3.5 Incremental modeling

As elements of the scene are transformed, lighting is adjusted, or as the geometry or surface properties of objects within a scene are modified, the previous state of each affected holo-point is recorded and new, updated values are computed. These state variables include:

- actual x,y,z position,
- quantized y and z bins,
- normal vector (not yet implemented),
- amplitude,
- the hololine it belongs to,
- the ID number of its associated elemental fringe,
- the number of samples in that fringe to use,
- visibility / occlusion spans (not yet implemented),

- a pointer marking both the fringe sample to start on, and
- a pointer marking the hologram sample where contribution begins.

During every subsequent model update, if a holo-point's new and previous position or amplitude differ, the point is marked for change in the hologram and its new state is recorded.

If a holo-point's new location is outside of the available display volume, or if it occupies a location between $\pm 0.5mm$ (where its interference computation becomes mathematically undefined), the point is temporarily marked *invisible* until it is transformed back into view. Fringes for invisible holo-points will be subtracted from the working hologram, but not added back in until the point comes back into view. This constraint on visibility effectively clips the object against the bounds of the display volume, and forms a kind of "dead zone" straddling the hologram plane through which image points disappear as they enter, and reappear as they leave.

We refer to this entire collection of operations as *incremental modeling*. If occlusion processing and shading calculations were performed in the implementation, the computation of visibility changes and surface shading would also be included here. Finally, the updated holo-points are sorted by new y-value and reassigned to appropriate hologram lines. If a holo-point leaves a hololine and is assigned to a new one, both old and new hololines are marked for update. Holo-points are then matched to their nearest 0.5mm depth increment and the corresponding elemental fringe.

In all of our implementations of this work, incremental modeling is performed on a dual-processor SGI Onyx workstation, having a serial, SCSI, and HIPPI (High-Performance Parallel Interface) connection to the Cheops Image Processing System^{154,165}, which provides both computation and the framebuffer for the *second generation* MIT holovideo system.

4.3.6 Indexing: determining elemental fringe contributions to the hologram

Once incremental modeling is accomplished, the next steps are to determine for each visible holo-point, which samples in that fringe to use, and which samples in the hologram that fringe should contribute to. First, to determine how to render a holo-point with an elemental fringe in the hologram, the holo-point's depth relative to the $z=0$ plane is used to find whether the elemental fringe samples should be displayed forward or in reversed order. Next, each holo-point's position is used to determine the appropriate sample boundaries of their associated elemental fringe, and finally, the hologram samples the fringe will contribute to. If a holo-point's x-y value is near the center of the hologram, all of the samples in its associated elemental fringe will contribute to the final hologram. However, elemental fringes may be clipped by hologram edges when holo-point locations are offset from the hologram center, as a result of the reconstruction geometry as shown in Figure 4.16.

4.3.7 Incremental updating

In our implementation, the working hologram has a 32-bit integer representation, and is comprised of the superposition of all contributing holo-points' elemental fringes.

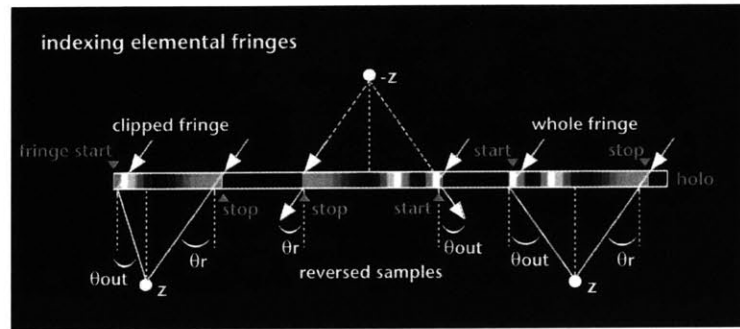


Figure 4.16 Indexing elemental fringes

Before any updates, an initial working hologram is computed by summing the elemental fringes assigned to all holo-points. This initial working hologram is kept in state, and is subsequently updated incrementally. During *incremental update*, each marked holo-point has its old fringe (scaled by its old amplitude) subtracted from the working hologram, and its new fringe (scaled by its new amplitude) added. Both operations are performed on the appropriate working hologram samples, and using the appropriate fringe samples.

Each time a working hologram line is operated upon, the beginning and end of the span of updated samples in that line is noted, and all updated lines are marked for renormalization. After all elemental fringe operations have been completed, the maximum and minimum values within the updated span on each hololine is determined. If this value matches the line's previous max and min values, then only the samples within the updated span are renormalized; otherwise, the entire line is renormalized to match the bit-depth of the framebuffer. In our implementation, the updated hololine samples are mapped to values between 0 and 255, and substituted into the 8-bit "display" hologram.

Next, in our system, the display hologram must be "conditioned" to suit holovideo's scan convention and the difference between the SGI's and Cheops' byte-ordering. Characteristics of the *second generation* holovideo system's electro-optical architecture and its boustrophedonic horizontal scan convention^{161,164} require each hologram line within a set of 18 to be written to the framebuffer in sample-reversed order from the previous set of 18 lines. This line-flipping and byte-swapping is the last computation performed before the entire 36MB hologram is transferred to Cheops via HIPPI (100MB/s) for redisplay.

4.3.8 Updating the holographic display

The Cheops Imaging System¹⁶⁵, developed at the MIT Media Laboratory, is a modular video processing and display system designed for real-time video coding experiments and interactive image display applications. This system is used for computation and

storage of holograms and holographic stereograms, and as a framebuffer for the *second generation* holovideo display.

Cheops' modular hardware architecture includes four basic module types connected by two types of buses, a *Global* bus and two *Nile* buses. The 32-bit Global bus is used for supervisory communication and for small data transfers between modules. The 24-bit Nile bus (120 MB/sec) is intended for the high-speed transfer of large blocks of image data in three channels (8 bits of each red, green, and blue). To take advantage of this built-in 3-channel structure on Cheops, we store and transfer our hologram lines in groups of three called *triplets*.

Our particular Cheops has a HIPPI card, a memory card (M1), a processor card (P2), and 6 output cards (as well as some special-purpose hardware¹⁵⁴) connected to each other by the Nile and Global data pathways. The incremental computation described above could be performed rapidly on the P2, according to instructions sent via SCSI from the host SGI computer, for instance. However, the P2 has only 32 MB of memory, which is insufficient to locally store all the data required for this processing (an 18MB elemental fringe table, a 144MB integer working hologram, and a 36MB display hologram). Only the fringe table fits on the P2, while the working and display holograms can be stored on the M1, and retrieved as necessary during processing. However, transferring integer hololines via the Global bus is slow, and the Nile data pathway truncates the integer working hologram data to 8 bits during transfer. Thus, while a redesign of the hologram data representation could allow Cheops to be a good engine for computation, our current design is not well-suited for this particular hardware architecture. Currently, we use Cheops *only* as a display server and holovideo framebuffer.

In our system, a service (*holoPut*) running on Cheops monitors the HIPPI device for incoming data. Once *holoPut* receives the incoming hologram, it sends the bytes via the Nile bus, directly to the output cards for display. The transfer and redisplay of the hologram takes 0.6 seconds; this flat cost per update may be further diminished by transferring only *changed* lines via HIPPI to the output cards, a modification to be implemented in future work. The entire pipeline is shown in Figure 4.17.

4.3.9 Discussion of algorithm performance

Our goal was to design a method for computational holography that provided interactive display while opening avenues to improve the visual quality of the final image. Thus, to address these different aspects of algorithm performance, we will first discuss the computational speed of the incremental computing method, and compare it with the performance of other computational methods. To examine computational speed, we will consider the algorithm's three stages described previously: *incremental modeling*, *indexing*, and *incremental updating*, focusing most of our effort on the last. In the next section, we will discuss the visual quality of the final images and compare them to those generated by other methods.

Incremental modeling: Within the scope of incremental modeling, elements of the scene are transformed, lighting may be adjusted, and/or the geometry or surface

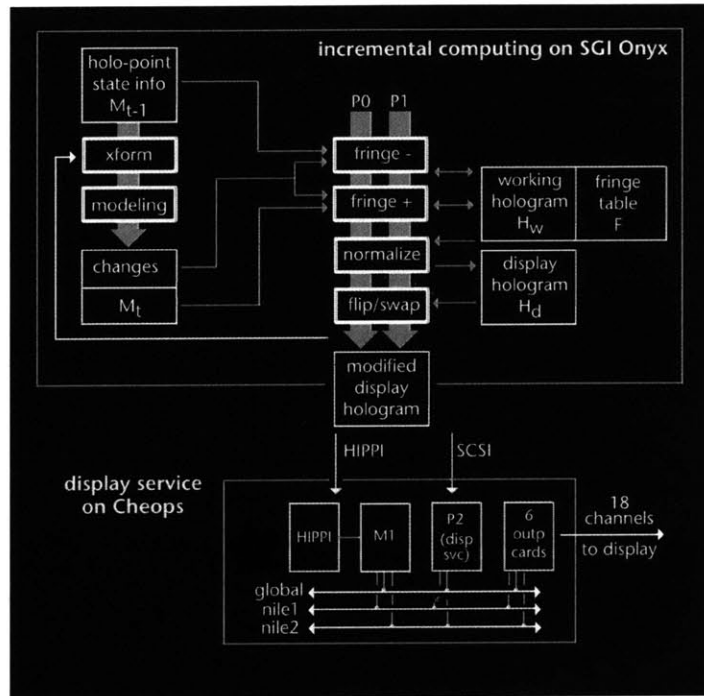


Figure 4.17 Incremental computing pipeline

properties of objects within a scene are modified. Since scene modeling in any implementation may vary from rudimentary to quite elaborate, no performance measures will be included here. Instead, we will briefly address the general computational effort required for different types of modeling changes, and among different hologram computation methods.

Modeling changes applied to the scene will require quite variable computational effort. For instance, if scene lighting, the illumination model, or object color changes, then new holo-point amplitudes will have to be computed using equation 4.2.9 (or a similar illumination model). If the entire scene is transformed, then all holo-point position and normal vectors will undergo the matrix multiply, *and* new holo-point amplitudes must be computed. Further, changed holo-points must be quantized in depth, assigned new fringes, and reassigned to hololines. If performed on *all* holo-points, these kinds of modeling operations would require the same computation time in *any* fully-computed approach, and can become costly in scenes densely populated with holo-points. Of course, the computational stereogram approach most rapidly accomplishes this step using graphics hardware to render more conventional geometric primitives into a set of 2D images.

If, however, only one object in the scene, or part of an object is updated, then the

incremental computing method provides a means for updating *only* the affected holo-points. An exception holds for occlusion processing, in which visibility tests may need to consider otherwise unchanged geometry too. Other computational methods consider all holo-points, whether the entire scene has changed, or just one small piece of it. Thus, compared to other methods, the incremental approach provides computational economy in the scene-modeling stage if *only part* of the scene is updated. If the entire scene changes, the incremental approach requires computational effort equal to other fully-computed approaches in the scene-modeling stage.

Indexing: The small computational cost of indexing is fairly uniform for holo-points at all depths (about 0.035msec per point), but slightly higher for points at negative depths. This modest increase stems from the need to index the elemental fringes in reverse order. As a fairly flat cost per holo-point, indexing time increases linearly with the number of updated holo-points. So, in indexing too, the incremental approach provides computational savings if only part of the scene needs to be updated, and matches a straight table-lookup approach in computational effort if the entire scene changes.

Incremental updating: In this stage, fringes may be scaled by an old amplitude and subtracted from the working hologram, others may be scaled by a new amplitude and added, and affected lines will be renormalized. Again, depending on the type of modeling changes that occur, the amount and exact flavor of incremental updating will vary. For instance, if the scene lighting or an object's surface property changes, then the hologram samples representing each point's contributing elemental fringe need to be scaled by a_{new}/a_{old} to reflect the update; no fringe additions or subtractions are necessary. Then, all hologram lines (but for ones containing only zeros) will require renormalization.

If the entire scene is transformed, then all holo-points' old amplitude-scaled elemental fringes must be subtracted, new amplitude-scaled fringes must be added, and every non-zero hologram line must be renormalized. In this most computationally-expensive case, computing a fresh initial working hologram may be faster than subtracting old fringes and adding the new (depending on normalization requirements).

The speed of these computations is variable, being affected by the depth of holo-points, the number of hololines which require renormalization, and the length of the spans of samples on these lines. The plots in Figures 4.18 and 4.19 show the time cost of adding and subtracting an elemental fringe as a function of its depth in millimeters from the hologram plane.

From these plots, we can see that adding an elemental fringe for a holo-point at 50mm takes approximately 4.45msec, and adding a fringe for a holo-point at 10mm takes less than one msec on average. For both adding and subtracting elemental fringes, the time cost is a linear function of the number of samples involved in the operation.

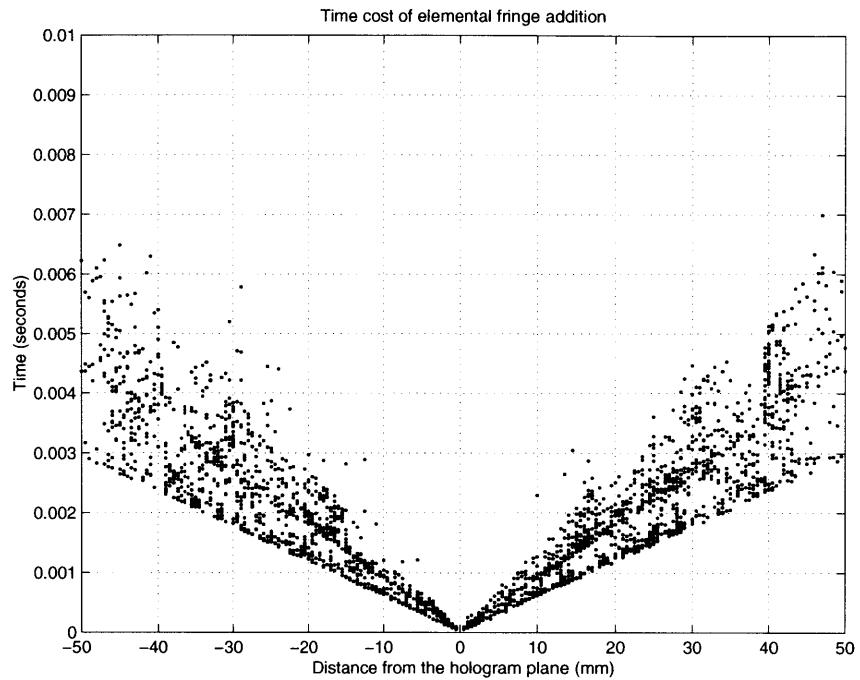


Figure 4.18 Cost of adding an elemental fringe as a function of holo-point depth.

Normalization issues: The question of how best to normalize a hologram with limited bit-depth remains open to investigation. Optical holography may be used as an analogy, wherein an optimal modulation depth for an emulsion is achieved by controlling the hologram's exposure time. For the entire hologram, the calculated exposure depends on the part of the emulsion subject to the greatest object beam intensity. Analogously, we could compute a minimum and maximum sample value over the entire computed hologram at each update and use these boundary values to remap all other samples to 8 bits.

However, since performing the required value comparison at each sample is costly, we choose to normalize each hologram line independently. This way, at each update, only affected hologram lines are renormalized. Furthermore, by noting the spans of changed samples on a hololine at each update and comparing the span's max and min sample values with those for the entire hololine in the last iteration, it can be determined when *only that span* requires normalization. In our implementation, when the max and min sample values of the span do *not* match the hologram line's values, the entire line is renormalized with the greater max and lesser min. This computation is not always correct, and errs on the side of under modulation rather than clipping against the available range. Furthermore, in our approach, one lone point on one

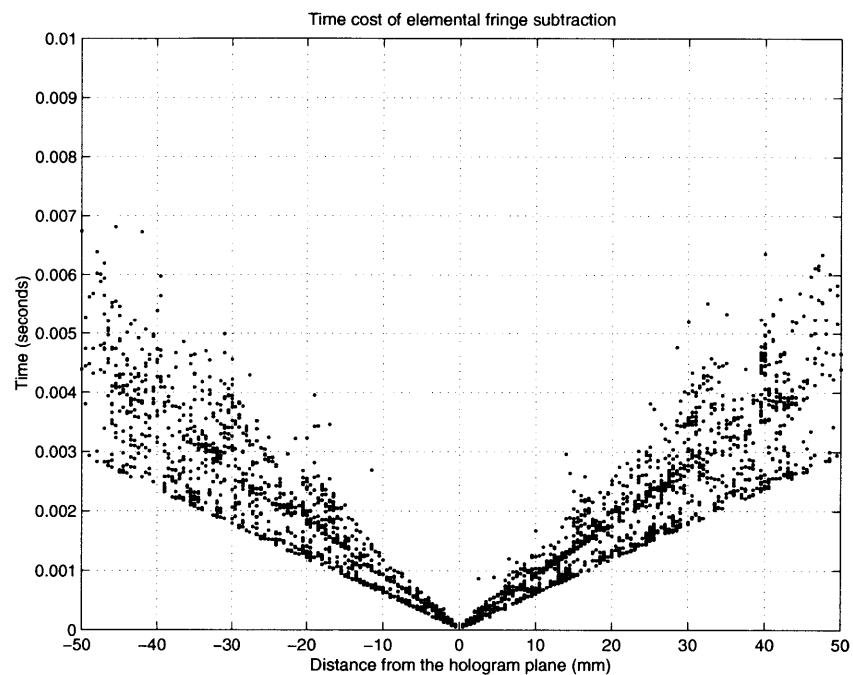


Figure 4.19 Cost of subtracting an elemental fringe as a function of holo-point depth

hololine will appear brighter than each of many closely-spaced points whose fringes overlap on another hololine. Consequently, changes in holographic image brightness are sometimes apparent as the underlying model is transformed in space, and a vertical line may appear brighter than a horizontal line.

Whether normalizing with a global hologram min and max, or normalizing each line independently, considering the shallow modulation depth (only 256 greyscale *fringe* values), the still unqualified gamma correction necessary for our particular display, the uneven inter-line display characteristics, and the presence of coherent illumination artifacts, the number of discernible grey levels in the output image is unknown and likely much less than 256. Quantifying greyscale resolution of the output and determining appropriate the gamma correction are certainly required steps toward improving image quality on the holovideo display. However, our work thus far has focused only on increasing the speed of normalization with this line-by-line, or span-by-span approach; the other issues remain for future work.

Figure 4.20 shows the time cost of normalizing a span of varying sample-length. In cases when an entire hololine must be renormalized, this operation takes roughly between 27 and 53 msec, depending on how many zeros the line contains. In our implementation, the greatest time cost is in the value comparisons to compute new

min and max hologram values. Once those values and the number of hololine samples to be processed is determined, the normalization is quickly performed using bit-shift operations; as implemented, this approximation may further under modulate the fringes in error by a factor of two.

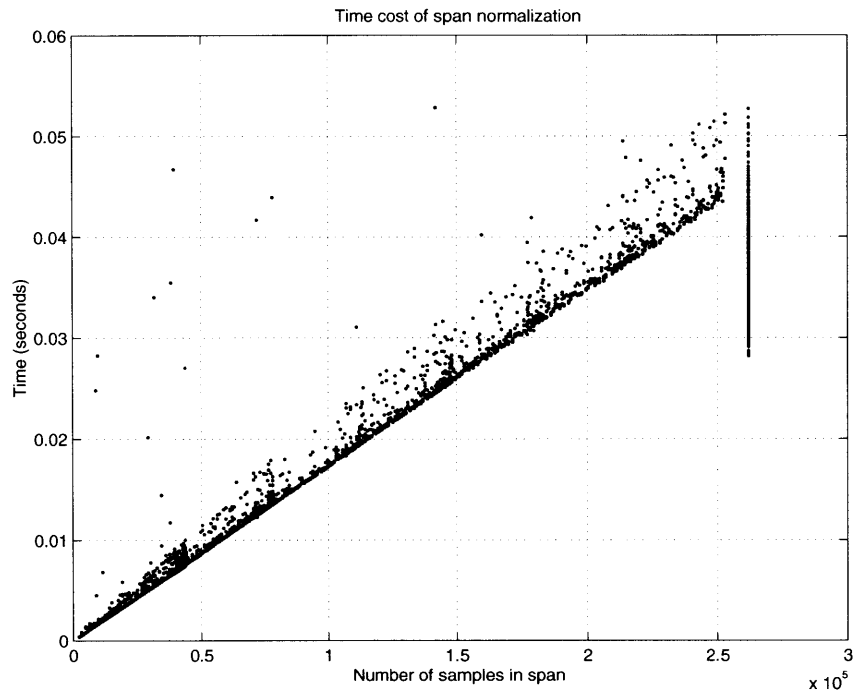


Figure 4.20 Time cost of normalizing a span of samples of varying length

All told, incremental updating has variable cost, depending on the depth of the changing holo-points and, most critically, on the number of hologram lines which require renormalizing. Since finding hologram min and max values during normalization is the most expensive operation required in this method, incremental computing may still produce a hologram more rapidly than recomputing a new initial working hologram, especially if the image only spans some hologram lines. Similarly, this new approach may be faster than producing computational stereograms (without hardware assistance) since they will require global renormalization at every update.

Figure 4.21 shows minimum, maximum and average times required to incrementally compute (using two 250MHz processors) a hologram of holo-points transformed randomly in 3D space. (The plot does not indicate the one-time 12 second cost of precomputing the table of elemental fringes, the variable cost of computing the initial working hologram, or the 0.6 second cost per update for transferring the 36MB

hologram to the display.) Looking at the data describing the *maximum* compute time, one can observe an initial sharp linear increase in time with increasing number of holo-points, as the renormalization of more hologram lines is required. Once all (144) hologram lines are involved, a slower linear increase in compute time is evident, as y- and z-quantization for more holo-points is required, and as integer operations on more elemental-fringes and more hologram samples are necessary. The data describing the *minimum* compute time reflects less modeling and update costs for the set of holo-points, normalization of only one or a few hologram lines, and operations on smaller spans of samples within each hololine.

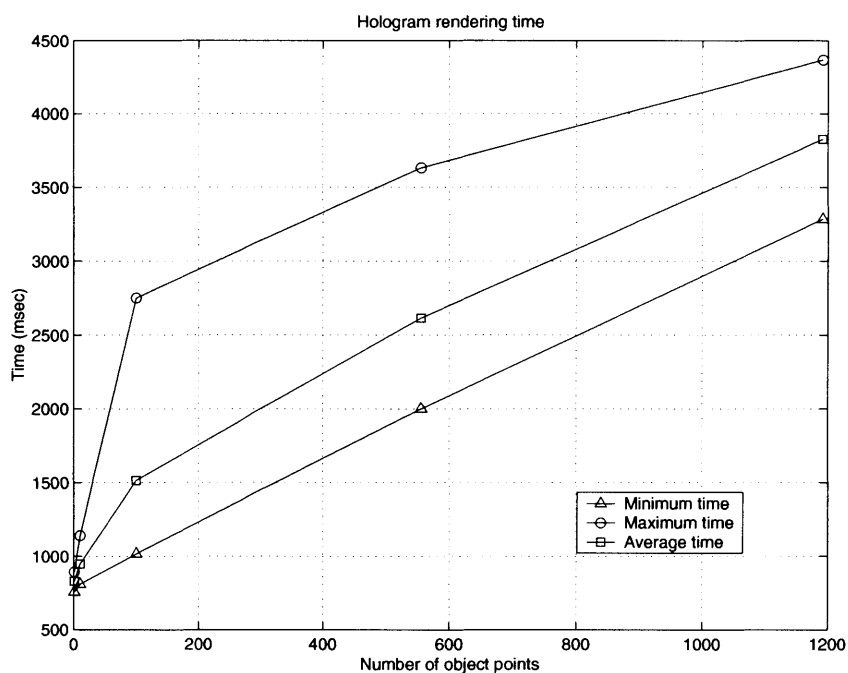


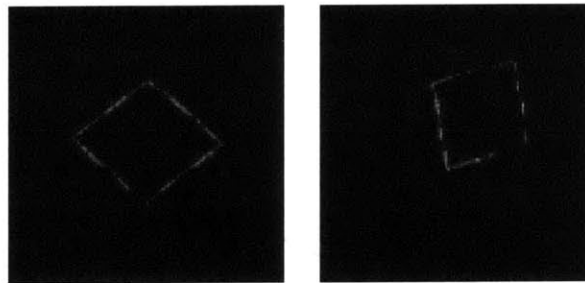
Figure 4.21 Time required to compute holograms of varying numbers of points (2 processors)

In summary, the incremental computing method offers the most rapid generation of fully-computed holographic images when partial scene changes require recomputation. When the entire scene changes, this method can still perform a fresh table-lookup summation to compute a new initial working hologram usable for subsequent incremental changes, and only renormalize hololines that require it. To generate holograms using this method requires a one-time computational cost to pre-compute the elemental fringe table and initial working hologram; then update time depends on the number of hologram lines that change, the number of points that change, and the depth of those points. Thus, the incremental computing method provides a way to loosely relate computational effort to the area of the hologram

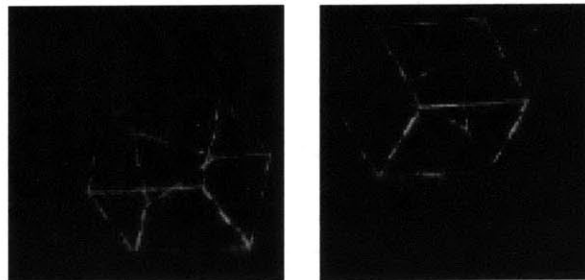
affected by a change in the scene and the scale of that change.

4.3.10 Visual appearance of images

Holographic images of a square populated with 200 holo-points, transformed in three dimensions are shown in Figure 4.22 (updated and displayed on holovideo at 1 frame per 2.5 seconds). Holographic images of a cut cube populated with 556 holo-points, similarly transformed, updated and displayed at 1 frame per 3 seconds are also shown. The modeling for these (and all images produced so far) include no occlusion processing or shading calculations; all holo-point amplitudes are equal to unity, so no scaling of subtracted or added elemental fringes is necessary.



(a) 2D square transformed in 3D space (0.4 frames/sec)



(b) 3D cut cube transformed in 3D space (0.33 frames/sec)

Figure 4.22 Holographic images rendered with the incremental computing method

As a result, the current pictorial quality of the final wireframe images is more graphic than photorealistic, though more sophisticated shading methods can certainly be incorporated. Since this worked was developed with the intent of facilitating arbitrary visual updates in a holo-haptic system, elaborate visual rendering must currently be traded for speed to minimize intermodality disparities; for our purposes, wireframe images are acceptable. Outside of the holo-haptic context, however, techniques supporting the most realistic rendering possible should be investigated.

As regards other computational shortcomings, the previously mentioned problems

rooted in normalization are not strongly evident in the figure, since no shown polygon edge is quite horizontal. However, the effect line-by-line normalization can have on a holo-point's apparent brightness can be quite distracting.

Some characteristics of the holovideo system itself diminish the visual quality of the images. Perhaps most troubling is the variation in gain among modulator channels, which causes some hololines to be quite bright, and some to be very dim — this effect is most clearly seen in Figure 4.22(a). Also, as is also evident in the figure, while horizontal and vertical image lines are more sharply displayed, diagonal lines can exhibit discontinuous thickness and slope due to slight optical misalignment in the holovideo system. All images are monochromatic and exhibit speckle.

Display limitations aside, with careful computation and the addition of more sophisticated techniques in the modeling step, the pictorial quality of these images will certainly improve (though computation time will also increase). For visual comparison, a wireframe computational stereogram, a complex interference modeled hologram, and an incrementally-computed hologram were generated from the same geometry (Figure 4.23). From the figure, it can be seen that incremental computation generates an image very similar to traditional complex interference modeling. This result allows us to pursue more realistic visual rendering with incremental computation without the overwhelming computational expense that usually accompanies more traditional physical modeling.

Both the complex interference modeling and incremental computation methods produce an image with sharper lines than the stereogram image exhibits; however, higher pixel resolution in the set of 2D views would improve this result. Parallax, which is, of course, not evident from the photographs, is more convincing in the fully- and incrementally-computed holograms, though the reasons for this are not entirely qualified yet. This casual (and perhaps first) side-by-side look at a stereogram and fully-computed holograms, generated from the same data set and also modeled similarly, shows stereograms to compare surprisingly well.

The fully-computed approaches and the "diffraction-specific" stereogram approaches currently provide different *flavors* of display, though both form images by diffractive means. Likely, both the visual and the physical distinction between these two computational approaches will diminish with the combination of high-resolution light-field rendering, use of a greater number of component views and more refined pre-computed diffractive elements. An electro-holographic display with much higher visual quality will help us to better compare images produced with the two techniques; currently, display shortcomings currently prevent us from observing more subtle computational image differences on our systems. As we become able to tune its quality, diffraction-specific computing will likely provide us with a way of "dialing" between image-based and object-based fully-computed hologram rendering methods, to suit display capabilities¹⁶⁷.

4.3.11 Overall timing comparison to other methods

Finally, to situate incremental computing among other methods with more fixed



(a) traditional complex interference-modeled cut cube



(b) incrementally-computed cut cube



(c) stereogram image of a cut cube

Figure 4.23 Traditional complex interference-modeled, incrementally-computed, and computed stereogram images of same geometry

computational cost, a table describing the breakdown of average incremental computation cost is provided in Figure 4.24, as well as a table comparing other computational methods employing different hardware (Figure 4.25). All indicated computation times depend, of course, on implementation and hardware platforms. In all cases, we have eliminated the computational cost of scene modeling, since this may vary among techniques.

In Figure 4.24, we break out several steps of the algorithm (precomputing, initial computing, incremental-update, incremental-normalization, conditioning, and transfer) being run on an SGI Onyx with 2 250MHz processors. *Precomputing*: First we consider the time required to pre-compute a table of 100 elemental fringes; in our implementation, 12 seconds are required using one processor.

Initial computing: Next, we must compute an initial working hologram as a sum of elemental fringes for holo-points, normalize every line and condition the hologram for Cheops and the display. Depending on the number of holo-points and their depths, the time required for these operations will vary; for 737 holo-points distributed throughout the display volume, an average of 16.2 seconds were required for these operations, using one processor.

Incremental update: Then, as the model changes, the working hologram is updated by subtracting away old fringes and adding the new; the average time to update all 737 holo-points' fringes is 0.705 seconds using two processors, each operating on an average of 49 hologram lines.

Incremental normalization: To renormalized the updated spans on the changed hololines takes an average of 1.593 seconds, using two processors that each operate on an average of 49 hologram lines.

Conditioning: To byte-swap for Cheops and flip lines for the holovideo display, two processors, each operating on 72 lines, require an average of 0.769 seconds per update.

Transfer: Finally, it is important to consider the time required to shuttle a 36MB image to the framebuffer. Depending on the electronic subsystems serving the electro-holographic display, the time spent communicating the enormous amount of data in just one holographic frame can vary greatly. Using our HIPPI link to Cheops provides us with an almost-direct channel to the output cards; 0.6 seconds are required to transfer an entire hologram per update in our current implementation.

Thus finally, for an object defined by 737 holo-points, the initial compute and display time totals 28.6 seconds, and the incremental update and display time is only 3.67 seconds.

For the comparison among methods in Figure 4.25, we have provided benchmarks for the same computational platform, as well as for some implementations using special-purpose hardware. Here, we examine the amount of time required to compute the

incremental computation stage	time (SGI Onyx 250MHz)
precomputing 100-element fringe table	12 sec (1 processor)
computing initial working hologram for 737 holo-points (compute, normalize, condition)	16.2 sec (1 processor)
average incremental update of 737 holo-points transformed in 3D	0.705 sec (2 processors, 49 lines each)
average non-global incremental normalization per update	1.593 sec (2 processors, 49 lines each)
average conditioning for display (flipping lines, byte swapping)	0.769 sec (2 processors, 72 lines each)
average HIPPI transfer to display	0.6 sec (entire hologram)

TOTALS: initial compute & display time: 28.60 sec
incremental update & display time: 3.67 sec
(averaged over 108 timesteps)

Figure 4.24 Breakdown of incremental computing method computation times for a 36MB hologram

initial hologram using an SGI 250MHz Onyx with a single processor. Using complex interference simulation to compute a 36MB hologram from 737 holo-points requires 217.7 seconds. A stereogram, using 32 pre-rendered perspective views and 32 pre-computed basis fringes containing 1024 samples each, requires 243 seconds to compute. The complex interference modeling and stereogram approaches will both require roughly the same time to compute a frame each time the scene is modified.

For 737 holo-points, the incremental computing method outperforms both other methods substantially (by more than a factor of ten), requiring only 15.1 seconds to compute the initial working and display holograms. Incremental computing will always outperform complex interference modeling; if the holo-point count were to increase dramatically, as it would for surface-modeled geometry, eventually stereogram computing would become faster. *As mentioned previously however, if only part of the scene changes, incremental computing can offer a rapid way to produce new frames at a fraction of even stereogram computing cost.*

	object / input	platform	time
complex interference simulation	737 holo-points	SGI Onyx (1 processor)	217.7 sec
stereogram	32 pre-rendered views, 256x144 pixels/view	SGI Onyx (1 processor)	243.9 sec
		IBM PVS	5 sec
	32 pre-computed basis fringes, 1024 bytes/fringe	Cheops with 2 superposition stream processors	6 sec
incremental computation (initial working holo)	elemental fringe table, 737 holo-points	SGI Onyx (1 processor)	15.1 sec

(Times for computing and normalizing only.)

Figure 4.25 Performance comparison to other methods of computing 36MB holographic images

4.3.12 Toward interactive holographic display with better image quality

As we continue to push for both *greater interaction speed* and *greater realism* in electro-holographic systems, eventually, these two pursuits will not be so greatly polarized. As a concrete step in this direction, the technique of incremental computing offers a way to update holograms locally in response to partial scene changes, and in proportion to the scale of scene change. This method can offer tremendous computational economy and can also increase the potential for visual realism in computed images.

The holographic images produced in this early work are simple spatial wireframe renderings, though more elaborate modeling certainly should be incorporated into the algorithm. The most important feature yet to be added is occlusion processing; provisions for this are most crucial to rendering a scene that appears solid and convincingly 3D. Additionally, shadows, texture, reflections, and smooth shading are important for rendering realistic scenes.

If we step back for a minute to consider computational holography's almost dichotomous goals — computational speed and realistic rendering — we might situate the computing methods we have described in this chapter within the "goal space" as shown in Figure 4.26. Here, we have represented overlap between methods since there is unquestionably variability in the performance of each. The ultimate goal, of course, is to produce highly realistic images that can be updated at least 30 times every second. No algorithm as implemented on current hardware platforms can achieve this goal now, but incremental computing incorporating these methods for shading, texturing *etc.*, may move us closer to this stated goal.

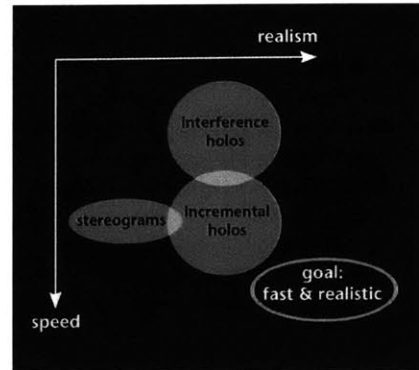


Figure 4.26 Computational holography's goal space and resident computing methods

Other important future work includes an algorithm that might adaptively populate a model's geometry to suit the display characteristics, based on the model's orientation. A more compelling future research challenge would be to consider alternative, more compact, or more physically-based surface representations. The incremental computing approach is applicable in any framework which models objects as a collection of primitives, providing that *transformation* of those primitives can be expressed with simple transformations of their corresponding diffractive patterns. While approximating continuous surfaces with a multiplicity of isotropic spherical emitters can produce impressive images, this model does not address complicated light-matter interaction, occurring on a microscopic scale, that gives surfaces their unique and sensual *material* look. Once qualified, computational models of fields generated by scattering from a rough surface¹⁶⁸, for instance, could be associated with some new fashion of geometric primitives¹³⁶. Even if computationally intensive to produce, elemental diffractive patterns might be pre-computed, tabled, and then used to incrementally update a more realistically rendered image.

This kind of elaborate modeling would be appropriate — and may be entirely essential — for a display technology capable of remarkable visual realism, whose images are viewed at close range, that may also be combined with high-fidelity haptic feedback. For now, even slowly computed holograms of realistically rendered geometry could provide a system comparable to *Touch*, described next, for inspecting *static* scenes. And, in its present simple form, incremental computing permits us to begin experimenting with arbitrarily updating a *dynamic* holographic image in response to haptically-applied model changes, as we do in the system *Poke*, described in a later chapter.

5. *Touch*: Two static haptic holograms

5.1 Motivation and goals

While building *Touch*, our goal was to render a simple prototype visuo-haptic workspace that would exhibit many of the same properties as a physical one; specifically, we wanted the demonstration to achieve some of the properties a computational plastic should exhibit. Assembled entirely from geometry data and computational models, the demonstration would certainly have programmable look and feel. And to mimic sensorimotor apprehension in the physical world, we wanted to move in this space with our effectors, and see and feel our effectors in contact with objects or agents represented within it.

By optically projecting a real image using a hologram, we sought to situate the visual image within the viewer's manipulatory space. To provide convincing haptic display and exploration of object shape, precisely registered force feedback was essential, particularly when *contact* with elements of the three dimensional scene is both seen and felt. Also essential was a method of holographic display which would not allow the interacting hand to block the reconstructing illumination. The system we chose for addressing these two design requirements couples full-parallax, edge-illuminated holographic stereograms and a six degree-of-freedom force-feedback device in a coincident display volume about 10mm on a side. *Touch* displays simple geometrical shapes, which can be inspected both visually and haptically. The next four sections describe our hologram design and recording implementations, the haptic simulation, and the composite system results.

5.2 System design

5.2.1 Holograms

In addition to providing a workspace which admits the hand without blocking illumination, edge-illuminated holograms have a compact design resistant to image distortions due to improper illumination¹⁶⁹. Providing horizontal *and* vertical parallax in these holograms allows greater freedom of movement throughout the viewzone without the astigmatic image distortions¹⁵⁸ associated with horizontal-parallax-only (HPO) holograms.

To generate any full-parallax holographic stereogram, many thousands of component computer graphic views of a scene may be required. Traditionally, after being carefully rendered according to constraints of the hologram viewing and recording geometries, these views were saved to a computer disk for subsequent holographic exposure. Thus, the production of each holographic stereogram was characterized by three distinct processes: designing the image, rendering the component views, and exposing these stored images. This sequential process expanded the production timeline and multiplied the number of resources required for stereograms in the production pipeline concurrently.

While this process was the standard even before sizable disk space and computer graphics hardware acceleration were commonly-available commodities, we hoped to streamline production by merging computer graphic rendering and holographic printing. To accomplish this goal, we built a suite of client-server based tools for design and preview, rendering, and printing. The nucleus of these tools is a rendering

server (*HoloServe*) which provides hardware-assisted computer graphic rendering to client applications. Using a client utility for scene modeling and preview (*HoloBuild*), a hologram designer can:

- import 3D model data,
- interactively light the scene,
- tailor the computer graphics camera to match hologram geometry,
- change model rendering parameters and
- request that a specific perspective view be rendered by HoloServe.

When a design is complete, HoloBuild can save a configuration file that completely describes the hologram design. Subsequently, a hologram printing program (*HoloPrint*) can request a specific component view of the design from HoloServe, which uses the configuration file to rapidly render and display it. Upon notification of rendering completion, HoloPrint sends appropriate exposure and frame-advance control sequences to the hologram printer. In this manner, a newly rendered frame is recorded and the printer is readied for the next exposure.

Both HoloBuild and HoloServe were designed to accommodate a wide variety of hologram viewing geometries, and both one-step and multiple-step printing processes. For any given optical printing setup, the specific geometries of the hologram printer can be specified in HoloBuild during the design process, recorded in the configuration file, and used later during hologram printing.

The first implementation of HoloBuild and HoloServe used SGI's Graphics Library (gl); HoloBuild runs on any SGI and HoloServe runs on an SGI Onyx with Reality Engine. Versions of both, current at the time of this writing, use OpenGL, and HoloServe provides single- or double-frustum rendering¹⁷⁰ to generate perspective views of complicated computer graphic models in a few seconds. These design, rendering, and printing processes communicate over ethernet (TCP/IP) using remote procedure calls (rpc). A diagram of the entire hologram production pipeline is shown in Figure 5.1. Two simple edge-illuminated holographic stereograms were produced using this suite of applications to demonstrate *Touch*.

5.2.2 Haptics

For force display, *Touch*, as well as the systems *Lathe* and *Poke*, use the PHANTOM haptic interface (Figure 5.2), a mechanical linkage and a three degree-of-freedom passive gimbal that supports a thimble or stylus used by the hand. Six encoders on the device provide positional information resolved to approximately 0.1 mm, and three servo motors provide force display up to roughly 8 Newtons in a workspace of approximately 290x400x560mm³.

The stylus or thimble are used to probe the simulated scene while device encoders are polled to compute the position of the end effector (stylus or thimble tip). This information is checked against the geometry of the scene; if contact is detected, appropriate torque commands are delivered to the device's three servo motors, and a restoring force is felt by the hand holding the stylus.

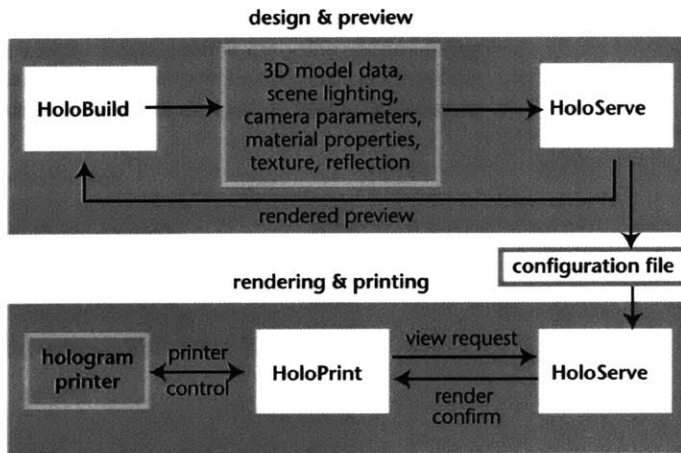


Figure 5.1 Generating *Touch* holograms



Figure 5.2 Hand grasping the stylus of the PHANTOM haptic interface

Our haptic simulation, originally dubbed *HoloFeel*, renders a force image of the scene geometry in space, which we co-locate with the projected holographic image. From a haptic scene design client, *HoloFeel* receives the scene geometry and haptic modeling parameters that correspond to a given holographic image and the force simulation

can then be felt with the PHANToM. HoloFeel currently runs on an SGI Octane, with an average servo rate of 10KHz. Its basic functionality includes the ability to render some implicit and parametric surfaces, object surface and bulk properties (static and sliding friction, compliance, damping, and texture), some ambient effects and some physical dynamics. A diagram of the entire *Touch* system are provided in Figure 5.3.

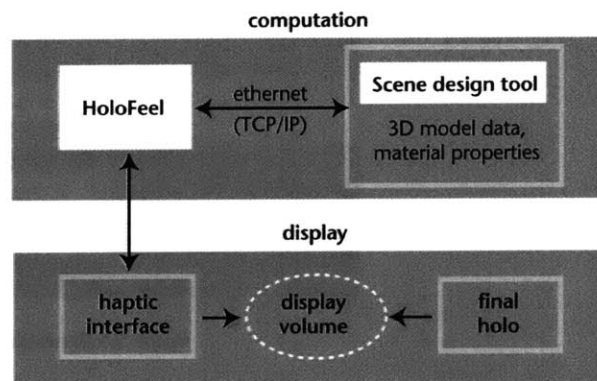


Figure 5.3 Touch system: combining holograms and haptic simulation

5.3 Results

5.3.1 Two haptic holograms

HoloBuild, HoloServe and HoloPrint, were used to produce two two-optical-step edge-illuminated holographic stereograms used in conjunction with haptic simulations. The simpler of the two holograms displayed a hemisphere affixed to a vertically oriented plane. The plane was both visually and haptically textured with a vertical grating; the hemisphere itself was visually texture mapped but had no haptic surface relief.

The full-parallax master hologram (first optical step) permitted a broad angular range of head motion of approximately 50 degrees horizontally and 30 degrees vertically (Figure 5.4). The master was comprised of 15000 rendered frames, and the final edge-illuminated hologram was printed in an additional optical transfer step. Mastering and transferring steps¹⁷¹ used a recording wavelength of 528nm.

The total depth of the final hologram was approximately 40 mm, all in front of the image plane. Image plane width and height were each 100 mm. The hologram was illuminated with an LED centered at 520nm, which yields a bright image with slight spectral blurring. The multi-modal scene presented was intended to have very few formal features; this tangible hologram provided a simple example with which to examine perceptual tolerances for spatial misregistration and mismatches in curvature of the visual and haptic models (e.g. by replacing the haptic hemisphere with a hemi-ovoid).

The second and slightly more complex holo-haptic example used an arrangement of blocks forming a maze, which is oriented against a vertical back plane. The blocks varied in size and spacing, and the channels formed between them are narrow. The

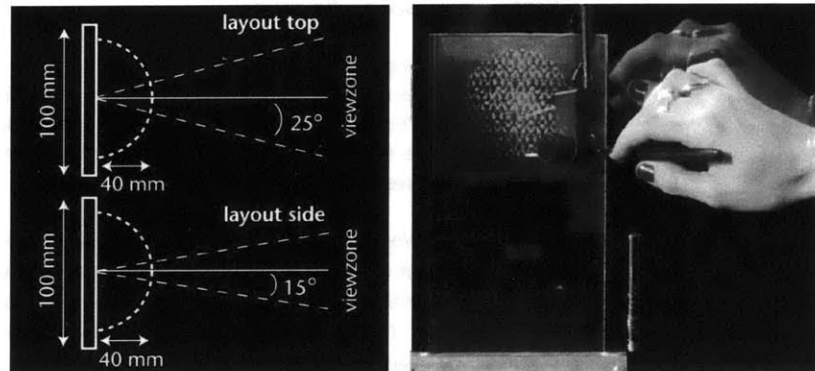


Figure 5.4 Hemisphere hologram and spatial layout

back plane was visually and haptically texture mapped with a vertical grating.

The full-parallax master provided a much smaller horizontal range for head motion in the final hologram, with approximately a 35-degree field of view (Figure 5.5). The master was comprised of 6700 exposures of pseudoscopically-rendered frames and the final hologram was produced in an additional optical transfer step. As in the previous hologram, the mastering and transferring¹⁷¹ steps used a recording wavelength of 528nm.

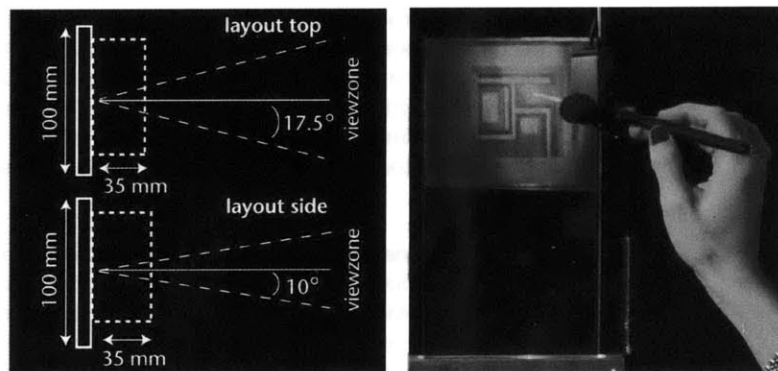


Figure 5.5 Block hologram and spatial layout

The total depth of the final hologram was approximately 35 mm, and the entire image reconstructed in front of the hologram plane. Image plane width and height were each 100 mm. The final hologram was illuminated with an LED centered at 520 nm. The model presented contains more image features (edges) than did the simple hemisphere hologram; the maze hologram offered us a chance to examine maze-tracing performance in this coincident and an offset (using stereo graphics) visuo-

haptic workspace configuration.

5.3.2 Feeling holograms

Three questions most relevant to our goals are: *how did the holograms look?*, *how did the holograms feel?*, and *did the combined simulation allow natural movement in the workspace?*. We address these questions based on both careful observation and subjective impressions.

How did the holograms look? Both holograms were illuminated by an LED whose expanding light did not have a uniform intensity profile. Thus, the reconstructed images appeared bright in the center but dimmed closer to their edges. The resulting apparent (and unlikely) variation in reflectance of the holographically-projected object is perhaps the greatest detriment to both the impression of solidity, and overall image quality. Additionally, the broad spectrum of the LED caused some blur in the reconstructed image as a function of distance from the image plane. To improve both of these conditions, we illuminated the holograms with coherent light. This did provide more consistent illumination over the hologram's area and eliminated blur. While the image appeared very bright, solid, and three-dimensional throughout the viewzone, the introduction of speckle and the monochromatic nature of laser light still gave the images an unnatural look.

How did the holograms feel? The haptic simulations were stable and precise to a fraction of a millimeter. While we only provide simple simulations of surface and bulk properties (distinguishable from those of most natural materials), the force display of the models was compelling and believable. Especially when combining the haptic simulation with a visual image, certain expectations about how the surfaces *should feel* become apparent. For example, surfaces which look hard and shiny should probably feel hard and smooth, not soft and sticky. The physically-based coupling between visual and haptic surface properties should be kept in mind when modeling these kinds of multimodal simulations. The bulk properties of the surfaces we simulated were slightly too compliant and viscous for the surface properties ascribed to the visual image.

Did the combined simulation allow natural movement in the workspace? When the haptic model's geometry was precisely aligned with the holographic image, the impression of inspecting a single, multimodal image was quite strong. Generally speaking, one would not likely inspect a real hemisphere or block with a probe rather than using the whole hand, if given a choice. Thus, at the outset, we must consider ourselves to be subverting a natural interaction style somewhat. Yet the dexterous use of a hand-held tool to inspect or manipulate objects does apply in some scenarios, such as surgery, sculpting or driving a screw. These are the types of applications for which our experiments are most relevant.

Two interactions provided the most compelling impression of a multimodal spatial image: tapping on the image (making a coarse reality-check), and inspecting small image detail like an edge or a bump (focusing attention on the supporting multimodal cues). During larger sweeping motions over a surface, conflicting visual and haptic

cues seemed to be less critical. If the PHANToM stylus tip was rapidly moved over the surface of the hemisphere, small distortions in the haptic model's curvature were difficult to detect; yet binocular vision *could* detect these changes if the tip was held stationary against the force model and visual image detail was not spatially coincident.

Perhaps the weakest aspect of these multimodal experiments is that physical volumes can penetrate holographic ones. All parts of the interacting hand and the haptic apparatus, but for the stylus tip, can freely pass through the holographic image. Thus, while the workspace does admit natural movement of the hand and hand-held tool, it cannot report all visual and haptic cues we would expect from a physically-based analogue.

5.4 Discussion

As we readily observe in our everyday interactions, harmonious multisensory stimulation usually gives rise to correct perception of objects and events. The broad body of work on multi-sensory interaction indicates that some disparity between visual and haptic information can distort the overall percept while still being tolerated. The ability of sensorimotor systems to adapt to discordant sensory input permits us to perform well even in the presence of distortion, so long as sensory feedback is available. This fact is extremely useful in offset visuo-haptic workspace configurations, wherein the tracked hand or device position is represented as a graphical element on the visual display and the user never actually visually observes her hand. In such workspace configurations, slight spatial misregistrations, or changes in scale between the visual and haptic display can be virtually unnoticeable. Yet too much intermodality disparity can cause the visual and haptic cues to be perceived as entirely separate events, and may be quite confusing or annoying.

Tolerances are lower still when visual and haptic workspaces are superimposed. We alluded earlier to some conflicts that arise between what is seen and what is felt in our coincident holo-haptic workspace; these intra- and intersensory conflicts are described in turn below.

5.4.1 Spatial misregistration

When exploring a surface with the Phantom and visually monitoring the device, simultaneous visual and haptic cues to the surface location are available. When we *feel* contact, the visible location of the stylus tip is perceived to be co-located with the haptic surface. During contact, if the holographic surface and the haptic surface are not precisely aligned, the misregistration can be strikingly obvious to vision. These conflicting visual cues erode the impression of sensing a single object. Instead, the impression of two separate representations is evident. This condition is shown in Figure 5.6.

5.4.2 Occlusion violations

As mentioned earlier, occlusion is perhaps the most powerful cue to layout in a scene. When we see the image of an object being blocked by the image of another, we understand the occluded object to be farther from our eye than the occluding one. In our holo-haptic systems, it is possible to position the haptic apparatus between hologram and image and actually block its reconstruction; in an observer's view of the

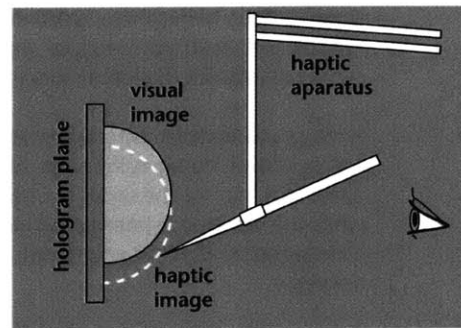


Figure 5.6 Conflicting cues: spatial misregistration

scene, occlusion relationships contradict other depth cues reporting true scene layout as shown in Figure 5.7. Even in the presence of correct depth accounting from stereopsis and motion parallax, Perception appears to favor the depth ordering reported by occlusion relationships.

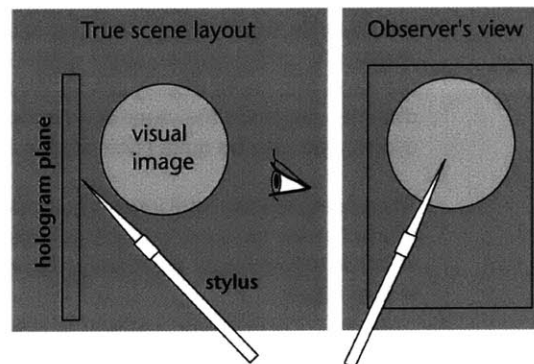


Figure 5.7 Conflicting cues: occlusion violation

5.4.3 Volume violations

Obviously, holograms present spatial images which cannot by themselves exhibit a restoring force when penetrated by an object. With no haptic simulation running to detect collisions with model surfaces and to display contact forces, the haptic apparatus is free to pass through the holographic image undeterred. Our haptic simulation can prevent a single point on the stylus from penetrating the model, but current device limitations preclude emulation of the kind of multi-point contact that occurs in the physical world.

During each haptic control loop cycle, HoloFeel checks for a surface collision all along the stylus probe; even if it finds many, it can only compute and display forces for one. If a model surface has been penetrated by the stylus tip, it is assumed the viewer's primary attention is focused there, and forces due to this collision are computed and

displayed. However, if not the tip, but other points along the probe have penetrated the model, then the collision closest to the tip is used for computation and display.

The situation permits another kind of occlusion problem, which we call a volume violation, to occur as shown in Figure 5.8. While the stylus tip is seen and felt in contact with some geometry, the stylus may be rotated around its tip and swept through proximal holographic image volume. Parts of the user's hand may also penetrate the image. Seeing such physical objects and holographic image coexist in the same physical volume presents a confusing impression of depth and object solidity in the scene.

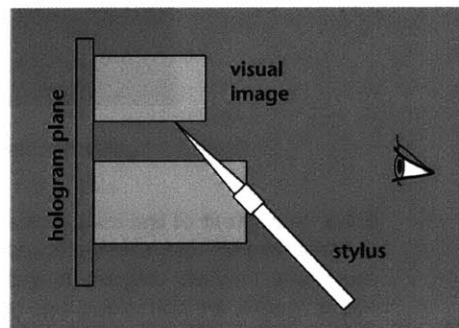


Figure 5.8 Conflicting cues: volume violations

5.4.4 Optical-haptic surface property mismatch

As briefly mentioned earlier, an artifact of a hologram's diffractive properties is the chromatic blurring that occurs with broad spectrum illumination. In the transmission edge-illuminated holograms used in this work, the holographic image plays out high and farther from the hologram in wavelengths shorter than the recording wavelength, and lower and closer in longer ones. If the illumination source used in hologram reconstruction is not monochromatic, spectral blur will be evident in the final image. Image elements close to the hologram plane will be quite clear, but those farther from the hologram plane will exhibit blur in accordance with source bandwidth.

Since a viewer generally expects scene elements closer to the eye to be more keenly resolvable, the blurry image elements near the viewer challenge the impression of image solidity. This condition, shown in Figure 5.9, is recognized as problematic on its own, but adding coincident haptic display causes further difficulty. Usually an object's bulk material properties (*e.g.* stiffness) remain uniform throughout the display volume. If the haptic and visual output are precisely in register, then near the hologram plane the stylus will be exactly coincident with an imaged surface during contact. However, far from the image plane, the stylus will visually penetrate the blurry image of the surface by a substantial distance before contact is felt. As mentioned earlier, misregistration between the image surface and stylus tip during contact, especially when close to the viewer's eye, can diminish the simulation quality. In addition, visual

and haptic information presented is conflicting; by visual report, the surface qualities change substantially with depth though their haptic quality remains the same.

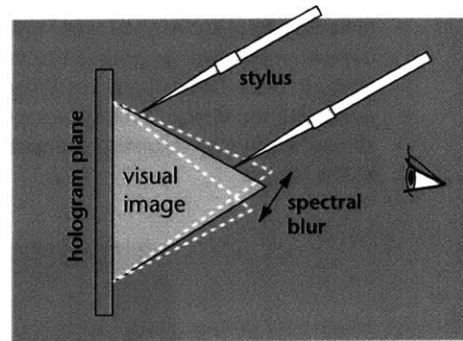


Figure 5.9 Conflicting cues: optical-haptic surface property mismatch

5.4.5 In context of the ideal holo-haptic system

In these several ways, the haptic and holographic images are not in perfect spatial synchrony; however they are in excellent temporal register — since the visual and haptic models are both static, one display modality never lags the other. The characteristics of *Touch*, in relation to our ideal holo-haptic system mentioned earlier, is summarized in Figure 5.10. Here we can see that haptic update is adequate enough to maintain simulation stability, both holographic and haptic models are in good spatial and temporal register, but that the visual display cannot be modified.

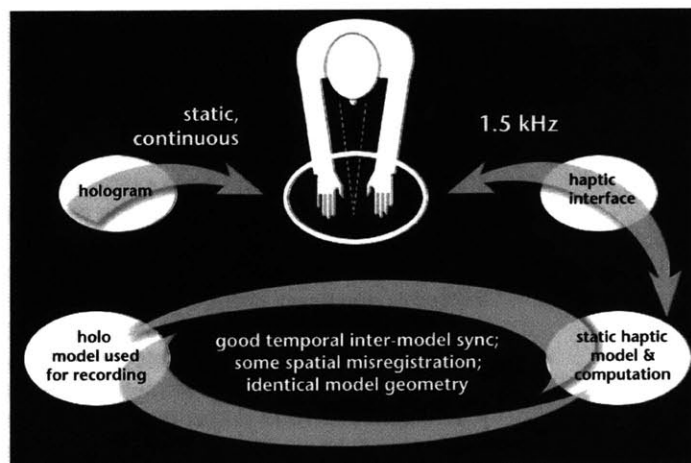


Figure 5.10 *Touch*: good holo-haptic register but no hologram update

5.5 From static to dynamic display

Despite the presence of cue conflicts, these full-parallax haptic-holograms presented a compelling way to haptically inspect simulated spatial objects using a compact tabletop display. When a person taps the edge of a block, or visually and haptically locates a trough in an object's texture with the stylus tip and follows it through the image, the impression of a single multimodal representation is quite strong. As indicated in Figure 5.10 above, the principal disadvantage to these displays (and no small one at that) is that they are static; they have utility for inspecting three dimensional shapes, but do not permit interaction with or modification of the data presented. Our next experiments sought to investigate the addition of dynamism to holo-haptic display.

6. *Lathe*: A haptic hologram with limited update

6.1 Motivation and goals

The logical extension of using a haptic device to inspect a static holographic image is to allow an operator to *modify* the geometry of the underlying model through haptic interaction. Our primary goal was to produce a simulated haptic and holography surface that was not just touchable, but that could also be *reshaped* by an interacting user. To produce "rewriteable" holograms, we used the MIT second generation holographic video display (*holovideo*)¹⁷², and registered its display volume with that of the PHANToM.

The 3D image produced by holovideo supports the most important depth cues: stereopsis, motion parallax, occlusion, and many pictorial and physiological cues to depth. However, choosing an electro-holographic display does currently require trading off image quality (as compared with conventional displays) for 3D display with limited dynamism. As electro-holography moves beyond its infancy, we expect the underlying technology will accommodate good image quality and acceptable framerate.

In this experiment, providing a visual frame rate of at least 20 frames/second was our hope *and* a challenge; holovideo's current computational subsystems have limited computation and communication bandwidth, enabling it to display one frame every few seconds at best.

6.2 Overview

In the *Lathe* demonstration, we present holographic and force images of a vertically-oriented, spinning cylindrical stock, which can be carved along its length into an arbitrary surface of revolution. The holographic and force images of the stock are overlapped and precisely registered to provide a free-standing three-dimensional image in which the changes due to carving can be both seen and felt.

Two distinct systems comprise the computation which feeds the displays; the *haptics module* (a modified version of HoloFeel), which performs force modeling, and the *holovideo module* which pre-computes holograms and drives rapid local holographic display updates based on changes to the model. An additional intermediate process, the *Workspace Resource Manager*, initiates and manages both the haptics and hologram modules, is notified of geometry changes imparted to the haptic model, and requests hologram updates to local regions of the visual display where changes have occurred. The entire system overview is shown in Figure 6.1.

We designed *Lathe*'s model to be rotationally symmetric during carving, permitting several simplifying measures for improving the visual update rate: first, rather than being entirely recomputed, the hologram itself is updated only in regions of change; second, updates are *precomputed*, and swapped into the hologram in appropriate locations. From the point of view of a user, who is holding the stylus and pressing it into the holographic image, a single multimodal representation of the simulation can be seen and felt changing in response to the applied force.

6.3 Haptic simulation

The haptic simulation represents a spinning surface of revolution (initially a cylindrical "stock") which can be interactively "lathed" by user. The haptic cylinder, initially and in subsequent stages of carving, is a parametric surface of revolution with two algorithmically-defined caps. It has a mass of 1 gm, an algorithmically defined vertical grating as surface texture, static and dynamic frictional properties, stiff spring bulk resistance, and rotates about its axis at one revolution per second. The cylinder model straddles a static haptic plane (which spatially corresponds with the physical output plane of the holovideo optical system); the haptic plane is modeled with the same bulk

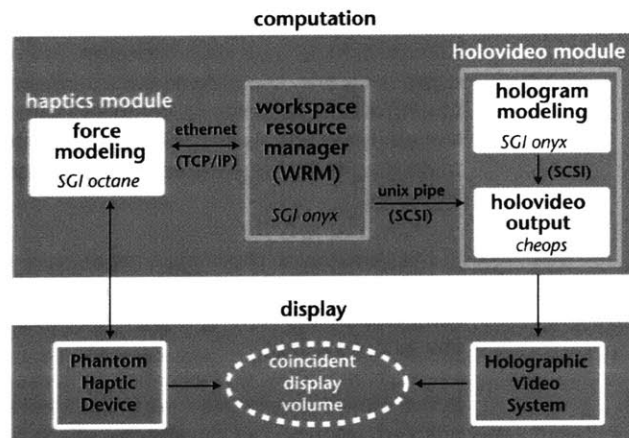


Figure 6.1 *Lathe* system overview

and frictional properties as the cylinder. Currently, the haptics simulation is implemented on an SGI Octane with an average servo rate of 5KHz.

6.3.1 Haptics modeling

The haptic stock maintains rotational symmetry about its vertical axis initially and in all subsequent stages of carving. The radius profile of the surface of revolution is represented as a nonuniform rational B-spline (NURB) curve with 28 control points, all of which are initially set to the same radius value (25mm) to let us begin with a cylinder. The curve evaluated between the middle 21 points defines the profile of the cylinder body; the remaining top three and bottom four points lie beyond the actual extent of the cylinder, and serve to "lock" the shape at its top and bottom, respectively. Each control point has an associated "ghost" point, G_i , which is actively displaced due to user-applied force. Once displaced, the ghost point acts as an attractor which smoothly draws its control point to the displaced location, over a number of timesteps.

As force is exerted on the shape at height h , between control points P_i and P_{i+1} , the two corresponding ghost points straddling the penetration location are displaced

toward the central cylinder axis by a fraction, k , of the penetration threshold, D , once it has been exceeded. The lower ghost point is displaced by tkD , and the upper by $(1-t)kD$, where t is the normalized distance between control points.

As points P_i and P_{i+1} are displaced toward their ghost points, a new radius for the surface of revolution at height h , and a new surface normal are computed by evaluating the NURB formulation. If contact occurs directly on a control point, then that ghost point alone is displaced by kD . Thus, the cylinder can be felt spinning beneath the user's touch, and when contacted and pressed with enough force, the circumference of the cylinder is modified and surface is felt to deform (Figure 6.2).

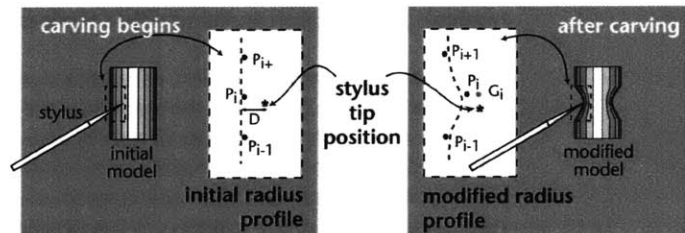


Figure 6.2 Carving and model deformation

The parameters k and D can be adjusted to make carving the rotating cylinder require more or less force. A minimum radius of 15mm is enforced, so that once the surface has deformed this much, the control points update no further. The control point density, approximately 0.4 points/mm, was experimentally determined to be high enough to keep the region affected by carving small (and thereby minimize the amount of hologram updating required), yet sparse enough to avoid unstable deep notching of the haptic surface.

6.4 Workspace Resource Manager

The Workspace Resource Manager (WRM) both initiates and manages the haptics simulation and hologram updating. First, it initializes its own spline model of the surface of revolution, which starts as a cylinder of desired height and radius. It then launches the haptic simulation by making client calls to the haptics module requesting creation of a haptic cylinder of the same height and radius at a desired location. The haptics module commences physical simulation of this spinning cylinder, and computes collisions of the Phantom tip with the computational model as described previously.

Changes in the haptic cylinder's underlying control points are automatically communicated from the haptics module to the WRM approximately 30 times per second. The information sent contains the location where change begins on the curve (the number of the bottom-most control point), and values of six surrounding control points, ordered from bottom to top. It is assumed that model changes occur reasonably slowly, so that no more than six control points are updated within 33 ms.

This communication rate means that we can only guarantee reporting accurate model changes in a region of the stock roughly 15 mm high, within an update interval. Though this assumption usually puts us within the realm of normal interaction speed, eventually, communicating a variable number of control points to reflect the precise region of change would be more robust. Our subsequent work has implemented this change.

6.4.1 Translating model changes into hologram changes

Once the WRM receives an update message, the changed control points are used to modify its own representation of the radius profile. The spline model of the stock is continuous, of course, and varies smoothly along its carved profile; in our implementation, translating changes in this spline model to hologram updates requires going through an intermediate representation.

This intermediate representation, called the *stack* (a state vector with 120 elements, shown in Figure 6.3), models the stock as a pile of 120 disks. The number of disks in the stack represents the number of display lines occupied by the final holographic image. Each disk in the stack has a radius computed from the spline model at a corresponding height and then *quantized* to one of five values. These quantized values represent the radii of the cylinders in five precomputed holograms used to assemble the final image, as described in the next section.

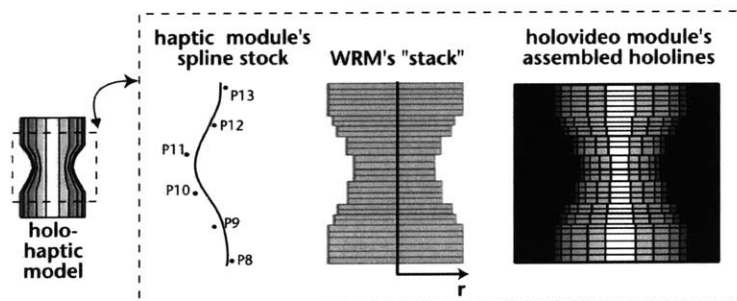


Figure 6.3 Representations of the stock throughout the system

A set of six holovideo display lines correspond to the space between any two adjacent control points in the WRM's model. When control points have changed, we recompute stack radii for the set of display lines spanning these updated points. Since we currently report a no more than six control points to the WRM per timestep, if all six (or more) control points have changed, the curve will actually be affected beyond the region they span. However, to conserve our communication bandwidth, we compute updates only for the 30 display lines bracketed by these six points, and assume that subsequent carving operations on the curve above and below these points will serve to incorporate the necessary changes into the stack, if a bit late. In the current implementation, the WRM's spline model is also rendered to a graphics display using SGI's Graphics Library to provide a means for visually monitoring a user's

performance, since holovideo’s viewzone only comfortably accommodates one viewer at a time (Figure 6.4).

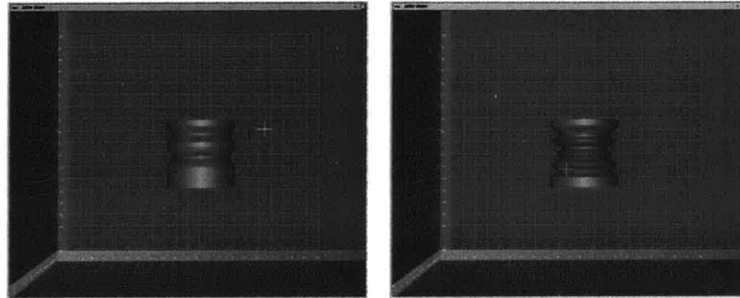


Figure 6.4 Rendered views of stock with cursor indicating stylus tip

The number of changes in the stack, and in turn, the number of display lines to be updated depend on exactly which model control points are displaced. In the center of the stock, carving may require that as many as 30 display lines be updated per timestep. In regions near the top or bottom of the carved shape, a smaller region of the curve contributes to the visible extent of the shape, so fewer display lines will require changing.

To communicate these changes to the holovideo display, the stack elements corresponding to changed display lines are assigned a *radius code* based on their quantized values as shown below:

radius (mm)	25.0	22.5	20.0	17.5	15.0
code	5	4	4	2	1

A compact message, which contains the number of the hololine marking the start of the update region, the number of lines that need to be updated, and the radius codes of each new line, is sent to the holovideo output module. With this information, a new hologram can be assembled and displayed.

6.5 Hologram simulation

As mentioned previously, we employ the second generation of holovideo in this work¹⁷². This system is capable of displaying monochromatic, horizontal-parallax-only (HPO) images in a volume of 150 x 57.5 x 150 mm³, and the viewing angle is 30° (Figure 6.5).

For the present purpose, we may consider holovideo to be a black box which accepts two inputs: a computer-generated hologram and light. The output of the black box is a 3D holographic image whose visual and geometrical characteristics depend on how the CGH was computed. As noted in Chapter 4, each CGH contains an enormous amount of data — 36 megasamples (at 1 byte per sample) — apportioned into 144

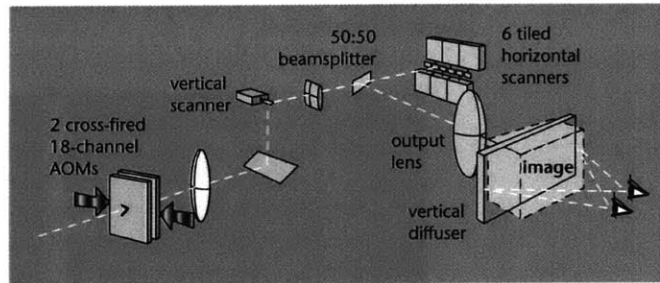


Figure 6.5 Holovideo system

hololines of 256 kilosamples each. The CGH is made available to the display via a framebuffer. As mentioned in chapter 4, because holovideo has a non-standard display format, the Cheops image-processing system was extended to support it. To reiterate, Cheops has three different module types: processor, input/memory, and output, and an optional memory module provides up to 0.5 Gbytes local to the system. These modules are interconnected by two types of linear buses. One, the Nile bus, is capable of sustained high bandwidth (>100 Mbyte/sec) transfer of samples and the other, the Global bus, is capable of 32 Mbyte/sec transfer¹⁶⁵.

6.5.1 Precomputed holograms and holovideo indexing

Because it is not yet possible to compute 36 Mbyte holograms in real time, for *Lathe* we use a precomputed set of five HPO holograms of wireframe cylinders to efficiently *assemble* a hologram of the stack. These holograms were computed using interference modeling¹³⁶, described previously, which produces a sharp and vividly three-dimensional reconstructed image. Each precomputed hologram displays a cylinder with a different radius: the initial cylinder, and four progressively smaller ones, r_{cyl} (mm) = {25.0, 22.5, 20.0, 17.5, 15.0}, ending with the minimum-radius cylinder (Figure 6.6).

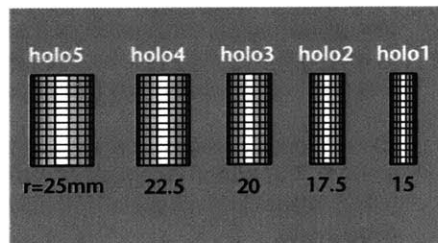


Figure 6.6 Precomputed hologram set

All holographic cylinders are 47.9 mm high. These holograms, from largest to smallest radius, are loaded sequentially into the Cheops memory module. At system start-up, the cylinder with the largest radius is displayed. As the initial haptic cylinder is carved,

the hologram module receives the WRM's message containing changed lines and their radius codes. Upon receiving this update message, the holovideo module instructs Cheops to collect the appropriate new hololines from the precomputed set of holograms, and assemble them into the displayed image. This is accomplished by indexing into the memory module with the radius code to acquire the correct line from the correct hologram, and then writing the hololine to the output card.

Thus, each time the image requires updating, a small chunk of the final hologram is re-assembled by replacing current hololines with appropriate lines from the precomputed holograms (Figure 6.7). The resulting final hologram displays an *approximation* to the original haptic model's shape.

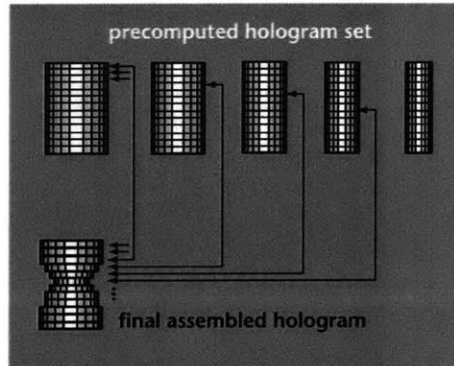


Figure 6.7 Assembling the final hologram from the precomputed set

The entire process is summarized in Figure 6.8, from carving the haptic model, through the use of changed control points to update the WRM's "stack", to the assembling of pre-computed hololines to create a new visual image.

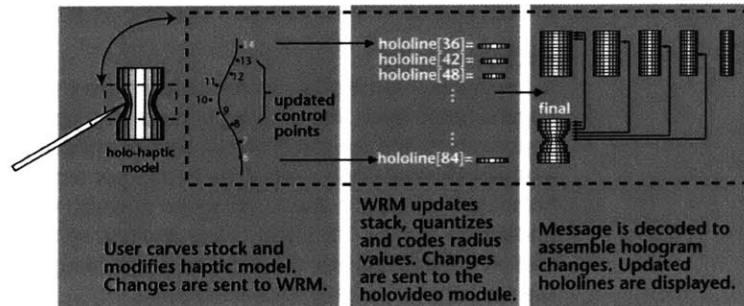


Figure 6.8 Lathe system: from carving to hologram update

6.6 Results and discussion

When an operator carves the holographic surface of revolution with the PHANTOM, the hologram image changes due to force apparently applied by the tip of the stylus. The resulting shape can be explored by moving the stylus tip around the surface without exerting sufficient force for carving. Physical objects in the workspace may also be explored, so that both physical and simulated forces can be displayed to the operator alternatively in the same "mixed-reality" workspace. When the operator maintains the correct viewing distance for holovideo, the perception of a single multimodal stimulus is quite convincing. Images of an operator interacting with the image are shown in Figure 6.9. Images of the holographic stock, in different stages of carving are shown at the right of the figure. Additionally, once the model has been

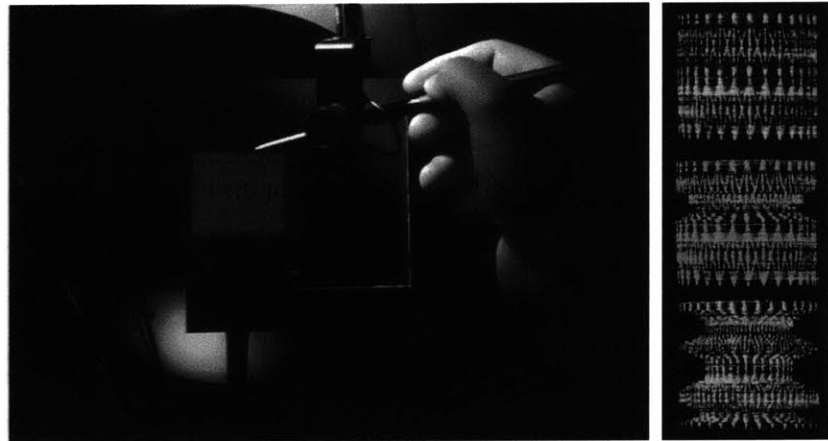


Figure 6.9 User carving the haptic hologram and three carved images

carved into "finished" form, it can be dispatched to a 3D printer which constructs a same-scale physical prototype of the digital design (Figure 6.10). Demonstrating a digital path from iterative design to physical prototyping gives us an early glimpse of how such technology might someday be incorporated into the design studio.

6.6.1 System lag

A compelling multimodal representation depends heavily on minimizing, to imperceptible levels, the time lag between the operator effecting changes in the haptic model and the result of that change appearing on the visual display. A reasonable visual update rate (20+ frames per second) is not currently possible on holovideo, principally due to the speed at which we can communicate with and update the display. The effect of the resulting system lag, on the order of 0.5 sec., is that an operator can see the stylus tip penetrating into the holographic surface before the surface is apparently subtracted away.

This effect is somewhat mitigated by rendering the visual image as a wireframe; were the model rendered as a smooth-shaded solid, seeing the stylus's visible surface



Figure 6.10 3D printed prototype of a carved design

penetration before the effects of carving would be more egregious. Nevertheless, we view this system lag as an artifact of the system's current computational architecture, and sufficiently small to prove the efficacy of haptic holovideo. Higher bandwidth spatial light modulators, efficient data compression techniques, improvements in computation speed, and higher bandwidth data pipelines will all help to alleviate this problem in future generations of the holovideo system.

Unlike many head-tracked parallax displays, a holographic display provides the full range of horizontal parallax in the viewzone regardless of viewer position. Thus, no lag is encountered with motion of the operator's head. Additionally, no jitter from tracker noise is present and no special eyewear is necessary to perceive the stereo information.

6.6.2 Differences in visual and haptic renderings

Our haptic simulation models a spinning surface of revolution, but the visual representation does not spin. In order to represent a spinning holographic image, we need to be able to update all the hololines spanned by the image at a reasonable rate. As mentioned above, our system currently suffers a low frame rate with the update of only 32 lines; thus we chose to forgo animating the spinning of the holographic surface. To our great (and unexpected) advantage, users have exhibited a surprising acceptance of the rotationally-symmetric carving behavior despite the visual discrepancy, once they assume the underlying lathe metaphor. Nevertheless, when visual update can be more rapid, the appropriate visual animation should be included.

When the stylus tip is touched to a detail on the holographic image, touch, stereopsis and horizontal motion parallax reinforce the perception that the stylus and the holographic surface detail are spatially co-located. However, as is the case for all HPO holograms, the lack of vertical parallax causes a slight vertical shift that increases with image depth to accompany vertical head motion. Here, we cannot claim, as we did with full parallax static holograms, that motion within the viewzone is unconstrained and offers undistorted viewing of the scene. Rather, a viewer must observe from the correct distance and height to receive an undistorted view.

6.6.3 Differences between simulation and real task

Differences between the visual and haptic feedback in our stylized lathe simulation and the experience of carving on an actual lathe are important to note. Among them are that the simple material properties we currently simulate are quite different from those of wood or metal moving against a cutting tool. Additionally, since a “cut” applied at an instantaneous position on the cylinder surface results in a surface modification that extends around the *entire* shape circumference, the user does not experience the feeling of continuously removing material as the shape spins under the stylus. Of course, the visual image shows a short wireframe and luminous stock, without even a hint of imitating the look of a more natural material. And finally, another obvious departure from reality is the 90° change in orientation of the lathe axis; clearly, rather than closely mimicking, the Lathe system only borrows thematically from the actual task.

6.6.4 Sensory affordances and conflicts

The sensory conflicts in this display include all those found in the static holo-haptic displays. Most often encountered are volume violations due to system lag. At the moment when an operator feels that the stylus tip is in contact with the surface, if the tip is seen either penetrating the surface or not making contact at all due to disparities between the visual and haptic output, the visual discrepancy is striking. Our simulation is always vulnerable to this problem when the operator is actively carving the surface.

Like static haptic holograms, the display does afford binocular disparity, motion parallax and pictorial cues to depth and layout. Unlike stereoscopic, half-silvered mirror displays, the hand and haptic apparatus are able to correctly occlude the holographic image. The converse, however, is not true; thus, unless hologram reconstruction is blocked, occlusion relationships are correct in the visual scene.

6.6.5 General discussion

The dynamic and the static systems described thus far offer interaction with a holographic images on the table-top; this marks a long-held goal in the field of holography. In both of these systems, holographic images in the manipulatory space are accompanied by real objects as well (at very least the hand and haptic apparatus). In the resulting mixed-reality setting, visual, haptic and physical behavior differences between the holographic image and juxtaposed physical objects can be quite striking.

Even if we have done our best to render the holographic images with a solid, three-dimensional appearance, discrepancies between spatial images and real objects call attention to the boundary between simulation and reality. Noticeable distinction between real and synthetic objects may not necessarily impact performance in this space, but to the extent that we want to render a *physically believable* scene, we need to identify and consider the underlying issues more carefully.

The method we use to assemble holograms is obviously valid only for a very narrow class of objects — models whose geometry can be described as a stack of “prefabbed” parts. Here, the rotational symmetry of our object permits us to interchange lines from several precomputed holograms to construct a small set of different shapes. Thus, this

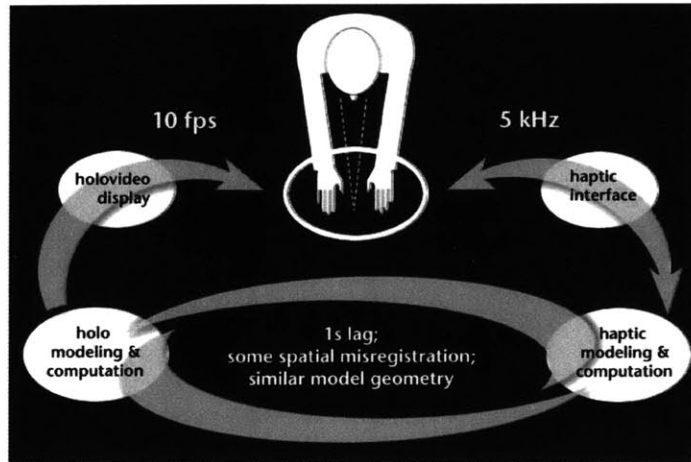


Figure 6.11 Lathe: visual update lagging haptic update by about a second

specific method of updating holograms is of limited value to 3D modeling applications. (It must be also noted that this method of hologram assembly is valid only for HPO holograms; for full-parallax holograms which would also have active "holo columns", locally updating the fringe pattern would be more complicated.) The intermodal disparity resulting from using a holographic model that approximates the haptic one is reflected in Figure 6.11, along with *Lathe's* lag in visual update and its low frame rate. Importantly, the figure indicates the path now opened between hologram modeling and computation and interactive display update. In this sense, we are progressing toward the architecture of an ideal holo-haptic system, though it will still be necessary to reduce lag, increase frame rate and to insure the holographic and haptic models are given by the same underlying representation.

Even while we may not meaningfully engage our skills for sculpting or materials-working using the holo-haptic *Lathe*, this system stands as a basic proof-of-concept, and reveals clues for how a truly useful holo-haptic tool might be implemented. Accordingly, a specific contribution of this work to the field of computational holography is the concept of determining and *interactively applying a set of pre-computed diffractive elements*, which can be locally added to, or removed from a hologram to achieve visual changes *at the image level*^{173,174,175,176}. If we can imagine a more general set of precomputed-elements, rapid incremental hologram updates might be used to effect arbitrary image modifications and to increase the degree to which underlying hologram and haptic models correspond. In our next experiment, we sought a such a way to render holograms, so that *more flexible* modeling changes to *identical* hologram and haptic scene geometry would be possible.

7. *Poke*: A haptic hologram with arbitrary update

7.1 Motivation and goals

So far we have presented static holographic objects with programmable look and feel, and which express bulk and surface properties when probed with a hand-held device. We have demonstrated rotationally-symmetric holographic objects which can similarly be felt, and also reshaped in a very limited fashion. However, to demonstrate all the properties a computational plastic should exhibit, this technology must still provide some nominal affordance for use; this might be accomplished by allowing more arbitrary interactive reshaping of the surface.

Of course, we can construct a haptic simulation alone, in which some model representation can be explored and arbitrarily modified in real time; and we can render a computer graphic view of that model in real time — even a stereo view. It is the holography that presents the most significant engineering challenge for an interactive venue. While holography has much to offer as a spatial imaging technique, the rigors of hologram computation and data transfer must be addressed before holography becomes a viable display method in all but contrived interactive applications.

Thus, permitting more arbitrary changes to be haptically introduced into the underlying model, and more critically, finding a way to propagate these arbitrary changes into the hologram *at interactive rates* were the two primary goals of our third experiment.

7.2 Overview

In the *Poke* system, we present coincident holographic and force images of a sheet of pliable material, which can be haptically felt, poked, and deformed by a user holding a force feedback device. The holographic and force images are, as in our previous experiments, spatially overlapped and metrically registered to provide a free-standing image that can be both seen and felt in the same spatial location.

Poke's system architecture is slightly different from that of *Lathe*; there are two distinct subsystems providing the haptic and holographic simulations. The *haptics module* performs force modeling and controls the PHANToM haptic interface; the *holovideo module* uses precomputed hologram elements to interactively compute rapid local hologram updates; and a display service called *holoPut* updates the hologram as required. The holovideo module receives user-applied haptic model changes through a client connection to the haptics module, updates the hologram, and sends it via a HIPPI link to *holoPut*, running on Cheops. The entire system overview is shown below in figure 7.1.

Poke models the pliable sheet as a parametric surface whose control points are also visually rendered to form the holographic image of the surface. From a user's perspective, a surface which feels smooth and continuous and is visually defined by a minimal grid of points can be pushed against with a hand-held stylus and dimpled in any location inside its perimeter. Changes are both seen and felt in response to the applied force.

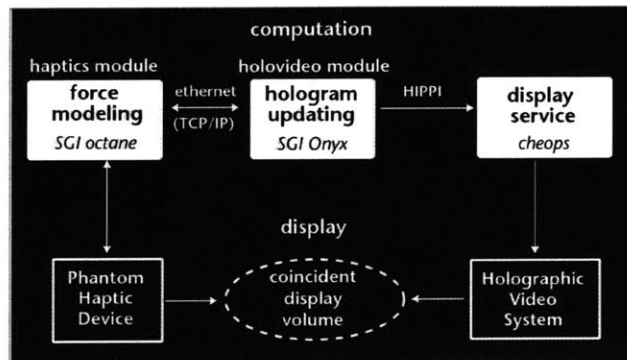


Figure 7.1 Poke system overview

7.3 Haptic simulation

The haptics module represents the pliable sheet as a Catmull-Rom spline surface, which smoothly interpolates a grid of 40 x 25 control points. These points initially describe a vertically oriented (in the x-y plane) flat surface located at $z=30\text{mm}$ in front of holovideo's image plane, approximately 50 mm high and 80 mm wide. Continuing beyond the parametric surface's extent is a haptic plane which both prevents exploration behind the deformable surface, and protects holovideo's optical components from contacting the haptic apparatus.

All around the surface perimeter, the outer most three control points have their positions locked so that all edges of the deformable surface remain attached to the surrounding plane when the interior control points are haptically manipulated. As a user pushes the PHANTOM stylus tip into the surface with enough force, the sheet is deformed as the interior control points are displaced. Each interior control point $C_{i,j}$, where $3 \leq i \leq 22$ and $3 \leq j \leq 37$, has an associated "ghost point" $G_{i,j}$. To deform the surface, it is the ghost points that are first displaced and then behave as attractors that draw the actual control points smoothly toward their new position over several timesteps.

The method of displacing control points in *Poke's* haptics module is similar to the one used in the *Lathe* system; the surface model is shown below in figure 7.2. The grid of control points is modeled as a sheet of masses connected by "locking" springs. Each point is connected by a spring to its up, down, left, and right neighbors, and each spring is capable of being stretched, and then locked in its extended position at each timestep. The springs resist being stretched, but do not restore when the application of external force stops. Control points are constrained to only move in depth.

7.3.1 Surface simulation

Since the surface lies almost entirely behind the $z=30\text{mm}$ plane in this model, we only begin collision testing when the PHANTOM stylus tip is near the surface, at $z \leq 35\text{mm}$. When the stylus tip position $\mathbf{P} = (P_x, P_y, P_z)$ is located close to the surface, we evaluate the Catmull-Rom formulation to check for contact. We use the x-y position of the

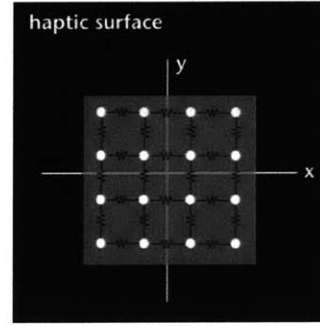


Figure 7.2 Haptic sheet model

stylus to determine the z-height of the surface, Q_z as

$$\begin{aligned} Q_z(s, t) &= \mathbf{S} \mathbf{M}_{CR} \mathbf{C}_z \mathbf{M}_{CR}^T \mathbf{T}^T \\ &= \mathbf{S} \mathbf{M}_{CR} \mathbf{C}_z \bullet \mathbf{T} \mathbf{M}_{CR} \end{aligned}$$

where $s:[0, 1]$ and $t:[0, 1]$ are the surface parameters, and \mathbf{T} and \mathbf{S} are given by

$$\begin{aligned} \mathbf{T} &= \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \\ \mathbf{S} &= \begin{bmatrix} s^3 & s^2 & s & 1 \end{bmatrix} \end{aligned}$$

\mathbf{M}_{CR} is the Catmull-Rom matrix

$$\mathbf{M}_{CR} = \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix}$$

and the geometry matrix \mathbf{C}_z , is determined from the z-values of control points that bracket the x-y location of the stylus tip, as given by

$$\mathbf{C}_z = \begin{bmatrix} C_{z_{i+2,j-1}} & C_{z_{i+2,j}} & C_{z_{i+2,j+1}} & C_{z_{i+2,j+2}} \\ C_{z_{i+1,j-1}} & C_{z_{i+1,j}} & C_{z_{i+1,j+1}} & C_{z_{i+1,j+2}} \\ C_{z_{i,j-1}} & C_{z_{i,j}} & C_{z_{i,j+1}} & C_{z_{i,j+2}} \\ C_{z_{i-1,j-1}} & C_{z_{i-1,j}} & C_{z_{i-1,j+1}} & C_{z_{i-1,j+2}} \end{bmatrix}$$

where C_{ij} is the control point for which $C_{x_{i,j}} \leq P_x < C_{x_{i,j+1}}$, and $C_{y_{i,j}} \leq P_y < C_{y_{i+1,j}}$.

If $P_z \leq Q_z$, the stylus tip is in contact with the surface, and a restoring force is computed and displayed. The surface normal vector at the contact point $Q = (P_x, P_y, Q_z)$ is given by the cross product of the two tangent vectors

$$\frac{\partial \mathbf{Q}}{\partial t} = \frac{\partial}{\partial t} (\mathbf{S} \mathbf{M}_{CR} \mathbf{G} \mathbf{M}_{CR}^T \mathbf{T}^T)$$

and

$$\frac{\partial \mathbf{Q}}{\partial s} = \frac{\partial}{\partial s} (\mathbf{S} \mathbf{M}_{CR} \mathbf{G} \mathbf{M}_{CR}^T \mathbf{T}^T)$$

In areas where the sheet has been greatly deformed, resulting in a very steeply-oriented surface, this simple method of projecting the stylus location onto the surface may generate slightly inaccurate results. As is shown in figure 7.3, where the surface is steeply sloped, the restoring force is computed for penetration relative to a surface point that is *not* the closest surface point to the stylus tip. Consequently, a restoring force of significantly greater magnitude, and inappropriate direction is displayed.

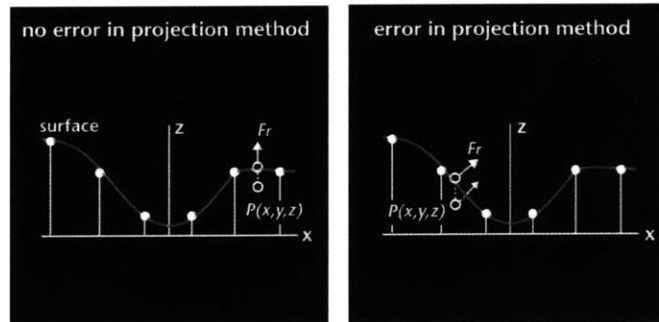


Figure 7.3 Projection method instabilities

To improve the simulation without incurring too much computational expense, several additional measures can be taken. In *Poke*'s implementation, when the computed restoring force magnitude is sufficiently great that we expect the surface point to be inappropriate, nearby surface points are sampled in search of one nearer to the stylus tip. With just a few samples, we can improve the behavior of the simulation; if the haptic servo rate permits a more careful search, we can converge on the correct solution.

7.3.2 Surface deformation

When the surface is being only inspected (but not deformed), the PHANToM stylus tip penetrates the surface by an amount $\delta = (Q_z - P_z) < D$, where D is some preset

threshold distance.

Irrespective of the true location of the contact point relative to its four surrounding control points, computation of the restoring force *magnitude* (and of deformation, also) proceeds as if the contact point were precisely centered among them. The restoring force *direction* is given by the actual surface normal at \mathbf{Q} . Thus, when the stylus tip penetrates the undeformed surface by amount δ , which is less than the preset threshold D , the restoring force is given by

$$\mathbf{F}_R = 4\delta k_1 \hat{\mathbf{n}}_Q$$

where k_1 is the spring coefficient for each of four *implicit* links connecting the contact point to its immediate surrounding four control points, as is shown in figure 7.4. So, before the surface deforms, it feels uniformly springy over its entire area.

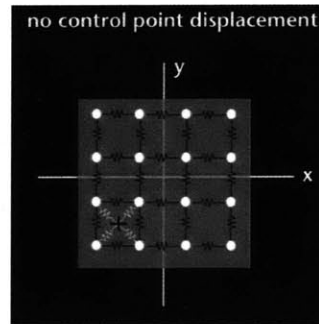


Figure 7.4 Pre-deformation restoring force

When δ increases beyond the threshold distance D , the four \mathbf{G}_{ij} closest to the contact location are displaced *in depth only* by the amount $\delta - D$, where the parameter D is tuned to give the simulation an acceptable behavior. The restoring force now includes the contribution of more springs, since the displaced points begin to tug on their neighbors. This restoring force is given by

$$\mathbf{F}_R = (4\delta k_1 + k_2 \Delta_{\text{neighbors}}) \hat{\mathbf{n}}_Q$$

where k_2 is the spring coefficient for the links between control points, and

$$\begin{aligned} \Delta_{\text{neighbors}} = & (C_{z_i, j-1} - C_{z_i, j}) + \\ & (C_{z_i-1, j} - C_{z_i, j}) + \\ & (C_{z_i-1, j+1} - C_{z_i, j+1}) + \\ & (C_{z_i, j+2} - C_{z_i, j+1}) + \\ & (C_{z_i+2, j+1} - C_{z_i+1, j+1}) + \\ & (C_{z_i+1, j+2} - C_{z_i+1, j+1}) + \\ & (C_{z_i+2, j} - C_{z_i+1, j}) + \\ & (C_{z_i+1, j-1} - C_{z_i+1, j}) \end{aligned}$$

again, when $C_{i,j}$ is the control point such that $C_{x_{i,j}} \leq P_x < C_{x_{i,j+1}}$, and $C_{y_{i,j}} \leq P_y < C_{y_{i+1,j}}$. In this way, as the area of deformation increases, the restoring force displayed to the user also increases. Both k_1 and k_2 are tuned to adjust the compliance of the surface. As the displaced $G_{i,j}$ draw their control points (at a constant rate) to the new location over several timesteps, the surface is felt to deform smoothly (figure 7.5).

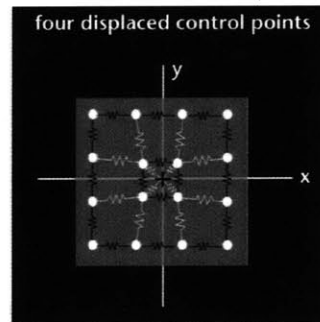


Figure 7.5 Surface deformation

Of course, as the $P_{i,j}$ around the contact location are displaced relative to *their* neighbors, these neighbors will eventually become involved in the deformation also. In general the computed restoring force is based on the number of points being actively displaced at any timestep.

To determine how far the displacement should spread radially after the first four points are displaced, neighboring points are examined in a spiral fashion and displaced as necessary. Links between points, initially of length L , are locked at each timestep as they are stretched, and can extend to a maximum length of

$$(L^2 + D^2)^{1/2}$$

In this way, the surface maintains its new shape rather than relaxing back to the original one.

Without question, this algorithm for surface deformation departs from any model found in nature; but by constraining our control points to displace *only in depth*, we afford many subsequent simplifications in the hologram modeling and minimize the data which will be sent by the haptics module. Though the simulation does not model a known physical material, its behavior is similar to poking a frictionless "pin-block" which has been covered with a stretchable skin (figure 7.6). The Catmull-Rom spline formulation insures that the skinned surface touches all the pins, so when a user pokes it, the resulting deformation behaves intuitively.

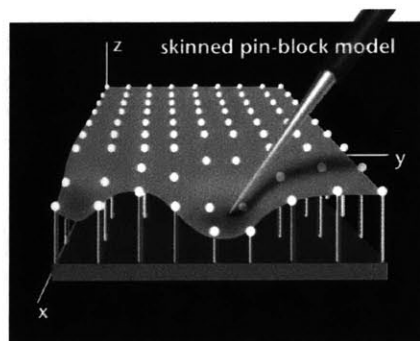


Figure 7.6 Haptic model as a pin-block with stretchable skin

7.3.3 Communicating changes to the holovideo module

Approximately 30 times per second, the control points affected by a user are collected and sent to the holovideo module so that the visual display can be made current. The haptics module assigns to each control point an integer identification value, which corresponds to a point in the holovideo module's own model. A message is assembled which contains the number of updated control points, followed by a list of points' ID values and new z-values, and is sent via ethernet and TCP/IP to the holovideo module. Once the holovideo module receives the message, it translates haptic model changes into hologram changes, and sends them to the visual display.

7.4 Hologram updates

The holovideo module initiates and manages the haptics simulation and performs the hologram computation. It begins with the same initial set of control points as does the haptics module, and this grid of uniformly-spaced control points constitutes a minimal description of the surface which we use for visual display as well. In general, its job is to use changed control points, sent by the haptics module, to update its own representation of the deforming sheet, and to compute corresponding modifications to the holographic image. Like the WRM in the *Lathe* system, *Poke's* holovideo module uses an intermediate representation of the model which aids the translation of haptic changes to holographic ones.

The holovideo module's intermediate representation of the model data is called the "holo-object", and is comprised of primitives called "holo-points", or h_i . These holo-points represent coherent point sources that isotropically emit spherical wavefronts, and are distributed on the surface of the pliable sheet. In the initial holo-object, the holo-points correspond exactly to the haptic module's initial set of control points, and thus model the same vertically-oriented flat sheet 80mm wide and 50mm high, situated at $z=30\text{mm}$ in front of the holovideo output plane. This intermediate representation can be rendered to a graphics window as shown in Figure 7.7 in order to observe a user's actions, or for help in debugging.

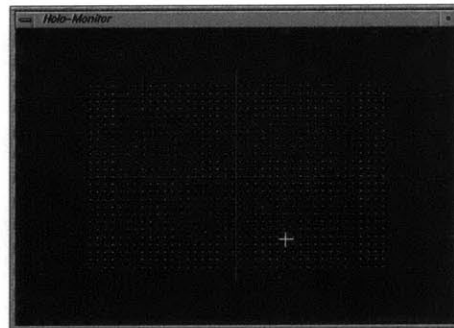


Figure 7.7 Graphics rendering of intermediate "holo-point" representation

As a "bridge" between haptic and holographic representations, each holo-point shares an identification value with its twin control point in the haptic model and is also matched with one of a set of *pre-computed elemental fringes*, as described more thoroughly in Chapter 4.

7.4.1 Using precomputed elemental fringes to assemble a hologram

In essence, each of these pre-computed fringes is a one dimensional holographic pattern capable of imaging a point into the holovideo display volume at some particular z -depth. The fringe's x - y position within the hologram determines the x - y position of the imaged point. A set of 100 elemental fringes is used, sufficient to express points at 0.5mm depth increments between $\pm 50\text{mm}$ of holovideo's output plane.

In order to *arbitrarily* update the underlying model geometry and reflect these changes in the final hologram at interactive rates, the *incremental computing* algorithm was developed. As described in Chapter 4, this method uses the above-mentioned table of pre-computed (16-bit) elemental fringes, which are associated with model primitives (in our case, holo-points). The depth of each holo-point in our model is used to determine the elemental fringe appropriate to represent it. Then, the initial 32-bit *working* hologram of an object is assembled by combining all elemental fringes

needed to reconstruct the images of all holo-points at their prescribed spatial locations. *This method provides the means for Poke to rapidly propagate arbitrary changes in the haptic model, through the holo-points, into the working hologram, and finally into an 8-bit normalized hologram suitable for our holovideo display.*

7.4.2 Updating the hologram as holo-points change

In the holovideo module's main loop, first, incoming control point changes are used to update all corresponding holo-points. Since its model is comprised only of holo-points that are co-located with the control points, we simply copy incoming changed control point z-values to their twin holo-points. If we were to densely populate the surface with holo-points instead, we would need to evaluate the Catmull-Rom formulation to update all effected holo-points *not* co-located with a control point. For the purposes of demonstrating Poke, we chose to implement the simpler (but certainly less visually impressive) surface representation. Also, because the pre-computed elemental fringes can only image points between -50 and 50mm at half-mm increments, this assignment effectively *depth quantizes* the location of each holo-point. Of course, we could compute a fringe table to represent a much larger depth range, and a smaller depth increment; however, this range and resolution are sufficient for the task of representing a sheet of pliable material with shallow, user-administered dimples.

Finally, after having their positions updated, all changed holo-points are re-quantized in z and y, and reassigned to hololines. These updated holo-points are then used to feed the hologram computation which proceeds as follows: 1) changed holo-points have their old fringes subtracted from their old lines in the working hologram, 2) new elemental fringes are assigned to each changed holo-point, 3) the span of samples to use in each elemental fringe and the hologram samples each fringe should contribute to are determined, 4) appropriate spans of new elemental fringes are added to the working hologram in the right place, and 5) corresponding new lines in the final *display hologram* are computed by normalizing all changed working hologram lines to 8 bits. In *Poke*, each of these steps are divided between two processors, each operating on either even or odd hologram lines. A hologram of the holo-points shown in Figure 7.7, generated using a wavelength of 0.5mm, is shown below in Figure 7.8.

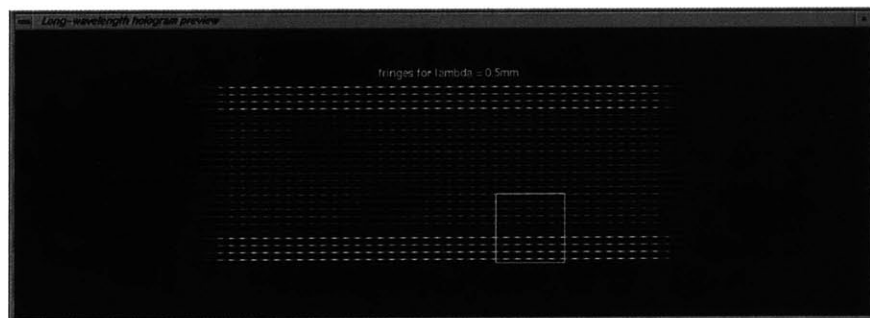


Figure 7.8 Example poke hologram using lambda = 0.5mm

The particular hardware architecture we use also requires us to subsequently byte-swap the display hologram, and flip every other set of 18 hologram lines from right to left (to match the scan convention of the holovideo display). These operations are performed before sending the 36MB display hologram over a HIPPI link to the holovideo display server, *holoPut*. An overview of all model representations used throughout the system as described thus far is shown below in figure 7.9.

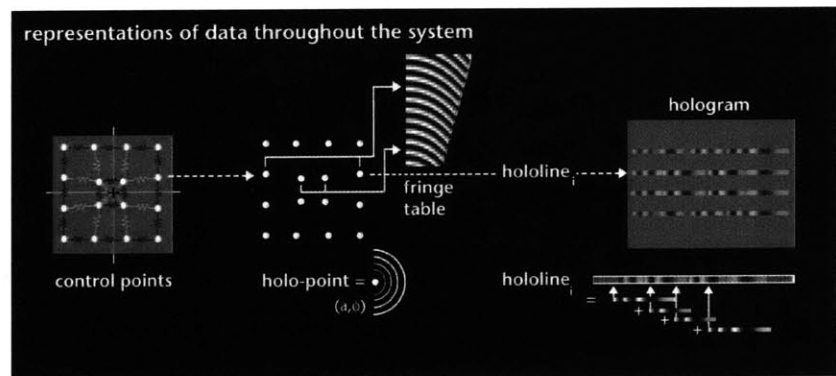


Figure 7.9 Different models used throughout the *Poke* system

7.5 Results and discussion

7.5.1 Update

The performance of the entire system has much to do with the various computational platforms used in its implementation. Specifically, the haptics module (which again uses the PHANToM haptic interface for force display) runs on a 195MHz SGI Octane, computes the haptics simulation and serves updated control point information to the holovideo module. The haptics servo loop runs at an average of 5KHz, and we can dial the frequency with which updated control points are delivered to the holovideo module for recomputation. Currently, these data are sent roughly 30 times per second, but, notably, we constrain this message to include no more than 100 points per transfer. Like in *Lathe*, if more than this quota of points change, their update will not be propagated to the hologram unless they are subsequently changed again. As long as a user doesn't operate too fast, the hologram is able to reflect the haptic model very well.

The holovideo module runs on a dual 250 MHz processor SGI Onyx, establishes a client connection to the haptics module over ethernet using TCP/IP, and then uses both processors to perform hologram updates based on the haptic model changes it receives. Given that the holovideo module is never called upon to update the contribution of more than 100 holo-points at a time, and many of these points are guaranteed to be on the same hololine, hologram recomputation can be accomplished relatively quickly.

The holovideo module delivers updated holograms to the holovideo display server *holoPut*, running on the Cheops Imaging System, via a 100MB/sec HIPPI link.

Currently, since we are transferring the entire 36MB hologram to Cheops for every update, each HIPPI transfer requires a fixed cost of 0.6s. Thus, if all other computation were instantaneous, we could offer a visual update of one and a fraction frames per second at best. System performance could be considerably improved by transferring only the changed hologram lines (in the case of *Poke* where only 100 holo-points may change per iteration, we would update at most 12 lines at a time instead of 144); such a modification might reduce transfer time by a factor of ten. *holoPut*, running on Cheops, receives and transfers incoming hologram data via the Nile bus directly to the output cards for display. While this transfer hasn't been timed, it is negligibly small.

Like *Lathe*, the system shows substantial lag between a user's poking input and the visual manifestation of the change. Of course, appropriate changes to the haptic model can be immediately felt; the delay between user input and visual update is about 2 seconds, on average, as indicated in Figure 7.10. An initial latency of as much as 0.03s is present since haptic model control points are not continuously transmitted. Then, lag accrues in several places: in the incremental hologram computation (~0.1s), normalization (~0.4s), conditioning (byte-swapping and line flipping) (~0.3s), and transmission to Cheops (~0.6s). The visual frame rate that *Poke* finally achieves is between 0.25 and 0.3 frames per second; the haptic update rate is quite high at 5kHz.

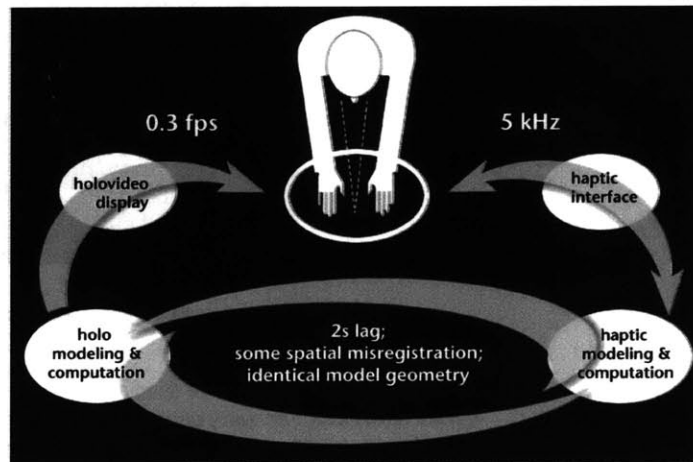


Figure 7.10 *Poke*: improved spatial synchrony but degraded temporal synchrony

7.5.2 Look and feel

In light of this simple visual representation used by *Poke*, the computer graphic and holographic renderings are not exactly bursting with pictorial cues to depth and layout. *Poke's* holographic pliable surface is implied by an array of reconstructed points; the shape of this surface is clearly visible with binocular vision and motion parallax but would not be evident to a monocular, unmoving viewer. Simplifying the model was essential to demonstrating image update rates that would support

interaction. In retrospect, given the speed with which our hologram can be recomputed, we might have used a more elaborate visual representation and still have achieved a nominally useful update rate. Happily, as it turns out, the simple visual representation used in the *Poke* system and the shallow depths represented are more forgiving of the intermittent spatial misregistration that occurs due to lag.

Because all hologram lines are not necessarily normalized to the same value range (a consequence of our computation method), they may vary noticeably in brightness. This visual artifact is amplified by the independent gains of the 36 modulator channels used in the system; thus, holographic images exhibit nonuniform brightness from top to bottom. Further, as a consequence of the x-y scanning in the opto-mechanical output stage of the system, there is a slight bow-scan distortion in the image plane. Since our image forms a rectangle just in front of the image plane, this distortion begins to affect its shape. Finally, an image showing the holographic image and an interacting user is shown in Figure 7.11.



Figure 7.11 Interacting with *Poke*'s haptic hologram

In summary, with a cluster of specialized tools, we have assembled a suite of primitive demonstrations of an early computational plastic. These demonstrations allow us to inspect and modify an *apparent* rudimentary, programmable, material surface. We might advance from here along several interesting fronts: improving the palette of visual and haptic material qualities we can ascribe to surfaces, improving image characteristics of our visual display and shrinking the obtrusive physical mechanisms of all displays. Additionally, using faster and specialized computational subsystems, and allowing both processor and framebuffer to share the same bus would substantially improve system update, and are developments already underway. Most interesting of course, would be new experiments with this primitive computational plastic which are more creative and less mechanistic, and which begin to inspire us with its potential for serious application, expression, and play.

8. Discussion and longer view

8.1 Material or just simulated material?

In this dissertation, we have periodically referred to *mimetic* display; here, we deploy this term more loosely than it might be used in a more formal artistic or literary context. Strictly speaking, the amalgam of computation and display we envision would be both *mimetic* and *illusory*, capable of portraying and embodying the character of a simulation and of fooling the eye and hand with sharply realistic output. Here, we will clarify these terms and examine their underlying issues a bit.

8.1.1 Mimesis and behavioral believability

Research in the areas of artificial life, lifelike autonomous agents, and character-based animation to some extent embrace ideas of behaviorally *mimetic* modeling and rendering. Common in these disciplines is the desire to display behavior that appears to be *believable*, and mimesis is the key to its success. Essentially, mimetic modeling and rendering manifest subtle details and cues that evoke the deeper character of the simulation, and through which a viewer might emotionally or logically experience or empathize with whatever is depicted. A mimetic depiction of, for instance, *evil* would not be so powerfully accomplished by a picture of someone evil. Rather, to embody, characterize, and animate evil, many not-so-literal cues might be used to evoke it: threatening, shadowy shape, perhaps insect-like, unpredictable movements, and so on. To mimic *wheat*, one might use infinite amber, or slender, arching curves, and slow swaying movement. Whether this modeling and rendering is applied through painting, acting or performance, through the meaning, rhythm, and sound of words, or through the movement and expression of animated characters, they offer us the opportunity to connect to something *other than ourselves*; and this is the source of their importance and power.

Work on lifelike autonomous agents has determined several requirements for creating believable synthetic characters; some of these requirements are essentially channels through which mimetic modeling and rendering might occur. Bryan Loyall¹⁷⁷ notes seven such requirements in his dissertation on believable agents. Slightly reworded here for clarity, these requirements are attributes that a character should display to be believable: personality, emotion, self-motivation, ability to change, sociality, consistency of expression, and apparent *life*. (This last requirement is quite broad and we'll not discuss the many means toward achieving it here.) Of course, these encapsulations — and there are likely to be more yet-unqualified ones — have established roots in character animation, and truly ancient ones based in story-telling.

Importantly, we need not be displaying a *character* to consider using mimetic modeling and rendering. These concepts might enrich any scenario, scene, or rendered element we display. Because they often make the difference between casual observation and a deeper cognitive and emotional engagement, they are important to consider. Moreover, even if a display fails to present a perfect illusion (which is likely to be the case), behavioral mimesis can be a powerful mechanism for suspending our disbelief.

8.1.2 Illusion and physical believability

Illusory modeling and rendering, on the other hand, has to do with how closely a displayed element can be made to resemble a physical counterpart. Such *physical-bievability* poses many technological problems for combinations of computation and display — some of them easy, and some not so. From experiments in Artificial and Mixed Reality applications, Drascic *et al.*¹⁷⁸ have also noted a list of some of these easy and hard problems, which we merge with our own. These issues affecting a simulation's *physical believability* can be categorized into three groups: *spatio-temporal calibration and user metric errors*, *display limitations*, and *reality modeling limitations*. These problems must be better qualified and addressed in order for the boundary between a purely simulated material and a physical one to be blur. We'll mention each of these categories in turn.

8.1.3 Spatio-temporal calibration and user metric errors

First, spatio-temporal calibration and user metric errors come from a failure to match displayed output to the geometry of a display space, and a failure to render unnoticeable the delay between user input and system response. These types of errors occur in modeling of computer graphic camera parameters and stereo rendering, in misaligned or poorly calibrated optical capture geometries, poorly measured eye position and head movement, from all sources of latency and lag in a system, and from multiple, misregistered display outputs. Many of these problems (but for perfect tracking, latency, and lag) can currently be addressed by careful design, and by allowing rendering and capture camera parameters to be adjusted for the needs of different viewers.

8.1.4 Display limitations

Next are current display limitations, most of which can be expected to improve over time. Among the limitations of current visual displays are things like: limited field of view, reduced resolution and spatial frequency content, limited dynamic range, contrast ratios and color gamut, noticeable depth quantization, accommodation and convergence mismatch, and the inability to match a viewer's depth of focus.

Haptic displays currently suffer even more limitations. The skin is a complicated organ populated with many different kinds of sensors, and the muscles and joints house even more. Thus, the many kinds of mechanical and thermal stimulation we can sense gives rise to rich and varied perceptions; it suffices to say that we don't yet have a display capable of delivering such elaborate sensory stimulation to the body. Moreover, since tactile and force senses cannot be remotely stimulated, any useful device requires substantial contact with our bodies. Unless using an "encountered-type" haptic device, or using the device like a familiar hand-held tool, this constant contact with the haptic display technology can seem contrived at best, and extremely annoying at worst.

8.1.5 Reality modeling limitations

The last category is comprised of *reality modeling limitations*, many of which have not yet been qualified. These concerns go to the heart of modeling convincing illusions. Some are easy to pinpoint, and we have discussed them previously in Chapter 5; among the most important, occlusion violations and volume violations are both easily

manifested in spatial display systems, but not encountered in our ordinary real-world interaction. As such, both of these very difficult problems significantly degrade the impression of a solid, three-dimensional displayed output.

Difference in illumination between real and simulated scenes is something we are particularly good at noticing. Some of the issues involved in rendering realistic light and shadow are beginning to be formalized, largely as a result of the film industry's push to perfectly composite rendered scenes with photographed ones. In current practice, to aid the generation of simulated sequences, physical scene lighting for a new shot is captured by photographing both specular and diffuse spheres in the scene, and also by keeping detailed written notes about light placement, brightness and color. Scene lighting is subsequently reconstructed in the computer graphics studio by modeling according to the notes and iteratively adjusting available parameters to visually match a rendered specular and diffuse spheres to the photographed ones. In this way, radiance distribution and color are adjusted to create as seamless a match as possible between composited elements¹⁷⁹.

In the fields of Computer Graphics and Mixed Reality, researchers are investigating the use of photometric modeling techniques^{180,181} for more automated merging of virtual objects and real scenes. Among the various approaches to the issue are techniques to determine reflectance models for real-world surfaces and radiance distributions in real scenes. Many of these techniques take captured images of real scenes as input, and then use this information to model and illuminate synthetic objects. Other techniques address the importance of generating shadows whose character matches others in the real scene.

Also, the perception of shape, lightness and color are interlinked^{182,183}. Recent work suggests that knowledge of light source position¹⁸⁴, and some basic inherent knowledge of the chromatic effects of mutual illumination¹⁸⁵ (the light reflecting between object surfaces) may mediate our perception of three dimensional shape and surface color. While, traditional computer graphic ray-tracing techniques render convincing specular reflection and dispersionless refraction, they do little to model other radiation reflected or emitted from objects in a scene. Radiosity techniques¹⁸⁶ do provide a way to model the play of light between surfaces but are more complicated, more computationally expensive, and thus not commonly used. These techniques may offer a way to simulate an appearance of light bouncing between real and rendered surfaces in a scene; such a subtle visual cue might present an effective way to marry real and simulated elements.

Certainly, our holographic images exhibit a wholly different visual character than any part of the world they reconstruct in. Because they rely on the phenomena of diffraction to produce their spatial images, their spectral output may always be given to slight misbehavior. Nevertheless, if they can be used to present highly resolvable spatial image detail to which we can freely converge and accommodate, they may still represent a *physically believable* material — just one with a unique and recognizable visual character.

8.2 Where we are

8.2.1 Our residence within the broad class of computational plastic technologies

Our introduction stated that a good prototype of computational plastic should have several basic properties: it must allow us to interactively shape and reshape a surface, must admit programmable appearance and behavior, must be available to the senses just like a physical material, and when fashioned into an object, must offer some unique physical affordances for use.

Together, our three demonstrations, *Touch*, *Lathe*, and *Poke*, have primitively met these basic requirements; thus we will consider haptic holography an early instantiation of computational plastic. Full parallax *Touch* demonstrated the most tightly coupled multimodal simulation, with force cues appearing to arise from contact with the holographically displayed surfaces in all but a few pathological arrangements of hand, tool, and visual output. With its constrained carving, *Lathe* allowed interactive reshaping of a surface, though spatial registration and the temporal synchrony of visual and haptic outputs were slightly compromised. *Poke*, with its arbitrarily deformable surface gave us truly programmable look and feel of the multimodal output. As shown in Figure 8.1, *Poke* becomes an early resident in the spectrum of technologies comprising computational plastic.

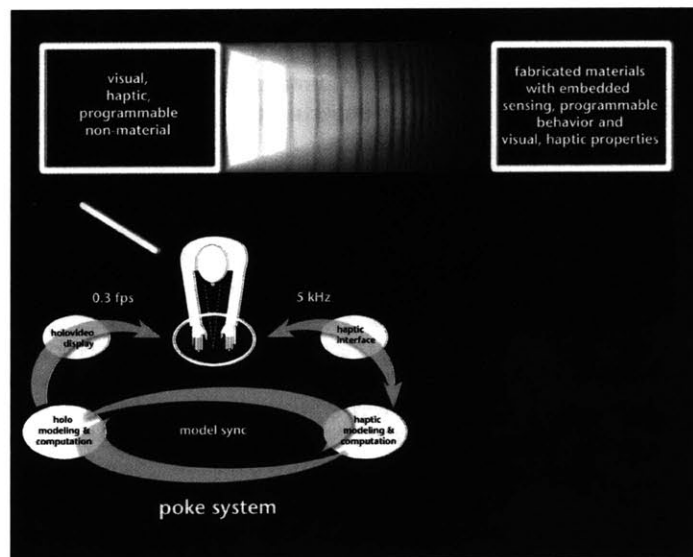


Figure 8.1 *Poke*'s early place in the spectrum of computational plastic technologies

8.2.2 What else will it take to assemble our ideal holo-haptic system?

Computational holography is notoriously bandwidth-greedy. *Poke*'s visual frame rate is too low and the lag between haptic and visual updates is too great as a consequence of this basic current need for greater computation, communication, and modulation

bandwidth. These issues fairly well encapsulate the major *non-computational* challenges to building the next generation of holographic video displays.

Consider for instance, that to engineer a 0.5m x 0.5m full parallax display with an angle of view of 60 degrees (a size and view angle analogous to high-end LCD computer monitors at the time of this writing), a holographic fringe pattern would need to represent a maximum spatial frequency f_{max} of approximately

$$f_{max} = \frac{2 \sin 30^\circ}{\lambda} \cong \frac{2 \times 0.5}{0.5 \times 10^{-6}} = 2 \times 10^6 \frac{cy}{mm}$$

and the required sampling frequency f_s would have to be at least twice that

$$f_s \geq 2f_{max} = 4 \times 10^6 \frac{cy}{mm}$$

in order to avoid aliasing artifacts. Thus, the horizontal and vertical resolution of such a display would each be 2×10^9 samples, requiring every frame of holographic video to contain 4×10^{18} samples in total. Further, a full-color display would require that many samples *for each color channel*. And we would expect that, depending on which techniques are used to compute the fringe pattern and for gamma correction, one byte per sample may not be enough to represent, say, 256 distinct levels of greyscale in the final image.

All these considerations roughly put our requirement at a currently astonishing 10^{20} bytes per frame of holographic video! The spatial light modulator employed by this display system must be able to accommodate this enormous amount of data and the high sampling frequencies resident in the fringe patterns. Further, we need to recompute our fringe pattern at least 30 times every second, and consequently, to communicate 3×10^{21} bytes per second through the computational, electronic, opto-electronic or mechatronic subsystems that constitute this hypothetical display system. These numbers are prohibitive in terms of contemporary technology; Figure 8.2 notes the disparity between technological means, current at the time of this writing, and those required by such a hypothetical full parallax holovideo display.

In a practical sense, these currently discouraging computation, communication, and modulation bandwidth requirements may argue instead for the development of large format *stereogram* displays in the near term, with static, pre-printed or -computed diffractive elements used to optically relay two-dimensional views to the viewer. In this case, for the same 0.5m x 0.5m display, a very high-resolution stereographic frame comprised of 1k x 1k two-dimensional views, each containing about 10k x 10k pixels (such that the image plane is sampled at roughly 500dpi), could require on the order of 10^{14} bytes per frame — without giving up vertical parallax.

	future requirements	current specifications
computation	> ~Zetaflop (10^{12} Gigaflops) processor speed (single or aggregate)	~GHz scale processor speed
communication	> ~Zetabytes/s IO channel	1-10 GB/s internal bus speeds ~100MB/s (HIPPI spec) external I/O channel ~100MB/s -1GB/s (InfiniBand spec) internal/external data channel
modulation	~ Exabyte resolution + lower (~10ms) switching speed -or- lower spatial resolution + higher switching speed	AOMs: high res; analog; nonuniform response; expensive; 55MHz/channel LCDs: low res; low (<100Hz) frame rate DMDs: low res (~400k pixels); high frame rate (~50kHz) other MEMs: early, experimental

Figure 8.2 The gulf: current technological specs and requirements of future systems

As computation, sensing, and display technology improve, we expect that the design trends for future spatial display systems will seek to make their trappings *less apparent* to people using them. In the much longer term, we imagine that these technologies will also migrate into fabrication materials themselves (Figure 8.3), thus permitting experimentation with *physical instantiations* of computational plastic materials.

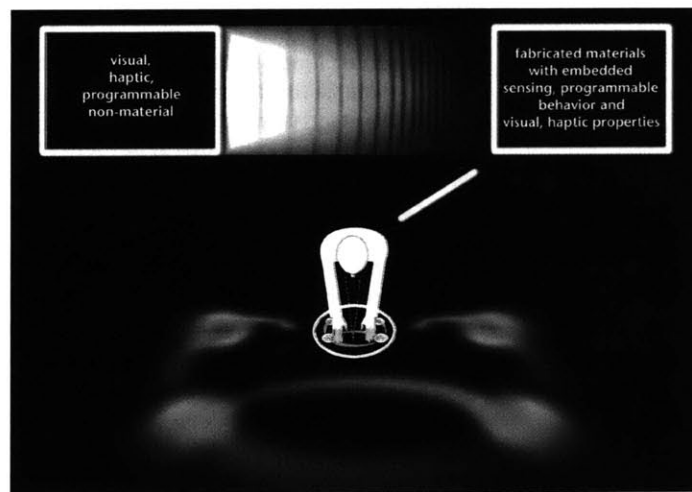


Figure 8.3 Migrating sensing, display, computation, and actuation into fabrication materials

8.3 Summary and contributions

Located in perhaps the world's best facility for creating holograms, we have the ability to create many kinds of visual spatial displays here. We have long watched visitors apply the "touch test" to aerial images, and observed their apparent delight when finding nothing with their fingertips. So this entire project began with some casual musing: could we double this delight by producing an encounter for those fingertips? We also heard automotive designers say that, while observing light play across a body design was important, feeling the shape with gloved, moving hands and moving arms was also crucial to their understanding. Thus, we began to think about the *usefulness* of ascribing haptic properties to our images. Finally, after experimenting with the combination of holo-haptic displays and computational modeling, the surfaces we assembled began to take on a decidedly *material* quality: visible, touchable, and also *programmable* — with a bit of an unearthly character. Thus our notion of a computational plastic unfolded, with our work situated at one extreme of its definition. By folding holography into a mixture of display and modeling, we provide a new kind of physically and behaviorally mutable substrate for practical and creative use.

Consequently, and as its most general contribution, this work marks a notable turning point for holography. Within the last decade the literature has reported that advances in display technology and computational methods have ushered interactive holographic technology to within steps of its consummate form. This may be more or less true, depending upon how one defines "interactive", and what one expects from a physically mimetic display. In contrast, we would argue that holography still has a great distance to go. For those childhood imaginations fed by sci-fi TV and hollywood, even a firm respect for the genuinely difficult problems populating the gap between hologram-fiction and -reality does little to temper the anticlimax of state-of-the-art, sparsely rendered and noisy images, merely rotating or translating in ratchety increments. However, in this work, rather than just *displaying* a holographic image or *updating* that image as the engineering achievement, we have now pressed holography into actual service in a primitive interactive application, and have allowed it to recede a bit from the center of our engineering attention. Without question, we'll have to bring it back! But to catch a glimpse of its practical use is both meaningful and noteworthy.

As a more practical contribution, we have demonstrated a successful alloy of holographic and haptic elements; when spatially overlapped and metrically registered, these modal elements combine to give the impression of a material surface situated within the workspace. Display limitations are still numerous; the holographic viewzone provides only a window through which to see the image, force display is only correct for contact at the tip of the device's end-effector. But within the bounds of these and other technological restrictions, we have presented a suite of simple but compelling holo-haptic demonstrations.

We have also offered a general behavior-based classification of interactive space (*manipulatory, ambulatory, and reconnaissance space*) which provides an organization of generalized space-specific behaviors and the space-specific sensory cues that best inform them. Designing within this framework can collapse a seemingly unbounded

list of human factors considerations into one holistic and ecologically-sound package that informs sensorial and interaction design, and might influence information design too. As it stands, this spatial framework can help us to investigate and weigh the usefulness of different sensing, displays, and controllers and the repertoire of actions they admit. It can also keep sensorial and interaction design more tightly coupled to our sensorimotor abilities for specific spatial tasks.

As supporting technology, we have contributed a new algorithm for locally and incrementally updating holographic fringe patterns from a set of precomputed elements, rather than entirely recomputing them. This incremental computing method allows us to rapidly and interactively change the holographic image to reflect underlying model changes. Currently, it relies upon isotropic spherical emitters as holographic scene primitives. However, the method may be extended to include other primitive representations as well, providing that *transformation* of those primitives can be expressed with simple transformations of their corresponding diffractive patterns.

8.4 Onward

In this dissertation, we have tried to strike a delicate balance between establishing our work's contribution, and being frank about its primitive manifestation. There is still an enormous gap between our stopping-point and the demonstration of beautiful, interactive computational plastic that blurs the distinction between simulation and material enough to suspend our disbelief. And there are still its many practical and meaningful uses to be discovered.

We'll conclude by pointing out some of the more obvious *computational* topics whose investigation will help us move further along the way. Foremost, in the area of electronic holography, an arsenal of realistic rendering techniques should be constructed — techniques which can produce an amplitude, as a function of angle, for all holographic primitives populating the scene. These techniques should incorporate as many pictorial cues to depth and layout as possible, and should admit flexible modeling of scene illumination.

Since new *material surfaces* are what we are essentially creating, new kinds of holographic primitives might model the variable and sensuous way light and matter interact at this interface. Even if these primitives are expensive to simulate or compute, they might be generated only once and tabled like our own simple spherical emitters. Certainly, to match a more careful visual rendering of a surface, haptic modeling of material interactions will have to progress. Upon seeing a material, we form certain expectations about how it should feel; thus visual and haptic modeling and rendering will have to proceed together to produce convincing combined output.

Truly, the most creative and interesting future work is in inventing new ways to *use* a holo-haptic (or other) programmable plastic; rendering *things* out of a new malleable material, situating them in physical space, observing them, interacting with them, keeping company with them? What emerges from active experimentation, we expect, will be quite interesting and unexpected.

9. References

Chapter 2. Perception and Action

- [1] Baecker, R.M., Grudin, J., Buxton, W.A.S., and Greenberg, S., Readings in *Human-Computer Interaction: Toward the Year 2000*. San Francisco: Morgan Kaufmann Publishers. (1995).
- [2] Shedroff, N., Information Interaction Design: A Unified Field Theory of Design. In R. Jacobson (Ed.), *Information Design*, Cambridge: MIT Press. (1999).
- [3] Hershenson, M., *Visual space perception: a primer*. MIT Press, Cambridge. (1999).
- [4] Boring, E.G., *Sensation and perception in the history of experimental psychology*. New York: Appleton-Centurey-crofts. (1942).
- [5] Gibson, J.J., *The perception of the visual world*. Houghton Mifflin, Boston. (1950).
- [6] Thomas, F., and Johnston, O., *The Illusion of Life: Disney Animation*, Hyperion. (1995).
- [7] Kaufman, L., *Sight and mind: an introduction to visual perception*, Oxford University Press, New York. (1974).
- [8] Hochberg, J., Perception. In J.W. Kling and L.A. Riggs (Eds.), *Handbook of experimental psychology* 3rd ed., 396-550. Holt, Rinehart & Winston, New York. (1971).
- [9] Arditi, A., Binocular vision. In K.R. Boff, L. Kaufman, and J.P. Thomas (Eds.), *Handbook of perception and human performance*, Vol. 1, 1-41. Wiley, NY. (1986).
- [10] Bulthoff, H.H., and Mallot, H.A., Interpreting depth modules: Stereo and shading. *Journal of the Optical Society of America, A*, 5, 1749-1758. (1988).
- [11] Dees, J.W., Accuracy of absolute visual distance and size estimation in space as a function of stereopsis and motion parallax. *Journal of Experimental Psychology*, 72, 466-476. (1966).
- [12] Doshier, B.A., Sperling, G., and Wurst, S.A., Tradeoffs between stereopsis and proximity luminance covariation as determinants of perceived 3D structure. *Vision Research*, 26, 973-990. (1986).
- [13] Fisher, S.K., and Cuiffreda, K.J., Accommodation and apparent distance. *Perception*, 17, 609-621. (1988).
- [14] Gogel, W.C., Convergence as a cue to absolute distance. *Journal of Psychology*, 52, 287-301. (1961).

- [15] Graham, M.E., and Rogers, B.J., Simultaneous and successive contrast effects in the perception of depth from motion-parallax and stereoscopic information. *Perception*, 11, 247-262. (1982).
- [16] Ono, H., Rogers, B.J., Ohmi, M., and Ono, M.E., Dynamic occlusion and motion parallax in depth perception. *Perception*, 17, 255-266. (1988).
- [17] Terzopoulos, D., Integrating visual information from multiple sources. In A. Pentland (Ed.), *From pixels to predicates*, 111-142. Norwood, NJ: Ablex. (1986).
- [18] Uomori, K., and Nishida, S., The dynamics of the visual system in combining conflicting KDE and binocular stereopsis cues. *Perception & Psychophysics*, 55, 526-536. (1994).
- [19] Wallach, H., Moore, M.E., and Davidson, L., Modification of stereoscopic depth perception. *American Journal of Psychology*, 76, 191-204. (1963).
- [20] Wallach, H., and Karsh, E.B., Why the modification of stereoscopic depth-perception is so rapid. *American Journal of Psychology*, 76, 413-420. (1963).
- [21] Foley, J.M., Stereoscopic distance perception. In S.R. Ellis, M.K. Kaiser, and A.C. Grunwald, (Eds.), *Pictorial Communication in Virtual and Real Environments*, London: Taylor & Francis. (1991).
- [22] Foley, J.M., Binocular distance perception: Egocentric distance tasks. *Journal of Experimental Psychology: Human Perception and Performance*, 11, 133-149. (1985).
- [23] Stubenrauch, C.F., and Leith, E.N., Use of a hologram in depth-cue experiments. *Journal of the Optical Society of America*, 61, 1268-1269. (1971).
- [24] Cutting, J.E., and Vishton, P.M., Perceiving Layout and Knowing Distances: The Integration, Relative Potency, and Contextual Use of Different Information about Depth. In W. Epstein and S. Rogers, (Eds.), *Handbook of Perception and Cognition: Perception of Space and Motion*, Vol. 5, 69-117. Academic Press, San Diego. (1995).
- [25] Da Silva, J.A., Scales of perceived egocentric distance in a large open field: Comparison of three psychophysical methods. *American Journal of Psychology*, 98, 119-144. (1985).
- [26] Bishop, P.O., Vertical disparity, egocentric distance and stereoscopic depth constance: A new interpretation, *Proceedings of the Royal Society of London B*, 237, 445-469. (1989).
- [27] Loomis, J.M., Da Silva, J.A. Fujita, N., and Fukusima, S.S., Visual space perception and visually directed action. *Journal of Experimental Psychology: Human Perception and Performance*, 18, 906-921. (1992).

- [28] Norman, J.F., Todd, J.T., Perotti, V.J., and Tittle, J.S., The visual perception of three-dimensional length. *Journal of Experimental Psychology: Human Perception and Performance*, 22, 173-186. (1996).
- [29] Nagata, S., How to reinforce perception of depth in single two-dimensional pictures. In S.R. Ellis, M.K. Kaiser, and A.C. Grunwald, (Eds.), *Pictorial Communication in Virtual and Real Environments*. London: Taylor & Francis. (1991).
- [30] Trevarthan, C.B., Two mechanisms of vision in primates. *Psychol. Forschung*, 31, 299-337. (1968).
- [31] Schneider, G.E., Two visual systems: brain mechanisms for localization and discrimination are dissociated by tectal and cortical lesions. *Science*, 163, 895-902. (1969).
- [32] Ungerleider, L.G., and Mishkin, M., Two cortical visual systems. In D.J. Ingle, M.A. Goodale, and R.J.W. Mansfield, (Eds.), *Analysis of visual behavior*, 549-586. MIT Press, Cambridge, MA. (1982).
- [33] Lettvin, J.Y., Maturana, H.R., McCulloch, W.S., and Pitts, W.H., What the Frog's Eye Tells the Frog's Brain. *Proceedings of the IRE*, Vol. 47, No. 11, pp. 1940-51. (1959). Reprinted in McCulloch, W.S., *Embodiments of Mind*. MIT Press. (1965).
- [34] Sparks, D.L., and May, L.E., Signal transformations required for the generation of saccadic eye movements. *Annual Review of Neuroscience*, 13, 309-336. (1990).
- [35] Gibson, J.J., *The senses considered as perceptual systems*. Houghton Mifflin, Boston. (1966).
- [36] Gibson, J.J., On the analysis of change in the optic array in contemporary research in visual space and motion perception. *Scand. J. Psychol.*, 18, 161-163. (1977).
- [37] Goodale, M.A., and Milner, A.D., Separate visual pathways for perception and action. *Trends Neurosci.*, 15, 20-5. (1992).
- [38] Milner, D.A., and Goodale, M.A., *The Visual Brain in Action*. Oxford Psychology Series No. 27, Oxford University Press. (1995).
- [39] Aglioti, S., Goodale, M.A., and DeSouza, J.F.X, Size-contrast illusions deceive the eye but not the hand., *Curr. Biol.*, 5, 679-685. (1995).
- [40] Jeannerod, M., A Dichotomous Visual Brain?, *Psyche, An interdisciplinary journal of research on consciousness*. (1999).
- [41] Jeannerod, M., *The cognitive neuroscience of action*. Oxford, Blackwell. (1997).

- [42] Faillenot, I., Toni, I., Decety, J., Gregoire, M.C., and Jeannerod, M., Visual pathways for object-oriented action and object identification. Functional anatomy with PET. *Cerebral Cortex*, 7, 77-85. (1997).
- [43] Ghez, C., Posture. In E.R. Kandel, J.H. Schwartz, T.M. Jessell, (Eds.), *Principles of Neural Science*, Norwalk Connecticut: Appleton & Lange. (1991).
- [44] Ghez, C., The Control of Movement. In E.R. Kandel, J.H. Scharz, T.M. Jessell, (Eds.), *Principles of Neural Science*, Norwalk Connecticut: Appleton & Lange. (1991).
- [45] Oman, C.M., Sensory conflict in motion sickness: an Observer Theory approach. In S.R. Ellis (Ed.), *Pictorial Communication in Real and Virtual Environments*. London: Taylor & Francis. (1991).
- [46] Philbeck, J.W., Visually directed walking to briefly glimpsed targets is not biased toward fixation location. *Perception*, 29, 256-272. (2000).
- [47] Okabe, A., Aoki, K., and Hamamoto, W., Distance and direction judgement in a large-scale natural environment: Effects of a slope and winding trail. *Environment and Behavior*, 18(6) 755-772. (1986).
- [48] Lederman, S.J., Klatzky, R.L., Collins, A., and Wardell, J., Exploring environments by hand or foot: Time-based heuristics for encoding distance in movement space. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, Vol. 13, No. 4, 606-614. (1987).
- [49] Acredolo, L., Small- and large-scale spatial concepts in infancy and childhood. In L. Liben, A. Patterson, & N. Newcombe (Eds.), *Spatial representation and behavior across the life span*, 63-82. New York: Academic Press. (1981).
- [50] Garling, T., Book, A., and Lindberg, E., Adults' memory representations of the spatial properties of their everyday physical environment. In R. Cohen (Ed.), *The development of spatial cognition*, 141-184. Hillsdale, NJ: Erlbaum. (1985).
- [51] Downs, R.J., and Stea, D., Cognitive maps and spatial behavior. *Image and Environment*, Chicago: Aldine Publishing Company, 8-26. (1973).
- [52] Siegel, A.W., and White, S.H., The development of spatial representations of large-scale environments. In H.W. Rees (Ed.), *Advances in child development and behavior*, 10-55. New York: Academic Press. (1975).
- [53] Golledge, R.G., Place recognition and wayfinding: making sense of space. *Geoforum*, 23, 199-214. (1992).
- [54] Darken, R.P., Allard, T., and Achille, L.B., Spatial Orientation and Wayfinding in Large-Scale Virtual Spaces: An Introduction. *Presence Vol. 7, No. 2*, 101-107. (1998).

- [55] Jeannerod, M., Intersegmental coordination during reaching at natural objects. In J. Long & A. Baddeley (Eds.), *Attention and Performance IX*. Hillsdale, NJ: Earlbaum. (1981).
- [56] Jeannerod, M., The timing of natural prehension movements. *Journal of Motor Behavior*, 26, 235-254. (1984).
- [57] Wing, A.M., Turton, A., and Fraser, C., Grasp size and accuracy of approach in reaching. *Journal of Motor Behavior*, 18, 245-260. (1986).
- [58] Klatzky, R.L., Lederman, S.J., and Purdy, K.A., When is vision useful during a familiar manipulatory task?. *Proceedings of the ASME Dynamics Systems and Control Division*, DSC-Vol. 58, ASME. (1986).
- [59] MacKenzie, C.L. and Iberall, T. *The Grasping Hand*. Amsterdam: North-Holland (Elsevier Science B.V.). (1994).
- [60] Gentilucci, J., Toni, I., Chieffi, S., and Pavesi, G., The role of proprioception in the control of prehension movements: a kinematic study in a peripherally deafferented patient and in normal subjects. *Experimental Brain Research* 99, 483-500. (1994).
- [61] Connolly, J.D., Goodale, M.A., The role of visual feedback of the hand position in the control of manual prehension. *Experimental Brain Research*, 125, 281-286. (1999).
- [62] Lovelace, E.A., Vision and kinesthesia in accuracy of hand movement. *Perceptual and Motor Skills*, 68(3), 07-714. (1989).
- [63] Goodale, M.A., Jakobson, L.S., and Keillor, J.M., Differences in the visual control of pantomimed and natural grasping movements. *Neuropsychologia*, 32, 1159-1178. (1994).
- [64] Opitz, D., Gegenfurtner, K.R., and Bulthoff, H.H., A comparison of grasping real and virtual objects. *Proceedings 19th European Conference on Visual Perception*, Strasbourg, France, September 9-13. (1996).
- [65] Westling, G., and Johansson, R., Responses in glabrous skin mechanoreceptors during precision grip in humans. *Experimental Brain Research*, 66, 128-140. (1987).
- [66] Bejczy, A.K., Sensors, controls, and man-machine interface for advanced teleoperation. *Science*, 208(4450), 1327-1335. (1980).
- [67] Howe, R., and Kontarinis, D., Task Performance with a Dextrous Teleoperated Hand System. *Proceedings of SPIE*, Vol 1833, Boston, MA, 199-207. November, (1992).
- [68] Gordon, A.M., Forssberg, H., Johansson, R.S., and Westling, G., Visual size cues in the programming of manipulative forces during precision grip. *Experimental Brain Research*, 83, 477-482. (1991).

- [69] Gordon, A.M. Forssberg, H., Johansson, R.S., and Westling, G., The integration of haptically acquired size information in the programming of precision grip. *Experimental Brain Research*, 83, 483-488. (1991).
- [70] Marteniuk, R., MacKenzie, C., Jeannerod, M., Athenes, S., and Dugas, C., Constraints on human arm movement trajectories. *Canadian Journal of Psychology*, 41, 365-378. (1987).
- [71] Gentilucci, M., Chieffi, S., Scarpa, M., and Castiello, U., Temporal coupling between transport and grasp components during prehension movements: Effects of visual perturbation. *Behavioral Brain Research*, 47(1), 71-82. (1991).
- [72] Lederman, S.J., and Klatzky, R.L., Hand Movements: A Window into Haptic Object Recognition. *Cognitive Psychology* 19, 342-368. (1987).
- [73] McRuer, D.T., and Jex, H.R., A Review of Quasi-Linear Pilot Models. *IEEE Transactions Human Factors in Electronics, HFE-8*, 3, 231-238. (1967).
- [74] Servos, P., Goodale, M.A., and Jakobson, L.S., The Role of Binocular Vision in Prehension: a Kinematic Analysis. *Vision Research*, Vol. 32, No. 8, 1513-1521. (1992).
- [75] Servos, P., and Goodale, M.A., Binocular vision and the on-line control of human prehension. *Experimental Brain Research*, 98, 119-127. (1994).
- [76] Servos, P., and Goodale, M.A., Monocular and binocular control of human interceptive movements. *Experimental Brain Research*, 119, 92-102. (1998).
- [77] Dijkerman, C.H., and Milner, D.A., The perception and prehension of objects oriented in the depth plane. *Experimental Brain Research*, 118, 408-414. (1998).
- [78] Mon-Williams, M., and Dijkerman, C.H., The use of vergence information in the programming of prehension. *Experimental Brain Research*, 128, 578-582. (1999).
- [79] Marotta, J.J., Kruyer, A., and Goodale, M.A., The role of head movements in the control of manual prehension. *Experimental Brain Research*, 120, 134-138. (1998).
- [80] Marotta, J.J., Perrot, T.S., Nicolle, D., Servos, P., and Goodale, M.A., Adapting to monocular vision: grasping with one eye. *Experimental Brain Research* 104, 107-114. (1995).
- [81] Marotta, J.J., Goodale, M.A., The role of learned pictorial cues in the programming and control of grasping. *Experimental Brain Research*, 121, 465-470. (1998).
- [82] Gauthier, G.M., Vercher, J.-L., Mussa Ivaldi, F., and Marchetti, E., Oculo-manual tracking of visual targets: control learning, coordination control and coordination model. *Exp. Brain Res.* 73: pp. 127-137. (1988).

- [83] McCullough, M., *Abstracting Craft, The Practiced Digital Hand*. Cambridge: MIT Press. (1996).
- [84] From The Sculptor Speaks, *The Listener*. (London), XVIII, 449, 18 August. (1937).
- [85] Lee, J., Su, V., Ren, S., Hsiao, J., Hongladaromp, R., and Ishii, H., HandSCAPE. *Conference Abstracts and Applications of SIGGRAPH '99, Emerging Technologies*. Los Angeles, California, USA. ACM Press, 168. August 8-13, (1999).
- [86] Frei, P., Su, V., and Ishii, H., curlybot. *Conference Abstracts and Applications of SIGGRAPH '99, Emerging Technologies*. Los Angeles, California, USA. ACM Press, 173. August 8-13, (1999).
- [87] Underkoffler, J., and Ishii, H., Urp: A Luminous-Tangible Workbench for Urban Planning and Design. *Proceedings of Conference on Human Factors in Computing Systems (CHI '99)*, Pittsburgh, Pennsylvania, USA. ACM Press, 386-393. May 15-20, (1999).
- [88] Ullmer, B., and Ishii, H., mediaBlocks: Tangible Interfaces for Online Media (video). *Extended Abstracts of Conference on Human Factors in Computing Systems (CHI99)*, Pittsburgh, Pennsylvania, USA. ACM Press, 31-32. May 15-20, (1999).
- [89] Underkoffler, J., and Ishii, H., Illuminating Light: A Casual Optics Workbench (video). *Extended Abstracts of Conference on Human Factors in Computing Systems (CHI'99)*, Pittsburgh, Pennsylvania, USA. ACM Press, 5-6. May 15-20, (1999).
- [90] Underkoffler, J. and Ishii, H., Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface. *Proceedings of Conference on Human Factors in Computing Systems (CHI '98)*, Los Angeles. ACM Press, 542-549. April, (1998).
- [91] Underkoffler, J., Chak, D., Santos, G., and Ishii, H., The Luminous Room: Some Of It, Anyway. *Conference Abstracts and Applications of SIGGRAPH '99, Emerging Technologies*, Los Angeles, California, USA. ACM Press, 167. August 8-13, (1999).
- [92] Roscoe, S.N., The eyes prefer real images. In S.R. Ellis (Ed.), *Pictorial Communication in virtual and real environments*. New York: Taylor & Francis, 577-585. (1991).
- [93] Ellis, S.R., and Bucher, U.J., The relationship of binocular convergence and errors in judged distance to virtual objects. *Proceedings of the International Federation of Automatic Control*, Boston, MA. June 27-29, (1995).
- [94] Sparacino, F., et al., City of News. *Ars Electronia Festival*, Linz Austria, Sept. 8-13, (1997).
- [95] Sparacino, F., Davenport, G., and Pentland, A., Wearable Cinema/Wearable City: bridging physical and virtual spaces through wearable computing. *IMAGINA 2000*, Montecarlo, Jan 31st - Feb 3rd, (2000).

- [96] Spence, J. D., *The Memory Palace of Matteo Ricci*. New York: Viking. (1984).
- [97] Ishii, H. and Ullmer, B., Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. *Proceedings of Conference on Human Factors in Computing Systems (CHI '97)*, ACM Press, 234-241. (1997).
- [98] Ishii, H., Fletcher, R., Lee, J., Choo, S., Berzowska, J., Wisneski, C., Cano, C., Hernandez, A., and Bulthaup, C., musicBottles. *Conference Abstracts and Applications of SIGGRAPH '99, Emerging Technologies*, Los Angeles, California, USA. ACM Press, 174. August 8-13, (1999).
- [99] Burke, E., *A Philosophical Enquiry into the Origin of our Ideas of the Sublime and the Beautiful*, London. (1757).
- [100] Bergdoll, B., *European Architecture 1750-1890*. New York: Oxford University Press. (2000).
- [101] Wloka, M., Lag in multiprocessor virtual reality. *Presence*, 1(4), 50-63. (1995).
- [102] Hettinger L.J., an Riccio G.E., Visually Induced Motion Sickness in Virtual Environments. *Presence*, 1(3), 306-310. (1992).
- [103] Ware, C., and Balakrishnan, R., Reaching for objects in VR displays: lag and frame rate. *ACM Transactions of Computer-Human Interaction*, Vol. 1, No. 4, 331-356. (1994).
- [104] Richard, P., Birebent, G., Coiffet, P., Burdea, G., Gomez, D., and Langrana, N., Effect of Frame Rate and Force Feedback on Virtual Object Manipulation. *Presence*, Vol. 5, No. 1, 95-108. Winter, (1996).
- [105] Liang, J., Shaw, C., Green, M., On temporal-spatial realism in the virtual reality environment. *Proceedings of the ACM SIGGRAPH and SIGCHI Symposium on User Interface Software and Technology (UIST'91)*, 19-25. (1991).
- [106] Hall, E.T., *The Hidden Dimension*, Doubleday, New York. (1966).
- Chapter 3: Putting visual display in the manual workspace**
- [107] Fitzmaurice, G., Ishii, H., and Buxton, W., Bricks: Laying the Foundations for Graspable User Interfaces. *Proceedings of Conference on Human Factors in Computing Systems (CHI '95)*, ACM, Denver, 442-449. (1995).
- [108] Wellner, P., Mackay, W., and Gold, R., Computer Augmented Environments: Back to the Real World. *CACM*, Vol. 36, No. 7. (1993).
- [109] Ishii, H. and Ullmer, B., Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. (*CHI '97*), ACM, Atlanta, 234-241. March, (1997).

- [110] Ullmer, B. and Ishii, H., The metaDESK: Models and Prototypes for Tangible User Interfaces. *Proceedings of User Interface Software Technology (UIST '97)*, ACM Press, (1997).
- [111] Underkoffler, J. and Ishii, H., Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface. *Proceedings of Conference on Human Factors in Computing Systems*, ACM Press, 542-549. (1998).
- [112] Underkoffler, J., and Ishii, H., Urp: A Luminous-Tangible Workbench for Urban Planning and Design. *Proceedings of Conference on Human Factors in Computing Systems (CHI '99)*, Pittsburgh, Pennsylvania, USA. ACM Press, 386-393. (1999).
- [113] Yokokohji, Y., Hollis, R.L., and Kanade, T., Vision-based Visual/Haptic Registration for WYSIWYF Display. *International Conference on Intelligent Robots and Systems*, 1386-1393. (1996).
- [114] Yokokohji, Y., Hollis, R.L., and Kanade, T., WYSIWYF Display: A Visual/Haptic Interface to Virtual Environments. *Presence*, Vol. 8, No. 4. (1999).
- [115] Deering, M., High resolution virtual reality. *Proceedings of SIGGRAPH'92*, 195-202. (1992).
- [116] Taylor, R.M., Robinett W., Chi, V.L., Brooks, Jr., F.P., Wright, W.V., Williams, R.S., and Snyder, E.J., The Nanomanipulator: A Virtual-Reality Interface for a Scanning Tunneling Microscope. *Computer Graphics: Proceedings of SIGGRAPH '93*. (1993).
- [117] Schmandt, C., Spatial Input/Display Correspondence in a Stereoscopic Computer Graphic Work Station. *Computer Graphics*, Vol. 17, No. 3. (1983).
- [118] Durlach, N.I., von Wiegand, T.E., Zeltzer, D., Srinivasan, M., Salisbury, K., Brock, D., Sachtler, W.L., Pfautz, J., Schloerb, D., and Lathan, C., *Virtual environment technology for training (VETT): Annual report for MIT work performed during year 2*. Office of Naval Research, Contract N61339-94-C-0087. (1995).
- [119] von Wiegand, T.E., Schloerb, D.W., and Sachtler, W.L., Virtual Workbench: Near-Field Virtual Environment System with Applications. *Presence*, Vol. 8, No. 5. (1999).
- [120] Billinghurst, M., Kato, H., May, R., and Kraus, S., Shared Space. *Conference Abstracts and Applications: Emerging Technologies, SIGGRAPH'99*, ACM Press. (1999).
- [121] Krueger, M., *Artificial Reality II*. Addison-Wesley. (1991).
- [122] Maes, P., Darrell, T., Blumberg, B., and Pentland, A., The ALIVE System: Wireless, Full-Body Interaction with Autonomous Agents, *ACM Special Issue on Multimedia and Multisensory Virtual Worlds*. (1996).
- [123] Jones, M.R.E., The Haptic Hologram, *Proceedings of SPIE, Fifth International Symposium on Display Holography*, Vol. 2333, 444-447. (1994).

Chapter 4. Computational Holography

- [124] Gabor, D., A New Microscopic Principle. *Nature*, 161, 777-8. (1948).
- [125] Denisyuk, Yu. N., Photographic reconstruction of the optical properties of an object in its own scattered radiation field. *Soviet Physics - Doklady*, 7, 543-5. (1962).
- [126] Leith, E.N., Upatnieks, J., Wavefront Reconstruction with Continuous Tone Objects. *J. Opt. Soc. Am.* 53, 1377. (1963).
- [127] Brown, B.R., and Lohmann, A.W., Complex spatial filtering with binary masks. *Applied Optics* 5, 967. (1966).
- [128] Brown, B.R., and Lohmann, A.W., Computer Generated Binary Holograms. *IBM Journal of Research and Development*, 13, 160-167. (1969).
- [129] Lee, W.H., Sampled Fourier Transform Hologram Generated by Computer. *Applied Optics*, 9, 1949. (1970).
- [130] Burckhardt, C.B., A Simplification of Lee's Method of Generating Holograms by Computer. *Applied Optics*, 9, 1949. (1970).
- [131] Lesem, L.B., Hirsch, P.M., and Jordan Jr., J.A., The Kinoform: A New Wavefront Reconstruction Device. *IBM Journal of Research and Development*, 13, 150-155. (1969).
- [132] Chu, D.C., Fienup, J.R., and Goodman, J.W., Multi-emulsion, On-axis, Computer Generated Hologram. *Applied Optics*, 12, 1386-8. (1973).
- [133] Kollin, J.S., Benton, S.A., and Jepsen, M.L., Real-Time Display of 3-D Computed Holograms by Scanning the Image of an Acousto-Optic Modulator. In G.M. Morris, (Ed.), *SPIE Proc. Vol. 1136, Holographic Optics II: Principles and Applications*. (1989).
- [134] St.-Hilaire, P., Benton, S.A., Lucente, M., Jepsen, M.L., Kollin, J., Yoshikawa, H., and Underkoffler, J.S., Electronic display system for computational holography. *SPIE Vol. 1212, Practical Holography IV*. (1990).
- [135] Underkoffler, J.S., *Development of Parallel Processing Algorithms for Real-Time Computed Holography*. SB Thesis, Department of Electric Engineering and Computer Science, Massachusetts Institute of Technology. (1988).
- [136] Underkoffler, J.S., *Toward Accurate Computation of Optically Reconstructed Holograms*. SM Thesis, Media Arts and Sciences Section, Massachusetts Institute of Technology. (1991).
- [137] Lucente, M., Interactive Computation of Holograms using a Look-up Table. *Journal of Electronic Imaging*, Vol. 2, No. 1. (1993).

- [138] Yoshikawa, H., Iwase, S., and Oneda, T., Fast Computation of Fresnel Holograms employing Difference. *SPIE Proceedings Vol. 3956, Practical Holography XIV and Holographic Materials VI*. (2000).
- [139] Pappu, R.S., Nonuniformly sampled computer-generated holograms. *Optical Engineering*. (1996).
- [140] Lucente, M., Holographic bandwidth compression using spatial subsampling. *Optical Engineering*, Vol. 35, No. 6, 1529-1537. (1996).
- [141] Benton, S.A., Survey of holographic stereograms. *SPIE Vol 367. Processing and Display of Three-Dimensional Data*, Bellingham, WA. (1983).
- [142] Halle, M.W., *The Generalized Holographic Stereogram*. SM Thesis, program in Media Arts and Sciences, Massachusetts Institute of Technology. (1991).
- [143] Halle, M.W., Holographic stereograms as discrete imaging systems. In S.A. Benton, (Ed.), *SPIE Proc. Vol 2176: Practical Holography VIII*. (1994).
- [144] Halle, M.W., Autostereoscopic Displays and Computer Graphics. *Computer Graphics* (A publication of ACM Siggraph), Vol. 31, No. 2. (1997).
- [145] Pappu, R.S., Sparrell, C.J., Underkoffler, J.S., Kropp, A., Chen, B., and Plesniak, W., A Generalized Pipeline for Preview and Rendering of Synthetic Holograms. *Proceedings of the SPIE, Practical Holography XI*. (1997).
- [146] Tricoles, G., Computer generated holograms: an historical review. *Applied Optics*, Vol. 26, No. 20, 4351-4360. (1987).
- [147] Waters, J.P., Three-dimensional Fourier Transform Method for Synthesizing binary Holograms. *Journal of the Optical Society of America*, 58, 1284-8. (1968).
- [148] Leseberg, D., and Bryngdahl, E., Methods of Digital Holography. *Applied Optics*, Vol. 23, No. 14. (1984).
- [149] Leseberg, D., Frere, Computer-generated holograms of 3-D objects composed of tilted planar segments. *Applied Optics*, Vol. 27, No. 14. (1988).
- [150] Leseberg, D., Computer-generated three-dimensional image holograms. *Applied Optics*, Vol. 31, No. 2. (1992).
- [151] Underkoffler, J.S., Occlusion Processing and Smooth Surface Shading for Fully Computed Synthetic Holography. In S.A. Benton, (Ed.), *Proceedings of the IS&T/SPIE's Symposium on Electronic Imaging, Practical Holography XI*. (1997).
- [152] Burch, J.J., A Computer Algorithm for Synthesis of Spatial Frequency Filters. *Proc. IEEE* 55, 599. (1967).

- [153] Lucente, M., Optimization of Hologram Computation for Real-Time Display. *SPIE Proceeding #1667 Practical Holography VI*. (1992).
- [154] Watlington, J., Lucente, M., Sparrell, C.J., Bove, V.M. Jr., and Tamitani, I., A hardware architecture for rapid generation of electro-holographic fringe patterns. *SPIE Proceedings, Vol. 2406, Practical Holography IX*, 2406-23. (1995).
- [155] Lucente, M., and Galyean, T., Rendering Interactive Holographic Images. *Proceedings of the ACM SIGGRAPH95*. (1995).
- [156] Lucente, M., *Diffraction-Specific Fringe Computation for Electro-Holography*. Ph.D Thesis, Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology. (1994).
- [157] St.-Hilaire, P., Benton, S.A., Lucente, M., and Hubel, P.M., Color images with the MIT holographic video display. In S.A. Benton, (Ed.), *SPIE Vol 1667, Practical Holography VI*. (1992).
- [158] Halle, M.W., Benton, S.A., Klug, M.A., and Underkoffler, J.S., The Ultragram: a generalized holographic stereogram. In S.A. Benton, (Ed.), *SPIE Vol. 1461, Practical Holography V*, 142-155. (1991).
- [159] Pappu, R., and Plesniak, W., Haptic interaction with holographic video images. *Proceedings of the IS&T/SPIE's Symposium on Electronic Imaging, Practical Holography XII*. (1998).
- [160] Plesniak, W., and Pappu, R., Coincident display using haptics and holographic video. *Proceedings of ACM SIGCHI Conference on Human Factors in Computing Systems*. (1998).
- [161] St.-Hilaire, P., *Scalable Optical Architectures for Electronic Holography*. Ph.D Thesis, MIT Program in Media Arts and Sciences, Massachusetts Institute of Technology. (1994).
- [162] Croll, B.M., *Rendermatic: An Implementation of the Rendering Pipeline*. SM Thesis, Department of Architecture, Massachusetts Institute of Technology. (1986).
- [163] Foley, J.D., et al., *Computer Graphics: Principles and Practice*. Addison-Wesley. (1990).
- [164] St.-Hilaire, P., Lucente, M., Sutter, J.D., Pappu, R.S., Sparrell, C.J., and Benton, S.A., Scaling up the MIT holographic video system. *Proceedings of the Fifth International Symposium on Display Holography*, Lake Forest College. (1994).
- [165] Bove, V.M. Jr., and Watlington, J., Cheops: A data-flow processor for real-time video processing. MIT Media Laboratory Technical Memo. (1993).

[166] Oppenheim, A.V., and Schafer, R.W., *Discrete-time Signal Processing*. Prentice Hall. (1989).

[167] Pappu, R.S., personal communication, November 2000.

[168] Beckmann, P., and Spizzichino, A., *The scattering of Electromagnetic Waves from Rough Surfaces*. Artech House, Inc., Norwood, MA. (1987).

Chapter 5. Touch: two static haptic holograms

[169] Benton, S.A., Birner, S.M, and Shirakura, A., Edge-Lit Rainbow Holograms. In S.A. Benton, (Ed.), *SPIE Proc. Vol. #1212, Practical Holography IV*. 149-157. (1990).

[170] Halle, M., and Kropp, A., Fast computer graphics rendering for full parallax spatial displays. In S.A. Benton (Ed.), *Proceedings of the IS&T/SPIE's Symposium on Electronic Imaging, Practical Holography XI*. (1997).

[171] Plesniak, W., and Klug, M., Tangible holography: adding synthetic touch to 3D display. In S.A. Benton, (Ed.), *SPIE Practical Holography XI*, San Jose, CA. (1997).

Chapter 6. Lathe: a haptic hologram with limited update

[172] St.-Hilaire, P., *Scalable Optical Architectures for Electronic Holography*. PhD Thesis, MIT Program in Media Arts and Sciences, Massachusetts Institute of Technology. (1994).

[173] Pappu, R., and Plesniak, W., Haptic interaction with holographic video images. *Proceedings of the IS&T/SPIE Symposium on Electronic Imaging, Practical Holography XII*. (1998).

[174] Plesniak, W., and Pappu, R., Coincident display using haptics and holographic video. *Proceedings of Conference on Human Factors in Computing Systems (CHI'98)*, ACM. (1998).

[175] Plesniak, W., and Pappu, R., Spatial interaction with haptic holograms. *Proceedings of the IEEE International Conference on Multimedia Computing and Systems, (ICMCS'99)*. (1999).

[176] Plesniak, W., Pappu, R., and Benton, S.A., (invited, in press). In C.J. Kuo, (Ed.), *3D Holographic Imaging*, Wiley-Interscience.

Chapter 8. Discussion and larger view

[177] Loyall, A.B., *Believable Agents: Building Interactive Personalities*. Ph.D. Thesis. Technical Report CMU-CS-97-123, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA. (1997).

[178] Drascic, D., and Milgram, P., Perceptual issues in Augmented Reality. *SPIE Volume 2653: Stereoscopic Displays and Virtual Reality Systems III*, 123-134. (1996).

[179] Personal communication from Jodi Whitsel, Bluesky Studios, December 1999.

- [180] Fournier, A., Gunawan, A., and Romanzin, C., Common illumination between real and computer generated scenes. *Proc. Graphics Interface '93*, 254-262. (1993).
- [181] Ikeuchi, K., Sat, Y., Nishino, K., and Sato, I., Photometric Modeling for Mixed Reality. In Y. Ohta, and H. Tamura, (Eds.), *Mixed Reality: Merging Virtual and Real Worlds*. Ohmsha & Springer-Verlag. (1999).
- [182] Ramachandran, V.S., Perceiving Shape from Shading. *Scientific American* 127-138. (1988).
- [183] Adelson, E.H., Perceptual organization and the judgement of brightness. *Science* 262, 2042-2044. (1993).
- [184] Bloj, M.G., and Hurlbert, A.C., The influence of shape and illuminant position cues on surface lightness perception. *Invest. Ophthalmol. Vis. Sci.* 40, S981. (1999).
- [185] Bloj, M., Kersten, D., and Hurlbert, A.C., 3D Shape Perception Influences Color Perception Via Mutual Illumination. *Nature*. (1999).
- [186] Goral, C.M., Fredrickson, B., and Baeverstad, H.L.Jr., Modeling the Interaction of Light Between Diffuse Surfaces. *Siggraph '84*, 213-222. (1984).